

AIX เวอร์ชัน 7.2

ข้อมูลอ้างอิงคำสั่ง วอลุ่ม 2, d - h

**IBM**



AIX เวอร์ชัน 7.2

ข้อมูลอ้างอิงคำสั่ง วอลุ่ม 2, d - h

**IBM**

หมายเหตุ  
ก่อนที่คุณจะใช้ข้อมูลนี้และผลิตภัณฑ์ที่สนับสนุน โปรดอ่านข้อมูลใน “คำประกาศ” ในหน้า 823

This edition applies to AIX Version 7.2 and to all subsequent releases and modifications until otherwise indicated in new editions.

© ลิขสิทธิ์ของ IBM Corporation 2015, 2016.

© Copyright IBM Corporation 2015, 2016.

# สารบัญ

เกี่ยวกับเอกสารนี้ . . . . .	vii	คำสั่ง diff . . . . .	150
Highlighting . . . . .	vii	คำสั่ง diff3 . . . . .	154
Case sensitivity in AIX . . . . .	vii	คำสั่ง diffmk . . . . .	156
ISO 9000 . . . . .	vii	คำสั่ง dig . . . . .	157
Support for the single UNIX specification . . . . .	vii	คำสั่ง digest . . . . .	163
<b>d.</b> . . . . .	<b>1</b>	คำสั่ง dircmp . . . . .	164
คำสั่ง dacinet. . . . .	1	คำสั่ง dirname . . . . .	166
คำสั่ง dadmin . . . . .	3	คำสั่ง disable . . . . .	167
คำสั่ง date . . . . .	5	คำสั่ง diskusg. . . . .	169
คำสั่ง dbts . . . . .	8	คำสั่ง dispgid . . . . .	171
คำสั่ง dbx. . . . .	9	คำสั่ง dispuid . . . . .	172
คำสั่ง dc. . . . .	81	คำสั่ง dist . . . . .	174
คำสั่ง dcp . . . . .	83	คำสั่ง dmpuncompress . . . . .	177
คำสั่ง dd . . . . .	89	คำสั่ง dnssec-keygen . . . . .	178
เมธอด defif . . . . .	94	คำสั่ง dnssec-makekeyset . . . . .	180
เมธอด definet . . . . .	96	คำสั่ง dnssec-signkey . . . . .	181
คำสั่ง defragfs . . . . .	96	คำสั่ง dnssec-signzone . . . . .	183
คำสั่ง defvsvd . . . . .	99	คำสั่ง dodisk . . . . .	184
คำสั่ง deleteX11input . . . . .	102	คำสั่ง domainname . . . . .	186
คำสั่ง delta . . . . .	103	คำสั่ง domlist. . . . .	187
คำสั่ง deroff . . . . .	105	คำสั่ง dosdel . . . . .	188
คำสั่ง detachrset . . . . .	106	คำสั่ง dosdir . . . . .	189
คำสั่ง devinstall . . . . .	108	คำสั่ง dosformat . . . . .	190
คำสั่ง devnm . . . . .	109	คำสั่ง dosread . . . . .	192
คำสั่ง devrsrv. . . . .	110	คำสั่ง doswrite . . . . .	194
คำสั่ง df . . . . .	118	คำสั่ง dp . . . . .	195
คำสั่ง dfmounts . . . . .	124	dpid2 Daemon . . . . .	196
คำสั่ง dfpd . . . . .	126	คำสั่ง dping . . . . .	198
คำสั่ง dfscck . . . . .	127	คำสั่ง drmgr . . . . .	200
คำสั่ง dfshares . . . . .	129	คำสั่ง drslot . . . . .	201
คำสั่ง dhcraction. . . . .	130	คำสั่ง dscrctl . . . . .	204
dhcpcd Daemon . . . . .	132	คำสั่ง dscreen. . . . .	205
dhcpcd6 Daemon . . . . .	134	คำสั่ง dshbak . . . . .	206
dhcprd Daemon . . . . .	135	คำสั่ง dsh . . . . .	208
คำสั่ง dhcpsconf . . . . .	137	คำสั่ง dsllpaccept . . . . .	216
dhcpsd Daemon . . . . .	139	คำสั่ง dsllpaccess . . . . .	217
dhcpsdv6 Daemon . . . . .	140	คำสั่ง dsllpadmin . . . . .	218
คำสั่ง diag. . . . .	142	คำสั่ง dsllpdisable . . . . .	222
คำสั่ง diaggetrto . . . . .	145	คำสั่ง dsllpenable . . . . .	223
คำสั่ง diagrpt . . . . .	147	คำสั่ง dsllpprotocol . . . . .	224
คำสั่ง diagsetrto . . . . .	148	คำสั่ง dsllpreject . . . . .	226
คำสั่ง diction . . . . .	149	คำสั่ง dsllpsearch . . . . .	227
		คำสั่ง dspcat . . . . .	229

คำสั่ง dspmsg . . . . .	230
คำสั่ง dtaction . . . . .	232
คำสั่ง dtappintegrate . . . . .	234
คำสั่ง dtlogin . . . . .	236
คำสั่ง dtscript . . . . .	264
คำสั่ง dtsession . . . . .	265
คำสั่ง dtterm . . . . .	274
คำสั่ง du . . . . .	281
คำสั่ง dump . . . . .	283
คำสั่ง dumpcheck . . . . .	285
คำสั่ง dumpctrl . . . . .	287
คำสั่ง dumpfs . . . . .	293

**e . . . . . 295**

คำสั่ง echo . . . . .	295
คำสั่ง ed หรือ red . . . . .	297
คำสั่ง edit . . . . .	336
คำสั่ง edquota . . . . .	344
คำสั่ง efsenable . . . . .	347
คำสั่ง efskeymgr . . . . .	349
คำสั่ง efskstoldif . . . . .	353
คำสั่ง efsmgr . . . . .	355
คำสั่ง egrep . . . . .	357
คำสั่ง eimadmin . . . . .	360
คำสั่ง elogevent . . . . .	369
คำสั่ง emgr . . . . .	371
คำสั่ง emstat . . . . .	377
คำสั่ง emsvcsctrl . . . . .	379
คำสั่ง enable . . . . .	382
คำสั่ง enotifyevent Command, notifyevent . . . . .	384
คำสั่ง enq . . . . .	386
คำสั่ง enroll . . . . .	395
คำสั่ง encrypt . . . . .	396
คำสั่ง entstat . . . . .	403
คำสั่ง env . . . . .	408
คำสั่ง epkg . . . . .	410
คำสั่ง eqn . . . . .	419
คำสั่ง errclear . . . . .	421
คำสั่ง errctrl . . . . .	423
คำสั่ง errdead . . . . .	428
errdemon Daemon . . . . .	429
คำสั่ง errinstall . . . . .	432
คำสั่ง errlogger . . . . .	435
คำสั่ง errmsg . . . . .	436
คำสั่ง errpt . . . . .	438
คำสั่ง errstop . . . . .	444
คำสั่ง errupdate . . . . .	445

คำสั่ง ethchan_config . . . . .	452
คำสั่ง ewallevent . . . . .	454
คำสั่ง ex . . . . .	456
คำสั่ง execerror . . . . .	458
คำสั่ง execrset . . . . .	458
คำสั่ง expand . . . . .	460
คำสั่ง expfilt . . . . .	462
คำสั่ง explain . . . . .	463
คำสั่ง explore . . . . .	463
คำสั่ง exportfs . . . . .	465
คำสั่ง exportvg . . . . .	472
คำสั่ง expr . . . . .	474
คำสั่ง exptun . . . . .	478
คำสั่ง extendlv . . . . .	479
คำสั่ง extendvg . . . . .	482

**f . . . . . 485**

คำสั่ง f . . . . .	485
คำสั่ง factor . . . . .	487
คำสั่ง true หรือ false . . . . .	488
คำสั่ง reboot หรือ fastboot . . . . .	489
คำสั่ง fc . . . . .	490
คำสั่ง fccheck . . . . .	493
คำสั่ง fcclear . . . . .	495
คำสั่ง fcdecode . . . . .	498
คำสั่ง fcdispfid . . . . .	500
คำสั่ง fcfilter . . . . .	501
คำสั่ง fcinit . . . . .	503
คำสั่ง fclogerr . . . . .	507
คำสั่ง fcpushstk . . . . .	515
คำสั่ง fcreport . . . . .	521
คำสั่ง fcstat . . . . .	523
คำสั่ง fcstkrpt . . . . .	526
คำสั่ง fcteststk . . . . .	528
คำสั่ง fddistat . . . . .	531
คำสั่ง fdformat . . . . .	534
คำสั่ง fdpr . . . . .	535
คำสั่ง fencevsd . . . . .	541
คำสั่ง ff . . . . .	543
คำสั่ง fg . . . . .	545
คำสั่ง fgrep . . . . .	546
คำสั่ง file . . . . .	549
คำสั่ง filemon . . . . .	551
คำสั่ง fileplace . . . . .	566
คำสั่ง find . . . . .	568
คำสั่ง finger . . . . .	577
fingerd Daemon . . . . .	580

คำสั่ง fish . . . . .	582
คำสั่ง flcopy . . . . .	583
คำสั่ง flush-secldapclntd . . . . .	584
คำสั่ง fmt . . . . .	585
คำสั่ง fold . . . . .	586
คำสั่ง folder . . . . .	588
คำสั่ง folders . . . . .	591
คำสั่ง forcercpoffline . . . . .	593
คำสั่ง format . . . . .	595
คำสั่ง fortune . . . . .	597
คำสั่ง forw . . . . .	598
คำสั่ง fpm . . . . .	602
คำสั่ง asa, fpr . . . . .	605
คำสั่ง frctr . . . . .	606
คำสั่ง from . . . . .	609
คำสั่ง fsck . . . . .	610
คำสั่ง fsck_cacheofs . . . . .	615
คำสั่ง fsdb . . . . .	615
คำสั่ง fsplit . . . . .	629
คำสั่ง ftp . . . . .	630
ftpd Daemon . . . . .	643
คำสั่ง fuser . . . . .	651
คำสั่ง fwtmp . . . . .	653
คำสั่ง fxfer . . . . .	654

**g . . . . . 667**

gated Daemon . . . . .	667
คำสั่ง gdc . . . . .	670
คำสั่ง gencat . . . . .	673
คำสั่ง gencopy . . . . .	674
คำสั่ง gencore . . . . .	675
คำสั่ง genfilt . . . . .	676
คำสั่ง geninstall . . . . .	678
คำสั่ง genkex . . . . .	682
คำสั่ง genkld . . . . .	683
คำสั่ง genld . . . . .	684
คำสั่ง gennames . . . . .	685
คำสั่ง gensyms . . . . .	686
คำสั่ง gentun . . . . .	688
คำสั่ง genxlt . . . . .	690
คำสั่ง get . . . . .	692
คำสั่ง getconf . . . . .	701
คำสั่ง getdev . . . . .	709
คำสั่ง getdgrp . . . . .	711
คำสั่ง getea . . . . .	714
คำสั่ง getopt . . . . .	715
คำสั่ง getopt . . . . .	717

คำสั่ง getrunmode . . . . .	719
คำสั่ง getsecconf . . . . .	720
คำสั่ง getsyslab . . . . .	721
คำสั่ง gettable . . . . .	722
คำสั่ง gettrc . . . . .	723
คำสั่ง getty . . . . .	724
คำสั่ง gprof . . . . .	726
คำสั่ง grap . . . . .	731
คำสั่ง greek . . . . .	735
คำสั่ง grep . . . . .	736
กลุ่ม groups . . . . .	739
คำสั่ง grpck . . . . .	740
คำสั่ง grpsvcctrl . . . . .	743
gssd Daemon . . . . .	746

**h . . . . . 749**

คำสั่ง ha.vsd . . . . .	749
คำสั่ง ha_star . . . . .	753
คำสั่ง ha_vsd . . . . .	754
haemd Daemon . . . . .	755
คำสั่ง haemd_HACMP . . . . .	756
คำสั่ง haemqvar . . . . .	757
คำสั่ง haemtrcoff . . . . .	761
คำสั่ง haemtrcon . . . . .	764
คำสั่ง haemunkrm . . . . .	766
hagsd Daemon . . . . .	768
คำสั่ง hagsns . . . . .	771
คำสั่ง hagsvote . . . . .	773
คำสั่ง halt หรือ fasthalt . . . . .	776
คำสั่ง hangman . . . . .	778
คำสั่ง hash . . . . .	779
คำสั่ง hatsoptions . . . . .	781
คำสั่ง head . . . . .	783
คำสั่ง help . . . . .	784
คำสั่ง hfistat . . . . .	785
คำสั่ง hmcauth . . . . .	790
คำสั่ง host . . . . .	792
คำสั่ง host9 . . . . .	794
คำสั่ง hostent . . . . .	796
คำสั่ง hostid . . . . .	798
hostmibd Daemon . . . . .	799
คำสั่ง hostname . . . . .	802
คำสั่ง hosts2ldif . . . . .	803
คำสั่ง hp . . . . .	804
คำสั่ง hplj . . . . .	805
คำสั่ง hpmcount . . . . .	806
คำสั่ง hpmstat . . . . .	812

คำสั่ง hps_dump . . . . .	817
คำสั่ง htable . . . . .	818
คำสั่ง hty_load . . . . .	819
คำสั่ง hyphen . . . . .	821

**คำประกาศ . . . . . 823**

สิ่งที่ต้องพิจารณาเกี่ยวกับนโยบายความเป็นส่วนตัว . . . . .	825
เครื่องหมายการค้า . . . . .	825

<b>ดัชนี . . . . .</b>	<b>827</b>
------------------------	------------

---

## เกี่ยวกับเอกสารนี้

เอกสารนี้ให้ผู้ใช้ชั้นปลายมีข้อมูลรายละเอียดโดยสมบูรณ์ เกี่ยวกับคำสั่งสำหรับระบบปฏิบัติการ AIX® คำสั่งจะแสดงไว้ตามลำดับตัวอักษร และตามหมวดหมู่ และมีคำอธิบายที่สมบูรณ์ซึ่งได้กำหนดไว้ให้กับคำสั่ง และแฟล็กที่มีอยู่ หากสามารถใช้งานได้ การแสดงรายการคำสั่งแต่ละคำสั่ง จะมีตัวอย่างประกอบ วอลุ่มนี้มีคำสั่ง AIX ที่เริ่มต้นด้วย อักษร d ถึง h เอกสารคู่มือนี้ยังมีอยู่ใน CD เอกสารคู่มือที่มาพร้อมกับระบบปฏิบัติการ

---

## Highlighting

The following highlighting conventions are used in this document:

<b>Bold</b>	Identifies commands, subroutines, keywords, files, structures, directories, and other items whose names are predefined by the system. Bold highlighting also identifies graphical objects, such as buttons, labels, and icons that the you select.
<i>Italics</i>	Identifies parameters for actual names or values that you supply.
Monospace	Identifies examples of specific data values, examples of text similar to what you might see displayed, examples of portions of program code similar to what you might write as a programmer, messages from the system, or text that you must type.

---

## Case sensitivity in AIX

Everything in the AIX operating system is case sensitive, which means that it distinguishes between uppercase and lowercase letters. For example, you can use the `ls` command to list files. If you type `LS`, the system responds that the command is not found. Likewise, `FILEA`, `FiLea`, and `filea` are three distinct file names, even if they reside in the same directory. To avoid causing undesirable actions to be performed, always ensure that you use the correct case.

---

## ISO 9000

ISO 9000 registered quality systems were used in the development and manufacturing of this product.

---

## Support for the single UNIX specification

The AIX operating system is designed to support The Open Group's Single UNIX Specification Version 3 (UNIX 03) for portability of operating systems based on the UNIX operating system. Many new interfaces, and some current ones, have been added or enhanced to meet this specification. To determine the correct way to develop a UNIX 03 portable application, see The Open Group's UNIX 03 specification on The UNIX System website (<http://www.unix.org>).



---

## d

คำสั่ง AIX ต่อไปนี้เริ่มต้นด้วยตัวอักษร d

---

### คำสั่ง **dacinet**

#### วัตถุประสงค์

การดูแลความปลอดภัยในพอร์ต TCP ในคอนฟิกูเรชัน CAPP/EAL4+

#### ไวยากรณ์

**dacinet acflush**

**dacinet aclclear** *Service* | *Port*

**dacinet acladd** *Service* | [-] *addr* [/ *prefix\_length*] [**u**:*user* | *uid* | **g**:*group* | *gid*]

**dacinet acldel** *Service* | [-] *addr* [/ *prefix\_length*] [**u**:*user* | *uid* | **g**:*group* | *gid*]

**dacinet aclls** *Service* | *Port*

**dacinet setpriv** *Service* | *Port*

**dacinet unsetpriv** *Service* | *Port*

**dacinet lspriv**

#### คำอธิบาย

คำสั่ง **dacinet** ถูกใช้เพื่อดูแลความปลอดภัยในพอร์ต TCP โปรดดูที่ส่วน คำสั่งย่อย สำหรับรายละเอียดเกี่ยวกับฟังก์ชันต่างๆ ของ **dacinet**

# คำสั่งย่อ

ไอเท็ม  
acladd

## คำอธิบาย

เพิ่มรายการ ACL เข้ากับตารางเคอร์เนลที่เก็บ รายการควบคุมการเข้าถึงที่ใช้โดยคำสั่ง **dacinet** ไวยากรณ์ของพารามิเตอร์สำหรับคำสั่งย่อ **acladd** มีดังนี้:

`[-]addr[/length][u:useruid|g:groupid]`พารามิเตอร์ ถูกกำหนดไว้ดังนี้:

**addr** ชื่อโฮสต์ DNS หรือ IPv4 (หรือ IPv6) แอดเดรส "-" นำหน้าแอดเดรสหมายถึง ว่ารายการ ACL ถูกใช้เพื่อปฏิเสธการเข้าถึง

**length** แสดงว่า **addr** จะถูกใช้เป็นเน็ตเวิร์กแอดเดรส ไม่ใช่โฮสต์แอดเดรส ด้วยบิต **length** แรกที่นำมาจาก **addr**

## u:useruid

ตัวระบุผู้ใช้เป็นทางเลือก ถ้าไม่มีการระบุ **uid** ผู้ใช้ทั้งหมดในโฮสต์ที่ระบุหรือ subnet ได้รับสิทธิเข้าถึง เซอร์วิส ถ้าระบุเฉพาะผู้ใช้ที่ระบุที่ได้รับสิทธิเข้าถึง

## g:groupid

ตัวระบุกลุ่มเป็นทางเลือก ถ้าไม่มีการระบุ **gid** ผู้ใช้ทั้งหมดในโฮสต์ที่ระบุหรือ subnet ได้รับสิทธิเข้าถึง เซอร์วิส ถ้าระบุเฉพาะกลุ่มที่ระบุที่ได้รับสิทธิเข้าถึง

acldclear  
acldel

ล้างค่า ACL สำหรับเซอร์วิสหรือพอร์ตที่ระบุ

ลบรายการ ACL ออกจากตารางเคอร์เนลที่เก็บ รายการควบคุมการเข้าถึงที่ใช้โดยคำสั่ง **dacinet** คำสั่งย่อ **dacinet acldel** ลบ รายการออกจาก ACL เฉพาะถ้าถูกเรียกใช้ด้วยพารามิเตอร์ที่ตรงกับ ที่ถูกใช้ในการเพิ่มรายการให้กับ ACL ไวยากรณ์ของพารามิเตอร์สำหรับคำสั่งย่อ **acldel** เป็นดังนี้:

`[-]addr[/length][u:useruid|g:groupid]`พารามิเตอร์ ถูกกำหนดไว้ดังนี้:

**addr** ชื่อโฮสต์ DNS หรือ IPv4 (หรือ IPv6) แอดเดรส "-" นำหน้าแอดเดรสหมายถึง ว่ารายการ ACL ถูกใช้เพื่อปฏิเสธการเข้าถึง

**length** แสดงว่า **addr** จะถูกใช้เป็นเน็ตเวิร์กแอดเดรส ไม่ใช่โฮสต์แอดเดรส ด้วยบิต **length** แรกที่นำมาจาก **addr**

## u:useruid

ตัวระบุผู้ใช้เป็นทางเลือก ถ้าไม่มีการระบุ **uid** ผู้ใช้ทั้งหมดในโฮสต์ที่ระบุหรือ subnet ได้รับสิทธิเข้าถึง เซอร์วิส ถ้าระบุเฉพาะผู้ใช้ที่ระบุที่ได้รับสิทธิเข้าถึง

## g:groupid

ตัวระบุกลุ่มเป็นทางเลือก ถ้าไม่มีการระบุ **gid** ผู้ใช้ทั้งหมดในโฮสต์ที่ระบุหรือ subnet ได้รับสิทธิเข้าถึง เซอร์วิส ถ้าระบุเฉพาะกลุ่มที่ระบุที่ได้รับสิทธิเข้าถึง

acldflush  
aclds

ลบค่า ACL ทั้งหมดที่กำหนดไว้ในระบบ ทำให้ พอร์ต TCP ทั้งหมดเข้าถึงไม่ได้ เมื่อมีการร้องขอการเชื่อมต่อ นอกจากจากผู้ใช้ root ในโฮสต์ และยังลบพอร์ตที่มี privilege เช่นกระบวนการที่สามารถเชื่อมกับ พอร์ตทั้งหมดที่สูงกว่า 1024

แสดงรายการ ACL สำหรับเซอร์วิสหรือพอร์ตที่ระบุ **dacinet aclds 0** จะแสดงรายการ ACL ดีฟอลต์สำหรับกระบวนการพิสูจน์ตัวตน จากมุมมองทางโลจิคัล ดีฟอลต์ ACL ถูกผนวกกับ ACL สำหรับเซอร์วิส หากไม่มีรายการบน ACL ที่ตรงกับผู้ใช้ที่พยายามเชื่อมต่อกับเซอร์วิส การเข้าถึงจะถูกปฏิเสธ หากมีรายการหนึ่งรายการหรือมากกว่า รายการแรกที่มี **usergroup@host|subnet** ที่ตรงกับ ผู้ร้องขอการเชื่อมต่อจะกำหนดความสามารถของผู้ใช้เพื่อเชื่อมต่อกับเซอร์วิส ดังนั้นจึงเป็นไปได้ที่จะปฏิเสธเซอร์วิสสำหรับสมาชิกของกลุ่มที่เข้าถึงเซอร์วิสเท่านั้น โดยการเพิ่มรายการ การปฏิเสธสำหรับสมาชิกดังกล่าวก่อนที่คุณจะเพิ่มรายการการอนุญาตสำหรับกลุ่ม

lspriv

แสดงรายการเซอร์วิสที่มี privilege ทั้งหมดหรือพอร์ตที่ไม่มี privilege ถาวร (คือแสดงเฉพาะเซอร์วิสที่มี privilege ที่มี หมายเลขพอร์ตมากกว่า 1024)

setpriv

ทำให้เซอร์วิสหรือพอร์ตที่ระบุมีสิทธิ์ เพื่อให้เฉพาะกระบวนการที่มีสิทธิ์ superuser เท่านั้นที่อาจเชื่อมกับพอร์ต และเสนอเซอร์วิสบนพอร์ตนั้น พอร์ตต่ำกว่า 1024 ถูกละเว้น เนื่องจากมี privilege แบบถาวร

unsetpriv

ทำให้เซอร์วิสหรือพอร์ตที่ระบุไม่มีสิทธิ์ ที่กระบวนการใดๆ อาจเชื่อมต่อกับ กระบวนการใดๆ อาจยังเชื่อมกับพอร์ตใดๆ ในช่วงของพอร์ตที่ใช้ชั่วคราวในปัจจุบัน โดยไม่คำนึงว่าพอร์ตนั้นจะถูกทำเครื่องหมายว่า เป็นพอร์ตที่มีสิทธิ์หรือไม่

# ไฟล์

---

## คำสั่ง dadmin

### วัตถุประสงค์

ใช้เพื่อเคียวรีและแก้ไขสถานะของเซิร์ฟเวอร์ DHCP

### ไวยากรณ์

```
dadmin [-?] [-v] [-h Hostname] [-n interval] [-f] -d IpAddress [-x] -i | [-x] -s | -t onloff|Value | -q IpAddress | -r  
IpAddress | -p IpAddress | -c ClientId
```

### คำอธิบาย

คำสั่ง dadmin จะเปิดใช้งานผู้ดูแลระบบ DHCP เพื่อเคียวรีและปรับเปลี่ยนสถานะของฐานข้อมูลเซิร์ฟเวอร์ DHCP ซึ่งช่วยให้ผู้ดูแลระบบสามารถเคียวรี เซิร์ฟเวอร์ DHCP ทั้งแบบโลคัลหรือแบบรีโมต สำหรับสถานะของ IP แอดเดรส เคียวรีสำหรับพูลของ IP แอดเดรส เคียวรีสำหรับไคลเอ็นต์ ลบการแม็พ IP แอดเดรสรีเฟรชเซิร์ฟเวอร์ และเปลี่ยนระดับการติดตามของเซิร์ฟเวอร์

คำสั่ง dadmin เข้ากันได้อีกกับเวอร์ชันก่อนหน้าของเซิร์ฟเวอร์ DHCP เพื่อแสดงรายการและรีเฟรชสถานะ IP แอดเดรส

เมื่อทำการเคียวรีข้อมูล IP address คำสั่ง dadmin ส่งกลับสถานะของ IP address และขึ้นอยู่กับสถานะของ IP แอดเดรส คำสั่ง dadmin อาจ ส่งคืน lease duration, start duration, start lease time, last leased time, เซิร์ฟเวอร์ สนับสนุน DNS หรือไม่, การอัปเดตเร็กคอร์ดสำหรับ IP แอดเดรสนี้ และ ตัวบ่งชี้ไคลเอ็นต์ที่แม็พกับ IP แอดเดรสนี้

เมื่อเคียวรีข้อมูล ไคลเอ็นต์ คำสั่ง dadmin จะส่งคืน IP แอดเดรสและสถานะ IP แอดเดรสของไคลเอ็นต์ เวลาสุดท้ายที่ไคลเอ็นต์ได้รับ IP แอดเดรส ชื่อโฮสต์ และชื่อโดเมนที่ไคลเอ็นต์ใช้ เซิร์ฟเวอร์สนับสนุน DNS หรือไม่ และการอัปเดตเร็กคอร์ดสำหรับ IP แอดเดรสนี้

เมื่อคุณปรับเปลี่ยนระดับการติดตามเซิร์ฟเวอร์ คำสั่ง dadmin จะตั้งค่าและส่งคืนระดับการติดตามเซิร์ฟเวอร์ ในรูปของมาสก์การติดตาม มาสก์นี้จะแสดงบิตสตริงที่แต่ละบิตแสดงว่ารายการบันทึกเฉพาะถูกติดตาม โดยเซิร์ฟเวอร์หรือไม่ (โปรดดูที่ “ไฟล์คอนฟิกูเรชันของ DHCP Server” ในเอกสารคู่มือ แบบออนไลน์) จากลำดับความสำคัญน้อยที่สุดไปจนถึงมากที่สุด รายการล็อกมีดังนี้ LOG\_NONE, LOG\_SYSERR, LOG\_OBJERR, LOG\_PROTOCOL และ LOG\_PROTERR (ค่าเหมือนกัน), LOG\_WARN, AND LOG\_CONFIG (ค่าเหมือนกัน), LOG\_EVENT และ LOG\_PARSEERR (ค่าเหมือนกัน), LOG\_ACTION, LOG\_INFM, LOG\_ACNTING, LOG\_STAT, LOG\_TRACE, LOG\_START และ LOG\_RTRACE

หมายเหตุ: LOG\_START ไม่สามารถ ถูกปิดใช้งานได้ ซึ่งแสดงถึงช่วงของมาสก์จาก 0x0800 ถึง 0x1FFF

### แฟล็ก

ไอเท็ม	คำอธิบาย
-c <i>ClientId</i>	ส่งคืนสถานะสำหรับไคลเอ็นต์เฉพาะที่เซิร์ฟเวอร์ DHCP อาจรู้จัก <i>ClientId</i> แสดงตัวบ่งชี้ไคลเอ็นต์ที่ DHCP ไคลเอ็นต์ใช้เพื่อระบุตัวเอง หรือไฟล์ที่สามารถระบุเป็นอักขระเลขฐานสิบหกเท่านั้น หรือในการแสดงค่า TYPE-STRING ที่เซิร์ฟเวอร์ DHCP ใช้
-d <i>IpAddress</i>	ลบข้อมูล lease ที่เชื่อมโยงกับ IP แอดเดรส <i>IpAddress</i> ดังนั้น แอดเดรสจะถูกเปลี่ยนเป็นสถานะ FREE และพร้อมใช้งาน สำหรับการเชื่อมโยงอีกครั้ง
-f	ใช้ร่วมกับแฟล็ก -d แฟล็ก -f บังคับการลบแอดเดรสโดยไม่ต้องมีการพร้อมต์ ลบข้อมูล lease ที่เชื่อมโยงกับ IP
-h <i>Hostname</i>	ใช้เพื่อระบุเซิร์ฟเวอร์ DHCP ปลายทาง <i>Hostname</i> เป็นชื่อหรือ IP address ก็ได้
-i	กำหนดค่าเริ่มต้นเซิร์ฟเวอร์ DHCP ใหม่ แฟล็กนี้ส่งสัญญาณให้กับเซิร์ฟเวอร์เพื่อซิงค์ฐานข้อมูลและรีสตาร์ทโดยการไฟล์คอนฟิกูเรชันอีกครั้ง
-n <i>interval</i>	แสดงสถิติเซิร์ฟเวอร์ ข้อมูลสรุป และช่วงเวลาที่ยังรอขอ
-p <i>IpAddress</i>	ส่งคืนสถานะของแต่ละ DHCP ใน subnet <i>IpAddress</i> ถูกใช้เพื่อระบุ subnet กับรายการ
-q <i>IpAddress</i>	ส่งคืนสถานะของ IP แอดเดรสที่กำหนด
-r <i>IpAddress</i>	กำหนด IP แอดเดรสให้อยู่ในสถานะ Free
-s	ส่งคืนสถานะของแต่ละแอดเดรสในพูลที่ตั้งค่าของเซิร์ฟเวอร์ DHCP
-t on off  <i>Value</i>	เปลี่ยนระดับการติดตามของเซิร์ฟเวอร์ DHCP ค่าการติดตาม ถูกรายงานในรูปแบบเลขฐานสิบหกที่แสดงมาสู่การติดตาม ที่ใช้อยู่บนเซิร์ฟเวอร์ <i>Value</i> สามารถระบุเป็นรูปแบบ เลขฐานสิบหรือเลขฐานสิบหก คีย์เวิร์ด on และ off เปิดใช้ หรือปิดใช้ บิตเดียวที่เวลาที่มีการติดตามมาสู่
-v	รันคำสั่งในโหมดรายละเอียด
-x	ใช้ Version 1 ของโปรโตคอล <b>dadmin</b> แฟล็ก -x ถูกใช้เพื่อเชื่อมต่อกับเซิร์ฟเวอร์ DHCP รีลิสก่อนหน้า และใช้ได้เฉพาะสำหรับแฟล็ก -i และ -s ตามด้วย 6 เมื่อคุณเชื่อมต่อกับเซิร์ฟเวอร์ DHCPv6
-?	แสดงไวยากรณ์การใช้

## สถานะออก

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

เมื่อต้องการรักษาความปลอดภัยการเชื่อมต่อจาก **dadmin** ไคลเอ็นต์ เซิร์ฟเวอร์ DHCP จะอนุญาตการเชื่อมต่อจากเซิร์ฟเวอร์เอง หรือจาก ระบบรีโมดที่รวมอยู่ในไฟล์ `.rhosts` ของ `superuser` เท่านั้น เมื่อต้องการไม่ให้ผู้ใช้ทั่วไปแก้ไขการแม็พแอดเดรสของเซิร์ฟเวอร์ DHCP ผู้ดูแลระบบต้องตรวจสอบแน่ใจว่าการเรียกใช้คำสั่ง **dadmin** จะจำกัดเฉพาะผู้ใช้ที่เหมาะสมบนระบบที่อนุญาตให้เข้าถึง

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/sbin/dadmin</code>	พคำสั่ง <b>dadmin</b>

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“dhcpsd Daemon” ในหน้า 139

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `.rhosts`

### ไฟล์คอนฟิกูเรชันเซิร์ฟเวอร์ DHCP

TCP/IP แอดเดรสและการกำหนด พารามิเตอร์ – Dynamic Host Configuration Protocol

TCP/IP daemons

---

## คำสั่ง date

### วัตถุประสงค์

แสดงหรือตั้งวันที่หรือเวลา

### ไวยากรณ์

เมื่อต้องการตั้งค่าวันที่และเวลาในฐานะผู้ใช้ Root

```
/usr/bin/date [-n] [-u] [Date] [+FieldDescriptor ...]
```

เมื่อต้องการแสดงวันที่และเวลา

```
/usr/bin/date [-u] [+FieldDescriptor ...]
```

เมื่อต้องการปรับเปลี่ยนเวลาเป็นวินาทีในฐานะผู้ใช้ root

```
/usr/bin/date [-a [+|-]sss[.fff]
```

### คำอธิบาย

ข้อควรสนใจ: อย่าเปลี่ยนวันที่เมื่อระบบกำลังรันโดยมีผู้ใช้มากกว่าหนึ่งคน

คำสั่ง `date` เขียน วันที่และเวลาปัจจุบันไปที่เอาต์พุตมาตรฐาน ถ้ามีการเรียกโดยไม่มีแฟล็ก หรือมีรายการแฟล็กที่เริ่มต้นด้วย + (เครื่องหมายบวก) มิฉะนั้น คำสั่งจะตั้งค่าวันที่ปัจจุบัน เฉพาะผู้ใช้ root เท่านั้นที่สามารถเปลี่ยน วันที่และเวลา คำสั่ง `date` จะพิมพ์ข้อความการใช้งานบน แฟล็กหรืออินพุตใดๆ ที่ไม่รู้จัก

รูปแบบ ต่อไปนี้สามารถใช้เมื่อคุณตั้งวันที่โดยใช้พารามิเตอร์ `Date` :

- `mmddHHMM[YYyy]`
- `mmddHHMM[yy]`

ตัวแปรที่ใช้กับพารามิเตอร์ `Date` ถูกกำหนดไว้ดังนี้:

ไอเท็ม	คำอธิบาย
<code>mm</code>	ระบุตัวเลขเดือน
<code>dd</code>	ระบุจำนวนวันในเดือน
<code>HH</code>	ระบุชั่วโมงในหนึ่งวันโดยใช้นาฬิกา 24 ชั่วโมง
<code>MM</code>	ระบุตัวเลขนาฬิกา
<code>YY</code>	ระบุตัวเลข 2 หลักแรกของปี หมายเหตุ: หากคุณไม่ได้ระบุ 2 หลักแรกของปี ค่าในช่วง 70 - 99 จะอ้างถึงศตวรรษที่ 20 คือปี 1970 - 1999 เช่นเดียวกัน ค่าในช่วง 00 - 37 จะหมายถึงปีในศตวรรษที่ 21 คือ 2000 - 2037
<code>yy</code>	ระบุตัวเลข 2 หลักสุดท้ายของปี หมายเหตุ: คำสั่ง <code>date</code> ยอมรับค่าปี 4 หลักเป็นอินพุต ตัวอย่างเช่น หากระบุปีแบบสี่หลัก คำสั่ง <code>date</code> จะพยายามตั้งค่าปีเป็น <code>YYyy</code> และจะล้มเหลวสำหรับค่าซึ่งอยู่นอกช่วง (น้อยกว่า 1970 และมากกว่า 2105) สำหรับปีในช่วง 2038 - 2105 ให้ระบุปีในรูปแบบ <code>yyyy</code>

ปีปัจจุบันถูกใช้เป็นตัวฟอลต์ เมื่อไม่มีการระบุปี ระบบทำงานในแบบ Coordinated Universal Time (CUT)

ถ้าคุณต่อท้ายคำสั่ง `date` ด้วย + (เครื่องหมายบวก) และคำสั่ง `descriptor`, คุณสามารถควบคุมเอาต์พุตของคำสั่ง คุณต้องนำหน้าแต่ละฟิลด์ `descriptor` ด้วย % (เครื่องหมายเปอร์เซ็นต์) ระบบ แทนที่ฟิลด์ `descriptor` ด้วยค่าที่ระบุ ป้อนตัวอักษร % เป็น %% (เครื่องหมายเปอร์เซ็นต์สองเครื่องหมาย) คำสั่ง `date` คัดลอกอักขรอื่นไปที่เอาต์พุตโดยไม่มีการเปลี่ยนแปลง คำสั่ง `date` ปิดท้ายสตริงด้วยอักขระขึ้นบรรทัดใหม่ เสมอ

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a [+   - ]sss[.fff]	ปรับเวลาอย่างช้าๆ เป็น sss.fff วินาที (fff แสดง เศษส่วนของวินาที) การปรับนี้จะได้ทั้งค่าบวกหรือลบ นาฬิกาของระบบจะถูกปรับให้เร็วขึ้นหรือช้าลงจนกว่าจะได้ตามจำนวนของวินาทีที่ระบุ
-n	ห้ามตั้งเวลาโดยรวมบนระบบทั้งหมดใน local area network ที่ซึ่งใครในข่ายนั้น
-u	แสดงหรือตั้งเวลาในแบบ Coordinated Universal Time (CUT)

## ฟิลด์ Descriptors

ไอเท็ม	คำอธิบาย
%a	แสดงชื่อวันทำงานแบบย่อของโลแคล
%A	แสดงชื่อวันทำงานแบบเต็มของโลแคล
%b	แสดงชื่อเดือนแบบย่อของโลแคล
%B	แสดงชื่อเดือนแบบเต็มของโลแคล
%c	แสดงวันที่และเวลาที่เหมาะสมกับโลแคล (ดีฟอลต์)
%C	แสดง 2 หลักแรกของปีที่มีสี่หลักเป็นเลขฐานสิบ (00-99) ปีถูกหารด้วย 100 และปัดเป็น จำนวนเต็ม
%d	แสดงวันของเดือนเป็นตัวเลขฐานสิบ (01-31) ในฟิลด์ตัวเลขสองหลัก 0 ถูกใช้เป็นตัวเลขแทนที่ว่าง
%D	แสดงวันที่ในรูปแบบ %m/%d/%y
%e	แสดงวันของเดือนเป็นตัวเลขฐานสิบ (1-31) ในฟิลด์ตัวเลขสองหลัก ช่องว่างถูกใช้เป็นตัวเลขแทนที่ว่าง
%h	แสดงชื่อเดือนแบบย่อของโลแคล (ความหมายเหมือน %b)
%H	แสดงชั่วโมง (นาฬิกา 24-ชั่วโมง) เป็นตัวเลขฐานสิบ (00-23)
%I	แสดงชั่วโมง (นาฬิกา 12-ชั่วโมง) เป็นตัวเลขฐานสิบ (01-12)
%j	แสดงวันของปีเป็นตัวเลขฐานสิบ (001-366)
%k	แสดงนาฬิกาแบบ 24 ชั่วโมงเป็นเลขที่เต็มช่องว่าง ซิตขวา (0 - 23)
%m	แสดงเดือนของปีเป็นตัวเลขฐานสิบ (01-12)
%M	แสดงนาทีเป็นตัวเลขฐานสิบ (00-59)
%g	แทรกอักขระขึ้นบรรทัดใหม่
%p	แสดงค่าเทียบเท่า AM หรือ PM ของโลแคล
%r	แสดงเวลานาฬิกา 12-ชั่วโมง (01-12) โดยใช้รูปแบบ AM-PM ในโลแคล POSIX ซึ่งเทียบเท่ากับ %I:%M:%S%p
%S	แสดงวินาทีเป็นเลขฐานสิบ (00 - 59)
%s	แสดงจำนวนวินาทีตั้งแต่ January 1, 1970, Coordinated Universal Time (CUT)
%t	แทรกอักขระ <tab>
%T	แสดงนาฬิกา 24-ชั่วโมง (00-23) ในรูปแบบเทียบเท่ากับ HH:MM:SS
%u	แสดงวันในสัปดาห์เป็นเลขฐานสิบในช่วง 1-7 (วันอาทิตย์ = 7) อ้างอิงถึง %w ฟิลด์ descriptor
%U	แสดงสัปดาห์ของปี (Sunday เป็นวันแรกของสัปดาห์) เป็นตัวเลขฐานสิบ [00 - 53] วันทั้งหมดในปีใหม่ที่นำหน้าด้วยวันอาทิตย์ แรกจะถือว่าเป็นสัปดาห์ที่ 0
%V	แสดงสัปดาห์ของปีเป็นเลขฐานสิบในช่วง 01-53 (วันจันทร์ใช้เป็นตัวแรกของสัปดาห์) หากสัปดาห์ที่มี วันที่ 1 มกราคมมีสี่วันหรือมากกว่าในปีใหม่ จะถือว่าเป็นสัปดาห์ที่ 01 ไม่เช่นนั้น จะเป็นสัปดาห์ที่ 53 ของปีก่อน
%w	แสดงวันในสัปดาห์เป็นเลขฐานสิบในช่วง 0-6 (วันอาทิตย์ = 0) อ้างอิงถึง %u ฟิลด์ descriptor
%W	แสดงตัวเลขสัปดาห์ของปีเป็นตัวเลขฐานสิบ (00-53) นับวันจันทร์เป็นวันแรกของสัปดาห์
%x	แสดงการแสดงผลวันที่ที่เหมาะสมของโลแคล
%X	แสดงการแสดงผลเวลาที่เหมาะสมของโลแคล
%y	แสดงตัวเลขสองหลักสุดท้ายของปี (00-99)
%Y	แสดงปีแบบตัวเลขสี่หลักเป็นตัวเลขฐานสิบ
%Z	แสดงชื่อเขตเวลา หรือไม่มีอักขระ ถ้าไม่สามารถ กำหนดเขตได้
%%	แสดงอักขระ % (เครื่องหมายเปอร์เซ็นต์)

## ฟิลด์ Descriptors ที่มีการตัดแปลง

ฟิลด์ descriptor %E และ %O สามารถแก้ไขเพื่อให้ระบุรูปแบบหรือข้อมูลจำเพาะที่ต่างกัน ดังอธิบายใน LC\_TIME Category สำหรับ Locale Definition Source File Format ใน *Files Reference* ถ้า คีย์เวิร์ดที่สัมพันธ์กัน (ดูที่คีย์เวิร์ด era, era\_year, era\_d\_fmt และ alt\_digits) ไม่ถูกระบุหรือไม่สนับสนุน locale ปัจจุบันค่าฟิลด์ descriptor ที่ไม่ถูกเปลี่ยนแปลงจะถูกนำมาใช้

ไอเท็ม	คำอธิบาย
%Ec	แสดงการแสดงวันที่และเวลาที่เหมาะสมเพื่อเป็นทางเลือกของ โลกแคล
%EC	แสดงชื่อของปีฐาน (หรือช่วงเวลาอื่น) ในการแสดงค่าทางเลือกของ โลกแคล
%Ex	แสดงการแสดงวันที่ที่เป็นทางเลือกของ โลกแคล
%EX	แสดงการแสดงเวลาที่เป็นการทางเลือกของ โลกแคล
%Ey	แสดงออฟเซตจากฟิลด์ descriptor %EC (ปีเท่านั้น) ในการแสดงทางเลือกของ locale
%EY	แสดงการแสดงปีที่เป็นทางเลือกแบบเต็ม
%Od	แสดงวันของเดือนโดยใช้สัญลักษณ์ตัวเลข ที่เป็นทางเลือกของ โลกแคล
%Oe	แสดงวันของเดือนโดยใช้สัญลักษณ์ตัวเลข ที่เป็นทางเลือกของ โลกแคล
%OH	แสดงชั่วโมง (นาฬิกา 24-ชั่วโมง) โดยใช้สัญลักษณ์ตัวเลข ที่เป็นทางเลือกของ โลกแคล
%OI	แสดงชั่วโมง (นาฬิกา 12-ชั่วโมง) โดยใช้สัญลักษณ์ตัวเลข ที่เป็นทางเลือกของ โลกแคล

ไอเท็ม	คำอธิบาย
%Om	แสดงเดือนโดยใช้สัญลักษณ์ตัวเลข ที่เป็นทางเลือกของ โลกแคล
%OM	แสดงนาฬิกาโดยใช้สัญลักษณ์ตัวเลขที่เป็นทางเลือกของ โลกแคล
%OS	แสดงวันที่โดยใช้สัญลักษณ์ตัวเลขที่เป็นทางเลือกของ โลกแคล
%Ou	แสดงวันในสัปดาห์เป็นตัวเลขในการแสดงที่เป็นทางเลือกของ โลกแคล (Monday=1)
%OU	แสดงตัวเลขสัปดาห์ของปีโดยใช้สัญลักษณ์ตัวเลขที่เป็นทางเลือก ของ โลกแคล Sunday ถือเป็นวันแรกของสัปดาห์
%OV	แสดงตัวเลขสัปดาห์ของปีโดยใช้สัญลักษณ์ตัวเลขที่เป็นทางเลือก ของ โลกแคล Monday ถือเป็นวันแรกของสัปดาห์
%Ow	แสดงวันในสัปดาห์เป็นตัวเลขในการแสดงที่เป็นทางเลือกของ โลกแคล (Sunday=0)
%OW	แสดงตัวเลขสัปดาห์ของปีโดยใช้สัญลักษณ์ตัวเลขที่เป็นทางเลือก ของ โลกแคล Monday ถือเป็นวันแรกของสัปดาห์
%Oy	แสดงปี (ออฟเซตจาก %C) ในการแสดงที่เป็นทางเลือก

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	วันที่ถูกเขียนเสร็จสมบูรณ์
>0	มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

- เมื่อต้องการแสดงวันที่และเวลาปัจจุบัน ให้ป้อน:

```
date
```

- เมื่อต้องการตั้งค่าวันที่และเวลา ให้ป้อน:

```
date 0217142590
```

สำหรับระบบที่ใช้ CST เป็นเขตเวลา คำสั่งนี้จะตั้ง วันที่และเวลาเป็น Sat Feb 17 14:25:00 CST 1990

หมายเหตุ: คุณต้องมีสิทธิ์ root ในการเปลี่ยนวันที่และเวลา

### 3. เมื่อต้องการแสดงวันที่และเวลาในรูปแบบที่ระบุให้ป้อน:

```
date +"%r %a %d %h %y (Julian Date: %j)"
```

คำสั่งนี้จะแสดงวันที่ที่แสดงในตัวอย่าง 2 เป็น:

```
02:25:03 PM Fri 17 Feb 90 (Julian Date: 048)
```

## ตัวแปรสภาวะแวดล้อม

ตัวแปรสภาวะแวดล้อมต่อไปนี้จะมีผลกับการทำงานของคำสั่ง `date`

ไอเท็ม	คำอธิบาย
<code>LANG</code>	กำหนด locale ที่จะใช้เมื่อทั้ง <code>LC_ALL</code> และ ตัวแปรสภาวะแวดล้อมที่สัมพันธ์กัน (เริ่มต้นด้วย <code>LC_</code> ) ไม่ได้ระบุ locale
<code>LC_ALL</code>	กำหนด locale ที่จะใช้เพื่อแทนที่ค่าใดๆ สำหรับ หมวดหมู่ locale ที่ระบุโดยค่าติดตั้งของ <code>LANG</code> หรือ ตัวแปรสภาวะแวดล้อมใดๆ ที่เริ่มต้นด้วย <code>LC_</code>
<code>LC_CTYPE</code>	กำหนด locale สำหรับการแปลลำดับของไบต์ ของข้อมูลข้อความเป็นอักขระ (ตัวอย่าง อักขระไบต์เดี่ยวเทียบกับ อักขระหลายไบต์ในอาร์กิวเมนต์)
<code>LC_MESSAGES</code>	กำหนดภาษาที่ข้อความจะถูกเขียน
<code>LC_TIME</code>	กำหนดเนื้อหาของสตริงวันที่และเวลาที่เขียน โดยคำสั่ง <code>date</code>
<code>NLSPATH</code>	กำหนดที่ตั้งของแค็ตตาล็อกข้อความสำหรับการประมวลผล <code>LC_MESSAGES</code>
<code>TZ</code>	ระบุเขตเวลาที่วันที่และเวลาถูกเขียน นอกจากระบุแฟล็ก -u ถ้าตัวแปร <code>TZ</code> ไม่ได้ถูกตั้งค่าและไม่มีการระบุแฟล็ก -u จะใช้เขตเวลาดีฟอลต์ของระบบที่ไม่ระบุ

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `localtime`

คำสั่ง `time`

หมวดหมู่ `LC_TIME`

การทำความเข้าใจกับ locale

---

## คำสั่ง `dbts`

### วัตถุประสงค์

ดีบั๊ก thin server

### ไวยากรณ์

```
dbts [-V] ThinServer
```

### คำอธิบาย

คำสั่ง `dbts` จะเริ่มต้น thin server ในโหมดดีบั๊ก คำสั่งจะตรวจสอบว่า thin server ถูกเริ่มต้นก่อนหน้านี้ในโหมดดีบั๊กหรือไม่โดยการค้นหา บุตอิมเมจการดีบั๊กที่สร้างขึ้นสำหรับ thin server ถ้าไม่พบ อิมเมจทั่วไปที่ thin server ใช้จะถูกโคลนและ ดีบั๊กบุตอิมเมจ ถูกจากสร้างโคลน เพื่อให้ thin server สามารถบูตเข้าสู่ดีบั๊ก โหมด ดีบั๊กบุตอิมเมจโคลนใช้ระเบียบการตั้งชื่อต่อไปนี้:

```
{COSI name}_{thin server name}-debug
```

หลังจาก thin server เสร็จสิ้นการใช้อิมเมจทั่วไปของการติบัก คำสั่ง swts ต้องถูกรันเพื่อสลับ thin server ไปที่อิมเมจทั่วไปอื่น คำสั่ง rmcosi จะลบอิมเมจทั่วไปของการติบักที่สร้างขึ้นจาก คำสั่ง dbts คำสั่ง dbts สามารถรันบน NIM master หรือ thin server

## แฟล็ก

ไอเท็ม  
-v

คำอธิบาย  
เปิดใช้ verbose ติบักเอาต์พุตขณะคำสั่ง dbts รัน

## สถานะออก

ไอเท็ม  
0  
>0

คำอธิบาย  
คำสั่งดำเนินการเสร็จด้วยผลสำเร็จ  
มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

ค่าควบคุมการเข้าใช้: คุณต้องมีสิทธิ์ root ในการรันคำสั่ง dbts

## ตัวอย่าง

1. เมื่อต้องการติบักบูต thin server ชื่อ lobo ที่ใช้อิมเมจทั่วไป ชื่อ cosi1 ให้ป้อน:

```
dbts lobo
```

ติบักบูตอิมเมจชื่อ cosi1\_lobo-debug ถูกสร้างเพื่อบูต lobo เข้าสู่ ติบักโหมด

## Location

```
/usr/sbin/dbts
```

## ไฟล์

ไอเท็ม  
/etc/niminfo

คำอธิบาย  
มีตัวแปรที่ใช้โดย NIM

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง lsts

คำสั่ง mkts

คำสั่ง nim

คำสั่ง rmts

---

## คำสั่ง dbx

## วัตถุประสงค์

จัดเตรียมสภาวะแวดล้อมเพื่อการติบักและรันโปรแกรม

## ไวยากรณ์

```
dbx [-a ProcessID] [-B DebugFile] [-c CommandFile] [-I Directory] [-E DebugEnvironment] [-p oldpath=newpath:.../  
pathfile] [-u] [-F] [-L] [-r] [-x] [-v] [-C CoreFile|ObjectFile [CoreFile]]
```

## คำอธิบาย

คำสั่ง **dbx** มีโปรแกรมดีบั๊กเชิงสัญลักษณ์สำหรับโปรแกรม C, C++ และ Fortran เพื่อให้คุณสามารถดำเนินการต่อไปนี้:

- ตรวจสอบอ็อบเจกต์และคอร์ไฟล์
- จัดเตรียมสภาวะแวดล้อมที่มีการควบคุมสำหรับการรันโปรแกรม
- เช็จุดพักที่คำสั่งที่เลือกหรือ รันโปรแกรมครั้งละหนึ่งบรรทัด
- ดีบั๊กใช้ตัวแปรเชิงสัญลักษณ์และแสดง ตัวแปรในรูปแบบที่ถูกต้อง

พารามิเตอร์ *ObjectFile* เป็นอ็อบเจกต์ไฟล์ (ที่รันได้) ที่สร้างโดยคอมไพเลอร์ ใช้แฟล็ก -g (สร้างตารางสัญลักษณ์) เมื่อทำการคอมไพล์ โปรแกรมของคุณเพื่อสร้างข้อมูลคำสั่ง **dbx** ที่จำเป็น

**หมายเหตุ:** แฟล็ก -g ของคำสั่ง **cc** ต้องใช้เมื่ออ็อบเจกต์ไฟล์ถูกคอมไพล์ ถ้าแฟล็ก -g ไม่ถูกใช้หรือ ถ้าการอ้างอิงสัญลักษณ์ ถูกลบออกจากไฟล์ **xcoff** ด้วยคำสั่ง **strip** ความสามารถเชิงสัญลักษณ์ของ คำสั่ง **dbx** จะถูกจำกัด นอกจากนี้อย่าใช้ คอมไพเลอร์อ็อพชัน -O เพื่อออปติไมซ์โปรแกรม ที่คุณวางแผนที่จะดีบั๊กด้วย **dbx** การออปติไมซ์จัดเรียงโค้ดใหม่ และดำเนินการข้อมูลดีบั๊ก จำกัดค่าการดีบั๊กโปรแกรมเรียกทำงานเพิ่มเติมด้วยคำสั่ง **dbx**

ถ้าแฟล็ก -c ไม่ถูก ระบุ คำสั่ง **dbx** จะตรวจสอบหาไฟล์ **.dbxinit** ใน ไดเรกทอรี **\$HOME** ของผู้ใช้ จากนั้นจะตรวจสอบ ไฟล์ **.dbxinit** ในไดเรกทอรีปัจจุบันของผู้ใช้ ถ้าไฟล์ **.dbxinit** มีอยู่ในไดเรกทอรีปัจจุบัน ไฟล์นั้นจะเขียนทับไฟล์ **.dbxinit** ใน ไดเรกทอรี **\$HOME** ของผู้ใช้ ถ้าไฟล์ **.dbxinit** มีอยู่ในไดเรกทอรี **\$HOME** ของผู้ใช้ หรือไดเรกทอรีปัจจุบัน คำสั่งย่อยของไฟล์นั้นรัน ตอนเริ่มต้น ของเซชันการดีบั๊ก ใช้เอดิเตอร์เพื่อสร้างไฟล์ **.dbxinit**

ถ้า *ObjectFile* ไม่ถูก ระบุ **dbx** จะถามถึงชื่อของ อ็อบเจกต์ไฟล์ที่จะถูกตรวจสอบ ดีฟอลต์คือ **a.out** ถ้าไฟล์ **core** มีอยู่ในไดเรกทอรีปัจจุบัน หรือพารามิเตอร์ *CoreFile* ถูกระบุ **dbx** จะรายงานตำแหน่งที่โปรแกรมผิดพลาด ตัวแปร รีจิสเตอร์ และหน่วย ความจำที่เก็บในอิมเมจหลัก อาจถูกตรวจสอบจนกระทั่งการเรียกใช้งาน *ObjectFile* เริ่มต้นขึ้น ที่ จุดนั้นโปรแกรมดีบั๊ก **dbx** พร้อมต์รอรับคำสั่ง

แฟล็ก -B ใช้เพื่อระบุอ็อบเจกต์ไฟล์ทางเลือก หรือไฟล์ **.stab** แยกต่างหากที่มีข้อมูลการดีบั๊กเกี่ยวกับการเริ่มทำงาน อ็อบเจกต์ไฟล์ ทางเลือกสามารถระบุขณะเชื่อมต่อไปยังกระบวนการเท่านั้น ข้อมูลดีบั๊กถูกอ่านจากอ็อบเจกต์ไฟล์รองนี้หรือดีบั๊กไฟล์ **.stab** แทนสำเนาดีบั๊กของ กระบวนการที่รันอยู่ อ็อบเจกต์ไฟล์รองนี้ต้องเป็นสำเนา ที่ **unstripped** ของอ็อบเจกต์ไฟล์ต้นฉบับ มิฉะนั้นจะถูกละเว้น ใช้ แฟล็ก -B เมื่อขนาดของส่วนดีบั๊ก มีขนาดใหญ่ ใช้สำเนาที่ **strip** ของอ็อบเจกต์ไฟล์ขณะรันและ **unstripped** สำเนาขณะทำการดีบั๊ก ดีบั๊กไฟล์ **.stab** สามารถถูกสร้าง ผ่านอ็อพชันตัวเชื่อมโยง **-bstabsplit** ถ้าไม่ระบุแฟล็ก -B สำหรับโปรแกรม **stabsplit** คำสั่ง **dbx** จะพยายามรับไฟล์ **.stab** ที่ตรงกันจากไดเรกทอรีโปรแกรม

## การจัดการนิพจน์

โปรแกรม **dbx** สามารถแสดง นิพจน์ได้หลากหลาย คุณสามารถระบุนิพจน์ ในโปรแกรมดีบั๊ก **dbx** ด้วยไวยากรณ์ C ที่มีส่วยขยาย Fortran บางส่วน

โอเปอเรเตอร์ต่อไปนี้ใช้ได้กับโปรแกรม:

## ไอเท็ม

\* (เครื่องหมายดอกจัน) หรือ ^ (คาร์เรต์)  
[] (วงเล็บเหลี่ยม) หรือ () (วงเล็บ)  
.(จุด)

& (แอมเปอร์แซนด์)

.. (สองจุด)

## คำอธิบาย

หมายถึงการเปลี่ยนทิศทางหรือการยกเลิกการอ้างอิงตัวชี้

หมายถึงนิพจน์ซึบสคริปต์ของ array

ใช้ฟิลต์นี้อ้างอิงโอเปอเรเตอร์ที่มีตัวชี้และโครงสร้าง

(structures) ตัวดำเนินการนี้ทำให้ตัวดำเนินการ C-> (ลูกศร)

ไม่จำเป็นต้องใช้ แม้ว่า จะอนุญาตให้ใช้ได้

รับแอดเดรสของตัวแปร

แยกขอบเขตบนและล่าง เมื่อระบุ subsection ของ array ตัวอย่าง:

n[1..4]

ชนิดของการดำเนินการต่อไปนี้ใช้ได้ ในนิพจน์ในสคริปต์โปรแกรม:

## ไอเท็ม

Algebraic

Bitwise

Logical

Comparison

อื่นๆ

## คำอธิบาย

=, -, \*, / (การหาร floating), div (การหารจำนวนเต็ม), mod, exp (การยกกำลัง)

~, !, bitand, xor, ~. <<, >>

or, and, not, !!, &&

<, >, <=, >=, <> หรือ !=, = หรือ ==

(typename), sizeof

นิพจน์โลจิคัลและการเปรียบเทียบ สามารถใช้เป็นเงื่อนไขใน stop และ trace

ชนิดนิพจน์จะถูกตรวจสอบ คุณเขียนทับ ชนิดนิพจน์โดยใช้การเปลี่ยนชื่อหรือโอเปอเรเตอร์ casting สามรูปแบบ ของการเปลี่ยนชื่อชนิดคือ *Typename(Expression)*, *Expression!Typename* และ *(Typename) Expression* ต่อไปนี้เป็น ตัวอย่างซึ่งตัวแปร *x* คือจำนวนเต็ม ที่มีค่า 97:

```
(dbx) print x
```

```
97
```

```
(dbx) print char (x), x \ char, (char) x, x
```

```
'a' 'a' 'a' 97
```

## การแก้ไขบรรทัดคำสั่ง

คำสั่ง **dbx** มีคุณลักษณะการแก้ไขบรรทัดรับคำสั่งคล้ายกับคุณลักษณะที่มีให้โดย Korn Shell โหมด **vi** มีคุณลักษณะการแก้ไข **vi-like** ขณะที่โหมด **emacs** ให้คุณมีการควบคุมเหมือนกับ **emacs**

คุณลักษณะ เหล่านี้สามารถเปิดใช้โดยใช้คำสั่งย่อย **dbx** คือ **set -o** หรือ **set edit** เมื่อต้องการเปิดใช้การแก้ไข บรรทัดรับคำสั่งในลักษณะ **vi** คุณควรพิมพ์คำสั่งย่อย **set edit vi** หรือ **set -o vi**

คุณยังสามารถใช้ตัวแปรสถานะแวดล้อม **EDITOR** เพื่อตั้งค่าโหมดแก้ไข

คำสั่ง **dbx** จะบันทึก ประวัติของคำสั่ง ซึ่งถูกป้อนในบรรทัดรับคำสั่ง ในไฟล์ประวัติ **.dbxhist** หากไม่ได้ตั้งค่าตัวแปรสถานะแวดล้อม **DBXHISTFILE** ไฟล์ประวัติ **\$HOME/.dbxhist** จะถูกใช้

โดยดีฟอลต์ **dbx** บันทึกข้อความของคำสั่ง 128 สุดท้ายที่ป้อน ตัวแปรสถานะแวดล้อม **DBXHISTSZIE** สามารถถูกใช้เพื่อเพิ่มค่าจำกัดนี้

## แฟล็ก

## ไอเอ็ม

-a ProcessID

-BDebugFile

-C CommandFile

-C CoreFile

-E DebugEnvironment

-p oldpath=newpath:.../pathfile

-F

-L

-I Directory

-r

-u

-v

-x

## คำอธิบาย

ผนวกดีบั๊กโปรแกรมกับกระบวนการที่รันอยู่ เมื่อต้องการผนวกดีบั๊กโปรแกรม คุณต้องมีสิทธิในการส่งสัญญาณไปที่กระบวนการนี้ ใช้คำสั่ง ps เพื่อกำหนด process ID ถ้าคุณมีสิทธิการใช้งาน โปรแกรม dbx อินเทอร์เน็ตกระบวนการโดยใช้การเรียกของระบบ ptrace เพื่อส่งสัญญาณ SIGTRAP ไปที่กระบวนการซึ่งไม่สามารถละเว้นสัญญาณ SIGTRAP จากนั้นจำกำหนดชื่อเต็มของ อ็อบเจกต์ไฟล์ อ่านข้อมูลเชิงสัญลักษณ์ และพร้อมรอรับคำสั่ง

แฟล็กนี้อนุญาตให้คุณระบุไฟล์การดีบั๊กทางเลือกเมื่อเริ่มทำงาน

รันคำสั่งย่อย dbx ในไฟล์ ก่อนการอ่านอินพุตมาตรฐาน ไฟล์ที่ระบุในไดเรกทอรี \$HOME ถูกประมวลผลก่อน จากนั้นไฟล์ ในไดเรกทอรีปัจจุบันถูกประมวลผล ไฟล์คำสั่งในไดเรกทอรีปัจจุบัน เขียนทับไฟล์คำสั่งในไดเรกทอรี \$HOME ถ้าไฟล์ที่ระบุไม่มีอยู่ในไดเรกทอรี \$HOME หรือไดเรกทอรีปัจจุบันข้อความค่าเตือนจะถูกแสดง คำสั่งย่อย source สามารถถูกใช้ เมื่อโปรแกรม dbx ถูกเริ่มทำงาน

วิเคราะห์คอร์ไฟล์โดยไม่ต้อง ระบุอ็อบเจกต์ไฟล์ ในกรณีนี้ คำสั่ง dbx ใช้อ็อบเจกต์ไฟล์ที่กล่าวถึง ในคอร์ไฟล์ถ้าไฟล์อยู่ใน ไดเรกทอรีปัจจุบัน และตรงกับคอร์ไฟล์ มิฉะนั้น จะดำเนินการต่อ โดยไม่มีอ็อบเจกต์ไฟล์ แฟล็กนี้ถูกละเลยถ้าคุณใช้ หลังจากแฟล็ก -r หรือแฟล็ก -a

ระบุตัวแปรสภาวะแวดล้อมสำหรับดีบั๊กโปรแกรม

ระบุการแทนที่สำหรับพาสไลบรารี เมื่อการตรวจสอบคอร์ไฟล์ หรือการผนวกกับกระบวนการ ในรูปแบบ oldpath=newpath ตัวแปร oldpath ระบุค่าที่จะถูก แทนที่ (ตามที่ระบุในคอร์ไฟล์หรือส่วนตัวโหนดของกระบวนการเมื่อทำการผนวก) ตัวแปร newpath ระบุข้อมูลที่จะนำมาแทนที่ ตัวแปร oldpath และ newpath เป็นได้ทั้งพาสสมบูร์ม พาสบางส่วน พาสสัมพันธ์ หรือพาสสัมพันธ์ การแทนที่ข้อมูลหลายรายการ ถูกแยกด้วยโคลอน อีกทางหนึ่ง แฟล็ก -p อาจระบุชื่อของไฟล์ซึ่งการแก้ไข ในรูปแบบที่อธิบายก่อนหน้านี้จะถูกอ่าน อนุญาตเฉพาะหนึ่งการแก้ไขต่อบรรทัด เมื่อการแก้ไขถูกอ่านจากไฟล์ ถ้าคุณใช้แฟล็ก -p เมื่อทำการผนวกกับกระบวนการ ข้อมูลดีบั๊กถูกอ่านจาก พาสไฟล์ที่แทนที่ พาสไฟล์ที่ต้องตรงกับสำเนาของไลบรารีที่รันอยู่

สามารถใช้เพื่อปิดโหมด lazy read และทำให้คำสั่ง dbx อ่านสัญลักษณ์ทั้งหมดขณะเริ่มทำงาน โดยดีฟอลต์โหมดการอ่านแบบ lazy เปิดใช้: โดยจะอ่านข้อมูลตารางสัญลักษณ์ที่จำเป็นเท่านั้นเกี่ยวกับการเริ่มต้นเซสชัน dbx ในโหมดนี้ dbx ไม่อ่านตัวแปรโลคัล และพิมพ์ข้อมูลเชิงสัญลักษณ์ของตัวแปรที่ไม่ถูกอ่าน ดังนั้น คำสั่งเช่น whereis i อาจไม่แสดงรายการอินสแตนซ์ทั้งหมดของ ตัวแปรโลคัล ในทุกฟังก์ชัน

รักษาสัญลักษณ์ linkage

(ตัวพิมพ์ใหญ่ i) รวมไดเรกทอรีที่ระบุโดยตัวแปร Directory ในรายการของไดเรกทอรี ที่ค้นหาไฟล์ต้นฉบับ ค่าดีฟอลต์คือค้นหาไฟล์ต้นฉบับ ในไดเรกทอรีต่อไปนี้:

- ไดเรกทอรีที่พบไฟล์ต้นฉบับ เมื่อไฟล์ถูกคอมไพล์ ไดเรกทอรีนี้ถูกค้นหาเฉพาะ ถ้าคอมไพลเลอร์วางซอร์สพาสไว้ในอ็อบเจกต์
- ไดเรกทอรีปัจจุบัน
- ไดเรกทอรีซึ่งโปรแกรมตั้งอยู่ในปัจจุบัน

รันอ็อบเจกต์ไฟล์ทันที ถ้าจบการทำงานสำเร็จ ออกจากดีบั๊กโปรแกรม dbx มิฉะนั้น จะเข้าสู่ดีบั๊กโปรแกรม และเหตุผลที่จบการทำงานจะถูกรายงาน

หมายเหตุ: ยกเว้นระบุ -r คำสั่ง dbx จะพร้อมผู้ใช้และรอคำสั่ง

ทำให้คำสั่ง dbx นำหน้าสัญลักษณ์ชื่อไฟล์ ด้วย @ (เครื่องหมาย at) แฟล็กนี้ลดความเป็นไปได้ ของชื่อสัญลักษณ์ที่คลุมเครือ

ทำให้คำสั่ง dbx ข้ามการตรวจสอบความถูกต้องของคอร์ไฟล์ แฟล็กนี้อนุญาตให้คุณวิเคราะห์ส่วนที่ใช้ได้ของคอร์ไฟล์ แมวบางส่วนจะไม่ถูกต้อง

ป้องกันมีคำสั่ง dbx ดึงอักขระ \_ (เครื่องหมายขีดเส้นใต้ ยาว) ออกจากสัญลักษณ์ที่สร้างขึ้นในซอร์สโค้ด Fortran แฟล็กนี้อนุญาต dbx ให้แยก ระหว่างสัญลักษณ์ที่เหมือนกันยกเว้นอักขระ underscore เช่น xxx และ xxx\_

## ตัวอย่าง

1. ตัวอย่างต่อไปนี้อธิบายวิธีเริ่ม ดีบั๊กโปรแกรม dbx พร้อมกับ กระบวนการ ตัวอย่างใช้โปรแกรมชื่อ samp.c โปรแกรม C นี้ขั้นแรกถูกคอมไพล์ด้วยแฟล็ก -g เพื่อสร้างอ็อบเจกต์ไฟล์ที่มีการอ้างอิงตารางเชิงสัญลักษณ์ ในกรณีนี้ โปรแกรมชื่อ samp:

```
$ cc -g samp.c -o samp
```

เมื่อโปรแกรม **samp** ถูกรัน ระบบปฏิบัติการจะรายงานข้อผิดพลาดของบัส และเขียนคอร์รัปเมจไปที่ได้เรียกทอรีทำงานของคุณ ดังนี้:

```
$ samp
Bus Error - core dumped
```

เมื่อต้องการกำหนดตำแหน่งที่ข้อผิดพลาดเกิดขึ้นให้ป้อน:

```
$ dbx samp
```

ระบบจะส่งคืนข้อความต่อไปนี้:

```
dbx version 3.1
Type 'help' for help.
reading symbolic information . . . [
using memory image in core]
   25  x[i] = 0;
(dbx) quit
```

2. ตัวอย่างนี้อธิบายวิธีผนวก **dbx** เข้ากับกระบวนการ ตัวอย่างนี้ใช้โปรแกรมต่อไปนี้ **looper.c**:

```
main()
{
    int i,x[10];

    for (i = 0; i < 10;);
}
```

โปรแกรมไม่มีการสิ้นสุดเนื่องจาก *i* ไม่ถูกเพิ่มค่า คอมไพล์ **looper.c** ด้วยแฟล็ก **-g** เพื่อรับความสามารถในการดีบั๊กเชิงสัญลักษณ์:

```
$ cc -g looper.c -o looper
```

รัน **looper** จาก บรรทัดคำสั่งและดำเนินการขึ้นตอนต่อไปนี้เพื่อผนวก **dbx** กับโปรแกรมขณะที่รันอยู่:

- a. เมื่อต้องการผนวก **dbx** กับ **looper** คุณต้องกำหนด process ID ถ้าคุณไม่รัน **looper** เป็นการประมวลผลส่วนหลัง คุณต้องมี Xwindow อื่นเปิดอยู่ จาก Xwindow นี้ให้ป้อน:

```
ps -u UserID
```

โดยที่ *UserID* คือ ล็อกอิน ID ของคุณ กระบวนการที่แฉ่คที่ทั้งหมดที่เป็นของคุณถูกแสดง ดังนี้:

PID	TTY	TIME	COMMAND
68	console	0:04	sh
467	lft3	10:48	looper

ในตัวอย่างนี้ process ID ที่เชื่อมโยงกับ **looper** คือ 467

- b. เมื่อต้องการผนวก **dbx** กับ **looper** ให้ป้อน:

```
$ dbx -a 467
```

ระบบจะส่งคืนข้อความต่อไปนี้:

```
Waiting to attach to process 467 . . .
Successfully attached to /tmp/looper.
dbx is initializing
Type 'help' for help.
reading symbolic information . . .
```

```
attached in main at line 5
5   for (i = 0; i < 10;);
(dbx)
```

ขณะนี้คุณสามารถเคียวี และดีบั๊กกระบวนการ ได้เสมือนว่ากระบวนการนั้นเริ่มต้นทำงานด้วย dbx

3. เมื่อต้องการเพิ่มไคเร็กทอรีให้กับรายการของไคเร็กทอรีที่จะถูกค้นหาไฟล์ต้นฉบับของไฟล์เรียกทำงาน objfile คุณสามารถป้อน:

```
$dbx -I /home/user/src -I /home/group/src
objfile
```

คำสั่งย่อย use อาจถูกใช้สำหรับฟังก์ชันนี้เมื่อ dbx เริ่มทำงาน คำสั่ง use รีเซ็ตรายการของไคเร็กทอรีซึ่งแฟล็ก -I เพิ่มไคเร็กทอรีให้กับรายการ

4. เมื่อต้องการใช้แฟล็ก -r ให้ป้อน:

```
$ dbx -r samp
```

ระบบ จะส่งคืนข้อความต่อไปนี้:

```
Entering debug program . . .
dbx version 3.1
Type 'help' for help.
reading symbolic information . . .
bus error in main at line 25
 25  x[i] = 0;
(dbx) quit
```

แฟล็ก -r อนุญาตให้คุณ ตรวจสอบสถานะของกระบวนการของคุณในหน่วยความจำ แม้ว่าคอร์ อิมเมจไม่ถูกรับเข้ามา

5. เมื่อต้องการระบุตัวแปรสถานะแวดล้อมสำหรับดีบั๊กโปรแกรม ให้ป้อน:

```
dbx -E LIBPATH=/home/user/lib -E LANG=Ja_JP objfile
```

6. เมื่อต้องการระบุอ็อบเจ็กต์ทางเลือก และไลบรารีชณะเชื่อมโยง กับกระบวนการ ให้ป้อน:

```
dbx -a 467 -B debug_samp -p /usr/lib/=/dir/debug_libs/
```

7. เมื่อต้องการระบุไฟล์ดีบั๊กที่แยกเมื่อเริ่มทำงาน ให้ป้อน:

```
dbx -B /usr/debug_samp.stab debug_samp
```

## คำสั่งย่อย dbx

หมายเหตุ: คำสั่งย่อยสามารถใช้ เฉพาะขณะรันโปรแกรมดีบั๊ก dbx

ไอเท็ม	คำอธิบาย
/	ค้นหารูปแบบไปข้างหน้าในไฟล์ต้นฉบับปัจจุบัน
?	ค้นหาแบบย้อนกลับในไฟล์ต้นฉบับปัจจุบัน
addcmd	เพิ่มคำสั่งย่อย dbx เขากับจำนวนของเหตุการณ์ที่ระบุ
alias	สร้างนามแฝงสำหรับคำสั่งย่อย dbx
assign	กำหนดค่าให้กับตัวแปร
attribute	แสดงข้อมูลเกี่ยวกับแอตทริบิวต์อ็อบเจ็กต์ ทั้งหมดหรือ เฉพาะที่เลือก
call	รันอ็อบเจ็กต์โค้ดที่เชื่อมโยงกับโพรซีเจอร์หรือฟังก์ชัน ที่ระบุ
case	เปลี่ยนวิธีที่โปรแกรมดีบั๊ก dbx ตีความสัญลักษณ์
catch	สตาทการจับสัญญาณก่อนที่สัญญาณ นั้นจะถูกส่งไปที่แอฟพลิเคชันโปรแกรม
clear	ลบการหยุดทั้งหมดที่บรรทัดซอร์สโดยเฉพาะ
cleari	เอาจุดหยุดทั้งหมดที่แอดเดรสออก
condition	แสดงข้อมูลเกี่ยวกับตัวแปรเงื่อนไข ทั้งหมด หรือเฉพาะที่เลือก

ไอเอ็ม	คำอธิบาย
cont	ดำเนินการเรียกใช้แอปพลิเคชัน โปรแกรมต่อ จากจุดหยุดปัจจุบันจนกว่าโปรแกรม เสร็จสิ้นหรือพบจุดหยุดอื่น
corefile	แสดงข้อมูลระดับสูงเกี่ยวกับคอร์ไฟล์
coremap	แสดงการแม็พของส่วนพื้นที่แอดเดรสโดยเฉพะ
delcmd	ลบคำสั่งย่อย dbx ที่เชื่อมโยง กับหมายเลขเหตุการณ์ที่ระบุ
delete	ลบการติดตามและการหยุด ที่เกี่ยวข้อง กับหมายเลขเหตุการณ์ที่ระบุ และจำนวน tskip สำหรับ thread
detach	ดำเนินการเรียกใช้แอปพลิเคชันต่อ และออกจาก ดีบั๊กโปรแกรม
เปิดใช้งาน	ปิดการติดตามและการหยุดที่ตรงตามหมายเลข เหตุการณ์ที่ระบุ
display memory	แสดงเนื้อหาของหน่วยความจำ
down	ย้ายฟังก์ชันปัจจุบันลง ในสแต็ค
dump	แสดงชื่อและค่าของตัวแปรในโพธิ์เตอร์ที่ระบุ
edit	สตาร์ทเอดิเตอร์บนไฟล์ที่ระบุ
เปิดใช้งาน	เปิดการติดตามและการหยุดที่ตรงตามหมายเลข เหตุการณ์ที่ระบุ
fd	แสดงข้อมูลไฟล์ file descriptor
file	เปลี่ยนไฟล์ต้นฉบับปัจจุบันไปเป็นไฟล์ที่ระบุ
frame	เปลี่ยนฟังก์ชันปัจจุบันเป็นฟังก์ชัน ที่ตรงกับหมายเลขเฟรมสแต็คที่ระบุ
func	เปลี่ยนฟังก์ชันปัจจุบันเป็น ฟังก์ชันหรือโพธิ์เตอร์ที่ระบุ
goto	ทำให้บรรทัดซอร์สที่ระบุเป็นบรรทัดที่จะรันถัดไป
gotoi	เปลี่ยนแอดเดรส program counter
handler	แสดงข้อมูลเกี่ยวกับ pthreads atfork และ cancelation cleanup handlers
วิธีใช้	แสดงข้อมูลวิธีใช้หรือคำสั่งย่อยหรือหัวข้อ dbx
ignore	หยุดการจับสัญญาณก่อนที่สัญญาณ นั้นจะถูกส่งไปที่แอปพลิเคชันโปรแกรม
kthread	แสดงข้อมูลเกี่ยวกับเคอร์เนล threads
limitbp	จำกัดจำนวนครั้งที่สามารถรัน จุดหยุด
list	แสดงบรรทัดของไฟล์ต้นฉบับปัจจุบัน
listi	แสดงคำสั่งจากแอปพลิเคชันโปรแกรม
malloc	แสดงข้อมูลเกี่ยวกับการใช้งานโปรแกรม ของระบบย่อย malloc
map	แสดงข้อมูลเกี่ยวกับคุณสมบัติโหลด ของแอปพลิเคชัน
move	เปลี่ยนบรรทัดถัดไปที่จะถูกแสดง
multproc	เปิดหรือปิดใช้งานการดีบั๊ก multiprocess
mutex	แสดงข้อมูลเกี่ยวกับ mutexes ทั้งหมดหรือ เฉพาะที่เลือก
next	รันโปรแกรมแอปพลิเคชันที่บรรทัดซอร์สถัดไป
nexti	รันโปรแกรมแอปพลิเคชันที่คำสั่งเครื่อง ถัดไป
onceblock	แสดงข้อมูลเกี่ยวกับ once blocks
plugin	เรียกคำสั่งย่อย plug-in หรือแสดง ชื่อของ plug-ins ที่พร้อมใช้
pluginload	โหลด plug-in
pluginunload	ยกเลิกการโหลด plug-in
print	พิมพ์ค่าของนิพจน์หรือรันโพธิ์เตอร์และพิมพ์ รหัสที่ส่งกลับของโพธิ์เตอร์นั้น
printbp	พิมพ์จำนวนครั้งที่รัน จุดหยุด
proc	แสดงข้อมูลเกี่ยวกับกระบวนการ
t	เปลี่ยนพรมตคำสั่ง dbx
quit	หยุดโปรแกรมดีบั๊ก dbx
registers	แสดงค่าของ รีจิสเตอร์เนกประสงค์, รีจิสเตอร์ system-control, รีจิสเตอร์ floating-point และ รีจิสเตอร์คำสั่งปัจจุบัน
rerun	เริ่มการใช้งานแอปพลิเคชันด้วย อาร์กิวเมนต์ก่อนหน้านี
resource	แสดงข้อมูลเกี่ยวกับรีซอร์สที่เป็นของ หรือรออยู่ โดย pthreads
return	รันโปรแกรมแอปพลิเคชันจนถึงการส่งกลับไป โพธิ์เตอร์ที่ระบุ
rwlock	แสดงข้อมูลเกี่ยวกับ rwlocks
รัน	เริ่มรันแอปพลิเคชัน
หน้าจอ	เปิด Xwindow สำหรับการโต้ตอบกับคำสั่ง dbx
set	กำหนดค่าสำหรับตัวแปรของโปรแกรมดีบั๊ก dbx
sh	ส่งคำสั่งไปที่เชลล์ที่จะถูกรัน
skip	ดำเนินการรันแอปพลิเคชันโปรแกรม ต่อจากจุดหยุดทำงานปัจจุบัน
source	อ่านคำสั่งย่อย dbx จากไฟล์
status	พิมพ์รายละเอียดเกี่ยวกับจุดหยุด รวมทั้ง แสดงการติดตามที่แอ็คทีฟ คำสั่งย่อย stop และ จำนวน tskip เธรดที่คงเหลือ
step	รันหนึ่งบรรทัดซอร์ส
stepi	รันหนึ่งคำสั่งเครื่อง
stophwp	เซ็ตฮาร์ดแวร์ watchpoint หยุด
stop	หยุดการรันแอปพลิเคชันโปรแกรม
stopi	เซ็ตการหยุดทำงานที่ตำแหน่งที่ระบุ

ไอเท็ม	คำอธิบาย
thdata	แสดงข้อมูลของเฮดเฉพาะ
เฮด	แสดงและควบคุม threads
tls	แสดงข้อมูลเพิ่มเติมเพื่อกำหนดค่าเริ่มต้น TLS
tm_status	แสดงและแปลค่าที่เก็บไว้ใน ตัวแปร \$texasr
tnext	รันเฮดขึ้นไปทีบรรทัดซอร์สถัดไป
tnexti	รันเฮดขึ้นไปทีคำสั่งเครื่องถัดไป
ติดตาม	พิมพ์ข้อมูลการติดตาม
tracehwp	เซตการติดตาม watchpoint ของฮาร์ดแวร์
tracei	เปิดการติดตาม
tskip	ข้ามจุดหยุดสำหรับเฮด
tstep	รัน thread สำหรับหนึ่งบรรทัดซอร์ส
tstepi	รัน thread สำหรับหนึ่งคำสั่งเครื่อง
tstop	ตั้งค่าการหยุด breakpoint ระดับซอร์สสำหรับเฮด
tstophwp	ตั้งค่าการหยุด watchpoint ของฮาร์ดแวร์ระดับเฮด
tstopi	ตั้งค่าการหยุด breakpoint ระดับคำสั่งสำหรับเฮด
ttrace	ตั้งค่าการติดตามระดับซอร์สสำหรับเฮด
ttracehwp	ตั้งค่าการติดตาม watchpoint ของฮาร์ดแวร์ระดับเฮด
ttracei	ตั้งค่าการติดตามระดับคำสั่งสำหรับเฮด
unalias	เอา alias ออก
unset	ลบตัวแปร
up	ย้ายฟังก์ชันปัจจุบันไปไวบนสแต็ก
use	ตั้งคารายการของไดเรกทอรีที่จะถูกค้นหาเมื่อ ค้นหาไฟล์เซตไฟล์ต้นฉบับ
whatis	แสดงการประกาศของคอมไพเนตโปรแกรมแอ็พพลิเคชัน
where	แสดงรายการของโพรซีเจอร์และฟังก์ชันแอ็คทีฟ
whereis	แสดงการระบุคุณสมบัติเพิ่มเติมของสัญลักษณ์ทั้งหมดซึ่งมีชื่อชื่อ ตรงกับตัวระบุที่กำหนด
which	แสดงการระบุคุณสมบัติแบบเต็มของตัวระบุที่กำหนด

## / Subcommand

/ [ *RegularExpression* [ / ] ]

คำสั่งย่อย / ค้นหา ไปข้างหน้าในไฟล์ต้นฉบับปัจจุบันตามรูปแบบที่ระบุโดยพารามิเตอร์ *RegularExpression* การป้อนคำสั่งย่อย / โดยไม่มีอาร์กิวเมนต์ทำให้คำสั่ง dbx ค้นหาไปข้างหน้าสำหรับนิพจน์ทั่วไปก่อนหน้านี้ การค้นหาย้อนกลับ เมื่อถึงจุดสิ้นสุดของไฟล์

## ตัวอย่าง

1. เมื่อต้องการค้นหาไปข้างหน้าในไฟล์ต้นฉบับปัจจุบัน เพื่อหาหมายเลข 12 ให้ป้อน:

/ 12

2. เมื่อต้องการทำซ้ำการค้นหาก่อนหน้านี้ให้ป้อน:

/

ดูที่คำสั่งย่อย ? (ค้นหา) และรูทีนย่อย regcmp

## ? Subcommand

? [ *RegularExpression* [ ? ] ]

? ค้นหา ย้อนกลับในไฟล์ต้นฉบับปัจจุบัน เพื่อหารูปแบบที่ระบุโดย พารามิเตอร์ *RegularExpression* การป้อน คำสั่งย่อย ? ที่ไม่มีอาร์กิวเมนต์ทำให้คำสั่ง dbx ทำการค้นหาย้อนกลับเพื่อหานิพจน์ทั่วไป ก่อนหน้านี้ การค้นหาย้อนกลับ เมื่อถึงจุดสิ้นสุดของไฟล์

## ตัวอย่าง

1. เมื่อต้องการค้นหาย้อนกลับในไฟล์ต้นฉบับปัจจุบัน สำหรับตัวอักษร z ให้อ่าน:  
?z
2. เมื่อต้องการทำซ้ำการค้นหาก่อนหน้านี้ให้อ่าน:  
?

ดูที่คำสั่งย่อย / (ค้นหา) และรูทีนย่อย `regcmp`

## คำสั่งย่อย `addcmd`

```
addcmd { Number... | all } "commands_string"
```

คำสั่งย่อย `addcmd` จะเพิ่ม คำสั่งย่อย `dbx` เข้ากับเหตุการณ์ที่ระบุ เหตุการณ์ที่ระบุนี้ จะรันเมื่อจุดหยุด จุดติดตาม หรือจุดเฝ้าดู สอดคล้องกับ เหตุการณ์นั้นถูกเรียกใช้ คำสั่งย่อย `dbx` สามารถถูกระบุผ่านพารามิเตอร์ `"commands_string"` ซึ่งคือกลุ่มของคำสั่งย่อย `dbx` ที่แยกโดยเซมิโคลอน (;) เหตุการณ์ซึ่งคำสั่งย่อย `dbx` จะถูกเพิ่ม สามารถถูกระบุผ่านพารามิเตอร์ `Number` หรือคำสั่งย่อย `dbx` สามารถถูกเพิ่มให้กับเหตุการณ์ทั้งหมดโดยใช้แฟล็ก `all`

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>all</code>	เพิ่มคำสั่งย่อย <code>dbx</code> ให้กับเหตุการณ์ทั้งหมด

## ตัวอย่าง

1. เมื่อต้องการเพิ่มคำสั่งย่อย `where` ให้กับหมายเลขเหตุการณ์ 1 ให้อ่าน:  
`addcmd 1 "where"`
2. เมื่อต้องการเพิ่มคำสั่งย่อย `registers` ให้กับหมายเลขเหตุการณ์ 2 ให้อ่าน:  
`addcmd 2 "registers"`
3. เมื่อต้องการเพิ่มคำสั่งย่อย `where` และ `registers` ให้กับ หมายเลขเหตุการณ์ 3 ให้อ่าน:  
`addcmd 3 "where;registers"`

ดูที่คำสั่งย่อย `clear`, คำสั่งย่อย `delcmd`, คำสั่งย่อย `delete`, คำสั่งย่อย `disable`, คำสั่งย่อย `enable`, คำสั่งย่อย `stop`, คำสั่งย่อย `status` และคำสั่งย่อย `trace` รวมทั้งดูที่ การตั้งค่า และการลบจุดหยุดใน *in General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย `alias`

```
alias [ Name [ [ (Arglist) ] String | Subcommand ] ]
```

คำสั่งย่อย `alias` สร้าง aliases สำหรับคำสั่งย่อย `dbx` พารามิเตอร์ `Name` เป็น alias ที่ถูกสร้างขึ้น พารามิเตอร์ `String` เป็นชุดของคำสั่งย่อย `dbx` ที่หลังจากการเรียกใช้คำสั่งย่อยนี้ สามารถถูกอ้างอิงไปถึง โดย `Name` ถ้าคำสั่งย่อย `alias` ถูกใช้โดยไม่มีพารามิเตอร์ จะแสดง aliases ปัจจุบันทั้งหมด

## ตัวอย่าง

1. เมื่อต้องการแทนที่ rr สำหรับ rerun ให้ป้อน:

```
alias rr rerun
```

2. เมื่อต้องการรันสองคำสั่งย่อย print n และ step เมื่อ printandstep ถูกพิมพ์ที่บรรทัดคำสั่ง ให้ป้อน:

```
alias printandstep "print n; step"
```

3. คำสั่งย่อย alias ยังสามารถถูกใช้เป็นส่วนช่วยเหลือแม่โครที่จำกัด ตัวอย่างเช่น:

```
(dbx) alias px(n) "set $hexints; print n; unset $hexints"  
(dbx) alias a(x,y) "print symname[x]->symvalue._n_n.name.Id[y]"  
(dbx) px(126)  
0x7e
```

ในตัวอย่างนี้ alias px พิมพ์ ค่าในเลขฐานสิบหกโดยไม่มีผลถาวรกับสภาวะแวดล้อม การดีบั๊ก

### คำสั่งย่อย assign

**assign** *Variable=Expression*

คำสั่งย่อย assign กำหนดค่าที่ระบุโดยพารามิเตอร์ *Expression* กับตัวแปรที่ระบุโดยพารามิเตอร์ *Variable*

### ตัวอย่าง

1. เมื่อต้องการกำหนดค่าของ 5 ให้กับตัวแปร x ให้ป้อน:

```
assign x = 5
```

2. เมื่อต้องการกำหนดค่าของตัวแปร y ให้กับตัวแปร x ให้ป้อน:

```
assign x = y
```

3. เมื่อต้องการกำหนดค่าอักขระ 'z' ให้กับตัวแปร z ให้ป้อน:

```
assign z = 'z'
```

4. เมื่อต้องการกำหนดค่าบูลีน false ให้กับตัวแปรชนิดโลจิคัล B ให้ป้อน:

```
assign B = false
```

5. เมื่อต้องการกำหนดสตริง "Hello World" ให้กับตัวชี้อักขระ Y ให้ป้อน:

```
assign Y = "Hello World"
```

6. เมื่อต้องการปิดใช้งานการตรวจสอบ การพิมพ์ ให้ตั้งค่าตัวแปรโปรแกรมดีบั๊ก dbx คือ \$unsafeassign โดยการป้อน:

```
set $unsafeassign
```

### ดูที่ การแสดงและการแก้ไขตัวแปร

### คำสั่งย่อย attribute

**attribute** [*AttributeName* ...]

คำสั่งย่อย attribute แสดงข้อมูลเกี่ยวกับ thread ผู้ใช้, mutex หรืออ็อบเจกต์แอดทริบิวต์ เส้นใยที่กำหนดโดยพารามิเตอร์ *AttributeName* ถ้าไม่มีการระบุพารามิเตอร์ อ็อบเจกต์แอดทริบิวต์ทั้งหมดจะถูกแสดง

สำหรับแต่ละอ็อบเจกต์แอดทริบิวต์ที่แสดง ข้อมูลต่อไปนี้ ถูกแสดง:

ไอเท็ม	คำอธิบาย
attr	ระบุชื่อเชิงสัญลักษณ์ของแอตทริบิวต์อ็อบเจกต์ในฟอร์ม \$aAttributeNumber
obj_addr	ระบุแอดเดรสของแอตทริบิวต์อ็อบเจกต์
type	ระบุชนิดของอ็อบเจกต์แอตทริบิวต์ ค่านี้สามารถเป็น thr, mutex หรือ cond สำหรับเซตผู้ใช้ mutexes และตัวแปรเงื่อนไขตามลำดับ
state	ระบุสถานะของแอตทริบิวต์อ็อบเจกต์ ค่านี้สามารถเป็น valid หรือ inval
stack	ระบุแอตทริบิวต์ stacksize ของ thread แอตทริบิวต์ อ็อบเจกต์
scope	ระบุแอตทริบิวต์ scope ของ thread แอตทริบิวต์อ็อบเจกต์ ค่านี้กำหนดขอบเขต contention ของเซต และกำหนด ชุดของเซตซึ่งต้อง contend เพื่อการประมวลผลรีซอร์ส ค่าเป็นได้ทั้ง sys หรือ pro สำหรับระบบหรือขอบเขตการแข่งขันของกระบวนการ
prio	ระบุแอตทริบิวต์ priority ของ thread แอตทริบิวต์อ็อบเจกต์
sched	ระบุแอตทริบิวต์ schedpolicy ของเซตแอตทริบิวต์ อ็อบเจกต์ แอตทริบิวต์ควบคุมนโยบายการจัดกำหนดการ และเป็นได้ทั้ง fifo, rr (round robin) หรือ other
p-shar	ระบุแอตทริบิวต์ process-shared ของ mutex หรืออ็อบเจกต์แอตทริบิวต์เงื่อนไข mutex หรือ condition เป็น process-shared ถ้าสามารถ ถูกเข้าถึงได้โดย threads ที่เป็นของกระบวนการที่ต่างกัน ค่า เป็นได้ทั้ง yes หรือ no
protocol	ระบุโปรโตคอลแอตทริบิวต์ของ mutex แอตทริบิวต์นี้ กำหนดผลของการถือค่า mutex บนระดับความสำคัญของ threads ค่าเป็นได้ทั้ง no_prio, prio หรือ protect
clock	ระบุแอตทริบิวต์ clock ของอ็อบเจกต์แอตทริบิวต์ เงื่อนไข แอตทริบิวต์นี้กำหนด clock ที่ต้องถูกใช้ เมื่อ thread ที่รอตัวแปรเงื่อนไขตามที่จะระบุในการ หมดเวลาใช้งาน ค่าเป็นได้ทั้ง realtime หรือ monotonic

### Notes:

- คำสั่งย่อย **print** ของดีบั๊กโปรแกรม **dbx** รู้จักชื่อแอตทริบิวต์เชิงสัญลักษณ์ และสามารถถูกใช้เพื่อแสดง สถานะของอ็อบเจกต์ที่เกี่ยวข้อง
- แอตทริบิวต์ที่มีอยู่ขึ้นกับการนำอ็อพชัน POSIX มาใช้

### ตัวอย่าง

- เมื่อต้องการแสดงข้อมูลเกี่ยวกับแอตทริบิวต์ทั้งหมด ให้ป้อน:

```
attribute
```

เอาต์พุตมีข้อมูลเหมือนดังนี้:

```
attr  obj_addr  type  state  stack  scope  prio
sched  p-shar
$a1   0x200035c8  mutex valid          no
$a2   0x20003628  cond  valid          no
$a3   0x200037c8  thr   valid  57344  sys    126  other
$a4   0x200050f8  thr   valid  57344  pro    126  other
```

- เมื่อต้องการแสดงข้อมูลเกี่ยวกับแอตทริบิวต์ 1 และ 3 ให้ป้อน:

```
attribute 1 3
```

เอาต์พุตมีข้อมูลเหมือนดังนี้:

```
attr  obj_addr  type  state  stack  scope  prio
sched  p-shar
$a1   0x200035c8  mutex valid          no
$a3   0x200037c8  thr   valid  57344  sys    126  other
```

ดูที่คำสั่งย่อย **condition**, คำสั่งย่อย **mutex**, คำสั่งย่อย **print** และคำสั่งย่อย **thread** สำหรับคำสั่ง **dbx**

และดูที่ การสร้าง Threads, การใช้ Mutexes และ การใช้ตัวแปรเงื่อนไข ใน *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย call

`call Procedure ( [ Parameters ] )`

คำสั่งย่อย `call` รัน โพรซีเจอร์ที่ระบุโดยพารามิเตอร์ `Procedure` โค้ดส่งคืนจะไม่ถูกพิมพ์ ถ้ามีการระบุพารามิเตอร์จะถูกส่งไปที่ โพรซีเจอร์ที่ถูกรัน

หมายเหตุ: คำสั่งย่อย `call` ไม่สามารถถูกใช้เพื่อเรียก ฟังก์ชันที่รับพารามิเตอร์เวกเตอร์

### ตัวอย่าง

เมื่อต้องการเรียกใช้คำสั่งขณะรันคำสั่ง `dbx` ให้ป้อน:

```
(dbx) call printf("hello")  
hello
```

### `printf` ส่งคืนสำเร็จ

## คำสั่งย่อย case

`case [ default | mixed | lower | upper ]`

คำสั่งย่อย `case` เปลี่ยน วิธีที่ตีบทโปรแกรม `dbx` แปลสัญลักษณ์ การจัดการดีพอลต์ของสัญลักษณ์ใช้ข้อมูลจากภาษาปัจจุบัน ถ้าภาษาปัจจุบันคือ C, C++ หรือโมระบุสัญลักษณ์ไม่ถูกเปลี่ยน ถ้าภาษาปัจจุบัน เป็น Fortran สัญลักษณ์จะถูกเปลี่ยนเป็นตัวพิมพ์เล็ก ใช้คำสั่งย่อยนี้ ถ้าสัญลักษณ์จำเป็นต้องถูกแปล ในแบบที่ไม่ตรงกับภาษาปัจจุบัน

การป้อนคำสั่งย่อย `case` ที่ไม่มีพารามิเตอร์แสดงโหมด `case` ปัจจุบัน

### แฟล็ก

ไอเท็ม	คำอธิบาย
<code>default</code>	แปรตามภาษาปัจจุบัน
<code>mixed</code>	ทำให้สัญลักษณ์ถูกแปลตามที่แสดงจริง
<code>lower</code>	ทำให้สัญลักษณ์ถูกแปลเป็นตัวพิมพ์เล็ก
<code>upper</code>	ทำให้สัญลักษณ์ถูกแปลเป็นตัวพิมพ์ใหญ่

### ตัวอย่าง

- เมื่อต้องการแสดงโหมด `case` ปัจจุบัน ให้ป้อน:  
`case`
- เมื่อต้องการกำหนดให้ `dbx` แปลสัญลักษณ์เป็นตามที่แสดง ให้ป้อน:  
`case mixed`
- เมื่อต้องการกำหนดให้ `dbx` แปลสัญลักษณ์เป็นตัวพิมพ์ใหญ่ ให้ป้อน:  
`case upper`

ดูที่ การเปลี่ยนตัวแปร เป็นตัวพิมพ์เล็กและตัวพิมพ์ใหญ่

## คำสั่งย่อย catch

`catch [ SignalNumber | SignalName ]`

คำสั่งย่อย `catch` เริ่มการจับสัญญาณที่ระบุ ก่อนที่สัญญาณจะถูกส่ง ไปที่โปรแกรมแอฟพลิเคชัน คำสั่งย่อยนี้มีประโยชน์ เมื่อแอฟพลิเคชัน โปรแกรมที่ถูกดีบั๊กจัดการสัญญาณเช่น อินเตอร์รัปต์ สัญญาณ ที่จะถูกจับสามารถถูกระบุโดยหมายเลขหรือ โดยชื่อโดยใช้พารามิเตอร์ `SignalNumber` หรือ `SignalName` ตามลำดับ ชื่อสัญญาณไม่สนใจตัวพิมพ์ และคำนำหน้า SIG เป็นทางเลือก ถ้าพารามิเตอร์ `SignalNumber` และ `SignalName` ไม่ถูกระบุ สัญญาณทั้งหมด ถูกดักโดยดีฟอลต์ ยกเว้นสัญญาณ `SIGHUP`, `SIGCLD`, `SIGALARM` และ `SIGKILL` ถ้าไม่มีการระบุอาร์กิวเมนต์ รายการปัจจุบันของสัญญาณที่จะถูกจับถูก แสดง

### ตัวอย่าง

1. เมื่อต้องการแสดงรายการของสัญญาณในปัจจุบัน ที่พบโดยคำสั่ง `dbx` ให้ป้อน:

```
catch
```

2. เมื่อต้องการจับสัญญาณ `SIGALARM`, ให้ป้อน:

```
catch SIGALARM
```

ดูที่คำสั่งย่อย `ignore` และการจัดการสัญญาณ

### คำสั่งย่อย `clear`

`clear SourceLine`

คำสั่งย่อย `clear` ลบการหยุดทั้งหมด ที่บรรทัดซอร์สโดยเฉพาะ พารามิเตอร์ `SourceLine` สามารถถูกระบุในสองรูปแบบ:

- เป็นจำนวนเต็ม
- เป็นสตริงชื่อไฟล์ตามด้วย : (โคลอน) และจำนวนเต็ม

### ตัวอย่าง

เมื่อต้องการลบจุดพักที่เซตที่บรรทัด 19 ให้ป้อน:

```
clear 19
```

คำสั่งย่อย `cleari` และคำสั่งย่อย `delete` รวมทั้งดูที่ การตั้งค่าและการลบจุดหยุดใน *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย `cleari`

`cleari Address`

คำสั่งย่อย `cleari` ลบจุดพักทั้งหมดที่แอดเดรสที่ระบุโดยพารามิเตอร์ `Address`

### ตัวอย่าง

1. เมื่อต้องการลบจุดพักที่เซตที่แอดเดรส `0x100001b4` ให้ป้อน:

```
cleari 0x100001b4
```

2. เมื่อต้องการลบจุดพักที่เซตที่ไพธอนแอดเดรส `main()` ให้ป้อน:

```
cleari &main
```

ดูที่คำสั่งย่อย `clear` คำสั่งย่อย `delete` และการตั้งค่าและการลบ จุดหยุดใน *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย `condition`

`condition` [ `wait` | `nowait` | `ConditionNumber` ... ]

คำสั่งย่อย `condition` แสดงข้อมูลเกี่ยวกับตัวแปรเงื่อนไข ถ้าไม่มีการระบุพารามิเตอร์ `ConditionNumber` คำสั่งย่อย `condition` แสดงข้อมูลเกี่ยวกับตัวแปรเงื่อนไขที่ระบุ ถ้าไม่มีการระบุแฟล็กหรือพารามิเตอร์ คำสั่งย่อย `condition` แสดง ตัวแปรเงื่อนไขทั้งหมด

ข้อมูลที่แสดงสำหรับแต่ละเงื่อนไข เป็นดังนี้:

ไอเท็ม	คำอธิบาย
<code>cv</code>	ระบุชื่อเชิงสัญลักษณ์ของตัวแปรเงื่อนไข ในรูปแบบ <code>\$cConditionNumber</code>
<code>obj_addr</code>	ระบุแอดเดรสหน่วยความจำของตัวแปรเงื่อนไข
<code>clock</code>	ระบุแอดเดรสทริกซ์ <code>clock</code> ของตัวแปรเงื่อนไข
<code>num_wait</code>	ระบุจำนวน <code>threads</code> ที่รอตัวแปรเงื่อนไข
<code>waiters</code>	แสดง <code>threads</code> ผู้ใช้ซึ่งรออยู่ในตัวแปรเงื่อนไข

หมายเหตุ: คำสั่งย่อย `print` ของดีบั๊กโปรแกรม `dbx` รู้จักชื่อตัวแปรเงื่อนไขเชิงสัญลักษณ์ และสามารถถูกใช้เพื่อแสดงสถานะของอ็อบเจกต์ที่เกี่ยวข้อง

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>wait</code>	แสดงตัวแปรเงื่อนไขซึ่งมี <code>threads</code> รออยู่
<code>nowait</code>	แสดงตัวแปรเงื่อนไขซึ่งไม่มี <code>threads</code> รออยู่

## ตัวอย่าง

- เมื่อต้องการแสดงข้อมูลเกี่ยวกับ ตัวแปรเงื่อนไขทั้งหมด ให้ป้อน:  
`condition`
- เมื่อต้องการแสดงข้อมูลเกี่ยวกับ ตัวแปรเงื่อนไขทั้งหมด ซึ่งมี `threads` รออยู่ ให้ป้อน:

```
condition wait
```

- เมื่อต้องการแสดงข้อมูลเกี่ยวกับตัวแปรเงื่อนไข 3 ให้ป้อน:

```
condition 3
```

เอาต์พุตมีข้อมูลเหมือนดังนี้:

```
cv      obj_addr      num_wait  waiters
$c3     0x20003290      0
```

ดูที่คำสั่งย่อย `attribute` คำสั่งย่อย `mutex`, คำสั่งย่อย `print` และคำสั่งย่อย `thread`

รวมทั้งดูที่ การใช้ตัวแปรเงื่อนไข ใน *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย `cont`

22 AIX เวอร์ชัน 7.2: ข้อมูลอ้างอิงคำสั่งวอลุ่ม 2, d - h

`cont [ SignalNumber | SignalName ]`

คำสั่งย่อย `cont` ดำเนินการเรียกใช้ แอปพลิเคชันโปรแกรมต่อจากจุดปัจจุบันจนกว่า โปรแกรมเสร็จสิ้นหรือถึงจุดพักอื่น ถ้า สัญญาณถูกระบุ โดยหมายเลขที่ระบุในพารามิเตอร์ `SignalNumber` หรือโดยชื่อที่ระบุใน พารามิเตอร์ `SignalName` ก็ตาม โปรแกรมยังคงทำงานต่อเสมือนว่า ได้รับสัญญาณ ชื่อสัญญาณไม่คำนึงถึงขนาดตัวพิมพ์ และส่วนนำหน้า `SIG` ไม่มีก็ได้ ถ้าไม่มี สัญญาณถูกระบุ โปรแกรมจะยังคงทำต่อเสมือนว่า ไม่ได้ถูกหยุดทำงาน

### ตัวอย่าง

1. เมื่อต้องการดำเนินโปรแกรมต่อจากจุดหยุดทำงานปัจจุบัน ให้ป้อน:  
`cont`
2. เมื่อต้องการดำเนินโปรแกรมต่อ เหมือนกับที่ได้รับสัญญาณ `SIGQUIT` ให้ป้อน:  
`cont SIGQUIT`

คำสั่งย่อย `detach` คำสั่ง `dbx` , คำสั่งย่อย `goto` สำหรับคำสั่ง `dbx` , คำสั่งย่อย `next` สำหรับคำสั่ง `dbx` คำสั่งย่อย `skip` สำหรับคำสั่ง `dbx` , คำสั่งย่อย `step` คำสั่ง `dbx`

### คำสั่งย่อย `corefile`

คำสั่งย่อย `corefile` แสดงข้อมูลจากส่วนหัวของไฟล์หลัก รวมทั้ง ชื่อไฟล์เรียกทำงาน ข้อมูลเวอร์ชันรูปแบบไฟล์หลัก แฟล็กระบุว่า ข้อมูลใดที่จะมีให้ สัญญาณที่ทำให้เกิดการหยุดชะงัก และ โหมดการเรียกใช้งานของกระบวนการที่พิมพ์ข้อมูลหลักออกมา

### คำสั่งย่อย `coremap`

`coremap [ stack | data | sdata | mmap | shm | loader ]`

คำสั่งย่อย `coremap` แสดงการแม็พของส่วนพื้นที่แอดเดรสเฉพาะ ถ้าคุณไม่ระบุชื่อขอบเขต คำสั่งย่อย `coremap` แสดงการแม็พ ที่มีอยู่ทั้งหมด

### ตัวอย่าง

1. เมื่อต้องการแสดงการแม็พของขอบเขตหน่วยความจำที่แบ่งใช้ให้ป้อน:  
`coremap shm`
2. เมื่อต้องการแสดงการแม็พของขอบเขตที่แม็พหน่วยความจำให้ป้อน:  
`coremap mmap`
3. เมื่อต้องการแสดงการแม็พของขอบเขตทั้งหมดที่อธิบายโดย รายการตัวโหลด ให้ป้อน:  
`coremap loader`
4. เมื่อต้องการแสดงการแม็พที่มีทั้งหมด ให้ป้อน:  
`coremap`

ดูที่คำสั่งย่อย `corefile`

### คำสั่งย่อย `delcmd`

`delcmd EventNumber { Number... | all }`

คำสั่งย่อย `delcmd` ลบคำสั่งย่อย `dbx` ที่เชื่อมโยงกับเหตุการณ์ที่ระบุ คำสั่งย่อย `dbx` ที่จะถูกลบสามารถระบุผ่านพารามิเตอร์ `Number` หรือคำสั่งย่อยทั้งหมด `dbx` ที่เชื่อมโยงกับเหตุการณ์ที่ระบุ สามารถถูกลบออกโดยใช้แฟล็ก `all` พารามิเตอร์ `EventNumber` ระบุเหตุการณ์ซึ่งคำสั่งย่อย `dbx` จะ ลบ

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>all</code>	ลบคำสั่งย่อย <code>dbx</code> ทั้งหมดที่เชื่อมโยงกับ เหตุการณ์ที่ระบุ

## ตัวอย่าง

1. เมื่อต้องการลบคำสั่งย่อย `dbx` ทั้งหมดจากเหตุการณ์ หมายเลข 2 ให้ป้อน:  

```
delcmd 2 all
```
2. เมื่อต้องการลบคำสั่งย่อย `dbx` หมายเลข 1 จากหมายเลขเหตุการณ์ 3 ให้ป้อน:  

```
delcmd 3 1
```
3. เมื่อต้องการลบคำสั่งย่อย `dbx` หมายเลข 1 และ 2 จากหมายเลข เหตุการณ์ 2 ให้ป้อน:  

```
delcmd 2 1 2
```

ดูที่คำสั่งย่อย `addcmd`, คำสั่งย่อย `clear`, คำสั่งย่อย `delete`, คำสั่งย่อย `disable`, คำสั่งย่อย `enable`, คำสั่งย่อย `stop`, คำสั่งย่อย `status`, คำสั่งย่อย และคำสั่งย่อย `trace` รวมทั้งดูที่ การตั้งค่า และการลบจุดหยุดใน *in General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย `delete`

```
delete { Number ... | all | tskip [for $tthreadnumber] }
```

คำสั่งย่อย `delete` ลบการติดตามและการหยุด ออกจากแอ็พพลิเคชันโปรแกรมและจำนวน `tskip` สำหรับ thread การติดตาม และการหยุดที่จะถูกลบสามารถระบุผ่าน พารามิเตอร์ `Number` หรือการติดตามและ การหยุดทั้งหมดสามารถถูกลบโดยใช้แฟล็ก `all` ใช้คำสั่งย่อย `status` เพื่อแสดงตัวเลขที่ถูกเชื่อมโยง โดยโปรแกรมดีบั๊ก `dbx` ที่มีการติดตาม หรือการหยุด

จำนวน `tskip` ที่เหลือซึ่งถูกเซตโดยใช้คำสั่งย่อย `tskip` สำหรับ thread สามารถถูกลบโดยใช้แฟล็ก `tskip` ใช้คำสั่งย่อย `status` เพื่อแสดงจำนวน `tskip` ของ thread ที่เหลือ ถ้าไม่มีการระบุ thread จะใช้ thread ปัจจุบัน

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>all</code>	ลบการติดตามและการหยุดทั้งหมด
<code>for \$t threadnumber</code>	ระบุหมายเลข thread

## ตัวอย่าง

1. เมื่อต้องการลบการติดตามและการหยุดทั้งหมดจากแอ็พพลิเคชันโปรแกรม ให้ป้อน:  

```
delete all
```
2. เมื่อต้องการลบการติดตามและการหยุดสำหรับหมายเลขเหตุการณ์ 4 ให้ป้อน:  

```
delete 4
```
3. เมื่อต้องการลบจำนวน `tskip` สำหรับ thread 3 ให้ป้อน:

**24** AIX เวอร์ชัน 7.2: ข้อมูลอ้างอิงคำสั่งวอลุ่ม 2, d - h

delete tskip for \$t3

4. เมื่อต้องการลบจำนวน **tskip** สำหรับ thread ปัจจุบัน ให้ป้อน:

delete tskip

ดูที่คำสั่งย่อย **clear** , คำสั่งย่อย **cleari** , คำสั่งย่อย **status** , คำสั่งย่อย **tskip** และ การตั้งค่าและการลบจุดพัก ใน *in General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย detach

**detach** [ *SignalNumber* | *SignalName* ]

คำสั่งย่อย **detach** ดำเนินการใช้งาน แอปพลิเคชันโปรแกรมต่อและออกจากดีบั๊กโปรแกรม สัญญาณ สามารถถูกระบุโดย:

- ชื่อให้ใช้พารามิเตอร์ *SignalName*
- ตัวเลข ให้ใช้พารามิเตอร์ *SignalNumber*

ชื่อสัญญาณไม่คำนึงถึงขนาดตัวพิมพ์ และส่วนนำหน้า **SIG** ไม่มีก็ได้

ถ้าสัญญาณถูกระบุ โปรแกรมยังคงทำต่อเสมือนว่าได้รับสัญญาณนั้น ถ้าไม่มีสัญญาณถูกระบุ โปรแกรมจะยังคงทำต่อเสมือนว่า ไม่มีการหยุดทำงานเกิดขึ้น

### ตัวอย่าง

1. เมื่อต้องการดำเนินการทำงานของแอปพลิเคชันต่อแล้วออกจาก **dbx**, ให้ป้อน:

detach

2. เมื่อต้องการออกจาก **dbx** และดำเนินการทำงานของแอปพลิเคชันเหมือนกับได้รับสัญญาณ **SIGREQUEST**, ให้ป้อน:

detach SIGREQUEST

โปรดดูที่ การใช้โปรแกรมดีบั๊ก **dbx**

### คำสั่งย่อย disable

**disable** { *Number ... all* }

คำสั่งย่อย **disable** ปิดใช้งานการติดตาม และการหยุดทำงาน ที่เชื่อมโยงกับเหตุการณ์ดีบั๊ก การติดตามและการหยุดที่จะถูกปิดใช้งานสามารถถูกระบุผ่าน พารามิเตอร์ *Number* หรือการติดตามและการหยุดทั้งหมดสามารถถูกปิดใช้งานโดยใช้แฟล็ก **all** ใช้คำสั่งย่อย **status** เพื่อแสดงหมายเลขของเหตุการณ์ ที่เชื่อมโยงโดยโปรแกรมดีบั๊ก **dbx** ที่มีการติดตาม หรือการหยุด

### แฟล็ก

ไอเท็ม	คำอธิบาย
all	ลบการติดตามและการหยุดทั้งหมด

### ตัวอย่าง

1. เมื่อต้องการปิดการติดตามและการหยุดทั้งหมดจากแอปพลิเคชันโปรแกรม ให้ป้อน:

disable all

2. เมื่อต้องการปิดการติดตามและการหยุดทำงาน สำหรับหมายเลขเหตุการณ์ 4 ให้พิมพ์:

disable 4

สำหรับข้อมูลเพิ่มเติม ดูที่ คำสั่งย่อย enable, คำสั่งย่อย delete และ คำสั่งย่อย status

รวมทั้ง ดูที่ การตั้งค่าและการลบบจุดหยุด in *General Programming Concepts: Writing and Debugging Programs*

คำสั่งย่อย display memory

```
{ Address,Address/|Address/[ Count ] } [ Mode ] [ >File ]
```

คำสั่งย่อย display memory ซึ่งไม่มีคีย์เวิร์ดเพื่อเริ่มคำสั่ง แสดงส่วนของ หน่วยความจำที่ควบคุมโดยปัจจัยต่อไปนี้:

ขอบเขตของหน่วยความจำที่แสดงถูกควบคุมโดย การระบุ:

- สองพารามิเตอร์ Address โดยที่บรรทัดทั้งหมดระหว่างสองแอดเดรสถูกแสดง หรือ
- หนึ่งพารามิเตอร์ Address ซึ่งเริ่มแสดงข้อมูลและ Count ที่ กำหนดจำนวนบรรทัดที่แสดงจาก Address

ระบุการกำหนดแอดเดรสเชิงสัญลักษณ์โดยนำหน้าชื่อด้วย & (แอมเปอร์แซนด์) Addresses เป็นได้ตั้งนิพจน์ที่สร้างจากแอดเดรสอื่น และโอเปอเรเตอร์ + (เครื่องหมายบวก), - (เครื่องหมายลบ) และ \* (indirection) นิพจน์ที่ปิดด้วยวงเล็บถูกแปลเป็นแอดเดรส

- รูปแบบที่หน่วยความจำถูกแสดง ถูกควบคุมโดยพารามิเตอร์ Mode ดีฟอลต์ สำหรับพารามิเตอร์ Mode คือโหมด ปัจจุบัน ค่าเริ่มต้นของ Mode คือ X โหมดที่เป็นไปได้รวมถึง:

ไอเท็ม	คำอธิบาย
b	พิมพ์ไบต์ในแบบฐานแปด
c	พิมพ์ไบต์เป็นอักขระ
d	พิมพ์ short word เป็นทศนิยม
D	พิมพ์ long word เป็นทศนิยม
Df	พิมพ์ตัวเลข decimal float ความเที่ยงตรงสองเท่า
DDf	พิมพ์ตัวเลข decimal float ความเที่ยงตรงสี่เท่า
f	พิมพ์จำนวนจริงความเที่ยงตรงหนึ่งเท่า
g	พิมพ์จำนวนจริงความเที่ยงตรงสองเท่า
h	พิมพ์ไบต์ในแบบเลขฐานสิบหก
Hf	พิมพ์ตัวเลข decimal float ความเที่ยงตรงหนึ่งเท่า
i	พิมพ์คำสั่งเครื่อง
lld	พิมพ์ตัวเลขทศนิยมที่มีเครื่องหมาย 8 ไบต์
llu	พิมพ์ตัวเลขทศนิยมที่ไม่มีเครื่องหมาย 8 ไบต์
llx	พิมพ์ตัวเลขฐานสิบหกที่ไม่มีเครื่องหมาย 8 ไบต์
llo	พิมพ์ตัวเลขฐานแปดที่ไม่มีเครื่องหมาย 8 ไบต์
o	พิมพ์ short word ในรูปแบบฐานแปด
O	พิมพ์ long word เป็นฐานแปด
p	พิมพ์แอดเดรส/ตัวชี้ในรูปแบบเลขฐานสิบหก
q	พิมพ์ตัวเลข extended-precision floating-point
s	พิมพ์สตริงของอักขระที่ปิดท้ายด้วย null ไบต์
x	พิมพ์ short word เป็นเลขฐานสิบหก
X	พิมพ์ long word เป็นเลขฐานสิบหก

แฟล็ก

ไอเท็ม คำอธิบาย  
>File เปลี่ยนทิศทางเอาต์พุตไปที่ไฟล์ที่ระบุ

### ตัวอย่าง

1. เมื่อต้องการแสดงหนึ่ง long word ของเนื้อหาหน่วยความจำ ในแบบเลขฐานสิบหก เริ่มต้นที่แอดเดรส 0x3ffffe460, ให้ป้อน:  
`0x3ffffe460 / X`
2. เมื่อต้องการแสดงสองไบต์ของเนื้อหาหน่วยความจำ เป็นอักขระ เริ่มที่ตัวแปรแอดเดรส y ให้ป้อน:  
`&y / 2c`
3. เมื่อต้องการแสดงอิลิเมนต์ตัวที่หกถึงแปดของสตริงอักขระ Fortran a\_string ให้ป้อน:  
`&a_string + 5, &a_string + 7/c`

ดูที่ การตรวจสอบ แอดเดรสหน่วยความจำ in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย down

`down [ Count ]`

คำสั่งย่อย `down` ย้าย ฟังก์ชันปัจจุบันลงเป็น จำนวนระดับ `Count` ของสแต็ก ฟังก์ชันปัจจุบันถูกใช้เพื่อกำหนดชื่อ ดีฟอลต์สำหรับพารามิเตอร์ `Count` คือหนึ่ง

### ตัวอย่าง

1. เมื่อต้องการย้ายลงหนึ่งระดับในสแต็ก ให้ป้อน:  
`down`
2. เมื่อต้องการย้ายลงสามระดับในสแต็ก ให้ป้อน:  
`down 3`

ดูที่คำสั่งย่อย `up` คำสั่งย่อย `where` subcommand, และ การแสดงการติดตามสแต็ก in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย dump

`dump [ Procedure | "PATTERN" ] [ >File ]`

คำสั่งย่อย `dump` แสดง ชื่อและค่าของตัวแปรทั้งหมดในโปรซีเจอร์ที่ระบุ หรือตัวแปรที่ตรงกับรูปแบบที่ระบุ ถ้าพารามิเตอร์ `Procedure` เป็นจุด (.) ตัวแปรที่แอสคีย์ทั้งหมดจะแสดงขึ้น ถ้าพารามิเตอร์ `Procedure` หรือ "PATTERN" ถูกระบุ โปรซีเจอร์ปัจจุบันจะถูกใช้ พารามิเตอร์ "PATTERN" เป็นนิพจน์ wildcard ที่มี \*, ? และ [ ] meta-characters เมื่อใช้ "PATTERN" จะแสดงสัญลักษณ์ที่ตรงกันทั้งหมดในขอบเขตโกลบอล (จาก โปรซีเจอร์ทั้งหมด) ถ้าใช้แฟล็ก >File เอาต์พุตถูกเปลี่ยนทิศทางไปยังไฟล์ที่ระบุ

### แฟล็ก

ไอเอ็ม คำอธิบาย  
>File เปลี่ยนทิศทางเอาต์พุตไปที่ไฟล์ที่ระบุ

### ตัวอย่าง

1. เมื่อต้องการแสดงชื่อและค่าของตัวแปรในโปรแกรมปัจจุบัน ให้ป้อน:  
dump
2. เมื่อต้องการแสดงชื่อและค่าของตัวแปรในโปรแกรม add\_count ให้ป้อน:  
dump add\_count
3. เมื่อต้องการแสดงชื่อและค่าของตัวแปรเริ่มต้นจากอักขระ s ให้ป้อน:  
dump "s\*"
4. เมื่อต้องการเปลี่ยนทิศทางชื่อและค่าของตัวแปรในโปรแกรมปัจจุบันไปที่ไฟล์ var.list ให้ป้อน:  
dump > var.list

ดูที่ การแสดงและการแก้ไขตัวแปร in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย edit

**edit** [ Procedure | File ]

คำสั่งย่อย **edit** เริ่มทำงานเอดิเตอร์บนไฟล์ที่ระบุ ไฟล์อาจถูกระบุ ผ่านพารามิเตอร์ *File* หรือโดยการระบุพารามิเตอร์ *Procedure* ที่มีเอดิเตอร์เริ่มทำงานบนไฟล์ที่มีโปรแกรมที่นั่น ถ้าไม่ระบุไฟล์ เอดิเตอร์เริ่มทำงานบนซอร์สไฟล์ ปัจจุบัน ดีฟอลต์คือเอดิเตอร์ vi แทนค่าดีฟอลต์โดย การรีเซ็ตตัวแปรสถานะแวดล้อม EDITOR เป็นชื่อของ เอดิเตอร์ที่ต้องการ

### ตัวอย่าง

1. เมื่อต้องการเริ่มเอดิเตอร์ในไฟล์ต้นฉบับ ปัจจุบัน ให้ป้อน:  
edit
2. เมื่อต้องการเริ่มเอดิเตอร์บนไฟล์ main.c ให้ป้อน:  
edit main.c
3. เมื่อต้องการเริ่มเอดิเตอร์บนไฟล์ที่มีโปรแกรม do\_count() ให้ป้อน:  
edit do\_count

ดูที่คำสั่งย่อย list, คำสั่ง vi หรือ vedit

### คำสั่งย่อย enable

**enable** { Number ... all }

คำสั่งย่อย **enable** เปิดใช้งานการติดตาม และการหยุดทำงาน ที่เชื่อมโยงกับเหตุการณ์ดีบั๊ก การติดตามและการหยุดที่จะถูกเปิดใช้งานสามารถถูกระบุผ่าน พารามิเตอร์ *Number* หรือการติดตามและ การหยุดทั้งหมดสามารถถูกเปิดใช้โดยใช้แฟล็ก **all** ใช้คำสั่งย่อย **status** เพื่อแสดงหมายเลขของเหตุการณ์ ที่เชื่อมโยงโดยโปรแกรมดีบั๊ก **dbx** ที่มีการติดตาม หรือการหยุด

### แฟล็ก

ไอเท็ม            คำอธิบาย  
all                ลบการติดตามและการหยุดทั้งหมด

### ตัวอย่าง

1. เมื่อต้องการเปิดการติดตามและการหยุดทั้งหมดจากแอฟพลิเคชันโปรแกรม ให้ป้อน:  
`enable all`
2. เมื่อต้องการเปิดการติดตามและการหยุดทำงาน สำหรับหมายเลขเหตุการณ์ 4 ให้พิมพ์:  
`enable 4`

สำหรับข้อมูลเพิ่มเติม ดูที่ คำสั่งย่อย `disable`, คำสั่งย่อย `delete`, คำสั่งย่อย `status`

รวมทั้ง ดูที่ การตั้งค่าและการลบจุดหยุด in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย `fd`

`fd [ raw ] [ start [ end ] ]`

คำสั่งย่อย `fd` แสดงข้อมูลไฟล์ descriptor การใช้ตัวเลือก `raw` ทำให้ เอาต์พุตถูกแสดงในรูปแบบ raw hex อาร์กิวเมนต์ทางเลือกอื่น รวมถึงดัชนี `start` และ `end` ถ้าไม่มีการกำหนดดัชนี ข้อมูลเกี่ยวกับไฟล์ descriptors ทั้งหมด ที่มีจะถูกแสดง ใช้หนึ่งดัชนีแสดงหนึ่งไฟล์ descriptor; สองดัชนีเพื่อระบุขอบเขต

### ตัวอย่าง

1. เมื่อต้องการดูข้อมูลเกี่ยวกับ descriptors ไฟล์ทั้งหมดในรูปแบบ hex ให้พิมพ์:  
`fd raw`
2. เมื่อต้องการดูข้อมูลเกี่ยวกับ descriptors ไฟล์ในช่อง 3 ถึง 5 ให้พิมพ์:  
`fd 3 5`

### คำสั่งย่อย `file`

`file [ File ]`

คำสั่งย่อย `file` เปลี่ยนไฟล์ต้นฉบับปัจจุบัน เป็นไฟล์ที่ระบุโดยพารามิเตอร์ `File` ไม่มีการเขียนไปที่ไฟล์นั้น พารามิเตอร์ `File` สามารถระบุชื่อพาธแบบเต็มไปที่ไฟล์ ถ้าพารามิเตอร์ `File` ไม่ระบุพาธ โปรแกรม `dbx` จะพยายามค้นหาไฟล์โดยการค้นหาพาธที่ใช้งาน ถ้าไม่ระบุพารามิเตอร์ `File` คำสั่งย่อย `file` แสดงชื่อของไฟล์ต้นฉบับปัจจุบัน คำสั่งย่อย `file` ยังแสดงชื่อพาธแบบเต็มหรือพาธสัมพัทธ์ของไฟล์ ถ้ารู้จัก พาธ

### ตัวอย่าง

1. เมื่อต้องการเปลี่ยนไฟล์ต้นฉบับปัจจุบันเป็นไฟล์ `main.c` ให้ป้อน:  
`file main.c`
2. เมื่อต้องการแสดงชื่อของไฟล์ต้นฉบับปัจจุบัน ให้ป้อน:  
`file`

ดูที่คำสั่งย่อย **func** รวมทั้ง ดูที่ การเปลี่ยนแปลงไฟล์หรือโพรซีเจอร์ ปัจจุบัน และการแสดง ไฟล์ ปัจจุบัน in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย **frame**

**frame** [ *num* ]

คำสั่งย่อย **frame** เปลี่ยนฟังก์ชันปัจจุบันเป็นฟังก์ชัน ที่ตรงกับหมายเลขเฟรมสแต็กที่ระบุ *num* ฟังก์ชันปัจจุบันถูกใช้เพื่อกำหนดชื่อ การกำหนดหมายเลขของสแต็กเฟรมเริ่มตั้งแต่ สแต็กเฟรมของแอสคทีฟฟังก์ชันขณะนี้ (ฟังก์ชันเฟรมที่ แอสคทีฟในขณะนี้จะมีหมายเลขเป็น 0 เสมอ) ถ้ามี *n* เฟรม เฟรมของฟังก์ชัน **main** จะมีหมายเลขเป็น *n*-1 เมื่อไม่มีการระบุหมายเลขเฟรม ข้อมูลเกี่ยวกับฟังก์ชัน ที่เชื่อมโยงกับเฟรมปัจจุบันถูกแสดง

### ตัวอย่าง

1. เมื่อต้องการย้ายหมายเลขเฟรม 2 ให้ป้อน:

```
frame 2
```

2. เมื่อต้องการแสดงฟังก์ชันปัจจุบันบนสแต็ก ให้ป้อน:

```
frame
```

ดูที่คำสั่งย่อย **up** และ **down** รวมทั้ง ดูที่ การเปลี่ยนแปลง ไฟล์และโพรซีเจอร์ ปัจจุบัน และการแสดงการติดตามสแต็ก in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย **func**

**func** [ *Procedure* ]

คำสั่งย่อย **func** เปลี่ยน ฟังก์ชันปัจจุบันเป็นโพรซีเจอร์หรือฟังก์ชันที่ระบุโดยพารามิเตอร์ *Procedure* ถ้าพารามิเตอร์ *Procedure* ไม่ถูกระบุ ฟังก์ชันปัจจุบันที่เป็นดีฟอลต์ถูกแสดง การเปลี่ยน ฟังก์ชันปัจจุบัน เป็นไฟล์ต้นฉบับปัจจุบันทางอ้อมเป็น ไฟล์ที่มีฟังก์ชันใหม่ ขอบเขตปัจจุบันที่ใช้สำหรับการกำหนดชื่อถูกเปลี่ยนไปเช่นกัน

### ตัวอย่าง

1. เมื่อต้องการเปลี่ยนฟังก์ชันปัจจุบันเป็นโพรซีเจอร์ **do\_count** ให้ป้อน:

```
func do_count
```

2. เมื่อต้องการแสดงชื่อของฟังก์ชันปัจจุบัน ให้ป้อน:

```
func
```

ดูที่คำสั่งย่อย **file** รวมทั้ง ดูที่ การเปลี่ยนแปลงไฟล์หรือโพรซีเจอร์ ปัจจุบัน in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย **goto**

**goto** *SourceLine*

คำสั่งย่อย **goto** ทำให้ บรรทัดซอร์สที่ระบุถูกรันถัดไป โดยปกติ บรรทัดซอร์ส ต้องอยู่ในฟังก์ชันเดียวกับบรรทัดซอร์สปัจจุบัน เมื่อต้องการเขียนทับ ข้อจำกัดนี้ ให้ใช้คำสั่งย่อย **set** กับแฟล็ก **\$unsafegoto**

## ตัวอย่าง

เมื่อต้องการเปลี่ยนบรรทัดถัดไปที่จะถูกดำเนินการเป็นบรรทัด 6 ให้อ่าน:

```
goto 6
```

ดูที่คำสั่งย่อย `cont` , คำสั่งย่อย `gotoi` , และคำสั่งย่อย `set`

## คำสั่งย่อย `gotoi`

`gotoi Address`

คำสั่งย่อย `gotoi` เปลี่ยน program counter แอดเดรสเป็นแอดเดรสที่ระบุโดยพารามิเตอร์ `Address`

## ตัวอย่าง

เมื่อต้องการเปลี่ยน program counter แอดเดรสเป็นแอดเดรส `0x100002b4` ให้อ่าน:

```
gotoi 0x100002b4
```

ดูที่คำสั่งย่อย `goto`

## คำสั่งย่อย `handler`

```
handler { atfork | cancel_cleanup [ all | pthread id ] }
```

คำสั่งย่อย `handler` แสดงข้อมูล เกี่ยวกับ `atfork` หรือ cancellation cleanup handlers ที่รีจิสเตอร์โดยใช้ `pthread_atfork` และ `pthread_cleanup_push` ตามลำดับ การใช้ชื่อพจนานุกรม `atfork`, ชื่อ ของรูทีนที่รีจิสเตอร์เป็น `pre`, `parent` และ `child` `atfork` handlers ถูกแสดง (โดยมีอาร์กิวเมนต์ตามลำดับในกรณีของ `atfork` handlers ที่ทำงานได้กับระบบที่ไม่ใช่ `posix`) ชื่อพจนานุกรม `cancel_cleanup` ทำให้การแสดงผล cancellation cleanup handlers ที่รีจิสเตอร์ทั้งหมด ที่มี พารามิเตอร์ `pthread id` ทางเลือกระบุ `pthread` เฉพาะ หรือ `all` ระบุ `pthread` ทั้งหมด ถ้าไม่มีค่าใดถูกกำหนด cancellation cleanup handlers สำหรับ `pthread` ปัจจุบันจะแสดง ถ้ามีค่าอยู่

## ตัวอย่าง

1. เมื่อต้องการดูข้อมูลเกี่ยวกับ `atfork` handlers ที่รีจิสเตอร์ทั้งหมด ให้พิมพ์:

```
handler atfork
```

2. เมื่อต้องการดูข้อมูลเกี่ยวกับ cancellation cleanup handlers ที่รีจิสเตอร์สำหรับ `pthread` ปัจจุบัน ให้พิมพ์:

```
handler cancel_cleanup
```

3. เมื่อต้องการดูข้อมูลเกี่ยวกับ cancellation cleanup handlers ที่รีจิสเตอร์สำหรับอ็อบเจกต์ `pthread` ที่อ้างอิงเป็น `$t2` ให้พิมพ์:

```
handler cancel_cleanup 2
```

## คำสั่งย่อย `help`

```
help [ Subcommand | Topic ]
```

คำสั่งย่อย **help** แสดงข้อมูล วิธีใช้สำหรับคำสั่งย่อย **dbx** หรือหัวข้อ ขึ้นกับพารามิเตอร์ที่คุณระบุ ป้อนคำสั่งย่อย **help** ด้วยพารามิเตอร์ **Subcommand** แสดง ประโยคไวยากรณ์และคำอธิบายของคำสั่งย่อยที่ระบุ การป้อน คำสั่งย่อย **help** กับพารามิเตอร์ **Topic** แสดงคำอธิบายรายละเอียดของหัวข้อที่ระบุ คุณไม่จำเป็นต้องจัดเตรียมสตริงหัวข้อทั้งหมดด้วยคำสั่งย่อย **help** โปรแกรม **dbx** สามารถจดจำหัวข้อได้ ถ้าคุณกำหนด ชับสตริงเริ่มจากจุดเริ่มต้นของหัวข้อ หัวข้อต่อไปนี้มีอยู่:

ไอเท็ม	คำอธิบาย
startup	แสดงอ็อปชันเริ่มทำงานของ <b>dbx</b>
execution	แสดงคำสั่งย่อย <b>dbx</b> ที่สัมพันธ์กับการเรียกใช้โปรแกรม
จุดหยุด	แสดงคำสั่งย่อย <b>dbx</b> ที่สัมพันธ์กับจุดพัก และการติดตาม
ไฟล์	แสดงคำสั่งย่อย <b>dbx</b> สำหรับการเข้าถึง ไฟล์ต้นฉบับ
data	แสดงคำสั่งย่อย <b>dbx</b> สำหรับการเข้าถึง ตัวแปรโปรแกรมและข้อมูล
machine	แสดงรายละเอียดของคำสั่งย่อย <b>dbx</b> สำหรับการดีบักกระดุมเครื่อง
สถานะแวดล้อม	แสดงคำสั่งย่อย <b>dbx</b> สำหรับการตั้งค่าคอนฟิกูเรชันและสถานะแวดล้อม <b>dbx</b>
threads	แสดงคำสั่งย่อย <b>dbx</b> สำหรับการเข้าถึง อ็อบเจกต์ที่สัมพันธ์กับ thread
นิพจน์	อธิบายไวยากรณ์นิพจน์และโอเปอเรเตอร์ <b>dbx</b>
scope	อธิบายวิธีที่ <b>dbx</b> กำหนดชื่อจากขอบเขตที่ต่างกัน
set_variables	แสดงตัวแปรดีบัก <b>dbx</b> พร้อมกับ รายละเอียดการใช้งาน
usage	แสดงคำสั่งย่อย <b>dbx</b> ทั้งหมดพร้อมกับ รายละเอียดย่อ

## ตัวอย่าง

- เมื่อต้องการแสดงคำสั่งย่อยและหัวข้อ **dbx** ที่มีอยู่ทั้งหมด ให้ป้อน:  
วิธีใช้
- เมื่อต้องการแสดงคำอธิบาย ของคำสั่งย่อย **dbx** คือ **list** ให้ป้อน:  
`help list`
- เมื่อต้องการแสดงคำอธิบาย ของหัวข้อ **dbx** คือ **set\_variables** ให้ป้อน:  
`help set_variables`

## คำสั่งย่อย ignore

**ignore** [ *SignalNumber* | *SignalName* ]

คำสั่งย่อย **ignore** หยุดการจับสัญญาณที่ระบุ ก่อนที่สัญญาณจะถูกส่ง ไปที่โปรแกรมแ็พพลิเคชัน คำสั่งย่อยนี้มีประโยชน์ เมื่อแ็พพลิเคชัน โปรแกรมที่ถูกดีบักจัดการสัญญาณเช่น อินเทอร์รัปต์

สัญญาณที่จะถูกจับสามารถถูกระบุได้โดย:

- หมายเลข ด้วยพารามิเตอร์ *SignalNumber*
- ชื่อ ด้วยพารามิเตอร์ *SignalName*

ชื่อสัญญาณไม่คำนึงถึงขนาดตัวพิมพ์ คำนำหน้า **SIG** เป็นทางเลือก

ถ้า พารามิเตอร์ *SignalNumber* และ *SignalName* ถูกระบุ สัญญาณทั้งหมดยกเว้นสัญญาณ **SIGHUP**, **SIGCLD**, **SIGALRM** และ **SIGKILL** จะถูกดักโดยดีฟอลต์ ดีบักโปรแกรม **dbx** ไม่สามารถละเว้นสัญญาณ **SIGTRAP** ถ้ามาจาก กระบวนการภายนอกดีบักเกอร์ ถ้าไม่มีอาร์กิวเมนต์ถูกระบุ รายการสัญญาณที่ถูกข้ามในขณะนี้จะแสดง

## ตัวอย่าง

เมื่อต้องการให้ **dbx** ละเว้น สัญญาณการเตือนหมดเวลาใช้งาน ที่ส่งไปที่แ็พพลิเคชันโปรแกรม ให้ป้อน:

ignore alrm

ดูที่คำสั่งย่อย `catch` และดูที่ Handling Signals in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย `kthread`

`kthread [ raw ] [ info | ru ] [ tid ]`

คำสั่งย่อย `kthread` แสดงข้อมูล เกี่ยวกับเรดเคอร์เนล การใช้ชื่อพจนานุกรม `raw` ทำให้แสดงเอาต์พุตทั้งหมดในรูปแบบ hex ไม่ว่าเอาต์พุตนั้น สามารถแสดงในรูปแบบที่ผู้ใช้เข้าใจได้มากกว่าได้หรือไม่ การไม่ใช้อาร์กิวเมนต์ ข้อมูลสรุปเกี่ยวกับเรดเคอร์เนลทั้งหมดจะถูกพิมพ์ ระบุ ID เรดที่เป็นตัวเลขที่ทำให้คำสั่ง `dbx` แสดงข้อมูล เกี่ยวกับเรดเดียว ชื่อพจนานุกรม `info` สร้างเอาต์พุตรายละเอียดเพิ่มเติม เกี่ยวกับเรดจากโครงสร้างเรดผู้ใช้ การใช้ชื่อพจนานุกรม `ru` แสดงสมาชิกข้อมูล `ti_ru` ซึ่งมีข้อมูลการใช้รีซอร์ส

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับเรดผู้ใช้ ดูที่ คำสั่งย่อย `thread`

## ตัวอย่าง

1. เมื่อต้องการค้นหาข้อมูลเกี่ยวกับเรดที่ขณะนี้กำลังรันอยู่ อันดับแรกคุณต้องระบุข้อมูลเกี่ยวกับเรดทั้งหมดโดยพิมพ์ข้อความต่อไปนี้บนบรรทัดรับคำสั่ง:

```
kthread
```

เรด ที่รัน (หรือสามารถรันได้) ก่อนที่คำสั่ง `dbx` จะหยุดกระบวนการจะถูกทำเครื่องหมายด้วยเครื่องหมายดอกจัน เลือก ID เรดที่ต้องการจากเอาต์พุตและพิมพ์:

```
kthread info tid
```

2. เมื่อต้องการดูข้อมูลรีซอร์สในแบบเลขฐานสิบหกเกี่ยวกับเรดทั้งหมดให้พิมพ์:

```
kthread raw ru
```

## limitbp Subcommand

`limitbp ( bp1, Limit ) [ ( bp2, [ + ] Limit ) ... ]`

คำสั่งย่อย `limitbp` สั่งให้คำสั่ง `dbx` หยุดการรันโปรแกรมดีบั๊ก ต่อเมื่อจุดหยุดถูกเรียกใช้งานเป็นจำนวนครั้งตามที่ระบุ ถ้าอักขระ `+` อยู่หน้าขีดจำกัด ขีดจำกัดของเหตุการณ์ นั้นถูกเปลี่ยนเป็นผลรวมของขีดจำกัดที่ระบุใน คำสั่งย่อย และค่านับของจำนวนครั้งที่เหตุการณ์ ถูกเรียกใช้งานไปแล้ว นั่นคือ คำสั่ง `dbx` หยุดรัน โปรแกรมดีบั๊กเมื่อจุดหยุดจะถูกเรียกใช้งานเป็นจำนวน `Limit` ที่ระบุคำสั่งย่อย `limitbp` รัน เรียบร้อย

## ตัวอย่าง

1. เมื่อต้องการสั่งให้คำสั่ง `dbx` หยุดการเรียกใช้งานโปรแกรม ดีบั๊กเมื่อจุดหยุด 1 ถูกเรียกใช้งาน เป็นครั้งที่ 10 ให้ป้อน:

```
limitbp (1, 10)
```

2. เมื่อต้องการสั่งให้คำสั่ง `dbx` หยุดการเรียกใช้งานโปรแกรม ดีบั๊กเมื่อจุดหยุด 1 จะถูกเรียกใช้งาน ครั้งที่ 15 หรือจุดหยุด 2 จะ ถูกเรียกใช้งานครั้งที่ 20 หรือทั้งสอง ให้ป้อน:

```
limitbp (1, 15) (2, 20)
```

3. เมื่อต้องการสั่งให้คำสั่ง `dbx` หยุดการเรียกใช้งานโปรแกรม ดีบั๊กเมื่อจุดหยุด 1 จะถูกเรียกใช้งาน ครั้งที่ 20 หลังจากคำสั่งย่อย `limitbp` รัน ให้ป้อน:

limitbp (1, +20)

## คำสั่งย่อย list

**list** [*Procedure* | *SourceLine-Expression* [, *SourceLine-Expression*] | **at** *Address* ]

คำสั่งย่อย **list** แสดงจำนวนบรรทัดที่ระบุของไฟล์ต้นฉบับ จำนวนบรรทัด ที่แสดงถูกระบุโดยใช้หนึ่งในวิธีต่อไปนี้:

- โดยการระบุโพรซีเจอร์โดยใช้พารามิเตอร์ *Procedure*  
ในกรณีนี้ คำสั่งย่อย **list** แสดงบรรทัดเริ่มต้น ไม่กี่บรรทัดก่อนจุดเริ่มต้นของโพรซีเจอร์ ที่ระบุจะกว่าหน้าต่างรายการจะเต็ม
- โดยการระบุหมายเลขบรรทัดซอร์สเริ่มต้นและสิ้นสุด โดยใช้ พารามิเตอร์ *SourceLine-Expression*  
พารามิเตอร์ *SourceLine-Expression* ต้องประกอบด้วยหมายเลขบรรทัดที่ถูกต้องอาจตามด้วย + (เครื่องหมายบวก) หรือ - (เครื่องหมายลบ) และตามด้วยเลขจำนวนเต็ม นอกจากนี้ *SourceLine* ของ \$ (เครื่องหมายดอลลาร์) ต้องใช้เพื่อแสดงหมายเลขบรรทัดปัจจุบัน *SourceLine* ของ @ (เครื่องหมาย at) อาจถูกใช้เพื่อแสดงหมายเลขบรรทัด ถัดไปที่จะแสดงรายการ
- โดยการระบุตัวแปร `$listwindow internal dbx`  
หากใช้ คำสั่งย่อย **list** โดยไม่มีพารามิเตอร์ จำนวนของบรรทัดที่ระบุโดยตัวแปร `$listwindow` จะถูกพิมพ์ เริ่มต้นจากบรรทัดต้นทางในปัจจุบัน เมื่อต้องการเปลี่ยน จำนวนบรรทัดดีฟอลต์ ให้ตั้งค่าตัวแปร `$listwindow` เป็นจำนวนบรรทัดที่ต้องการ ตัวแปร `$listwindow` เป็นตัวแปรโปรแกรมดีบั๊กพิเศษ เมื่อเริ่มต้น ตัวแปร `$listwindow` ถูกตั้งค่าเป็น 10

หากเว้นบรรทัดต้นทางที่สอง จะพิมพ์เฉพาะบรรทัดแรกเท่านั้น

บรรทัดทั้งหมดจากหมายเลขบรรทัดแรกที่ระบุ ถึงหมายเลขบรรทัดที่สองที่ระบุ จะถูกแสดงทั้งหมด

เมื่อคุณระบุ แอดเดรสหลังจากพารามิเตอร์ **at** ในคำสั่งย่อย **list** คำสั่งย่อย **list** จะแสดงบรรทัดต้นทางที่ สอดคล้องกับบรรทัดที่ระบุ แอดเดรสสามารถระบุเป็น เลขทศนิยมหรือเลขจำนวนเต็มฐานสิบหกที่ไม่มีเครื่องหมาย หรือเป็น mnemonic ที่สอดคล้อง รีจิสเตอร์ เช่น `$iar`, `$tfiar` และ `$tfhar` หรือตัวแปรดีบั๊ก

## ตัวอย่าง

1. เมื่อต้องการแสดงบรรทัด 1 ถึง 10 ในไฟล์ปัจจุบัน ให้ป้อน:  

```
list 1,10
```
2. เมื่อต้องการแสดง 10, หรือ `$listwindow`, บรรทัดรอบโพรซีเจอร์ `main` ให้ป้อน:  

```
list main
```
3. เมื่อต้องการแสดง 11 บรรทัดรอบบรรทัดปัจจุบัน ให้ป้อน:  

```
list $-5,$+5
```
4. คุณสามารถใช้นิพจน์จำนวนเต็ม ที่เกี่ยวกับการบวกและการลบในนิพจน์ *SourceLineExpression* ตัวอย่างเช่น:  

```
(dbx) list $
4 {

(dbx) list 5
5 char i = '4';

(dbx) list sub
23 char *sub(s,a,k)
```

```

24 int a;
25 enum status k; . . .

(dbx) move
25
(dbx) list @ -2
23 char *sub(s,a,k)

```

5. คุณสามารถแสดงบรรทัดต้นทางที่สอดคล้องกับ แอดเดรสเฉพาะตัวอย่างเช่น :

```

(dbx) r
[1] stopped in main at line 5
    5 int i, sum = 0;

(dbx) list at $iar
source file: "tt.c"
    5 int i, sum = 0;
    6 int last = 0;
    7
    8 scanf("%d", &last);
    9
   10 for ( i = 1; i <=last; i++ ) {
   11 sum += i;
   12 }
   13 printf("sum = %d\n", sum);
   14

(dbx) list at ($iar+16)
source file: "tt.c"
    8 scanf("%d", &last);
    9
   10 for ( i = 1; i <= last; i++ ) {
   11 sum += i;
   12 }
   13 printf("sum = %d\n", sum);
   14
   15 return 0;
   16 }

```

ดูที่คำสั่งย่อย **edit** คำสั่งย่อย **listi** และคำสั่งย่อย **move** นอกจากนี้ ดูที่ การแสดงไฟล์ปัจจุบัน in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย listi

**listi** [*Procedure* | **at** *SourceLine* | *Address* [, *Address* ]]

คำสั่งย่อย **listi** แสดงชุดของคำสั่งที่ระบุจากไฟล์ต้นฉบับ คำสั่ง ที่แสดงถูกระบุโดย:

- การระบุพารามิเตอร์ *Procedure* , โดยที่คำสั่งย่อย **listi** แสดงคำสั่ง จากจุดเริ่มต้นของโปรซีเจอร์ที่ระบุจนกว่าหน้าต่างรายการจะเต็ม
- การใช้แฟล็ก **at** *SourceLine* โดยที่คำสั่งย่อย **listi** แสดงคำสั่งเริ่มต้นที่บรรทัดซอร์สที่ระบุ และดำเนินต่อจนกว่าหน้าต่างที่แสดงจะเต็ม ตัวแปร *SourceLine* สามารถถูกระบุเป็นจำนวนเต็ม หรือสตริงชื่อไฟล์ที่ตามด้วย : (โคลอน) และจำนวนเต็ม
- การระบุแอดเดรสเริ่มต้นและสุดท้าย โดยใช้พารามิเตอร์ *Address* โดยที่คำสั่งทั้งหมด ระหว่างสองแอดเดรส ถูกแสดง

ถ้าคำสั่งย่อย `listi` ถูกใช้โดยไม่มีแฟล็กหรือพารามิเตอร์ คำสั่ง `$listwindow` ถัดไป จะถูกแสดง เมื่อต้องการเปลี่ยนขนาดปัจจุบันของหน้าต่าง `list` ให้ใช้ คำสั่งย่อย `set $listwindow=Value`

## โหมด Disassembly

โปรแกรม `dbx` สามารถแยกคำสั่งแอสเซมบลีสำหรับสถาปัตยกรรม POWER<sup>®</sup> family หรือ PowerPC<sup>®</sup> ในโหมดเริ่มต้น โปรแกรม `dbx` แสดง คำสั่งสำหรับสถาปัตยกรรมซึ่งโปรแกรมรันอยู่

ตัวแปร `$instructionset` และ `$mnemonics` ของคำสั่งย่อย `set` สำหรับคำสั่ง `dbx` อนุญาตให้คุณเขียนทับ โหมดการถอดแยกภาษาแอสเซมบลีที่พอลต์ สำหรับข้อมูลเพิ่มเติม ดูที่คำสั่งย่อย `set` สำหรับคำสั่ง `dbx`

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>at SourceLine</code>	ระบุบรรทัดซอร์สเริ่มต้นสำหรับการแสดงรายการ

## ตัวอย่าง

- เมื่อต้องการแสดงข้อมูล 10 รายการถัดไปหรือคำสั่ง `$listwindow`, ให้ป้อน:  
`listi`
- เมื่อต้องการแสดงคำสั่งเครื่องที่เริ่มต้น ที่บรรทัดซอร์ส 10 ให้ป้อน:  
`listi at 10`
- เมื่อต้องการแสดงคำสั่งเครื่อง เริ่มต้นที่บรรทัดซอร์ส 5 ในไฟล์ `sample.c` ให้ป้อน:  
`listi at "sample.c":5`
- เมื่อต้องการแสดงคำสั่งระหว่างแอดเดรส `0x10000400` และ `0x10000420`, ให้ป้อน:  
`listi 0x10000400, 0x10000420`

ดูที่คำสั่งย่อย `list` และคำสั่งย่อย `set` และดูที่ การดีบั๊กที่ระดับเครื่อง ด้วย `dbx` in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย malloc

`malloc` [`> File`]

คำสั่งย่อย `malloc` ที่ไม่มีอ็อปชันพิมพ์ รายการของอ็อปชันและนโยบายการจัดสรรที่เปิดใช้ และสรุปสถิติของการใช้ `malloc` ตั้งแต่กระบวนการเริ่มทำงาน

`malloc` [`allocation` [`{ address|size|heap|pid|tid|time }` `{ "<"|"=="|">"|!="|~=" }` `Value` ]]`> File`]

อ็อปชัน `allocation` ของคำสั่งย่อย `malloc` แสดงรายการที่เรียงลำดับของการจัดสรรที่กระบวนการ ครอบครองอยู่ขณะนี้ การใช้ อาร์กิวเมนต์ attribute `RELOP value` ทางเลือก อนุญาตให้ใช้การเลือกการจัดสรรที่แอดดที่พีที่แคบลง

`malloc` [`freespace` [`{ address|size|heap }` `{ "<"|"=="|">"|!="|~=" }` `Value` ]]`> File`]

อ็อพชัน `freespace` ของคำสั่งย่อย `malloc` แสดงรายการที่เรียงลำดับของพื้นที่ว่างที่มีทั้งหมดในฮีบของ กระบวนการ การใช้ อาร์กิวเมนต์ `attribute RELOP value` ทางเลือก อนุญาตให้ใช้การเลือกโหมดพื้นที่ว่างที่แคบลง

หมายเหตุ: โอเปอเรเตอร์ `~=` สามารถใช้ได้เฉพาะกับ อ็อพชันแอดเดรสเท่านั้น ตัวดำเนินการนี้ใช้เพื่อดึงข้อมูลพื้นที่ว่าง หรือ โหนด การจัดสรรซึ่งมีแอดเดรสที่ระบุอยู่

### `malloc address`

คำสั่งย่อย `malloc` ที่มีแอดเดรสแสดงรายละเอียดโหนดของแอดเดรส แอดเดรส ต้องไม่เป็นแอดเดรสเริ่มต้นของโหนดที่จัดสรร หรือโหนดว่าง

### แฟล็ก

ไอเท็ม	คำอธิบาย
> <code>File</code>	เปลี่ยนทิศทางเอาต์พุตไปที่ไฟล์ที่ระบุ

สำหรับข้อมูลเพิ่มเติม ดูที่ การจัดสรรหน่วยความจำ ระบบ การใช้ระบบย่อย `malloc` ใน *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย `map`

```
map { [Format] [entry ModuleNumber [, ModuleNumber] | Address | SymbolName ] [for $threadnumber] [ > File ] }
```

คำสั่งย่อย `map` แสดง คุณสมบัติสำหรับส่วนที่โหลดของแ็พพลิเคชัน ข้อมูลนี้ สามารถรวม ชื่อโมดูล ชื่อสมาชิก จุดเริ่มต้นข้อความ จุดสิ้นสุดข้อความ ความยาวข้อความ จุดเริ่มต้นข้อมูล จุดสิ้นสุดข้อมูล ความยาวข้อมูล จุดเริ่มต้นข้อมูล TLS , จุดสิ้นสุดข้อมูล TLS, ความยาวข้อมูล TLS และไฟล์ descriptor สำหรับแต่ละโมดูล ที่โหลด รายการที่จะถูกแสดงสามารถถูกระบุในวิธีต่อไปนี้:

- โดยการระบุรายการเดียวโดยใช้พารามิเตอร์ `ModuleNumber`
- โดยการระบุช่วงของรายการโดยใช้พารามิเตอร์ `ModuleNumber` ที่แยกด้วยสองคอมมา
- โดยการระบุแอดเดรสที่จะถูกกำหนดให้กับโมดูลที่โหลดโดยใช้ พารามิเตอร์ `Address`
- โดยการระบุชื่อสัญลักษณ์ที่จะถูกกำหนดให้กับโมดูลที่โหลด โดยใช้พารามิเตอร์ `SymbolName`

เมื่อมีการเรียกโดยไม่มีการระบุหนึ่งในข้อกำหนดด้านบน คำสั่งย่อย `map` จะแสดงข้อมูลสำหรับส่วนที่โหลดของแ็พพลิเคชันทั้งหมด

อาร์กิวเมนต์ `Format` ระบุโหมดเอาต์พุต สำหรับรายละเอียดโมดูลที่โหลด รายการต่อไปนี้ มีค่าที่เป็นไปได้สำหรับอาร์กิวเมนต์ `Format` :

ไอเท็ม abbr	คำอธิบาย ระบุโหนดเอาต์พุตแบบย่อ ซึ่งประกอบด้วยบรรทัดหนึ่งสำหรับแต่ละโหนดที่โหนดที่มีหมายเลขรายการ ชื่อโหนด และชื่อสมาชิก เป็นทางเลือกสำหรับโหนดนั้น
ปกติ	ระบุโหนดเอาต์พุตปกติ ซึ่งประกอบด้วย หมายเลขรายการ ชื่อโหนด ชื่อสมาชิก จุดเริ่มต้นข้อความ ความยาวข้อความ จุดเริ่มต้นข้อมูล ความยาวข้อมูล และไฟล์ descriptor สำหรับแต่ละโหนดที่โหนด ถ้าโหนดที่โหนดมีข้อมูล TLS จุดเริ่มต้นข้อมูล TLS และความยาวข้อมูล TLS ถูกแสดง
ดิบ	ระบุโหนดเอาต์พุต raw ซึ่งประกอบด้วย บรรทัดที่ไม่ได้จัดรูปแบบสำหรับแต่ละโหนดที่มีฟิลด์ที่แยกด้วยช่องว่าง: หมายเลขรายการ ชื่อโหนดที่มี ชื่อสมาชิกที่เป็นทางเลือก จุดเริ่มต้นข้อความ จุดสิ้นสุดข้อความ ความยาวข้อความ ความยาวข้อมูล จุดเริ่มต้นข้อมูล จุดสิ้นสุดข้อมูล ความยาวข้อมูล และไฟล์ descriptor ถ้าโหนดที่โหนดมีข้อมูล TLS จุดเริ่มต้นข้อมูล TLS จุดสิ้นสุดข้อมูล TLS และความยาวข้อมูล TLS จะถูกแสดง
verbose	ระบุโหนดเอาต์พุต verbose ซึ่งประกอบด้วย หมายเลขรายการ ชื่อโหนด ชื่อสมาชิก จุดเริ่มต้นข้อความ จุดสิ้นสุดข้อความ ความยาวข้อความ จุดเริ่มต้นข้อมูล จุดสิ้นสุดข้อมูล ความยาวข้อมูล และไฟล์ descriptor สำหรับแต่ละโหนดที่โหนด ถ้าโหนดที่โหนดมีข้อมูล TLS จุดเริ่มต้นข้อมูล TLS จุดสิ้นสุดข้อมูล TLS และความยาวข้อมูล TLS จะถูกแสดง

หากไม่ได้ระบุพารามิเตอร์ *Format* คำสั่ง `dbx` จะใช้ค่าของตัวแปรภายใน `$mapformat` หากไม่ได้ระบุพารามิเตอร์ *Format* และไม่ได้ตั้งค่า `$mapformat` คำสั่ง `dbx` จะแสดงข้อมูลโหนดที่ถูกโหนด ในโหมดปกติ

ข้อมูล TLS ของ thread ที่ระบุถูกแสดงถ้า โหนดที่โหนดมีข้อมูล TLS ถ้าไม่มีการระบุ thread จะใช้ thread ปัจจุบัน

## แฟล็ก

ไอเท็ม >File entry ModuleNumber [, ModuleNumber] for \$t threadnumber	คำอธิบาย เปลี่ยนทิศทางเอาต์พุตไปที่ไฟล์ที่ระบุ ระบุโหนดหรือช่วงของโหนดที่จะถูก แสดง ระบุหมายเลข thread
--	---

## ตัวอย่าง

- เมื่อต้องการแสดงรายการโหนดที่โหนดทั้งหมดในโหมดย่อ ให้พิมพ์:  
`map abbr`
- เมื่อต้องการแสดงรายการโหนดที่โหนด 3 ถึง 5 ในโหมดรายละเอียด ให้พิมพ์:  
`map verbose entry 3,5`
- เมื่อต้องการแสดงรายการโหนดที่โหนดที่มีแอดเดรส 0x20001000 ให้พิมพ์:  
`map 0x20001000`
- เมื่อต้องการแสดงรายการโหนดที่โหนดที่มีตัวแปร `example` ให้พิมพ์:  
`map example`
- เมื่อต้องการแสดงรายการโหนดที่โหนดในโหมดปกติที่มีข้อมูล TLS ของโหนดสำหรับเรด 2 ให้พิมพ์:  
`map normal for $t2`

สำหรับข้อมูลเพิ่มเติม ดูที่ตัวแปรภายใน `$mapformat` และดูที่ การดีบั๊กที่ระดับเครื่อง ด้วย `dbx` in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย move

`move SourceLine`

คำสั่งย่อย `move` เปลี่ยนบรรทัด ถัดไปที่จะถูกแสดงเป็นบรรทัดที่ระบุโดยพารามิเตอร์ `SourceLine` คำสั่งย่อยนี้เปลี่ยน ค่าของตัวแปร `@` (เครื่องหมาย at)

ตัวแปร `SourceLine` สามารถถูกระบุ เป็นจำนวนเต็มหรือสตริงชื่อไฟล์ที่ตามด้วย : (โคลอน) และจำนวนเต็ม

### ตัวอย่าง

1. เมื่อต้องการเปลี่ยนบรรทัดถัดไปที่จะถูกแสดง เป็นบรรทัด 12 ให้อ่าน:

```
move 12
```

2. เมื่อต้องการเปลี่ยนบรรทัดถัดไปที่จะถูกแสดง เป็นบรรทัด 5 ในไฟล์ `sample.c`, ให้อ่าน:

```
move "sample.c":5
```

ดูที่คำสั่งย่อย `list` รวมทั้ง ดูที่ การแสดงไฟล์ ปัจจุบัน in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย `multiproc`

`multiproc [ on | parent | child | off ]`

คำสั่งย่อย `multiproc` ระบุการทำงานของดีบั๊กโปรแกรม `dbx` เมื่อกระบวนการที่แยกและมากเกินไปถูกสร้าง แฟล็ก `on` ถูกใช้เพื่อระบุว่าเซสชัน `dbx` ใหม่ถูกสร้างขึ้นเพื่อดีบั๊กพาทชาสต์ของ `fork dbx` ต้นฉบับยังคงดีบั๊กพาทพาเรนต์ แฟล็ก `parent` และ `child` ถูกใช้เพื่อระบุพาทเดี่ยวของฟอร์กเพื่อติดตาม แฟล็กทั้งหมดยกเว้น `off` เปิดให้ `dbx` ติดตาม กระบวนการ `exceed` แฟล็ก `off` ปิดใช้งานการดีบั๊ก หลายกระบวนการ ถ้าไม่แฟล็กถูกระบุ คำสั่งย่อย `multiproc` ส่งคืนค่าสถานะของการดีบั๊กหลายกระบวนการ

โปรแกรม `dbx` ใช้ X Window System สำหรับการดีบั๊กแบบมีหลายกระบวนการ โปรแกรม `dbx` เปิดหน้าต่างหลายหน้าต่างตามที่เป็นสำหรับหลายการประมวลผล หัวเรื่องสำหรับ แต่ละหน้าต่างไชลด์คือ process ID (pid) ของกระบวนการไชลด์ เมื่อต้องการ สนับสนุนระหว่างกระบวนการให้ใช้เทคนิคการจัดการกับ X Window System เพื่อเปิดใช้งานหน้าต่างที่จะแสดงเซสชันคำสั่ง `dbx` หากระบบไม่มีการสนับสนุน X Window System ข้อความเตือนจะถูกแสดงเมื่อดีบั๊กเกอร์ forks และโปรแกรม `dbx` จะทำการดีบั๊กต่อเฉพาะกระบวนการพาเรนต์ การดีบั๊กหลายกระบวนการไม่สำเร็จได้ด้วยเหตุผล ต่อไปนี้:

- โปรแกรม `dbx` จะไม่รันในสถานะแวดล้อม X Window System
- X Window System กำลังทำงานแต่ตัวแปร `dbx global $xdisplay` ไม่ถูกตั้งค่า เพื่อแสดงชื่อที่ถูกต้อง ตัวแปร `$xdisplay` ถูกกำหนดค่าเริ่มต้นเป็นตัวแปรสถานะแวดล้อม เซลล์ `DISPLAY` คำสั่งย่อย `set Name=Expression dbx` สามารถถูกใช้เพื่อเปลี่ยนค่าของชื่อจอแสดงผล
- ไดรฟ์ทอรัส /tmp ไม่อนุญาตการเข้าถึงเพื่ออ่านหรือเขียนกับโปรแกรมที่ทำการดีบั๊ก โปรแกรม `dbx` ต้องการพื้นที่จำนวนมากใน ไดรฟ์ทอรัสนี้ เมื่อทำการควบคุมสถานะแวดล้อม Xwindow
- ระบบไม่มีรีซอร์ส เพื่อรองรับ Xwindow ใหม่

ถ้า `$xdisplay` ถูกตั้งค่าเป็นการแสดง รีโมต ผู้ใช้อาจไม่สามารถเห็น Xwindow ที่สร้างขึ้นใหม่ หากค่าติดตั้ง `$xdisplay` ไม่ถูกต้อง X Window System หรือรีซอร์สของระบบรายการสาเหตุของความล้มเหลว

โปรแกรม `dbx` ไม่ แยกระหว่างชนิดความล้มเหลวที่ต่างกัน แต่ข้อความต่อไปนี้ ถูกส่งเมื่อคำสั่งย่อยไม่สำเร็จ:

```
คำเตือน: multiproc คำสั่งย่อย dbx ล้มเหลว dbx
continued with multiproc disabled.
```

คอนฟิกูเรชันที่ผู้ใช้กำหนดเองของหน้าต่างที่สร้างใหม่สามารถถูกกำหนดภายใต้ชื่อแอ็พพลิเคชัน `dbx_term` ในไฟล์ `.Xdefaults`

## แฟล็ก

ไอเท็ม คำอธิบาย  
on เปิดใช้การดีบั๊กหลายกระบวนการ  
off ปิดใช้การดีบั๊กหลายกระบวนการ

## ตัวอย่าง

- เมื่อต้องการตรวจสอบสถานะของการดีบั๊กแบบหลายกระบวนการให้ป้อน:  
`multproc`
- เมื่อต้องการเปิดการดีบั๊กหลายกระบวนการให้ป้อน:  
`multproc on`
- เมื่อต้องการปิดการดีบั๊กหลายกระบวนการให้ป้อน:  
`multproc off`

ดูที่คำสั่งย่อย `screen` และรูทีนย่อย `fork` รวมทั้งดูที่ การดีบั๊กโปรแกรม ที่เกี่ยวกับหลายกระบวนการ in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย mutex

`mutex [ lock | unlock | thnum | utid | MutexNumber ... ]`

คำสั่งย่อย `mutex` แสดงข้อมูลเกี่ยวกับ mutexes ถ้ามีการกำหนดพารามิเตอร์ `MutexNumber` คำสั่งย่อย `mutex` แสดงผลข้อมูลเกี่ยวกับ mutexes ที่ระบุ ถ้าไม่มีการระบุแฟล็กหรือพารามิเตอร์ คำสั่งย่อย `mutex` แสดงผลข้อมูลเกี่ยวกับ mutexes ทั้งหมด

ข้อมูลที่แสดงสำหรับแต่ละ mutex เป็นดังนี้:

ไอเท็ม	คำอธิบาย
<code>mutex</code>	ระบุชื่อเชิงสัญลักษณ์ของ mutex ในฟอร์ม <code>\$mMutexNumber</code>
<code>type</code>	ระบุชนิดของ mutex: <code>non-rec</code> (ไม่มีการเรียกซ้ำ), <code>recursi</code> (เรียกซ้ำ) หรือ <code>fast</code>
<code>obj_addr</code>	ระบุแอดเดรสหน่วยความจำของ mutex
<code>lock</code>	ระบุสถานะล็อกของ mutex: <code>yes</code> ถ้า mutex ถูกล็อก, <code>no</code> ถ้าไม่ล็อก
<code>เจ้าของ</code>	ถ้า mutex ถูกล็อก หมายถึงชื่อเชิงสัญลักษณ์ของ thread ผู้ใช้เก็บ mutex ไว้
<code>blockers</code>	แสดง threads ผู้ใช้ซึ่งถูกล็อกบนตัวแปร mutex นี้

หมายเหตุ: คำสั่งย่อย `print` ของโปรแกรมดีบั๊ก `dbx` รู้จักชื่อ mutex เชิงสัญลักษณ์ และสามารถใช้เพื่อแสดง สถานะของอ็อบเจ็กต์ที่สอดคล้อง

## แฟล็ก

ไอเท็ม	คำอธิบาย
lock	แสดงผลข้อมูลเกี่ยวกับ mutexes ที่ล็อก
unlock	แสดงผลข้อมูลเกี่ยวกับ mutexes ที่ไม่ถูกล็อก
thnum	แสดงผลข้อมูลเกี่ยวกับ mutexes ทั้งหมดที่ thread มีอยู่
utid	แสดงผลข้อมูลเกี่ยวกับ mutexes ทั้งหมดที่ถือครองโดยเธรดผู้ใช้ ที่มี id เธรดผู้ใช้ตรงกับ id เธรดผู้ใช้

## ตัวอย่าง

1. เมื่อต้องการแสดงข้อมูลเกี่ยวกับ mutexes ทั้งหมดให้ป้อน:

```
mutex
```

2. เมื่อต้องการแสดงข้อมูลเกี่ยวกับ mutexes ที่ล็อกทั้งหมดให้ป้อน:

```
mutex lock
```

3. เมื่อต้องการแสดงข้อมูลเกี่ยวกับ mutexes หมายเลขสี่ห้าและหกให้ป้อน:

```
mutex 4 5 6
```

เอาต์พุตมีข้อมูลเหมือนดังนี้:

```
mutex  obj_addr      type    lock owner  blockers
$m4    0x20003274        non-rec no
$m5    0x20003280        recursi no
$m6    0x2000328a        fast   no
```

4. เมื่อต้องการแสดงข้อมูลเกี่ยวกับ mutexes ทั้งหมดที่ thread 1, มีอยู่ให้ป้อน:

```
mutex thnum 1
```

5. เมื่อต้องการแสดงข้อมูลเกี่ยวกับ mutexes ทั้งหมดที่ thread มีอยู่ซึ่ง thread id ผู้ใช้คือ 0x0001 ให้ป้อน:

```
mutex utid 0x0001
```

ดูที่คำสั่งย่อย **attribute**, คำสั่งย่อย **condition**, คำสั่งย่อย **print** และคำสั่งย่อย **thread**

และดูที่ การใช้ Mutexes *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย next

**next** [*Number*]

คำสั่งย่อย **next** รัน แอ็พพลิเคชันโปรแกรมไปที่บรรทัดซอร์สถัดไป พารามิเตอร์ *Number* ระบุจำนวนครั้งที่คำสั่งย่อย **next** รัน ถ้าไม่ระบุพารามิเตอร์ *Number*, **next** รัน เพียงครั้งเดียว

ถ้าคุณใช้คำสั่งย่อย **next** ในแอ็พพลิเคชันโปรแกรม multithreaded, threads ผู้ใช้ทั้งหมด ที่รันระหว่างการดำเนินการ แต่โปรแกรมดำเนินการทำงานต่อจนกว่า thread ที่รันอยู่ทำงานถึงบรรทัดซอร์สที่ระบุ ถ้าคุณต้องการทำขั้นตอนการรันเธรดเท่านั้น ให้ใช้คำสั่งย่อย **set** เพื่อตั้งค่าตัวแปร **\$hold\_next** การตั้งค่าตัวแปรนี้อาจทำให้เกิด deadlock เนื่องจากเธรดที่กำลังรันอยู่นั้นรอการล็อก ที่เกิดขึ้นโดยหนึ่งในเธรดที่ถูกล็อก

## ตัวอย่าง

1. เมื่อต้องการดำเนินการทำงานต่อไปที่บรรทัดซอร์สถัดไปให้ป้อน:

```
next
```

2. เมื่อต้องการดำเนินงานต่อไปที่บรรทัดซอร์สที่สามต่อจากบรรทัดซอร์สปัจจุบัน ให้ป้อน:

```
next 3
```

ดูที่คำสั่งย่อย `cont`, คำสั่งย่อย `goto`, คำสั่งย่อย `nexti`, คำสั่งย่อย `set`, และ คำสั่งย่อย `step`

### คำสั่งย่อย `nexti`

`nexti` [ *Number* ]

คำสั่งย่อย `nexti` รัน แอปพลิเคชันโปรแกรมไปที่คำสั่งถัดไป พารามิเตอร์ *Number* ระบุ จำนวนครั้งที่คำสั่งย่อย `nexti` รัน ถ้าไม่ระบุพารามิเตอร์ *Number*, `nexti` รัน เพียงครั้งเดียว

ถ้าคุณใช้คำสั่งย่อย `nexti` ในแอปพลิเคชันโปรแกรม `multithreaded`, `threads` ผู้ใช้ทั้งหมด ที่รันระหว่างการทำงาน แต่โปรแกรมดำเนินการทำงานต่อจนกว่า `thread` ที่รันอยู่ทำงานถึงคำสั่งเครื่องที่ระบุ ถ้าคุณต้องการทำขั้นตอนการรันเธรดเท่านั้น ให้ใช้คำสั่งย่อย `set` เพื่อตั้งค่าตัวแปร `$hold_next` การตั้งค่าตัวแปรนี้อาจทำให้เกิด `deadlock` เนื่องจากเธรดที่กำลังรันอยู่นั้นรอการล็อก ที่เกิดขึ้นโดยหนึ่งในเธรดที่ถูกบล็อก

### ตัวอย่าง

1. เมื่อต้องการดำเนินการทำงานไปที่คำสั่งเครื่อง ถัดไปให้ป้อน:

```
nexti
```

2. เมื่อต้องการดำเนินงานต่อไปที่คำสั่งเครื่อง ที่สามต่อจากคำสั่งเครื่องปัจจุบัน ให้ป้อน:

```
nexti 3
```

ดูที่คำสั่งย่อย `gotoi`, คำสั่งย่อย `next`, คำสั่งย่อย `set` และ คำสั่งย่อย `stepi` และ ดูที่ การรันโปรแกรมที่ระดับเครื่อง in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย `onceblock`

`onceblock` [ `uninit` | `done` ]

คำสั่งย่อย `onceblock` แสดงข้อมูลเกี่ยวกับ บล็อกของโค้ดที่กำหนดค่าเริ่มต้นที่รีจิสเตอร์โดยใช้รูทีน `pthread_once` เมื่อไม่มีอาร์กิวเมนต์ ข้อมูล เกี่ยวกับค่าที่รีจิสเตอร์ทั้งหมดเมื่อถูกบล็อกจะแสดง แฟล็กทางเลือก `uninit` และ `done` แสดงเฉพาะบล็อก `once` ที่ไม่ได้ดำเนินการ หรือได้ดำเนินการแล้ว ตามลำดับ ขณะที่การระบุหมายเลข `once ID` แสดง ข้อมูลสำหรับบล็อก `once` เดียว

หมายเหตุ: สำหรับคำสั่งย่อย `onceblock` ที่จะทำงานขณะดีบั๊กกระบวนการที่ดำเนินอยู่ ตัวแปรสภาวะแวดล้อม `AIXTHREAD_ONCE_DEBUG` ต้องถูกตั้งค่าเท่ากับ `ON` ในทำนองเดียวกัน ถ้าคำสั่งดีบั๊ก ไฟล์หลัก ถ้าตัวแปรไม่ `on` อยู่เมื่อกระบวนการรัน คำสั่งย่อย `onceblock` จะไม่สามารถจัดหาข้อมูลได้

### ตัวอย่าง

1. เมื่อต้องการค้นหาว่ามีบล็อกครั้งเดียวใดที่ยังไม่ถูกเรียกใช้งาน ให้พิมพ์:

```
onceblock uninit
```

### คำสั่งย่อย `plugin`

**plugin** [ *Name* [ *Command* ] ]

คำสั่งย่อย **plugin** ส่งคำสั่งที่ระบุโดยพารามิเตอร์ *Command* ไปยังปลั๊กอินที่ระบุโดยพารามิเตอร์ *Name* ถ้าไม่มีการระบุพารามิเตอร์ชื่อของปลั๊กอินที่มีทั้งหมดถูกแสดง

### ตัวอย่าง

1. เมื่อต้องการแสดงปลั๊กอินที่มีอยู่ทั้งหมด ให้พิมพ์:  
`plugin`
2. เมื่อต้องการเริ่มทำงานคำสั่งย่อย "help" ของปลั๊กอินที่ชื่อ "sample" ให้พิมพ์:  
`plugin sample help`
3. เมื่อต้องการเริ่มทำงานคำสั่งย่อย "interpret 0x20000688" ของปลั๊กอินที่ชื่อ "xyz" ให้พิมพ์:  
`plugin xyz interpret 0x20000688`

ดูที่คำสั่งย่อย **pluginload** และคำสั่งย่อย **pluginunload** และดูที่ การพัฒนาสำหรับ dbx Plug-in Framework ใน *General Programming Concepts*

### คำสั่งย่อย **pluginload**

#### **pluginload** *File*

คำสั่งย่อย **pluginload** โหลดปลั๊กอิน ที่ระบุโดยพารามิเตอร์ *File* พารามิเตอร์ *File* ต้องระบุพาธไปยังปลั๊กอิน

หมายเหตุ: เนื่องจากคำสั่ง **dbx** ดีฟอลต์ เป็นกระบวนการ 64 บิต คุณต้องใช้คำสั่ง **dbx** ในเวอร์ชัน 32 บิต ที่ชื่อ **dbx32** เพื่อโหลดปลั๊กอิน 32 บิต

### ตัวอย่าง

เมื่อต้องการโหลดปลั๊กอินชื่อ "sample" ที่ "/home/user/dbx\_plugins/libdbx\_sample.so" ให้พิมพ์:  
`pluginload /home/user/dbx_plugins/libdbx_sample.so`

ดูที่คำสั่งย่อย **plugin** และคำสั่งย่อย **pluginunload** และดูที่ การพัฒนาสำหรับ dbx Plug-in Framework ใน *General Programming Concepts*

### คำสั่งย่อย **pluginunload**

#### **pluginunload** *Name*

คำสั่งย่อย **pluginunload** ยกเลิกการโหลดปลั๊กอิน ที่ระบุโดยพารามิเตอร์ *Name*

### ตัวอย่าง

เมื่อต้องการยกเลิกการโหลดปลั๊กอินชื่อ "sample" ให้พิมพ์:  
`pluginunload sample`

ดูที่คำสั่งย่อย **plugin** และคำสั่งย่อย **pluginload** และดูที่ การพัฒนาสำหรับ dbx Plug-in Framework ใน *General Programming Concepts*

### คำสั่งย่อย **print**

**print** *Expression* ...

**print** *Procedure* ( [*Parameters*] )

คำสั่งย่อย **print** มีการดำเนินการอย่างใดอย่างหนึ่งต่อไปนี้:

- พิมพ์ค่าของรายการนิพจน์ ที่ระบุโดยพารามิเตอร์ *Expression*
- เรียกใช้งานโปรซีเจอร์ที่ระบุโดยพารามิเตอร์ *Procedure* และพิมพ์ค่าส่งคืนของโปรซีเจอร์ตานั้น พารามิเตอร์ที่ถูกรวมไว้ถูกส่งไปที่โปรซีเจอร์

### ตัวอย่าง

1. เมื่อต้องการแสดงค่าของ *x* และค่าของ *y* ที่เลื่อน ไปทางซ้ายสองบิต ให้ป้อน:  

```
print x, y << 2
```
2. เมื่อต้องการแสดงค่าที่ส่งคืนโดยการเรียกใช้รoutines *sbrk* ที่มีอาร์กิวเมนต์ 0 ให้ป้อน:  

```
print sbrk(0)
```

ดูที่คำสั่งย่อย **assign** , คำสั่งย่อย **call** และคำสั่งย่อย **set**

### **printbp** Subcommand

**printbp** [*bp1*] [*bp2*] ... | **all**

คำสั่งย่อย **printbp** สั่งให้คำสั่ง **dbx** พิมพ์จำนวนครั้งที่แต่ละจุดหยุดหรือคำสั่งย่อย ทั้งหมดถูกรัน และรายละเอียดของขีดจำกัด บนจุดหยุด ถ้าขีดจำกัดถูกตั้งไว้

### ตัวอย่าง

1. เมื่อต้องการสั่งให้คำสั่ง **dbx** พิมพ์จำนวนครั้งที่จุดหยุด 1 ถูกรัน และรายละเอียดของขีดจำกัด ที่ตั้งค่า ให้ป้อน:  

```
printbp 1
```
2. เมื่อต้องการสั่งให้คำสั่ง **dbx** พิมพ์จำนวนครั้งที่จุดหยุด 1 และ 2 ถูกรัน และเพื่อจำกัดจำนวนครั้งที่จุดหยุด 1 และ 2 สามารถได้รับอนุญาตให้รับถ้ามีขีดจำกัดถูกตั้งไว้ ให้ป้อน:  

```
printbp 1, 2
```
3. เมื่อต้องการสั่งให้คำสั่ง **dbx** พิมพ์จำนวนครั้งที่จุดหยุดทั้งหมดถูกรัน และรายละเอียดของขีดจำกัดของจุดหยุดใดๆ ถ้ามี ให้ป้อน:  

```
printbp all
```

### คำสั่งย่อย **proc**

**proc** [*raw*] [*cred* | *cru* | *ru* | *sigflags* | *signal* ]

คำสั่งย่อย `proc` แสดงข้อมูล เกี่ยวกับกระบวนการ การใช้อ็อปชัน `raw` ทำให้เอาต์พุตถูกแสดงใน raw hex, แทนการถูกแปลเป็นค่า ในรูปแบบที่มนุษย์เข้าใจได้มากกว่า การใช้คำสั่งย่อย `proc subcommand` โดยไม่มีอาร์กิวเมนต์เพิ่มเติมจะให้ข้อมูลทั่วไปเกี่ยวกับ กระบวนการ ตามที่ถูกเก็บไว้ในโครงสร้างข้อมูลของกระบวนการผู้ใช้ อ็อปชัน `cred` แสดงเนื้อหาของสมาชิกข้อมูล `pi_cred` ซึ่งอธิบาย credentials ของกระบวนการ อ็อปชัน `cru` และ `ru` แสดง สมาชิกข้อมูล `pi_cru` and `pi_ru` ตามลำดับ ซึ่งมีข้อมูลการใช้รีซอร์ส อ็อปชัน `sigflags` และ `signal` แสดงข้อมูลที่สัมพันธ์กับ สถานะสัญญาณปัจจุบัน และ handlers สัญญาณที่รีจิสเตอร์ ตามที่อยู่ใน สมาชิกข้อมูล `pi_sigflags` และ `pi_signal`

### ตัวอย่าง

1. เมื่อต้องการดูข้อมูลการใช้รีซอร์สสำหรับกระบวนการปัจจุบัน (หรือ คอร์ไฟล์) ใน raw hex ให้พิมพ์:

```
proc raw ru
```

2. เมื่อต้องการดูข้อมูล handler สัญญาณให้พิมพ์:

```
proc signal
```

### คำสั่งย่อย prompt

```
prompt [ "String" ]
```

คำสั่งย่อย `prompt` เปลี่ยนพร้อมต์คำสั่ง `dbx` เป็นสตริงที่ ระบุโดยพารามิเตอร์ `String`

### ตัวอย่าง

เมื่อต้องการเปลี่ยนแปลง พร้อมต์เป็น `dbx>` ให้ป้อน:

```
prompt "dbx>"
```

ดูที่ การกำหนดพร้อมต์ `dbx` ใหม่ in *General Programming Concepts: Writing and Debugging Programs*.

### คำสั่งย่อย quit

#### quit

คำสั่งย่อย `quit` สิ้นสุด กระบวนการทั้งหมดที่รันอยู่ในเซสชันการดีบั๊ก `dbx`

### ดูที่คำสั่งย่อย detach

### คำสั่งย่อย registers

```
registers [ ALL | $tthreadnumber ... ] [ >File ]
```

คำสั่งย่อย `registers` แสดงค่าของรีจิสเตอร์อเนกประสงค์ รีจิสเตอร์ควบคุม ระบบ รีจิสเตอร์เทคนิคม เวกเตอร์ และรีจิสเตอร์คำสั่งปัจจุบัน

- รีจิสเตอร์อเนกประสงค์ถูกแสดงโดยตัวแปร `$rNumber` โดยที่พารามิเตอร์ `Number` ระบุจำนวนของ รีจิสเตอร์

หมายเหตุ: ค่า รีจิสเตอร์อาจถูกตั้งค่าเป็นค่าฐานสิบหก `Oxdeadbeef` ค่าเลขฐานสิบหก `Oxdeadbeef` เป็น ค่าการกำหนดค่าเริ่มต้นที่กำหนดให้กับรีจิสเตอร์อเนกประสงค์เมื่อทำการกำหนดค่าเริ่มต้น กระบวนการ

- การลงทะเบียน floating point แสดงโดยตัวแปร `$frNumber` โดยดีฟอลต์รีจิสเตอร์อิงดัชนีไม่ถูกแสดง เมื่อต้องการแสดงรีจิสเตอร์อิงดัชนีให้ใช้คำสั่งย่อย `unset $noflregs dbx`
- เวกเตอร์รีจิสเตอร์ถูกแสดงโดยตัวแปร `$vrNumber` ตัวแปรภายใน `$novregs` ควบคุมว่าเวกเตอร์รีจิสเตอร์จะถูกแสดงหรือไม่ ตัวแปร `$novregs` variable ถูกเซตโดยดีฟอลต์ และเวกเตอร์รีจิสเตอร์ไม่ถูกแสดง เมื่อ `$novregs` ไม่ถูกเซต เวกเตอร์รีจิสเตอร์ใช้ได้ (โดยการดีบั๊กโปรแกรม บนโพรเซสเซอร์ที่ทำงานได้กับเวกเตอร์ หรือการวิเคราะห์คอร์ไฟล์ที่มีสถานะของเวกเตอร์รีจิสเตอร์) จากนั้นเวกเตอร์รีจิสเตอร์ทั้งหมดถูกแสดง (vr0-vr31, vrsave, vscr) เวกเตอร์รีจิสเตอร์ยังสามารถถูกอ้างอิงได้โดยพิมพ์ตัวอย่างตัวแปรเวกเตอร์รีจิสเตอร์ `$vrNf` (float), `$vrNs` (short), และ `$vrNc` (char) สามารถถูกใช้กับ คำสั่งย่อย `print` และ `assign` เพื่อแสดงและ เซตเวกเตอร์รีจิสเตอร์ตามชนิด
- เวกเตอร์สเกลาร์รีจิสเตอร์ถูกแสดงโดยตัวแปร `$vsrNumber` โดยดีฟอลต์เวกเตอร์สเกลาร์รีจิสเตอร์ไม่ถูกแสดง ยกเลิกการเซตตัวแปร `$novsregs` เพื่อแสดงเวกเตอร์สเกลาร์รีจิสเตอร์เมื่อเวกเตอร์สเกลาร์รีจิสเตอร์ ใช้ได้ (โดยการดีบั๊กโปรแกรมบนโพรเซสเซอร์ที่ทำงานได้กับเวกเตอร์สเกลาร์ หรือการวิเคราะห์คอร์ไฟล์ที่มีสถานะเวกเตอร์สเกลาร์รีจิสเตอร์) เนื่องจากเวกเตอร์สเกลาร์รีจิสเตอร์เป็นซูเปอร์เซตของรีจิสเตอร์ทศนิยม เดิม และรีจิสเตอร์เวกเตอร์ ตัวแปรดีบั๊ก `$novsregs` เมื่อไม่ตั้งค่า จะมาก่อน `$noflregs` และ `$novsregs` เมื่อใดที่สถานะเวกเตอร์สเกลาร์รีจิสเตอร์ถูกต้อง จากนั้นคำสั่งย่อย `registers` จะ แสดงเวกเตอร์สเกลาร์รีจิสเตอร์พร้อมกับ aliases รีจิสเตอร์ที่สืบทอด ตามมาในเครื่องหมายปีกกานามแฝงรีจิสเตอร์ทศนิยม สอดคล้องกับ 64 บิตตำแหน่ง เวกเตอร์สเกลาร์รีจิสเตอร์ยังสามารถถูก อ้างอิงตามชนิดเหมือนกับเวกเตอร์รีจิสเตอร์ ตัวอย่าง ตัวแปรเวกเตอร์สเกลาร์รีจิสเตอร์ `$vsrNf` (float), `$vsrNs` (short), `$vsrNc` (char), `$vsrNg` (double) และ `$vsrNll` (long long) สามารถถูกใช้กับคำสั่งย่อย `print` และ `assign` เพื่อแสดง และเซตเวกเตอร์สเกลาร์รีจิสเตอร์ตามชนิด
- ในสภาวะแวดล้อม multithreaded อีอ็อปชัน ALL แสดงรายละเอียดรีจิสเตอร์ สำหรับ threads ที่มีทั้งหมด รายละเอียดรีจิสเตอร์ของแต่ละ threads ถูกแสดงโดยการระบุหมายเลข thread ตามด้วยคำสั่งย่อย `registers` การใช้คำสั่งย่อย `registers` โดยไม่มีอีอ็อปชันแสดง รีจิสเตอร์สำหรับเรดปัจจุบัน

หมายเหตุ: คำสั่งย่อย `registers` ไม่สามารถแสดงรีจิสเตอร์ ถ้า thread ปัจจุบันอยู่ในคอร์เนลโหมด

## แฟล็ก

ไอเท็ม คำอธิบาย  
>File เปลี่ยนทิศทางเอาต์พุตไปที่ไฟล์ที่ระบุ

ดูที่คำสั่งย่อย `set` และคำสั่งย่อย `unset` และดูที่ การใช้รีจิสเตอร์เครื่อง in *General Programming Concepts: Writing and Debugging Programs*

## ตัวอย่าง

เมื่อต้องการ แสดงรายละเอียดรีจิสเตอร์ของเรด \$t1, \$t2 และ \$t3 ให้ป้อน:

```
registers $t1 $t2 $t3
```

ดูที่คำสั่งย่อย `set` และคำสั่งย่อย `unset` และดูที่ การใช้รีจิสเตอร์เครื่อง in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย `rerun`

```
rerun [ Arguments ] [ < File ] [ > File ] [ > > File ] [ 2> File ] [ 2>> File ] [ >& File ] [ >>& File ]
```

คำสั่งย่อย `rerun` เริ่มการทำงานของอ็อบเจ็กต์ไฟล์ โดย `Arguments` ถูกส่ง เป็นอาร์กิวเมนต์บรรทัดรับคำสั่ง ถ้าพารามิเตอร์ `Arguments` ไม่ระบุ อาร์กิวเมนต์จากคำสั่งย่อย `run` หรือ `rerun` จะถูกใช้ซ้ำ

## แฟล็ก

ไอเท็ม	คำอธิบาย
<File	เปลี่ยนทิศทางอินพุตเพื่อที่จะรับอินพุตจาก File
>File	เปลี่ยนทิศทางเอาต์พุตไปที่ File
>>File	ผนวกเอาต์พุตที่เปลี่ยนทิศทางไปที่ File
2>File	เปลี่ยนทิศทางข้อผิดพลาดมาตรฐานไปที่ File
2>>File	ผนวกข้อผิดพลาดมาตรฐานที่เปลี่ยนทิศทางไปที่ File
>&File	เปลี่ยนทิศทางเอาต์พุตและข้อผิดพลาดมาตรฐานไปที่ File
>>&File	ผนวกเอาต์พุตและข้อผิดพลาดมาตรฐานไปที่ File

## ดูที่คำสั่งย่อย `run`

## คำสั่งย่อย `resource`

```
resource { owner | waiter } [ all | pthread id ]
```

คำสั่งย่อย `resource` แสดงข้อมูลเกี่ยวกับ รีซอร์สที่ pthreads ถือครองอยู่หรือรอคอยอยู่ อาร์กิวเมนต์แรกจำเป็นต้องมี ระบุว่า คุณสนใจที่จะ ดู pthreads ที่เป็นเจ้าของรีซอร์ส หรือกำลังรอรีซอร์ส อาร์กิวเมนต์ที่สองสามารถถูกใช้เพื่อระบุ pthreads ทั้งหมด หรือเฉพาะเจาะจง ถ้าไม่มีการระบุข้อมูลใด เฉพาะข้อมูลที่สัมพันธ์กับ pthread ปัจจุบันจะถูกแสดง ถ้ามี

**หมายเหตุ:** คำสั่งย่อย `resource` มีประโยชน์ เฉพาะสำหรับการดีบักกระบวนการที่รัน โดยมีตัวแปรสถานะแวดล้อมการ ดีบัก หลายตัวแปรถูกเซตเป็น ON ซึ่งรวมถึง AIXTHREAD\_MUTEX\_DEBUG, AIXTHREAD\_COND\_DEBUG, AIXTHREAD\_RWLOCK\_DEBUG, AIXTHREAD\_READ\_OWNER และ AIXTHREAD\_WAITLIST\_DEBUG ถ้า ตัวแปรเหล่านี้ไม่ถูกเปิด ขณะทำการดีบักกระบวนการที่ทำงานอยู่ หรือไม่ได้เปิดอยู่เมื่อดีบักเกอร์คอร์ ไฟล์ถูกสร้าง คำสั่งย่อย `resource` จะสามารถเรียกข้อมูลได้น้อยหรือไม่ได้เลย เนื่องจาก การใช้คุณลักษณะเหล่านี้จะลดประสิทธิภาพการทำงานลง ขอแนะนำให้ เรียกทำงานคุณลักษณะเหล่านี้เพื่อการดีบักเท่านั้น

## ตัวอย่าง

1. เมื่อต้องการตรวจสอบว่า pthread ปัจจุบันถือครองรีซอร์สอยู่หรือไม่ ให้พิมพ์:  
resource owner
2. เมื่อต้องการดูรีซอร์สที่มี pthreads รออยู่ ให้พิมพ์:  
resource waiter all

## คำสั่งย่อย `return`

```
return [ Procedure ]
```

คำสั่งย่อย `return` ทำให้แฉีพพลิเคชันโปรแกรมทำงานจนถึงมีการส่งคืนไปที่ไพร์ซีเตอร์ ที่ระบุโดยพารามิเตอร์ `Procedure` ถ้า พารามิเตอร์ `Procedure` ไม่ถูกระบุ การทำงานจะหยุดเมื่อไพร์ซีเตอร์ปัจจุบันส่งคืนค่า

## ตัวอย่าง

1. เมื่อต้องการดำเนินการทำงานต่อไปที่รูทีนที่เรียก ให้ป้อน:

สังคิน

2. เมื่อต้องการดำเนินการโพรซีเดอร์ main ต่อให้ป้อน:

```
return main
```

### คำสั่งย่อย rwlock

**rwlock** [read | write | *RwlockNumber*....]

คำสั่งย่อย **rwlock** แสดงข้อมูลเกี่ยวกับ rwlocks ถ้ามีการกำหนดพารามิเตอร์ *RwlockNumber* คำสั่งย่อย **rwlock** แสดงผลข้อมูลเกี่ยวกับ rwlocks ที่ระบุ ถ้าไม่มีการระบุแฟล็กหรือพารามิเตอร์ คำสั่งย่อย **rwlock** แสดงผลข้อมูลเกี่ยวกับ rwlocks ทั้งหมด

ข้อมูลสำหรับแต่ละ **rwlock** เป็น ดังนี้:

ไอเท็ม	คำอธิบาย
rw1	ระบุชื่อเชิงสัญลักษณ์ของ rwlock ในฟอร์ม <i>rw RwlockNumber</i>
flag_value	ระบุคูปแฟล็ก
เจ้าของ	ระบุเจ้าของ rwlock
status	ระบุผู้ที่ถือครอง rwlock ค่าคือ read (ถ้าถือครองโดยผู้อ่าน), write (ถ้าถือครองโดยผู้เขียน), free (ถ้าว่าง)
wsleep[#]	ระบุการบล็อก threads ในการเขียน # ระบุจำนวนรวมของการบล็อก threads ในการเขียน
rsleep[#]	ระบุการบล็อก threads ในการอ่าน # ระบุจำนวนรวมของการบล็อก threads ในการอ่าน

**หมายเหตุ:** คำสั่งย่อย **print** ของดีบั๊กโปรแกรม **dbx** รู้จักชื่อ **rwlock** เชิงสัญลักษณ์ และสามารถถูกใช้เพื่อแสดงสถานะของอ็อบเจกต์ที่เกี่ยวข้อง

### แฟล็ก

ไอเท็ม	คำอธิบาย
read	แสดงข้อมูลเกี่ยวกับ rwlocks ทั้งหมดซึ่งสถานะอยู่ในโหมด read
write	แสดงข้อมูลเกี่ยวกับ rwlocks ทั้งหมดซึ่งสถานะอยู่ในโหมด write

### ตัวอย่าง

1. เมื่อต้องการแสดงข้อมูลเกี่ยวกับ rwlocks ทั้งหมดให้ป้อน:

```
rwlock
```

เอาต์พุต คล้ายกับ:

```
rw1      flag_value   owner status
$rw1    1           $t1   write
        rsleeps[   0]:
        wsleeps[   0]:
```

2. เมื่อต้องการแสดงข้อมูลเกี่ยวกับ rwlocks ทั้งหมดในโหมด write:

```
rwlock write
```

เอาต์พุต คล้ายกับ:

```
rw1      flag_value   owner status
$rw1    1           $t1   write
        rsleeps[   0]:
        wsleeps[   0]:
```

ดูที่คำสั่งย่อย `attribute`, คำสั่งย่อย `condition`, คำสั่งย่อย `mutex`, คำสั่งย่อย `print`, คำสั่งย่อย และ `thread`

## คำสั่งย่อย `run`

```
run [ Arguments ] [ <File ] [ >File ] [ > >File ] [ 2> >File ] [ >&File ] [ > >&File ]
```

คำสั่งย่อย `run` สตาร์ท อ็อบเจกต์ไฟล์ โดย `Arguments` ถูกส่ง เป็นอาร์กิวเมนต์บรรทัดรับคำสั่ง

## แฟล็ก

ไอเท็ม	คำอธิบาย
<File	เปลี่ยนทิศทางอินพุตเพื่อที่จะรับอินพุตจาก File
>File	เปลี่ยนทิศทางเอาต์พุตไปที่ File
2>File	เปลี่ยนทิศทางขอผิดพลาดมาตรฐานไปที่ File
> >File	ผนวกเอาต์พุตที่เปลี่ยนทิศทางไปที่ File
2> >File	ผนวกขอผิดพลาดมาตรฐานที่เปลี่ยนทิศทางไปที่ File
>&File	เปลี่ยนทิศทางเอาต์พุตและขอผิดพลาดมาตรฐานไปที่ File
> >&File	ผนวกเอาต์พุตและขอผิดพลาดมาตรฐานไปที่ File

## ตัวอย่าง

เมื่อต้องการรันแอปพลิเคชันที่มีอาร์กิวเมนต์ `blue` และ `12` ให้ป้อน:

```
run blue 12
```

ดูที่คำสั่งย่อย `rerun`

## คำสั่งย่อย `screen`

### หน้าจอ

คำสั่งย่อย `screen` เปิด Xwindow เพื่อการโต้ตอบกับคำสั่ง `dbx` คุณทำงานต่อในหน้าต่างซึ่งกระบวนการถูกสร้างขึ้น

คำสั่งย่อย `screen` ต้องรันขณะที่โปรแกรมดีบั๊ก `dbx` กำลังรันในสภาวะแวดล้อม X Window System ถ้าคำสั่งย่อย `screen` ถูกเรียกใช้ในสภาวะแวดล้อมที่ไม่ใช่ Xwindow โปรแกรม `dbx` จะแสดงข้อความเตือน และทำการดีบั๊กต่อ เสมือนว่าไม่ได้เรียกใช้คำสั่งย่อย `screen` คำสั่งย่อย `screen` อาจทำงานไม่สำเร็จในสถานการณ์ต่อไปนี้:

- โปรแกรม `dbx` จะไม่รันในสภาวะแวดล้อม X Window System
- X Window System กำลังทำงานแต่ตัวแปร `dbx global $xdisplay` ไม่ถูกตั้งค่า เพื่อแสดงชื่อที่ถูกต้อง ตัวแปร `$xdisplay` ถูกกำหนดค่าเริ่มต้นเป็นตัวแปรสภาวะแวดล้อม `DISPLAY` คำสั่งย่อย `dbx set Name=Expression` เปลี่ยน ค่าของชื่อจอแสดงผล
- X Window System รันอยู่ แต่ตัวแปรสภาวะแวดล้อม `TERM` ไม่ถูกตั้งค่าเป็นชื่อคำสั่งที่ถูกต้องเพื่อเริ่มต้นหน้าต่างใหม่
- ไดรฟ์ทอริ /tmp ไม่อนุญาตการเข้าถึงเพื่ออ่านหรือเขียนกับโปรแกรม โปรแกรม `dbx` ต้องการพื้นที่จำนวนไม่มากในไดเรกทอรีนี้ เมื่อคำสั่ง `screen` ถูกเรียกใช้งาน
- ระบบไม่มีรีซอร์สเพียงพอ เพื่อรองรับ Xwindow ใหม่

โปรแกรม `dbx` ไม่ได้ แยกระหว่างชนิดความล้มเหลวที่ต่างกัน แต่โปรแกรม ส่งข้อความต่อไปนี้:

คำเตือน: screen คำสั่งย่อย dbx ล้มเหลว dbx continues.

ถ้า `$xdisplay` ถูกตั้งค่าเป็นการแสดงรีโมต คุณอาจไม่สามารถเห็น Xwindow ที่เพิ่งสร้างใหม่ หากค่าที่ตั้ง `$xdisplay` ไม่ถูกต้อง X Window System หรือรีซอร์สของระบบอื่นๆ รายงานปัญหา

คอนฟิกูเรชันที่ผู้ใช้งานกำหนดเองของหน้าต่าง ที่สร้างใหม่ สามารถถูกกำหนดภายใต้ชื่อแอ็พพลิเคชัน `dbx_term` ในไฟล์ `.Xdefaults`

## ตัวอย่าง

เมื่อต้องการเปิด Xwindow เพื่อโต้ตอบคำสั่ง `dbx` ให้ป้อน:

หน้าจอ

ดูที่ การแยกเอาต์พุต dbx ออกจากเอาต์พุตโปรแกรม in *General Programming Concepts: Writing and Debugging Programs* และ AIXwindows ภาพรวม, ใน *AIX Version 6.1 AIXwindows Programming Guide*.

## คำสั่งย่อย set

`set [ Variable=Expression ]`

คำสั่งย่อย `set` กำหนดค่าสำหรับตัวแปรที่บั๊กโปรแกรม `dbx` ค่าถูกระบุโดยพารามิเตอร์ `Expression` ตัวแปรพารามิเตอร์ถูกระบุโดยพารามิเตอร์ `Variable` ชื่อของตัวแปรต้องไม่ขัดแย้งกับชื่อในโปรแกรมที่กำลังถูกดีบั๊ก ตัวแปรถูกขยายเป็นนิพจน์ที่ตรงกันภายในคำสั่งอื่น ถ้าคำสั่งย่อย `set` ถูกใช้โดยไม่มีอาร์กิวเมนต์ ตัวแปรที่เซตในปัจจุบันจะถูกแสดง

ตัวแปร	คำอธิบาย
<code>\$catchbp</code>	จับจุดพักระหว่างการเรียกใช้คำสั่งถัดไป
<code>\$codepage</code>	ระบุโค้ดเซตที่จะใช้เพื่อการแปล อักขระภายในโปรแกรม เมื่อระบุกับโค้ดเพจที่ต้องการ อักขระทั้งหมดถูกอ่านจากโค้ดเซตที่ระบุและแปลง เป็นโค้ดเซตที่ใช้อยู่โดยสภาวะแวดล้อมปัจจุบัน
<code>\$compact_bt_ident</code>	ระบุขีดจำกัดสำหรับจำนวนของอักขระ ในชื่อตัวบ่งชี้ที่สามารถพิมพ์ในการติดตามสแต็ก ขีดจำกัดที่ระบุ ต้องเป็นเลขจำนวนเต็มบวกในช่วง 4-128 หาก ตัวแปรนี้ถูกตั้งค่าโดยไม่ได้ระบุค่าจำกัดใดๆ ค่าดีฟอลต์ของจำนวนอักขระที่สามารถพิมพ์ได้ คือ 8
	หากตั้งค่าตัวแปร และชื่อตัวบ่งชี้ยาวกว่าขีดจำกัดที่ระบุอักขระหรือมากกว่า คำสั่ง <code>dbx</code> จะพิมพ์ จำนวนอักขระที่ระบุของชื่อตัวบ่งชี้ต้นฉบับในการติดตามสแต็กตามด้วยจุดสามจุด (...).
<code>\$compact_bt_string</code>	ตัวอย่างเช่น หากชื่อตัวบ่งชี้คือ <code>variable_example</code> ซึ่งยาว 16 อักขระ และขีดจำกัดที่ระบุคือ 7 ชื่อตัวบ่งชี้ที่พิมพ์จะเป็น <code>variabl...</code> อย่างไรก็ตาม หากชื่อของ ตัวบ่งชี้ คือ <code>variable_1</code> ซึ่งยาว 10 อักขระ และขีดจำกัดที่ระบุ คือ 7 คำสั่ง <code>dbx</code> จะไม่ตัด ชื่อตัวบ่งชี้เป็นเจ็ดอักขระตามด้วย จุด โดยจะพิมพ์เป็น <code>variable_1</code>
	ระบุขีดจำกัดสำหรับจำนวนของอักขระ ในสตริงอาร์กิวเมนต์ฟังก์ชันที่สามารถพิมพ์ในการติดตามสแต็ก ขีดจำกัดที่ระบุ ต้องเป็นเลขจำนวนเต็มบวกในช่วง 4-128 หาก ตัวแปรนี้ถูกตั้งค่าโดยไม่ได้ระบุค่าจำกัดใดๆ ค่าดีฟอลต์ของจำนวนอักขระที่สามารถพิมพ์ได้ คือ 8
	หากตั้งค่าตัวแปร และความยาวของสตริงยาวกว่าขีดจำกัดที่ระบุอักขระหรือมากกว่า คำสั่ง <code>dbx</code> จะพิมพ์ จำนวนของอักขระที่ระบุของสตริงต้นฉบับในการติดตามสแต็ก ตามด้วยจุดสามจุด (...).
	ตัวอย่างเช่น หากสตริง คือ <code>string_example</code> ซึ่งยาว 14 อักขระ และขีดจำกัดที่ระบุคือ 5 สตริงจะพิมพ์เป็น <code>string...</code> อย่างไรก็ตาม หากสตริงคือ <code>string_1</code> ซึ่งยาว 8 อักขระ และขีดจำกัดที่ระบุคือ 5 คำสั่ง <code>dbx</code> จะไม่ตัดสตริงเป็นห้าอักขระตามด้วยจุด โดยจะพิมพ์เป็น <code>string_1</code>
<code>\$deferevents</code>	เปิดคุณลักษณะเหตุการณ์ที่เลื่อนเวลาออกไป
<code>\$display_address_name</code>	แสดงตัวบ่งชี้ตัวแปรสมาชิกและ แอดเดรสหน่วยความจำที่ตัวบ่งชี้ใช้เมื่อตรวจสอบชุดของ แอดเดรสหน่วยความจำที่ใช้คำสั่ง <code>dbx</code>

ตัวแปร	คำอธิบาย
\$expandunions	แสดงค่าสำหรับแต่ละส่วนของ variant records หรือ unions
\$frame	ใช้สแต็กเฟรมที่ชี้โดยแอดเดรสที่กำหนด โดยค่าของ \$frame สำหรับการติดตามสแต็ก และการเข้าถึงตัวแปรโลคัล
\$hexchars	พิมพ์อักขระเป็นค่าเลขฐานสิบหก
\$hexin	แปลแอดเดรสเป็นเลขฐานสิบหก
\$hexints	พิมพ์จำนวนเต็มเป็นค่าเลขฐานสิบหก
\$hexstrings	พิมพ์ตัวชี้อักขระเป็นเลขฐานสิบหก
\$hold_next	ถือค่า threads ทั้งหมดยกเว้น thread ที่รันระหว่างคำสั่งย่อย cont, next, nexti และ step การตั้งค่าตัวแปรนี้อาจทำให้เกิด deadlock เนื่องจากเธรดที่กำลังรอนั้นรอการล็อกที่เกิดขึ้นโดยหนึ่งในเธรดที่ถูกบล็อก
\$ignoreifhandler	ไม่หยุดทำงานเมื่อโปรแกรมของคุณได้รับสัญญาณซึ่งมี handler ที่รีจิสเตอร์
\$ignoreload	ไม่หยุดทำงานเมื่อโปรแกรมดำเนินการที่น้อยลง load, unload หรือ loadbind
\$ignorenonbptrap	ไม่หยุดทำงานเมื่อโปรแกรมของคุณพบคำสั่งที่ไม่จับ จุดพัก และมี SIGTRAP handler ที่รีจิสเตอร์
\$instructionset	เขียนทับโหมดการถอดแยกภาษาแอสเซมบลีฟอลต์ รายการต่อไปนี้มีค่าที่เป็นไปได้สำหรับพารามิเตอร์ <i>Expression</i> :
	"default" ระบุสถาปัตยกรรมซึ่งโปรแกรม dbx รันอยู่
	"com" ระบุชุดคำสั่งสำหรับโหมด intersection ทั่วไปของสถาปัตยกรรม PowerPC และ POWER family ดีฟอลต์โปรแกรม dbx เป็นตัวช่วยจำ POWER processor-based
	"pwr" ระบุชุดคำสั่งและคำย่อสำหรับสถาปัตยกรรม POWER family
	"pwrx" ระบุชุดคำสั่งและคำย่อสำหรับการนำ POWER2 มาใช้ของสถาปัตยกรรม POWER family สำหรับ AIX 5.1 และรุ่นสูงกว่า
	"pwr6" ระบุชุดคำสั่งและคำย่อสำหรับการนำ POWER6® ไปใช้ของสถาปัตยกรรม PowerPC
	"pwr7" ระบุชุดคำสั่งและคำย่อสำหรับการนำ POWER7® ไปใช้ของสถาปัตยกรรม PowerPC
	"pwr8" ระบุชุดคำสั่งและ mnemonics สำหรับการนำ POWER8® ไปใช้ของสถาปัตยกรรม PowerPC
	"601" ระบุชุดคำสั่งและคำย่อสำหรับ PowerPC 601 RISC Microprocessor สำหรับ AIX 5.1 และรุ่นสูงกว่า
	"603" ระบุชุดคำสั่งและคำย่อสำหรับ PowerPC 603 RISC Microprocessor สำหรับ AIX 5.1 และรุ่นสูงกว่า
	"604" ระบุชุดคำสั่งเครื่องและ mnemonics สำหรับ PowerPC 604™ RISC Microprocessor
	"970" ระบุชุดคำสั่งและคำย่อสำหรับไมโครโพรเซสเซอร์ PowerPC 970
	"ppc" ระบุชุดคำสั่งและคำย่อที่กำหนดในสถาปัตยกรรม POWER processor-based ไม่รวมคำสั่งทางเลือก คำแนะนำเหล่านี้พร้อมใช้งาน ในการนำไปใช้งาน POWER processor-based ทั้งหมด ยกเว้น PowerPC 601 RISC Microprocessor ใน AIX 5.1 และต่ำกว่า
	"any" ระบุคำสั่ง POWER processor-based หรือ POWER family ที่ใช้ได้ สำหรับชุดคำสั่งที่ทับซ้อน ดีฟอลต์คือตัวย่อ POWER processor-based
\$java	ถ้าไม่มีค่าถูกเซตสำหรับพารามิเตอร์ <i>Expression</i> โปรแกรม dbx จะใช้โหมดการถอดแยกภาษาแอสเซมบลีฟอลต์ เมื่อตั้งค่าง่ายค่าตัวแปรต่อไปนี้ ทำให้คำสั่ง dbx อยู่ในโหมดเพื่อดีบั๊กแอปพลิเคชัน Java เมื่อไม่เซต เป็นการไม่เซตตัวแปรต่อไปนี้เช่นกัน:
	<b>\$ignorenonbptrap</b> ยับยั้งการแจ้งเตือนของคำสั่งกับดักที่สร้างโดยคอมไพเลอร์ Java Just-In-Time (JIT)
\$listwindow	ระบุจำนวนบรรทัดเพื่อแสดงฟังก์ชันและ หมายเลขให้กับการเมื่อคำสั่งย่อย list ถูกใช้โดยไม่มีพารามิเตอร์ ดีฟอลต์คือ 10 บรรทัด
\$mapaddrs	สตรัทการแม็ปแอดเดรส ยกเลิกการเซต \$mapaddrs หยุด การแม็ปแอดเดรส

ตัวแปร  
\$mapformat

คำอธิบาย  
ระบุโหมดเอาต์พุตที่พอลต์สำหรับคำสั่งย่อย map

- "abbr" ระบุโหมดเอาต์พุตแบบย่อ ซึ่งประกอบด้วย บรรทัดหนึ่งสำหรับแต่ละโมดูลที่โหนดที่มีหมายเลขรายการ ชื่อโมดูล และชื่อสมาชิกเป็นทางเลือก สำหรับโมดูลนั้น
- "normal" ระบุโหมดเอาต์พุตปกติ ซึ่งประกอบด้วย หมายเลขรายการ ชื่อโมดูล ชื่อสมาชิก จุดเริ่มต้นข้อความ ความยาวข้อความ จุดเริ่มต้นข้อมูล ความยาวข้อมูล และไฟล์ descriptor สำหรับแต่ละโมดูลที่โหนด ถ้าโมดูลที่โหนดมีข้อมูล TLS จุดเริ่มต้นข้อมูล TLS และความยาวข้อมูล TLS ถูกแสดง
- "raw" ระบุโหมดเอาต์พุต raw ซึ่งประกอบด้วย บรรทัดที่ไม่ได้จัดรูปแบบสำหรับแต่ละโมดูลที่มีฟิลด์ที่แยกด้วยช่องว่าง: หมายเลขรายการ ชื่อโมดูลที่มี ชื่อสมาชิกที่เป็นทางเลือก จุดเริ่มต้นข้อความ จุดสิ้นสุดข้อความ ความยาวข้อความ ความยาวข้อมูล จุดเริ่มต้นข้อมูล จุดสิ้นสุดข้อมูล ความยาวข้อมูล และไฟล์ descriptor ถ้าโมดูลที่โหนดมีข้อมูล TLS จุดเริ่มต้น ข้อมูล TLS จุดสิ้นสุดข้อมูล TLS และความยาวข้อมูล TLS จะถูกแสดง

"verbose"  
ระบุโหมดเอาต์พุต verbose ซึ่งประกอบด้วย หมายเลขรายการ ชื่อโมดูล ชื่อสมาชิก จุดเริ่มต้นข้อความ จุดสิ้นสุดข้อความ ความยาวข้อความ จุดเริ่มต้นข้อมูล จุดสิ้นสุดข้อมูล ความยาวข้อมูล และไฟล์ descriptor สำหรับแต่ละโมดูลที่โหนด ถ้าโมดูลที่โหนดมีข้อมูล TLS จุดเริ่มต้น ข้อมูล TLS จุดสิ้นสุดข้อมูล TLS และความยาวข้อมูล TLS จะถูกแสดง

\$mnemonics

หากไม่ได้ตั้งค่าสำหรับพารามิเตอร์ *Expression* โปรแกรม dbx จะใช้โหมดเอาต์พุตปกติ เปลี่ยนชุดของคำย่อที่จะถูกใช้โดยโปรแกรม dbx เมื่อทำการถอดแยกภาษาแอสเซมบลี

- "default" ระบุคำย่อที่ใกล้เคียงที่สุดกับชุดคำสั่งเครื่อง ที่ระบุ
- "pwr" ระบุคำย่อสำหรับสถาปัตยกรรม POWER family
- "ppc" ระบุคำย่อที่กำหนดในหนังสือสถาปัตยกรรม POWER processor-based โดยแยกคำสั่งทางเลือก

ถ้าไม่มีค่าถูกตั้งสำหรับพารามิเตอร์ *Expression* โปรแกรม dbx จะใช้ mnemonics ที่ใกล้เคียงกับชุดคำสั่งเครื่อง ที่ระบุมากที่สุด

\$noargs  
\$nofregs  
\$novregs  
\$novsregs  
\$octint  
\$octints  
\$pretty

เว้นอาร์กิวเมนต์จากคำสั่งย่อย เช่น *where*, *up*, *down* และ *dump*  
เว้นการแสดงผลของรีจิสเตอร์ดัชนีจากคำสั่งย่อย *registers*  
เว้นการแสดงผลของเวกเตอร์รีจิสเตอร์จากคำสั่งย่อย *registers*  
เว้นการแสดงผลของเวกเตอร์สเกลาร์รีจิสเตอร์จากคำสั่งย่อย *registers*  
แปลแอดเดรสเป็นเลขฐานแปด  
พิมพ์จำนวนเต็มเป็นเลขฐานแปด  
แสดงค่าโครงสร้างข้อมูล complex C และ C++ (struts, unions, arrays) ในรูปแบบ *พิมพ์ให้อ่านง่าย* ด้วยคำสั่งย่อย *print*

"on" ระบุการพิมพ์ pretty กับแต่ละค่าบนตัวบรรทัดเองและ กับการเยื้อง เพื่อแสดงขอบเขตสแตคของแต่ละค่า

"verbose"  
ระบุการพิมพ์ pretty กับแต่ละค่าบนตัวบรรทัดเองและ กับชื่อที่มีการระบุ เพื่อแสดงขอบเขตสแตคของแต่ละค่า ชื่อที่ระบุประกอบด้วยรายการที่แยกด้วยจุดของบล็อกวงนอก กับค่าที่เชื่อมโยง

"off" ระบุการพิมพ์ pretty เป็นปิด ค่านี้เป็นค่าดีพอลต์  
แสดงชนิดไดนามิกของอ็อบเจกต์ C++ กับคำสั่ง *print / dump* โดยดีพอลต์ตัวแปรนี้ไม่ถูกเซต  
ทำซ้ำคำสั่งก่อนหน้านี้ ถ้าไม่มีการป้อนคำสั่ง  
บล็อกสัญญาณที่ไปยังโปรแกรมของคุณ  
แสดง Virtual Function Table ขณะพิมพ์อ็อบเจกต์ C++ ด้วยคำสั่ง *print / dump* โดยดีพอลต์ไม่มีการเซต  
แสดงหมายเลขเฟรมและรีจิสเตอร์เซตสำหรับแต่ละแอ็คทีฟ ฟังก์ชันหรือไพธอนที่แสดงโดยคำสั่งย่อย *where*

<p>ตัวแปร \$stepignore</p>	<p>คำอธิบาย ควบคุมวิธีที่คำสั่ง dbx ทำงาน เมื่อคำสั่งย่อย step/tstep รัน บนบรรทัดซอร์สที่เรียกทีอื่น ซึ่งไม่มีข้อมูล การดีบั๊ก ตัวแปรนี้เปิดใช้คำสั่งย่อย step/tstep เพื่อข้ามรู้ทีขนาดใหญ่ ซึ่งไม่มีข้อมูล การดีบั๊ก รายการต่อไปนี้มี ค่าที่เป็นไปได้สำหรับพารามิเตอร์ Expression :</p> <p>"function" ดำเนินฟังก์ชันของคำสั่งย่อย next/tnext สำหรับคำสั่ง dbx นี้คือค่าดีฟอลต์</p> <p>"module" ดำเนินฟังก์ชันของคำสั่งย่อย next/tnext ถ้าฟังก์ชันอยู่ในโหนดโมดูลซึ่งไม่มีข้อมูลดีบั๊ก (เช่นไลบรารีระบบ)</p> <p>"none" ดำเนินฟังก์ชันของคำสั่งย่อย stepi/tstepi สำหรับคำสั่ง dbx ในแบ็กกราวนด์จนกว่า คำสั่งทำงานถึงคำสั่งที่มีข้อมูลซอร์ส ณ จุดนั้น dbx แสดง ตำแหน่งที่หยุดการเรียกใช้งาน</p>
<p>\$trace_good_transaction</p>	<p>ใช้คำสั่ง dbx เพื่อแสดงข้อความต่อไปนี้ทุกครั้งที่ธุรกรรม transactional memory (TM) เสร็จสมบูรณ์</p> <p>กระบวนการ {PID} จากดำเนินการธุรกรรม - \$texasr, \$tfiar, \$tfhar ถูกต้องและอาจถูกตรวจสอบ</p>
<p>\$thcomp \$unsafeassign</p>	<p>ธุรกรรมที่เสร็จสมบูรณ์ไม่ถูกตรวจสอบเนื่องจากตัวแปรจะปิดใช้งานโดยดีฟอลต์ เมื่อ \$thcomp ถูกเซต ข้อมูล ที่แสดงโดยคำสั่ง thread th- ถูกแสดง ในรูปแบบที่บีบอัด</p> <p>ปิดการตรวจสอบชนิดที่เข้มงวด ระหว่างสองด้านของคำสั่ง assign เมวาทตัวแปร \$unsafeassign จะถูกตั้งค่า สองด้านของข้อความสั่ง assign อาจไม่มีชนิดหน่วยเก็บข้อมูล ที่มีขนาดต่างกัน</p>
<p>\$unsafebounds \$unsafeCALL \$unsafegoto \$vardim \$xdisplay</p>	<p>ปิดการตรวจสอบ subscript บน arrays</p> <p>ปิดการตรวจสอบชนิดอย่างเข้มงวดสำหรับอาร์กิวเมนต์ที่ส่งไปที่การเรียกทีย่อย หรือฟังก์ชัน</p> <p>ปิดการตรวจสอบปลายทางคำสั่งย่อย goto</p> <p>ระบุความยาวมิติที่จะใช้เมื่อพิมพ์ arrays ด้วยขอบเขตที่ไม่ได้กำหนด ค่าดีฟอลต์คือ 10</p> <p>ระบุชื่อที่แสดงสำหรับ X Window System สำหรับใช้กับคำสั่งย่อย multproc หรือคำสั่งย่อย screen ดีฟอลต์คือค่าตัวแปร DISPLAY ของเซลล์</p>

ตัวแปร \$unsafe จัดกักการใช้งานของดีบั๊กโปรแกรม dbx ในการตรวจหาข้อผิดพลาด

### ตัวอย่าง

- เมื่อต้องการเปลี่ยนจำนวนบรรทัดที่แสดงรายการ เป็น 20 ให้ป้อน:

```
set $listwindow=20
```
- เมื่อต้องการปิดใช้งานการตรวจสอบการพิมพ์บนคำสั่งย่อย assign ให้ป้อน:

```
set $unsafeassign
```
- เมื่อต้องการดีสแอสเซมเบิลคำสั่งเครื่องสำหรับตัวประมวลผล POWER7 ให้ป้อน:

```
set $instructionset="pwr7"
```
- เมื่อต้องการแสดงสตริงที่เข้ารหัสในโค้ดเซต IBM-eucCN ให้ป้อน:

```
set $codepage="IBM-eucCN"
```
- เมื่อต้องการระบุขีดจำกัดสี่อักขระในตัวบ่งชี้ และขีดจำกัดสิบสองอักขระในตัวระบุที่แสดงใน ในการติดตามสแต็ก ให้ป้อนคำสั่งต่อไปนี้:

```
set $compact_bt_ident=6
set $compact_bt_string=12
```

การติดตามสแต็กที่ใช้ตัวบ่งชี้ เช่น long\_identifier, long\_variable\_name\_str และ recursive\_fun และสตริง เช่น this\_is\_a\_really\_long\_string มีลักษณะคล้ายกับเอาต์พุตดังต่อไปนี้:

```
long_i...(a = 11, long_v... = "this_is_a_re..."), line 3 in "example.c"
recurs...( ), line 13 in "example.c"
```

ดูที่คำสั่งย่อย `unset` และดูที่ การเปลี่ยนเอาต์พุตการพิมพ์ ด้วยตัวแปรดักโปรแกรมพิเศษ in *General Programming Concepts: Writing and Debugging Programs*

### `set edit [vi, emacs] or set -o [vi, emacs] Subcommand`

คำสั่งย่อย `set` ที่มีอ็อปชัน `-o` หรือ `edit` อาจถูกใช้เพื่อเปิด โหมดหนึ่งในโหมดแก้ไขบรรทัด ถ้ามีการกำหนดคำสั่ง `set -o vi` หรือ `set edit vi` คุณจะถูกลำเข้าสู่โหมดอินพุตของ เอดิเตอร์รายบรรทัด `vi` ถ้ามีการ กำหนดคำสั่ง `set -o emacs` หรือ `set edit emacs` คุณจะถูกลำเข้าสู่โหมดอินพุตของ เอดิเตอร์รายบรรทัด `emacs`

#### ตัวอย่าง

1. เมื่อต้องการเปิดเอดิเตอร์รายบรรทัด `vi` ให้ป้อน:

```
set -o vi
or
set edit vi
```

### คำสั่งย่อย `sh`

`sh [ Command ]`

คำสั่งย่อย `sh` ส่ง คำสั่งที่ระบุโดยพารามิเตอร์ `Command` ไปที่เชลล์เพื่อดำเนินการ ตัวแปรสถานะแวดล้อม `SHELL` กำหนดเชลล์ที่จะถูกใช้ ดีฟอลต์คือเชลล์ `sh` ถ้าไม่มีการระบุอาร์กิวเมนต์ การควบคุม ถูกถ่ายโอนไปที่เชลล์

#### ตัวอย่าง

1. เมื่อต้องการรันคำสั่ง `ls` ให้ป้อน:  

```
sh ls
```
2. เมื่อต้องการออกจากเชลล์ให้ป้อน:  

```
sh
```
3. เมื่อต้องการใช้ตัวแปรสถานะแวดล้อม `SHELL` ป้อน:  

```
sh echo $SHELL
```

ดูที่ การรันคำสั่ง Shell จาก `dbx` in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย `skip`

`skip [ Number ]`

คำสั่งย่อย `skip` ดำเนิน การทำงานของแ็พพลิเคชันโปรแกรมต่อจากจุดหยุดปัจจุบัน จำนวนของจุดพักเท่ากับค่าของพารามิเตอร์ `Number` ถูกข้ามและดำเนินการ จากนั้นหยุดการทำงานเมื่อถึงจุดพักถัดไป หรือเมื่อโปรแกรมเสร็จสิ้น ถ้าไม่ระบุพารามิเตอร์ `Number` ค่าดีฟอลต์จะเป็นหนึ่ง

#### ตัวอย่าง

เมื่อต้องการดำเนินการทำงานต่อจนถึงจุดพักที่สอง ให้ป้อน:

```
skip 1
```

และดูที่คำสั่งย่อย cont

คำสั่งย่อย source

source File

คำสั่งย่อย source อ่านคำสั่งย่อย dbx จากไฟล์ที่ระบุโดยพารามิเตอร์ File

ตัวอย่าง

เมื่อต้องการอ่านคำสั่งย่อย dbx ในไฟล์ cmdfile ให้ป้อน:

```
source cmdfile
```

ดูที่ การอ่านคำสั่งย่อย dbx จากไฟล์ in *General Programming Concepts: Writing and Debugging Programs*

คำสั่งย่อย status

```
status [ more ] [ >File ]
```

คำสั่งย่อย status แสดง breakpoints, tracepoints และ watchpoints ที่ผู้ใช้กำหนดเองทั้งหมด นอกจากจำนวน thread tskip ที่เหลือ (เซ็ทโดยใช้คำสั่งย่อย tskip) ถ้ามีการระบุพารามิเตอร์เพิ่มเติม คำสั่งย่อย status จะแสดง คำสั่งย่อย dbx ที่เชื่อมโยงกับ breakpoints, tracepoints และ watchpoints เช่นกัน คำสั่งย่อย status แสดงเหตุการณ์ที่เปิดใช้งาน ซึ่งมีวงเล็บสี่เหลี่ยม ([]) ล้อมรอบหมายเลข เหตุการณ์ เหตุการณ์ที่ปิดใช้งาน ซึ่งมีจุด (..) ล้อมรอบหมายเลขเหตุการณ์ และเหตุการณ์ที่เลื่อนออกไป ซึ่งมีวงเล็บมุม (<>) ล้อมรอบหมายเลขเหตุการณ์

แฟล็ก > ส่งเอาต์พุตของคำสั่งย่อย status ไปยังไฟล์ ที่ระบุในพารามิเตอร์ File

แฟล็ก

ไอเท็ม	คำอธิบาย
>File	เปลี่ยนทิศทางเอาต์พุตไปที่ File

ตัวอย่าง

1. เมื่อต้องการแสดง breakpoints, tracepoints และ watchpoints ที่ผู้ใช้กำหนดเองทั้งหมด และจำนวน thread tskip ที่เหลือ ให้พิมพ์:

```
status
```

เอาต์พุตมีข้อมูลเหมือนดังนี้:

```
[1] stop at 13
[2] stop at 14
.3. stop at 15
.4. stop at 16
[5] stop at 17 ( count = 0, limit = 3 )
<6> stop at 18 if g > 10
<7> stop in func
```

Remaining tskip counts:  
tskip 2 for \$t1  
tskip 1 for \$t5

ในเอาต์พุตตัวอย่าง เหตุการณ์ 3 และ 4 ถูกปิดใช้งาน และเหตุการณ์ 6 และ 7 ถูกหน่วงเวลา

- เมื่อต้องการแสดงจุดหยุด จุดติดตาม และจุดเฝ้าดูที่ผู้ใช้กำหนดเองทั้งหมด ด้วยคำสั่งย่อย dbx ที่เกี่ยวข้องให้ป้อน:  
status more

เอาต์พุต คล้ายกับ:

```
[1] stop at 13  
[1] where  
.2. stop at 14  
[1] where  
[2] registers  
<3> stop at 15 if g > 10  
[1] where; registers
```

ดูที่คำสั่งย่อย addcmd , คำสั่งย่อย clear , คำสั่งย่อย delete , คำสั่งย่อย delcmd , คำสั่งย่อย tskip , คำสั่งย่อย stop และคำสั่งย่อย trace สำหรับคำสั่ง dbx

และดูที่ การตั้งค่าและการลบบจุดพัก in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย step

step [Number]

คำสั่งย่อย step รัน บรรทัดซอร์สของแอปพลิเคชันโปรแกรม ระบุจำนวนบรรทัด ที่จะถูกเรียกใช้งานด้วยพารามิเตอร์ Number ถ้าพารามิเตอร์ Number ถูกเว้นไว้ ค่าดีฟอลต์ คือ 1

ถ้าคุณใช้คำสั่งย่อย step ในแอปพลิเคชันโปรแกรม multithreaded, threads ผู้ใช้ทั้งหมด ที่รันระหว่างการดำเนินการ แต่โปรแกรมดำเนินการทำงานต่อจนกว่า thread ที่รันอยู่ทำงานถึงบรรทัดซอร์สที่ระบุ ถ้าคุณต้องการทำขั้นตอนการรันเธรดเท่านั้น ให้ใช้คำสั่งย่อย set เพื่อตั้งค่าตัวแปร \$hold\_next การตั้งค่าตัวแปรนี้อาจทำให้เกิด deadlock เนื่องจากเธรดที่กำลังรันอยู่นั้นรอการล็อก ที่เกิดขึ้นโดยหนึ่งในเธรดที่ถูกบล็อก

หมายเหตุ: ใช้ตัวแปร \$stepignore ของ คำสั่งย่อย set เพื่อควบคุม ลักษณะการทำงานของคำสั่งย่อย step ตัวแปร \$stepignore เปิดใช้คำสั่งย่อย step เพื่อข้ามรูทีนขนาดใหญ่ซึ่ง ไม่มีข้อมูลดีบั๊ก

## ตัวอย่าง

- เมื่อต้องการดำเนินการทำงานสำหรับหนึ่งบรรทัดซอร์ส ให้ป้อน:  
step
- เมื่อต้องการดำเนินการทำงานสำหรับห้าบรรทัดซอร์ส ให้ป้อน:  
step 5
- เมื่อต้องการป้องกันโปรแกรม dbx ไม่ให้ทำฟังก์ชัน printf ที่ละคำสั่ง ดังแสดงในโค้ดตัวอย่าง ต่อไปนี้:  
60 printf ("hello world \n");

ให้ป้อน:

```
set $stepignore="function"; step
```

ดูที่คำสั่งย่อย **cont** คำสั่งย่อย **goto** , คำสั่งย่อย **next** , คำสั่งย่อย **set** , คำสั่งย่อย **stepi**

### คำสั่งย่อย **stepi**

**stepi** [ *Number* ]

คำสั่งย่อย **stepi** รับคำสั่งของแอ็พพลิเคชันโปรแกรม ระบุจำนวน คำสั่งที่จะถูกเรียกใช้ในพารามิเตอร์ *Number* ถ้าพารามิเตอร์ *Number* ถูกเว้นไว้ ค่าดีฟอลต์ คือ 1

ถ้าใช้กับแอ็พพลิเคชันโปรแกรม **multithreaded** คำสั่งย่อย **stepi** ทำงานที่ละบรรทัดเฉพาะ **thread** ที่รันอยู่ **threads** ผู้ใช้อื่นทั้งหมดยังคงหยุดทำงาน

### ตัวอย่าง

1. เมื่อต้องการดำเนินการทำงานต่อสำหรับหนึ่งคำสั่งเครื่อง ให้ป้อน:

```
stepi
```

2. เมื่อต้องการดำเนินการทำงานต่อสำหรับห้าคำสั่งเครื่อง ให้ป้อน:

```
stepi 5
```

ดูที่คำสั่งย่อย **gotoi** คำสั่งย่อย **nexti** และคำสั่งย่อย **step**

### คำสั่งย่อย **stop**

```
stop { [ Variable ] [ at SourceLine | in Procedure | on load [ "ModuleName" ] ] [ if Condition ] } [ { "Limit" } ]
```

คำสั่งย่อย **stop** หยุด แอ็พพลิเคชันโปรแกรม เมื่อเป็นไปตามเงื่อนไขที่กำหนด โปรแกรม ถูกหยุดการทำงานเมื่อ:

- *เงื่อนไข* จะเป็น true เมื่อแฟล็ก **if** *Condition* ถูกนำมาใช้
- *Procedure* ถูกเรียกถ้าแฟล็ก **in** *Procedure* ถูกใช้
- *Variable* ถูกเปลี่ยนถ้ามีการระบุพารามิเตอร์ *Variable*
- การทำงานมาถึงหมายเลขบรรทัด *SourceLine* ถ้าแฟล็ก **at** *SourceLine* ถูกใช้  
ตัวแปร *SourceLine* สามารถถูกระบุเป็นจำนวนเต็มหรือสตริงชื่อไฟล์ที่ตามด้วย : (โคลอน) และจำนวนเต็ม
- โมดูล *ModuleName* ที่โหลดถูกโหลด หรือยกเลิกการโหลดถ้าใช้แฟล็ก **on load** และ ระบุพารามิเตอร์ *ModuleName*  
ตัวแปร *ModuleName* เป็นทางเลือกที่สามารถถูกระบุ เป็นชื่อโมดูลเดี่ยว หรือเป็นชื่อโมดูลที่คู่กับชื่อสมาชิก ในรูปแบบ:  
*ModuleName*( *MemberName* )
- โมดูลที่โหลดใดๆ ที่ถูกโหลด หรือเลิกโหลดถ้าแฟล็ก **on load** ถูกใช้และไม่ได้ระบุพารามิเตอร์ *ModuleName*

คุณสามารถตั้งค่าพารามิเตอร์ *Limit* เพื่อ ส่งให้คำสั่ง **dbx** ข้ามเงื่อนไขสำหรับจำนวนครั้งที่ระบุ อีกนัยหนึ่ง พารามิเตอร์ *Limit* ระบุจำนวนครั้งที่เงื่อนไขที่ระบุต้องครบตาม ก่อนที่การเรียกใช้งานโปรแกรมตีบั๊กจะหยุดทำงาน

หลังจาก คำสั่งเหล่านั้น โปรแกรมดีบั๊ก dbx ตอบกลับด้วย ข้อความรายงานเหตุการณ์ที่คำสั่งสร้างขึ้นเป็นผลลัพธ์ของคำสั่งของคุณ ข้อความรวมถึง event ID ที่เชื่อมโยง กับจุดพักของคุณกับการแปลคำสั่งของคุณ ไวยากรณ์ของการแปลความหมาย อาจไม่เหมือนกับคำสั่งของคุณ ตัวอย่างเช่น :

```
stop in main
[1] stop in main
stop at 19 if x == 3
[2] stop at "hello.c":19 if x = 3
stop in func
<3> stop in func
stop g
<4> stop g
stop in getdata {3}
[5] stop in getdata ( count = 0, limit = 3 )
```

หมายเลขในวงเล็บเหลี่ยม ([]) คือ identifiers เหตุการณ์ที่เชื่อมโยงกับจุดพัก ดีบั๊กโปรแกรม dbx เชื่อมโยงหมายเลขเหตุการณ์กับแต่ละคำสั่งย่อย stop เมื่อโปรแกรมถูกหยุด เนื่องจากผลของหนึ่งในเหตุการณ์ identifier เหตุการณ์ถูกแสดงตามบรรทัดปัจจุบันเพื่อแสดง เหตุการณ์ที่ทำให้โปรแกรมหยุดทำงาน ตัวเลขในวงเล็บมุม (<>) คือตัวบ่งชี้เหตุการณ์สำหรับเหตุการณ์ที่หน่วงเวลา เหตุการณ์ที่เลื่อนไป คือเหตุการณ์ที่ไม่มี breakpoint, tracepoint หรือ watchpoint เชื่อมโยงอยู่ และถูกสร้างเมื่อคำสั่ง input เชื่อมโยง สัญลักษณ์ที่ไม่ได้ถูกโหลดลงหน่วยความจำในขณะนี้ เหตุการณ์ ปกติที่แสดงในวงเล็บเหลี่ยม ([]) ถูกแปลงลงในเหตุการณ์ที่เลื่อนไป เช่นกัน เมื่อโมดูลที่ตรงกันถูกยกเลิกการโหลด เมื่อใดก็ตามที่โมดูลที่ตรงกับเหตุการณ์ที่เลื่อนไปถูกโหลดลงในหน่วยความจำ เหตุการณ์ที่เลื่อนไปถูกแปลงลงในเหตุการณ์ปกติ และ breakpoint, tracepoint หรือ watchpoint ที่ตรงกันถูกสร้าง เหตุการณ์ที่คุณสามารถมีอยู่ร่วมกับเหตุการณ์ภายในที่สร้างโดยคำสั่ง dbx ดังนั้นจำนวนเหตุการณ์อาจไม่เรียงลำดับเสมอไป

ขีดจำกัดอาจเกี่ยวข้องกับเหตุการณ์หลังจาก ถูกสร้างขึ้นโดยใช้คำสั่งย่อย limitbp เมื่อต้องการดูขีดจำกัดที่เกี่ยวข้องกับเหตุการณ์ สามารถใช้คำสั่งย่อย printbp

ใช้คำสั่งย่อย status เพื่อดูหมายเลขเหล่านี้ คุณสามารถเปลี่ยนทิศทางเอาต์พุตจาก status ไปที่ไฟล์ ใช้คำสั่งย่อย delete หรือ clear เพื่อปิดคำสั่งย่อย stop หรือใช้คำสั่งย่อย enable หรือ disable ใช้คำสั่งย่อย addcmd เพื่อเพิ่มคำสั่งย่อย dbx เข้ากับหมายเลขเหตุการณ์ที่ระบุ และคำสั่งย่อย delcmd เพื่อลบคำสั่งย่อย dbx ที่เชื่อมโยงออกจาก หมายเลขเหตุการณ์ที่ระบุ

ในแอปพลิเคชันโปรแกรม multithreaded threads ผู้ใช้ทั้งหมดถูกหยุดทำงาน เมื่อมี thread ผู้ใช้ทำงานถึงจุดพัก จุดหยุดที่กำหนดบนบรรทัดซอร์ส หรือฟังก์ชัน ถูกเข้าถึงโดยเธรดผู้ใช้ที่เรียกใช้บรรทัดหรือฟังก์ชัน ยกเว้น คุณระบุเงื่อนไข (เช่นในตัวอย่างที่ 9) alias ต่อไปนี้ระบุเงื่อนไขโดยอัตโนมัติ:

- **bftth**(Function, ThreadNumber)
- **blth**(SourceLine, ThreadNumber)

ThreadNumber คือ ส่วนหมายเลขของชื่อ thread เชิงสัญลักษณ์ตามที่รายงานโดยคำสั่งย่อย thread (ตัวอย่าง 5 คือ ThreadNumber สำหรับ thread ชื่อ \$t5) นามแฝงเหล่านี้ เป็นแม่โครซึ่งสร้างคำสั่งย่อยเพิ่มอย่างแท้จริงดังแสดง ในตัวอย่างต่อไปนี้:

```
stopi at &Function if ($running_thread == ThreadNumber)
stop at SourceLine if ($running_thread == ThreadNumber)
```

## แฟล็ก

ไอเท็ม	คำอธิบาย
at <i>SourceLine</i>	ระบุหมายเลขบรรทัด
if <i>Condition</i>	ระบุเงื่อนไข เช่น true
in <i>Procedure</i>	ระบุโปรซีเจอร์ที่จะถูกเรียก
on load <i>ModuleName</i>	ระบุโมดูลที่โหลดที่จะถูกมอนิเตอร์

## ตัวอย่าง

- เมื่อต้องการหยุดการประมวลผลที่คำสั่งแรกในโปรซีเจอร์ main ให้ป้อน:

```
stop in main
```

- เมื่อต้องการหยุดการประมวลผลเมื่อค่าของตัวแปร x ถูกเปลี่ยนบนบรรทัดที่ 12 ของการประมวลผล ให้ป้อน:

```
stop x at 12
```

- เมื่อต้องการหยุดการประมวลผลที่บรรทัดที่ 5 ในไฟล์ sample.c ให้ป้อน:

```
stop at "sample.c":5
```

- เมื่อต้องการตรวจสอบค่าของ x ในแต่ละครั้งที่คำสั่ง dbx ระบุที่นัยภายใน func1 ให้ป้อน:

```
stop in func1 if x = 22
```

- เมื่อต้องการตรวจสอบค่าของ x ในแต่ละครั้งที่คำสั่ง dbx เริ่มต้นรัน func1 ให้ป้อน:

```
stopi at &func1 if x = 22
```

- เมื่อต้องการหยุดโปรแกรมเมื่อค่าของ Variable เปลี่ยนแปลง ให้ป้อน:

```
stop Variable
```

- เมื่อต้องการหยุดโปรแกรมเมื่อ Condition ประเมินเป็น true ให้ป้อน:

```
stop if (x > y) and (x < 2000)
```

- ตัวอย่างต่อไปนี้แสดงวิธี แสดงเหตุการณ์แอ็คทีฟและการลบออก:

```
status
[1] stop in main
[2] stop at "hello.c":19 if x = 3
delete 1
status
[2] stop at "hello.c":19 if x = 3
clear 19
status
(dbx)
```

คำสั่ง **delete** ลบเหตุการณ์ ตาม identifier เหตุการณ์ คำสั่ง **clear** ลบ จุดพักตามหมายเลขบรรทัด

- เมื่อต้องการวางจุดหยุดที่เริ่มต้นของ func1 เฉพาะเมื่อถูกเรียกใช้โดยเธรด \$t5 ให้ป้อนคำสั่งที่เทียบกับ ต่อไปนี้:

```
stopi at &func1 if ($running_thread == 5)
```

หรือ

```
bfth(func1, 5)
```

- เมื่อต้องการหยุดโปรแกรม เมื่อมีโมดูลถูกโหลดหรือยกเลิกการโหลด ให้ป้อน:

```
stop on load
```

- เมื่อต้องการหยุดโปรแกรมเมื่อโมดูล Module ถูกโหลด หรือยกเลิกการโหลด ให้ป้อน:

```
stop on load "Module"
```

12. เมื่อต้องการหยุดโปรแกรมเมื่อสมาชิก Member ของโมดูล Module ถูก โหลดหรือยกเลิกการโหลด ให้ป้อน:

```
stop on load "Module(Member)"
```

13. เมื่อต้องการหยุดทำงานโปรแกรมในฟังก์ชัน getdata เมื่อ ถูกเรียกใช้ เป็นครั้งที่สาม ให้ป้อน:

```
stop in getdata {3}
```

ดูที่คำสั่งย่อย **addcmd**, คำสั่งย่อย **clear**, คำสั่งย่อย **delete**, คำสั่งย่อย **delcmd**, คำสั่งย่อย **disable**, คำสั่งย่อย **enable**, คำสั่งย่อย **limitbp**, คำสั่งย่อย **printbp**, คำสั่งย่อย **status**, คำสั่งย่อย **stopi** และคำสั่งย่อย **trace** และดูที่ การตั้งค่าและการลบจุดพัก in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย stophwp

### stophwp Address Size

คำสั่งย่อย **stophwp** เช็ตการหยุด watchpoint ของฮาร์ดแวร์สำหรับขอบเขตหน่วยความจำที่ระบุ โปรแกรมหยุดเมื่อเนื้อหาของขอบเขตเปลี่ยนแปลง

#### Notes:

1. ความสำเร็จของคำสั่งย่อย **stophwp** ขึ้นกับ ฮาร์ดแวร์ คุณลักษณะนี้มีเฉพาะใน POWER630 และ POWER4 เป็นต้นไป
2. จากผลของข้อจำกัดฮาร์ดแวร์ในการที่สามารถเช็ตเฉพาะ watchpoint เดียว เหตุการณ์ watchpoint ที่แอสซีฟทำงานขัดแย้งกัน เมื่อพยายามสร้างเหตุการณ์ watchpoint ฮาร์ดแวร์ด้วย **stophwp** และ **tracehwp** ดังนั้น เหตุการณ์ก่อนหน้านี้ต้องถูกลบก่อนการสร้างเหตุการณ์ใหม่ และ เนื่องจาก watchpoint ของซอฟต์แวร์ที่แอสซีฟมีอยู่ (ที่สร้างโดยการเรียก ของคำสั่งย่อย **stop** และ **trace** ) ลดผลการทำงานที่ได้รับของ watchpoints ฮาร์ดแวร์ ชนิดของ เหตุการณ์เหล่านี้ยังทำให้มีความขัดแย้ง ซึ่งต้องถูกลบก่อนการสร้าง watchpoint ฮาร์ดแวร์

## ตัวอย่าง

1. เมื่อต้องการหยุดโปรแกรม เมื่อเนื้อหาของขอบเขตหน่วยความจำ 4 ไบต์เริ่มต้น ที่แอดเดรส 0x200004e8 เปลี่ยนแปลง ให้ป้อน:

```
stophwp 0x200004e8 4
```

## ดูที่คำสั่งย่อย tracehwp

## คำสั่งย่อย stopi

```
stopi { [Address] [ at Address | in Procedure ] [ if Condition ] }
```

คำสั่งย่อย **stopi** เช็ตจุดหยุดที่สถานที่ที่ระบุ:

- โดยการใส่แฟล็ก **if Condition** โปรแกรม จะหยุดเมื่อระบุเงื่อนไขเป็น true
- โดยการใส่พารามิเตอร์ **Address** โปรแกรมจะหยุด เมื่อเนื้อหาของ **Address** เปลี่ยนแปลง
- โดยการใส่แฟล็ก **at Address** การหยุดจะตั้ง ไว้ที่แอดเดรสที่ระบุ
- โดยการใส่แฟล็ก **in Procedure** โปรแกรมจะหยุด เมื่อ **Procedure** ถูกเรียกใช้

## แฟล็ก

ไอเท็ม	คำอธิบาย
if Condition	ระบุเงื่อนไข เช่น true
in Procedure	ระบุโปรซีเจอร์ที่จะถูกเรียก
at Address	ระบุแอดเดรสคำสั่งเครื่อง

### ตัวอย่าง

- เมื่อต้องการหยุดการประมวลผลที่แอดเดรส 0x100020f0 ให้ป้อน:  
stopi at 0x100020f0
- เมื่อต้องการหยุดการประมวลผลเมื่อเนื้อหาของแอดเดรส 0x100020f0 เปลี่ยนแปลงให้ป้อน:  
stopi 0x100020f0
- เมื่อต้องการหยุดการประมวลผลเมื่อเนื้อหาของแอดเดรส 0x100020f0 ถูกเปลี่ยนโดยเรด \$t1 ให้ป้อน:  
stopi 0x200020f0 if (\$running\_thread == 1)

ดูที่คำสั่งย่อย stop และดูที่ การดีบั๊กที่ระดับเครื่อง ด้วย dbx in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย thdata

thdata [ \$threadnumber [ all | key1 ... ] ... ] | [ all ]

คำสั่งย่อย thdata พิมพ์ข้อมูลของเรดโดยเฉพาะที่สัมพันธ์กับคีย์ต่างๆ ซึ่งสร้างโดยใช้ฟังก์ชัน pthread\_key\_create() คุณสามารถใช้คำสั่งย่อย thdata ด้วยวิธีต่อไปนี้

คำสั่ง	การดำเนินการ
thdata [ all ]	พิมพ์ข้อมูลที่ระบุเฉพาะ thread ซึ่งเชื่อมโยงกับ คีย์ทั้งหมดสำหรับ threads ที่พร้อมใช้งานทั้งหมด
thdata \$t1 [ all ]	พิมพ์ข้อมูลที่ระบุเฉพาะ thread- ซึ่งเชื่อมโยงกับคีย์ทั้งหมดสำหรับ \$t1 thread
thdata \$t1 key1 key2	พิมพ์ข้อมูลที่ระบุเฉพาะ thread- ซึ่งเชื่อมโยงกับคีย์ key1 และ key2 สำหรับ \$t1 thread
thdata \$t1 key1 key2 \$t2 key1	พิมพ์ข้อมูลที่ระบุเฉพาะ thread- ซึ่งเชื่อมโยงกับคีย์ key1 และ key2 สำหรับ \$t1 thread และข้อมูลที่ระบุเฉพาะ thread- ซึ่งเชื่อมโยงกับคีย์ key1 สำหรับ \$t2 thread

### ตัวอย่าง

- หากต้องการพิมพ์ข้อมูลที่เชื่อมโยงกับ thread ปัจจุบัน พร้อมกับคีย์ที่พร้อมใช้งานทั้งหมด ให้ป้อน:

```
(dbx) thdata $t1
Thread : 1
Key : 1 Data pointer : 0x200f7a28
Key : 2 Data pointer : 0x200f7aa8
Key : 3 Data pointer : 0x200f7ac4
(dbx)
```

- หากต้องการพิมพ์ข้อมูลที่เชื่อมโยงกับ threads จำนวนมากและคีย์จำนวนมาก ให้ป้อน:

```
(dbx) thdata $t1 2 3 $t2
Thread : 1
Key : 2 Data pointer : 0x200f7aa8
Key : 3 Data pointer : 0x200f7ac4
```

Thread : 2  
Key : 2 Data pointer : 0x200f7b24  
Key : 3 Data pointer : 0x200f7ba4  
(dbx)

โปรดดูข้อมูลที่ระบุเฉพาะ Thread ในหลักการเขียนโปรแกรมทั่วไป : การเขียนและการดีบั๊กโปรแกรม

## คำสั่งย่อย thread

### การแสดงผล Threads ที่เลือก

**thread** { [ info ] [ - ] [ ThreadNumber ... ] } | **current** | **run** | **susp** | **term** | **wait**

### เลือกแต่ละ Thread

**thread current** [ - ] ThreadNumber

### ถือค่าหรือรีลีส Threads

**thread** { **hold** | **unhold** } [ - ] [ ThreadNumber ... ]

### วิธีใช้สำหรับอ็อปชันที่แสดง

**thread** { **help** }

คำสั่งย่อย **thread** แสดงและควบคุม threads ผู้ใช้

ฟอร์มแรกของคำสั่งย่อย **thread** สามารถแสดงข้อมูลในสองรูปแบบ ถ้าคำสั่งย่อย **thread** คือ **th** ข้อมูลที่แสดงจะอยู่ในรูปแบบแรก ถ้าคำสั่งย่อย **thread** คือ **th** ข้อมูลที่แสดงจะอยู่ในรูปแบบที่สอง ถ้าไม่กำหนดพารามิเตอร์ ข้อมูลเกี่ยวกับ threads ผู้ใช้ทั้งหมดจะถูกแสดง ถ้าพารามิเตอร์ *ThreadNumber* ถูกกำหนด ข้อมูลเกี่ยวกับ threads ผู้ใช้ที่ตรงกันจะถูกแสดง เมื่อ คำสั่งย่อย **thread** แสดงเกรด บรรทัดเกรด ปัจจุบันจะนำหน้าด้วย > ถ้าเกรดที่รันอยู่ไม่เหมือนกับเกรดปัจจุบัน, บรรทัดจะถูกนำหน้าด้วย \* ข้อมูลที่แสดง โดยคำสั่งย่อย **thread** ในทั้งสองรูปแบบ ถูกอธิบายทางด้านล่าง

ข้อมูลที่แสดงโดยคำสั่งย่อย **thread** ในรูปแบบแรกเป็นดังนี้:

ไอเท็ม	คำอธิบาย
เกรด	ระบุชื่อเชิงสัญลักษณ์ของ thread ผู้ใช้ในรูปแบบ \$tThreadNumber
state-k	ระบุสถานะของคอร์เนล thread (ถ้า thread ผู้ใช้ ถูกผนวกกับคอร์เนล thread) ซึ่งมีค่าเป็น run, wait, susp หรือ term สำหรับการรัน รอหยุดทำงานชั่วคราว หรือ จบการทำงาน
wchan	ระบุเหตุการณ์ซึ่งคอร์เนล thread รอหรือพักการทำงาน ถ้า thread ผู้ใช้ ถูกผนวกกับคอร์เนล thread)
state-u	ระบุสถานะของ thread ผู้ใช้ สถานะเป็นไปได้คือ running, blocked หรือ terminated
k-tid	ระบุคอร์เนล thread identifier (ถ้า thread ผู้ใช้ ถูกผนวกกับคอร์เนล thread)
mode	ระบุโหมด (คอร์เนลหรือผู้ใช้) ซึ่ง thread ผู้ใช้ ถูกหยุด (ถ้า thread ผู้ใช้ถูกผนวกเข้ากับคอร์เนล thread)
held	ระบุว่า thread ผู้ใช้ถูกคือครอบงำหรือไม่
scope	ระบุสโคป contention ของ thread ผู้ใช้ ซึ่งเป็นได้ทั้ง sys หรือpro สำหรับ สโคป contention ระบบหรือกระบวนการ
function	ระบุชื่อของฟังก์ชัน thread ผู้ใช้

ข้อมูลที่แสดงโดยคำสั่งย่อย **thread** ใน รูปแบบที่สองถูกกำหนดทางด้านล่าง โดยตีฟอลด์ สำหรับคำสั่งย่อย **thread** subcommand **th** - ข้อมูลถูกแสดงในฟอร์ม long

ไอเท็ม	คำอธิบาย
ไธรด	ระบุชื่อเชิงสัญลักษณ์ของ thread ผู้ใช้ในรูปแบบ \$t ThreadNumber

### ข้อมูลที่เกี่ยวข้องของไธรดเคอร์เนล

ไอเท็ม	คำอธิบาย
tid	ระบุ thread identifier ผู้ใช้ (ถ้า thread ผู้ใช้ ถูกผนวกกับเคอร์เนล thread)
pri	ระบุระดับความสำคัญของเคอร์เนล thread
sched	ระบุนโยบายกำหนดการของเคอร์เนล thread ค่านี้สามารถเป็นนโยบายการกำหนดเวลาแบบ fif, oth, rr, for fifo, other หรือ round robin
state	ระบุสถานะของเคอร์เนล thread (ถ้า thread ผู้ใช้ ถูกผนวกกับเคอร์เนล thread) ค่านี้สามารถเป็น run, wait, susp หรือ zomb สำหรับการรัน รอหยุดทำงานชั่วคราว หรือ zombie

### ข้อมูลที่เกี่ยวข้องของไธรดผู้ใช้

ไอเท็ม	คำอธิบาย
tid	ระบุ thread identifier ผู้ใช้
pri	ระบุระดับความสำคัญของ user1 thread
sched	ระบุนโยบายกำหนดการของ thread ผู้ใช้ ค่านี้สามารถเป็นนโยบายการกำหนดเวลาแบบ fif, oth, rr, for fifo, other หรือ round-robin
state	ระบุสถานะของ thread ผู้ใช้ ค่านี้สามารถเป็น running, creating, suspended, blocked, runnable หรือ terminated
state	ระบุสถานะผู้ใช้เป็นเลขฐานสิบหก
flags	ระบุค่าสำหรับแฟล็ก pthread เป็นเลขฐานสิบหก
wchan	ระบุเหตุการณ์ซึ่งเคอร์เนล thread รอหรือพักการทำงาน ถ้า thread ผู้ใช้ ถูกผนวกกับเคอร์เนล thread)
mode	ระบุโหมด (เคอร์เนลหรือผู้ใช้) ซึ่ง thread ผู้ใช้ ถูกหยุด (ถ้า thread ผู้ใช้ถูกผนวกเข้ากับเคอร์เนล thread)
held	ระบุว่าไธรดผู้ใช้ถูกพัก
scope	ระบุขอบเขต contention ของไธรดผู้ใช้ ค่านี้สามารถเป็น sys หรือ pro สำหรับขอบเขต contention ระบบหรือกระบวนการ
การยกเลิก	pending ระบุว่าการยกเลิกถูกค้างอยู่หรือไม่
state	ระบุโหมดและสภาวะของการยกเลิก
	ถ้าการยกเลิก ไม่ค้างอยู่ และสภาวะและโหมดถูกเปิดใช้งานและหน่วงเวลาตามลำดับ ดังนั้นจะถูกแทนด้วย ed
	ถ้าสภาวะการยกเลิกและโหมด ถูกเปิดใช้งาน และอะซิงโครนัส จะถูกแทนด้วย ea และถ้าโหมดไม่เปิดใช้งาน จะถูกแทนด้วย d
	ถ้า การยกเลิกค้างอยู่ และสภาวะและโหมดการยกเลิก ถูกเปิดใช้งานและหน่วงเวลาตามลำดับ ดังนั้นจะถูกแทนด้วย ED ถ้าสภาวะการยกเลิกและโหมดถูกเปิดใช้งาน และอะซิงโครนัส จะถูก แทนด้วย EA และถ้าโหมดไม่เปิดใช้งาน จะถูกแทนด้วย D

ไอเท็ม	คำอธิบาย
joinable	ระบุว่าไธรดสามารถถูกรวมเข้าหรือไม่
boosted	ระบุค่า boosted ของ thread
function	ระบุชื่อของฟังก์ชัน thread ผู้ใช้
cur sig	ระบุค่าสัญญาณปัจจุบัน

ถ้าชุดอ็อพชัน \$thcomp ถูกตั้งค่า ข้อมูล จะถูกแสดงในรูปแบบบีบอัดดังแสดงในตัวอย่างต่อไปนี้

m	mode	(k)ernel (u)ser
k	k-state	(r)unning (w)aiting (s)uspended (z)ombie
u	u-state	(r)unning (R)unnable (s)uspended (t)erminated
		(b)locked (c)reating
h	held	(yes) (n)o
s	scope	(s)ystem (p)rocess
c	cancellation	not pending: (e)nabled & (d)eferred,

		(e)nabled & (a)sync, (d)isabled
	pending	: (E)nabled & (D)eferred, (E)nabled & (A)sync, (D)isabled
j	joinable	(yes) (n)o
b	boosted	value of boosted field in pthread structure
plk	kernel thread	(oth)er (fif)o (rr)-> round-robin policy
plu	user thread	(oth)er (fif)o (rr)-> round-robin policy
prk	kernel thread	hex number policy
pru	user thread	hex number policy
k-tid		kernel thread id in hex
u-tid		pthread id in hex
fl		value of flags field in pthread structure in hex
sta		value of state field in pthread structure in hex
cs		value of the current signal
wchan		event for which thread is waiting
function		function name

รูปแบบที่สองของคำสั่งย่อย **thread** ถูกใช้เพื่อเลือก thread ปัจจุบัน คำสั่งย่อย **print**, **registers** และ **where** ของ ดิบั๊กโปรแกรม **dbx** ทั้งหมดทำงานในบริบทของ thread ปัจจุบัน คำสั่งย่อย **registers** ไม่สามารถแสดงรีจิสเตอร์ ถ้า thread ปัจจุบันอยู่ในเคอร์เนลโหมด

รูปแบบที่สามของคำสั่งย่อย **thread** ถูกใช้เพื่อควบคุมการดำเนินการของ thread สามารถถือครอง Threads ไว้ได้โดยใช้แฟล็ก **hold** หรือรีลีสโดยใช้แฟล็ก **unhold** เธรดที่พักไว้จะไม่ทำต่อจนกว่าเธรดจะถูกรีลีส

**หมายเหตุ:** คำสั่งย่อย **print** ของโปรแกรมดิบั๊ก **dbx** จะรู้จักชื่อเธรดเชิงสัญลักษณ์ และสามารถใช้เพื่อแสดงสถานะของอ็อบเจกต์ที่สอดคล้อง

## แฟล็ก

<b>ไอเท็ม</b>	<b>คำอธิบาย</b>
<b>current</b>	ถ้าพารามิเตอร์ <i>ThreadNumber</i> ไม่ถูกระบุ จะแสดง thread ปัจจุบัน ถ้าพารามิเตอร์ <i>ThreadNumber</i> ถูกกำหนด, เลือก thread ผู้ใช้ที่ระบุเป็น thread ปัจจุบัน
<b>วิธีใช้</b>	แสดงข้อมูลทั้งหมดเกี่ยวกับอ็อบเจกต์ thread ที่ถูกแสดงเมื่อคำสั่ง <b>th</b> - ถูกใช้
<b>hold</b>	ถ้าไม่กำหนดพารามิเตอร์ <i>ThreadNumber</i> , ถือค่าและแสดง threads ผู้ใช้ทั้งหมด ถ้ามีการระบุพารามิเตอร์ <i>ThreadNumber</i> , ถือค่าและแสดง threads ผู้ใช้ที่ระบุ
<b>unhold</b>	ถ้าไม่กำหนดพารามิเตอร์ <i>ThreadNumber</i> , รีลีสและแสดง threads ผู้ใช้ที่ถือค่าก่อนหน้านี้ทั้งหมด ถ้ามีการระบุพารามิเตอร์ <i>ThreadNumber</i> , รีลีสและแสดง threads ผู้ใช้ที่ระบุ
<b>info</b>	ถ้าไม่กำหนดพารามิเตอร์ <i>ThreadNumber</i> , แสดงรูปแบบข้อมูลแบบยาว แสดง threads ผู้ใช้ทั้งหมด ถ้ามีการระบุพารามิเตอร์ <i>ThreadNumber</i> แสดงรูปแบบข้อมูลแบบยาว แสดง threads ผู้ใช้ที่ระบุ
	แฟล็กก่อนหน้านี้ทั้งหมดใช้อ็อบเจกต์ [-] ถ้ามีการกำหนด อ็อบเจกต์นี้ ข้อมูล thread ที่แสดงอยู่ในรูปแบบที่สอง และในรูปแบบข้อมูลแบบยาว นอกจากมีการเซตอ็อบเจกต์ <code>set \$thcomp</code>
<b>รัน</b>	แสดง threads ซึ่งอยู่ในสถานะ run
<b>susp</b>	แสดง threads ซึ่งอยู่ในสถานะ susp
<b>term</b>	แสดง threads ซึ่งอยู่ในสถานะ term
<b>wait</b>	แสดง threads ซึ่งอยู่ในสถานะ wait

## ตัวอย่าง

1. เมื่อต้องการแสดงข้อมูลเกี่ยวกับ threads ที่อยู่ในสถานะ wait ให้ป้อน:

```
thread wait
```

เอาต์พุตมีข้อมูลเหมือนดังนี้:

```
thread state-k wchan state-u k-tid mode held scope function
$t1 wait running 17381 u no pro main
$t3 wait running 8169 u no pro iothread
```

2. เมื่อต้องการแสดงข้อมูลเกี่ยวกับ threads ที่กำหนดให้ป้อน:

```
thread 1 3 4
```

เอาต์พุตมีข้อมูลเหมือนดังนี้:

```
thread state-k wchan state-u k-tid mode held scope function
$t1 wait running 17381 u no pro main
$t3 wait running 8169 u no pro iothread
>$t4 run running 9669 u no pro save_thr
```

3. เมื่อต้องการทำให้ thread 4 เป็น thread ปัจจุบัน ให้ป้อน:

```
thread current 4
```

4. เมื่อต้องการถือค่า thread หมายเลข 2 ให้ป้อน:

```
thread hold 2
```

5. เมื่อต้องการแสดงข้อมูลเกี่ยวกับ threads ที่อยู่ในสถานะรอในรูปแบบที่สองให้ป้อน:

```
thread wait -
```

เอาต์พุต คล้ายกับ:

```
thread m k u h s c j b kpl upl kpr upr k_tid u_tid fl sta wchan function
*$t1 u r w n p e d y 0 oth oth 61 1 0043e5 000001 51 004 main
$t3 u r w n p e d y 0 oth oth 61 1 001fe9 000102 51 004 iothread
>$t4 u r r n p e d y 0 oth oth 61 1 0025c5 000203 50 064 save_thr
```

6. เมื่อต้องการแสดงข้อมูลเกี่ยวกับ thread หลาย thread ที่กำหนดในรูปแบบที่สองให้ป้อน:

```
thread - 1 2 3
```

เอาต์พุต คล้ายกับ:

```
thread m k u h s c j b kpl upl kpr upr k_tid u_tid fl sta wchan function
*$t1 u r w n p e d y 0 oth oth 61 1 0043e5 000001 51 004 main
$t3 u r w n p e d y 0 oth oth 61 1 00fe9 000102 51 004 iothread
>$t4 u r r n p e d y 0 oth oth 61 1 0025c5 000203 50 064 save_thr
```

ดูที่คำสั่งย่อย **attribute**, คำสั่งย่อย **condition**, คำสั่งย่อย **mutex**, คำสั่งย่อย **print**, คำสั่งย่อย **registers**, และคำสั่งย่อย **where**

รวมทั้งดูที่ การสร้างเธรดใน *General Programming Concepts: Writing and Debugging Programs*

คำสั่งย่อย **tls**

**tls map**

คำสั่งย่อย `tl` รับเพียงหนึ่งแฟล็กเท่านั้น ที่ใช้เพื่อแสดงจุดเริ่มต้นเพิ่มเพดการกำหนดค่าเริ่มต้น TLS และความยาว สำหรับแต่ละโมดูล TLS ที่โหลด

### คำสั่งย่อย `tm_status`

`tm_status`

คำสั่งย่อย `tm_status` แสดงเนื้อหาของตัวแปร `$texasr` (ข้อยกเว้นธุรกรรม และรีจิสเตอร์สรุป) และแปลเนื้อหาเพื่อพิจารณาสาเหตุ และธรรมชาติของความล้มเหลวของธุรกรรม

### ตัวอย่าง

เมื่อต้องการแสดงและแปลค่าที่เก็บไว้ในตัวแปร `$texasr` ให้ป้อนคำสั่งต่อไปนี้:

```
(dbx) tm_status
```

เอาต์พุตคล้ายกับตัวอย่างต่อไปนี้ จะถูกแสดง:

```
REGISTER : TEXASR = 0x100000018C000001
```

Bit(s)	Field	Meaning
0-7	Failure Code	TM_SIG_DELIVERED   Failed due to signal delivery
7	Failure Persistent	Failure is transient
31	Abort	Execution of TM instruction caused Abort
32	Suspended	Failure while in Suspended State
34-35	Privilege	During Failure process-thread privilege state was 0
36	Failure	Summary Failure recording has been performed
37	TFIAR (in)exact	TFIAR is exact
38	Rollback Only Transaction	non-ROT tbegin. initiated
52-63	Transaction Level	1

### คำสั่งย่อย `tnext`

`tnext` [*Number*]

คำสั่งย่อย `tnext` รัน thread ที่รันอยู่ จนถึงบรรทัดซอร์สถัดไป พารามิเตอร์ `Number` ระบุจำนวนครั้งที่คำสั่งย่อย `tnext` รัน ถ้าไม่ระบุพารามิเตอร์ `Number`, `tnext` รันเพียงครั้งเดียว คำสั่งย่อยนี้สามารถเริ่มทำงานบนเธรดที่มีขอบเขตระบบเท่านั้น

threads ทั้งหมดถูกระหว่างการดำเนินการนี้ เมื่อต้องการจับจุดพัก ระหว่างการดำเนินการนี้ ให้เซตตัวแปร `$catchbp dbx` ถ้าตัวแปร `$catchbp` ถูกตั้งค่าและถึง จุดหยุดสำหรับเธรดอื่น คำสั่งย่อย `tnext` จะไม่ถูกทำซ้ำสำหรับจำนวนครั้งที่เหลือ

### ตัวอย่าง

1. เมื่อต้องการดำเนินการทำงานของ thread ที่รันอยู่จนถึงบรรทัดซอร์ส ถัดไป ให้ป้อน:

```
tnext
```

2. เมื่อต้องการดำเนินการทำงานของ thread ที่รันอยู่ไปถึงบรรทัดซอร์ส ที่สามต่อจากบรรทัดซอร์สปัจจุบัน ให้ป้อน:

```
tnext 3
```

ดูที่คำสั่งย่อย `tnexti` รวมทั้งดูที่ การดีบั๊กโปรแกรมที่เกี่ยวกับหลายเธรด in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย tnexti

**tnexti** [*Number*]

คำสั่งย่อย **tnexti** รัน thread ที่รันอยู่จนถึงคำสั่งถัดไป พารามิเตอร์ *Number* ระบุ จำนวนครั้งที่คำสั่งย่อย **tnexti** รัน ถ้าไม่ระบุพารามิเตอร์ *Number*, **tnexti** รันเพียงครั้งเดียว คำสั่งย่อยนี้สามารถเริ่มทำงานบนเธรดที่มีขอบเขตระบบเท่านั้น

threads ทั้งหมดถูกระหว่างการดำเนินการนี้ เมื่อต้องการจับจุดพัก ระหว่างการดำเนินการนี้ ให้ใช้ตัวแปร `$catchbp dbx` ถ้าตัวแปร `$catchbp` ถูกตั้งค่าและถึง จุดหยุดสำหรับเธรดอื่น คำสั่งย่อย **tnexti** จะไม่ถูกทำซ้ำสำหรับจำนวนครั้งที่เหลือ

### ตัวอย่าง

1. เมื่อต้องการดำเนินการทำงานของ thread ที่รันอยู่จนถึงคำสั่งเครื่อง ถัดไป ให้ป้อน:

```
tnexti
```

2. เมื่อต้องการดำเนินการทำงานของ thread ที่รันอยู่ไปถึงคำสั่งเครื่อง ที่สามต่อจากบรรทัดซอร์สปัจจุบัน ให้ป้อน:

```
tnexti 3
```

ดูที่คำสั่งย่อย **tnext** รวมทั้งดูที่ การดีบั๊กโปรแกรมที่เกี่ยวกับหลายเธรด in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย trace

**trace** [*SourceLine* | *Expression at SourceLine* | *Procedure* | [*Variable*] [*at SourceLine* | *in Procedure*] | **on load** *ModuleName*] [*if Condition*]

คำสั่งย่อย **trace** พิมพ์ข้อมูลการติดตามสำหรับ โปรซีเจอร์ ฟังก์ชัน บรรทัดซอร์ส นิพจน์ หรือตัวแปร ที่ระบุเมื่อโปรแกรมรัน ตัวแปร *SourceLine* สามารถถูกระบุ เป็นจำนวนเต็มหรือสตริงชื่อไฟล์ที่ตามด้วย : (โคลอน) และจำนวนเต็ม สามารถระบุเงื่อนไขได้ ดีบั๊กโปรแกรม **dbx** เชื่อมโยงตัวเลขกับแต่ละคำสั่งย่อย **trace** ใช้คำสั่งย่อย **status** เพื่อดูหมายเลขเหล่านี้ ใช้คำสั่งย่อย **delete** เพื่อปิดการติดตาม คุณสามารถเปิดและปิดการติดตามโดยใช้คำสั่งย่อย **enable** และ **disable** ตามลำดับ

คำสั่งย่อย **trace** สามารถแสดงข้อมูลการ ติดตาม เมื่อโมดูลถูกโหลดหรือยกเลิกการโหลดโดยกระบวนการที่ดีบั๊ก พารามิเตอร์ *ModuleName* เป็นทางเลือกที่สามารถถูกระบุ เป็นชื่อโมดูลเดียว หรือเป็นชื่อโมดูลที่คู่กับชื่อสมาชิก ในรูปแบบ:

```
ModuleName(MemberName)
```

หากใช้ แฟล็ก **on load** โดยไม่มีพารามิเตอร์ *ModuleName* คำสั่ง **dbx** จะติดตามการโหลดและการเลิกโหลดของโมดูลทั้งหมด

โดยดีฟอลต์ การติดตาม อิงตามกระบวนการ เมื่อต้องการทำให้ เป็นการติดตามแบบอิงเธรด ให้ระบุเธรดในเงื่อนไข (ดังในตัวอย่างที่ 8)

### แฟล็ก

## ไอเท็ม

at SourceLine  
if Condition  
in Procedure  
on load ModuleName

## คำอธิบาย

ระบุบรรทัดซอร์ส โดยที่ระบุบรรทัดซอร์สซึ่งพบนิพจน์ที่ถูกติดตาม  
ระบุเงื่อนไขสำหรับจุดเริ่มต้นของการติดตาม การติดตาม เริ่มต้นเฉพาะ if Condition เป็น true  
ระบุโปรซีเจอร์ที่จะถูกใช้เพื่อค้นหาโปรซีเจอร์ที่จะใช้เพื่อค้นหาโปรซีเจอร์หรือตัวแปร ที่ถูกติดตาม  
ระบุโหนดโมดูลที่จะถูกมอนิเตอร์

## ตัวอย่าง

1. เมื่อต้องการติดตามแต่ละการเรียกใช้ไปยังโปรซีเจอร์ printf ให้ป้อน:

```
trace printf
```

2. เมื่อต้องการติดตามการประมวลผลบรรทัดที่ 22 ในไฟล์ hello.c ให้ป้อน:

```
trace "hello.c":22
```

3. เมื่อต้องการติดตามการเปลี่ยนแปลงกับตัวแปร x ภายในโปรซีเจอร์ main ให้ป้อน:

```
trace x in main
```

4. เมื่อต้องการติดตามแอดเดรสข้อมูล 0x2004000 ให้ป้อน:

```
set $A=0x2004000
```

```
trace $A
```

**หมายเหตุ:** คำสั่งย่อย tracei ถูกออกแบบเพื่อติดตามแอดเดรส

5. คุณสามารถจำกัดการพิมพ์บรรทัดซอร์สเป็นเมื่อ Procedure ที่ระบุ แอ็คทีฟ คุณยังสามารถระบุ Condition ทางเลือกเพื่อควบคุมว่าเมื่อใดที่ข้อมูลการติดตามต้องถูกสร้างขึ้น ตัวอย่างเช่น:

```
(dbx) trace in sub2
```

```
[1] trace in sub2
```

```
(dbx) run
```

```
trace in hellosub.c: 8 printf("%s",s);
```

```
trace in hellosub.c: 9 i = '5';
```

```
trace in hellosub.c: 10 }
```

6. คุณสามารถแสดงข้อความในแต่ละครั้งที่โปรซีเจอร์ถูกเรียกหรือส่งคืน เมื่อโปรซีเจอร์ถูกเรียก ข้อมูลรวมพารามิเตอร์ที่ส่งผ่าน และชื่อของรูทีน ที่เรียก เมื่อส่งคืน, ข้อมูลมีค่าส่งคืนจาก Procedure ตัวอย่างเช่น:

```
(dbx) trace sub
```

```
[1] trace sub
```

```
(dbx) run
```

```
calling sub(s = "hello", a = -1, k = delete) from function main
```

```
returning "hello" from sub
```

7. คุณสามารถพิมพ์ค่าของ Expression เมื่อ โปรแกรมทำงานถึงบรรทัดซอร์สที่ระบุ หมายเลขบรรทัดและไฟล์จะถูกพิมพ์ แต่บรรทัดซอร์สไม่ถูกพิมพ์ ตัวอย่างเช่น:

```
(dbx) trace x*17 at "hellosub.c":8 if (x > 0)
```

```
[1] trace x*17 at "hellosub.c":8 if x > 0
```

```
(dbx) run
```

```
at line 8 in file "hellosub.c": x*17 = 51
```

```
(dbx) trace x
```

```
[1] trace x
```

```
initially (at line 4 in "hello.c"): x = 0
```

```
after line 17 in "hello.c": x = 3
```

8. เมื่อต้องการติดตามการเปลี่ยนแปลงกับตัวแปร x ที่ทำโดยเรด \$t1 ให้ป้อน คำสั่งต่อไปนี้:

```
(dbx) trace x if ($running_thread == 1)
```

- เมื่อต้องการติดตามการโหลดหรือการเลิกโหลดโมดูลทั้งหมด ให้ป้อน คำสั่งต่อไปนี้:

```
trace on load
```

- เมื่อต้องการติดตามการโหลดหรือการเลิกโหลดของโมดูล Module ให้ป้อนคำสั่งต่อไปนี้:

```
trace on load "Module"
```

- เมื่อต้องการติดตามการโหลดหรือการเลิกโหลดของสมาชิก Member ใน โมดูล Module ให้ป้อนคำสั่งต่อไปนี้:

```
trace on load "Module(Member)"
```

และดูที่คำสั่งย่อย **tracei**

คำสั่งย่อย **tracehwp**

**tracehwp** *Address Size*

คำสั่งย่อย **tracehwp** ใช้การหยุด watchpoint ของฮาร์ดแวร์สำหรับขอบเขตหน่วยความจำที่ระบุ ดับเบิลโปรแกรม dbx พิมพ์ข้อมูลการติดตาม เมื่อเนื้อหาของขอบเขต เปลี่ยนแปลง

**Notes:**

- ความสำเร็จของคำสั่งย่อย **tracehwp** ขึ้นกับ ฮาร์ดแวร์ คุณลักษณะนี้มีเฉพาะใน POWER630 และ POWER4 เป็นต้นไป
- จากผลของข้อจำกัดฮาร์ดแวร์ในการที่สามารถเซตเฉพาะ watchpoint เดียว เหตุการณ์ watchpoint ที่แอสซิมป์ทำงานขัดแย้งกัน เมื่อพยายามสร้างเหตุการณ์ watchpoint ฮาร์ดแวร์ด้วย **stophwp** และ **tracehwp** ดังนั้น เหตุการณ์ก่อนหน้านั้นต้องถูกลบก่อนการสร้างเหตุการณ์ใหม่ และ เนื่องจาก watchpoint ของซอฟต์แวร์ที่แอสซิมป์มีอยู่ (ที่สร้างโดยการเรียก ของคำสั่งย่อย **stop** และ **trace**) ลดผลการทำงานที่ได้รับของ watchpoints ฮาร์ดแวร์ ชนิดของ เหตุการณ์เหล่านี้ยังทำให้มีความขัดแย้ง ซึ่งต้องถูกลบก่อนการสร้าง watchpoint ฮาร์ดแวร์

**ตัวอย่าง**

- เมื่อต้องการติดตามแต่ละครั้งที่เนื้อหาของขอบเขตหน่วยความจำขนาด 4 ไบต์เริ่มต้น ที่แอดเดรส 0x200004e8 เปลี่ยนแปลง ให้ป้อนคำสั่ง ต่อไปนี้:

```
tracehwp 0x200004e8 4
```

ดูที่คำสั่งย่อย **stophwp**

คำสั่งย่อย **tracei**

```
tracei [ [ Address ] [ at Address | in Procedure ] | Expression at Address ] [ if Condition ]
```

คำสั่งย่อย **tracei** เปิดการติดตามเมื่อ:

- เนื้อหาของแอดเดรสที่ระบุโดยพารามิเตอร์ *Address* จะเปลี่ยนแปลง หากมีการรวมแฟล็ก *Address*
- คำสั่งที่ *Address* จะรันหากระบุพารามิเตอร์ **at** *Address*
- โพสิเตอร์ที่ระบุโดย *Procedure* จะแอสซิมป์ หากมีการรวมแฟล็ก **in** *Procedure*
- เงื่อนไขที่ระบุโดยพารามิเตอร์ *Condition* จะเป็น true หากมีการรวมแฟล็ก **if** *Condition*

**แฟล็ก**

ไอเท็ม  
at Address  
if Condition  
in Procedure

คำอธิบาย  
ระบุแอดเดรสการติดตามถูกเปิดใช้เมื่อคำสั่งที่แอดเดรสนี้ถูกรัน  
ระบุเงื่อนไข การติดตามถูกเปิดใช้เมื่อพบเงื่อนไขที่ระบุ  
ระบุโปรซีเจอร์ การติดตามถูกเปิดใช้เมื่อโปรซีเจอร์นี้ แอ็คทีฟ

## ตัวอย่าง

- เมื่อต้องการติดตามแต่ละคำสั่งที่ถูกประมวลผล ให้ป้อนคำสั่งต่อไปนี้:  

```
tracei
```
- เมื่อต้องการติดตามแต่ละครั้งที่คำสั่งที่แอดเดรส 0x100020f0 ถูกประมวลผล ให้ป้อนคำสั่งต่อไปนี้:  

```
tracei at 0x100020f0
```
- เมื่อต้องการแต่ละครั้งที่เนื้อหาของตำแหน่งหน่วยความจำ 0x20004020 เปลี่ยนแปลง ขณะที่โปรซีเจอร์ main แอ็คทีฟ ให้ป้อนคำสั่งต่อไปนี้:  

```
tracei 0x20004020 in main
```
- เมื่อต้องการติดตามแต่ละครั้งที่คำสั่งที่แอดเดรส 0x100020f0 ถูกประมวลผล โดยเรด \$t4 ให้ป้อนคำสั่งต่อไปนี้:  

```
tracei at 0x100020f0 if ($running_thread == 4)
```

ดูที่คำสั่งย่อย **trace** นอกจากนี้โปรดดูที่ การดีบักระดับเครื่องโดยใช้ **dbx** in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย tskip

**tskip** [Number]

คำสั่งย่อย **tskip** ดำเนินการทำงาน ของ thread ที่รันอยู่จากจุดหยุดทำงานปัจจุบัน จำนวนจุดหยุดระดับเรดที่ระบุโดยพารามิเตอร์ **Number** ถูกข้ามสำหรับเรดที่กำลังรัน คำสั่งย่อยนี้สามารถเริ่มทำงาน บนเรดที่มีขอบเขตระบบเท่านั้น

threads อื่นทั้งหมดรันระหว่างการดำเนินการนี้ และ breakpoints และ watchpoints ทั้งหมดที่ระบุโดยผู้ใช้จะถูกจับข้อมูล การกระทำสามารถหยุดได้เมื่อมีเรดเข้าสู่จุดหยุดหรือ watchpoint แม้ว่าการเรียกใช้งานที่เริ่มต้นโดยคำสั่งย่อย **tskip** สามารถหยุดทำงานเนื่องจากเหตุการณ์ของเรดอื่น แต่จำนวน **tskip** ที่ระบุสำหรับเรดก่อนหน้ายังคงแอ็คทีฟ และจำนวนจุดหยุดระดับเรดที่ระบุโดยจำนวน **tskip** ถูกข้ามสำหรับเรดนั้นเมื่อดำเนินกระบวนการต่อ เมื่อเรด สิ้นสุดจำนวน **tskip** ที่สัมพันธ์กับเรด จะถูกลบออก

ใช้คำสั่งย่อย **status** เพื่อดูจำนวน **tskip** ที่เหลือ สำหรับ threads ใช้คำสั่งย่อย **delete** เพื่อลบจำนวน **tskip** ที่เหลือ สำหรับ threads

## ตัวอย่าง

เมื่อต้องการดำเนินการทำงานต่อจนถึง breakpoint ระดับ thread ที่สอง เริ่มต้นจากจุดหยุดปัจจุบันสำหรับ thread ที่รันอยู่ ให้ป้อน:

```
tskip 1
```

ดูที่คำสั่งย่อย **cont** รวมทั้งดูที่ การดีบั๊กโปรแกรมที่เกี่ยวกับหลายเรด in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย tstep

70 AIX เวอร์ชัน 7.2: ข้อมูลอ้างอิงคำสั่งวอลุ่ม 2, d - h

## **tstep** [*Number*]

คำสั่งย่อย **tstep** รันจำนวนบรรทัดซอร์สที่ระบุจากบรรทัดซอร์สปัจจุบันสำหรับ thread ที่รันอยู่ พารามิเตอร์ *Number* ระบุจำนวนครั้งที่คำสั่งย่อย **tstep** รัน ถ้าไม่ระบุพารามิเตอร์ *Number*, **tstep** รันเพียงครั้งเดียว คำสั่งย่อยนี้สามารถเริ่มทำงานบนเธรดที่มีขอบเขตระบบเท่านั้น

threads ทั้งหมดถูกระงับระหว่างการดำเนินการนี้ ถ้า `$hold_next` ถูกตั้งค่า เธรดทั้งหมดยกเว้นเธรดที่กำลังรันจะถูกหยุดพัก

**หมายเหตุ:** ใช้ตัวแปร `$stepignore` ของคำสั่งย่อย `set` เพื่อควบคุมการทำงานของคำสั่งย่อย **tstep** ตัวแปร `$stepignore` เปิดใช้คำสั่งย่อย **tstep** เพื่อข้ามรูทีนขนาดใหญ่ซึ่งไม่มีข้อมูลดีบั๊ก

### **ตัวอย่าง**

1. เมื่อต้องการดำเนินการทำงานของ thread ที่รันอยู่ อีกหนึ่งบรรทัดซอร์ส ถัดไป ให้ป้อน:  
`tstep`
2. เมื่อต้องการดำเนินการทำงานของ thread ที่รันอยู่ อีกห้าบรรทัดซอร์ส ถัดไป ให้ป้อน:  
`tstep 5`
3. เมื่อต้องการป้องกันโปรแกรม `dbx` ไม่ให้ทำงานฟังก์ชัน `printf` ที่ละคำสั่ง ตามที่แสดงในโค้ดตัวอย่าง:  
`60 printf ("hello world /n");`

ให้ป้อน:

```
set $stepignore="function"; step
```

ดูที่คำสั่งย่อย `cont`, คำสั่งย่อย `goto`, คำสั่งย่อย `tnext`, คำสั่งย่อย `set` และคำสั่งย่อย `tstepi` รวมทั้งดูที่ การดีบั๊กโปรแกรมที่เกี่ยวข้องกับหลายเธรด in *General Programming Concepts: Writing and Debugging Programs*

## **คำสั่งย่อย tstepi**

### **tstepi** [*Number*]

คำสั่งย่อย **tstepi** รันคำสั่ง ตามจำนวนที่ระบุจากคำสั่งปัจจุบัน สำหรับ thread ที่รันอยู่ พารามิเตอร์ *Number* ระบุ จำนวนครั้งที่คำสั่งย่อย **tstepi** รัน ถ้าไม่ระบุพารามิเตอร์ *Number*, **tstepi** รันเพียงครั้งเดียว คำสั่งย่อยนี้สามารถเริ่มทำงานบนเธรดที่มีขอบเขตระบบเท่านั้น

threads ทั้งหมดถูกระงับระหว่างการดำเนินการนี้ ถ้า `$hold_next` ถูกตั้งค่า เธรดทั้งหมดยกเว้นเธรดที่กำลังรันจะถูกหยุดพัก

### **ตัวอย่าง**

1. เมื่อต้องการดำเนินการทำงานของ thread ที่รันอยู่ อีกหนึ่งคำสั่งเครื่อง ให้ป้อน:  
`tstepi`
2. เมื่อต้องการดำเนินการทำงานของ thread ที่รันอยู่ อีกห้าคำสั่งเครื่อง ให้ป้อน:  
`tstepi 5`

ดูที่คำสั่งย่อย `gotoi`, คำสั่งย่อย `tnexti` และ คำสั่งย่อย `tstep` รวมทั้งดูที่ การดีบั๊กโปรแกรมที่เกี่ยวข้องกับหลายเธรด in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย tstop

```
tstop { in Procedure | [Variable] at SourceLine [ if Condition ] } [for $tthreadnumber]
```

คำสั่งย่อย **tstop** ใช้สำหรับการหยุดของจุดพักระดับซอร์ส สำหรับ thread และหยุดแอ็พพลิเคชันโปรแกรมเมื่อ thread ที่ระบุทำงานถึงจุดพัก เรดที่ระบุต้องมีอยู่ในเวลาเดียวกับการสร้างเหตุการณ์ เรดปัจจุบันถูกใช้ถ้าไม่มีเรดถูกระบุ thread ที่ระบุ ถูกหยุดเมื่อเงื่อนไขต่อไปนี้จะเกิดขึ้น:

- มีการใช้แฟล็ก *if Condition* และ *Condition* เป็น true
- มีการใช้แฟล็ก *in Procedure* และ *Procedure* ถูกเรียก
- มีการใช้แฟล็ก *at SourceLine* และทำงานถึงหมายเลขบรรทัด *SourceLine* ตัวแปร *SourceLine* สามารถถูกระบุ เป็นจำนวนเต็มหรือสตริงชื่อไฟล์ที่ตามด้วยโคลอน (:) และจำนวนเต็ม

จุดพักระดับ Thread สามารถถูกใช้บน threads สโคประบบเท่านั้น เมื่อจุดหยุดระดับเรด และระดับกระบวนการถูกเข้าถึงในเวลาเดียวกัน จุดหยุดทั้งสองถูกประมวลผล และจุดหยุดระดับ เรดถูกพอร์ต เมื่อเรดจบการทำงาน เหตุการณ์ที่สัมพันธ์กับเรดจะถูกกลับ

## แฟล็ก

ไอเท็ม	คำอธิบาย
<i>at SourceLine</i>	ระบุหมายเลขบรรทัด
<i>for \$tthreadnumber</i>	ระบุหมายเลข thread
<i>if Condition</i>	ระบุเงื่อนไข (ตัวอย่าง true)
<i>in Procedure</i>	ระบุโปรซีเจอร์ที่จะถูกเรียก

## ตัวอย่าง

1. เมื่อต้องการหยุดการทำงานที่คำสั่งแรกในโปรซีเจอร์ **func** ขณะรัน thread 2 ให้ป้อน:  
tstop in func for \$t2
2. เมื่อต้องการหยุดการทำงานของ thread ปัจจุบันเมื่อค่าของ ตัวแปร **x** ถูกเปลี่ยนบนบรรทัด 12 ของการทำงาน ให้ป้อน:  
tstop x at 12

ดูที่คำสั่งย่อย **ttrace** รวมทั้งดูที่ การดีบั๊กโปรแกรมที่เกี่ยวข้องกับหลายเรด in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย tstophwp

```
tstophwp address size [for $tthreadnumber]
```

คำสั่งย่อย **tstophwp** ใช้สำหรับการหยุดของ watchpoint ฮาร์ดแวร์ระดับ thread สำหรับขอบเขตหน่วยความจำที่ระบุ โปรแกรมหยุดเมื่อเนื้อหาของขอบเขตเปลี่ยนแปลง ขณะรัน thread ที่ระบุ เรดที่ระบุต้องมีอยู่ในเวลาเดียวกับการสร้างเหตุการณ์ เรดปัจจุบันถูกใช้ถ้าไม่มีเรดถูกระบุ เหตุการณ์ watchpoint ระดับ thread สามารถถูกใช้ เฉพาะ threads สโคประบบเท่านั้น เมื่อเรดจบการทำงาน เหตุการณ์ที่สัมพันธ์กับเรดจะถูกกลับ

## Notes:

1. ความสำเร็จของคำสั่งย่อย `tstophwp` ขึ้นอยู่กับฮาร์ดแวร์ คุณลักษณะนี้มีเฉพาะใน POWER630 และ POWER4 เป็นต้นไป
2. จากผลของข้อจำกัดทางฮาร์ดแวร์ อนุญาตเฉพาะให้เซตได้เพียง watchpoint เดียว, เหตุการณ์ thread watchpoint ที่แอดเดรสที่ทำงาน ชัดแย้งกัน เมื่อพยายามสร้างเหตุการณ์ watchpoint ฮาร์ดแวร์อื่น สำหรับ thread เดียวกันโดยใช้ `tstophwp` และ `ttracelhwp` เพื่อหลีกเลี่ยงปัญหานี้ เหตุการณ์ก่อนหน้าต้องถูกลบก่อนการสร้าง เหตุการณ์ใหม่ เนื่องจาก watchpoint ของซอฟต์แวร์ที่แอดเดรสที่มีอยู่ (ที่สร้างโดยการเรียก ของคำสั่งย่อย `stop` และ `trace`) สามารถลดผลการทำงานที่ได้รับของ watchpoints ฮาร์ดแวร์ ชนิดเหตุการณ์ เหล่านี้ยังต้องถูกลบ ก่อนทำการสร้าง watchpoint ฮาร์ดแวร์ใหม่ เพื่อหลีกเลี่ยงความขัดแย้ง
3. เมื่อ process-level watchpoint มีอยู่ เรดไม่มี thread-level watchpoint ฝ้าดูตำแหน่ง watchpoint ของกระบวนการ ถ้าเรดมี จุดฝ้าดูระดับเรด เรดจะฝ้าดูตำแหน่งจุดฝ้าดู เรด
4. watchpoint ฮาร์ดแวร์ระดับ Thread และ watchpoint ฮาร์ดแวร์ระดับกระบวนการ มีอยู่ร่วมกันได้และไม่ขัดแย้งกัน
5. ถ้าจุดฝ้าดูระดับกระบวนการและระดับเรดมีอยู่สำหรับ แอดเดรสเดียวกัน เหตุการณ์จุดฝ้าดูระดับกระบวนการจะถูก รายงาน

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>for \$t threadnumber</code>	ระบุหมายเลข thread

## ตัวอย่าง

เมื่อต้องการหยุดโปรแกรม เมื่อ thread 2 รันอยู่และเนื้อหาของ ขอบเขตหน่วยความจำ 4-ไบต์ เริ่มต้นที่แอดเดรส 0x200004e8 เปลี่ยนแปลง ให้ป้อน:

```
tstophwp 0x200004e8 4 for $t2
```

ดูที่คำสั่งย่อย `ttracelhwp` รวมทั้งดูที่ การดีบักโปรแกรมที่เกี่ยวกับหลายเรด in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย tstopi

```
tstopi { in Procedure | [Address] at Address [ if Condition ] } [for $tthreadnumber]
```

คำสั่งย่อย `tstopi` ตั้งค่าการหยุดจุดหยุดระดับคำสั่งเครื่องสำหรับเรด เรด ที่ระบุต้องมีอยู่ในเวลาเดียวกับการสร้างเหตุการณ์ เรดปัจจุบันถูกใช้ถ้าไม่มีเรดถูกระบุ thread ที่ระบุ ถูกหยุดเมื่อเงื่อนไขต่อไปนี้จะเกิดขึ้น:

- มีการใช้แฟล็ก `if Condition` และ `Condition` เป็น true
- มีการใช้แฟล็ก `in Procedure` และ `Procedure` ถูกเรียก
- มีการใช้แฟล็ก `at Address` และทำงานถึง `Address`

จุดพักระดับ Thread สามารถถูกเซตบน threads สโคประบบเท่านั้น เมื่อมีการเข้าสู่จุดหยุด thread-level และ process-level พรอมกัน ทั้งสองจุดหยุดจะถูกดำเนินการและจุดหยุด thread-level ถูกรายงาน เมื่อเรดจบการทำงาน เหตุการณ์ที่สัมพันธ์กับเรดจะถูกลบ

## แฟล็ก

ไอเท็ม	คำอธิบาย
at Address	ระบุแอดเดรสคำสั่งเครื่อง
for \$t threadnumber	ระบุหมายเลข thread
if Condition	ระบุเงื่อนไข
in Procedure	ระบุโปรซีเจอร์ที่จะถูกเรียก

## ตัวอย่าง

- เมื่อต้องการหยุดการทำงานที่แอดเดรส 0x100020f0 ขณะรัน thread 2 ให้ป้อน:  
tstopi at 0x100020f0 for \$t2
- เมื่อต้องการหยุดการทำงานเมื่อเข้าสู่โปรซีเจอร์ func ขณะรัน thread ปัจจุบันให้ป้อน:  
tstopi in func

ดูที่คำสั่งย่อย ttracei รวมทั้งดูที่ การดีบั๊กโปรแกรมที่เกี่ยวกับหลายเธรด in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย ttrace

```
ttrace { [Variable] at SourceLine | Procedure } [ if Condition ] [for $tthreadnumber]
```

คำสั่งย่อย ttrace พิมพ์ข้อมูลการติดตาม เมื่อ thread ที่ระบุสำหรับ โปรซีเจอร์ ฟังก์ชัน บรรทัดซอร์ส และตัวแปร ที่ระบุ ตัวแปร SourceLine สามารถถูกระบุ เป็นจำนวนเต็มหรือสตริงชื่อไฟล์ที่ตามด้วยโคลอน (:) และจำนวนเต็ม ดีบั๊กโปรแกรม dbx เชื่อมโยงตัวเลขกับแต่ละคำสั่งย่อย ttrace ใช้คำสั่งย่อย status เพื่อดูหมายเลขเหล่านี้ ใช้คำสั่งย่อย delete เพื่อปิดการติดตาม คุณสามารถเปิดและปิดการติดตามโดยใช้คำสั่งย่อย enable และ disable ตามลำดับ

เธรดปัจจุบันถูกใช้ถ้าไม่มีเธรดถูกระบุ การติดตามระดับ Thread สามารถถูกเซตสำหรับ thread สโคประบบเท่านั้น เธรด ที่ระบุ ต้องมีอยู่ในเวลาเดียวกับการสร้างเหตุการณ์ เมื่อเธรดสิ้นสุด เหตุการณ์ที่เกี่ยวข้องกับเธรดจะถูกปล่อย

## แฟล็ก

ไอเท็ม	คำอธิบาย
at SourceLine	ระบุบรรทัดซอร์ส โดยที่ระบุบรรทัดซอร์สซึ่งพบนิพจน์ที่ถูก ติดตาม
for \$t threadnumber	ระบุหมายเลข thread
if Condition	ระบุเงื่อนไขสำหรับจุดเริ่มต้นของการติดตาม การติดตาม เริ่มต้นเฉพาะถ้า Condition เป็น true
in Procedure	ระบุโปรซีเจอร์ที่ใช้เพื่อค้นหาโปรซีเจอร์ที่จะใช้เพื่อค้นหาโปรซีเจอร์หรือตัวแปร ที่ถูกติดตาม

## ตัวอย่าง

- เมื่อต้องการติดตามแต่ละการเรียกไปที่โปรซีเจอร์ printf ขณะรัน thread 2 ให้ป้อน:  
ttrace printf for \$t2
- เมื่อต้องการติดตามการประมวลผลบรรทัดที่ 22 ในไฟล์ hello.c/ ขณะที่เธรดปัจจุบันกำลังรัน ให้ป้อน:  
ttrace "hello.c":22

ดูที่คำสั่งย่อย ttracei รวมทั้งดูที่ การดีบั๊กโปรแกรมที่เกี่ยวกับหลายเธรด in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย ttracei

**ttracei** [Address] at Address [ if Condition ] } [for \$tthreadnumber]

คำสั่งย่อย **ttracei** เปิดการติดตามสำหรับ thread ที่ระบุเมื่อเงื่อนไขต่อไปนี้จะเกิดขึ้น:

- มีการใช้แฟล็ก **if Condition** และ **Condition** เป็น true
- มีการระบุแฟล็ก **at Address** และคำสั่ง **at Address** อยู่นั้น

เรดปัจจุบันถูกใช้ถ้าไม่มีเรดถูกระบุ การติดตามระดับ Thread สามารถถูกใช้สำหรับ thread สโคประบบเท่านั้น เรด ที่ระบุ ต้องมีอยู่ในเวลาเดียวกับที่ทำการสร้างเหตุการณ์ เมื่อ เรดสิ้นสุด เหตุการณ์ที่เกี่ยวข้องกับเรดจะถูกปล่อย

## แฟล็ก

ไอเท็ม	คำอธิบาย
at Address	ระบุแอดเดรส การติดตามถูกเปิดใช้เมื่อคำสั่ง ที่แอดเดรสนั้นถูกรัน
for \$t threadnumber	ระบุหมายเลข thread
if Condition	ระบุเงื่อนไข การติดตามถูกเปิดใช้เมื่อพบเงื่อนไข ที่ระบุ

## ตัวอย่าง

1. เมื่อต้องการติดตาม ทุกครั้งที่คำสั่งที่แอดเดรส 0x100020f0 ถูก เรียกใช้ ขณะที่ thread 3 รันอยู่ ให้ป้อน:

```
ttracei at 0x100020f0 for $t3
```

2. เมื่อต้องการติดตามทุกครั้งที่คำสั่งที่แอดเดรส 0x100020f0 ถูก เรียกใช้โดย thread ปัจจุบัน ให้ป้อน:

```
ttracei at 0x100020f0
```

ดูที่คำสั่งย่อย **ttrace** รวมทั้งดูที่ การดีบั๊กโปรแกรมที่เกี่ยวกับหลายเรด in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย **ttracehwp**

**ttracehwp** address size [for \$tthreadnumber]

คำสั่งย่อย **ttracehwp** ใช้การติดตามของ watchpoint ฮาร์ดแวร์ระดับ thread สำหรับขอบเขตหน่วยความจำที่ระบุ ดีบั๊ก โปรแกรม **dbx** พิมพ์ข้อมูลการติดตาม เมื่อเนื้อหาของขอบเขต เปลี่ยนแปลงขณะรัน thread ที่ระบุ เรด ที่ระบุต้องมีอยู่ในเวลา เดียวกับการสร้างเหตุการณ์ เรดปัจจุบันถูกใช้ถ้าไม่มีเรดถูกระบุ เหตุการณ์ watchpoint ระดับ thread สามารถถูกใช้ เฉพาะ threads สโคประบบเท่านั้น เมื่อเรดจบการทำงาน เหตุการณ์ที่สัมพันธ์กับเรดจะถูกปล่อย

## หมายเหตุ:

1. ความสำเร็จของคำสั่งย่อย **ttracehwp** ขึ้นกับฮาร์ดแวร์ คุณลักษณะนี้มีเฉพาะใน POWER630 และ POWER4 เป็นต้นไป
2. จากผลของข้อจำกัดทางฮาร์ดแวร์ อนุญาตเฉพาะให้ใช้ได้เพียง watchpoint เดียว, เหตุการณ์ thread watchpoint ที่แอดเดรส ที่ทำงาน ขัดแย้งกัน เมื่อพยายามสร้างเหตุการณ์ watchpoint ฮาร์ดแวร์อื่น สำหรับ thread เดียวกันโดยใช้ **tstophwp** และ **ttracehwp** เพื่อหลีกเลี่ยงปัญหานี้ เหตุการณ์ก่อนหน้าต้องถูกลบก่อนการสร้าง เหตุการณ์ใหม่ เนื่องจาก watchpoint ของซอฟต์แวร์ที่แอดเดรสที่มีอยู่ (ที่สร้างโดยการเรียก ของคำสั่งย่อย **stop** และ **trace**) สามารถลดผลการทำงานที่ได้รับของ watchpoints ฮาร์ดแวร์ ชนิดเหตุการณ์ เหล่านี้ยังต้องถูกลบ ก่อนทำการสร้าง watchpoint ฮาร์ดแวร์ใหม่ เพื่อหลีกเลี่ยง ความขัดแย้ง
3. เมื่อ process-level watchpoint มีอยู่ เรดไม่มี thread-level watchpoint ฝ้าดูตำแหน่ง watchpoint ของกระบวนการ ถ้า ธรดมี จุดฝ้าดูระดับเรด เรดจะฝ้าดูตำแหน่งจุดฝ้าดู เรด

4. watchpoint ฮาร์ดแวร์ระดับ Thread และ watchpoint ฮาร์ดแวร์ระดับกระบวนการ มีอยู่ร่วมกันได้และไม่ขัดแย้งกัน
5. ถ้าจุดเฝ้าดูระดับกระบวนการและระดับเธรดมีอยู่สำหรับ แอดเดรสเดียวกัน เหตุการณ์จุดเฝ้าดูระดับกระบวนการจะถูก รายงาน

## แฟล็ก

ไอเท็ม คำอธิบาย  
for \$t threadnumber ระบุหมายเลข thread

## ตัวอย่าง

เมื่อต้องการติดตามแต่ละครั้งที่เนื้อหาของขอบเขตหน่วยความจำ 4-ไบต์เริ่มต้น ที่แอดเดรส 0x200004e8 เปลี่ยนแปลงขณะรัน thread 2 ให้ป้อน:

```
ttracehwp 0x200004e8 4 for $t2
```

ดูที่คำสั่งย่อย **tstophwp** รวมทั้งดูที่ การดีบั๊กโปรแกรมที่เกี่ยวข้องกับหลายเธรด in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย unalias

**unalias** Name

คำสั่งย่อย **unalias** ลบ alias ที่ระบุโดยพารามิเตอร์ Name

## ตัวอย่าง

เมื่อต้องการลบ alias ชื่อ printx ให้ป้อน:

```
unalias printx
```

ดูที่คำสั่งย่อย **alias** และดูที่ การสร้างคำสั่งย่อย Aliases in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย unset

**unset** Name

คำสั่งย่อย **unset** ลบตัวแปรดีบั๊กโปรแกรม dbx ที่เชื่อมโยงกับ ชื่อที่ระบุโดยพารามิเตอร์ Name

## ตัวอย่าง

เมื่อต้องการลบการยับยั้งตัวแปรที่แสดง รีจิสเตอร์อิงดซัน ให้ป้อน:

```
unset $noflregs
```

ดูที่คำสั่งย่อย **set** และดูที่ การเปลี่ยนเอาต์พุตการพิมพ์ ด้วยตัวแปรการดีบั๊กพิเศษ in *General Programming Concepts: Writing and Debugging Programs*

## คำสั่งย่อย up

`up [ Count ]`

คำสั่งย่อย `up` ย้าย ฟังก์ชันปัจจุบันขึ้นเป็น จำนวนระดับ `Count` ของสแต็ก ฟังก์ชันปัจจุบันถูกใช้เพื่อกำหนดชื่อ ดีฟอลต์ สำหรับ พารามิเตอร์ `Count` คือหนึ่ง

ตัวอย่าง

1. เมื่อต้องการลบฟังก์ชันปัจจุบันขึ้นไปบนสแต็ก 2 ระดับ ให้ป้อน:

`up 2`

2. เมื่อต้องการแสดงฟังก์ชันปัจจุบันบนสแต็ก ให้ป้อน:

`up 0`

ดูที่คำสั่งย่อย `down` รวมทั้ง ดูที่การเปลี่ยนแปลงไฟล์หรือโพธิ์เตอร์ ปัจจุบัน การแสดงการติดตาม สแต็ก in *General Programming Concepts: Writing and Debugging Programs*

คำสั่งย่อย `use`

```
use [ { + | Directory | 'RegularExpression = NewPath' } ... ]
```

คำสั่งย่อย `use` เช็ต รายการของไดเรกทอรีที่จะถูกค้นหาและการแม่พพาธที่จะถูกใช้ เมื่อดีบั๊กโปรแกรม `dbx` ค้นหา ไฟล์ต้นฉบับ ถ้าคำสั่งย่อย `use` ถูกระบุโดยไม่มี อาร์กิวเมนต์ รายการปัจจุบันของไดเรกทอรีที่จะถูกค้นหาและการแม่พพาธที่จะถูกใช้ จะถูกแสดง

@ (เครื่องหมาย at) เป็นไดเรกทอรีพิเศษที่กำหนดให้โปรแกรม `dbx` ให้ค้นหาที่ข้อมูลชื่อพพาธแบบเต็ม ในอ็อบเจกต์ไฟล์ ถ้ามี ถ้าคุณมีไดเรกทอรีแบบสัมพันธ์ที่เรียก @ เพื่อค้นหา คุณต้องใช้ ./@ ใน พาคค้นหา

คำสั่งย่อย `use` ใช้ + (เครื่องหมายบวก) เพื่อเพิ่มไดเรกทอรีหรือการแม่พพาธเพิ่มเติมเข้ากับ รายการของไดเรกทอรีที่ต้องการค้นหา + แสดง รายการปัจจุบันของไดเรกทอรี และการแม่พพาธเมื่อระบุเป็นอินพุต ให้กับคำสั่งย่อย `use` เมื่อต้องการผนวกไดเรกทอรีต่อท้าย หรือการแม่พพาธ กับท้ายของรายการปัจจุบัน ต้องระบุ + หน้าไดเรกทอรีหรือการแม่พพาธใหม่ เมื่อต้องการเพิ่มไดเรกทอรีที่ตอนต้น ของรายการปัจจุบัน ต้องระบุ + หลัง ไดเรกทอรีหรือการแม่พพาธใหม่ หากคุณมีไดเรกทอรีที่ชื่อ + ให้ระบุชื่อพพาธแบบเต็มสำหรับไดเรกทอรี (เช่น ./+ หรือ /tmp/+)

คำสั่งย่อย `use` แพลสตรึงที่อยู่ใน [ และ ] (วงเล็บเหลี่ยม) ซึ่งมี = (เครื่องหมายเท่ากับ) เป็นการแม่พพาธ การแม่พพาธเหล่านี้ ถูกใช้กับไดเรกทอรี @ พิเศษ ช่วยให้ผู้ใช้แสดงตำแหน่งซอร์สไฟล์ได้ง่ายขึ้น ในกรณีที่ตั้งโครงสร้างไดเรกทอรีของซอร์สไฟล์ ถูกเปลี่ยนแปลงตำแหน่งใหม่ หลังการคอมไพล์

กฎต่อไปนี้จะใช้เมื่อมีการพยายามค้นหาไฟล์ต้นฉบับ ระหว่างการดีบั๊ก:

- ไดเรกทอรีในรายการถูกหาค่าในลำดับที่ระบุ
- เมื่อทำการประเมินผลไดเรกทอรีในรายการ ไดเรกทอรีถูกใช้ค้นหา ไฟล์ที่ระบุ ถ้าไฟล์มีอยู่ในไดเรกทอรีและอ่านได้ ไฟล์นี้จะถูกใช้
- เมื่อทำการประเมินผลของไดเรกทอรี @ พิเศษ เมื่อมีอย่างน้อยหนึ่งการแม่พพาธถูกระบุ ถ้าส่วน *RegularExpression* ของการแม่พพาธตรงกับ *n* อักขระแรกของข้อมูลชื่อ พพาธแบบเต็มในอ็อบเจกต์ไฟล์ และการแทนค่าของส่วน *NewPath* ของการแม่พพาธจะได้ไฟล์ที่อ่านได้ ไฟล์นี้จะถูกใช้

- เมื่อทำการประเมินผลไดเร็กทอรี@ พิเศษ เมื่อไม่มีการแม็พพาร ถูกระบุ หรือไม่พบค่าที่ตรง ไดเร็กทอรีที่สอดคล้อง กับข้อมูลชื่อพารแบบเต็มจะถูกค้นหา ถ้าไฟล์มีอยู่ในไดเร็กทอรีและอ่านได้ ไฟล์นี้จะถูกใช้
- ถ้ามีการแม็พมากกว่าหนึ่งพารให้ไฟล์ที่อ่านได้ การแม็พ พารที่มี *RegularExpression* ตรงกับอักขระมากที่สุด (1 ... n) ของข้อมูลชื่อพารแบบเต็ม (นั่นคือ ระบุเจาะจงมากที่สุด) ถูกใช้และไฟล์ผลลัพธ์ที่ได้ถูกนำไปใช้
- ถ้ามีการแม็พพารที่ให้ไฟล์ที่อ่านได้มากกว่าหนึ่งพาร และแต่ละ การแม็พพารมีข้อมูลจำเพาะเหมือนกัน การแม็พพารที่ไกลที่สุดกับ จุดเริ่มต้นของรายการถูกใช้และไฟล์ที่เป็นผลลัพธ์จะถูกนำมาใช้

หมายเหตุ: ถ้าไดเร็กทอรี@ พิเศษไม่ใช่สมาชิกของ รายการ การแม็พพารใดๆ ที่อาจถูกระบุจะถูกข้าม

### ตัวอย่าง

1. เมื่อต้องการเปลี่ยนรายการของไดเร็กทอรีที่จะถูกค้นหาเป็นไดเร็กทอรี ปัจจุบัน (.), ไดเร็กทอรีพารেন্ট (..) และไดเร็กทอรี /tmp ให้ป้อน:  

```
use . . . /tmp
```
2. เมื่อต้องการเปลี่ยนรายการของไดเร็กทอรีที่จะถูกค้นหาเป็นไดเร็กทอรี ปัจจุบัน (.), ไดเร็กทอรีซึ่งไฟล์ต้นฉบับตั้งอยู่ในเวลา คอมไพล์ (@) และไดเร็กทอรี ../source ให้ป้อน:  

```
use . @ ../source
```
3. เมื่อต้องการเพิ่มไดเร็กทอรี /tmp2 เข้ากับรายการของ ไดเร็กทอรีที่ต้องการค้นหาให้ป้อน:  

```
use + /tmp2
```
4. เมื่อต้องการเพิ่มไดเร็กทอรี /tmp3 ให้กับจุดเริ่มต้น ของรายการไดเร็กทอรีที่จะถูกค้นหาให้ป้อน:  

```
use /tmp3 +
```
5. เมื่อต้องการแสดงไฟล์ต้นฉบับซึ่งข้อมูลชื่อพารแบบเต็ม เริ่มต้นด้วย /home/developer และขณะนี้อยู่ ภายใต้ /mnt ให้ป้อน:  

```
use + [/home/developer=/mnt]
```
6. เมื่อต้องการสั่งให้โปรแกรม dbx ค้นหาภายใต้ /latest ก่อนและจากนั้น หากไม่มีไฟล์ที่นั่น เมื่อค้นหาภายใต้ /stable สำหรับ ไฟล์ที่มีข้อมูลชื่อพารแบบเต็มที่เริ่มต้นด้วย /home/developer ให้ป้อน:  

```
use + [/home/developer=/latest] [/home/developer=/stable]
```

และดูที่คำสั่งย่อย edit และคำสั่งย่อย list

### คำสั่งย่อย whatis

**whatis** *Name*

คำสั่งย่อย **whatis** แสดงการประกาศของ *Name*, ซึ่งพารามิเตอร์ *Name* กำหนด ตัวแปร โพรซีเจอร์ หรือชื่อฟังก์ชัน แยกด้วยชื่อบล็อกเป็นทางเลือก

หมายเหตุ: ใช้คำสั่งย่อย **whatis** เฉพาะขณะรันโปรแกรมการดีบั๊ก dbx

### ตัวอย่าง

1. เมื่อต้องการแสดงการประกาศของตัวแปร x ให้ป้อน:  

```
whatis x
```

2. เมื่อต้องการแสดงการประกาศของโปรซีเดอร์ main ให้ป้อน:

```
whatis main
```

3. เมื่อต้องการแสดงการประกาศของตัวแปร x ภายในฟังก์ชัน main ให้ป้อน:

```
whatis main.x
```

4. เมื่อต้องการพิมพ์การประกาศของ enumeration, structure หรือ union tag ให้ใช้ \$\$TagName:

```
(dbx) whatis $$status  
enum $$status { run, create, delete, suspend };
```

### คำสั่งย่อย where

```
where [ all | $threadumber [(startframe endframe)] ...] [ startframe endframe ] [ >File ]
```

คำสั่งย่อย **where** แสดงรายการของ โปรซีเดอร์และฟังก์ชัน ที่แอสคิทที่พีที่เชื่อมโยงกับ หมายเลขเฟรม *startframe* ถึง *endframe* การกำหนดหมายเลขของสแต็กเฟรมเริ่มตั้งแต่สแต็กเฟรม ของแอสคิทที่พีฟังก์ชันขณะนี้ (ซึ่งจะเป็นหมายเลข 0 เสมอ) ถ้ามี *n* เฟรม เฟรมของฟังก์ชัน **main** จะมีหมายเลขเป็น *n*-1 โดยการใส่แฟล็ก *>File* เอาต์พุตของคำสั่งย่อยนี้สามารถถูกเปลี่ยนทิศทางไปยังไฟล์ ที่ระบุ

ในสภาวะแวดล้อมที่มีหลายเธรด อีอพชัน **all** จะแสดง รายละเอียดของสแต็กสำหรับเธรดทั้งหมดที่พร้อมใช้งาน รายละเอียดสแต็กของแต่ละ threads ถูกแสดงโดยการระบุหมายเลข thread ตามด้วยคำสั่งย่อย **where** ถ้าเฟรมเริ่มต้นและสิ้นสุดสำหรับแต่ละเธรด ไม่ถูกระบุ สแต็กเฟรมจะแสดงโดยหมายเลขเริ่มต้นและสิ้นสุด เฟรมโกลบอล คำสั่งที่ไม่มีอีอพชันจะแสดงสแต็ก เฟรมของเธรดปัจจุบัน

### แฟล็ก

ไอเท็ม	คำอธิบาย
<i>&gt;File</i>	เปลี่ยนทิศทางเอาต์พุตไปไฟล์ที่ระบุ

ดูที่คำสั่งย่อย **frame**, คำสั่งย่อย **up** และคำสั่งย่อย **down** และ ดูที่ การแสดงการติดตามสแต็ก in *General Programming Concepts: Writing and Debugging Programs*

### ตัวอย่าง

1. เมื่อต้องการแสดงรายละเอียดสแต็กของเธรดทั้งหมด ให้ป้อน:

```
where all
```

2. เมื่อต้องการแสดงรายละเอียดสแต็กของเธรด \$t1, \$t2 และ \$t3 ให้ป้อน:

```
where $t1 $t2 $t3
```

3. เมื่อต้องการแสดงรายละเอียดสแต็กของเธรด \$t2 ที่มีสแต็กเฟรม 2-3, \$t1 และ \$t3 ทั้งกับสแต็กเฟรม 1-4 ให้ป้อน:

```
where $t1 $t2(2 3) $t3 1 4
```

ดูที่คำสั่งย่อย **frame**, คำสั่งย่อย **up** และคำสั่งย่อย **down** และ ดูที่ การแสดงการติดตามสแต็ก in *General Programming Concepts: Writing and Debugging Programs*

### คำสั่งย่อย whereis

## whereis Identifier

คำสั่งย่อย **whereis** แสดงการระบุคุณสมบัติเต็มของสัญลักษณ์ทั้งหมดซึ่งมีชื่อชื่อ ตรงกับตัวระบุที่กำหนด ลำดับการพิมพ์ สัญลักษณ์ ไม่สำคัญ

### ตัวอย่าง

เมื่อต้องการแสดงชื่อที่ระบุคุณสมบัติ ของสัญลักษณ์ทั้งหมด ที่ชื่อ x ให้ป้อน:

```
whereis x
```

และดูที่คำสั่งย่อย **which**

## คำสั่งย่อย which

### which Identifier

คำสั่งย่อย **which** แสดงการระบุตัวบ่งชี้ แบบเต็ม การระบุคุณสมบัติเต็ม ประกอบด้วยรายการของบล็อกภายนอกที่สัมพันธ์ กับ identifier

### ตัวอย่าง

เมื่อต้องการแสดงการระบุคุณสมบัติเต็มของสัญลักษณ์ x ให้ป้อน:

```
which x
```

ดูที่คำสั่งย่อย **whereis** และ ดูที่ การกำหนดสโคปของชื่อ ใน *in General Programming Concepts: Writing and Debugging Programs*

## ไฟล์

ไอเท็ม	คำอธิบาย
a.out	อ็อบเจกต์ไฟล์ มีอ็อบเจกต์โค้ด
core	มีคอร์ดัมพ์
.dbxinit	มีคำสั่งเริ่มต้น

## แอฟพลิเคชันการดีบั๊ก ที่ใช้หน่วยความจำในการทำธุรกรรม

สำหรับแอฟพลิเคชันที่ใช้ หน่วยความจำในการทำธุรกรรม (TM) ตัวช่วยการดีบั๊กที่เชื่อถือได้ที่สุด คือ ตัวแปร transaction exception and summary register (*\$texasr*), transaction failure handler address register (*\$tfhar*), และ transaction failure instruction address register (*\$tfiar*)

ตัวแปร *\$texasr*, *\$tfhar* และ *\$tfiar* สามารถแสดงโดยใช้ คำสั่งย่อย **print** ซึ่งคล้ายกับการแสดง ตัวแปร *\$iar* อย่างไรก็ตาม คุณไม่สามารถดำเนินการกับค่า ในรีจิสเตอร์เหล่านี้โดยใช้คำสั่งย่อย **assign**

ตัวแปร *\$tfhar* และ *\$tfiar* ประกอบด้วยแอดเดรสของคำสั่งจากเซกเมนต์ข้อความที่ถูกดีบั๊กใน ความล้มเหลวของธุรกรรม เช่นเดียวกับตัวแปร *\$iar* สองบิตด้านซ้ายสุดของตัวแปรรีจิสเตอร์ *\$tfhar* และ *\$tfiar* จะถูกสงวนไว้และไม่สามารถพิจารณา สำหรับการอ่านแอดเดรส จากตัวแปรรีจิสเตอร์เหล่านี้

**หมายเหตุ:** เมื่อคุณใช้ตัวแปรจีเอสเตอร์ `$tfiar` และ `$tfhar` กับคำสั่งย่อย `list` คำสั่ง `dbx` จะดึงข้อมูลแอดเดรสหลังจากแยกสองบิตด้านซ้ายสุด จากเนื้อหาของตัวแปรจีเอสเตอร์เหล่านี้

(dbx) list at \$tfiar

หรือ

(dbx) list at \$tfhar

สาเหตุของ ความล้มเหลวของธุรกรรมสามารถกำหนดโดยการใช้ตัวแปร `$texasr` คำสั่งย่อย `tm_status` จะแปลสาเหตุและธรรมชาติของความล้มเหลวของธุรกรรม

คำสั่ง `dbx` จะมอนิเตอร์การมีอยู่ของธุรกรรมและแสดงสาเหตุของ ความล้มเหลวของธุรกรรมผ่านชุดของข้อความต่อไปนี้ ข้อความต่อไปนี้แสดงแสดงหลังจากคุณรันคำสั่งย่อย `run`, คำสั่งย่อย `rerun` หรือคำสั่งย่อย `continue`

- กระบวนการ {PID} อาจทำให้ธุรกรรมล้มเหลว - `$texasr`, `$tfiar`, `$tfhar` ใช้ได้และอาจถูกตรวจสอบ
- กระบวนการ {PID} อาจดำเนินการธุรกรรม - `$texasr`, `$tfiar`, `$tfhar` ใช้ได้และอาจถูกตรวจสอบ  
ข้อความนี้แสดงเมื่อตั้งค่าตัวแปรภายใน `$trace_good_transaction` เท่านั้น
- กระบวนการ {PID} อยู่ใน Transactional State - การดำเนินการตีบักที่ผ่าน `dbx` อาจทำให้ธุรกรรมล้มเหลวซ้ำๆ หรือ มีลักษณะการทำงานที่ไม่ได้กำหนดไว้
- กระบวนการ {PID} อยู่ใน Suspended State - การดำเนินการตีบักที่ผ่าน `dbx` อาจทำให้ธุรกรรมล้มเหลวซ้ำๆ หรือ มีลักษณะการทำงานที่ไม่ได้กำหนดไว้

โดยที่ PID เป็น ID กระบวนการของกระบวนการที่ถูกตีบัก

**ข้อมูลที่เกี่ยวข้อง:**

คำสั่ง `adb`

คำสั่ง `a.out`

รูทีนย่อย `ptrace`

การตีบักโปรแกรม

---

## คำสั่ง `dc`

### วัตถุประสงค์

จัดให้มีเครื่องคิดเลขตั้งโต๊ะแบบโต้ตอบสำหรับการคำนวณคณิตศาสตร์เลขจำนวนเต็ม ที่มีความละเอียดไม่แน่นอน

### ไวยากรณ์

`dc [File]`

### คำอธิบาย

คำสั่ง `dc` คือเครื่องคิดเลข arbitrary-precision คำสั่ง `dc` รับอินพุต จากพารามิเตอร์ `File` หรืออินพุตมาตรฐาน จนกว่าจะอ่านถึงอักขระสิ้นสุดไฟล์ เมื่อคำสั่ง `dc` ได้รับอินพุต คำสั่งจะประเมินค่าและเขียนการประเมิน ไปยังเอาต์พุตมาตรฐาน โดยทำงานกับ

จำนวนเต็มฐานสิบ แต่คุณสามารถระบุให้เป็นตามข้อมูลอินพุต ตามข้อมูลเอาต์พุต และจำนวนของจุดทศนิยมที่จะใช้ คำสั่ง dc ถูกจัดโครงสร้างเหมือนการทาสแด้ก, การคำนวณ reverse Polish notation

คำสั่ง bc คือตัวประมวลผลก่อน สำหรับคำสั่ง dc โดยจัดเตรียม infix notation และไวยากรณ์เหมือนกับภาษา C ซึ่งมีฟังก์ชันและโครงสร้างการควบคุม สำหรับโปรแกรม

## คำสั่งย่อ

ไอเท็ม	คำอธิบาย
c	ล้างสแต็ก: คำสั่ง dc pop คำทั้งหมด บนสแต็ก
d	ทำซ้ำค่าบนสุดบนสแต็ก
f	แสดงค่าทั้งหมดบนสแต็ก
i	Pop ค่าบนสุดบนสแต็กและใช้ค่าเป็น radix ตัวเลขสำหรับอินพุตเพิ่มเติม
I	Push ค่าบนสุดจากบนสุดของสแต็ก
k	Pops ค่าบนสุดของสแต็กและใช้ค่านั้นเป็นค่าสเกล ที่ไม่มีค่าติดลบ จำนวนตำแหน่งที่เหมาะสมถูกแสดงบนเอาต์พุต และถูกดูแล ระหว่างการคูณ การหาร และการยกกำลัง การโต้ตอบของ scale factor, อินพุตฐาน และเอาต์พุตฐาน เป็นไปอย่างเหมาะสมถ้าทั้งหมดถูกเปลี่ยนแปลงร่วมกัน
l x	Push ค่าในรีจิสเตอร์ที่แสดงโดยตัวแปร x บนสแต็ก รีจิสเตอร์ที่แสดงโดยตัวแปร x จะไม่เปลี่ยนแปลง รีจิสเตอร์ทั้งหมดเริ่มต้นด้วยค่า 0
Lx	ปฏิบัติกับตัวแปร x เหมือนเป็นสแต็ก และ pops ค่าบนสุดลงในสแต็กหลัก
o	Pop ค่าบนสุดบนสแต็กและใช้ค่าเป็น radix ตัวเลขสำหรับเอาต์พุตเพิ่มเติม
O	Push ค่าเอาต์พุตจากบนสุดของสแต็ก
p	แสดงค่าบนสุดบนสแต็ก ค่าบนสุดไม่มีการเปลี่ยนแปลง
P	แปลค่าบนสุดของสแต็กเป็นสตริง ลบออก และแสดง ผล
q	ออกจากโปรแกรม ถ้าคำสั่ง dc กำลังรัน สตริง, จะ pops ระดับการเรียกซ้ำสองระดับ
Q	Pops ค่าบนสุดบนสแต็กและบนระดับการดำเนินการสตริง ตามค่านั้น
s x	Pops ค่าบนสุดของสแต็กและเก็บไว้ในรีจิสเตอร์ชื่อ x โดยที่ตัวแปร x สามารถเป็นอักขระใดก็ได้
Sx	ปฏิบัติกับตัวแปร x เหมือนเป็นสแต็ก โดยจะ pop ค่าบนสุดของของสแต็กหลักและ push ค่าดังกล่าวลงบนสแต็กที่แสดงโดยตัวแปร x
v	แทนที่อิลิเมนต์บนสุดบนสแต็กด้วยสแควร์รูท เศษที่มีอยู่ ของอ็อพชันจะถูกนำมาคิด หรือมีจะนั้นค่าสเกล จะถูกละเว้น
x	ปฏิบัติอิลิเมนต์บนสุดของสแต็กเป็นสตริงอักขระและรัน เป็นสตริงของคำสั่ง dc
X	แทนที่หมายเลขบนสุดของสแต็กด้วยค่าสเกล
z	พุดจำนวนอิลิเมนต์ในสแต็กด้วยจำนวนหลักใน ตัวเลขนั้น
Z	แทนที่ตัวเลขบนสุดในสแต็กด้วยจำนวนหลักใน ตัวเลขนั้น
Number	พุดค่าที่ระบุบนสแต็ก Number เป็นสตริงที่แยกไม่ได้ของตัวเลข 0 ถึง 9 เพื่อระบุ ค่าลบ นำหน้าด้วย _ (underscore) ตัวเลขสามารถมี ทศนิยม
+ - / * % ^	บวก (+), ลบ (-), คูณ (*), หาร(/), ส่วนที่เหลือ (%) หรือยกกำลัง (^) สองค่าบนสุดบนสแต็ก คำสั่ง dc pops สองรายการออกจากสแต็ก และ pushes ผลลัพธ์ กลับไปบนสแต็กที่ตำแหน่งเดิม คำสั่ง dc ละเว้น เศษของการยกกำลัง
[String]	นำพารามิเตอร์ String ในวงเล็บไปไว้ ตำแหน่งบนสุดของสแต็ก
[< > < >]x	ป้อนสองอิลิเมนต์บนสุดของสแต็กและเปรียบเทียบ ประเมินรีจิสเตอร์ที่แสดงโดยตัวแปร x หากเป็นไปตามความสัมพันธ์ที่ระบุ
!	แปลส่วนที่เหลือของบรรทัดเป็นคำสั่งระบบปฏิบัติการ
?	รับและรันบรรทัดอินพุต
::	คำสั่ง bc ใช้อักขระเหล่านี้สำหรับการดำเนินการ array

## ตัวอย่าง

1. เมื่อต้องการใช้คำสั่ง dc เป็นเครื่องคิดเลข ให้ป้อนคำสั่งต่อไปนี้:

```
You: 1 4 / p
System: 0
You: 1 k [ Keep 1 decimal place ]s.
      1 4 / p
System: 0.2
```

```

You: 3 k [ Keep 3 decimal places ]s.
      1 4 / p
System: 0.250
You: 16 63 5 / + p
System: 28.600
You: 16 63 5 + / p
System: 0.235

```

หมายเหตุสามารถใช้ในคำสั่ง dc ดังในตัวอย่างต่อไปนี้ ข้อคิดเห็นจะอยู่ในวงเล็บเหลี่ยมและสามารถตามด้วยอักขระ s. ข้อคิดเห็นในรูปแบบ [Comment]s. จะถูกข้าม โดยคำสั่ง dc เฉพาะหมายเหตุที่อยู่ในวงเล็บเหลี่ยมเท่านั้นที่จะเก็บไว้บนตานบนสุดของสแต็ก

เมื่อคุณป้อนนิพจน์คำสั่ง dc โดยตรงจากคีย์บอร์ด กด Ctrl-D เพื่อสิ้นสุดเซสชันคำสั่ง bc และกลับไปเชลล์บรรทัดคำสั่ง

## 2. เมื่อต้องการโหลดและรันไฟล์โปรแกรม dc ให้ป้อนคำสั่งต่อไปนี้:

```

You: dc prog.dc
      5 lf x p [ 5 factorial ]s.
System: 120
You: 10 lf x p [ 10 factorial ]s.
System: 3628800

```

รายการนี้จะแปลโปรแกรม dc ที่บันทึกไว้ใน โปรแกรมไฟล์ prog.c จากนั้นอ่านจาก คีย์บอร์ดของเวิร์กสเตชัน lf x จะประเมินฟังก์ชันที่เก็บไว้ในรีจิสเตอร์ f ซึ่งสามารถกำหนดในโปรแกรมไฟล์ prog.c เป็น:

```

[ f: compute the factorial of n ]s.
[ (n = the top of the stack) ]s.
[ If 1>n do b; If 1<n do r ]s.
  [d 1 >b d 1 <r] sf
[ Return f(n) = 1 ]s.
  [d - 1 +] sb
[ Return f(n) = n * f(n-1) ]s.
  [d 1 - lf x *] sr

```

คุณสามารถสร้างโปรแกรมไฟล์ dc ด้วยเท็กซ์เอดิเตอร์หรือด้วยแฟล็ก -c (compile) ของคำสั่ง bc เมื่อคุณป้อนนิพจน์คำสั่ง dc โดยตรงจากคีย์บอร์ด กด Ctrl-D เพื่อสิ้นสุดเซสชันคำสั่ง bc และกลับไปเชลล์บรรทัดคำสั่ง

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/dc	มีคำสั่ง dc

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง bc

## คำสั่ง dcp

### วัตถุประสงค์

รันคำสั่งพร้อมกันบนโน้ตและอุปกรณ์ฮาร์ดแวร์จำนวนมาก

## ไวยากรณ์

```
dcp [-h] [-V] [-q] [-a] [--all-nodes context_list] [-A] [--all-devices context_list] [-n node_list] [-N nodegroups] [-d device_list] [-D devicegroups] [-C context] [-f fanout] [-l user_ID] [-o node_options] [-O device_options] [-p] [-P] [-Q] [-r node_remote_copy] [--device-rcp device_remote_copy] [-R] [-t timeout] [-X env_list] [-T] [-v] source_file... target_path
```

## คำอธิบาย

คำสั่ง **dcp** คัดลอกไฟล์ไปยังและจากโหนดเป้าหมายแบบรีโมต อุปกรณ์ฮาร์ดแวร์ หรือทั้งสองอย่างพร้อมเพียงกัน เป้าหมายสามารถเลือกจากคอนเท็กซ์จำนวนมากได้ คอนเท็กซ์คือฐานข้อมูลเป้าหมายที่มีนิยามของโหนดและอุปกรณ์ เช่น NIM คำสั่ง **dcp** จะออกคำสั่งคัดลอกแบบรีโมตสำหรับแต่ละโหนดหรืออุปกรณ์ที่ระบุ เมื่อไฟล์ถูกดึงจากเป้าหมายไฟล์เหล่านั้นจะถูกวางลงใน *target\_path* โดยมีชื่อของโหนดแบบรีโมตหรืออุปกรณ์ที่ผนวกกับชื่อ *source\_file* ที่ถูกคัดลอก คำสั่ง `/usr/bin/rcp` คือโมเดลสำหรับไวยากรณ์และความปลอดภัย คำสั่ง **dcp** คือ DSM Distributed Shell Utility คอนพิกูเรชันและค่าติดตั้งสภาวะแวดล้อมสำหรับ **dsh** มีผลต่อลักษณะการทำงานของ **dcp** โปรดดูที่คำสั่ง **dsh** สำหรับรายละเอียด

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
TARGET CONTEXT	ข้อกำหนดคุณลักษณะของคอนเท็กซ์เป้าหมายเป็นข้อกำหนดเฉพาะสำหรับคำสั่ง <b>dcp</b> และ <b>dsh</b> โปรดดูที่บริบทเป้าหมายในเพจ <b>dsh man</b> สำหรับรายละเอียดเกี่ยวกับการระบุบริบทสำหรับคำสั่ง <b>dcp</b>
TARGET SPECIFICATION	ข้อกำหนดคุณลักษณะของเป้าหมายเป็นข้อกำหนดเฉพาะสำหรับคำสั่ง <b>dcp</b> และ <b>dsh</b> โปรดดูที่เพจ <b>dsh man</b> สำหรับรายละเอียดเกี่ยวกับการระบุเป้าหมายสำหรับคำสั่ง <b>dcp</b>
TARGET LISTS	ไวยากรณ์รายชื่อเป้าหมายเป็นไวยากรณ์เฉพาะสำหรับคำสั่ง <b>dcp</b> และ <b>dsh</b>
REMOTE USER	<i>user_ID</i> สามารถระบุได้สำหรับ คำสั่ง remote copy ข้อกำหนดคุณลักษณะผู้ใช้แบบรีโมตเป็นข้อกำหนดเฉพาะสำหรับคำสั่ง <b>dcp</b> และ <b>dsh</b>

ไอเอ็ม  
REMOTE COPY  
COMMAND

คำอธิบาย

คำสั่ง `dcp` ใช้คำสั่ง `remote copy` ที่สามารถกำหนดค่าได้เพื่อรันคำสั่งรีโมตบนเป้าหมายแบบรีโมต การสนับสนุน ได้รับการจัดเตรียมอย่างชัดเจนสำหรับคำสั่ง AIX Remote Shell `rcp`, คำสั่ง OpenSSH `scp` และคำสั่ง `rsync` สำหรับโหมดเป้าหมาย คำสั่ง `remote copy` จะถูกกำหนดโดยใช้พารามิเตอร์ในลำดับที่สูงกว่า:

1. แฟล็ก `-r`
2. ตัวแปรสถานะแวดล้อม `DCP_NODE_RCP`
3. คำสั่ง `/usr/bin/rcp`

สำหรับเป้าหมายของอุปกรณ์ `remote shell` จะถูกกำหนดไว้โดยการเรียงลำดับของการมาก่อนดังต่อไปนี้:

1. แฟล็ก `--device-rcp`
2. ตัวแปรสถานะแวดล้อม `DCP_DEVICE_RCP`
3. คำสั่ง `remote copy` อุปกรณ์ดีฟอลต์ถูกกำหนดโดย บริบทเป้าหมาย
4. แอ็ททริบิวต์ `RemoteCopyCmd` ที่ถูกกำหนดสำหรับ อุปกรณ์เป้าหมาย

คำสั่ง `remote copy` ถูกระบุด้วยแฟล็กบรรทัดรับคำสั่ง หรือตัวแปรสถานะแวดล้อมโดยใช้ไวยากรณ์ต่อไปนี้:

```
[context:]path[, [context:]path]...
```

โดยที่ `path` เป็น พาสไปยังคำสั่ง `remote copy` command และ `context`: ระบุ บริบทของคำสั่ง `remote copy` เพื่อใช้สำหรับการคัดลอกไฟล์ พารของคำสั่ง `remote copy` ที่ระบุโดยไม่มีบริบทใช้กับบริบทอื่นทั้งหมดที่ไม่ได้ระบุพารของคำสั่ง `remote copy` อย่างชัดเจน ในรายการ อ็อพชันของคำสั่ง `remote copy` สามารถกำหนดค่าโดยใช้ แฟล็กบรรทัดรับคำสั่งหรือตัวแปรสถานะแวดล้อม สำหรับโหมดเป้าหมาย อ็อพชันของคำสั่ง `remote copy` ถูกกำหนดโดยการเรียงลำดับ การมาก่อนต่อไปนี้:

1. แฟล็ก `-o`
2. ตัวแปรสถานะแวดล้อม `DCP_NODE_OPTS`

สำหรับเป้าหมายของอุปกรณ์ อ็อพชันของคำสั่ง `remote copy` จะถูกกำหนดไว้โดยการเรียงลำดับของการมาก่อนต่อไปนี้:

1. แฟล็ก `-O`
2. ตัวแปรสถานะแวดล้อม `DCP_DEVICE_OPTS`

อ็อพชันของคำสั่ง `remote copy` ถูกระบุโดยใช้ไวยากรณ์ต่อไปนี้:

```
[context:] "options" "[, [context:] "options"]...
```

โดยที่ `options` เป็น อ็อพชันของคำสั่ง `remote copy` และ `context`: ระบุ บริบทของอ็อพชันรีโมตเซลล์เพื่อใช้สำหรับการคัดลอกไฟล์ อ็อพชันที่ระบุ โดยไม่ใช้บริบทกับบริบทอื่นทั้งหมดที่ไม่ได้ระบุ อ็อพชันอย่างชัดเจนในรายการ อ็อพชัน ต้องระบุในเครื่องหมายคำพูดคู่ ("") เพื่อแยกความแตกต่างออกจาก อ็อพชัน `dcp`

ไอเท็ม  
COMMAND  
EXECUTION

คำอธิบาย

ระบุกระบวนการคำสั่ง remote copy ที่ทำงานพร้อมกัน (fanout) ที่สามารถระบุโดยใช้แฟล็ก -f หรือตัวแปรสภาวะแวดล้อม DSH\_FANOUT fanout ถูกจำกัดไว้โดยจำนวนของคำสั่ง remote shell ที่สามารถรัน แบบขนานได้ คุณสามารถทดลองใช้คำสั่ง DSH\_FANOUT บนเซิร์ฟเวอร์การจัดการของคุณเพื่อดูว่าค่าที่สูงกว่าเหมาะสมหรือไม่ ค่าการหมดเวลาสำหรับการเรียกใช้คำสั่ง remote copy สามารถระบุด้วย แฟล็ก -t หรือตัวแปรสภาวะแวดล้อม DSH\_TIMEOUT หากเป้าหมายแบบรีโมตใดๆ ไม่ตอบสนองภายในค่าหมดเวลา คำสั่ง dcp จะแสดงข้อความแสดงความคิดเห็นและออก แฟล็ก -T จัดเตรียมข้อมูลการติดตามการวินิจฉัยสำหรับการเรียกใช้คำสั่ง dcp คำติดตั้งดีพอลต์ และคำสั่ง remote copy ที่แท้จริงที่รันกับเป้าหมายแบบรีโมต จะถูกแสดง คำสั่ง dcp สามารถรันแบบไม่มีการโต้ตอบโดยใช้แฟล็ก -Q ซึ่งจะไม่มีเอาต์พุตมาตรฐานหรือข้อผิดพลาดมาตรฐานที่ถูกแสดง

พารามิเตอร์สำหรับ ตัวแปรนี้มีดังนี้:

source\_file...

ระบุพารที่สมบูรณ์สำหรับไฟล์ที่จะถูกตัดลอกไปยังหรือตัดออกจาก เป้าหมาย คุณสามารถระบุไฟล์จำนวนมากได้ เมื่อใช้กับแฟล็ก -R เฉพาะไดเรกทอรีเดี่ยวเท่านั้นที่สามารถระบุได้ เมื่อใช้กับแฟล็ก -P เฉพาะไฟล์เดี่ยวเท่านั้นที่สามารถระบุได้

target\_path

ระบุพารแบบสมบูรณ์เพื่อตัดลอกไฟล์ source\_file หนึ่งไฟล์หรือมากกว่า ไปยังเป้าหมาย หากระบุแฟล็ก -P ไว้ target\_path คือตำแหน่งโลคัลโฮสต์สำหรับไฟล์ที่ตัดลอก โครงสร้างไดเรกทอรีของไฟล์แบบรีโมตจะถูกสร้างขึ้นใหม่ภายใต้ target\_path และชื่อเป้าหมายแบบรีโมตจะถูกผนวกเข้ากับชื่อ source\_file ที่ถูกตัดลอกในไดเรกทอรี target\_path

## คีย์เวิร์ด

ไอเท็ม

-a  
-A  
--all-devices  
context\_list  
--all-nodes context\_list  
-C  
--contextcontext  
--device-rcp  
device\_remote\_copy

คำอธิบาย

รวมในเป้าหมายที่แสดงโหนดทั้งหมดที่ถูกนิยามใน คอนเท็กซ์ดีพอลต์ บริบทดีพอลต์สามารถตั้งค่าได้โดยใช้แฟล็ก -C หรือตัวแปรสภาวะแวดล้อม DSH\_CONTEXT  
รวมในเป้าหมายที่แสดงอุปกรณ์ทั้งหมดที่ถูกนิยามใน คอนเท็กซ์ดีพอลต์ บริบทดีพอลต์สามารถตั้งค่าได้โดยใช้แฟล็ก -C หรือตัวแปรสภาวะแวดล้อม DSH\_CONTEXT แฟล็กนี้ถูกปิดใช้งานบน HMCs  
จะรวมในรายการเป้าหมายของอุปกรณ์ทั้งหมดที่กำหนดไว้ใน บริบทที่แสดงรายการใน context\_list คอนเท็กซ์ดีพอลต์ไม่ได้รวมอยู่ในรายการนี้ แฟล็กนี้ถูกปิดใช้งานบน HMC  
จะรวมในรายการเป้าหมายของโหนดทั้งหมดที่กำหนดไว้ใน บริบทที่แสดงรายการใน context\_list คอนเท็กซ์ดีพอลต์ไม่ได้รวมอยู่ในรายการนี้  
ระบุพารแบบเต็มของคำสั่ง remote copy ที่ใช้เพื่อตัดลอกไฟล์ไปยังหรือจากอุปกรณ์เป้าหมาย คำสั่ง remote copy สำหรับบริบทเฉพาะสามารถกำหนดโดยการรวม context: ก่อน พารระบุบริบทดีพอลต์เพื่อใช้เมื่อคำสั่ง dcp แก่ชื่อเป้าหมาย ค่าคอนเท็กซ์ต้องสอดคล้องกับโมดูลส่วนขยายคอนเท็กซ์ที่ถูกต้องในไดเรกทอรี /opt/ibm/sysmgmt/dsm/pm/Context  
เริ่มทำงานระบบย่อยการตรวจสอบ ไวยากรณ์ device\_remote\_copy มีดังต่อไปนี้:  
[context:]path[, [context:]path]...

-dl--devices device\_list

แฟล็กนี้ ถูกปิดใช้งานบน HMCs คีย์เวิร์ดนี้จะอ่านคำสั่งใน ไฟล์คอนฟิกูเรชันและกำหนดรีโมตเซลล์สำหรับอุปกรณ์เป้าหมาย ระบุรายชื่อเป้าหมายของอุปกรณ์ เพื่อรวมในรายชื่อเป้าหมาย ไวยากรณ์ device\_list คือ:  
[context:] [user\_ID@] device\_name[,\  
[context:][user\_ID@]device\_name]...

-Dl--devicegroups  
devicegroups

แฟล็กนี้ ถูกปิดใช้งานบน HMCs  
จะรวมในรายการเป้าหมายของอุปกรณ์ทั้งหมดที่กำหนดใน กลุ่มอุปกรณ์ที่ระบุไว้ในรายการ devicegroups ไวยากรณ์ devicegroups มีดังนี้:  
[context:] [user\_ID@]devicegroup[,\  
[context:]  
[user\_ID@]devicegroup]...

แฟล็กนี้ ถูกปิดใช้งานบน HMCs

## ไอเท็ม

-f | --fanout *fanout*  
-l | --user *user\_ID*  
-h | --help  
-n | --nodes *node\_list*

## คำอธิบาย

ระบุค่า fanout สำหรับจำนวนสูงสุดของ กระบวนการรีโมตเซลล์ที่รันพร้อมกัน สามารถระบุการประมวลผลแบบตามลำดับ โดยการระบุค่า fanout เป็น 1 หากไม่ได้ระบุแฟล็กนี้ จะใช้ค่า fanout ดีฟอลต์ที่เป็น 64  
ระบุชื่อผู้ใช้แบบรีโมตที่ต้องการใช้ สำหรับการเรียกทำงาน remote copy  
แสดงข้อมูลการใช้งานคำสั่ง  
ระบุรายชื่อเป้าหมายของโหนดเพื่อรวม ในรายชื่อเป้าหมาย ไวยากรณ์ *node\_list* คือ:

```
[context:] [user_ID@]node_name[, \
[context:]
[user_ID@]node_name]...
```

-o | --node-options  
*node\_options*

ระบุอ็อปชันเพื่อผ่านไปยังคำสั่ง remote copy สำหรับโหนดเป้าหมาย อ็อปชันต้องถูกระบุ ภายในเครื่องหมายคำพูดคู่เพื่อแยกความแตกต่างออกจากแฟล็กคำสั่ง dcp อ็อปชันสำหรับโหนดในบริบทที่ระบุสามารถกำหนดโดยการรวม context: ก่อน รายการอ็อปชัน ไวยากรณ์ของ *node\_options* คือ:

```
[context:] "options" [, [context:] "options"] ...
```

-N | --nodegroups  
*nodegroups*

รวมโหนดทั้งหมดในรายการเป้าหมายที่กำหนดไว้ใน กลุ่มโหนดที่ระบุในรายการ *nodegroups* ไวยากรณ์ของ *nodegroups* คือ:

```
[context:] [user_ID@]nodegroup[, \
[context:]
[user_ID@]nodegroup]...
```

-O --device-options  
*device\_options*

ระบุอ็อปชันเพื่อผ่านคำสั่ง remote copy สำหรับอุปกรณ์เป้าหมาย อ็อปชันต้องถูกระบุ ภายในเครื่องหมายคำพูดคู่เพื่อแยกความแตกต่างออกจากแฟล็กคำสั่ง dcp อ็อปชันสำหรับอุปกรณ์ในบริบทเฉพาะสามารถกำหนดโดยการรวม context: ก่อน รายการอ็อปชัน ไวยากรณ์ของ *device\_options* คือ:

```
[context:] "options" [, [context:] "options"] ...
```

-p | --preserve  
-P | --pull

## แฟล็กนี้ถูกปิดใช้งานบน HMCs

สวณคุณสมบัติของไฟล์ต้นฉบับไว้ ตามที่ได้นำไปใช้งานโดยคำสั่ง remote copy ที่กำหนดคอนฟิกไว้  
ดึง (คัดลอก) ไฟล์จากเป้าหมาย และวางไว้ในไดเรกทอรี *target\_path* บนโฮสต์แบบโลคัล *target\_path* ต้องเป็นไดเรกทอรี ไฟล์  
ที่ถูกดึงจากเครื่องรีโมตมี *\_target* ผวนกเข้ากับชื่อไฟล์เพื่อแยกความแตกต่างระหว่างกัน เมื่อใช้แฟล็ก -P กับแฟล็ก -R  
*\_target* จะถูกผนวก กับไดเรกทอรี ซึ่งมีเพียงหนึ่งไฟล์ต่อการเรียกทำงานคำสั่ง dcp -P | --pull เท่านั้นที่สามารถดึงได้จาก  
เป้าหมายที่ระบุไว้

-Q  
-q | --show-config

รันคำสั่ง dcp แบบไม่มีการโต้ตอบ ซึ่งจะไม่เฝ้าดูพัฒนาหรือข้อผิดพลาดมาตรฐานเป้าหมายที่ถูกแสดง  
แสดงค่าติดตั้งสถานะแวดล้อมปัจจุบันที่เกี่ยวข้องกับ คำสั่งยูทิลิตี้ dsh แฟล็กนี้รวมค่าของตัวแปรสถานะแวดล้อมทั้งหมด และ  
ค่าติดตั้งสำหรับคอนเท็กซ์ที่ติดตั้งไว้และคอนเท็กซ์ที่ถูกต้องใน ปัจจุบัน ค่าติดตั้งแต่ละค่าจะขึ้นต้นด้วย context: เพื่อระบุ  
บริบทต้นฉบับของค่าติดตั้ง

-r | --node-rcp  
*node\_remote\_copy*

ระบุพารามิเตอร์เพิ่มเติมของคำสั่ง remote copy ที่ใช้เพื่อคัดลอกไฟล์ไปยังหรือจากโหนดเป้าหมาย คำสั่ง remote copy สำหรับบริบท  
เฉพาะสามารถกำหนดโดยการรวม context: ก่อน พารามิเตอร์ของ *node\_remote\_copy* คือ:

```
[context:]path[, [context:]path]...
```

-R | --recursive

หาก *path* มี rsync จะถือว่าคำสั่ง rsync ดำเนินการ คัดลอกแบบรีโมต

-t | --timeout *timeout*

คัดลอกไฟล์แบบซ้ำๆ จากโลคัลไดเรกทอรีไปยังเป้าหมายแบบรีโมต หรือเมื่อระบุด้วยแฟล็ก -P ซึ่งจะดึง (คัดลอก) ไฟล์แบบ  
ซ้ำๆ จากรีโมตไดเรกทอรีไปยัง โลคัลโฮสต์ สามารถระบุไดเรกทอรีต้นทางไดเรกทอรีเดียวโดยใช้พารามิเตอร์ *source\_file*  
ระบุเวลาในหน่วยวินาที เพื่อรอให้คำสั่ง remote copy เสร็จสิ้นแต่ละเป้าหมายแบบรีโมต หากเป้าหมาย ไม่ตอบสนองภายในค่า  
การหมดเวลา คำสั่ง dcp จะแสดงข้อความแสดงข้อผิดพลาดและหยุดกระบวนการคัดลอกแบบรีโมต สำหรับเป้าหมายแบบรี  
โมต หากไม่ได้รับไว้ คำสั่ง dcp จะรอให้กระบวนการ remote copy เสร็จสิ้นแต่ละเป้าหมายอย่างไม่มีกำหนด

-T | --trace  
-v | --verify

เรียกทำงานใหม่การติดตาม ส่งข้อความการวินิจฉัยคำสั่ง dcp ไปยังเอาต์พุตมาตรฐาน  
ตรวจสอบแต่ละเป้าหมายก่อนที่จะรันคำสั่งรีโมตใดๆ บนเป้าหมาย หากเป้าหมายไม่ตอบสนอง การประมวลผลคำสั่งรีโมต  
สำหรับเป้าหมายจะถูกยกเลิก

-X *env\_list*

ข้ามตัวแปรสถานะแวดล้อมของคำสั่ง dcp อ็อปชันนี้ยอมรับอาร์กิวเมนต์ ซึ่งเป็นรายการของชื่อตัวแปรสถานะแวดล้อมที่ค้น  
ด้วย เครื่องหมายคอมมา ที่ต้องถูกข้าม หากไม่มี อาร์กิวเมนต์ในอ็อปชันนี้ หรืออาร์กิวเมนต์คือสตริงว่างเปล่า ตัวแปรสถานะ  
แวดล้อม dcp ทั้งหมดจะไม่ถูกยอมรับ

## คำอธิบาย

แสดงข้อมูลเวอร์ชันสำหรับตัวแปรสถานะแวดล้อมของคำสั่ง dcp

### DSH\_CONTEXT

ระบุบริบทที่พอลต์เพื่อใช้เพื่อแก้ไขเป้าหมาย ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -C

### DSH\_DEVICE\_LIST

ระบุไฟล์ที่มีรายการของอุปกรณ์เป้าหมาย ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -d ตัวแปรสถานะแวดล้อมนี้ถูกข้ามบน HMCs

### DCP\_DEVICE\_OPTS

ระบุอ็อปชันเพื่อใช้สำหรับคำสั่งรีโมตเซลล์ที่มีอุปกรณ์เป้าหมายเท่านั้น ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -O ตัวแปรสถานะแวดล้อมนี้ถูกข้ามบน HMCs

### DCP\_DEVICE\_RCP

ระบุพารามิเตอร์เพิ่มเติมของคำสั่ง remote copy ที่ใช้เพื่อคัดลอกไฟล์ไปยังหรือจากอุปกรณ์เป้าหมาย ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก --device-rcp ตัวแปรสถานะแวดล้อมนี้ถูกข้ามบน HMCs

### DSH\_FANOUT

ระบุค่า fanout ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -f

### DCP\_NODE\_OPTS

ระบุอ็อปชันเพื่อใช้สำหรับคำสั่งคัดลอกแบบรีโมตที่มีโหนดเป้าหมายเท่านั้น ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -o

### DCP\_NODE\_RCP

ระบุพารามิเตอร์เพิ่มเติมของคำสั่งรีโมตที่ใช้เพื่อคัดลอกไฟล์ไปยังและจากโหนดเป้าหมาย ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -r

### DSH\_NODE\_LIST

ระบุไฟล์ที่มีรายการของโหนดเป้าหมาย ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -g ตัวแปรนี้ถูกแทนด้วยตัวแปร WCOLL DSH\_NODEGROUP\_PATH ตัวแปร DSH\_NODE\_LIST ยังระบุรายการที่ค้นด้วยเครื่องหมายโคลอนของไดเรกทอรีที่มีไฟล์กลุ่มโหนดสำหรับบริบท dsh เมื่อระบุแฟล็ก -a ในบริบท dsh รายการของชื่อโหนดที่ไม่ซ้ำกัน จะถูกรวบรวมจากไฟล์กลุ่มโหนดทั้งหมดในพาท

### DSH\_TIMEOUT

ระบุเวลาเป็นวินาที เพื่อรอเอาต์พุตจากแต่ละเป้าหมาย แบบรีโมต ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -t

### RSYNC\_RSH

ตัวแปรสถานะแวดล้อม rsync ระบุเซลล์เซลล์ที่จะใช้เป็น transport สำหรับคำสั่ง rsync สถานะการออกและค่าการออกสำหรับแต่ละการเรียกใช้คำสั่ง remote copy จะแสดงในข้อความจากคำสั่ง dcp หากค่าการออกของคำสั่ง remote copy ไม่ใช่ศูนย์ โค้ดส่งคืนที่ไม่ใช่ค่าศูนย์ จากคำสั่ง remote copy บ่งชี้ว่า เกิดข้อผิดพลาดขึ้น ในระหว่าง remote copy หากคำสั่ง remote copy พบข้อผิดพลาด การเรียกทำงานของ remote copy บนเป้าหมายนั้นถูกส่งผ่าน โค้ดการออกของคำสั่ง dcp เป็น 0 หากคำสั่ง dcp รันโดยไม่มีข้อผิดพลาด และคำสั่ง remote copy ทั้งหมดเสร็จสิ้นโดยมีโค้ดการออกเป็น 0 หากเกิดข้อผิดพลาดคำสั่งภายใน dcp หรือคำสั่ง remote copy ไม่เสร็จสมบูรณ์ ค่าการออกของคำสั่ง dcp จะมากกว่า 0 ค่าการออกจะเพิ่มขึ้น 1 สำหรับอินสแตนซ์ของการเรียกใช้คำสั่ง remote copy ที่ตามมาที่ไม่สำเร็จ

ความปลอดภัย: คำสั่ง dcp ไม่มีข้อกำหนดเกี่ยวกับคอนฟิกูเรชันด้านความปลอดภัย ข้อกำหนดความปลอดภัยของคำสั่งรีโมตทั้งหมด (การกำหนดคอนฟิก การพิสูจน์ตัวตน และการให้สิทธิ์) จะถูกกำหนดโดย คำสั่งรีโมตที่สำคัญที่กำหนดค่าไว้สำหรับคำสั่ง dcp การพิสูจน์ตัวตนและการให้สิทธิ์ จะถูกสมมติให้กำหนดคอนฟิกูเรชันระหว่างโลคัลโฮสต์และเป้าหมายแบบรีโมต การพร้อมตัวรหัสนับแบบโต้ตอบไม่ได้รับการสนับสนุน การเรียกทำงานถูกส่งผ่าน และข้อผิดพลาดจะถูกแสดงสำหรับเป้าหมายแบบรีโมต หากการพร้อมตัวรหัสนับผ่านเกิดขึ้น คอนฟิกูเรชันความปลอดภัยตามที่เกี่ยวข้องกับสถานะแวดล้อมแบบรีโมต และคำสั่ง remote shell เป็นแบบผู้ใช้กำหนดเอง เมื่อกำหนดค่า /usr/bin/rcp เป็นคำสั่งรีโมตของคุณโดยใช้ Kerberos เวอร์ชัน 5 คุณต้องรันคำสั่ง kinit ของ Kerberos ก่อนเพื่อขอรับตั๋ว ticket-granting คุณยังต้องตรวจสอบให้แน่ใจว่าหลักการ Kerberos ของคุณอยู่ในไฟล์ .k5login ในรีโมตโฮมไดเรกทอรี user&csqg;s บนเป้าหมาย

## ตัวอย่าง

1. เมื่อต้องการคัดลอกไฟล์ /tmp/etc/hosts จาก โคลล์โฮสต์ไปยังไต่เร็กทอรี /etc บน node3, node4, node5 และไปยังผู้ใช้ gregb บน device16 ใน บริบท NIM ให้ป้อนคำสั่งต่อไปนี้:

```
dcp -n node3-node5 -d NIM:gregb@device16 /tmp/etc/hosts /etc:
```

2. เมื่อต้องการคัดลอกไฟล์ /etc/hosts จากโหนดที่จัดการในคลัสเตอร์ไปยังไต่เร็กทอรี /tmp/hosts.dir บนโคลล์โฮสต์ ให้ป้อนคำสั่งต่อไปนี้:

```
dcp -aP /etc/hosts /tmp/hosts.dir
```

ส่วนต่อท้ายที่ระบุชื่อของเป้าหมายจะถูกผนวกเข้ากับแต่ละชื่อไฟล์ เนื้อหาของไต่เร็กทอรี /tmp/hosts.dir จะดูคล้ายกับ:

```
hosts._node1                hosts._node4                hosts._node7
hosts._node2                hosts._node5                hosts._node8
hosts._node3                hosts._node6
```

3. เมื่อต้องการคัดลอกไต่เร็กทอรี /var/log/testlogdir จาก เป้าหมายทั้งหมดใน NodeGroup1 ในบริบท NIM และ DeviceGroup4 ในบริบท dsh โดยมี fanout ที่เป็น 12 และบันทึกแต่ละไต่เร็กทอรีบน โคลล์โฮสต์เป็น /var/log.\_target ให้ป้อน คำสั่งต่อไปนี้:

```
dcp -C DSH -N NIM:NodeGroup1 -D DeviceGroup 4 -f 12 \ -RP /var/log/testlogdir /var/log
```

4. เมื่อต้องการคัดลอก /localnode/smallfile และ /tmp/bigfile ไปยัง /tmp บน node1 โดยใช้คำสั่ง rsync ให้ป้อนคำสั่งต่อไปนี้:

```
RSYNC_RSH=/usr/bin/ssh; dcp -r /usr/bin/rsync -o "-z" \ -n node1 /localnode/smallfile /tmp/bigfile /tmp
```

คำสั่งนี้ใช้ ตัวแปรสถานะแวดล้อม rsync with the RSYNC\_RSH และแฟล็ก -z บน rsync

5. เมื่อต้องการคัดลอกไฟล์ /etc/hosts จากโคลล์โฮสต์ไปยังโหนดทั้งหมดในคลัสเตอร์ และข้ามตัวแปรสถานะแวดล้อม dcp ทั้งหมด ให้ป้อนคำสั่งต่อไปนี้:

```
dcp -X -a /etc/hosts /etc/hosts
```

6. เมื่อต้องการคัดลอกไฟล์ /etc/hosts จาก node1 และ node2 ไปยัง ไต่เร็กทอรี /tmp/hosts.dir บนโคลล์โฮสต์ และข้ามตัวแปรสถานะแวดล้อม dcp ทั้งหมดยกเว้น DCP\_NODE\_OPTS ให้ป้อนคำสั่งต่อไปนี้:

```
dcp -n node1,node2 -P -X 'DCP_NODE_OPTS' /etc/hosts /tmp/hosts.dir
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dsh” ในหน้า 208

---

## คำสั่ง dd

### วัตถุประสงค์

แปลงและคัดลอกไฟล์

## ไวยากรณ์

```
dd [ bs=BlockSize ][ cbs=BlockSize ]  
  
[ conv= [ ascii | block | ebc | dic | lib | m | unblock ]  
  
[ l | case | ucase ] [ i | block ]  
  
[ noerror ] [ swab ] [ sync ]  
  
[ oblock ] [ notrunc ] [ count=  
InputBlocks ] [ files=InputFiles ] [ fskip=  
SkipEOFs ] [ ibs=InputBlockSize ] [ if=  
InFile ] [ obs=OutputBlockSize ] [ of=  
OutFile ] [ seek=RecordNumber ] [ skip=  
SkipInputBlocks ] [ span=yes|no ]  
  
dd [ Option=Value ]
```

## คำอธิบาย

คำสั่ง **dd** อ่านพารามิเตอร์ *InFile* หรืออินพุตมาตรฐาน ทำการแปลงที่ระบุ จากนั้นคัดลอกข้อมูลที่แปลงไปที่พารามิเตอร์ *OutFile* หรือเอาต์พุตมาตรฐาน ขนาดบล็อกอินพุตและเอาต์พุต สามารถถูก ระบุเพื่อใช้ประโยชน์จาก raw physical I/O

หมายเหตุ: คำว่า *บล็อก* หมายถึงปริมาณของการอ่านหรือเขียนข้อมูล โดยคำสั่ง **dd** ในหนึ่งการดำเนินการและไม่จำเป็นต้อง มีขนาดเท่ากับบล็อกของดิสก์

ซึ่งมีการระบุขนาดจำนวนไบต์ที่ต้องการ ตัวเลขลงท้ายด้วย **w**, **b** หรือ **k** ระบุการคูณโดย 2, 512 หรือ 1024 ตามลำดับ คู่ของ ตัวเลข แยกกันโดย **x** หรือ **\*** (เครื่องหมายดอกจัน) หมายถึงผลคูณ พารามิเตอร์ **count** ต้องการจำนวน บล็อก *ไม่ใช่* จำนวนไบต์ ที่จะถูกคัดลอก

การแม่พชุดอักขระที่สัมพันธ์กับแฟล็ก **conv=ascii** และ **conv=ebcdic** คือการดำเนินการเพื่อให้ครบสมบูรณ์ แฟล็กเหล่านี้แม่พ ระหว่างอักขระ ASCII และเซตย่อยของอักขระ EBCDIC ที่พบใน เวิร์กสเตชันและเครื่องเจาะบัตรส่วนใหญ่

ใช้ค่าพารามิเตอร์ **cbs** ถ้ามีการระบุการแปลง **block**, **unblock**, **ascii**, **ebcdic** หรือ **ibm** ถ้ามีการระบุพารามิเตอร์ **unblock** หรือ **ascii** คำสั่ง **dd** จะทำการแปลง ความยาวคงที่ไปเป็นความยาวผันแปรได้ มิฉะนั้น จะทำการแปลงจากความยาวผันแปรได้ไป เป็นความยาวคงที่ พารามิเตอร์ **cbs** กำหนดเป็นความยาวคงที่

**ข้อควรสนใจ:** ถ้ามีการระบุค่าพารามิเตอร์ **cbs** ที่น้อยกว่า อินพุตบล็อกที่เล็กที่สุด บล็อกที่แปลงจะถูกตัดทอน

หลังจากเสร็จสิ้น คำสั่ง **dd** รายงานจำนวน บล็อกอินพุตและเอาต์พุตทั้งหมดและบางส่วน

## หมายเหตุ:

1. โดยปกติ คุณต้องการเพียงสิทธิเข้าถึงในการเขียนไปที่ไฟล์เอาต์พุต อย่างไรก็ตาม เมื่อ ไฟล์เอาต์พุตไม่ใช่อุปกรณ์ การแอดเดสโดยตรง และคุณใช้แฟล็ก `seek` คุณต้องมีสิทธิเข้าถึงเพื่ออ่านไฟล์ด้วย
2. คำสั่ง `dd` แทรกอักขระขึ้นบรรทัดใหม่เฉพาะ เมื่อทำการแปลงด้วยชุดแฟล็ก `conv=ascii` หรือ `conv=unblock`; จะมีการเสริมเฉพาะเมื่อทำการแปลง ด้วยชุดแฟล็ก `conv=ebcdic`, `conv=ibm` หรือ `conv=block`
3. ใช้คำสั่ง `backup`, `tar` หรือ `cpio` แทนคำสั่ง `dd` เมื่อเป็นไปได้ในการคัดลอกไฟล์ไปที่เทป คำสั่งเหล่านี้ถูกออกแบบมาเพื่อใช้กับอุปกรณ์เทป สำหรับข้อมูลเพิ่มเติมสำหรับการใช้อุปกรณ์เทป เพื่อดูไฟล์พิเศษ `rmt`
4. ค่าขนาดบล็อกที่ระบุด้วยแฟล็ก `bs`, `ibs` และ `obs` ต้องเป็นจำนวนการคูณ ของขนาดบล็อกฟิสิกส์สำหรับสื่อที่จะนำมาใช้
5. เมื่อแฟล็ก `conv=sync` ถูกระบุคำสั่ง `dd` เสริมส่วนบล็อกอินพุต ด้วยค่าว่าง ดังนั้นคำสั่ง `dd` แทรกค่าว่างลงใน ส่วนกลางของ กระแสข้อมูล ถ้ามีการอ่านข้อมูลที่มิได้รับบล็อกข้อมูล สมบูรณ์ (ตามที่ระบุโดยแฟล็ก `ibs`) ซึ่ง เกิดขึ้นตามปกติเมื่ออ่าน ข้อมูลจากไฟฟ์
6. ถ้าแฟล็ก `bs` ถูกระบุโดยตัวเองและไม่มีการระบุการแปลง อื่นนอกจาก `sync`, `noerror` หรือ `notrunc` ข้อมูลจากแต่ละอินพุต บล็อก จะถูกเขียนเป็นเอาต์พุตบล็อกแยก ถ้าการอ่านข้อมูลส่งคืนค่าน้อยกว่า บล็อกเต็ม และไม่มีการระบุ `sync` เอาต์พุต บล็อก ที่เป็นผลลัพธ์จะมีขนาดเท่ากับอินพุตบล็อก ถ้าไม่มีการระบุแฟล็ก `bs` หรือมีการระบุการแปลงข้อมูลนอกจาก `sync`, `noerror` หรือ `notrunc` อินพุตจะถูกประมวลผลและเก็บไว้ในเอาต์พุตบล็อก ขนาดเต็ม จนกว่าจะถึงจุดสุดท้ายของอินพุต

## การขยายข้ามอุปกรณ์

`dd` สามารถถูกสั่งงานให้ขยายข้ามอุปกรณ์ ถ้า อินพุตไฟล์มีขนาดใหญ่กว่าขนาดฟิสิกส์ของอุปกรณ์เอาต์พุต

หมายเหตุ: ต้องใช้ความระมัดระวัง เมื่อทำการระบุขนาดบล็อก `bs` เป็นจำนวนคูณของขนาดฟิสิกส์ของอุปกรณ์ เนื่องจาก ขนาดบล็อกที่ไม่ถูกต้องจะมีผลให้ข้อมูลมีความขัดแย้ง หรือซ้อนทับกัน

การขยายของ `dd` ข้ามอุปกรณ์จะไม่เกิดขึ้น ถ้าหนึ่งในพารามิเตอร์ `InFile` หรือ `OutFile` เป็น `stdin` หรือ `stdout`

การขยายจะเกิดขึ้นในแบบที่ `dd` จะพร้อมต์ ข้อมูลที่ติดไประหว่างการเขียน ถ้าอุปกรณ์เอาต์พุตเต็ม ระหว่างอ่านจาก อุปกรณ์อินพุต `dd` จะพร้อมต์ข้อมูลติดไปถ้า ข้อมูลถูกอ่านอย่างสมบูรณ์จากอุปกรณ์อินพุตแม้เมื่ออุปกรณ์ยังไม่ถึงจุดสิ้นสุด ในกรณีนี้ อาจจำเป็นต้องกด 'n' เพื่อจบการทำงาน

## แฟล็ก

### ไอเท็ม

`bs=BlockSize`

`cbs=BlockSize`

`count=InputBlocks`

### คำอธิบาย

ระบุทั้งขนาดบล็อกอินพุตและเอาต์พุต แทนแฟล็ก `ibs` และ `obs` ค่าขนาดบล็อกที่ระบุด้วยแฟล็ก `bs` ต้องเป็น ผลคูณของขนาดบล็อกฟิสิกส์ สำหรับสื่อบันทึกที่ถูกใช้

ระบุขนาดบล็อกการแปลงสำหรับความยาวผันแปรได้ไปเป็นความยาวคงที่ และการแปลงความยาวคงที่ไป เป็นความยาวผันแปรได้ เช่น `conv=block`

คัดลอกเฉพาะบล็อกอินพุตที่ระบุโดยตัวแปร `InputBlocks`

## ไอเท็ม

conv= Conversion,....

## คำอธิบาย

ระบุอ็อปชันการแปลงหนึ่งอ็อปชันหรือมากกว่านั้น การแปลงแต่ละ การแปลงควรถูกแยกโดยคอมมา รายการต่อไปนี้อธิบายอ็อปชันที่เป็นไปได้:

**ascii** แปลง EBCDIC ไปเป็น ASCII อ็อปชันนี้ทำงานร่วมกันไม่ได้กับอ็อปชัน **ebcdic**, **ibm**, **block** และ **unblock**

**block** แปลงเร็กคอร์ดความยาวผันแปรได้ไปเป็นความยาวคงที่ ความยาวถูกกำหนด โดย conversion block size (cbs) อ็อปชันนี้ทำงานร่วมกันไม่ได้กับอ็อปชัน **ascii**, **ebcdic**, **ibm** และ **unblock**

**ebcdic** แปลง ASCII ไปเป็น EBCDIC มาตรฐาน อ็อปชันนี้ทำงานร่วมกันไม่ได้กับอ็อปชัน **ascii**, **ibm**, **block** และ **unblock**

**ibm** แปลง ASCII ไปเป็น IBM® เวอร์ชัน EBCDIC อ็อปชันนี้ทำงานร่วมกันไม่ได้กับอ็อปชัน **ascii**, **ebcdic**, **block** และ **unblock**

## iblock, oblock

ลดการสูญเสียข้อมูลให้น้อยที่สุด จากข้อผิดพลาดในการอ่านหรือเขียนข้อมูล บนอุปกรณ์ การแอ็คเซสโดยตรง ถ้าคุณระบุตัวแปร **iblock** และมีข้อผิดพลาดเกิดขึ้นระหว่างการอ่านบล็อก (โดยที่ขนาดบล็อกคือ 512 หรือ ขนาดที่ระบุโดยตัวแปร **ibs=InputBlockSize**) คำสั่ง **dd** พยายามอ่านซ้ำบล็อกข้อมูลในหน่วยที่มีขนาดเล็กกว่า ถ้าคำสั่ง **dd** สามารถกำหนดขนาดเช็ค เตอร์ของอุปกรณ์อินพุต คำสั่งจะอ่าน บล็อกที่เสียหายหนึ่งเช็คเตอร์ต่อครั้ง มิฉะนั้น จะอ่าน 512 ไบต์ ต่อครั้ง ขนาดบล็อกอินพุต (**ibs**) ต้องเป็นผลคูณ ของขนาดการทำซ้ำนี้ อ็อปชันนี้มีข้อมูล เสียหายที่สัมพันธ์กับข้อผิดพลาดการอ่านข้อมูล ไปที่เช็คเตอร์เดียว การแปลง **oblock** ทำงาน เหมือนกันในเอาต์พุต

**lcase** เปลี่ยนอักขระแบบตัวอักษรทั้งหมดเป็นตัวพิมพ์เล็ก

**noerror** ไม่หยุดการประมวลผลเมื่อมีข้อผิดพลาด

**notrunc** ไม่ตัดทอนไฟล์เอาต์พุต แต่บล็อกที่ไม่ได้มีการ เขียนข้อมูลไปที่เอาต์พุตอย่างชัดเจน จะถูกรักษาไว้

**ucase** เปลี่ยนอักขระแบบตัวอักษรทั้งหมดเป็นตัวพิมพ์ใหญ่

**swab** สลับคำคู่ไบต์ทั้งหมด

**sync** ขยายอินพุตบล็อกทั้งหมดไปเป็นค่า **ibs**

**unblock** แปลงบล็อกความยาวคงที่ไปเป็นความยาวผันแปรได้ ความยาวถูกกำหนด โดย conversion block size (cbs) อ็อปชันนี้ทำงานร่วมกันไม่ได้กับอ็อปชัน **ascii**, **ebcdic**, **ibm** และ **block**

**files=InputFiles**

คัดลอกไฟล์ตามจำนวนที่ระบุโดยค่าตัวแปร **InputFiles** ของอินพุตไฟล์ ก่อนสิ้นสุด (โดยเมื่ออินพุต เป็นเทป แม่เหล็กหรืออุปกรณ์ลักษณะเดียวกันเท่านั้น)

**fskip=SkipEOFs**

ข้ามอักขระสิ้นสุดไฟล์ตามจำนวนที่ระบุโดยตัวแปร **SkipEOFs** ก่อนเริ่มต้นการคัดลอก ตัวแปร **SkipEOFs** นี้มีประโยชน์สำหรับการจัดตำแหน่งบนเทปแม่เหล็ก multifile

**ibs=InputBlockSize**

ระบุขนาดอินพุตบล็อก ค่าดีฟอลต์คือ 512 ไบต์หรือหนึ่งบล็อก ค่าขนาดบล็อกที่ระบุด้วยแฟล็ก **ibs** ต้องมีค่า เป็นจำนวนการคูณของขนาดบล็อกฟิลิซัลสำหรับสื่อบันทึกที่จะนำมาใช้

**if=InFile**

ระบุชื่อไฟล์อินพุต ค่าดีฟอลต์คืออินพุตมาตรฐาน

**obs=OutputBlockSize**

ระบุขนาดเอาต์พุตบล็อก ค่าดีฟอลต์คือ 512 ไบต์หรือหนึ่งบล็อก ค่าขนาดบล็อกที่ระบุด้วยแฟล็ก **obs** ต้อง เป็นผลคูณของขนาดบล็อกฟิลิซัล สำหรับสื่อบันทึกที่ถูกใช้

**of=OutFile**

ระบุชื่อไฟล์เอาต์พุต ค่าดีฟอลต์คือเอาต์พุตมาตรฐาน

**seek=RecordNumber**

ค้นหาเร็กคอร์ดที่ระบุโดยตัวแปร **RecordNumber** จากจุดเริ่มต้นของไฟล์อินพุตก่อนการคัดลอก

**skip=SkipInputBlocks**

ข้ามค่า **SkipInputBlocks** ที่ระบุของบล็อกอินพุต ก่อนเริ่มการคัดลอก

**span=yes|no**

อนุญาตการขยายอุปกรณ์ ถ้าระบุ **yes** และทำงานตามดีฟอลต์ถ้าระบุเป็น **no** ดูที่ การขยายข้ามอุปกรณ์ สำหรับข้อมูลเพิ่มเติม..

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

Item	คำอธิบาย
0	ไฟล์อินพุตถูกคัดลอกเสร็จสมบูรณ์
>0	มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการแปลงไฟล์ข้อความ ASCII ไปเป็น EBCDIC ให้พิมพ์:

```
dd if=text.ascii of=text.ebcdic conv=ebcdic
```

คำสั่งนี้แปลงไฟล์ text.ascii ไปเป็นแบบ EBCDIC เก็บข้อมูลเป็นเวอร์ชัน EBCDIC ในไฟล์ text.ebcdic

**หมายเหตุ:** เมื่อคุณระบุพารามิเตอร์ `conv=ebcdic` คำสั่ง `dd` แปลงอักขระ ASCII <sup>^</sup>(circumflex) ไปเป็นอักขระ EBCDIC ที่ไม่ได้ใช้ (เลขฐานสิบหก 9A) และ ASCII <sup>~</sup>(tilde) ไปเป็น EBCDIC <sup>^</sup>(สัญลักษณ์ NOT)

2. เมื่อต้องการแปลงไฟล์ ASCII เร็กคอร์ดความยาวผันแปรได้ `/etc/passwd` ไปเป็นไฟล์เร็กคอร์ด EBCDIC ความยาวคงที่ 132 ไบต์ พิมพ์:

```
dd if=/etc/passwd cbs=132 conv=ebcdic of=/tmp/passwd.ebcdic
```

3. เมื่อต้องการแปลงไฟล์ 132-byte-per-record EBCDIC ไปเป็นบรรทัด ASCII ความยาวผันแปรได้ในแบบตัวพิมพ์เล็ก พิมพ์:

```
dd if=/tmp/passwd.ebcdic cbs=132 conv=ascii of=/tmp/passwd.ascii
```

4. เมื่อต้องการแปลงไฟล์ ASCII เร็กคอร์ดความยาวผันแปรได้ `/etc/passwd` ไปเป็นไฟล์เร็กคอร์ดความยาวคงที่ 132-ไบต์ใน IBM version of EBCDIC, พิมพ์:

```
dd if=/etc/passwd cbs=132 conv=ibm of=/tmp/passwd.ibm
```

5. เมื่อต้องการคัดลอกบล็อกจากเทปที่มี 1KB บล็อกไปที่เทปอื่นโดยใช้บล็อก 2KB พิมพ์:

```
dd if=/dev/rmt0 ibs=1024 obs=2048 of=/dev/rmt1
```

6. เมื่อต้องการใช้คำสั่ง `dd` เป็นตัวกรอง พิมพ์:

```
ls -l | dd conv=ucase
```

คำสั่งนี้แสดงรายการแบบยาวของไดเรกทอรีปัจจุบันเป็นตัวพิมพ์ใหญ่

**หมายเหตุ:** ผลการทำงานของคำสั่ง `dd` และคำสั่ง `cpio` ไปที่ 9348 Magnetic Tape Unit Model 12 สามารถเพิ่มประสิทธิภาพได้โดยการเปลี่ยนขนาดบล็อกดีฟอลต์ เมื่อต้องการเปลี่ยนขนาด บล็อก ให้ใช้คำสั่ง `chdev` ดังนี้:

```
chdev -l Device_name -a block_size=32k
```

7. เมื่อต้องการถ่ายโอนไปที่ดิสเก็ต 3.5-นิ้ว 1.4MB อย่างมีประสิทธิภาพโดยใช้ 36 บล็อก ขนาด 512 ไบต์ พิมพ์:

```
dd if=Filename of=/dev/rfd0 bs=36b conv=sync
```

คำสั่งนี้เขียนค่าของพารามิเตอร์ *Filename* ไปที่ อุปกรณ์ดิสก์หนึ่ง ไชลินเดอร์ต่อครั้ง `conv=sync` จำเป็นเมื่อมีการอ่าน จากดิสก์ และเมื่อขนาดไฟล์ไม่ได้ เป็นจำนวนคูณของขนาดบล็อกของดิสก์เกิด อย่าทำดังนี้ ถ้าอินพุตไปที่คำสั่ง `dd` คือ ไฟล์ไม่ใช่ไฟล์ จะมีการขยายอินพุต ส่วนใหญ่ด้วยค่าว่าง ไม่เพียงแค่บล็อกสุดท้าย

8. เมื่อต้องการคัดลอกบล็อกจากไฟล์อินพุตที่มีขนาดบล็อกตั้งเป็น 720b บล็อกลงใน ดิสก์เกิดขนาด 1.44MB พิมพ์:

```
dd if=testfile of=/dev/fd0 bs=720b conv=sync
```

หมายเหตุ: ถ้าไฟล์อินพุตมีขนาดใหญ่มากกว่าขนาดฟิสิกส์ ของอุปกรณ์เอาต์พุตแล้ว `dd` จะพร้อมท์เพื่อขออุปกรณ์อื่น

9. เมื่อต้องการคัดลอกบล็อกจากไฟล์อินพุตที่มีขนาดบล็อกตั้งเป็น 32k ไปที่ เทป พิมพ์:

```
dd if=inputfile of=/dev/rmt0 bs=32k conv=sync
```

10. เมื่อต้องการคัดลอกข้อมูลจากเทปไปที่ไฟล์ในไดเรกทอรีปัจจุบันที่มี ขนาดบล็อกตั้งเป็น 32k บล็อก ให้พิมพ์ดังนี้:

```
dd if=/dev/rmt0 of=outfile bs=32k conv=sync
```

11. เมื่อต้องการคัดลอกบล็อกจากไฟล์อินพุตที่มีขนาดบล็อกตั้งเป็น 720b, ลงใน ดิสก์เกิดขนาด 1.44MB ให้ป้อน:

```
dd if=testfile of=/dev/fd0 bs=720b conv=sync span=yes
```

หมายเหตุ: ถ้าไฟล์อินพุตมีขนาดใหญ่มากกว่าขนาดฟิสิกส์ อุปกรณ์เอาต์พุตแล้ว `dd` จะพร้อมท์เพื่อขอ อุปกรณ์อื่น

12. เมื่อต้องการคัดลอกบล็อกจากไฟล์อินพุตที่มีขนาดบล็อกตั้งเป็น 32k ไปที่เทปให้ป้อน:

```
dd if=inputfile of=/dev/rmt0 bs=32k conv=sync span=yes
```

13. เมื่อต้องการคัดลอกบล็อกข้อมูลจากเทปที่มีขนาดบล็อกตั้งเป็น 32k ไปที่ไฟล์ในไดเรกทอรีปัจจุบันให้ป้อน:

```
dd if=/dev/rmt0 of=outfile bs=32k conv=sync span=yes
```

## ไฟล์

Item	คำอธิบาย
/usr/bin/dd	มีคำสั่ง <code>dd</code>

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `cp`

คำสั่ง `tar`

การสำรองข้อมูลระบบ

ไฟล์ระบบปฏิบัติการ

---

## เมธอด defif

### วัตถุประสงค์

กำหนดเน็ตเวิร์กอินเตอร์เฟสในฐานข้อมูลคอนฟิกูเรชัน

### ไวยากรณ์

```
defif[ -c Class -s Subclass ]-t Type
```

## คำอธิบาย

เมธอด `defif` กำหนด อินสแตนซ์ที่ระบุของเน็ตเวิร์กอินเตอร์เฟซ โดยกำหนดเฉพาะอินเตอร์เฟซสำหรับ อะแดปเตอร์ที่ตั้งค่าอยู่ในปัจจุบัน เมื่อต้องการกำหนดอินสแตนซ์ที่ระบุ เมธอด `defif` กระทำดังต่อไปนี้:

1. สร้างอินสแตนซ์อินเตอร์เฟซแบบกำหนดเองในฐานข้อมูลคอนฟิกูเรชัน
2. สืบทอดชื่อโลจิคัลของอินสแตนซ์อินเตอร์เฟซ
3. เรียกข้อมูลแอตทริบิวต์ที่กำหนดไว้แล้ว
4. อัปเดตคลาสอ็อบเจกต์ Customized Dependency เพื่อสะท้อนถึงความเกี่ยวข้องกันของอินสแตนซ์อินเตอร์เฟซ
5. ตั้งค่าแฟล็กสถานะของอินสแตนซ์ของอินเตอร์เฟซ เป็น `defined`

## แฟล็ก

### ไอเท็ม

-c Class

-s Subclass

### คำอธิบาย

ระบุนามอินสแตนซ์อินเตอร์เฟซที่จะถูกกำหนด ค่าที่ใช้ได้คือ `if`

ระบุนามย่อยของอินสแตนซ์อินเตอร์เฟซที่จะถูกกำหนด ค่าที่ใช้ได้คือ:

**TR** Token-ring

**TH** Ethernet

**SL** Slip

**XT** X.25

**LO** Loopback

-t Type

ระบุชนิดของอินสแตนซ์อินเตอร์เฟซที่จะถูกกำหนด ค่าที่ใช้ได้คือ:

**tr** Token-ring

**en** Ethernet

**sl** Slip

**ie3** IEEE 802.3 Ethernet

**lo** Loopback

**xt** X.25

## ตัวอย่าง

เมื่อต้องการกำหนดอินสแตนซ์อินเตอร์เฟซเน็ตเวิร์กโทเค็นริง, ให้ป้อนเมธอดในรูปแบบต่อไปนี้:

```
defif -t tr
```

ข้อมูลที่เกี่ยวข้อง:

รูทีนย่อย `odm_run_method`

อินเตอร์เฟซเครือข่าย TCP/IP

Object Data Manager

การเขียนเมธอดอุปกรณ์

---

## เมธอด definet

### วัตถุประสงค์

กำหนดอินสแตนซ์ inet ในฐานข้อมูลคอนฟิกูเรชัน ระบบ

### ไวยากรณ์

```
definet [ -c Class]
```

### คำอธิบาย

เมธอด definet สร้างอ็อบเจกต์ในฐานข้อมูลคอนฟิกูเรชัน ODM ระบุแอตทริบิวต์ที่กำหนดเองของอินสแตนซ์ inet ซึ่งดำเนินการต่อไปนี้:

1. สร้างอินสแตนซ์ inet แบบกำหนดเอง
2. ตั้งค่าแฟล็กสถานะของอินสแตนซ์ inet เป็น defined

เมธอดนี้ถูกเรียกโดยคำสั่งระดับสูง mkdev และไม่ได้ออกแบบเพื่อให้เรียกใช้จากบรรทัดคำสั่ง

หมายเหตุ: เมธอด definet เป็นเครื่องมือโปรแกรมมิ่งและไม่ควรถูกเรียกใช้งานจากบรรทัดรับคำสั่ง

### แฟล็ก

ไอเท็ม

-c Class

คำอธิบาย

ระบุอินสแตนซ์ inet ที่จะถูกกำหนด ค่าที่ถูกต้องสำหรับตัวแปร Class คือ tcpip เท่านั้น

### ตัวอย่าง

เมื่อต้องการกำหนดอินสแตนซ์ inet0, เรียกเมธอดต่อไปนี้:

```
definet
```

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mkdev

รูทีนย่อย odm\_run\_method

Object Data Manager

การเขียนเมธอดอุปกรณ์

---

## คำสั่ง defragfs

### วัตถุประสงค์

เพิ่มพื้นที่ว่างที่ต่อเนื่องกันของระบบไฟล์

## ไวยากรณ์

```
defragfs [-q|-r|-s] { Device | FileSystem }
```

## คำอธิบาย

คำสั่ง `defragfs` เพิ่มพื้นที่ว่างที่ต่อเนื่องกัน ของระบบไฟล์ โดยการจัดการการจัดสรรพื้นที่ให้ต่อเนื่องกัน แทนการที่พื้นที่ว่างกระจัดไปทั่วติสก์ ระบบไฟล์ที่จะถูกจัดเรียง สามารถถูกระบุด้วยตัวแปร `Device` ซึ่ง คือชื่อพารามิเตอร์ของวอลุ่มสำหรับ (ตัวอย่าง `/dev/hd4`) และยังสามารถถูกระบุได้ด้วยตัวแปร `FileSystem` ซึ่งคือจุดประกอบเข้าไปในไฟล์ `/etc/filesystems`

คำสั่ง `defragfs` มีไว้สำหรับ ระบบไฟล์ที่กระจายตัวและที่มีการบีบอัด อย่างไรก็ตาม คุณสามารถใช้คำสั่ง `defragfs` ในการเพิ่มพื้นที่ว่างที่ต่อเนื่องกันในระบบไฟล์ที่ไม่กระจายตัว

คุณต้อง mount การอ่าน-การเขียน ของระบบไฟล์ สำหรับคำสั่งนี้เพื่อให้รันได้เสร็จสมบูรณ์ การใช้แฟล็ก `-q` แฟล็ก `-r` หรือแฟล็ก `-s` สร้าง รายงานการแตกแฟรกเมนต์ แฟล็กเหล่านี้ไม่เปลี่ยนแปลงระบบไฟล์

คำสั่ง `defragfs` ชำกว่าเมื่อเทียบกับระบบไฟล์ JFS2 ที่มี snapshot เนื่องจากจำนวนข้อมูลที่ต้องถูกคัดลอกลงใน อ็อบเจกต์หน่วยเก็บ snapshot คำสั่ง `defragfs` ส่งข้อความคำเตือน ถ้ามี snapshots คำสั่ง `snapshot` สามารถนำมาใช้เพื่อลบ snapshots แล้วใช้อีกครั้งเพื่อสร้าง snapshot ใหม่ หลังจากที่คุณคำสั่ง `defragfs` สมบูรณ์

## แฟล็ก

ไอเท็ม คำอธิบาย

- `-q` รายงานสถานะปัจจุบันของระบบไฟล์
- `-r` รายงานสถานะปัจจุบันของระบบไฟล์และสถานะที่จะมีผลถ้าคำสั่ง `defragfs` ถูกรันโดยไม่มีแฟล็ก `-q, -r` หรือ `-s`
- `-s` รายงานการแตกแฟรกเมนต์ในระบบไฟล์ อ็อบชันนี้ ทำให้ `defragfs` ส่งผ่านข้อมูล meta ในระบบไฟล์ ซึ่งอาจมีผลให้ผลการทำงานลดลง

## เอาต์พุต

บนระบบ JFS นิยามสำหรับข้อความที่รายงานโดยคำสั่ง `defragfs` มีดังนี้:

จำนวนของแฟรกเมนต์ว่าง

จำนวนของแฟรกเมนต์ในระบบไฟล์

จำนวนของแฟรกเมนต์ที่จัดสรร

จำนวนของแฟรกเมนต์ที่จัดสรรในระบบไฟล์

จำนวนของพื้นที่ว่างที่น้อยกว่าบล็อก

จำนวนของพื้นที่ว่างภายในระบบไฟล์ที่เล็กกว่า บล็อก พื้นที่ว่างคือชุดของแฟรกเมนต์ที่ต่อเนื่องกัน ที่ไม่ถูกจัดสรร

จำนวนของแฟรกเมนต์ที่ว่างในพื้นที่ว่างขนาดเล็ก

จำนวนรวมของแฟรกเมนต์ในพื้นที่ว่างขนาดเล็กทั้งหมด พื้นที่ว่าง ขนาดเล็กคือพื้นที่ที่เล็กกว่าบล็อก

จำนวนแฟรกเมนต์ที่ย้าย

จำนวนรวมแฟรกเมนต์ที่ย้าย

Number of logical blocks moved

จำนวนรวมของบล็อกโลจิคัลที่ย้าย

## จำนวนการพยายามจัดสรร

จำนวนครั้งที่แฟรกเมนต์ว่างถูกจัดสรรใหม่

## จำนวนแน่นอนที่ตรงกัน

จำนวนครั้งที่แฟรกเมนต์ถูกย้ายจะพอดีกับ พื้นที่ว่างบางส่วน

## จำนวนรวมของแฟรกเมนต์

จำนวนรวมของแฟรกเมนต์ในระบบไฟล์

## จำนวนแฟรกเมนต์ที่อาจถูกโอนย้าย

จำนวนแฟรกเมนต์ที่อาจถูกย้ายระหว่างการรวมแฟรกเมนต์

## ระบบไฟล์ FileSystem เป็นเปอร์เซ็นต์ n ที่เป็นแฟรกเมนต์

แสดงขนาดระบบไฟล์ที่เป็นแฟรกเมนต์ในรูปเปอร์เซ็นต์

บนระบบไฟล์ JFS2 นิยามสำหรับข้อความที่รายงานโดยคำสั่ง defragfs มีดังนี้:

## ผลรวมกลุ่มการจัดสรร

จำนวนของกลุ่มการจัดสรรในระบบไฟล์ กลุ่มการจัดสรรแบ่ง พื้นที่บนระบบไฟล์ออกเป็น chunks กลุ่มการจัดสรรอนุญาตให้นโยบายการจัดสรรรีซอร์ส JFS2 ใช้เมธอดที่รู้จักดีในการสร้างผลการทำงาน I/O ที่ดี

## กลุ่มการจัดสรรที่ลดแฟรกเมนต์แล้ว

จำนวนของกลุ่มการจัดสรรที่ถูกลดการเป็นแฟรกเมนต์แล้ว

## กลุ่มการจัดสรรถูกข้ามไป - ว่างทั้งหมด

จำนวนของกลุ่มการจัดสรรที่ถูกข้ามไปเนื่องจาก เป็นพื้นที่ว่างทั้งหมด

## กลุ่มการจัดสรรถูกข้ามไป - มีบล็อกที่ว่างอยู่น้อยเกินไป

จำนวนของกลุ่มการจัดสรรที่ถูกข้ามไปเนื่องจากมีบล็อกที่ว่างอยู่น้อยเกินไปในกลุ่ม เพื่อจะทำการจัดสรร

## กลุ่มการจัดสรรถูกข้ามไป - มีพื้นที่ว่างที่ต่อเนื่องกัน ขนาดใหญ่

จำนวนของกลุ่มการจัดสรรที่ถูกข้ามไปเนื่องจากมีพื้นที่ว่างที่ต่อเนื่องกันขนาดใหญ่ ซึ่งไม่เหมาะที่จะทำการลดแฟรกเมนต์

## กลุ่มการจัดสรรเป็นตัวแทนสำหรับการรวมแฟรกเมนต์

จำนวนของกลุ่มการจัดสรรที่พอดีกับการลดแฟรกเมนต์

## จำนวนของฟรีรันเนลีย์ในกลุ่มการจัดสรรตัวแทน

จำนวนเฉลี่ยของฟรีรันเนลีย์ต่อกลุ่มการจัดสรร สำหรับกลุ่มการจัดสรร ที่พบว่าพอดีกับการลดแฟรกเมนต์ ฟรีรันเนลีย์คือชุดต่อเนื่องของ บล็อกซึ่งไม่ได้ถูกจัดสรร

## จำนวนรวมของบล็อก

จำนวนรวมของบล็อกในระบบไฟล์

## จำนวนบล็อกที่อาจถูกโอนย้าย

จำนวนบล็อกที่อาจถูกย้ายระหว่างการรวมแฟรกเมนต์

## ระบบไฟล์ FileSystem เป็นเปอร์เซ็นต์ n ที่เป็นแฟรกเมนต์

แสดงขนาดระบบไฟล์ที่เป็นแฟรกเมนต์ในรูปเปอร์เซ็นต์

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

### ตัวอย่าง

1. เมื่อต้องการรวมแฟร็กเมนต์ระบบไฟล์ **/data1** ที่อยู่บนโลจิคัลวอลุ่ม **/dev/lv00**, ให้ป้อน:

```
defragfs /data1
```

2. เมื่อต้องการรวมแฟร็กเมนต์ระบบไฟล์ **/data1** โดยการระบุจุดประกอบเข้า ให้ป้อน:

```
defragfs /data1
```

3. เมื่อต้องการสร้างรายงานบนระบบไฟล์ **/data1** ที่ระบุสถานะปัจจุบันเช่นเดียวกับสถานะ หลังจากถูกรวมแฟร็กเมนต์ ให้ป้อน:

```
defragfs -r /data1
```

4. เมื่อต้องการสร้างรายงานเกี่ยวกับการแตกแฟร็กเมนต์ในระบบไฟล์ **/data1** ให้ป้อน:

```
defragfs -s /data1
```

### ไฟล์

ไอเท็ม

/etc/filesystems

คำอธิบาย

แสดงรายการของระบบไฟล์ที่รู้จัก และกำหนดคุณสมบัติไว้

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `crfs`

คำสั่ง `lsfs`

การบีบอัดข้อมูล JFS

หมายเลขตัวแปรของไอ-โหนด

---

## คำสั่ง `defvsd`

### วัตถุประสงค์

กำหนดโหนดใหม่หรือใช้ดิสก์ที่แบ่งใช้แบบเสมือน

### ไวยากรณ์

```
defvsd logical_volume_name global_group_name vsd_name
```

## คำอธิบาย

คำสั่งนี้ถูกรันเพื่อระบุโลจิกัลวอลุ่มที่อยู่ในกลุ่มวอลุ่มที่เข้าถึงได้ แบบโกลบอล เพื่อใช้เป็นดิสก์ที่แบ่งใช้เสมือน

คุณสามารถใช้ System Management Interface Tool (SMIT) ในการรันคำสั่ง `defvsd` ในการใช้ SMIT ให้ป้อน:

```
smit vsd_data
```

และเลือกอีพซัน **Define a Virtual Shared Disk**

## แฟล็ก

**-r** รีเซ็ทหมายเลขลำดับ ขาออกและที่ต้องการ สำหรับโหนดที่ระบุ บนโหนดซึ่งคำสั่งถูกรัน ใช้แฟล็กนี้ เมื่อโหนดอื่นถูกรีบูต cast out หรือดิสก์ที่แบ่งใช้เสมือนได้ถูกกำหนดค่าใหม่ บนโหนดนั้น โหนดที่ระบุยังมีการ cast in

**หมายเหตุ:** อีพซันนี้ควรถูกใช้เฉพาะภายใต้คำแนะนำโดยตรงจาก IBM Service. ไม่ควรนำมาใช้ภายใต้สภาวะปกติ

**-R** รีเซ็ทหมายเลขลำดับ ขาออกและที่ต้องการ สำหรับโหนดทั้งหมดบน โหนดซึ่งคำสั่งถูกรัน ใช้แฟล็กนี้หลังจากการรีบูตโหนด โหนดทั้งหมดในเครือข่ายดิสก์ที่แบ่งใช้แบบเสมือนจะถูก cast in

**หมายเหตุ:** อีพซันนี้ควรถูกใช้เฉพาะภายใต้คำแนะนำโดยตรงจาก IBM Service. ไม่ควรใช้ในสภาพการณ์ปกติ

**-p** เซ็ทระดับของดิสก์ที่แบ่งใช้เสมือนแบบขนานกับจำนวนที่ระบุ ขอบเขตที่ใช้ได้คือ 1 ถึง 9 ค่าดีฟอลต์คือ 9 ค่าที่มากขึ้นให้เวลาการตอบสนองที่ต่ำกว่ากับการร้องขอจำนวนมาก (ดูที่ *RSCT for AIX 5L™: Managing Shared Disks* สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการปรับแต่งผลการทำงานดิสก์ที่แบ่งใช้แบบเสมือน

ค่านี้คือพารามิเตอร์ `buf_cnt` บนการเรียก `uphysio` ที่ไดเรกทอรีอุปกรณ์ IP ดิสก์ที่แบ่งใช้แบบเสมือนสร้างในเคอร์เนล ใช้ `statvsd` เพื่อแสดง ค่าปัจจุบันบนโหนดซึ่งคำสั่งถูกรัน

**-k** Casts out หมายเลขโหนดที่ระบุในโหนดโลคัล โหนดโลคัล ละเว้นการร้องขอจากโหนด cast out ใช้ `-r` เพื่อ cast โหนดกลับมา

**หมายเหตุ:**

1. ก่อนใช้แฟล็กนี้ อ้างอิงถึงส่วน “ข้อจำกัด”
2. อีพซันนี้ควรถูกใช้เฉพาะภายใต้คำแนะนำโดยตรงจาก IBM Service. ไม่ควรนำมาใช้ภายใต้สภาวะปกติ

**-t** แสดงรายการตารางเส้นทางและส่วนหัว mbuf ที่แคชโดยไดเรกทอรีดิสก์ที่แบ่งใช้แบบเสมือน

**-T** ลบหรือรีเซ็ตเส้นทางที่แคชทั้งหมด

**-v vsd\_name ...**

รีเซ็ทสถิติในการร้องขอจำนวนการอ่านและเขียนข้อมูล บน ดิสก์ที่แบ่งใช้แบบเสมือนที่ระบุ

**-V** รีเซ็ทสถิติในการร้องขอจำนวนการอ่านและเขียนข้อมูล ของดิสก์ที่แบ่งใช้แบบเสมือนที่กำหนดค่าไว้ทั้งหมด

**-C** รีเซ็ทตัวนับไดเรกทอรีอุปกรณ์ดิสก์ที่แบ่งใช้แบบเสมือน ที่แสดงโดย คำสั่ง `statvsd` Exceptions เป็นข้อมูลส่งออกและต้องการหมายเลข ลำดับการร้องขอระหว่างโหนดโคลเ็นต์และเซิร์ฟเวอร์

**-K** ละทิ้งโหนดทั้งหมดบนโหนดโลคัล การร้องขอโลคัลยังคงทำได้

หมายเหตุ:

1. ก่อนใช้แฟล็กนี้ อ้างอิงถึงส่วน “ข้อจำกัด”
2. อีพซันนี้ควรถูกใช้เฉพาะภายใต้คำแนะนำโดยตรงจาก IBM Service. ไม่ควรนำมาใช้ภายใต้สภาวะปกติ

-M เซ็ตขนาดข้อความ IP สูงสุดของดิสก์ที่แบ่งใช้แบบเสมือน นี้เป็นบล็อกข้อมูล ขนาดใหญ่ที่สุดที่ดิสก์ที่แบ่งใช้แบบเสมือนส่งผ่านเน็ตเวิร์กสำหรับ การร้องขอ I/O ข้อจำกัดนี้มีผลกับขนาดบล็อก I/O ดิสก์ที่แบ่งใช้แบบเสมือนโลคัลเช่นกัน ค่าเป็นไบต์และต้อง ไม่มากกว่าค่าขนาด maximum transmission unit (MTU) ของเน็ตเวิร์ก โหนดทั้งหมด ควรใช้ค่าเดียวกัน ค่าที่แนะนำคือ:

- 61440 (60KB) สำหรับสวิตช์
- 8192 (8KB) สำหรับ jumbo frame Ethernet
- 1024 (1KB) สำหรับ 1500-byte MTU Ethernet

## พารามิเตอร์

*logical\_volume\_name*

คือชื่อของโลจิคัลวอลุ่มที่คุณต้องการระบุเป็นดิสก์ที่แบ่งใช้แบบเสมือน วอลุ่มโลจิคัลนี้ต้องอยู่ในกลุ่มวอลุ่มโกลบอลที่ระบุ ความยาวชื่อต้องน้อยกว่าหรือเท่ากับ 15 อักขระ

*global\_group\_name*

คือชื่อของกลุ่มวอลุ่มเข้าถึงได้แบบโกลบอลที่กำหนดก่อนหน้านี้ โดยคำสั่ง `vsdvg` ที่คุณต้องการระบุดิสก์ที่แบ่งใช้แบบเสมือน ความยาวชื่อต้องน้อยกว่าหรือเท่ากับ 31 อักขระ

*vsd\_name*

ระบุชื่อเฉพาะสำหรับดิสก์ที่แบ่งใช้แบบเสมือนใหม่ ชื่อนี้ต้อง เป็นชื่อเฉพาะภายใน RSCD เพียร์โดเมน และเพื่อหลีกเลี่ยงความขัดแย้ง การตั้งชื่อในอนาคต ชื่อควรเป็นชื่อเฉพาะทั่วทั้งคลัสเตอร์ทั้งหมด ระเบียบในการตั้งชื่อ ที่แนะนำคือ `vsdnnvg_name` ความยาวชื่อต้องน้อยกว่าหรือเท่ากับ 31 อักขระ

หมายเหตุ: ถ้าคุณระบุ *vsd\_name* ที่เป็นชื่อของอุปกรณ์อื่นแล้ว คำสั่ง `cfgvsd` จะทำงานไม่สำเร็จสำหรับดิสก์ที่แบ่งใช้แบบเสมือนนั้น ขอผิดพลาดนี้ประกันว่าไฟล์อุปกรณ์พิเศษที่สร้างสำหรับ ชื่อที่กำหนดไม่ซ้อนกันและทำลายไฟล์ที่มีชื่อเหมือนกัน ซึ่งมีอยู่ในชนิดอุปกรณ์อื่น (เช่น โลจิคัลวอลุ่ม)

## ความปลอดภัย

คุณต้องมีสิทธิ์การใช้งาน `root` เพื่อรันคำสั่งนี้

### ข้อจำกัด

คุณต้องใช้คำสั่งนี้จากโหนดที่ออนไลน์อยู่ในโดเมนแบบเพียร์ หากต้องการทำให้โดเมนเพียร์ออนไลน์ ให้ใช้คำสั่ง `starttrpdomain` เมื่อต้องการทำให้โหนดออนไลน์ในเพียร์โดเมนที่มีอยู่ ใช้คำสั่ง `starttrpnode` สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการสร้างและการดูแล RSCD เพียร์โดเมน อ้างอิงถึง *คู่มือการดูแล RSCD*

### ตัวอย่าง

1. ตัวอย่างต่อไปนี้จะระบุว่า บนกลุ่มวอลุ่มที่เข้าถึงได้แบบโกลบอล `vg1n1`, โลจิคัลวอลุ่มชื่อ `lv1vg1n1` ถูกใช้เป็นดิสก์ที่แบ่งใช้แบบเสมือนที่ชื่อ `vsd1vg1n1`

```
defvsd lv1vg1n1 vg1n1 vsd1vg1n1
```

## Location

/opt/rset/vsd/bin/defvsd

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง vsdatalst

คำสั่ง vsdvg

คำสั่ง undefvsd

---

## คำสั่ง deleteX1 1 input

### วัตถุประสงค์

ลบเร็กคอร์ดส่วนขยาย X11 input จากฐานข้อมูล ODM (Object Data Manager)

### ไวยากรณ์

`deleteX1 1 input DeviceName ...`

### คำอธิบาย

คำสั่ง `deleteX1 1 input` ถูกใช้เพื่อลบเร็กคอร์ดส่วนขยายอินพุต X11 จากฐานข้อมูล ODM สำหรับแต่ละ `DeviceName` ที่ระบุ, ฐานข้อมูล ODM จะค้นหาอินสแตนซ์ของอ็อบเจกต์ให้มากที่สุดเท่าที่เป็นไปได้ คำสั่งนี้เคียวรีผู้ใช้เพื่อตรวจสอบว่าจะทำการลบแต่ละอุปกรณ์จำเพาะที่พบหรือไม่สามารถระบุชื่อแค่บางส่วนได้

คำสั่งเป็นคำสั่ง root หรือ system user การดำเนินการ จะล้มเหลวโดยมีข้อผิดพลาดสิทธิการใช้งาน ถ้าผู้ใช้ที่ไม่ได้รับอนุญาตพยายามลบ เร็กคอร์ด

### พารามิเตอร์

ไอเท็ม	คำอธิบาย
<code>DeviceName</code>	ระบุชื่อของอุปกรณ์ส่วนขยายอินพุต X11

### โค๊ดระบุมความผิดพลาด

ไอเท็ม	คำอธิบาย
ไม่พบ <code>DeviceName</code> ในฐานข้อมูล ODM	ไม่พบอ็อบเจกต์ที่ตรงกับรูปแบบที่ระบุในฐานข้อมูล ODM
การใช้งาน: <code>deleteX1 1 input DeviceName</code>	ผู้ใช้ไม่ได้ระบุชื่ออุปกรณ์

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `addX1 1 input`

คำสั่ง `listX1 1 input`

---

## คำสั่ง delta

### วัตถุประสงค์

สร้าง delta ในไฟล์ SCCS

### ไวยากรณ์

```
delta [-r SID] [-s] [-n] [-g List] [-p] [-m ModificationRequestList] [-y [Comment]] File ...
```

### คำอธิบาย

คำสั่ง delta เพิ่มการเปลี่ยนแปลง ที่มีกับเวอร์ชันไฟล์ในไฟล์ Source Code Control System (SCCS) ที่มีชื่อที่เรียกข้อมูลโดยคำสั่ง `get -e`

คำสั่ง delta อ่าน g-files ที่สัมพันธ์กับไฟล์ s ี่ระบุ (ดูที่คำสั่ง `get` สำหรับรายละเอียดของไฟล์ที่สร้างและใช้โดย SCCS) และสร้าง delta ใหม่ บรรทัดใน g-file ต้องมีอักขระไม่เกิน 512 ตัว

ถ้าคุณระบุได้เรียกทอรีสำหรับค่า *File* คำสั่ง delta จะปฏิบัติตามการร้องขอ กับไฟล์ SCCS ทั้งหมดซึ่งได้เรียกทอรีได้ถูกเช็คเอาต์ก่อนหน้านั้น เพื่อการแก้ไข (คือไฟล์ทั้งหมดที่นำหน้าด้วย s.) ถ้าคุณระบุ - (เครื่องหมายลบ) ในตำแหน่งของค่า *File* คำสั่ง delta จะอ่านอินพุตมาตรฐานและแปลแต่ละบรรทัดเป็นชื่อของไฟล์ SCCS เมื่อคำสั่ง delta อ่านอินพุตมาตรฐาน คุณต้องระบุแฟล็ก -y คุณยังต้องระบุแฟล็ก -m ถ้าแฟล็กส่วนหัว v ถูกเซต คำสั่ง delta อ่านอินพุตมาตรฐานจนกว่าจะถึงอักขระสิ้นสุดไฟล์

หมายเหตุ: บรรทัด ที่ขึ้นต้นด้วยอักขระ SOH ASCII (ไบนารี 001) ไม่สามารถเก็บ ในไฟล์ SCCS ยกเว้น SOH ถูกใส่เครื่องหมายคำพูดโดยใช้ \ (แบคสแลช) SOH มีความหมายพิเศษกับ SCCS และทำให้เกิดข้อผิดพลาด

ใช้คำสั่ง `get` กับไฟล์ SCCS ตามด้วยคำสั่ง delta กับไฟล์เดียวกัน, ควรหลีกเลี่ยงเมื่อคำสั่ง `get` สร้าง ข้อมูลจำนวนมาก คุณควรเปลี่ยนการใช้คำสั่ง `get` และ delta แทน

คำสั่ง delta บันทึกการเปลี่ยนแปลงที่ทำ กับเวอร์ชันของไฟล์ SCCS เมื่อต้องการใช้คำสั่ง delta:

1. ใช้คำสั่ง `get -e` เพื่อรับไฟล์ในเวอร์ชันที่แก้ไขได้
2. แก้ไขไฟล์นั้น
3. ใช้คำสั่ง delta เพื่อสร้างไฟล์ SCCS เวอร์ชันใหม่

คำสั่ง delta จะพร้อมท์คุณ ขอหมายเหตุ ถ้าไม่ได้รับอ็อปชัน -y หมายเหตุ ใช้กับ delta และแสดงในส่วนหัวไฟล์ SCCS หมายเหตุจะไม่ถูกเรียกขึ้นมาเมื่อคุณใช้คำสั่ง `get` เพื่อเรียก delta และไม่แสดงในข้อความของไฟล์ที่เรียกขึ้นมา ใช้หมายเหตุ เพื่อบันทึกเหตุผลที่สร้าง delta

เมื่อต้องการดูหมายเหตุ ให้ใช้เอดิเตอร์ดูที่ไฟล์ SCCS, เขียนไฟล์ SCCS ไปที่จอแสดงผลด้วยคำสั่ง `cat` หรือพิมพ์ส่วนของไฟล์ ที่เลือกไปที่เอาต์พุตมาตรฐานโดยใช้คำสั่ง `prs` โปรดจำไว้ว่าอย่างเปลี่ยนแปลงเนื้อหาของไฟล์ SCCS โดยตรง เมื่อต้องการเปลี่ยนหมายเหตุ delta ให้ใช้คำสั่ง `cdc`

หมายเหตุ: อย่าใช้คำสั่ง delta บนไฟล์ถ้าไฟล์มีคีย์เวิร์ก identification ที่ขยายเพิ่ม เวอร์ชันของไฟล์อ่านอย่างเดียวกแทนที่คีย์เวิร์ดด้วยคำขอความ การใช้ คำสั่ง delta กับไฟล์อ่านอย่างเดียวกทำให้คีย์เวิร์ด สูญหายไป เมื่อต้องการกู้คืนจากสถานการณ์นี้ ให้ลบ delta หรือแก้ไขไฟล์ อีกครั้งและแทนที่คีย์เวิร์ด identification

SCCS ไม่อนุญาตให้ใช้คำสั่ง delta นอกจากมีสำเนาของไฟล์ที่แก้ไขได้อยู่

เมื่อต้องการป้องกันการสูญเสียคีย์เวิร์ด ให้ใช้คำสั่ง admin ด้วยแฟล็ก -f เพื่อระบุแฟล็กส่วนหัว i หลังจากนั้น ถ้าไม่มีคีย์เวิร์ดในเวอร์ชันไฟล์ จะทำให้เกิดข้อผิดพลาด

## แฟล็ก

### ไอเท็ม

-g List

-m ModificationRequestList

-n

-p

-r SID

-s

-y [Comment]

### คำอธิบาย

ระบุรายการของ SIDs (deltas) ที่จะถูกละเว้นเมื่อคำสั่ง get สร้าง g-file หลังจากคุณใช้แฟล็กนี้ คำสั่ง get จะละเว้น delta ที่ระบุเมื่อคำสั่งทำการสร้าง g-file ถ้าไฟล์ SCCS มีแฟล็กส่วนหัว v เซ็ตไว้ ต้องมีการระบุหมายเลข Modification Request (MR) เป็นเหตุผลสำหรับการสร้าง delta ใหม่

ถ้าคุณไม่ระบุแฟล็ก -m และ แฟล็กส่วนหัว v ถูกเซ็ต คำสั่ง delta จะอ่าน MRs จากอินพุตมาตรฐาน ถ้าอินพุตมาตรฐานคือ เวิร์กสเปซ คำสั่ง delta จะพร้อมคนเพื่อขอ MRs คำสั่ง delta ทำงานต่อไปเพื่อรับอินพุตจนกว่า คำสั่งอ่านอักขระสิ้นสุดไฟล์ คำสั่งอ่าน MRs ก่อนหมายเหตุเสมอ (ดูที่ แฟล็ก -y) คุณสามารถใช้ ช่องว่าง อักขระแท็บ หรือ ทั้งสองอย่างเพื่อแยก MRs ในรายการ

ถ้าแฟล็กส่วนหัว v มีค่า คำจะถูกแปลเป็นชื่อของโปรแกรมที่ตรวจสอบ หมายเลข MR ถ้าคำสั่ง delta ส่งคืนค่า exit ที่ไม่ใช่ศูนย์ จากโปรแกรมตรวจสอบ MR คำสั่ง delta จะถือว่าหมายเลข MR บางส่วนไม่ถูกต้องและหยุดการรัน

คง g-file ไว้, ซึ่งปกติถูกลบออกเมื่อคำสั่ง delta ประมวลผลเสร็จสิ้น เขียนข้อมูลความแตกต่างของไฟล์ SCCS (ในรูปแบบของคำสั่ง diff) ไปที่เอาต์พุตมาตรฐานก่อนและหลังการใช้คำสั่ง delta ดูที่คำสั่ง diff สำหรับคำอธิบายรูปแบบระบุ delta ซึ่งจะถูกสร้างในไฟล์ SCCS คุณต้องใช้ แฟล็กนี้ เฉพาะถ้าคำสั่ง get -e ที่ค้างอยู่ได้ถูกดำเนินการเสร็จสิ้นบนไฟล์ SCCS เดียวกันโดยบุคคลเดียวกัน คำ SID เป็นได้ทั้ง SID ที่ระบุบนคำสั่ง get หรือ SID ที่จะถูกสร้าง (ตามที่รายงานโดยคำสั่ง get) จะมีข้อผิดพลาดเกิดขึ้นถ้า SID ที่ระบุไม่สามารถถูกจำแนก ได้เป็นค่าเฉพาะ หรือถ้าต้องมีการระบุ SID แต่ไม่ได้ระบุ

ยบยั้งข้อมูลที่ปกติเขียนไปที่เอาต์พุตมาตรฐานเมื่อ คำสั่ง delta เสร็จสิ้นตามปกติ ระบุข้อความที่อธิบายเหตุผลในการสร้าง delta สตริง ที่มีค่าว่าง ถูกพิจารณาว่าเป็นค่าหมายเหตุที่ใช้ได้ ถ้าบรรทัดหมายเหตุของคุณ มีอักขระพิเศษหรือช่องว่าง บรรทัดต้องถูกปิดด้วย เครื่องหมายอัฒประกาศเดี่ยวหรือคู่

ถ้าคุณไม่ระบุแฟล็ก -y คำสั่ง delta จะอ่านหมายเหตุจากอินพุตมาตรฐาน จนกว่าจะพบบรรทัดว่างหรืออักขระสิ้นสุดไฟล์

สำหรับอินพุตจากคีย์บอร์ด คำสั่ง delta พร้อมต์ ข้อมูลหมายเหตุ ถ้าอักขระสุดท้ายของบรรทัดคือ \ (backslash), จะ ถูกละเว้น หมายเหตุต้องมีความยาวไม่เกิน 512 อักขระ

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย  
0 ดำเนินการเป็นผลสำเร็จ  
>0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการบันทึกการเปลี่ยนแปลงที่คุณได้ทำกับไฟล์ SCCS, ให้ป้อน:

```
delta s.prog.c
```

ซึ่งจะเพิ่ม delta ให้กับไฟล์ SCCS s.prog.c, บันทึกการเปลี่ยนแปลงที่เกิดขึ้นโดยการแก้ไข prog.c จากนั้นโปรแกรม delta จะถามคุณถึงหมายเหตุที่สรุปการเปลี่ยนแปลงที่คุณได้กระทำ ป้อนหมายเหตุแล้วป้อนอักขระสิ้นสุดไฟล์หรือกดปุ่มกลับสองครั้งเพื่อระบุว่าคุณเสร็จสิ้นการใส่หมายเหตุ

2. เมื่อต้องการบันทึกการเปลี่ยนแปลงที่คุณทำกับไฟล์ SCCS พร้อมกับหมายเหตุอธิบายสั้นๆ ให้ป้อน:

```
delta -y "This delta contains the payroll function" s.prog.c
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/delta	มีคำสั่ง delta

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง diff” ในหน้า 150

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง cat

คำสั่ง cdc

คำสั่ง prs

ภาพรวม Source Code Control System (SCCS)

---

## คำสั่ง deroff

### วัตถุประสงค์

ลบการสร้างคำสั่ง nroff, troff, tbl และ eqn ออกจากไฟล์

### ไวยากรณ์

```
deroff { -ma -me -ms [-mm [-ml] ] } [ -i | -l ] [ -k ] [ -p ] [ -u ] [ -w ] [ File ... ]
```

### คำอธิบาย

คำสั่ง deroff อ่าน ไฟล์ที่ระบุ (อินพุตมาตรฐานโดยดีฟอลต์) ที่มีข้อความภาษาอังกฤษ ลบการร้องขอ การเรียกแม่โคร backslash constructs ของ troff, eqn command constructs (ระหว่างบรรทัด .EQ และ .EN และระหว่างตัวคั่น), และคำอธิบาย คำสั่ง tbl ทั้งหมด จากนั้นเขียนส่วนที่เหลือของไฟล์ไปที่เอาต์พุต มาตรฐาน

โดยปกติคำสั่ง `deroff` ต่อกำหนดชื่อของไฟล์ที่รวมไว้ (การร้องขอคำสั่ง `.so` และ `.nx troff`) ถ้าไฟล์ได้ถูกรวมไว้แล้ว การร้องขอ `.so` เพื่อกำหนดชื่อจะถูกละเว้นและ การร้องขอ `.nx` กำหนดชื่อไฟล์นั้นสิ้นสุดการดำเนินการ

หมายเหตุ: คำสั่ง `deroff` ไม่ใช่ตัวแปลคำสั่ง `troff` แบบสมบูรณ์ ดังนั้นจึงอาจเกิดความสับสนกับ `subtle constructs` ข้อผิดพลาดส่วนใหญ่เกิดจาก มีเอาต์พุตมากเกินไป ไม่ใช่จากเอาต์พุตน้อยเกินไป

## พารามิเตอร์

ไอเท็ม คำอธิบาย

`File` ระบุไฟล์ข้อความภาษาอังกฤษสำหรับคำสั่ง `deroff` เพื่อลบผลของการประมวลผลคำสั่ง `troff`, `eqn` และ `tbl` ไฟล์ที่พอลต์คือ อินพุตมาตรฐาน

## แฟล็ก

ไอเท็ม คำอธิบาย

- `-ma` ละเว้นแมโคร `MA (man)` ในข้อความ เพื่อให้มีเพียงข้อความที่รันเป็นเอาต์พุต
- `-me` ละเว้นแมโคร `ME` ในข้อความเพื่อให้ข้อความที่รัน เท่านั้นเป็นเอาต์พุต นี้คือ คาส์พอลต์
- `-ml` ละเว้นแมโคร `MM` ในข้อความ (`-mm flag`) และลบโครงสร้างรายการ `MM` แฟล็ก `-mm` ต้องถูกระบุด้วยแฟล็กนี้  
หมายเหตุ: อย่าใช้แฟล็ก `-ml` ที่มี รายการซ้อนกัน
- `-mm` ละเว้นแมโคร `MM`
- `-ms` ละเว้นแมโคร `MS` ในข้อความเพื่อให้ข้อความที่รัน เท่านั้นเป็นเอาต์พุต
- `-i` ยับยั้งการประมวลผลของไฟล์ที่รวม
- `-l` ยับยั้งการประมวลผลของไฟล์ที่รวม ซึ่งชื่อเริ่มต้นด้วย `/usr/lib`, เช่นไฟล์แมโครใน `/usr/lib/tmac`
- `-k` รักษาบล็อกที่ระบุเพื่อเก็บไว้รวมกัน คาส์พอลต์คือบล็อกของข้อความที่เก็บไว้ ตัวอย่าง `.ne construct` ถูกลบออก
- `-p` ประมวลผลยูทอหนาพิเศษ
- `-u` ลบขีดเส้นใต้ ASCII และลำดับการควบคุมตัวหนาออก แฟล็กนี้ เซ็ตแฟล็ก `-w` โดยอัตโนมัติ
- `-w` ทำเอาต์พุตให้เป็นรายการคำ โดยมีหนึ่งคำต่อบรรทัดและลบ อักขระอื่นทั้งหมด มีฉะนั้น เอาต์พุตจะเป็นไปตามต้นฉบับ

ในข้อความ คำคือสตริงที่ขึ้นต้นด้วยตัวอักษร มีตัวอักษรอย่างน้อย สองตัวอักษร และประกอบด้วยตัวอักษร ตัวเลข แอมเปอร์แซนด์ (&) และอะโพสโทรฟี่ (') ในการเรียกแมโคร คำเป็นสตริงที่ขึ้นต้นด้วยตัวอักษร อย่างน้อยสองตัวอักษร โดยมีจำนวนรวมอย่างน้อยสามตัวอักษร ตัวคั่นคือ อักขระที่ไม่ใช่ตัวอักษร ตัวเลข เครื่องหมายวรรคตอน อะโพสโทรฟี่ และแอมเปอร์แซนด์ อะโพสโทรฟี่และแอมเปอร์แซนด์ต่อท้ายจะถูกเอาออกจากคำ

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `eqn`” ในหน้า 419

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `nroff`

คำสั่ง `tbl`

คำสั่ง `troff`

---

## คำสั่ง `detachrset`

### วัตถุประสงค์

แยก `rset` ออกจากการประมวลผล

### ไวยากรณ์

```
detachrset [ -P ] pid
```

## คำอธิบาย

คำสั่ง `detachrset` แยก `rset` ออกจากการประมวลผล การแยก `rset` ออกจากการประมวลผลจะอนุญาตให้การประมวลผลใช้ โพรเซสเซอร์และ/หรือพื้นที่หน่วยความจำ ในระบบ

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-P</code>	แยกพาร์ติชัน <code>rset</code> ออกจากการประมวลผลที่ระบุ ( <code>pid</code> )

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
<code>pid</code>	Process ID.

## ความปลอดภัย

ผู้ใช้ต้องมีสิทธิ์ `root` หรือมีความสามารถ `CAP_NUMA_ATTACH` และการประมวลผลเป้าหมายต้องมี `userid` ที่มีผลเช่นเดียวกันกับผู้เรียกคำสั่ง ผู้ใช้ต้องมีสิทธิ์ `root` ในการลบพาร์ติชัน `rset` ออกจากการประมวลผล (อ็อปชัน `-P`)

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ตัวอย่าง

เมื่อต้องการแยก `rset` ออกจากการประมวลผล 21414, พิมพ์:

```
detachrset 21414
```

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/detachrset</code>	มีคำสั่ง <code>detachrset</code>

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `execrset`” ในหน้า 458

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `attachrset`

คำสั่ง `lsrset`

คำสั่ง `mkrset`

---

## คำสั่ง devinstall

### วัตถุประสงค์

ติดตั้งการสนับสนุนซอฟต์แวร์สำหรับอุปกรณ์

### ไวยากรณ์

```
devinstall -f File -d Device[-s][-v]
```

### คำอธิบาย

คำสั่ง `devinstall` ติดตั้ง การสนับสนุนซอฟต์แวร์สำหรับอุปกรณ์ ซึ่งจะติดตั้งซอฟต์แวร์แพ็คเกจที่มีรายการใน ไฟล์ที่ระบุ โดยแฟล็ก `-f`

สำหรับอุปกรณ์ใหม่ส่วนใหญ่ที่ถูกเพิ่มหลังจากการติดตั้ง ซอฟต์แวร์เริ่มต้น ซอฟต์แวร์สำหรับอุปกรณ์ใหม่สามารถถูกติดตั้ง โดยใช้ แฟล็ก `-i` ของคำสั่ง `cfgmgr`

ในบางกรณี อุปกรณ์ใหม่แทนที่อุปกรณ์ที่จำเป็นในการเริ่มต้นเครื่อง ตัวอย่าง คุณอาจแทนที่อะแดปเตอร์การ์ด SCSI ที่สนับสนุนกลุ่มวอลุ่ม root หรือกราฟิกอะแดปเตอร์การ์ด ที่สนับสนุนคอนโซล ในกรณีนี้ เครื่องจะไม่สตาร์ทใหม่จนกว่าคุณจะได้ติดตั้งการสนับสนุนซอฟต์แวร์สำหรับอุปกรณ์ใหม่นี้ เมื่อต้องการดำเนินการดังนี้ ให้ปิดระบบของคุณและติดตั้งฮาร์ดแวร์ใหม่ตาม คำแนะนำที่มาพร้อมกับฮาร์ดแวร์ของคุณ ขั้นตอนต่อไป เริ่มการทำงานของเครื่องของคุณ ในโหมดดูแลรักษา ระหว่างกระบวนการเริ่มทำงาน อะแดปเตอร์จะถูกตรวจพบ และไฟล์ `/tmp/device.pkgs` จะถูกสร้างโดยมี ชื่อของซอฟต์แวร์แพ็คเกจที่จำเป็นในการสนับสนุนฮาร์ดแวร์ใหม่ เมื่อเครื่อง อยู่ในโหมดดูแลรักษา คุณสามารถติดตั้งซอฟต์แวร์สำหรับอุปกรณ์ใหม่นี้ โดยการรันคำสั่ง `devinstall`

### แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-f File</code>	ระบุไฟล์ที่มีรายการของรายการที่จะถูกติดตั้ง โดยปกติจะเป็นไฟล์ <code>/tmp/device.pkgs</code> ที่สร้างโดยคำสั่ง <code>cfgmgr</code>
<code>-d Device</code>	ระบุที่ตั้งของสื่อบันทึกการติดตั้ง ซึ่งสามารถเป็น อุปกรณ์ฮาร์ดแวร์ เช่น เทปหรือดิสเก็ต ไดเรกทอรีที่มีอิมเมจการติดตั้ง หรือตัวไฟล์อิมเมจการติดตั้งเอง เมื่อสื่อ การติดตั้งเป็น IBM Installation tape หรือ IBM Corrective Service tape, อุปกรณ์เทปควรถูกระบุเป็น <code>no-rewind-on-close</code> และ <code>no-retension-on-open</code> ตัวอย่างในกรณีนี้จะเป็น <code>/dev/rmt0.1</code> สำหรับเทปความจุสูงหรือ <code>/dev/rmt0.5</code> สำหรับเทป ความจุสูง สำหรับเทปที่ไม่ใช่ของ IBM ให้ใช้อ็อปชันที่ระบุโดย ผู้ผลิต
<code>-s</code>	เขียนทับไฟล์ <code>/var/adm/dev_pkg.fail</code> ไฟล์นี้มีรายการของแพ็คเกจที่ไม่ได้ติดตั้งอย่างสมบูรณ์ และสามารถถูกใช้เพื่อช่วยการกู้คืนหรือการติดตั้งจากซอร์สที่ต่างกัน
<code>-v</code>	ระบุอ็อปชัน <code>verbose</code> , ทำให้คำสั่ง <code>devinstall</code> แสดงข้อมูลเพิ่มเติมขณะประมวลผล

คำสั่ง `devinstall` ติดตั้ง แพ็คเกจอุปกรณ์ที่แสดงรายการในไฟล์ที่ระบุในบรรทัดคำสั่ง ซึ่งรัน คำสั่ง `geninstall` ด้วย `-I "acXge /var/adm/ras/devinst.log"`, โดยที่ a: apply, c: commit, X: extend fs, e: log และ `/var/adm/ras/devinst.log` คือชื่อพาล็อกไฟล์แบบเต็ม, g: auto\_include (ดูที่คำสั่ง `geninstall` สำหรับข้อมูลเพิ่มเติมเกี่ยวกับแฟล็กเหล่านี้) คำสั่ง `devinstall` ตรวจสอบไฟล์สคริปต์ที่สร้างโดยคำสั่ง `geninstall` สำหรับผลลัพธ์ของแต่ละความพยายามติดตั้งแพ็คเกจ และจาก ข้อมูลนี้ มีการสร้างไฟล์สองไฟล์ ไฟล์ `/var/adm/dev_pkg.fail` แสดงรายการแพ็คเกจที่ล้มเหลวในการติดตั้ง (ถ้ามี) ไฟล์ `/var/adm/dev_pkg.success` แสดงรายการแพ็คเกจทั้งหมดที่ติดตั้งสำเร็จ

## ค่าที่ส่งกลับ

ค่าส่งกลับ 0 ระบุว่าไม่มีแพ็คเกจ ถูกติดตั้ง

ค่าส่งกลับ 1 ระบุว่าอย่างน้อยหนึ่งแพ็คเกจ ได้ถูกติดตั้งเสร็จสมบูรณ์ และคำสั่ง **bosboot** ควรถูกเรียกใช้งาน

ค่าส่งกลับ 2 แสดงว่าคำสั่ง **devinstall** ล้มเหลว

ไฟล์ `/var/adm/dev_pkg.success` แสดงรายการ ที่ติดตั้งเสร็จสมบูรณ์ ไฟล์ `/var/adm/dev_pkg.fail` แสดงรายการแพ็คเกจที่ล้มเหลวในการติดตั้ง

## ความปลอดภัย

Privilege Control: เฉพาะผู้ใช้ **root** เท่านั้นที่สามารถรัน คำสั่งนี้ได้

## ตัวอย่าง

เมื่อต้องการติดตั้งซอฟต์แวร์เพื่อสนับสนุนอุปกรณ์ใหม่หลังจากคุณ ได้สตาร์ทเครื่องจากเทปการติดตั้งอุปกรณ์และเข้าสู่โหมดดูแลรักษาให้ป้อน:

```
devinstall -f ../tmp/device.pkgs -d /dev/rmt0.1
```

จากนั้นรันคำสั่ง **bosboot**

```
bosboot -ad /dev/ipldevice
```

## File

ไอเท็ม	คำอธิบาย
<code>/dev/rmtn</code>	ระบุอินเทอร์เฟซของเทปการสตรีมแบบ raw

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **bosboot**

คำสั่ง **cfgmgr**

คำสั่ง **installp**

---

## คำสั่ง devnm

### วัตถุประสงค์

ตั้งชื่ออุปกรณ์

### ไวยากรณ์

**devnm** *Path* ...

## คำอธิบาย

คำสั่ง `devnm` อ่านพารามิเตอร์ *Path*, ระบุไฟล์พิเศษที่สัมพันธ์กับระบบไฟล์ที่เม้าท์ที่พารามิเตอร์ *Path* อยู่ และเขียนชื่อไฟล์พิเศษไปที่เอาต์พุตมาตรฐาน แต่ละพารามิเตอร์ *Path* ต้องเป็นชื่อพาทแบบเต็ม

การใช้งานทั่วไปของคำสั่ง `devnm` command คือโดยไฟล์คำสั่ง `/etc/rc` เพื่อสร้างรายการตารางเม้าท์สำหรับอุปกรณ์ `root`

หมายเหตุ: คำสั่งนี้ใช้สำหรับระบบไฟล์โลคัลเท่านั้น

## ตัวอย่าง

1. เมื่อต้องการระบุอุปกรณ์ซึ่งมีไฟล์อยู่ให้ป้อน:

```
devnm /diskette0/bob/textfile
```

นี่จะเป็น การแสดงชื่อของไฟล์อุปกรณ์พิเศษซึ่งมีไฟล์ `/diskette0/bob/textfile` อยู่ถ้าดิสเก็ตถูกเม้าท์เป็นอุปกรณ์ `/diskette0`, คำสั่ง `devnm` จะแสดง:

```
fd0 /diskette0/bob/textfile  
rfd0 /diskette0/bob/textfile
```

นี่หมายความว่าไฟล์ `/diskette0/bob/textfile` อยู่ในดิสเก็ตไดรฟ์ `/dev/fd0`

2. เมื่อต้องการระบุอุปกรณ์ซึ่งมีระบบไฟล์ อยู่ให้ป้อน:

```
devnm /
```

นี่จะเป็นการ แสดงชื่อของอุปกรณ์ซึ่งระบบไฟล์ `root(/)` อยู่ รายการต่อไปนี้จะถูกแสดงบนจอแสดงผล:

```
hd0 /
```

นี่หมายความว่าระบบไฟล์ `root(/)` อยู่ในอุปกรณ์ `/dev/hd0`

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/dev</code>	ระบุไดเรกทอรี
<code>/usr/sbin/devnm</code>	มีคำสั่ง <code>devnm</code>

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `rc`

---

## คำสั่ง `devrsrv`

### วัตถุประสงค์

เคียวรีและหยุดพาทเดี่ยว และการสำรองอย่างต่อเนื่องบนอุปกรณ์

## ไวยากรณ์

```
devrsrv -c query [-e] | release | prin -s sa | (prout -s sa -r rkey -k sa_key -t prtype) -l devicename
```

```
devrsrv -f -l devicename
```

```
devrsrv -d
```

## คำอธิบาย

คำสั่ง **devrsrv** จะเคียวรีและหยุดพาธเดียวและการสำรองอย่างต่อเนื่องบน อุปกรณ์ คำสั่งจะรันแอ็คชันเซอร์วิส persistent reserve in (prin) และ persistent reserve out (prout)

คำสั่งย่อย **query** เคียวรีและแสดง สถานะการสำรองปัจจุบันของอุปกรณ์ คำสั่งย่อย **release** รีลีส์การสำรองบนอุปกรณ์โดยใช้ การสำรองพาธเดียว

คำสั่งย่อย **prin** แสดงคีย์การสำรองที่ลงทะเบียนแล้ว ผู้จัดการคีย์การสำรอง และข้อมูลความสามารถ คำสั่งย่อย **prout** ร้องขอ เซอร์วิสแอ็คชันที่สำรองอุปกรณ์สำหรับ การใช้งานเฉพาะ หรือที่แบ่งใช้ของพาธ I/O ไปยังอุปกรณ์ที่เจาะจง คำสั่งย่อย **prout** สนับสนุนเซอร์วิสแอ็คชันต่อไปนี้:

ไอเท็ม	คำอธิบาย
RELEASE	รีลีส์การสำรองอย่างต่อเนื่องที่ระบุไว้สำหรับอุปกรณ์
CLEAR	ล้างข้อมูลคีย์การสำรองทั้งหมดและ การสำรองอย่างต่อเนื่องทั้งหมด
PREEMPT	ครอบครองการสำรองอย่างต่อเนื่อง หรือถอนการลงทะเบียน หรือทั้งสอง
PREEMPT AND ABORT	ครอบครองการสำรองอย่างต่อเนื่องหรือลบการลงทะเบียน หรือทั้งคู่และหยุดงานทั้งหมดสำหรับพาธ I/O ที่ครอบครองไว้ทั้งหมดกับอุปกรณ์
REGISTER AND IGNORE KEY	ลงทะเบียนคีย์ใหม่แทน คีย์เก่า

## แฟล็ก

ไอเท็ม	คำอธิบาย
-c	ระบุคำสั่งย่อยต่อไปนี้:  <b>เคียวรี</b> เคียวรีและแสดงสถานะของการสำรองบนอุปกรณ์  <b>รีลีส์</b> รีลีส์อุปกรณ์ด้วยการสำรองพาธ-เดียว โดยใช้ SCSI-2  <b>prin</b> ระบุการสำรองการดำเนินการกับเซอร์วิสภายในอย่างต่อเนื่อง  <b>prout</b> ระบุการสำรองการดำเนินการกับเซอร์วิสภายนอกอย่างต่อเนื่อง แสดงรายการชื่อดิสก์และข้อมูลการระบุอื่นๆ สำหรับดิสก์ทั้งหมดที่ถูกเคียวรีหรือจัดการโดยใช้คำสั่ง <b>devrsrv</b>  <b>-d</b> หลีกเลี่ยงการเปิดดิสก์ในโหมด exclusive ซึ่งรวมทั้งโหมดเดี่ยวและ โหมดวินิจัย แฟล็กนี้ใช้ได้สำหรับการจองประเภท PR_exclusive และ PR_shared เท่านั้น หากนโยบายการจอง Object Data Manager (ODM) คือ single_path แฟล็กนี้จะถูกข้าม  <b>-e</b> <b>หมายเหตุ:</b> ในบางกรณี หากคุณใช้แฟล็กนี้ คำสั่ง <b>devrsrv</b> อาจไม่พิจารณาสถานะการจองของดิสก์ หรือไม่พิจารณาว่า ดิสก์เปิดอยู่แลวนโลคัลไฮสแตตหรือไม่  <b>-f</b> หยุดการสำรองที่จัดการโดยพาธ I/O อื่นๆ หรือไฮสแต สำหรับการสำรองพาธเดียว คำสั่ง <b>devrsrv</b> จะ ใช้แอ็คชัน SC_FORCED_OPEN เพื่อหยุดการสำรอง สำหรับการสำรอง persistent คำสั่ง <b>devrsrv</b> ออก คำสั่งย่อย <b>prout</b> พร้อมกับเซอร์วิสแอ็คชัน CLEAR เพื่อล้างค่า การสำรอง persistent และการรีจิสเตอร์ ระบุคีย์การสำรองของการดำเนินการกับเซอร์วิส แฟล็ก -k จำเป็นต้องมีสำหรับการดำเนินการกับเซอร์วิส REGISTER, PREEMPT และ PREEMPT_ABORT  <b>-l</b> ระบุชื่อของอุปกรณ์

## ไอเท็ม

-r	<b>คำอธิบาย</b> ระบุคีย์การสำรองข้อมูล แฟล็ก -r จำเป็นต้องมีสำหรับการดำเนินการกับเซอวิส REGISTER, PREEMPT, PREEMPT AND ABORT และ RELEASE
-s	ระบุการดำเนินการกับเซอวิสสำหรับ การสำรองอย่างต่อเนื่อง เซอวิสแอดคชันที่ใช้ได้สำหรับคำสั่งย่อย prin มีดังนี้:  0 READ KEYS  1 READ RESERVATION  2 REPORT CAPABILITIES  เซอวิสแอดคชันที่ใช้ได้สำหรับคำสั่งย่อย prout มีดังนี้:  2 RELEASE  3 CLEAR  4 PREEMPT  5 PREEMPT AND ABORT  6 REGISTER AND IGNORE EXISTING KEY
-t	ระบุประเภทของการสำรองอย่างต่อเนื่อง ประเภทของ การสำรองอย่างต่อเนื่องดังต่อไปนี้:  1 การเขียนเฉพาะ  2 การเข้าถึงเฉพาะ  3 การเขียนเฉพาะการลงทะเบียนเท่านั้น  4 การเข้าถึงเฉพาะการลงทะเบียนเท่านั้น  5 การเขียนเฉพาะการลงทะเบียนทั้งหมด  6 การเข้าถึงเฉพาะการลงทะเบียนทั้งหมด

## ตัวอย่าง

ต่อไปนี้เป็นตัวอย่างที่เกี่ยวข้องกับสถานการณ์จำลองอื่นๆ

### การดำเนินการกับเคียวรี

1. หากต้องการเคียวรีสถานะการสำรองของอุปกรณ์ hdisk0 เมื่อไม่สำรองโดยโฮสต์ใดๆ ให้ป้อนคำสั่งต่อไปนี้:

```
# devrsrv -c query -l hdisk0
```

```
Device Reservation State Information
```

```
=====
```

```
Device Name           : hdisk0  
Device Open On Current Host? : NO  
ODM Reservation Policy : SINGLE PATH RESERVE  
Device Reservation State : NO RESERVE
```

เอาต์พุตแสดงอุปกรณ์ที่ไม่ได้เปิดบนโฮสต์ปัจจุบันและนโยบายการสำรอง Object Data Manager (ODM) คือ SINGLE PATH RESERVE ซึ่งระบุว่านโยบายการสำรองถูกตั้งค่าใน ODM สำหรับ อุปกรณ์นี้ สถานะการสำรองอุปกรณ์บ่งชี้การสำรองที่แสดงบนอุปกรณ์ คุณสามารถค้นหาสถานะการสำรองอุปกรณ์โดยรัน ลำดับของคำสั่ง SCSI

2. หากต้องการเคียวรีสถานะการสำรองของอุปกรณ์ hdisk1 เมื่อถูกสำรองตามโฮสต์ ให้ป้อนคำสั่งต่อไปนี้:

```
# devrsrv -c query -l hdisk1
```

### อุปกรณ์ถูกสำรองโดยใช้การสำรองพาร์ติชันตามโฮสต์

Device Reservation State Information

```
=====
Device Name           : hdisk1
Device Open On Current Host? : NO
ODM Reservation Policy : SINGLE PATH RESERVE
Device Reservation State : SINGLE PATH RESERVE
```

3. หากต้องการตรวจสอบสถานะการสำรองของอุปกรณ์ hdisk2 เมื่อถูกสำรองไว้บนโฮสต์เดียวกันให้ป้อนคำสั่งต่อไปนี้:

```
# devrsrv -c query -l hdisk2
```

Device Reservation State Information

```
=====
Device Name           : hdisk2
Device Open On Current Host? : YES
ODM Reservation Policy : SINGLE PATH RESERVE
Device Reservation State : SINGLE PATH RESERVE
Path Id of Reserved Path : 0
```

4. หากต้องการตรวจสอบสถานะการสำรองของอุปกรณ์ hdisk2 เมื่อนโยบายการสำรอง ODM คือ PR SHARED และอุปกรณ์ไม่ได้สำรองไว้โดยโฮสต์ใดๆ ป้อนคำสั่งต่อไปนี้:

```
# devrsrv -c query -l hdisk0
```

Device Reservation State Information

```
=====
Device Name           : hdisk0
Device Open           : NO
ODM Reservation Policy : PR SHARED
ODM PR Key Value      : 7777
Device Reservation State : NO RESERVE
Registered PR Keys    :
555
777
PR Capabilities Byte[2] : 0xd SIP_C ATP_C PTPL_C
PR Capabilities Byte[3] : 0x0
PR Types Supported     : NOT VALID
```

คำอธิบายของฟิลด์ต่างๆ จากเอาต์พุตเคียวรีต่อไปนี้:

#### Registered PR Keys :

แสดงคีย์ที่ถูกริจิสเตอร์โดยคำสั่งย่อย **prout** พร้อมกับเซอร์วิสแอสซัน REGISTER จากพาร์ I/O ทั้งหมดที่กำลัง แบ่งใช้อุปกรณ์นี้

#### ขีดความสามารถ PR:

ระบุเนื้อหาของไบนารี 2 และ 3 ที่ส่งคืนโดยเซอร์วิสแอสซัน REPORT CAPABILITIES ของคำสั่งย่อย **prin** โปรดดู SPC มาตรฐานเพื่อตีความเอาต์พุตของตัวอย่าง 4

#### ประเภท PR ที่สนับสนุน:

แสดงชนิดการสำรองอย่างต่อเนื่องที่สนับสนุนโดย อุปกรณ์ที่รายงานโดยฟิลด์มาสก์ชนิดการสำรองอย่างต่อเนื่อง ในเอาต์พุตความสามารถของรายงาน

หากการสำรองอย่างต่อเนื่องถูกจัดการบนอุปกรณ์เอาต์พุตเคียวรี แสดงข้อมูลเพิ่มเติมเกี่ยวกับการสำรองอุปกรณ์ดังต่อไปนี้:

## ประเภทการสำรอง PR:

แสดงหนึ่งในค่าของ PR Types ที่อธิบายในส่วน แฟล็ก

## ค่าคีย์ตัวจัดการ PR:

แสดงค่าคีย์ PR ของตัวจัดการสำรองปัจจุบัน ค่าคีย์การสำรองอย่างต่อเนื่องคือ 0 หากประเภท PR คือ 5 หรือ 6

## การดำเนินการสำรองภายในอย่างต่อเนื่อง (prin)

1. หากต้องการอ่านคีย์การสำรองที่ลงทะเบียนแล้ว ให้ป้อนคำสั่งต่อไปนี้:

```
# devrsrv -c prin -s 0 -l hdisk0
```

```
Registered PR Keys      :  
555  
777
```

2. หากต้องการอ่านตัวจัดการสำรองปัจจุบัน และพิมพ์ให้ป้อนคำสั่งต่อไปนี้:

```
# devrsrv -c prin -s 1 -l hdisk0
```

```
PR Generation Value    : 2  
PR Type                 : PR_EA_RO (EXCLUSIVE ACCESS, REGISTRANTS ONLY)  
PR Holder Key Value    : 777
```

3. หากต้องการส่งคืนข้อมูลความสามารถ PR ที่สนับสนุนโดยส่งการดำเนินการความสามารถ ของรายงาน ให้ป้อนคำสั่งต่อไปนี้:

```
# devrsrv -c prin -s 2 -l hdisk0
```

```
PR Capabilities Byte[2] : 0xd SIP_C ATP_C PTPL_C  
PR Capabilities Byte[3] : 0x0  
PR Types Supported      : NOT VALID
```

## การดำเนินการสำรองภายนอกอย่างต่อเนื่อง (prout)

### การดำเนินการกับเซอวิส RELEASE

หากต้องการรีเซ็ตการสำรองอย่างต่อเนื่องจาก IT-nexus ที่ลงทะเบียนแล้ว และสำรองด้วยคีย์ 1777 และการสำรอง PR ประเภท 4 ป้อนคำสั่งต่อไปนี้:

```
# devrsrv -c prout -s 2 -r 1777 -t 4 -l hdisk0
```

```
Device Reservation State Information
```

```
=====
```

```
Device Name           : hdisk0  
Device Open On Current Host? : YES  
ODM Reservation Policy : PR SHARED  
ODM PR Key Value      : 7777  
Device Reservation State : PR SHARED  
Reservation will be cleared on the device. Do you want to continue y/n:y
```

หาก คุณรันเคียวรีตอนนี้ ผลลัพธ์จะแสดง Device Reservation State เป็น NO RESERVE

```
# devrsrv -c query -l hdisk0
```

```
Device Reservation State Information
```

```
=====
```

```
Device Name           : hdisk0  
Device Open           : NO
```

```
ODM Reservation Policy   : PR SHARED
ODM PR Key Value         : 7777
Device Reservation State : NO RESERVE
Registered PR Keys      :
555
1777
PR Capabilities Byte[2]  : 0xd SIP_C  ATP_C  PTPL_C
PR Capabilities Byte[3]  : 0x0
PR Types Supported       : NOT VALID
```

### การดำเนินการกับเซอวิส CLEAR

หากต้องการรีเซ็ตการสำรองอย่างต่อเนื่องและต้องการถอนการลงทะเบียนทั้งหมด จากเซิร์ฟเวอร์อุปกรณ์ที่ใช้การดำเนินการกับเซอวิส CLEAR โดยใช้พาร I/O ที่ลงทะเบียนแล้วด้วยคีย์ 555 ให้ป้อนคำสั่งต่อไปนี้:

```
# devrsrv -c prout -s 3 -r 555 -l hdisk0
```

```
Device Reservation State Information
```

```
=====
```

```
Device Name           : hdisk0
Device Open On Current Host? : YES
ODM Reservation Policy : PR SHARED
ODM PR Key Value       : 5555
Device Reservation State : PR SHARED
Reservation will be cleared on the device. Do you want to continue y/n:y
```

หากคุณรันเคียวรีในตอนนี้ การสำรองอย่างต่อเนื่องจะถูกกรีต และการลงทะเบียนจะถูกถอนออกจากอุปกรณ์

```
# devrsrv -c query -l hdisk0
```

```
Device Reservation State Information
```

```
=====
```

```
Device Name           : hdisk0
Device Open           : NO
ODM Reservation Policy : PR SHARED
ODM PR Key Value       : 5555
Device Reservation State : NO RESERVE
Registered PR Keys     : No Keys Registered
PR Capabilities Byte[2] : 0xd SIP_C  ATP_C  PTPL_C
PR Capabilities Byte[3] : 0x0
PR Types Supported     : NOT VALID
```

### แอ็คชันเซอวิส PREEMPT และ PREEMPT\_ABORT

หากต้องการครอบครองการสำรองอย่างต่อเนื่องที่ถูกจัดการด้วยผู้จัดการ 444 โดย IT-nexus อื่นๆ กับคีย์ 777 ที่ลงทะเบียนแล้ว ให้ป้อนคำสั่งต่อไปนี้:

```
# devrsrv -c prout -s 4 -r 777 -k 444 -t 2 -l hdisk0
```

ก่อนที่จะคุณจะรันคำสั่ง #devrsrv -c prout -s 4 -r 777 -k 444 -t 2 -l hdisk0 เอาต์พุตเคียวรีจะถูกแสดงดังต่อไปนี้:

```
# devrsrv -c query -l hdisk0
```

```
Device Reservation State Information
```

```
=====
```

```
Device Name           : hdisk0
```

```

Device Open          : NO
ODM Reservation Policy : PR SHARED
ODM PR Key Value     : 7777
Device Reservation State : PR EXCLUSIVE
PR Generation Value  : 5
PR Type              : PR_WE (WRITE EXCLUSIVE)
PR Holder Key Value  : 444
Registered PR Keys   :
777
444
PR Capabilities Byte[2] : 0xd SIP_C ATP_C PTPL_C
PR Capabilities Byte[3] : 0x0
PR Types Supported    : NOT VALID

```

หลังจากที่คุณรันคำสั่ง #devrsrv -c prout -s 4 -r 777 -k 444 -t 2 -l hdisk0 เอาต์พุตเดียวที่แสดงว่า การสำรองที่ถูกครอบครองโดย IT-nexus กับคีย์ 777 และคีย์ 444 ถูกถอนการลงทะเบียน

```
# devrsrv -c query -l hdisk0
```

```

Device Reservation State Information
=====
Device Name          : hdisk0
Device Open          : NO
ODM Reservation Policy : PR SHARED
ODM PR Key Value     : 7777
Device Reservation State : PR EXCLUSIVE
PR Generation Value  : 6
PR Type              : PR_EA (EXCLUSIVE ACCESS)
PR Holder Key Value  : 777
Registered PR Keys   :
777
PR Capabilities Byte[2] : 0xd SIP_C ATP_C PTPL_C
PR Capabilities Byte[3] : 0x0
PR Types Supported    : NOT VALID

```

### การดำเนินการ RELEASE สำหรับนโยบาย SINGLE PATH RESERVE

หากต้องการรีเซ็ตการสำรองบนอุปกรณ์ hdisk0 ให้ป้อนคำสั่งต่อไปนี้:

- สถานการณ์ที่ 1: โฮสต์ปัจจุบันคือเจ้าของการสำรอง

```
# devrsrv -c query -l hdisk0
```

```

Device Reservation State Information
=====
Device Name          : hdisk0
Device Open On Current Host? : YES
ODM Reservation Policy : SINGLE PATH RESERVE
Device Reservation State : SINGLE PATH RESERVE
Path Id of Reserved Path : 0

```

```
# devrsrv -c release -l hdisk0
```

```

Device Reservation State Information
=====
Device Name          : hdisk0
Device Open On Current Host? : YES

```

```
ODM Reservation Policy      : SINGLE PATH RESERVE
Device Reservation State     : SINGLE PATH RESERVE
Device is currently Open on this host by a process.Do you want to continue y/n:y
Command Successful
Reservation cleared on the device. Query operation may not work properly.
Close the application that holds the reservation and retry.
```

- สถานการณ์จำลองที่ 2: โฮสต์ปัจจุบันไม่ใช่เจ้าของการสำรอง

```
# devrsrv -c query -l hdisk0
```

```
Device Reservation State Information
=====
Device Name          : hdisk0
Device Open On Current Host? : NO
ODM Reservation Policy      : SINGLE PATH RESERVE
Device Reservation State     : SINGLE PATH RESERVE
```

Because the current host does not own the reservation on the device,  
try the force option if you want to break the reservation.

```
# devrsrv -f -l hdisk0
```

อุปกรณ์ ถูกสำรองแล้วโดยใช้การสำรองพาธ-เดียว โดยโฮสต์อื่น

```
Device Reservation State Information
=====
Device Name          : hdisk0
Device Open On Current Host? : NO
ODM Reservation Policy      : SINGLE PATH RESERVE
Device Reservation State     : SINGLE PATH RESERVE
Reservation will be cleared on the device. Do you want to continue y/n:y
```

หลังจาก คุณรันคำสั่งรีลีสเรียบร้อยแล้ว อีอพชั่นเคียวรีต้องแสดง NO RESERVE เป็นสถานะการสำรองอุปกรณ์

```
# devrsrv -c query -l hdisk0
```

```
Device Reservation State Information
=====
Device Name          : hdisk0
Device Open On Current Host? : NO
ODM Reservation Policy      : SINGLE PATH RESERVE
Device Reservation State     : NO RESERVE
```

## โหมดการบังคับ

อุปกรณ์ hdisk0 ถูกสำรองด้วยคีย์ 777 จากพาธ I/O อื่น หากต้องการรีลีสการสำรองนี้จากโคลเอ็นต์อื่น ให้ป้อนคำสั่งต่อไปนี้:

```
# devrsrv -f -l hdisk0
```

```
Device Reservation State Information
=====
Device Name          : hdisk16
Device Open On Current Host? : NO
ODM Reservation Policy      : PR SHARED
```

```
ODM PR Key Value      : 5555
Device Reservation State : PR SHARED
Reservation will be cleared on the device. Do you want to continue y/n:y
Command Successful
```

ก่อนที่คุณจะรันคำสั่ง # devrsrv -f -l hdisk0 เคียวรีจะแสดงเอาต์พุตต่อไปนี้:

```
# devrsrv -c query -l hdisk0
```

Device Reservation State Information

```
=====
Device Name           : hdisk0
Device Open           : NO
ODM Reservation Policy : PR SHARED
ODM PR Key Value      : 5555
Device Reservation State : PR EXCLUSIVE
PR Generation Value   : 1
PR Type               : PR_WE (WRITE EXCLUSIVE)
PR Holder Key Value   : 777
Registered PR Keys    :
777
PR Capabilities Byte[2] : 0xd SIP_C ATP_C PTPL_C
PR Capabilities Byte[3] : 0x0
PR Types Supported    : NOT VALID
```

หลังจากที่คุณเรียกใช้ คำสั่ง # devrsrv -f -l hdisk0 เอาต์พุตระบุว่า อุปกรณ์ไม่ถูกจองแล้ว

```
# devrsrv -c query -l hdisk0
```

Device Reservation State Information

```
=====
Device Name           : hdisk16
Device Open On Current Host? : NO
ODM Reservation Policy : PR SHARED
ODM PR Key Value      : 5555
Device Reservation State : NO RESERVE
Registered PR Keys    : No Keys Registered
PR Capabilities Byte[2] : 0x0
PR Capabilities Byte[3] : 0x0
PR Types Supported    : NOT VALID
```

ข้อมูลที่เกี่ยวข้อง:

Multiple Path I/O



T10 Technical Committee

---

## คำสั่ง df

### วัตถุประสงค์

รายงานข้อมูลเกี่ยวกับพื้นที่ว่างบนระบบไฟล์ เอกสารนี้อธิบาย คำสั่ง AIX df เช่นเดียวกับ System V ในเวอร์ชันของ df

## ไวยากรณ์

```
df [[-P] | [-I|-M|-i|-t|-v]] [-c] [-T {local|remote|fstype}] [-F {output1 output2 output3 ...}] [-k] [-m] [-g] [-s] [FileSystem ..|File...]
```

## คำอธิบาย

คำสั่ง `df` แสดงข้อมูล เกี่ยวกับพื้นที่ว่างรวมและพื้นที่ว่างที่มีอยู่บนระบบไฟล์ พารามิเตอร์ `FileSystem` ระบุชื่อของอุปกรณ์ซึ่งมีระบบไฟล์อยู่ได้เรียกทอริซึ่งระบบไฟล์ถูกเมาท์ หรือ ชื่อพาสส์เวิร์ดของระบบไฟล์ พารามิเตอร์ `File` ระบุไฟล์หรือไดเรกทอรีที่ไม่ใช่จุดเมาท์ ถ้ามีการระบุพารามิเตอร์ `File` คำสั่ง `df` แสดง ข้อมูลสำหรับระบบไฟล์ซึ่งมีไฟล์หรือไดเรกทอรีอยู่ ถ้าคุณไม่ได้ระบุพารามิเตอร์ `FileSystem` หรือ `File` คำสั่ง `df` แสดงข้อมูล สำหรับระบบไฟล์ที่เมาท์ขณะนี้ทั้งหมด สถิติระบบไฟล์ถูกแสดงในหน่วยบล็อก 512-ไบต์โดยดีฟอลต์

คำสั่ง `df` ดึงข้อมูลสถิติพื้นที่ว่าง ระบบไฟล์จากการเรียกระบบ `statfs` อย่างไรก็ตาม การระบุแฟล็ก `-s` รับข้อมูลสถิติจากโปรแกรมช่วยเหลือระบบไฟล์ virtual file system (VFS) ถ้าคุณไม่ได้ระบุ อาร์กิวเมนต์ด้วยแฟล็ก `-s` และโปรแกรมช่วยเหลือล้มเหลวในการ รับข้อมูลสถิติ สถิติการเรียกระบบ `statfs` จะถูก นำมาใช้ภายใต้เงื่อนไขที่มีข้อยกเว้นบางประการ เช่นระบบไฟล์กำลัง ถูกแก้ไขขณะคำสั่ง `df` กำลังรันอยู่ สถิติ ที่แสดงโดยคำสั่ง `df` อาจไม่ ตรง

**หมายเหตุ:** ในบางระบบไฟล์แบบรีโมต เช่น Network File System (NFS), ไม่มีข้อมูลทั้งหมดที่คำสั่ง `df` ต้องการ คำสั่ง `df` พิมพ์ข้อมูลว่าง สำหรับสถิติที่เซิร์ฟเวอร์ไม่มี

คำสั่ง `df` ไม่สนับสนุน NFSv4 filesystems อย่างเต็มรูปแบบ ใช้คำสั่ง `nfs4cl` เพื่อแยกบล็อกและข้อมูล

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-c</code>	แสดงเอาต์พุตในรูปแบบที่คั่นด้วยเครื่องหมายโคลอน
<code>-F { output1 output2 output3 ... }</code>	แสดงเฉพาะค่าที่ระบุโดยส่วนหัวในพารามิเตอร์เอาต์พุต โดย ดีฟอลต์ ระบบไฟล์และส่วนหัวที่บล็อกจัดสรรเปิดอยู่เสมอ
<code>-g</code>	ค่าต่อไปนี้เป็นค่าที่ยอมรับได้สำหรับส่วนหัว:
<code>-i</code>	<code>%m</code> ติดตั้งบน
<code>-I</code>	<code>%u</code> ใช้
<code>-k</code>	<code>%z</code> เปอร์เซ็นต์ที่ใช้
<code>-m</code>	<code>%f</code> ว่าง
<code>-M</code>	<code>%l</code> Inodes ที่ใช้
	<code>%n</code> Inodes ที่ว่าง
	<code>%p</code> เปอร์เซ็นต์ของ inodes ที่ใช้
<code>-g</code>	แสดงสถิติในหน่วยบล็อก GB ค่าเอาต์พุต สำหรับสถิติระบบไฟล์จะเป็นตัวเลขอิงดัชนีเนื่องจาก ค่าของแต่ละหน่วยเป็นไบต์มีค่าสูงมาก
<code>-i</code>	แสดงจำนวน inodes ที่ใช้และเปอร์เซ็นต์ของ inodes ที่ใช้สำหรับระบบไฟล์ เอาต์พุตนี้เป็นดีฟอลต์เมื่อระบบไฟล์ที่ระบุ ถูกเมาท์
<code>-I</code>	แสดงข้อมูลจำนวนรวมของบล็อก พื้นที่ที่ใช้ พื้นที่ว่าง เปอร์เซ็นต์ของพื้นที่ที่ใช้ และจุดเมาท์สำหรับ ระบบไฟล์
<code>-k</code>	แสดงสถิติในหน่วยบล็อก 1024-ไบต์
<code>-m</code>	แสดงสถิติในหน่วยบล็อก MB ค่าเอาต์พุต สำหรับสถิติระบบไฟล์จะเป็นตัวเลขอิงดัชนีเนื่องจาก ค่าของแต่ละหน่วยเป็นไบต์มีค่าสูงมาก
<code>-M</code>	แสดงข้อมูลจุดเมาท์สำหรับระบบไฟล์ในคอลัมน์ ที่สอง

**ไอเท็ม**      **คำอธิบาย**  
**-P**            แสดงข้อมูลเกี่ยวกับระบบไฟล์ในรูปแบบเคลื่อนย้ายได้ของ POSIX

เมื่อระบุแฟล็ก **-P** บรรทัดส่วนหัวจะแสดง ดังนี้:  
Filesystem 512-blocks Used Available Capacity Mounted on\n

ถ้าระบุแฟล็ก **-k**, **-m** หรือ **-g** เพิ่มจากแฟล็ก **-P** ส่วนหัวคอลัมน์ 512-บล็อก ถูกแทนที่โดยหน่วย ต่อมา ขึ้นอยู่กับแฟล็กที่ใช้กับแฟล็ก **-P**

สถิติระบบไฟล์ถูกแสดงในหนึ่งบรรทัดใน ลำดับดังต่อไปนี้:

*FileSystem, TotalSpace, UsedSpace, FreeSpace, UsedPercentage, MountPoint*

**-s**            แสดงสถิติบนระบบไฟล์ JFS หรือ Enhanced JFS ที่ไม่ไดเมท โดยอาร์กิวเมนต์บรรทัดคำสั่ง ถ้าไม่มีการระบุอาร์กิวเมนต์ แฟล็ก **-s** จะไม่มีผล ถ้าระบบไฟล์ที่ระบุโดยอาร์กิวเมนต์ ถูกเมาท์อยู่ในขณะนี้หรืออาร์กิวเมนต์เป็นไฟล์ แฟล็ก **-s** ไม่มีผลสำหรับอาร์กิวเมนต์นั้น เมื่อต้องการรวบรวมสถิติบนระบบไฟล์ที่ไม่ไดเมท อาร์กิวเมนต์ต้องเป็นจุดเมาท์ระบบไฟล์ JFS หรือ Enhanced JFS หรืออุปกรณ์, ระบบไฟล์ต้องถูกแสดงรายการใน */etc/filesystems*, และผู้ใช้ต้องมีสิทธิ์การเข้าถึงในการอ่าน อุปกรณ์

**-t**            รวมตัวเลขสำหรับผลรวมพื้นที่ที่จัดสรรในเอาต์พุต

**-T { local | remote | vfstype }**      กรองเอาต์พุตโดยใช้นิคมของระบบไฟล์ แฟล็กนี้สามารถมีพารามิเตอร์อย่างใดอย่างหนึ่งต่อไปนี้:

**local**      แสดงเฉพาะระบบไฟล์ Journaled File System (JFS) และ Enhanced Journaled File System (JFS2)

**remote**      แสดงระบบไฟล์ที่ไม่ใช่โลคัลทั้งหมด

**vfstype**      แสดงระบบไฟล์ของ virtual file system (VFS) เฉพาะเท่านั้น ตัวอย่างเช่น JFS, JFS2, Network File System เวอร์ชัน 4 (NFSv4) และอื่นๆ

**-v**            แสดงข้อมูลทั้งหมดสำหรับระบบไฟล์ที่ระบุ

ค่าของพารามิเตอร์เอาต์พุตที่มีแฟล็ก **-m** และ **-g** จะถูกปิดพิเศษเป็นค่าใกล้เคียง ตัวเลขทศนิยมตำแหน่งที่สองมากที่สุด ถ้ามีการระบุแฟล็กใดแฟล็กหนึ่ง **-k**, **-m** และ **-g** หรือทั้งสองแฟล็ก แฟล็กที่ระบุสุดท้าย จะมีผล

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

**ไอเท็ม**      **คำอธิบาย**  
**0**            ดำเนินการเป็นผลสำเร็จ  
**>0**         มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการแสดงข้อมูลเกี่ยวกับระบบไฟล์ที่เมาท์ทั้งหมด ให้ป้อน:

```
df
```

ถ้าระบบของคุณมีระบบไฟล์ที่เมาท์ */*, */usr*, */site* และ */usr/venus* เอาต์พุตจากคำสั่ง **df** ลักษณะเหมือนดังต่อไปนี้:

```
Filesystem 512-blocks Free   %Used   Iused   %Iused  Mounted on
/dev/hd0   19368   9976   48%     4714    5%      /
/dev/hd1   24212   4808   80%     5031   19%     /usr
/dev/hd2    9744   9352    4%     1900    4%     /site
/dev/hd3    3868   3856    0%      986    0%     /usr/venus
```

2. เมื่อต้องการแสดงข้อมูลเกี่ยวกับระบบไฟล์ */test* ในบล็อก 1024-ไบต์, ให้ป้อน:

```
df -k /test
```

```
Filesystem    1024 blocks   Free   %Used   Iused   %Iused  Mounted on
/dev/1v11      16384   15824    4%      18     1%     /tmp/ravi1
```

เมื่อต้องการแสดงสถิติระบบไฟล์ในบล็อกดิสก์ 1024-ไบต์

3. เมื่อต้องการแสดงข้อมูลเกี่ยวกับระบบไฟล์ /test ในบล็อก MB ให้ป้อน:

```
df -m /test
```

Filesystem	MB blocks	Free	%Used	Iused	%Iused	Mounted on
/dev/lv11	16.00	15.46	4%	18	1%	/tmp/ravi1

ข้อมูลนี้แสดงสถิติระบบไฟล์ในบล็อกดิสก์ MB ที่พิเศษเป็นค่าใกล้เคียง ตัวเลขทศนิยมตำแหน่งที่สองมากที่สุด

4. เมื่อต้องการแสดงข้อมูลเกี่ยวกับระบบไฟล์ /test ในบล็อก GB ให้ป้อน:

```
df -g /test
```

Filesystem	GB blocks	Free	%Used	Iused	%Iused	Mounted on
/dev/lv11	0.02	0.02	0%	18	1%	/tmp/ravi1

ข้อมูลนี้แสดงสถิติระบบไฟล์ในบล็อกดิสก์ GB ที่พิเศษเป็นค่าใกล้เคียง ตัวเลขทศนิยมตำแหน่งที่สองมากที่สุด

5. เมื่อต้องการแสดงพื้นที่ที่มีในระบบไฟล์ซึ่งเป็นที่ตั้งไดเรกทอรีปัจจุบัน ของคุณ ให้ป้อน:

```
cd/  
df .
```

เอาต์พุตจากคำสั่งนี้มีลักษณะเหมือนดังนี้:

Device	512-blocks	free	%used	iused	%iused	Mounted on
/dev/hd4	19368	9976	48%	4714	5%	/

6. เมื่อต้องการแสดงเอาต์พุตในรูปแบบที่ค้นด้วยเครื่องหมายโคลอน ให้ป้อน:

```
df -c
```

เอาต์พุตคล้ายกับตัวอย่าง ต่อไปนี้:

```
Filesystem:512-blocks:Free:%Used:Iused:%Iused:Mounted on  
/dev/hd4:491520:113168:77%:9930:42%:/  
/dev/hd2:5046272:27696:100%:43014:86%:/usr
```

7. เมื่อต้องการแสดงข้อมูลเกี่ยวกับระบบไฟล์ทั้งหมดที่ติดตั้งแบบโลคัล ให้ป้อน:

```
df -T local
```

เอาต์พุตคล้ายกับตัวอย่าง ต่อไปนี้:

Filesystem	512-blocks	Free	%Used	Iused	%Iused	Mounted on
/dev/hd4	5898240	2104184	65%	16390	7%	/
/dev/hd2	7602176	1698696	78%	56001	23%	/usr
/dev/hd9var	3014656	2190976	28%	10987	5%	/var
/dev/hd3	2883584	2137928	26%	1213	1%	/tmp
/dev/hd1	655360	645240	2%	1727	3%	/home
/dev/hd11admin	262144	261384	1%	5	1%	/admin
/proc	-	-	-	-	-	/proc
/dev/hd10opt	786432	362672	54%	8926	18%	/opt
/dev/livedump	524288	523552	1%	4	1%	/var/adm/ras/livedump
/aha	-	-	-	328	2%	/aha

8. เมื่อต้องการแสดงข้อมูลเกี่ยวกับระบบไฟล์ JFS2 ทั้งหมด ให้ป้อน:

```
df -T jfs2
```

เอาต์พุตคล้ายกับตัวอย่าง ต่อไปนี้:

Filesystem	512-blocks	Free	%Used	Iused	%Iused	Mounted on
/dev/hd4	5898240	2104184	65%	16390	7%	/
/dev/hd2	7602176	1698696	78%	56001	23%	/usr
/dev/hd9var	3014656	2190976	28%	10987	5%	/var
/dev/hd3	2883584	2137928	26%	1213	1%	/tmp
/dev/hd1	655360	645240	2%	1727	3%	/home
/dev/hd11admin	262144	261384	1%	5	1%	/admin
/dev/hd10opt	786432	362672	54%	8926	18%	/opt
/dev/livedump	524288	523552	1%	4	1%	/var/adm/ras/livedump

9. เมื่อต้องการแสดงข้อมูล free, used และ mounted on เกี่ยวกับระบบไฟล์ JFS2 ทั้งหมด ให้ป้อน:

```
df -T jfs2 -F %f %u %m
```

เอาต์พุตคล้ายกับตัวอย่าง ต่อไปนี้:

Filesystem	512-blocks	Free	%Used	Mounted on
/dev/hd4	5898240	2104184	65%	/
/dev/hd2	7602176	1698696	78%	/usr
/dev/hd9var	3014656	2190976	28%	/var
/dev/hd3	2883584	2137928	26%	/tmp
/dev/hd1	655360	645240	2%	/home
/dev/hd11admin	262144	261384		/admin
/dev/hd10opt	786432	362672		/opt
/dev/livedump	524288	523552		/var/adm/ras/livedump

## ไฟล์

### ไอเท็ม

/etc/filesystems  
/etc/vfs

### คำอธิบาย

แสดงรายการของระบบไฟล์ที่รู้จัก และกำหนดคุณสมบัติไว้  
มีคำอธิบายของชนิดของระบบไฟล์เสมือน

## คำสั่ง System V df

### วัตถุประสงค์

รายงานจำนวนบล็อกดิสก์และไฟล์ที่ว่าง

### ไวยากรณ์

```
/usr/sysv/bin/df [-a][-l][[[-e][-g][-n]]][[-i][-v]]|-t][FileSystem ...][File ...]
```

### คำอธิบาย

คำสั่ง df แสดงข้อมูล เกี่ยวกับพื้นที่ว่างรวมและพื้นที่ว่างที่มีอยู่บนระบบไฟล์ สถิติระบบไฟล์ถูกแสดง ในหน่วยบล็อก 512-  
ไบต์

### แฟล็ก

ไอเท็ม	คำอธิบาย
-a	ดำเนินการดำเนินการดีฟอลต์ และพิมพ์จุดเมาท์ ชื่ออุปกรณ์ จำนวนบล็อกที่ว่างและจำนวน inodes ที่ใช้(ไฟล์)
-e	พิมพ์เฉพาะจำนวนไฟล์ที่ว่าง
-g	พิมพ์โครงสร้าง statvfs ทั้งหมด อีพซันนี้เขียนทับอีพซัน -a, -e, -i, -n, -t และอีพซัน -v ตัวเลขสำหรับบล็อก ที่มี จำนวนรวมและพื้นที่ว่าง ถูกรายงานในบล็อก 512 ไบต์
-i	แสดงจำนวนรวมของ inodes จำนวนของ inodes ที่ว่าง จำนวนของ inodes ที่ใช้ และเปอร์เซ็นต์ของ inodes ที่ใช้อยู่
-l	รายงานระบบไฟล์โลคัลเท่านั้น
-n	พิมพ์ชนิดของระบบไฟล์
-t	รายงานผลรวมตัวเลขบล็อกที่จัดสรร
-v	รายงานเปอร์เซ็นต์ของบล็อกที่ใช้และจำนวนของ บล็อกที่ใช้และที่ว่าง

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
ไฟล์	พารามิเตอร์ File ระบุไฟล์หรือไดเรกทอรีที่ไม่ใช่จุดเมาท์ ถ้ามีการระบุพารามิเตอร์ File คำสั่ง df แสดงข้อมูลสำหรับระบบไฟล์ซึ่งมีไฟล์หรือไดเรกทอรีอยู่
FileSystem	พารามิเตอร์ FileSystem ระบุชื่อของอุปกรณ์ซึ่งมีระบบไฟล์อยู่ไดเรกทอรีซึ่งระบบไฟล์ถูกเมาท์ หรือชื่อพารามิเตอร์ของระบบไฟล์

หมายเหตุ: ถ้าไม่มีการระบุพารามิเตอร์ FileSystem หรือ File คำสั่ง df จะแสดงข้อมูลสำหรับระบบไฟล์ที่เมาท์ในขณะนี้ทั้งหมด

## สถานะออก

- 0 คำสั่งเสร็จสิ้นสมบูรณ์
- >0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการแสดงข้อมูลเกี่ยวกับระบบไฟล์ที่เมาท์ทั้งหมดให้ป้อน:

```
/usr/sysv/bin/df
```

เอาต์พุตคล้ายกับตัวอย่างต่อไปนี้:

```
/                (/dev/hd4      ):   19656 blocks   1504 files
/usr             (/dev/hd2      ):  1139904 blocks  20254 files
/var            (/dev/hd9var   ):   23096 blocks    512 files
/tmp            (/dev/hd3      ):    2464 blocks    204 files
/home           (/dev/hd1      ):   44208 blocks    146 files
/proc           (/proc         ):         0 blocks         0 files
/opt            (/dev/hd10opt  ):   13880 blocks    310 files
```

2. เมื่อต้องการแสดงข้อมูลเกี่ยวกับระบบไฟล์ซึ่งเป็นที่ตั้งไดเรกทอรีปัจจุบันของคุณให้ป้อน:

```
/usr/sysv/bin/df .
```

3. เมื่อต้องการแสดงจำนวนรวมของ inode จำนวนของ inodes ที่ว่างและ จำนวนของ inodes ที่ใช้ได้ในระบบไฟล์ที่เมาท์ทั้งหมดให้ป้อน:

```
/usr/sysv/bin/df -i
```

เอาต์พุตคล้ายกับตัวอย่างต่อไปนี้:

Mount Dir	Filesystem	iused	avail	itotal	%iused
/	/dev/hd4	1504	6688	8192	19%
/usr	/dev/hd2	20254	127202	147456	14%
/var	/dev/hd9var	512	3584	4096	13%
/tmp	/dev/hd3	204	5940	6144	4%
/home	/dev/hd1	146	14190	14336	2%
/proc	/proc	0	0	0	0
/opt	/dev/hd10opt	310	5834	6144	6%

4. เมื่อต้องการแสดงจำนวนรวมของบล็อก จำนวนของบล็อกที่ใช้ และจำนวนของบล็อกที่ว่างในระบบไฟล์ /tmp ให้ป้อน:

```
/usr/sysv/bin/df -v /tmp
```

5. เมื่อต้องการแสดงชนิดของระบบไฟล์ ให้ป้อน:

```
/usr/sysv/bin/df -n
```

6. เมื่อต้องการแสดงข้อมูล inode บนระบบไฟล์โลคัลทั้งหมด ให้ป้อน:

```
/usr/sysv/bin/df -i -l
```

7. เมื่อต้องการแสดงข้อมูลโครงสร้าง statvfs บนระบบไฟล์ทั้งหมด ให้ป้อน:

```
/usr/sysv/bin/df -g
```

8. เมื่อต้องการแสดงจำนวนของไฟล์ที่ว่างอยู่บนระบบไฟล์ ให้ป้อน:

```
/usr/sysv/bin/df -e
```

## ไฟล์

ไอเท็ม

/usr/sysv/bin/df

/etc/filesystems

คำอธิบาย

มีคำสั่ง System V df

มีข้อมูลระบบไฟล์

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fsck” ในหน้า 610

ข้อมูลที่เกี่ยวข้อง:

ไฟล์ระบบไฟล์

ระบบไฟล์

## คำสั่ง dfmounts

### วัตถุประสงค์

แสดงข้อมูลรีพอร์สที่เม้าท์

### ไวยากรณ์

```
dfmounts [ -F fstype ] [ -h ] [ server ... ]
```

## คำอธิบาย

คำสั่ง `dfmounts` พิมพ์ระบบโลคัลที่ถูก เมทแบบรีโมต โดยไคลเอ็นต์ผ่าน Network File System (NFS) และยังพิมพ์ รายการของไคลเอ็นต์ที่ได้เมทรีชอร์ส คำสั่ง `dfmounts` พิมพ์ส่วนหัวซึ่งต่อท้ายด้วยรายการของข้อมูลรีชอร์ส แยกกันด้วยอักขระ whitespace ภายในฟิลด์

สำหรับแต่ละรีชอร์ส จะมีการแสดงฟิลด์ต่อไปนี้:

### RESOURCE

สำหรับ NFS, มีการใช้เครื่องหมายขีดคั่น "-"

### SERVER

ระบุเครื่องซึ่งรีชอร์สถูกเมท

### PATHNAME

ระบุพาทของรีชอร์สที่แบ่งใช้

### CLIENTS

รายการที่แยกด้วยคอมมา ของระบบที่มีรายชื่อเมทที่อยู่ในขณะนี้

## แฟล็ก

### ไอเท็ม

-F *fstype*  
-h

### คำอธิบาย

ระบุ File System Type (*fstype*) สนับสนุนเพียงชนิด `nfs` ของระบบไฟล์  
ไม่แสดงบรรทัดส่วนหัวในเอาต์พุตของ `dfmounts`

## พารามิเตอร์

### ไอเท็ม

*Server*

### คำอธิบาย

แสดงระบบบนเน็ตเวิร์กที่ระบบโลคัลสามารถใช้รีชอร์สได้ *Server* พิมพ์รีชอร์สที่ถูกทำให้พร้อมใช้จากเครื่องร่วมกับไคลเอ็นต์ปัจจุบัน โดยใช้แต่ละรีชอร์ส ถ้าไม่มีการระบุพารามิเตอร์ คำสั่ง `dfmounts` จะพิมพ์ข้อมูลโดยถือว่าเซิร์ฟเวอร์เป็นระบบโลคัล สามารถระบุชื่อเซิร์ฟเวอร์หลายชื่อได้ด้วยคำสั่ง `dfmounts`

## สถานะออก

0 คำสั่งเสร็จสิ้นสมบูรณ์

>0 มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

## ตัวอย่าง

1. เมื่อต้องการพิมพ์ข้อมูลรีชอร์สที่เมทบนระบบ "mercury" สำหรับ ชนิดระบบไฟล์ "nfs", ให้ป้อน:

```
dfmounts -F nfs mercury
```

2. เมื่อต้องการพิมพ์ข้อมูลรีชอร์สที่เมทโดยไม่มีส่วนหัวบนระบบสำหรับ ชนิดระบบไฟล์ "nfs", ให้ป้อน:

```
dfmounts -hF nfs
```

## ไฟล์

ไอเท็ม  
/usr/bin/dfmounts  
/usr/lib/fs/nfs/dfmounts  
/etc/vfs

คำอธิบาย  
มีคำสั่ง generic System V dfmounts  
มีคำสั่ง System V dfmounts สำหรับ nfs  
มีรายละเอียดสำหรับการนำระบบไฟล์เสมือน ไปใช้งาน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dfshares” ในหน้า 129

---

## คำสั่ง dfpd

### วัตถุประสงค์

จัดเตรียมสล็อตโหนดเกี่ยวกับเซิร์ฟเวอร์ที่มีการทำโหนดบาลานซ์กับ Load Manager

### ไวยากรณ์

```
/usr/sbin/dfpd [ -d ] [ -f ConfigurationFile ]
```

### คำอธิบาย

DFP daemon (**dfpd**) รับบนเซิร์ฟเวอร์ที่มีการโหนดบาลานซ์ และเตรียมข้อมูลสล็อตโหนดเซิร์ฟเวอร์กับ Load Manager ทำให้ Load Manager สามารถส่งการเชื่อมต่อในอนาคตไปที่เซิร์ฟเวอร์ที่พร้อมมากกว่า ซึ่งช่วยในการบาลานซ์โหนด

เมื่อ **dfpd** daemon เริ่ม จะมีการอ่านข้อมูลคอนฟิกูเรชัน จากไฟล์ที่ระบุในพารามิเตอร์ *ConfigurationFile* ถ้าไม่มีการระบุพารามิเตอร์ **dfpd** daemon จะอ่านข้อมูลคอนฟิกูเรชันจากไฟล์ */etc/dfpd.conf*

เมื่อเริ่มต้นทำงาน **dfpd** daemon จะรอรับการเชื่อมต่อ จาก Load Manager บนพอร์ตที่ระบุในไฟล์คอนฟิกูเรชัน

### ไฟล์คอนฟิกูเรชัน DFP daemon

ไฟล์ */etc/dfpd.conf* ถูกอัปเดตได้โดยการ แก้ไขไฟล์ รายการในไฟล์ */etc/dfpd.conf* รวมถึง ข้อมูลต่อไปนี้:

รายการคีย์ MD5 ระบุคีย์ลับ (มีค่าได้มากถึง 64 อักขระ) ซึ่งควร เหมือนกันระหว่างโหนด DFP, เซิร์ฟเวอร์และ Load Manager ตัวอย่าง ของรายการคีย์ MD5 คือ:

```
md5key 1234567890abcdefabcdef12345678901234567890abcdefabcdef1234567890
```

รายการ Load Manager listener ระบุพอร์ตซึ่งเซิร์ฟเวอร์ DFP รอการเชื่อมต่อ สำหรับการเชื่อมต่อ Load Manager ตัวอย่างของรายการ Load Manager คือ:

```
ldlistener 9503
```

รายการเวลาสุญเปล่าของโพลระบุช่วงเวลาระหว่างการคำนวณต่อเนื่อง ของเวลาสุญเปล่าของ CPU ตัวอย่างของรายการเวลาสุญเปล่าของโพลคือ:

```
pollidletime 30
```

เวลาสุญเปล่าที่คำนวณถูกคูณด้วยค่า *mfactor* ก่อนทำการรายงานเวลาไปที่ Load Manager ซึ่งมีประโยชน์ ในการตัดสินใจปริมาณการทำงานระหว่างเครื่องที่มีความสามารถต่างกัน ค่าดีฟอลต์คือจำนวน CPU บนโหนด ตัวอย่างของรายการ *mfactor* คือ:

mfactor 1

## แฟล็ก

ไอเท็ม

-d  
-f *ConfigurationFile*

คำอธิบาย

รันโหมดดีบั๊กและไม่กลายเป็นกระบวนการ daemon  
กำหนดให้ daemon ใช้ *ConfigurationFile* ที่ระบุ

## คำสั่ง dfscck

### วัตถุประสงค์

ตรวจและซ่อมแซมสองระบบไฟล์พร้อมกัน บนไดรฟ์ต่างกัน

### ไวยากรณ์

```
dfscck [ FlagList1 ]FileSystem1 [ FlagList2 ]FileSystem2
```

### คำอธิบาย

คำสั่ง `dfscck` ช่วยให้คุณสามารถตรวจสอบสองระบบไฟล์พร้อมกันบนสองไดรฟ์ที่ต่างกัน ใช้พารามิเตอร์ `FlagList1` และ `FlagList2` เพื่อส่งแฟล็ก และพารามิเตอร์สำหรับเซตของระบบไฟล์สองเซต สำหรับรายการของแฟล็กที่ใช้ได้สำหรับ `FlagList1` และ `FlagList2`, ดูที่ส่วนแฟล็ก ใช้ - (เครื่องหมายลบ) เพื่อแยก กลุ่มระบบไฟล์ ถ้าคุณระบุแฟล็กเป็นส่วนหนึ่งของอาร์กิวเมนต์

คำสั่ง `dfscck` อนุญาตให้คุณโต้ตอบกับคำสั่ง `fsck` สองคำสั่ง พร้อมกัน เพื่อช่วยเหลือ คำสั่ง `dfscck` แสดง ชื่อระบบไฟล์กับแต่ละข้อความ เมื่อตอบสนองกับคำถามจาก คำสั่ง `dfscck`,ให้นำหน้าการตอบสนองของคุณด้วย 1 หรือ 2 เพื่อระบุว่าคำตอบอ้างอิงถึงกลุ่มระบบไฟล์ที่หนึ่งหรือที่สอง

**ข้อควรสนใจ:** อย่าใช้คำสั่ง `dfscck` เพื่อตรวจสอบระบบไฟล์ `root`

## แฟล็ก

ไอเท็ม

-d*BlockNumber*

-f

-i*i-NodeNumber*

-n

คำอธิบาย

ค้นหาการอ้างอิงไปที่บล็อกดิสก์ที่ระบุ เมื่อใดก็ตามที่คำสั่ง `fsck` พบไฟล์ที่มีบล็อกที่ระบุ ซึ่งแสดงหมายเลข `i-node` และชื่อพาธทั้งหมดที่อ้างอิงถึง

ดำเนินการตรวจสอบอย่างรวดเร็ว ภายใต้สถานการณ์ปกติ ระบบไฟล์เดี่ยวที่ได้รับผล โดยการหยุดระบบโดยไม่ต้องมีการปิดระบบอย่างถูกต้อง คือระบบที่ถูกเม้าท์เมื่อระบบหยุด แฟล็ก `-f` พร้อมคำสั่ง `fsck` ไม่ต้องตรวจ ระบบไฟล์ที่ถูกยกเลิกการเม้าท์เสร็จสมบูรณ์ คำสั่ง `fsck` ระบุได้โดยอาศัยการตรวจสอบแฟล็ก `s_fmmod` ใน `superblock` ระบบไฟล์ แฟล็กนี้ถูกเซตเมื่อใดก็ตาม ที่ระบบไฟล์ถูกเม้าท์และเคลียร์ เมื่อมีการยกเลิกการเม้าท์เสร็จสมบูรณ์ ถ้าระบบไฟล์ถูกยกเลิกการเม้าท์เสร็จสมบูรณ์ แสดงว่าไม่มีปัญหาใด เนื่องจากระบบไฟล์ส่วนใหญ่จะถูกยกเลิกการเม้าท์เสร็จสมบูรณ์ การไม่ตรวจสอบระบบไฟล์เหล่านี้ สามารถลดเวลาการตรวจสอบได้

ค้นหาการอ้างอิงไปที่ `i-node` ที่ระบุ เมื่อใดก็ตามที่คำสั่ง `fsck` พบการอ้างอิงใดเรียกทอริไปที่ `i-node` ที่ระบุ, จะมีการแสดงชื่อพาธแบบเต็มของการอ้างอิง

ถือว่ามีการตอบกลับ `no` ไปที่คำถามที่ถาม ทั้งหมดโดยคำสั่ง `fsck`; ไม่เปิดระบบไฟล์ ที่ระบุเพื่อการเขียน

## ไอเท็ม

-o Options

### คำอธิบาย

ส่งอ็อปชันโดยแยกด้วยคอมมาไปที่คำสั่ง fsck อ็อปชันเหล่านี้ถือว่าเป็นการนำมาใช้จำเพาะระบบไฟล์, ยกเว้นว่าข้อมูลต่อไปนี้จะได้รับการสนับสนุนในขณะนี้สำหรับระบบไฟล์ทั้งหมด:

### mountable

ทำให้คำสั่ง fsck จบการทำงานสำเร็จ ส่งคืนค่า 0, ถ้าระบบไฟล์ดังกล่าวสามารถเมทาได้ (clean) ถ้าระบบไฟล์ไม่สามารถเมทาได้ คำสั่ง fsck จบการทำงานโดยส่งคืนค่า 8

mytype ทำให้คำสั่ง fsck จบการทำงานสำเร็จ (0) ถ้าระบบไฟล์ดังกล่าวเป็นชนิดเดียวกับที่ระบุไว้ในไฟล์ /etc/filesystems หรือโดยแฟล็ก -V บน บรรทัดคำสั่ง มิฉะนั้น ค่า 8 จะถูกส่งคืน ตัวอย่าง, fsck -o mytype -V

-p

jfs / จบการทำงานด้วยค่า 0 ถ้า / (ระบบไฟล์ root) เป็นระบบไฟล์ journaled ไม่แสดงข้อความเกี่ยวกับปัญหาใดๆ แต่ทำการแก้ไขโดยอัตโนมัติ แฟล็กนี้ไม่ได้รับไลเซนส์ทั้งหมดเหมือนกับแฟล็ก -y และมีประโยชน์สำหรับการตรวจสอบอัตโนมัติ เมื่อระบบถูกเริ่มต้นตามปกติ คุณควรใช้แฟล็กนี้เป็นส่วนหนึ่งของขั้นตอนการเริ่มต้นระบบ เมื่อใดก็ตามที่ระบบถูกรันโดยอัตโนมัติ และยังคงอนุญาตการตรวจสอบคูชานาโดยกลุ่ม

-t File

ระบุพารามิเตอร์ File เป็นไฟล์ว่าง บนระบบไฟล์อื่นที่ไม่ใช่ระบบที่ถูกตรวจสอบ, ถ้าคำสั่ง fsck ไม่สามารถรับหน่วยความจำได้อย่างเพียงพอเพื่อความเสถียรไว้ ถ้าคุณไม่ระบุแฟล็ก -t และคำสั่ง fsck ต้องการไฟล์เริ่มต้น คำสั่งจะพร้อมต่อชื่อของไฟล์ เริ่มต้น อย่างไรก็ตาม ถ้าคุณได้ระบุแฟล็ก -p คำสั่ง fsck จะไม่เสร็จสมบูรณ์ ถ้าไฟล์เริ่มต้นไม่ใช่ไฟล์พิเศษ ไฟล์จะถูกเอาออกเมื่อคำสั่ง fsck สิ้นสุด

-V VfsName

ใช้คำอธิบายของระบบไฟล์เสมือนที่ระบุโดยตัวแปร VfsName สำหรับระบบไฟล์แทนที่ใช้ไฟล์ /etc/filesystems ในการระบุรายละเอียด ถ้าไม่ระบุแฟล็ก -V VfsName ในบรรทัดคำสั่งไฟล์ /etc/filesystems จะถูกตรวจสอบและ vfs=Attribute ของ stanza ที่ตรงกันจะถือว่าเป็นชนิดระบบไฟล์ที่ถูกต้อง

-y

ถือว่าตอบกลับ yes กับคำถามทั้งหมดที่ถามโดยคำสั่ง fsck แฟล็กนี้ให้คำสั่ง fsck ดำเนินการที่คำสั่งเห็นว่าจำเป็นใช้แฟล็กนี้เฉพาะกับระบบไฟล์ที่เสียหายอย่างร้ายแรง

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

1. เมื่อต้องการตรวจสอบระบบไฟล์สองระบบพร้อมกัน ในสองไดรฟ์ที่ต่างกัน ให้ป้อน:

```
dfscck -p /dev/hd1 - -p /dev/hd7
```

คำสั่งนี้ตรวจสอบทั้งสองระบบไฟล์พร้อมกัน, ถ้าระบบไฟล์บนอุปกรณ์ /dev/hd1 และ /dev/hd7 อยู่บนสองไดรฟ์ที่ต่างกัน คุณยังสามารถระบุชื่อระบบไฟล์ที่พบในไฟล์ /etc/filesystems

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/sbin/dfsc	มีคำสั่ง dfsc
/etc/filesystems	แสดงรายการของระบบไฟล์ที่รู้จัก และกำหนดคุณสมบัติไว้
/etc/vfs	มีคำอธิบายของชนิดของระบบไฟล์เสมือน
/etc/rc	มีคำสั่ง (รวมทั้งคำสั่ง fsck) ที่ถูกรันเมื่อระบบเริ่มทำงาน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fsck” ในหน้า 610

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง istat

ไฟล์ filsys.h

ระบบไฟล์

## คำสั่ง dfshares

### วัตถุประสงค์

แสดงรายการรีซอร์สที่มีจากระบบรีโมต

### ไวยากรณ์

```
dfshares [-F FileSystemType] [-h] [Server ...]
```

### คำอธิบาย

คำสั่ง dfshares ให้ข้อมูลเกี่ยวกับรีซอร์ส ที่พร้อมใช้กับโฮสต์ผ่าน Network File System คำสั่ง dfshares พิมพ์บรรทัดส่วนหัวตามด้วยรายการของบรรทัดที่มีช่องว่างเป็นตัวแยกฟิลด์

สำหรับแต่ละรีซอร์ส จะมีการแสดงฟิลด์ต่อไปนี้:

#### RESOURCE

แสดงชื่อรีซอร์สที่ถูกเอ็กซ์พอร์ตในรูปแบบ server:path

#### SERVER

แสดงเครื่องที่กำลังจัดเตรียมรีซอร์ส

#### ACCESS

แสดงสิทธิเข้าถึงที่ได้รับกับระบบไคลเอ็นต์ อย่างไรก็ตาม dfshares ไม่สามารถกำหนดข้อมูลนี้สำหรับรีซอร์ส NFS และดังนั้นสร้างฟิลด์พร้อมกับเครื่องหมายขีดค้น ("-")

#### TRANSPORT

แสดงผู้ให้บริการการรับส่งข้อมูลผ่านรีซอร์สที่ถูกแบ่งใช้ อย่างไรก็ตาม dfshares ไม่สามารถกำหนดข้อมูลนี้สำหรับรีซอร์ส NFS และดังนั้นสร้างฟิลด์พร้อมกับเครื่องหมายขีดค้น ("-")

### แฟล็ก

ไอเท็ม  
-F *FileSystemType*  
-h

คำอธิบาย  
ระบุชนิด filesystem สนับสนุนเพียงชนิด nfs ของระบบไฟล์  
ไม่แสดงบรรทัดส่วนหัวในเอาต์พุตของ dfshares

## พารามิเตอร์

ไอเท็ม  
Server

คำอธิบาย  
แสดงระบบบนเครือข่ายที่ได้จัดเตรียม รีซอร์สให้กับเครื่องโลคัล ถ้าไม่มีการระบุพารามิเตอร์นี้ คำสั่ง dfshares จะ  
พิมพ์ข้อมูลสำหรับระบบโลคัล ด้วยตัวเอง สามารถระบุชื่อเซิร์ฟเวอร์ได้มากกว่าหนึ่งชื่อด้วย dfshares

## สถานะออก

0 คำสั่งดำเนินการเสร็จด้วยผลสำเร็จ

>0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการพิมพ์ข้อมูลรีซอร์สบนระบบ "mercury" สำหรับ nfs type filesystem, ให้ป้อน:

```
dfshares -F nfs mercury
```

2. เมื่อต้องการพิมพ์ข้อมูลรีซอร์สโดยไม่มีส่วนหัวบนระบบ ให้ป้อน:

```
dfshares -hF nfs
```

## ไฟล์

ไอเท็ม  
/usr/bin/dfshares  
/usr/lib/fs/nfs/dfshares  
/etc/vfs

คำอธิบาย  
มีคำสั่ง generic System V dfshares  
มีคำสั่ง System V dfshares command สำหรับระบบไฟล์ชนิด nfs  
มีรายละเอียดสำหรับการนำระบบไฟล์เสมือน ไปใช้งาน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dfmounts” ในหน้า 124

---

## คำสั่ง dhcraction

### วัตถุประสงค์

จัดเตรียม สคริปต์ที่รันทุกครั้งที่ไคลเอ็นต์อัปเดต lease ของตัวเอง

### ไวยากรณ์

```
/usr/sbin/dhcraction HostName DomainName IPAddress LeaseTime ClientID { A | PTR | BOTH | NONE } { NONIM | NIM }
```

### คำอธิบาย

คำสั่ง dhcraction จัดเตรียมวิธีเพื่ออัปเดตเซิร์ฟเวอร์ DNS โดยการเรียกใช้คำสั่ง nsupdate ด้วยลำดับของเหตุการณ์ที่เหมาะสมเพื่ออัปเดตเร็กคอร์ด A, เร็กคอร์ด PTR หรือทั้งคู่ คำสั่ง dhcraction ถูกเรียกโดย DHCP ไคลเอ็นต์และเซิร์ฟเวอร์ daemons

โดยถูกเรียกใช้จากสตริง updateDNS ค่าที่ตั้งนี้สามารถกำหนดค่าได้เนื่องจากในบางสภาวะแวดล้อม โดยเฉพาะสภาวะแวดล้อมที่หลากหลาย บางไคลเอนต์อาจไม่สามารถอัปเดตเร็กคอร์ด A หรือเร็กคอร์ด PTR การดำเนินการดีพอลต์เพื่อให้ไคลเอนต์อัปเดตเร็กคอร์ด A และเซิร์ฟเวอร์ อัปเดตเร็กคอร์ด PTR อีอ็อปชันอาจถูกตั้งค่าในไฟล์คอนฟิกูเรชัน daemon เพื่ออนุญาตให้ใช้นโยบายใดๆ ที่ผู้ดูแลระบบเครือข่าย ต้องการ

คำสั่ง `dhcraction` ยังรัน NIM และ DHCP พร้อมกัน เมื่อใช้คำสั่ง `dhcraction` กับพารามิเตอร์ NIM จะพยายามอัปเดตกับอีออบเจ็กต์ NIM เมื่อ IP แอดเดรสมีการเปลี่ยนแปลง แอ็คชันนี้จะทำให้แอ็คชัน ซิงค์กันตลอดเวลา เมื่อต้องการทำดังกล่าว การดำเนินการบางอย่างที่ค้างอยู่อาจต้องถูกยกเลิก อีออบเจ็กต์จะถูกทำหมายเหตุและข้อความจะถูกส่งไปยังคอนโซลของเครื่องหลัก อีออบเจ็กต์ต้องไม่ถูกรีเซ็ทบ่อย แอดเดรสไม่ควรเปลี่ยนแปลงบ่อยในสภาวะแวดล้อม DHCP เฉพาะไคลเอนต์เท่านั้นที่ต้องตั้งค่าอีอ็อปชัน NONIM

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
ClientID	ระบุ ID ไคลเอนต์ที่ต้องการใช้เมื่อคุณอัปเดต เซิร์ฟเวอร์ DNS
DomainName	ระบุชื่อโดเมนที่ต้องการใช้เมื่อคุณอัปเดตเซิร์ฟเวอร์ DNS
HostName	ระบุชื่อโฮสต์ที่ต้องการอัปเดตในเซิร์ฟเวอร์ DNS
IPAddress	ระบุ IP แอดเดรสเพื่อเชื่อมโยงกับชื่อโฮสต์ในเซิร์ฟเวอร์ DNS
LeaseTime	ระบุระยะเวลาของการเชื่อมโยงระหว่างชื่อโฮสต์และ IP แอดเดรสในเซิร์ฟเวอร์ DNS เป็นวินาที

## อีอ็อปชัน

ไอเท็ม	คำอธิบาย
A PTR BOTH NONE	ระบุเร็กคอร์ด (หากมี) ที่ต้องการอัปเดตในเซิร์ฟเวอร์ DNS
NONIM NIM	ระบุวาระคริบัติต้องรันเพื่อช่วยให้ NIM และ DHCP ได้ตอบอย่างถูกต้องหรือไม่ ซึ่งต้องตั้งค่าเป็น NIM บน เซิร์ฟเวอร์ DHCP เท่านั้น

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

การควบคุมการเข้าถึง: ผู้ใช้ใดๆ แต่อาจต้องเป็น root สำหรับแอ็คชัน NIM บางอย่าง

## ไฟล์

ไอเท็ม  
/usr/sbin/dhcpaction  
/etc/dhcpd.ini

คำอธิบาย  
มีคำสั่ง `dhcpaction`  
มีไฟล์คอนฟิกูเรชัน DHCP โคลเอ็นต์

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“dhcpsd Daemon” ในหน้า 139

“dhcprd Daemon” ในหน้า 135

### ข้อมูลที่เกี่ยวข้อง:

ไฟล์คอนฟิกูเรชัน bootp

TCP/IP แอดเดรสและการกำหนด พารามิเตอร์ – Dynamic Host Configuration Protocol

---

## dhcpd Daemon

### วัตถุประสงค์

สร้างโคลเอ็นต์ Dynamic Host Configuration Protocol (DHCP) เซิร์ฟเวอร์แอดเดรสและข้อมูลคอนฟิกูเรชันสำหรับเซิร์ฟเวอร์ DHCP

### ไวยากรณ์

เมื่อต้องการสร้าง DHCP โคลเอ็นต์โดยใช้ System Resource Controller (SRC):

```
startsrc -s dhcpd [-a Argument] ...
```

เมื่อต้องการสร้าง DHCP โคลเอ็นต์โดยไม่ใช้ SRC:

```
dhcpd [-f ConfigurationFile] [-i IPAddress] [-l LeaseFile] [-n] [-o OptionsFile] [-r] [-t Seconds] [-T Minutes]
```

### คำอธิบาย

`dhcpd daemon` สร้าง โคลเอ็นต์ DHCP โดยการตั้งค่า IP (Internet Protocol) addresses และพารามิเตอร์อื่น โดยใช้โปรโตคอล DHCP

โดยปกติ `dhcpd daemon` จะเริ่มทำงานโดยไฟล์ `/etc/rc.tcpip` ที่โดยทั่วไป จะรันเมื่อระบบเริ่มทำงาน โดยดีฟอลต์ จะถูกใส่หมายเหตุและไม้รัน เมื่อระบบเริ่มทำงาน มีอ็อปชัน System Management Interface Tool (SMIT) เพื่อเปิดใช้งาน DHCP โคลเอ็นต์

`dhcpd daemon` จะอ่านไฟล์คอนฟิกูเรชันและพยายามเริ่มทำงานและขอรับ IP แอดเดรส และอ็อปชันคอนฟิกูเรชันอื่นสำหรับอินเตอร์เฟซที่ระบุไว้ในไฟล์คอนฟิกูเรชัน `dhcpd daemon` รันในแบ็กกราวนด์ขณะที่ระบบเริ่มทำงาน โดยจะขอรับแอดเดรสที่ได้รับมาแล้วใหม่ตามที่จำเป็น

dhcpcd daemon ยังรัน ในโหมด DHCP Inform เมื่อมีการใช้แฟล็ก -i โดยการรันโหมดนี้ ไคลเอ็นต์สามารถดึงข้อมูลคอนฟิกูเรชัน จากเซิร์ฟเวอร์ DHCP โดยไม่ขอรับ IP ซึ่งมีประโยชน์สำหรับ แอดเดรสแบบสแตติก แต่ไม่ใช่สำหรับไอเท็มแบบไดนามิก เช่น พรินต์เซิร์ฟเวอร์และ อีพชัณอื่น เมื่อคุณใช้แฟล็ก -i กับ พารามิเตอร์ IP แอดเดรส dhcpcd daemon จะรัน หนึ่งครั้ง สำหรับแอดเดรสที่ระบุ

คำสั่ง refresh สามารถถูกใช้เพื่อ ทำให้ dhcpcd daemon อ่านไฟล์คอนฟิกูเรชัน ซ้ำ SIGHUP ยังอาจใช้ เพื่อขอรับการตอบกลับ แบบเดียวกัน

ไฟล์คอนฟิกูเรชันดีฟอลต์ สำหรับ dhcpcd daemon คือ /etc/dhcpcd.ini ซึ่งมีข้อมูล การเข้าระบบและเน็ตเวิร์กอินเทอร์เน็ตเฟส

คุณสามารถใช้พาธด่วนของ smit usedhcp ของ SMIT เพื่อรันคำสั่งนี้

## แฟล็ก

ไอเท็ม	คำอธิบาย
-f ConfigurationFile	ระบุไฟล์คอนฟิกูเรชันที่จะใช้ ดีฟอลต์คือไฟล์ /etc/dhcpcd.ini
-i IPAddress	ระบุว่า dhcpcd daemon ต้องใช้โหมด DHCP Inform IP แอดเดรสจะบอก DHCP ถึงอินเทอร์เน็ตเฟสที่จะรับ ข้อมูล คอนฟิกูเรชัน
-l LeaseFile	ระบุไฟล์ lease อื่น ไฟล์ lease ถูกสร้างขึ้นโดย ไคลเอ็นต์เมื่อได้รับ lease โดยดีฟอลต์ไฟล์ lease คือ /etc/dhcpc.db
-n	ป้องกันอินเทอร์เน็ตเฟสจากการถูกตั้งค่าใหม่เมื่อได้รับ แอดเดรสใหม่
-o OptionsFile	ระบุไฟล์อ็อปชัน โดยดีฟอลต์ ไฟล์อ็อปชันคือ /etc/dhcpc.opt
-r	เรียกใช้ไคลเอ็นต์ daemon และจากนั้นหยุดทำงานเมื่อรันหนึ่งครั้ง
-t Seconds	ระบุจำนวนวินาทีที่ DHCP รอก่อนที่จะรันตัวเอง อยู่เบื้องหลัง ซึ่งจะอนุญาตให้ระบบทำการบูตต่อ หาก ไม่พบ เซิร์ฟเวอร์ DHCP
-T Minutes	ระบุเวลาเป็นนาที หาก DHCP ไคลเอ็นต์ ไม่สามารถกำหนดค่าแอดเดรสสำหรับอินเทอร์เน็ตเฟส (เช่น เนื่องจาก เซิร์ฟเวอร์ DHCP ไม่พร้อมใช้งาน) ภายในค่าการหมดเวลานี้ ไคลเอ็นต์จะหยุดความพยายามเพิ่มเติม

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: คุณต้องมีสิทธิ์ root ในการรัน คำสั่งนี้

## ไฟล์

## ไอเท็ม

/usr/sbin/dhcpd  
/etc/dhcpd.ini  
/etc/services  
/etc/inetd.conf

## คำอธิบาย

มี **dhcpd** daemon  
มีไฟล์คอนฟิกูเรชันไคลเอ็นต์ที่พอลต์  
กำหนดซ็อกเก็ตและโปรโตคอลที่ใช้สำหรับอินเทอร์เน็ตเซอร์วิส  
กำหนดเซอร์วิสที่ถูกควบคุมโดย **inetd** daemon

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **startsrc**

คำสั่ง **inetd**

ไฟล์คอนฟิกูเรชันเซิร์ฟเวอร์ DHCP

TCP/IP daemons

---

## dhcpd6 Daemon

### วัตถุประสงค์

สร้าง Dynamic Host Configuration Protocol สำหรับไคลเอ็นต์ IPv6 (DHCPv6) ขอรับ IPv6 แอดเดรสและข้อมูลคอนฟิกูเรชันสำหรับโหนด IPv6 จากเซิร์ฟเวอร์ DHCPv6

### ไวยากรณ์

เมื่อต้องการเริ่มต้น DHCPv6 ไคลเอ็นต์ โดยใช้ System Resource Controller (SRC):

```
startsrc -s dhcpd6 [ -a Argument ] ...
```

เมื่อต้องการเริ่มต้น DHCPv6 ไคลเอ็นต์ โดยไม่ใช้ SRC:

```
dhcpd6 [-f ConfigurationFileName] [-u Client_duid_File] [-p ClientPort] [-t SolicitTimeout]
```

### คำอธิบาย

**dhcpd6** daemon สร้างไคลเอ็นต์ DHCPv6 โดยการตั้งค่า IPv6 (Internet Protocol version 6) addresses และพารามิเตอร์อื่นโดยใช้โปรโตคอล DHCPv6

**dhcpd6** daemon โดยปกติถูกเริ่มโดยไฟล์ `/etc/rc.net` ซึ่งโดยปกติรันเมื่อบูต โดยที่พอลต์ จะถูกใส่หมายเหตุและไม่รันเมื่อเครื่องเริ่มทำงาน **dhcpd6** daemon รันในแบ็กกราวนด์ขณะที่ระบบเริ่มทำงาน

**dhcpd6** daemon อ่านไฟล์คอนฟิกูเรชันและพยายามเรียกข้อมูลและรับ IPv6 และ อีพซันคอนฟิกูเรชันอื่นสำหรับอินเทอร์เน็ตเฟสที่ระบุภายในไฟล์คอนฟิกูเรชัน แอดเดรสที่ได้รับ จากเซิร์ฟเวอร์จะถูก renew ตามความจำเป็นโดยเซิร์ฟเวอร์

เมื่อไคลเอ็นต์ DHCPv6 ไม่มีเซิร์ฟเวอร์ DHCPv6 ที่กำหนด IPv6 addresses ของไคลเอ็นต์, ไคลเอ็นต์สามารถรับเฉพาะข้อมูลคอนฟิกูเรชัน เช่น รายการของเซิร์ฟเวอร์ DNS หรือเซิร์ฟเวอร์ NTP ที่สามารถใช้ได้ ซึ่งมีประโยชน์เมื่อโหนด ถูกกำหนดค่าด้วยสแตติกแอดเดรส

คำสั่ง **refresh** สามารถถูกใช้เพื่อ ทำให้ **dhcpd6** daemon อ่านไฟล์คอนฟิกูเรชันซ้ำ **SIGHUP** ยังอาจใช้ เพื่อขอรับการตอบกลับแบบเดียวกัน

ไฟล์คอนฟิกูเรชัน `dhcpcd6` ดีฟอลต์ คือ `/etc/dhcpv6/dhcpc6.conf` ซึ่งมีข้อมูล การเข้าระบบและเน็ตเวิร์กอินเทอร์เน็ตเฟส

## แฟล็ก

### ไอเท็ม

-f *ConfigurationFileName*  
-p *ClientPort*  
-t *SolicitTimeout*  
-u *Client\_ duid\_File*

### คำอธิบาย

ระบุไฟล์คอนฟิกูเรชันที่จะใช้ ดีฟอลต์ คือ `/etc/dhcpv6/dhcpc6.conf`  
ระบุไคลเอ็นต์พอร์ตที่จะใช้ ดีฟอลต์คือ 546  
ระบุเวลาจนกว่าไคลเอ็นต์เรียกขอข้อมูลคอนฟิกูเรชัน จากเซิร์ฟเวอร์ก่อนจบการทำงาน  
ระบุไฟล์ identifier ของไคลเอ็นต์ที่จะใช้ ดีฟอลต์ คือ `/etc/dhcpv6/dhcpc6.duid`

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

### ไอเท็ม

0  
>0

### คำอธิบาย

ดำเนินการเป็นผลสำเร็จ  
มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: คุณต้องมีสิทธิ์ `root` ในการรัน คำสั่งนี้

## ตัวอย่าง

- เมื่อต้องการเริ่มต้น DHCPv6 ไคลเอ็นต์ด้วยไฟล์คอนฟิกูเรชัน `dhcpcd6.conf` ที่อยู่ใน `/usr/local` ให้พิมพ์คำสั่งต่อไปนี้:  

```
startsrc -s dhcpcd6 -a "-f /usr/local/dhcpcd6.conf"
```

## Location

`/usr/sbin/dhcpcd6`

## ไฟล์

### ไอเท็ม

`/usr/sbin/dhcpcd6`  
`/etc/dhcpv6/dhcpc6.conf`  
`/etc/dhcpv6/dhcpc6.db`  
  
`/etc/dhcpv6/dhcpc6.duid`

### คำอธิบาย

มี `dhcpcd6` ไคลเอ็นต์ daemon  
มีไฟล์คอนฟิกูเรชันดีฟอลต์  
มีไฟล์ lease ของไคลเอ็นต์ ไฟล์นี้ถูกสร้าง โดยไคลเอ็นต์ daemon และไม่สามารถกำหนดคอนฟิกได้  
มีไฟล์ identifier ของไคลเอ็นต์ ไฟล์นี้ถูกสร้าง โดยไคลเอ็นต์ daemon และไม่สามารถกำหนดคอนฟิกได้

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `startsrc`

---

## dhcprd Daemon

### วัตถุประสงค์

ส่งต่อแพ็กเก็ต BOOTP และ Dynamic Host Configuration Protocol (DHCP) ออกจากเครือข่าย โลคัล

## ไวยากรณ์

เมื่อต้องการส่งต่อข้อมูล ไปยังเซิร์ฟเวอร์ DHCP โดยใช้ System Resource Controller (SRC):

```
startsrc -s dhcprd [ -a Argument ] [ -a Argument ] ...
```

เมื่อต้องการส่งต่อข้อมูลไปยังเซิร์ฟเวอร์ DHCP โดยไม่ใช้ SRC:

```
dhcprd [ -f ConfigurationFile ]
```

## คำอธิบาย

dhcprd daemon รอรับการกระจาย แพ็กเก็ต, รับแพ็กเก็ต และส่งต่อไปที่เซิร์ฟเวอร์ที่เหมาะสม จะป้องกันไม่ให้บรอดคาสต์ เผยแพร่ไปยัง เครือข่ายอื่น DHCP relay agent จะจัดการการส่งต่อแพ็กเก็ต บรอดคาสต์ของ DHCP และ BOOTP โคลเอ็นต์ ออกจากเครือข่ายโลคัล และไปยังชุดของเซิร์ฟเวอร์ แพ็กเก็ตเริ่มต้นที่ส่งโดย BOOTP หรือ DHCP โคลเอ็นต์จะบรอดคาสต์ บนโลคัลอินเตอร์เฟซของ ระบบโคลเอ็นต์ แพ็กเก็ตเหล่านี้ไม่ได้รับอนุญาตให้ผ่านเน็ตเวิร์กเกตเวย์ และเราเตอร์ BOOTP หรือ DHCP relay agent dhcprd daemon จะส่งแพ็กเก็ตเหล่านี้ไปยังเซิร์ฟเวอร์ที่เหมาะสม

เซิร์ฟเวอร์ DHCP จะอ่านไฟล์ /etc/services เพื่อกำหนดว่าพอร์ตใดที่ต้องใช้สำหรับการรับคำร้องขอ เซอร์วิสดีฟอลต์คือ dhcps เนื่องจากนี้เป็นพอร์ตเดียวกับที่ bootpd daemon ใช้ คุณสามารถรันได้เพียงหนึ่ง (dhcprd หรือ bootpd) daemon เท่านั้น หากคุณเลือก dhcprd คุณต้องยกเลิกการใส่หมายเหตุ bootp จากไฟล์ /etc/inetd.conf จากนั้นพิมพ์ refresh -s inetd บนบรรทัดรับคำสั่ง

หมายเหตุ: หาก bootpd daemon รันอยู่ โปรแกรมนี้ต้องหยุดทำงานก่อนที่คุณจะเริ่มต้น daemons

## แฟล็ก

ไทม์

-f ConfigurationFile

คำอธิบาย

ระบุไฟล์คอนฟิกูเรชันที่ใช้ ดีฟอลต์คือไฟล์ /etc/dhcprd.cnf

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไทม์ คำอธิบาย

0 ดำเนินการเป็นผลสำเร็จ

>0 มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: คุณต้องมีสิทธิ์ root ในการรัน คำสั่งนี้

## ไฟล์

## ไอเท็ม

/usr/sbin/dhcpd  
/etc/dhcpd.conf  
/etc/services  
/etc/inetd.conf

## คำอธิบาย

มี `dhcpd` daemon  
มีไฟล์คอนฟิกเรชันดีฟอลต์  
กำหนดซ็อกเก็ตและโปรโตคอลที่ใช้สำหรับ อินเทอร์เน็ตเซอร์วิส  
กำหนดเซอร์วิสที่ถูกควบคุมโดย `inetd` daemon

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `inetd`

TCP/IP แอดเดรสและการกำหนด พารามิเตอร์ – Dynamic Host Configuration Protocol

System Resource Controller

---

## คำสั่ง `dhcpsconf`

### วัตถุประสงค์

ปรับแต่งคอนฟิกเรชัน ของเซิร์ฟเวอร์ DHCP (Dynamic Host Configuration Protocol) ให้ง่ายขึ้นโดยใช้ ส่วนติดต่อผู้ใช้แบบกราฟิก (GUI)

### ไวยากรณ์

`dhcpsconf`

### คำอธิบาย

คำสั่ง `dhcpsconf` จะเปิด X Window System GUI ที่ช่วยให้ผู้ดูแลระบบเครือข่ายสามารถอ่าน บันทึก และแก้ไขไฟล์คอนฟิกเรชัน และยังช่วยให้คุณเริ่มต้นและเรียกข้อมูลสถิติจากเซิร์ฟเวอร์ที่รันอยู่

คำสั่ง `dhcpsconf` แสดง ชุดของรายการ รายการทางซ้ายแสดงอ็อปชัน และคีย์ที่มีอยู่ คำสั่ง `dhcpsconf` จะอ่านไฟล์ `/etc/options` เพื่อกำหนด อ็อปชันและคีย์พื้นฐานและเริ่มต้นด้วยข้อมูลเหล่านี้เป็น ชนิดรีซอร์สทั่วไป GUI ช่วยให้ผู้ดูแลระบบเน็ตเวิร์กกำหนดชุดของรีซอร์ส ที่มีการกำหนดชื่อ โดยเลือกปุ่มเมนูรีซอร์ส

รีซอร์สนิยามรีซอร์สช่วยให้ผู้บริหารเน็ตเวิร์ก สร้างอ็อปชันและข้อมูลเจาะจงที่อยู่บน เน็ตเวิร์ก ผู้ดูแลระบบเครือข่ายสามารถกำหนดและตั้งชื่อ เครือข่าย เครื่องพิมพ์ เซิร์ฟเวอร์ชื่อ เซิร์ฟเวอร์ DHCP และอ็อปเจ็กต์รีซอร์สอื่นที่ใช้ได้ เมื่อเสร็จสิ้นรีซอร์สใหม่เหล่านี้จะถูกเพิ่มเข้ากับคีย์และอ็อปชันที่แสดงบน พาเนลหลัก รีซอร์สเหล่านี้สามารถใช้เพื่อสร้างไฟล์คอนฟิกเรชันของเซิร์ฟเวอร์ หรือชุดของไฟล์คอนฟิกเรชันของเซิร์ฟเวอร์

GUI สตาร์ทด้วยมาตรฐานไฟล์ว่างเปล่า มาตรฐานไฟล์ อาจมีเซิร์ฟเวอร์เดี่ยวหรือนิยามของเซิร์ฟเวอร์จำนวนมาก และไฟล์ที่สามารถอ่านได้ของเซิร์ฟเวอร์จริง มาตรฐานไฟล์สามารถอ่านได้ โดยหนึ่งเซิร์ฟเวอร์ DHCP, แต่ข้อมูลเซิร์ฟเวอร์หลายชุดสามารถเก็บไว้ได้ ในไฟล์ ซึ่งทำให้ผู้ดูแลระบบเครือข่ายสามารถกำหนดคอนฟิก อิมเมจของเซิร์ฟเวอร์เดี่ยวของเครือข่าย สร้างชุดของเซิร์ฟเวอร์เพื่อจัดการข้อมูลชุดเดียวกัน และดูและดูแลรักษาข้อมูลทั้งหมดในไฟล์เดียว

อ็อปชันและคีย์จะถูกเพิ่มให้กับหน้าต่างเซิร์ฟเวอร์ โดย การเลือกคีย์หรืออ็อปชัน เลือกตำแหน่งในหน้าต่างแก้ไขที่จะกำหนดอ็อปชัน หรือคีย์ และเลือกปุ่มเพิ่มที่สอดคล้องส่วนของคีย์หรือ อ็อปชัน อ็อปชันถูกเพิ่มในหน้าต่างแก้ไข ที่ตำแหน่งที่ระบุ ถ้ารายการเป็นรีซอร์สที่มีชื่อกำหนดไว้ รีซอร์สจะถูกเพิ่ม ตามที่เป็นอยู่ หากไอเท็มเป็นหนึ่งในค่าดีฟอลต์มาตรฐาน ดังนั้นหน้าต่างที่ร้องขอคำสำหรับไอเท็มจะปรากฏขึ้น

เซิร์ฟเวอร์ DHCP จะถูกเพิ่มเช่นเดียวกับคีย์อื่น ยกเว้นจะระบุระบบในเครือข่ายที่รับผิดชอบสำหรับไอเท็ม ภายในขอบเขตของเซิร์ฟเวอร์ คีย์มีลำดับการจัดขอบเขตและการกำหนด ไวยากรณ์ หมายถึงเหตุไม่ใช้คีย์จริงๆ แต่สามารถใช้ได้ทุกที่

เซิร์ฟเวอร์อาจมีการระบุเครือข่าย คลาส โคลเอ็นต์ หรืออ็อปชันอยู่ใน เครือข่ายอาจมีซับเน็ต คลาส โคลเอ็นต์ หรืออ็อปชัน ซับเน็ตอาจมีคลาส โคลเอ็นต์ หรือ อ็อปชัน คลาสและโคลเอ็นต์อาจมีเฉพาะอ็อปชัน

เซิร์ฟเวอร์มีชุดของพารามิเตอร์คอนฟิกูเรชัน ที่ใช้กับเซิร์ฟเวอร์เท่านั้น โดยถูกระบุโดยคีย์เซิร์ฟเวอร์ DHCP ในรายการ คีย์ หรือโดยใช้อ็อปชันดีฟอลต์เซิร์ฟเวอร์ได้ แถบเมนู Server เซิร์ฟเวอร์ดีฟอลต์เซิร์ฟเวอร์ใช้กับมาสเตอร์ไฟล์ เซิร์ฟเวอร์ DHCP ที่ระบุภายในมาสเตอร์ไฟล์ได้รับอ็อปชันดีฟอลต์ แต่อาจถูกแก้ไข

ไอเท็มใดๆ ที่ถูกกำหนดในหน้าต่าง Edit อาจถูกแก้ไข เปลี่ยนชื่อ ดู หรือลบออก ซึ่งจะช่วยให้คุณวางไอเท็ม ดูว่าไอเท็มเหมาะสมและเปลี่ยนแปลงตามความจำเป็น

เมื่อไฟล์คอนฟิกูเรชันเสร็จสมบูรณ์ มาสเตอร์ไฟล์ไฟล์เดียวอาจถูกบันทึกและชุดของไฟล์เซิร์ฟเวอร์ อาจถูกสร้างขึ้น ปุ่ม เมนูไฟล์ และปุ่ม เมนู เซิร์ฟเวอร์ มีอ็อปชันเดียวกัน ปุ่ม บันทึกไฟล์ ใช้สำหรับการบันทึกมาสเตอร์ไฟล์ ปุ่ม บันทึกเซิร์ฟเวอร์ ใช้สำหรับการบันทึกเซิร์ฟเวอร์ที่เฉพาะเจาะจงลงในไฟล์

ปุ่ม เมนูไฟล์ ยังมีอ็อปชัน ออก, อ็อปชัน เปิดเพื่อดึงไฟล์ และ อ็อปชัน สร้างใหม่ เพื่อลบทุกสิ่งๆ ที่สร้างขึ้น จนถึงปัจจุบัน

ปุ่ม เมนูการดำเนินการ มีปุ่มสถานะ ปุ่มเริ่มต้น ปุ่มหยุด ปุ่มรีเฟรช และปุ่มส่งไฟล์คอนฟิกูเรชัน จากปุ่มเหล่านี้รีโมตเซิร์ฟเวอร์ สามารถรายงานสถานะ รีเฟรชตัวเองด้วยไฟล์คอนฟิกูเรชันใหม่ อาจถูกหยุดการทำงาน และไฟล์คอนฟิกูเรชันสามารถถูกส่งและรีสตาร์ท

ปุ่ม วิธีใช้ มีชุดของคำสั่ง วิธีใช้ที่อธิบายไอเท็มหน้าต่างแต่ละไอเท็ม

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

การควบคุมการเข้าถึง: ผู้ใช้ใดๆ

## ไฟล์

ไอเท็ม  
/usr/sbin/dhcpsconf  
/etc/dhcpd.conf

คำอธิบาย  
มีคำสั่ง dhcpsconf  
มีไฟล์คอนฟิกูเรชันไคลเอ็นต์ฟอลด์

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“dhcpd Daemon” ในหน้า 132

### ข้อมูลที่เกี่ยวข้อง:

ไฟล์คอนฟิกูเรชันไคลเอ็นต์ DHCP

TCP/IP แอดเดรสและการกำหนด พารามิเตอร์ – Dynamic Host Configuration Protocol

---

## dhcpsd Daemon

### วัตถุประสงค์

สร้างเซิร์ฟเวอร์ Dynamic Host Configuration Protocol (DHCP) ส่งข้อมูลแอดเดรสและคอนฟิกูเรชันไปที่ไคลเอ็นต์ DHCP

### ไวยากรณ์

เมื่อต้องการให้บริการข้อมูล กับ DHCP ไคลเอ็นต์โดยใช้ System Resource Controller (SRC):

```
startsrc -s dhcpsd [ -a Argument ] [ -a Argument ] ...
```

เมื่อต้องการให้บริการข้อมูลกับ DHCP ไคลเอ็นต์โดยไม่มี SRC:

```
dhcpsd [ -f ConfigurationFile ]
```

### คำอธิบาย

เซิร์ฟเวอร์ DHCP จัดการกับการกำหนดและการดูแลรักษาการกำหนดแอดเดรสแบบไดนามิก และยังจัดการการการแจกแจงข้อมูล configuration เพิ่มเติม dhcpsd daemon รัน ในแบ็กกราวนด์และดูแลฐานข้อมูลของข้อมูลเซิร์ฟเวอร์ที่มี พารามิเตอร์การบันทึก, ช่วง IP (Internet Protocol) address, ข้อมูลเน็ตเวิร์ก configuration อื่น และข้อมูลความสามารถในการเข้าถึง ฐานข้อมูลเริ่มต้นถูกระบุ โดยไฟล์คอนฟิกูเรชัน ไฟล์คอนฟิกูเรชันมีข้อมูลทั้งหมดในการ เริ่มการปรับแต่งไคลเอ็นต์ DHCP

เซิร์ฟเวอร์ DHCP ดูแลรักษาฐานข้อมูลของแอดเดรสที่จัดเตรียมไว้และผู้ที่ได้รับแอดเดรสเหล่านั้น ฐานข้อมูลเหล่านี้ถูกเก็บในไฟล์ /etc/dhcpsd.ar และ /etc/dhcpsd.cr เมื่อเริ่มทำงาน เซิร์ฟเวอร์จะอ่านไฟล์คอนฟิกูเรชันและตั้งค่าฐานข้อมูลเริ่มต้น ของแอดเดรสที่พร้อมใช้งาน เซิร์ฟเวอร์ยอมรับคำสั่ง refresh หรือสัญญาณ SIGHUP เพื่ออ่านไฟล์คอนฟิกูเรชัน อีกครั้ง

เซิร์ฟเวอร์ DHCP จะอ่านไฟล์ /etc/services เพื่อกำหนดพอร์ตที่ต้องใช้สำหรับการรับคำร้องขอ เซอร์วิสที่ฟอลด์คือ dhcps เนื่องจากนี้เป็นพอร์ตเดียวกับที่ bootpd daemon ใช้ คุณสามารถรันได้เพียงหนึ่ง (dhcpsd หรือ bootpd) daemon เท่านั้น หากคุณเลือก dhcpsd daemon คุณต้องทำหมายเหตุ bootp จาก ไฟล์ /etc/inetd.conf จากนั้นป้อน refresh -s inetd บนบรรทัดรับคำสั่ง

หมายเหตุ: หาก bootpd daemon รันอยู่ โปรแกรมนี้ต้องหยุดทำงานก่อนที่คุณจะเริ่มต้น daemons

## แฟล็ก

ไอเท็ม  
-f ConfigurationFile

คำอธิบาย  
ระบุไฟล์คอนฟิกูเรชันที่จะใช้

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย  
0 ดำเนินการเป็นผลสำเร็จ  
>0 มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: คุณต้องมีสิทธิ์ root ในการรัน คำสั่งนี้

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/sbin/dhcpsd	มี dhcpsd daemon
/etc/services	กำหนดซ็อกเก็ตและโปรโตคอลที่ใช้สำหรับ อินเทอร์เน็ตเซอร์วิส
/etc/inetd.conf	กำหนดเซอร์วิสที่ถูกควบคุมโดย inetd daemon

สิ่งอ้างอิงที่เกี่ยวข้อง:

“dhcpsd Daemon” ในหน้า 132

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง startsrc

System Resource Controller

TCP/IP daemons

---

## dhcpsdv6 Daemon

### วัตถุประสงค์

สร้างเซิร์ฟเวอร์ Dynamic Host Configuration Protocol (DHCPv6) ส่ง ข้อมูลแอดเดรสและคอนฟิกูเรชันไปที่ไคลเอ็นต์ DHCPv6

### ไวยากรณ์

เมื่อต้องการให้บริการข้อมูล กับ DHCPv6 ไคลเอ็นต์โดยใช้ System Resource Controller (SRC):

```
startsrc -s dhcpsdv6 [-a Argument]
```

เมื่อต้องการให้บริการข้อมูลกับ DHCP ไคลเอ็นต์โดยไม่มี SRC:

`dhcpsdv6 [-d] [-f ConfigurationFile] [-a DadminPort] [-p ServerPort]`

## คำอธิบาย

เซิร์ฟเวอร์ DHCPv6 จัดการการกำหนดและการดูแลรักษา การกำหนดไดนามิกแอดเดรส และยังจัดการการแจกแจงข้อมูล configuration เพิ่มเติม `dhcpsd` daemon รัน ในแบ็กกราวนด์และดูแลฐานข้อมูลของข้อมูลเซิร์ฟเวอร์ที่มี พารามิเตอร์การบ้านทิก, ช่วง IP (Internet Protocol) address, ข้อมูลเน็ตเวิร์ก configuration อื่น และข้อมูลความสามารถในการเข้าถึง ฐานข้อมูล เริ่มต้นถูกระบุโดยไฟล์คอนฟิกูเรชัน ไฟล์คอนฟิกูเรชันมีข้อมูลทั้งหมดในการ เริ่มการปรับแต่งไคลเอ็นต์ DHCP

เซิร์ฟเวอร์ DHCPv6 ดูแลรักษาฐานข้อมูลขอแอดเดรสที่จัดเตรียมและผู้ที่ได้รับแอดเดรสฐานข้อมูลเหล่านี้ถูกเก็บในไฟล์ `/etc/dhcpv6/db_file.crbk` and `/etc/dhcpv6/db_file.cr` เมื่อเริ่มทำงาน เซิร์ฟเวอร์จะอ่านไฟล์คอนฟิกูเรชันและตั้งค่า ฐานข้อมูลเริ่มต้น ของแอดเดรสที่พร้อมใช้งาน เซิร์ฟเวอร์ยอมรับคำสั่งรีเฟรช หรือสัญญาณ `SIGHUP` เพื่ออ่านไฟล์คอนฟิกูเรชัน อีกครั้ง

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a	ระบุพอร์ต Dadmin; ค่าดีฟอลต์คือ 942
-d	แสดงข้อมูลการติบัก
-f <i>ConfigurationFile</i>	ระบุไฟล์คอนฟิกูเรชันที่จะใช้ โดยดีฟอลต์ ไฟล์คอนฟิกูเรชัน คือ <code>/etc/dhcpv6/dhcpsdv6.cnf</code>
-p	พอร์ตที่เซิร์ฟเวอร์ใช้เพื่อรับฟังคำร้องขอที่เข้ามา โดยดีฟอลต์คือพอร์ต 547

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: คุณต้องมีสิทธิ์ `root` ในการรัน คำสั่งนี้

## ตัวอย่าง

- เมื่อต้องการสตาร์ท DHCPv6 เซิร์ฟเวอร์ด้วยไฟล์คอนฟิกูเรชัน `dhcpsdv6.cnf` ที่อยู่ใน `/usr/local`, ให้พิมพ์:  

```
startsrc -s dhcpsdv6 -a "-f /usr/local/dhcpsdv6.cnf"
```

## Location

`/usr/sbin/dhcpsdv6`

## ไฟล์

## ไอเท็ม

```
/usr/sbin/dhcpdsv6  
/etc/dhcpv6/db_file.cr  
/etc/dhcpv6/db_file.crbk  
/etc/dhcpv6/dhcpdsv6.duid
```

## คำอธิบาย

มี `dhcpdsv6` daemon  
มีไคลเอ็นต์เร็กคอร์ด ไฟล์นี้ถูกสร้างโดย เซิร์ฟเวอร์ daemon และ ไม่สามารถปรับแต่งได้  
มีไคลเอ็นต์เร็กคอร์ด ไฟล์นี้ถูกสร้างโดย เซิร์ฟเวอร์ daemon และ ไม่สามารถปรับแต่งได้  
มีไฟล์ identifier ของเซิร์ฟเวอร์ ไฟล์นี้ถูกสร้างโดย เซิร์ฟเวอร์ daemon และ ไม่สามารถปรับแต่งได้

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `startsrc`

---

## คำสั่ง `diag`

### วัตถุประสงค์

ทำการค้นหาปัญหาของฮาร์ดแวร์

### ไวยากรณ์

```
diag [[-a] | [-s [-c]]] [-E days] [-e] | [-d Device [-c] [-v] [-e] [-A]] | [-B [-c]] | [-T taskname] | [-S testsuite] |  
-c -d Device -L pending | complete ]
```

## คำอธิบาย

คำสั่ง `diag` เป็นจุดเริ่มต้นในการรันอ็อปชันงานต่างๆ และการช่วยเหลือเซอร์วิส การช่วยเหลืองานและเซอร์วิสส่วนใหญ่ขึ้นอยู่กับแพลตฟอร์ม การช่วยเหลืองานและเซอร์วิสต่อไปนี้สามารถใช้ได้:

- รันการวินิจฉัย
- แสดงหรือเปลี่ยนอ็อปชันรันไทม์ของการวินิจฉัย
- แสดงข้อมูลแนะนำเซอร์วิส
- แสดงผลลัพธ์การวินิจฉัยก่อนหน้านี้
- แสดงรายงานข้อผิดพลาดเกี่ยวกับฮาร์ดแวร์
- แสดงข้อมูลผลิตภัณฑ์ซอฟต์แวร์
- แสดงรายการคอนฟิกูเรชันและรีซอร์ส
- แสดงข้อมูลผลิตภัณฑ์ฮาร์ดแวร์ที่สำคัญ
- แสดงแอ็ททริบิวต์รีซอร์ส
- เปลี่ยนแปลงข้อมูลผลิตภัณฑ์ที่สำคัญ
- พอร์ม็ิตส์อับนัทิก
- รับรองอับนัทิก
- แสดงรูปแบบการทดสอบ
- Local area network analyzer
- เพิ่มรีซอร์สเข้ากับรายการของรีซอร์ส
- ลบรีซอร์สออกจากรายการของรีซอร์ส

- SCSI bus analyzer
- ดาวน์โหลดไมโครโค้ด
- แสดงหรือเปลี่ยน bootlist
- การวินิจฉัยเป็นระยะ
- การสำรองข้อมูลและเรียกคืนสื่อบันทึก
- การดูแลดิสก์
- กำหนดค่า dials และ LPFkeys
- เพิ่มหรือลบ drawer Config
- สร้างดิสเก็ตคอนฟิกูเรชันที่กำหนดเอง
- อัปเดตการวินิจฉัยเกี่ยวกับดิสก์
- กำหนดค่าอะแดปเตอร์ ISA
- เซลล์พร้อมตัวของระบบปฏิบัติการ (โหมดเซอร์วิสแบบออนไลน์เท่านั้น)
- แสดงหรือเปลี่ยนคอนฟิกูเรชันของไมโครโปรเซสเซอร์
  - เปิดและปิดใช้งานตัวประมวลผล
- แสดงหรือเปลี่ยนคอนฟิกูเรชัน BUMP
  - อัปเดตแฟลช EPROM ด้วยอิมเมจไบนารีใหม่
  - แสดงหรือเปลี่ยนโหมดการวินิจฉัย
  - แสดงหรือเปลี่ยนหมายเลขโทรศัพท์รีโมตและการปรับแต่งโมเด็ม
- แสดงหรือเปลี่ยน electronic mode switch
- ประมวลผลสื่อบันทึกเสริม (โหมดสแตนด์บายเท่านั้น)
- ดาวน์โหลดไมโครโค้ดทั่วไป
- รับการวิเคราะห์บันทึกข้อผิดพลาด
- การช่วยเหลือเซอร์วิสสำหรับใช้กับอีเทอร์เน็ต
- 7135 RAIDiant array service aids
- การระบุและการถอดอุปกรณ์ SCSI
- การช่วยเหลือเซอร์วิสเทปไดรฟ์ SCSD
- การช่วยเหลืออัตราข้อผิดพลาด Escon บิต
- ระบุพีซีดีดิสก์ PCI RAID
- กำหนดค่า ring indicate Power On Policy (CHRP)
- กำหนดค่า surveillance policy (CHRP)
- กำหนดค่า reboot policy (CHRP)
- กำหนดค่า remote maintenance policy (CHRP)
- บันทึกและเรียกคืน hardware management policies (CHRP)
- แสดง firmware device node information (CHRP)
- Spare sector availability
- อัปเดตระบบหรือ service processor flash (CHRP)

- แสดง system environmental sensors (CHRP)
- แสดง checkstop analysis results
- วิเคราะห์บันทึกทำงานภายในของอะแดปเตอร์
- บันทึกแอ็คชันการซ่อมแซม
- Flash SK-NET FDDI firmware
- แสดงระดับไมโครโค้ด

คุณสามารถใช้พาด่วน **smit diag** ของ System Management Interface Tool (SMIT) เพื่อรันคำสั่งนี้

## แฟล็ก

หมายเหตุ: ผู้ใช้ส่วนใหญ่ไม่ต้องใช้ แฟล็กใดๆ เนื่องจากคำสั่ง **diag** เป็นโปรแกรมที่ทำงานด้วยเมนู

ไอเท็ม	คำอธิบาย
-A	ระบุโหมดขั้นสูง คุณยังต้องระบุอุปกรณ์โดยใช้แฟล็ก -d
-a	ประมวลผลการเปลี่ยนแปลงใดๆ ในคอนฟิกูเรชันของฮาร์ดแวร์โดยการสอบถามว่ารีซอร์สที่หายไปถูกถอดออก ปิด หรืออื่นๆ เป็นต้น รีซอร์สที่หายไป (ระบุด้วย 'M') และพาสของรีซอร์สที่หายไป (ระบุด้วย 'P') ถูกรวมในรายการการเลือกรีซอร์สการวินิจฉัย
-B	สั่งให้การทดสอบการวินิจฉัยรันการทดสอบระบบฐาน การวิเคราะห์บันทึกข้อผิดพลาดยังถูกดำเนินการบนพื้นที่ในระบบฐานที่สนับสนุน การวิเคราะห์บันทึกข้อผิดพลาด
-c	ระบุว่าเครื่องไมรวมทำงาน ไม่มีการถามคำถาม ผลลัพธ์จะถูกเขียนไปที่เอาต์พุตมาตรฐาน คุณยังต้องใช้แฟล็กที่เป็นทางเลือก ที่ระบุอุปกรณ์ที่ต้องการทดสอบ (d, B หรือ s)
-d Device	ระบุอุปกรณ์เพื่อรันการทดสอบการวินิจฉัย
-E Days	ระบุจำนวนวันที่ใช้สำหรับการค้นหาบันทึกข้อผิดพลาด ระหว่างรันการวิเคราะห์บันทึกข้อผิดพลาด แฟล็กนี้ทำงานกับแฟล็กอื่น
-e	ดำเนินการวิเคราะห์บันทึกข้อผิดพลาด ถ้ามีการสนับสนุนบนอุปกรณ์ที่เลือก ไม่มีการทำการทดสอบ แฟล็กนี้ต้องใช้ร่วมกับแฟล็ก -d ไม่เช่นนั้นเมนูการเลือกรีซอร์สจะปรากฏขึ้น หากใช้ร่วมกับแฟล็ก -v แฟล็ก -v จะมีความสำคัญกว่าและแฟล็ก -e จะถูกข้าม
-S testsuite	ระบุชุดการทดสอบเฉพาะของอุปกรณ์ที่ต้องการทดสอบ: <ol style="list-style-type: none"> <li>1. ระบบฐาน</li> <li>2. อุปกรณ์ I/O</li> <li>3. อุปกรณ์อะซิงค์</li> <li>4. อุปกรณ์กราฟิก</li> <li>5. อุปกรณ์ SCSI</li> <li>6. อุปกรณ์หน่วยเก็บ</li> <li>7. อุปกรณ์ทั่วไป</li> <li>8. อุปกรณ์มัลติมีเดีย</li> </ol>
-L pending   complete	บันทึกแอ็คชันการซ่อมแซมสำหรับรีซอร์สที่ระบุด้วยอ็อปชัน -d และ -c ใช้พารามิเตอร์ <b>pending</b> หากมีการเปลี่ยนชิ้นส่วน แต่ยังไม่ทราบวาชิ้นส่วนยังอยู่ในระบบหรือไม่ ใช้พารามิเตอร์ <b>complete</b> หากมีการเปลี่ยนชิ้นส่วน แต่ยังไม่ทราบวาชิ้นส่วนยังอยู่ในระบบ
-s	รันการทดสอบการวินิจฉัยบนรีซอร์สทั้งหมด

ไอเท็ม  
-T *taskname*

#### คำอธิบาย

ระบุงาน Fastpath เฉพาะที่ต้องการรัน รายการต่อไปนี้ แสดงงาน fastpath ในปัจจุบัน :

**format** งานฟอร์แมตสื่อบันทึก

**certify** งานรับรองสื่อบันทึก

#### download

งานดาวน์โหลดไมโครโค้ด

#### disp\_mcode

งานแสดงระดับไมโครโค้ด

#### chkspares

งาน Spare sector availability

#### identifyRemove

งาน Hot plug

-v

หมายเหตุ: งานขึ้นอยู่กับแพลตฟอร์มและอุปกรณ์ บางงาน อาจไม่พร้อมใช้งานบนระบบ  
รันการทดสอบการวินิจฉัยในโหมดการตรวจสอบระบบ ไม่มีการดำเนินการวิเคราะห์บันทึกข้อผิดพลาด ดีฟอลต์คือโหมด  
การกำหนดปัญหาที่ทดสอบอุปกรณ์ และรันการวิเคราะห์บันทึกข้อผิดพลาด ถ้าใช้กับแฟล็ก -e , แฟล็ก -v มีความสำคัญกว่า  
และแฟล็ก -e จะถูกละเว้น ต้องใช้ร่วมกับแฟล็ก -d เพื่อระบุอุปกรณ์ เพื่อรันการทดสอบการวินิจฉัย

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: เฉพาะผู้ใช้ root เท่านั้นที่สามารถรันคำสั่งนี้ได้

การควบคุม Privilege: กลุ่มระบบ

## ตัวอย่าง

เมื่อต้องการรัน การทดสอบการวินิจฉัยบนอุปกรณ์ scdisk0 โดยไม่มีคำถาม ให้ป้อน:

```
diag -d scdisk0 -c
```

## File

ไอเท็ม

/usr/sbin/diag

คำอธิบาย

มีคำสั่ง diag

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง diaggetrto”

“คำสั่ง diagsetrto” ในหน้า 148

---

## คำสั่ง diaggetrto

### วัตถุประสงค์

แสดงอ็พชันรันไทม์การวินิจฉัย

## ไวยากรณ์

diaggetrto [[-a][ -d][ -l][ -m][ -n][ -p][ -s]]

### คำอธิบาย

คำสั่ง `diaggetrto` แสดงค่าของ อีพซันรันไทม์การวินิจฉัย อีพซันรันไทม์ต่อไปนี้อาจสามารถแสดงได้ด้วยคำสั่ง `diaggetrto`:

#### แสดงเมนูการเลือกโหมดการวินิจฉัย

เมื่ออีพซันนี้ถูกปิด การวินิจฉัยรันในโหมดการระบุปัญหา เท่านั้น ดีฟอลต์คือเปิด

#### รวมการวินิจฉัยระดับสูง

เมื่ออีพซันนี้ถูกเปิด การวินิจฉัยรันในโหมดระดับสูง เมื่อรันจาก เมนูการเลือกงาน หรือบรรทัดคำสั่ง ค่าดีฟอลต์คือปิด

#### จำนวนวันที่ใช้ในการค้นหาบันทึกข้อผิดพลาด

อีพซันนี้ควบคุมระยะเวลาของรายการบันทึกข้อผิดพลาด ก่อนที่จะไม่ ถูกวิเคราะห์อีกต่อไปโดยโปรแกรมวินิจฉัย ค่าดีฟอลต์คือ 7

#### แสดงตัวบ่งชี้ความคับหน้า

เมื่อเปิดอีพซันนี้ แอ็พพลิเคชันการวินิจฉัยที่สนับสนุนตัวบ่งชี้ ความคับหน้าจะแสดงข้อมูลระยะเวลา ดีฟอลต์คือเปิด

#### วินิจฉัยการบันทึกเหตุการณ์

เมื่อเปิดอีพซันนี้, จะมีการวินิจฉัยบันทึกเหตุการณ์ ดีฟอลต์คือเปิด

#### ขนาดไฟล์บันทึกเหตุการณ์การวินิจฉัย

อีพซันนี้ควบคุมขนาดสูงสุดของบันทึกเหตุการณ์การวินิจฉัย ขนาด ที่อนุญาตคือการเพิ่มจำนวนเป็นหลายร้อย กิโลไบต์ ดีฟอลต์คือ 100K

## แฟล็ก

### ไอเท็ม

-a  
-d  
-l  
-m  
-n  
-p  
-s

### คำอธิบาย

แสดงค่าของ รวมการวินิจฉัย ระดับสูง  
แสดงค่าของ การบันทึก เหตุการณ์การวินิจฉัย  
แสดงค่าของ ขนาดไฟล์บันทึก เหตุการณ์การวินิจฉัย  
แสดงค่าของ แสดงเมนูการเลือก โหมดการวินิจฉัย  
แสดงค่าของ จำนวนวันที่ใช้ เพื่อค้นหาบันทึกข้อผิดพลาด  
แสดงค่าของ แสดงตัวบ่งชี้ ความคับหน้า  
แสดงอีพซันรันไทม์การวินิจฉัยทั้งหมด

## สถานะออก

0 คำสั่งดำเนินการเสร็จด้วยผลสำเร็จ

>0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการแสดงขนาดบันทึกเหตุการณ์การวินิจฉัย ให้พิมพ์:

```
/usr/lpp/diagnostics/bin/diaggetrto -l
```

- เมื่อต้องการตรวจสอบว่าตัวบ่งชี้ความคืบหน้าถูกเปิดอยู่หรือไม่ และต้องการตรวจสอบว่าการบันทึกเหตุการณ์การวินิจฉัยเปิดอยู่หรือไม่ ให้พิมพ์:

```
/usr/lpp/diagnostics/bin/diaggetrto -p -d
```

- เมื่อต้องการแสดงจำนวนวันที่จะค้นหาบันทึกข้อผิดพลาดให้พิมพ์:

```
/usr/lpp/diagnostics/bin/diaggetrto -n
```

## ไฟล์

ไอเท็ม

/usr/lpp/diagnostics/bin/diaggetrto

คำอธิบาย

มีคำสั่ง diagsetrto

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง diagsetrto” ในหน้า 148

“คำสั่ง diag” ในหน้า 142

## คำสั่ง diagrpt

### วัตถุประสงค์

แสดงผลผลการวินิจฉัยก่อนหน้านี้

### ไวยากรณ์

```
diagrpt [ [-o] | [-s mmdyy] | [-a] | [-r] ]
```

### คำอธิบาย

คำสั่ง **diagrpt** แสดงผลลัพธ์ของเซสชันวินิจฉัย ก่อนหน้านี้ ผลลัพธ์มีสามประเภทที่สามารถดูได้:

- ไฟล์ผลลัพธ์การวินิจฉัยเก็บอยู่ในไดเรกทอรี /etc/lpp/diagnostic/data
- ข้อมูลบันทึกเหตุการณ์การวินิจฉัย
- ผลลัพธ์การวินิจฉัยเก็บอยู่ใน NVRAM บนระบบ CHRP

### แฟล็ก

ไอเท็ม

-o  
-s mmdyy  
-a  
-r

คำอธิบาย

แสดงไฟล์ผลลัพธ์การวินิจฉัยล่าสุดที่เก็บในไดเรกทอรี /etc/lpp/diagnostics/data  
แสดงไฟล์ผลลัพธ์การวินิจฉัยทั้งหมด ตั้งแต่วันที่ระบุ  
แสดงบันทึกเหตุการณ์การวิเคราะห์ในแบบเวอร์ชันยาว  
แสดงบันทึกเหตุการณ์การวิเคราะห์ในแบบเวอร์ชันย่อ

### ตัวอย่าง

- เมื่อต้องการแสดงรายการไฟล์ผลลัพธ์การวินิจฉัยทั้งหมดตั้งแต่ Jan 31, 1999, ให้ป้อน:

```
/usr/lpp/diagnostics/bin/diagrpt -s 013199
```

- เมื่อต้องการดูบันทึกเหตุการณ์การวินิจฉัยในแบบเวอร์ชันย่อ ให้พิมพ์:

/usr/lpp/diagnostics/bin/diagrpt -r

## File

ไอเท็ม

/usr/lpp/diagnostics/bin/diagrpt

คำอธิบาย

มีคำสั่ง `diagrpt`

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `diag`” ในหน้า 142

---

## คำสั่ง `diagsetrto`

### วัตถุประสงค์

เซตอ็อพชันรันใหม่การวินิจฉัย

### ไวยากรณ์

`diagsetrto` [ [ `-a on|off` ] [ `-d on|off` ] [ `-l Size` ] [ `-m on|off` ] [ `-n Days` ] [ `-p on|off` ] ]

### คำอธิบาย

คำสั่ง `diagsetrto` เซตค่าของจำนวน อ็อพชันรันใหม่การวินิจฉัย อ็อพชันรันใหม่ต่อไปนี้อาจถูกแก้ไขได้ด้วยคำสั่ง `diagsetrto`:

#### แสดงเมนูการเลือกโหมดการวินิจฉัย

เมื่ออ็อพชันนี้ถูกปิด การวินิจฉัยรันในโหมดการระบุปัญหา เท่านั้น ดีฟอลต์คือเปิด

#### รวมการวินิจฉัยระดับสูง

เมื่ออ็อพชันนี้ถูกเปิด การวินิจฉัยรันในโหมดระดับสูง เมื่อรันจาก เมนูการเลือกงาน หรือบรรทัดคำสั่ง ค่าดีฟอลต์คือปิด

#### จำนวนวันที่ใช้ในการค้นหาค้นหาบันทึกข้อผิดพลาด

อ็อพชันนี้ควบคุมระยะเวลาของรายการบันทึกข้อผิดพลาด ก่อนที่จะไม่ ถูกวิเคราะห์อีกต่อไปโดยโปรแกรมวินิจฉัย ค่าดีฟอลต์คือ 7

#### แสดงตัวบ่งชี้ความคืบหน้า

เมื่อเปิดอ็อพชันนี้ แอ็พพลิเคชันการวินิจฉัยที่สนับสนุนตัวบ่งชี้ ความคืบหน้าจะแสดงข้อมูลระยะเวลา ดีฟอลต์คือเปิด

#### วินิจฉัยการบันทึกเหตุการณ์

เมื่อเปิดอ็อพชันนี้, จะมีการวินิจฉัยบันทึกเหตุการณ์ ดีฟอลต์คือเปิด

#### วินิจฉัยขนาดล็อกไฟล์เหตุการณ์

อ็อพชันนี้ควบคุมขนาดสูงสุดของบันทึกเหตุการณ์การวินิจฉัย ขนาด ที่อนุญาตคือการเพิ่มจำนวนเป็นหลายร้อย กิโลไบต์ ดีฟอลต์คือ 100K

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a on   off	เช็คค่าของ รวมดการวินิจฉัย ระดับสูง
-d on   off	เช็คค่าของ การบันทึก เหตุการณ์การวินิจฉัย
-l Size	เช็คค่าของ ขนาดไฟล์บันทึก เหตุการณ์การวินิจฉัย
-m on   off	เช็คค่าของ แสดงเมนูการเลือก โหมดการวินิจฉัย
-n Days	เช็คค่าของ จำนวนวันที่ใช้ เพื่อค้นหาบันทึกข้อผิดพลาด
-p on   off	เช็คค่าของ แสดงตัวบ่งชี้ ความคืบหน้า

## สถานะออก

- 0 คำสั่งดำเนินการเสร็จด้วยผลสำเร็จ
- >0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

- เมื่อต้องการเช็คขนาดบันทึกเหตุการณ์การวินิจฉัยเป็น 500K ให้พิมพ์:  

```
/usr/lpp/diagnostics/bin/diagsetrto -l 500
```
- เมื่อต้องการปิดตัวบ่งชี้ความคืบหน้าและปิดการบันทึกเหตุการณ์การวินิจฉัย ให้พิมพ์:  

```
/usr/lpp/diagnostics/bin/diagsetrto -p off -d off
```
- เมื่อต้องการเช็คจำนวนวันที่จะค้นหาบันทึกข้อผิดพลาดเป็น 50 ให้พิมพ์:  

```
/usr/lpp/diagnostics/bin/diagsetrto -n 50
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/lpp/diagnostics/bin/diagsetrto	มีคำสั่ง diagsetrto

## สิ่งอ้างอิงที่เกี่ยวข้อง:

- “คำสั่ง diagsetrto” ในหน้า 145
- “คำสั่ง diag” ในหน้า 142

---

## คำสั่ง diction

### วัตถุประสงค์

ไฮไลต์ประโยคที่ไม่ชัดเจนหรือฟุ่มเฟือย

### ไวยากรณ์

```
diction [ -ml ] [ -mm ] [ -f PatternFile ] [ -n ] File ...
```

## คำอธิบาย

คำสั่ง **diction** ค้นหาประโยคทั้งหมด ในเอกสารภาษาอังกฤษที่มีวลีจากฐานข้อมูล ของการกำหนดคำที่ไม่ชัดเจนหรือฟุ่มเฟือย แต่ละวลีถูกปิดด้วย [] (วงเล็บเหลี่ยม) เนื่องจากคำสั่ง **diction** รันคำสั่ง **deroff** ก่อนค้นหาข้อความ, รวมไฟล์ส่วนหัวที่มี ข้อมูลการจัดรูปแบบที่เหมาะสมเป็นหนึ่งของอินพุต คำสั่ง **explain** มีพจนานุกรมคำพ้องแบบโต้ตอบสำหรับ วลีที่พบโดยคำสั่ง **diction**

ใช้แม่โครการจัดรูปแบบที่ไม่เป็นมาตรฐานอาจทำให้ แยกประโยคไม่ถูกต้อง โดยเฉพาะคำสั่ง **diction** ไม่เข้าใจแฟล็ก **-me**

## แฟล็ก

### ไอเท็ม

-f *PatternFile*  
-ml  
-mm  
-n

### คำอธิบาย

ระบุไฟล์ที่มีตัวอย่างของรูปแบบคำที่ไม่ชัดเจน ไฟล์นี้ถูกใช้ เพิ่มเติมจากไฟล์ด็พอลด์ ทำให้คำสั่ง **deroff** ข้ามรายการแม่โคร **mm** สามารถใช้ได้ถ้าเอกสารมี รายการส่วนของประโยคมากเกินไป เขียนทับด็พอลด์แม่โครแฟ็กเกจ **ms** ยับยั้งการใช้ด็พอลด์ไฟล์ เมื่อมีการใช้กับแฟล็ก **-f**; เฉพาะไฟล์ที่ระบุโดยพารามิเตอร์ *PatternFile* ที่ถูกใช้

## ไฟล์

### ไอเท็ม

/usr/lib/dict.d

### คำอธิบาย

มีรูปแบบด็พอลด์

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง **deroff**” ในหน้า 105

“คำสั่ง **explain**” ในหน้า 463

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **troff**

---

## คำสั่ง **diff**

### วัตถุประสงค์

เปรียบเทียบไฟล์ข้อความ

### ไวยากรณ์

เมื่อต้องการเปรียบเทียบเนื้อหาของสองไฟล์

```
diff [-cl -C Lines | -D [String]] [-e | -f | -n | -u | -U Lines] [-b] [-i] [-t] [-w] File1 File2
```

```
diff [-h] [-b] File1 File2
```

เมื่อต้องการเรียงเนื้อหาของไดเร็กทอรีและเปรียบเทียบไฟล์ที่ต่างกัน

```
diff [-cl -C Lines | -e | -f | -n | -u | -U Lines] [-b] [-i] [-l] [-r] [-s] [-S File] [-t] [-w] Directory1 Directory2
```

`diff [-h] [-b] Directory1 Directory2`

## คำอธิบาย

คำสั่ง `diff` เปรียบเทียบ ไฟล์ข้อความ สามารถเปรียบเทียบไฟล์เดี่ยวหรือเนื้อหาของไดเรกทอรี

หมายเหตุ: คำสั่ง `diff` ทำงานเฉพาะกับไฟล์อินพุต ที่เป็นไฟล์ข้อความเท่านั้น

ถ้ามีการระบุพารามิเตอร์ `Directory1` และ `Directory2` คำสั่ง `diff` จะเปรียบเทียบไฟล์ข้อความที่มีชื่อ เหมือนกันในทั้งสองไดเรกทอรี ไบนารีไฟล์ที่ต่างกัน, ไดเรกทอรีทั่วไป และไฟล์ที่อยู่ในหนึ่งไดเรกทอรีที่ถูกแสดง

เมื่อคำสั่ง `diff` ถูกรัน บนไฟล์ทั่วไป และเมื่อเปรียบเทียบไฟล์ที่ต่างกัน ระหว่างการเปรียบเทียบไดเรกทอรี คำสั่ง `diff` แจ้งถึงบรรทัดที่ต้องถูกเปลี่ยนในไฟล์เพื่อให้ไฟล์ตรงกัน ถ้าพารามิเตอร์ `File1` หรือ `File2` ไม่ใช่ไดเรกทอรี พารามิเตอร์ใดพารามิเตอร์หนึ่งอาจถูกกำหนดเป็น - (เครื่องหมายลบ) อินพุตมาตรฐานจะถูกนำมาใช้ ถ้าพารามิเตอร์ `File1` เป็น ไดเรกทอรี ไฟล์ในไดเรกทอรีซึ่งชื่อไฟล์ เหมือนกับพารามิเตอร์ `File2` จะถูกใช้

เอาต์พุตทั่วไปมีบรรทัดในรูปแบบเหล่านี้:

Lines Affected in File1	Action	Lines Affected in File2
Number1	a	Number2[,Number3]
Number1[,Number2]	d	Number3
Number1[,Number2]	c	Number3[,Number4]

บรรทัดเหล่านี้ทำซ้ำคำสั่งย่อย `ed` เพื่อแปลง `File1` ไปเป็น `File2` ตัวเลขก่อนที่ตัวอักษรการดำเนินการเป็นของ `File1`; ตัวเลขหลังจากเป็นของ `File2` ดังนั้น เมื่อเปลี่ยน `a` ไปเป็น `d` และอ่านจากขวา ไปซ้าย คุณสามารถบอกวิธีการแปลง `File2` ไปเป็น `File1` เหมือนในคำสั่ง `ed`, คู่เหมือน (โดยที่ `Number1 = Number2`) ถูกย่อลงเป็นตัวเลขเดียว

แต่ละบรรทัด, คำสั่ง `diff` แสดงบรรทัดทั้งหมดที่ได้รับผลใน ไฟล์แรกนำหน้าด้วย `<`: (เครื่องหมายน้อยกว่า, โคลอน), จากนั้นแสดงบรรทัดทั้งหมดที่ได้รับผลในไฟล์ที่สองนำหน้า ด้วย `>` (เครื่องหมายมากกว่า)

ค่าจบการทำงาน 0 หมายถึงไม่มีความแตกต่าง, 1 หมายถึงพบความแตกต่าง และ 2 หมายถึงมีข้อผิดพลาด

หมายเหตุ: ถ้ามีมากกว่าหนึ่งแฟล็ก `-c`, `-C`, `-D`, `-e`, `-f`, or `-n`, `-u`, หรือ `-U` ดังกล่าว ถูกระบุ แฟล็กสุดท้ายบนบรรทัดคำสั่ง จะได้รับความสำคัญที่สุด ระบบไม่ส่งข้อความแสดงความผิดพลาด

## แฟล็ก

ไอเท็ม

`-b`

คำอธิบาย

ทำให้จำนวนของช่องว่างที่ท้ายบรรทัด ถูกดำเนินการเป็นอักขระขึ้นบรรทัดใหม่เดี่ยว (อักขระช่องว่างก่อนหน้า อักขระขึ้นบรรทัดใหม่จะถูกละเว้น) และสตริงอื่นของ อักขระช่องว่าง ไม่รวมอักขระขึ้นบรรทัดใหม่ ในการเปรียบเทียบ อย่างเท่าเทียม

`-C Lines`

ทำให้คำสั่ง `diff` เปรียบเทียบ จำนวนบรรทัดของบริบท ที่ตัดออก กับค่า ที่ระบุโดยตัวแปร `Lines` แฟล็ก `-C` ดัดแปลงเอาต์พุตเล็กน้อย เอาต์พุต เริ่มต้นด้วย identification ของไฟล์ที่เกี่ยวข้องและวันที่ สร้าง แต่ละการเปลี่ยนแปลงถูกแยกโดย บรรทัดที่มีเครื่องหมาย \* (เครื่องหมายดอกจัน) สิบสองเครื่องหมาย บรรทัดที่เอาออกจาก `File1` ถูกทำเครื่องหมายด้วย - (เครื่องหมายลบ) และบรรทัดที่เพิ่มให้กับ `File2` ถูกทำเครื่องหมาย ด้วย + (เครื่องหมายบวก) บรรทัดที่เปลี่ยนจาก ไฟล์หนึ่งไปอีกไฟล์หนึ่งถูกทำเครื่องหมายในทั้งสองไฟล์ด้วย ! (เครื่องหมาย ตกใจ) การเปลี่ยนแปลงที่อยู่ใน บรรทัดบริบท ที่ตัดออก ที่ระบุของแต่ละไฟล์ถูกจัดกลุ่ม รวมกันเป็นเอาต์พุต

## ไอเท็ม

- c **คำอธิบาย**  
ทำให้คำสั่ง **diff** เปรียบเทียบ กับสามบรรทัดของบริบท ที่ตัดลอก แฟล็ก -c ตัดแปลงเอาต์พุตเล็กน้อย เอาต์พุต เริ่มต้นด้วย identification ของไฟล์ที่เกี่ยวข้องและวันที่ สร้าง แต่ละการเปลี่ยนแปลงถูกแยกโดยบรรทัดที่มีเครื่องหมาย \* (เครื่องหมายดอกจัน) สิบสองเครื่องหมาย บรรทัดที่เอาออกจาก *File1* ถูกทำเครื่องหมายด้วย - (เครื่องหมายลบ) และบรรทัดที่เพิ่มให้กับ *File2* ถูกทำเครื่องหมาย ด้วย + (เครื่องหมายบวก) บรรทัดที่เปลี่ยนจาก ไฟล์หนึ่งไปอีกไฟล์หนึ่งถูกทำเครื่องหมายในทั้งสองไฟล์ด้วย ! (เครื่องหมาย ตกใจ) การเปลี่ยนแปลงใน บรรทัดบริบท ที่ตัดลอก ที่ระบุของแต่ละไฟล์ถูกจัดกลุ่ม รวมกันเป็นเอาต์พุต
- D [ *String* ]  
ทำให้คำสั่ง **diff** สร้าง เวอร์ชันผสมรวมกันของ *File1* และ *File2* บน เอาต์พุตมาตรฐาน คอนโทรล C preprocessor ถูกรวมไว้ เพื่อที่ การคอมไพล์ผลลัพธ์โดยไม่มีกำหนด *String* เทา เทียบกับการคอมไพล์ *File1*, ขณะทำการกำหนด *String* ให้ผลเป็น *File2*
- e  
สร้างเอาต์พุตในฟอร์มที่เหมาะสมสำหรับใช้กับเอดิเตอร์ **ed** เพื่อแปลง *File1* เป็น *File2* เมื่อใช้แฟล็กนี้ โปรแกรมเซลล์ต่อไปนี้อาจช่วยเหลือการดูแลเวอร์ชันของไฟล์หลายเวอร์ชัน เฉพาะไฟล์ที่สืบทอดมา (§1) และลูกโซ่ของเวอร์ชันต่อเวอร์ชัน **ed** สคริปต์ (§2, §3, ...) ที่สร้างโดยคำสั่ง **diff** จำเป็นต้องพร้อม อยู่แล้ว เวอร์ชันล่าสุดที่แสดงบนเอาต์พุตมาตรฐาน ดังนี้:  

```
(shift; cat $*; echo '1,$p') | ed - $1
```
- f  
คำสั่งพิเศษถูกเพิ่มให้กับเอาต์พุตเมื่อแฟล็ก -e ถูกใช้เพื่อเปรียบเทียบไดเรกทอรี ดังนั้นผลลัพธ์ เป็นเซลล์สคริปต์สำหรับการแปลงไฟล์ข้อความที่เป็นปกติกับ สองไดเรกทอรีจากสถานะใน *Directory1* กับ สถานะใน *Directory2*  
**หมายเหตุ:** การแก้ไข สคริปต์ที่ทำโดยแฟล็ก -e or -f ไม่สามารถสร้างบรรทัดที่มีหนึ่ง (จุด) สร้างเอาต์พุตในฟอร์มที่ไม่เหมาะสมสำหรับใช้กับเอดิเตอร์ **ed**, แสดงการตัดแปลงที่จำเป็นในการ แปลง *File1* ไปเป็น *File2* ใน ลำดับย้อนกลับของที่สร้างภายใต้แฟล็ก -e
- h  
ทำการเปรียบเทียบอีกวิธีซึ่งอาจเร็วกว่า ถ้า ส่วนที่เปลี่ยนแปลงสั้นและถูกแยกกันได้ดีกว่า แฟล็ก -h ทำงานได้กับไฟล์ความยาวเท่าใดก็ได้ แฟล็ก -c, -C, -D, -e, -f, และ -n ไม่สามารถนำมาใช้กับแฟล็ก -h แฟล็กอื่นทั้งหมดยกเว้น แฟล็ก -b ถูกจะเว้นเมื่อใช้กับแฟล็ก -h
- i  
ละเว้นตัวพิมพ์ ตัวอย่าง ตัวพิมพ์เล็ก a ถูกปฏิบัติเช่นเดียวกับตัวพิมพ์ใหญ่ A
- l  
รูปแบบเอาต์พุตแบบยาว แต่ละผลลัพธ์จากคำสั่ง **diff** การเปรียบเทียบไฟล์ข้อความถูกไฟท์ผ่านคำสั่ง **pr** สำหรับการแบ่งหน้า ความแตกต่างอื่นคือมีการบันทึกและทำสรุปหลังจาก ความแตกต่างของไฟล์ข้อความทั้งหมดถูกรายงาน
- n  
สร้างผลลัพธ์เหมือนกับแฟล็ก -e, แต่ในลำดับตรงข้าม และมีจำนวนบรรทัดที่เปลี่ยนแปลงในแต่ละ คำสั่ง insert หรือ delete นี้เป็นฟอร์มที่ใช้โดย revision control system (RCS)
- r  
ทำให้แอ็พพลิเคชันของคำสั่ง **diff** ทำงานวนซ้ำกับไดเรกทอรีย่อยทั่วไปที่พบ
- s  
รายงานไฟล์ที่เหมือนกันเท่านั้น
- S [ *File* ]  
ละเว้นไฟล์ซึ่งชื่อเรียงกันก่อนไฟล์ที่ระบุ โดยตัวแปร *File* เมื่อเปรียบเทียบไดเรกทอรี แฟล็ก -S ใช้กับไดเรกทอรีที่ระบุในพารามิเตอร์ *Directory1* และ *Directory2* ถ้าคุณใช้แฟล็ก -r กับแฟล็ก -S, แฟล็ก -S ไม่ทำงานแบบวนซ้ำในไดเรกทอรีย่อย *Directory1* และ *Directory2*
- t  
ขยายแท็บในบรรทัดเอาต์พุต เอาต์พุตทั่วไปหรือแฟล็ก -c เอาต์พุตเพิ่มอักขระให้กับด้านหน้าของแต่ละ บรรทัด ซึ่งอาจมีผลกับการเอียงของบรรทัดซอร์สต้นฉบับและ ทำให้การแสดงผลเอาต์พุตยากต่อการแปล แฟล็กนี้รักษา การเอียงของซอร์สต้นฉบับ
- u  
ทำให้คำสั่ง **diff** เปรียบเทียบ กับสามบรรทัดของบริบทที่ยังไม่ได้อ่านร่วมกัน  
  
เอาต์พุตเหมือนกับ แฟล็ก -c, ยกเว้นว่าบรรทัดบริบท ไม่มีการทำซ้ำ แต่บรรทัด บริบท บรรทัดที่ลบ และที่เพิ่ม ถูกแสดงร่วมกัน โดยการแทรก
- U *Lines*  
ทำให้คำสั่ง **diff** เปรียบเทียบ จำนวนบรรทัดของบริบทที่ไม่รวมกัน กับค่า ที่ระบุโดยตัวแปร *Lines* เอาต์พุตเหมือนกับแฟล็ก -C, ยกเว้นว่าบรรทัดบริบท ไม่มีการทำซ้ำ แต่บรรทัด บริบท บรรทัดที่ลบ และที่เพิ่ม ถูกแสดงร่วมกัน โดยการแทรก
- w  
ละเว้นอักขระที่ว่างและแท็บ ทั้งหมดและปฏิบัติต่อสตริงว่าง อื่นทั้งหมดเหมือนกัน ตัวอย่าง, if ( a == b ) เหมือนกับ if(a==b)

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ไม่พบความแตกต่าง
1	พบความแตกต่าง
>1	มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการเปรียบเทียบสองไฟล์ให้ป้อน:

```
diff chap1.back chap1
```

ซึ่งจะแสดงความแตกต่าง ระหว่างไฟล์ chap1.bak และ chap1

2. เมื่อต้องการเปรียบเทียบสองไฟล์ขณะละเว้นความแตกต่าง ของจำนวนช่องว่างให้ป้อน:

```
diff -w prog.c.bak prog.c
```

ถ้าสองบรรทัดต่างกันเฉพาะในจำนวนของช่องว่างและแท็บระหว่างคำ คำสั่ง `diff -w` จะถือว่าทั้งสอง เหมือนกัน

3. เมื่อต้องการสร้างไฟล์ที่มีคำสั่ง ที่คำสั่ง `ed` สามารถใช้เพื่อสร้างไฟล์หนึ่งใหม่ จากอีกไฟล์ให้ป้อน:

```
diff -e chap2 chap2.old >new.to.old.ed
```

นี่เป็นการสร้างไฟล์ชื่อ `new.to.old.ed` ที่มีคำสั่งย่อ `ed` เพื่อเปลี่ยน `chap2` กลับไปเป็นเวอร์ชันของข้อความที่พบ ใน `chap2.old` ในกรณีส่วนใหญ่, `new.to.old.ed` เป็นไฟล์ที่เล็กกว่า `chap2.old` มาก คุณสามารถประหยัดพื้นที่ดิสก์โดยการลบ `chap2.old`, และคุณสามารถสร้างไฟล์นี้ขึ้นใหม่ ได้ทุกเวลาโดยป้อน:

```
(cat new.to.old.ed ; echo '1,$p') | ed - chap2 >chap2.old
```

คำสั่ง ในวงเล็บเพิ่ม `1, $p` ให้กับ จุดสิ้นสุดของคำสั่งการแก้ไขที่ส่งไปที่เอดิเตอร์ `ed`, `$p` ทำให้คำสั่ง `ed` เขียนไฟล์ไปที่ เอาต์พุตมาตรฐานหลังจากทำการแก้ไข จากนั้นลำดับคำสั่ง ที่เปลี่ยนแปลงถูกไฟฟ์ไปที่คำสั่ง `ed (| ed)`, และเอดิเตอร์จะ อ่านเป็นอินพุต มาตรฐาน แฟล็ก - ทำให้คำสั่ง `ed` ไม่แสดงขนาดไฟล์และข้อมูลพิเศษอื่น เนื่องจากจะไป ผสมกับข้อความ ของ `chap2.old`

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/diff</code>	มีคำสั่ง <code>diff</code>

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `bdiff`

คำสั่ง `cmp`

คำสั่ง `pr`

การเปลี่ยนทิศทางอินพุตและเอาต์พุต

---

## คำสั่ง diff3

### วัตถุประสงค์

เปรียบเทียบสามไฟล์

### ไวยากรณ์

```
diff3 [-e|-x|-E|-X|-3] File1 File2 File3
```

### คำอธิบาย

คำสั่ง diff3 เปรียบเทียบไฟล์ สามไฟล์และเขียนข้อมูลช่วงของข้อความที่ต่างกันไปที่เอาต์พุตมาตรฐาน, มีการแฟล็ก ด้วยโค้ดต่อไปนี้:

ไอเท็ม	คำอธิบาย
====	ทั้งสามไฟล์ต่างกัน
====1	File1 แตกต่าง
====2	File2 แตกต่าง
====3	File3 แตกต่าง

ชนิดของการเปลี่ยนแปลงที่จำเป็นในการแปลงช่วงที่กำหนด ของไฟล์ที่กำหนดเพื่อให้ตรงกับไฟล์อื่นถูกระบุในหนึ่งในสองวิธีในเอาต์พุต:

ไอเท็ม	คำอธิบาย
File:Number1 a	ข้อความจะถูกเพิ่มหลังจากหมายเลขบรรทัด Number1 ใน File, โดยที่ File คือ 1, 2 หรือ 3
File:Number1[,Number2]c	ข้อความในบรรทัดขอบเขต Number1 ถึงบรรทัด Number2 จะถูกเปลี่ยน ถ้า Number1 เหมือนกับ Number2 ขอบเขตอาจถูกย่อ เป็น Number1

เนื้อหาของขอบเขตเป็นไปตามการบ่งชี้ c เมื่อเนื้อหาของสองไฟล์เหมือนกัน คำสั่ง diff3 ไม่แสดงเนื้อหาของไฟล์ที่มีตัวเลขต่ำกว่า, แม้ว่าจะแสดงตำแหน่งของบรรทัดที่เท่ากันสำหรับไฟล์

หมายเหตุ: แก๊ซสคริปต์ที่สร้างโดยแฟล็ก -e ที่ไม่สามารถสร้าง บรรทัดที่มี. (จุด)

### แฟล็ก

ไอเท็ม	คำอธิบาย
-3	สร้างสคริปต์แก้ไขเพื่อรวมเฉพาะการเปลี่ยนแปลงที่มีแฟล็ก ====3
-E, -X	ซึ่งเหมือนกับ -e และ -x ตามลำดับ, แต่ทำงานกับการเปลี่ยนแปลงที่ซ้อนทับกัน (นั่นคือ, การเปลี่ยนแปลงที่จะถูกแฟล็ก ==== ในรายการปกติ) ต่างไป บรรทัดที่ซ้อนกันจากทั้งสองไฟล์ ถูกแทรกโดยสคริปต์แก้ไข ปิดล้อมด้วยบรรทัด <<<<<< และ >>>>>> อีอพชั่น -E ถูกใช้โดย Revision Control System (RCS) Merge เพื่อประกันว่าการเปลี่ยนแปลงที่ซ้อนทับกันในไฟล์ที่ผสมรวมจะถูกรักษาไว้ และแสดงให้ผู้ใช้ได้เห็น
-e	สร้างสคริปต์การแก้ไขสำหรับใช้กับคำสั่ง ed เพื่อรวมลงใน File1 การเปลี่ยนแปลงทั้งหมดระหว่าง File2 และ File3 (นั่นคือ การเปลี่ยนแปลงที่ปกติจะถูกแฟล็ก ==== และ ====3)
-x	สร้างสคริปต์แก้ไขเพื่อรวมเฉพาะการเปลี่ยนแปลงที่มีแฟล็ก ====

## ตัวอย่าง

เมื่อต้องการแสดงรายการความแตกต่างระหว่างสามไฟล์:

```
diff3 fruit.a fruit.b fruit.c
```

ถ้า fruit.a, fruit.b และ fruit.c มีข้อมูลต่อไปนี้:

fruit.a	fruit.b	fruit.c
banana	apple	grape
grape	banana	grapefruit
kiwi	grapefruit	kiwi
lemon	kiwi	lemon
mango	orange	mango
orange	peach	orange
peach	pear	peach
pare		

เอาต์พุตจากคำสั่ง `diff3` แสดงความแตกต่างระหว่างไฟล์เหล่านี้ดังต่อไปนี้ (หมายเหตุทางขวาไม่แสดงในเอาต์พุต)

```
==== All three files are different.
1:1,2c Lines 1 and 2 of the first file, fruit.a
  banana
  grape
2:1,3c Lines 1 through 3 of fruit.b
  apple
  banana
  grapefruit
3:1,2c Lines 1 and 2 of fruit.c
  grape
  grapefruit
====2 The second file, fruit.b, is different.
1:4,5c Lines 4 and 5 the same in fruit.a and fruit.c.
2:4a To make fruit.b look same, add after line 4.
3:4,5c
  lemon
  mango
==== The first file, fruit.a, is different.
1:8c
  pare
2:7c fruit.b line 7 and fruit.c line 8 are the same
  pear
3:7a
```

## ไฟล์

ไอเท็ม  
/usr/bin/diff3  
/usr/bin/diff3prog

คำอธิบาย  
ระบุคำสั่ง diff3  
ที่เรียกโดยเซลล์สคริปต์ diff3

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง ed หรือ red” ในหน้า 297

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง Files

การเปลี่ยนทิศทางอินพุตและเอาต์พุต

---

## คำสั่ง diffmk

### วัตถุประสงค์

ทำเครื่องหมายความแตกต่างระหว่างไฟล์

### ไวยากรณ์

```
diffmk [ { -abX| -aeX } [ -b ] [ -cbX| -ceX ] [ -dbX| -deX ] File1 File2 [ File3 ]
```

### คำอธิบาย

คำสั่ง **diffmk** เปรียบเทียบไฟล์ ภาษาอังกฤษที่ระบุโดยพารามิเตอร์ *File1* กับไฟล์พารามิเตอร์ *File2* จากนั้นจะสร้าง ไฟล์ที่สาม ที่มีการร้องขอของ **.mc** (สำหรับการสร้างเครื่องหมายการเปลี่ยนแปลง) สำหรับคำสั่ง **nroff** และ **troff** พารามิเตอร์ *File1* และ *File2* ระบุเวอร์ชันของไฟล์เก่าและใหม่ ตามลำดับ คำสั่ง **diffmk** เขียนไฟล์ที่สร้างใหม่ไปที่พารามิเตอร์ *File3*, ถ้าระบุ, หรือไปที่เอาต์พุต มาตรฐาน ไฟล์ *File3* มีบรรทัดของไฟล์ *File2* รวมทั้งการร้องขอ **.mc** ตัวจัดรูปแบบที่แทรก เมื่อไฟล์ *File3* ถูก จัดรูปแบบ ข้อความที่เปลี่ยนแปลงหรือที่แทรกถูกทำเครื่องหมายโดย **I** (แท่งแนวตั้ง) ทางขอบขวาของแต่ละบรรทัด \* (เครื่องหมายดอกจัน) ในขอบระนาบว่าบรรทัดถูกลบ

ถ้าตัวแปรสภาวะแวดล้อม **DIFFMARK** ถูกกำหนด, จะตั้งชื่อสตริงคำสั่งที่คำสั่ง **diffmk** ใช้ในการเปรียบเทียบไฟล์ (โดยปกติ คำสั่ง **diffmk** ใช้คำสั่ง **diff**) ตัวอย่าง, เพื่อจัดการไฟล์ที่มีขนาดใหญ่ มากได้ดีขึ้น คุณสามารถเซตตัวแปร **DIFFMARK** เป็น **diff -h**

### พารามิเตอร์

ไอเท็ม	คำอธิบาย
<i>File1</i>	ระบุไฟล์ภาษาอังกฤษที่ถูกเปรียบเทียบกับไฟล์ที่ระบุ โดยพารามิเตอร์ <i>File2</i> ผลลัพธ์ของการเปรียบเทียบ ประกอบด้วยไฟล์ที่ระบุโดยพารามิเตอร์ <i>File3</i> <i>File1</i> ถูกพิจารณาเป็นไฟล์ "เก่า"
<i>File2</i>	ระบุไฟล์ภาษาอังกฤษที่ถูกเปรียบเทียบกับไฟล์ที่ระบุ โดยพารามิเตอร์ <i>File1</i> ผลลัพธ์ของการเปรียบเทียบ ประกอบด้วยไฟล์ที่ระบุโดยพารามิเตอร์ <i>File3</i> <i>File2</i> ถูกพิจารณาเป็นไฟล์ "ใหม่"
<i>File3</i>	ระบุไฟล์ที่มีบรรทัดของไฟล์ <i>File2</i> และรวมตัวจัดรูปแบบที่แทรกการร้องขอ <b>.mc</b> สำหรับคำสั่ง <b>nroff</b> และ <b>troff</b> เนื้อหาของไฟล์นี้เป็นผลลัพธ์ของการเปรียบเทียบระหว่างไฟล์ ที่ระบุโดยพารามิเตอร์ <i>File1</i> และ <i>File2</i> เมื่อทำการฟอร์แมต ข้อความที่เปลี่ยนแปลงถูกทำเครื่องหมายโดย <b>I</b> แท่งแนวตั้ง ที่ด้านขอบขวาของแต่ละบรรทัด * (เครื่องหมายดอกจัน) ระบุว่าบรรทัด ถูกลบ ถ้าไม่มีการระบุ <i>File3</i> ผลลัพธ์ของ การเปรียบเทียบจะถูกเขียนไปที่อินพุตมาตรฐาน

## แฟล็ก

ไอเท็ม	คำอธิบาย
-abX	ใช้ X เพื่อทำเครื่องหมายการเริ่มต้นของบรรทัดที่เพิ่ม
-aeX	ใช้ X เพื่อทำเครื่องหมายจุดสิ้นสุดของบรรทัดที่เพิ่ม
-b	ละเว้นความแตกต่างที่เปลี่ยนแปลงเฉพาะแท็บหรือช่องว่างบน บรรทัด
-cbX	ใช้ X เพื่อทำเครื่องหมายจุดที่การเปลี่ยนแปลงในบรรทัดเริ่มต้น
-ceX	ใช้ X เพื่อทำเครื่องหมายจุดที่การเปลี่ยนแปลงในบรรทัดสิ้นสุด
-dbX	ใช้ X เพื่อทำเครื่องหมายจุดที่บรรทัดที่ลบเริ่มต้น
-deX	ใช้ X เพื่อทำเครื่องหมายจุดที่บรรทัดที่ลบสิ้นสุด

## ตัวอย่าง

1. เมื่อต้องการทำเครื่องหมายความแตกต่างระหว่างสองเวอร์ชันของ ไฟล์ข้อความ ให้ป้อน:

```
diffmk chap1.old chap1 chap1.nroff
```

นี้จะสร้างสำเนาของ chap1 ที่มีการร้องขอเครื่องหมายการเปลี่ยนแปลง nroff และ troff เพื่อแยกแยะข้อความที่ถูกเพิ่ม, เปลี่ยนแปลง หรือลบจาก chap1.old สำเนาที่ถูกบันทึกในไฟล์ chap1.nroff

2. เมื่อต้องการทำเครื่องหมายความแตกต่างโดยไม่มีข้อความ nroff และ troff ให้ป้อน:

```
diffmk -ab'>>New:' -ae'<<End New' \  
chap1.old chap1 chap1.nroff
```

ซึ่งทำให้คำสั่ง diffmk เขียน >>New: บนบรรทัด ก่อนส่วนของบรรทัดที่เพิ่มใหม่ให้กับ chap1, และ เขียน <<End New บน บรรทัดต่อจาก บรรทัดที่เพิ่ม การเปลี่ยนแปลงและการลบยังคงสร้างคำสั่ง nroff และ troff เพื่อใส่ (แท่งแนวตั้ง) หรือ \* (เครื่องหมายดอกจัน) ในขอบ

3. เมื่อต้องการใช้การร้องขอการทำเครื่องหมายคำสั่ง nroff และ troff อื่นและละเว้นการเปลี่ยนแปลงในช่องว่าง ให้ป้อน:

```
diffmk -b -cb'.mc %' chap1.old chap1 chap1.nroff
```

ซึ่งจะฝังคำสั่งที่เครื่องหมายเปลี่ยนแปลงด้วย % (เครื่องหมายเปอร์เซ็นต์) นอกจาก | (แท่งแนวตั้ง), และการลบด้วย \* (เครื่องหมายดอกจัน) ซึ่งไม่ได้ทำเครื่องหมายการเปลี่ยนแปลงที่เกี่ยวข้องเฉพาะจำนวนช่องว่าง หรือแท็บที่ต่างกัน ระหว่างคำ (-b)

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง diff” ในหน้า 150

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง nroff

คำสั่ง troff

---

## คำสั่ง dig

### วัตถุประสงค์

ยูทิลิตี้การค้นหา DNS

## ไวยากรณ์

```
dig [@server] [-b address] [-c class] [-f filename] [-k filename] [-p port#] [-q name] [-t type] [-x addr] [-y [hmac:]  
name:key] [-4] [-6] [name] [type] [class] [queryopt...]
```

```
dig [-h]
```

```
dig [global-queryopt...] [query...]
```

## คำอธิบาย

คำสั่ง **dig** (domain information groper) เป็นทูลที่ยืดหยุ่นสำหรับการสอบถาม DNS name servers คำสั่งทำการ ค้นหา DNS และแสดงคำตอบที่ถูกส่งกลับจากเซิร์ฟเวอร์ชื่อที่เคียวรี ผู้ดูแลระบบ DNS ส่วนใหญ่ใช้คำสั่ง **dig** เพื่อแก้ไขปัญหา DNS เนื่องจากความยืดหยุ่นง่ายต่อการใช้งาน และเอาต์พุตที่ชัดเจน แม้ว่า **dig** โดยปกติถูกใช้กับอาร์กิวเมนต์บรรทัดคำสั่ง ยังมีเบ็ตซ์ โหมดสำหรับการอ่าน การร้องขอการค้นหาจากไฟล์ ไม่เหมือนกับเวอร์ชันก่อนหน้านี้, การเพิ่ม BIND9 ของ **dig** อนุญาตให้เรียกใช้การค้นหาได้หลายครั้ง จากบรรทัดคำสั่ง นอกจากคำสั่งจะถูกแจ้งให้เคียวรีเซิร์ฟเวอร์ชื่อที่เจาะจง คำสั่ง **dig** จะพยายามค้นหาที่แต่ละเซิร์ฟเวอร์ ที่แสดงในไฟล์ `/etc/resolv.conf` ถ้า คุณไม่ได้ระบุอาร์กิวเมนต์บรรทัดคำสั่งหรืออ็อปชัน คำสั่ง **dig** จะทำการเคียวรี NS สำหรับ "." (root)

เป็นไปได้ที่จะเซตค่าดีฟอลต์ต่อผู้ใช้สำหรับคำสั่ง **dig** ผ่านไฟล์ `~/.digrc` คำสั่ง **dig** อ่านไฟล์นี้และใช้อ็อปชัน ก่อนอาร์กิวเมนต์บรรทัดคำสั่ง

ชื่อคลาส **IN** และ **CH** ซ้อนทับกับ **IN** และ **CH** ชื่อ โดเมนระดับบน เมื่อคุณค้นหาโดเมนระดับบนเหล่านี้ คุณสามารถใช้อ็อปชัน **-t** และ **-c** เพื่อระบุชนิดและคลาสหรือใช้อ็อปชัน **-q** เพื่อระบุโดเมนเนมหรือใช้ชื่อ **IN** และ **CH**

## แฟล็ก

ไอเท็ม	คำอธิบาย
<b>-b address</b>	ตั้งค่า IP แอดเดรสต้นทางของแอดเดรสที่ทำการเคียวรี คำนี้ต้องเป็นแอดเดรสที่ถูกต่อบนโฮสต์หนึ่งในอินเทอร์เน็ตเฟสเครือข่ายของโฮสต์ หรือ "0.0.0.0" หรือ "::" คุณสามารถระบุพอร์ตทางเลือกได้โดยการผนวก "#port"
<b>-c class</b>	ลบลงดีฟอลต์เคียวรีคลาส (IN สำหรับอินเทอร์เน็ต) ค่าพารามิเตอร์ <i>class</i> เป็นคลาสที่ใช้ได้ เช่น HS สำหรับเร็กคอร์ด Hesiod หรือ CH สำหรับ CHAOSNET
<b>-f filename</b>	ทำให้คำสั่ง <b>dig</b> ทำงานในเบ็ตซ์โหมดโดยการอ่านรายการของการร้องขอการค้นหาไปที่กระบวนการ จากชื่อไฟล์ที่ระบุ ไฟล์มีจำนวนของเคียวรี; หนึ่งต่อบรรทัด แต่ละรายการในไฟล์ต้องถูกจัดการในวิธีเดียวกัน กับที่แสดงเป็นเคียวรีกับคำสั่ง <b>dig</b> โดยใช้อินเทอร์เน็ตเฟสบรรทัดคำสั่ง
<b>-h</b>	พิมพ์ข้อมูลสรุปของอาร์กิวเมนต์บรรทัดคำสั่ง และอ็อปชัน
<b>-k filename</b>	ระบุไฟล์คีย์ TSIG โดยใช้อ็อปชัน <b>-k</b> เพื่อลงชื่อ DNS เคียวรีที่ส่งโดยคำสั่ง <b>dig</b>
<b>-p port#</b>	เคียวรีหมายเลขพอร์ตมาตรฐาน ค่าพารามิเตอร์ <i>port#</i> เป็นหมายเลขพอร์ตที่คำสั่ง <b>dig</b> ส่งเคียวรีไปแทน หมายเลขพอร์ต DNS 53 ที่เป็นมาตรฐาน คุณสามารถใช้อ็อปชันนี้เพื่อทดสอบ เซิร์ฟเวอร์ชื่อที่ถูกปรับแต่งให้รองรับเคียวรีรับ หมายเลขพอร์ตที่ไม่เป็นมาตรฐาน
<b>-q name</b>	แยกชื่อออกจากอาร์กิวเมนต์อื่น เซ็ตชื่อเคียวรีเป็นค่าพารามิเตอร์ <i>name</i> ที่ระบุ
<b>-t type</b>	เซตชนิดเคียวรีเป็นค่าพารามิเตอร์ <i>type</i> ซึ่งสามารถเป็นชนิดเคียวรีที่ใช้ได้ทั้งหมด ที่สนับสนุนใน BIND9 ชนิดดีฟอลต์เคียวรีคือ A, นอกจากว้ออ็อปชัน <b>-x</b> ถูกระบุเพื่อชี้ว่าเป็นการค้นหาย้อนกลับ การถ่ายโอนโซนสามารถถูกรองขอโดยการระบุ <i>type</i> ของ AXFR เมื่อ incremental zone transfer (IXFR) ต่อมมี, ค่าพารามิเตอร์ <i>type</i> ถูกเซตเป็น ixfr=N การถ่ายโอนโซนที่เพิ่มขึ้นมา มีการเปลี่ยนแปลงที่โต้ทำกับโซน เนื่องจากหมายเลขลำดับในเร็กคอร์ด SOA ของโซนเป็น N
<b>-x addr</b>	การค้นหาย้อนกลับแบบง่าย (การแม็ปแอดเดรส กับชื่อ) ค่าพารามิเตอร์ <i>addr</i> คือ IPv4 แอดเดรสในรูปแบบเลขฐานสิบที่มีจุด หรือ IPv6 แอดเดรสที่มีการแยกด้วยโคลอน เมื่อคุณใช้อ็อปชันนี้ไม่จำเป็นต้องให้อาร์กิวเมนต์ <i>name</i> , <i>class</i> , และ <i>type</i> คำสั่ง <b>dig</b> ทำการค้นหาโดยอัตโนมัติสำหรับชื่อเช่น 11.12.13.10. in-addr. arpa และ เซ็ตชนิดเคียวรีและคลาสเป็น PTR และ IN ตามลำดับ

ไอเอ็ม	คำอธิบาย
-y [hmac:]	ระบุตัวคีย์ TSIG เองในบรรทัดคำสั่ง; <i>hmac</i> เป็นชนิดของ TSIG ค่าดีฟอลต์คือ HMAC-MD5 ค่าพารามิเตอร์ <i>name</i> คือชื่อของคีย์
<i>name:key</i>	TSIG และค่าพารามิเตอร์ <i>key</i> เป็นคีย์จริง คีย์เป็นสตริงที่เข้ารหัสแบบ base-64, โดยปกติสร้างโดย <i>dnssec-keygen(8)</i> ต้องมีการระมัดระวัง เมื่อใช้อ็อปชัน -y กับระบบหลายผู้ใช้ เป็นคีย์สามารถเห็นได้ในเอาต์พุตจาก <i>ps(1)</i> หรือในไฟล์ประวัติของเชลล์ เมื่อใช้การพิสูจน์ตัวตน TSIG กับคำสั่ง <i>dig</i> เซิร์ฟเวอร์ชื่อที่ถูกเคียวรีจำเป็นต้อง รู้คีย์และอัลกอริทึมที่ใช้ ใน BIND, นี้ทำได้ โดยการระบุคีย์และข้อความเซิร์ฟเวอร์ที่เหมาะสมในไฟล์ <i>named.conf</i>
-4	บังคับให้คำสั่ง <i>dig</i> ใช้เพียงการรับส่งเคียวรี IPv4
-6	บังคับให้คำสั่ง <i>dig</i> ใช้เพียงการรับส่งเคียวรี IPv6

## พารามิเตอร์

ไอเอ็ม	คำอธิบาย
<i>global-queryopt...</i>	โกลบอลเคียวรีอ็อปชัน (ดูที่ การเคียวรีหลายเคียวรี)
<i>query</i>	เคียวรีอ็อปชัน (ดูที่ เคียวรี อ็อปชัน)

## เคียวรีอ็อปชัน

คำสั่ง *dig* มีจำนวนของเคียวรีอ็อปชัน ที่มีผลต่อการทำงานของการทำงานการค้นหา และผลลัพธ์ ที่แสดง บางอ็อปชันเซ็ทหรือรีเซ็ทแฟล็กบิตในส่วนหัวเคียวรี, บางส่วนกำหนดส่วนของคำตอบที่จะถูกพิมพ์, และบางส่วน กำหนดกลยุทธ์ของการหมดเวลาใช้งานและการทำซ้ำ แต่ละเคียวรีอ็อปชันถูกจำแนก โดยคีย์เวิร์ดที่นำหน้าด้วยเครื่องหมายบวก (+) บางคีย์เวิร์ดเซ็ทหรือรีเซ็ท อ็อปชันข้อมูลเหล่านี้สามารถถูกนำหน้าด้วยสตริง *no* เพื่อไม่ใช้ ความหมายของคีย์เวิร์ดนั้น คีย์เวิร์ดอื่นกำหนดค่าให้กับอ็อปชัน เช่น ระยะเวลาใช้งาน โดยมีรูปแบบ *+keyword=value* เคียวรีอ็อปชันมีดังนี้:

### +*[no]tcp*

ใช้หรือไม่ใช้ TCP เมื่อเคียวรีเซิร์ฟเวอร์ชื่อ การทำงานดีฟอลต์ คือใช้ UDP นอกจากมีการร้องขอเคียวรี AXFR หรือ IXFR, ซึ่งในกรณีดังกล่าวจะใช้การเชื่อมต่อ TCP

### +*[no]vc*

ใช้หรือไม่ใช้ TCP เมื่อเคียวรีเซิร์ฟเวอร์ชื่อ นี้เป็นไวยากรณ์ อีกแบบของ *+*[no]tcp** มีไว้เพื่อ คงความเข้ากันได้กับรุ่นก่อนหน้า *vc* หมายถึง virtual circuit

### +*[no]ignore*

ละเว้นการตัดข้อมูลในการตอบสนอง UDP แทนการทำซ้ำด้วย TCP โดยดีฟอลต์, การทำซ้ำ TCP จะถูกดำเนินการ

### +*domain=somename*

ตั้งค่ารายการค้นหาเพื่อให้มีโดเมนเดี่ยว *somename*, ถ้ามีการระบุในโดเมนใดเร็กทีฟในไฟล์ */etc/resolv.conf*, และเปิดใช้การค้นหา การประมวลผลรายการ เหมือนกับมีการระบุอ็อปชัน *+search*

### +*[no]search*

ใช้หรือไม่ใช้รายการค้นหาที่กำหนดโดยรายการค้นหา หรือโดเมนใดเร็กทีฟในไฟล์ */etc/resolv.conf* (ถ้ามี) โดยดีฟอลต์จะไม่ใช้รายการค้นหา

### +*[no]defname*

Deprecated, ปฏิบัติเหมือนเป็นคำพ้องสำหรับ *+*[no]search**

### +*[no]aaonly*

ตั้งค่าแฟล็ก "aa" ในเคียวรี

**+[no]adflag**

เช็คหรือไม่เช็คบิต AD (authentic data) ในเคียวรี ขณะนี้บิต AD มีความหมายมาตรฐานเฉพาะในการตอบกลับ ไม่ใช่ในเคียวรี แต่ความสามารถในการเช็คบิตในเคียวรีมีไว้เพื่อความสมบูรณ์

**+[no]cdflag**

เช็คหรือไม่เช็คบิต CD (checking disabled) ในเคียวรี การร้องขอให้เซิร์ฟเวอร์ไม่ต้องทำการตรวจสอบความถูกต้อง DNSSEC ของการตอบกลับ

**+[no]cl** แสดงหรือไม่แสดง CLASS เมื่อพิมพ์เร็กคอร์ด

**+[no]ttlid**

แสดงหรือไม่แสดง TTL เมื่อพิมพ์เร็กคอร์ด

**+[no]recursive**

สลับค่าติดตั้งของบิต RD (recursion desired) ใน คำถาม บิตนี้ถูกเช็คโดยดีฟอลต์ ซึ่งหมายความว่า dig โดยปกติส่งเคียวรีการเรียกซ้ำ การเรียกซ้ำถูกปิดใช้งานโดยอัตโนมัติเมื่อ เคียวรีอ็อปชัน +nssearch หรือ +trace ถูกใช้

**+[no]nssearch**

เมื่ออ็อปชันนี้ถูกเช็ค คำสั่ง dig จะพยายามค้นหาเซิร์ฟเวอร์ชื่อที่เชื่อถือได้ สำหรับโซนที่มี ชื่อที่ลิงค์หาและแสดงเร็กคอร์ด SOA ที่แต่ละเซิร์ฟเวอร์ชื่อ มีสำหรับโซน

**+[no]trace**

สลับการติดตามพาทัวแทน จากเซิร์ฟเวอร์ชื่อของ root สำหรับชื่อที่กำลังค้นหา การติดตามถูกปิดโดยดีฟอลต์ เมื่อการติดตาม ถูกเปิดใช้ คำสั่ง dig สร้างเคียวรีซ้ำ เพื่อแยกแยะชื่อที่กำลังถูกค้นหา ซึ่งติดตามการอ้างอิง จะเซิร์ฟเวอร์ root, แสดงคำตอบจากแต่ละเซิร์ฟเวอร์ที่ถูกใช้ เพื่อแยกแยะการค้นหา

**+[no]cmd**

สลับการพิมพ์ของหมายเหตุเบื้องต้น ในเอาต์พุตการระบุ เวอร์ชันของ dig และเคียวรีอ็อปชันที่ถูกใช้ หมายเหตุนี้ถูกพิมพ์โดยดีฟอลต์

**+[no]short**

ให้คำตอบที่กระชับ ค่าดีฟอลต์คือพิมพ์คำตอบใน แบบให้ข้อมูลมาก

**+[no]identify**

แสดงหรือไม่แสดง IP แอดเดรส และหมายเลขพอร์ตที่ให้กับ คำตอบเมื่อเปิดใช้อ็อปชัน +short ถ้ามีการร้องขอคำตอบแบบสั้น ค่าดีฟอลต์จะไม่ แสดงซอร์สแอดเดรสและหมายเลขพอร์ตของเซิร์ฟเวอร์ที่ จัดเตรียมคำตอบ

**+[no]comments**

สลับการแสดงบรรทัดหมายเหตุในเอาต์พุต ค่าดีฟอลต์ คือพิมพ์หมายเหตุ

**+[no]stats**

สลับการพิมพ์สถิติ: เมื่อเคียวรีถูกสร้าง ขนาดของการตอบกลับ และอื่นๆ การทำงานดีฟอลต์คือพิมพ์ สถิติเคียวรี

**+[no]qr**

พิมพ์หรือไม่พิมพ์เคียวรีขณะที่ส่งข้อมูล โดยดีฟอลต์ เคียวรีจะไม่ถูกพิมพ์

**+[no]question**

พิมพ์หรือไม่พิมพ์ส่วนคำถามของเคียวรี เมื่อ คำตอบถูกส่งกลับ ค่าดีฟอลต์คือพิมพ์ส่วนคำถามเป็น ดีฟอลต์

**+[no]answer**

แสดงหรือไม่แสดงส่วนคำตอบของการตอบกลับ ค่าดีฟอลต์ คือแสดง

**+[no]authority**

แสดงหรือไม่แสดงส่วนสิทธิ์ของการตอบกลับ คำดีพอลต์ คือแสดง

**+[no]additional**

แสดงหรือไม่แสดงส่วนเพิ่มเติมของการตอบกลับ คำดีพอลต์ คือแสดง

**+[no]all**

เซ็ทหรือเคลียร์แฟล็กการแสดงผลทั้งหมด

**+time=T**

เซ็ทหมดเวลาใช้งานสำหรับเคียวรีเป็น  $T$  วินาที ค่าการหมดเวลาดีพอลต์ คือ 5 วินาที ความพยายามเซ็ทค่าพารามิเตอร์  $T$  ให้น้อยกว่า 1 มีผลให้ค่าหมดเวลาใช้งานของเคียวรี 1 วินาทีถูกนำมาใช้

**+tries=A**

เซ็ทจำนวนครั้งในการเคียวรี UDP ไปที่เซิร์ฟเวอร์เป็นค่าพารามิเตอร์  $A$  แทนคำดีพอลต์ 3 ถ้าค่าพารามิเตอร์  $A$  น้อยกว่าหรือ เท่ากับศูนย์ จำนวนการพยายามซ้ำจะถูกปิดเป็น 1

**+retry=T**

เซ็ทจำนวนครั้งในการเคียวรี UDP ไปที่เซิร์ฟเวอร์เป็นค่าพารามิเตอร์  $T$  แทนคำดีพอลต์ 2 ไม่เหมือนกับ +tries, กรณีนี้ไม่รวม เคียวรีเริ่มต้น

**+ndots=D**

เซ็ทจำนวนของจุดที่แสดงในชื่อเป็นค่าพารามิเตอร์  $D$  เนื่องจากถูกพิจารณาว่าเป็นค่าสัมบูรณ์ คำดีพอลต์คือหนึ่ง ซึ่งถูกกำหนดโดยใช้คำสั่ง ndots ในไฟล์ /etc/resolv.conf หรือ 1 ถ้าไม่มีคำสั่ง ndots อยู่ในชื่อที่มีจุดน้อยกว่า ถูกแปลเป็นชื่อสัมพัทธ์และถูกค้นหาในโดเมนที่แสดงในไดเรกทีฟการค้นหาหรือโดเมนในไฟล์ /etc/resolv.conf

**+bufsize=B**

เซ็ทขนาดบัฟเฟอร์ข้อความ UDP ที่ประกาศโดยใช้ EDNSO เป็นไบต์  $B$  ขนาดสูงสุดและต่ำสุดของ บัฟเฟอร์นี้คือ 65535 และ 0, ตามลำดับ ค่าภายนอกช่วงนี้ ถูกบัตขึ้นหรือลงตามความเหมาะสม ค่าอื่นที่ไม่ใช่ศูนย์ ทำให้เคียวรี EDNS ถูกส่ง

**+edns=#**

ระบุเวอร์ชัน EDNS เพื่อทำการเคียวรี ค่าที่ใช้ได้คือ 0 ถึง 255 การตั้งค่าเวอร์ชัน EDNS ทำให้เคียวรี EDNS ถูกส่ง +noedns ลบเวอร์ชัน EDNS ที่จำไว้

**+[no]multiline**

พิมพ์เร็กคอร์ดเหมือนกับเร็กคอร์ด SOA ในรูปแบบหลายบรรทัด พร้อมกับหมายเหตุที่อ่านเข้าใจได้ ดีพอลต์คือพิมพ์แต่ละเร็กคอร์ด บนบรรทัดเดียวเพื่อช่วยการวิเคราะห์ค่าของเครื่องของเอาต์พุต dig

**+[no]fail**

อย่าลองกับเซิร์ฟเวอร์ถัดไปถ้าคุณได้รับ SERVFAIL ดีพอลต์คือไม่ลองกับเซิร์ฟเวอร์ถัดไป ซึ่งคือการย้อนกลับของการทำงานตัวแยกแยะ stub ปกติ

**+[no]besteffort**

พยายามแสดงเนื้อหาของข้อความที่รูปแบบไม่ถูกต้อง ดีพอลต์คือไม่แสดงคำตอบที่รูปแบบไม่ถูกต้อง

**+[no]dnssec**

ร้องขอเร็กคอร์ด DNSSEC ให้ถูกส่งโดยการเซ็ทบิต DNSSEC OK (DO) ในเร็กคอร์ด OPT ในส่วนเพิ่มเติมของเคียวรี

### **+[no]sigchase**

ติดตามลูกโซ่หลายชั้น DNSSEC ต้องการให้คำสั่ง **dig** ถูกคอมไพล์ด้วย -DDIG SIGCHASE

### **+trusted-key=# # # #**

ระบุไฟล์ที่มีคีย์ที่เชื่อถือเพื่อใช้กับ +sigchase แต่ละเร็กคอร์ด DNSKEY ต้องอยู่ในบรรทัด ของตัวเอง ถ้าไม่มีการระบุคำสั่ง **dig** จะค้นหา ไฟล์ /etc/trusted-key.key ตามด้วยไฟล์ trusted-key.key ในไดเรกทอรีปัจจุบัน ต้องการให้คำสั่ง **dig** ถูกคอมไพล์ด้วย -DDIG SIGCHASE

### **+[no]topdown**

เมื่อติดตามลูกโซ่หลายชั้น DNSSEC, ทำการตรวจสอบความถูกต้องจากบนลงล่าง ต้องการให้คำสั่ง **dig** ถูกคอมไพล์ด้วย -DDIG SIGCHASE

## **เคียวรีหลายเคียวรี**

การนำ BIND 9 มาใช้ของ **dig** สนับสนุน การระบุหลายเคียวรีบนบรรทัดคำสั่ง (นอกเหนือจากการสนับสนุน อ็อพชันแบ็ตซ์ไฟล์ -f) เคียวรีแต่ละเคียวรี สามารถรับชุดแฟล็กของตัวเอง อ็อพชันและเคียวรีอ็อพชัน

ในกรณีนี้แต่ละเคียวรีอาร์กิวเมนต์แสดงแต่ละเคียวรี ในไวยากรณ์บรรทัดคำสั่ง ซึ่งประกอบด้วย อ็อพชัน และ แฟล็กมาตรฐาน, ชื่อที่จะถูกค้นหา, ชนิดเคียวรีทางเลือก, คลาส และเคียวรีอ็อพชันที่ต้องระบุให้กับเคียวรีนั้น

ชุดโกลบอลของอ็อพชันเคียวรีซึ่ง ต้องถูกนำมาใช้กับเคียวรีทั้งหมด สามารถถูกระบุได้เช่นกัน โกลบอลเคียวรีอ็อพชันเหล่านี้ ต้องนำหน้าลำดับค่าชื่อแรก, คลาส, ชนิด, อ็อพชัน, แฟล็ก และเคียวรีอ็อพชันที่ระบุบนบรรทัดคำสั่ง โกลบอลเคียวรีอ็อพชัน (ยกเว้นอ็อพชัน +[no]cmd) สามารถ ถูกเขียนทับได้โดยชุดเคียวรีจำเพาะของเคียวรีอ็อพชัน ตัวอย่างเช่น:

```
dig +qr www.isc.org any -x 127.0.0.1 isc.org ns +noqr
```

สตริงคำสั่ง **dig** นี้แสดงวิธีที่คำสั่ง **dig** จะถูกใช้จากบรรทัดคำสั่งเพื่อสร้างการค้นหาสามการค้นหา: เคียวรี ทั้งหมดสำหรับ **www.isc.org**, การค้นหาย้อนกลับของ 127.0.0.1, และเคียวรีสำหรับเร็กคอร์ด NS ของ **isc.org** โกลบอล เคียวรีอ็อพชันของ +qr ถูกนำมาใช้ เพื่อให้ คำสั่ง **dig** แสดงเคียวรีเริ่มต้นที่สร้าง สำหรับแต่ละการค้นหา เคียวรีสุดท้ายมีโลคัลเคียวรีอ็อพชันของ +noqr, ซึ่งหมายถึงว่าคำสั่ง **dig** ไม่พิมพ์เคียวรีเริ่มต้นเมื่อทำการค้นหาเร็กคอร์ด NS สำหรับ **isc.org**

## **การสนับสนุน IDN**

ถ้าคำสั่ง **dig** ได้ถูกสร้างโดยมีการสนับสนุน internationalized domain name (IDN) คำสั่งสามารถรับและแสดงโดเมนเนม non-ASCII ได้ คำสั่ง **dig** แปลงการเข้ารหัสอักขระ ของโดเมนเนมอย่างเหมาะสม ก่อนการส่งการร้องขอไปที่เซิร์ฟเวอร์ DNS หรือแสดงการตอบกลับจากเซิร์ฟเวอร์ ถ้าคุณต้องการ ปิดการสนับสนุน IDN ให้กำหนดตัวแปรสภาวะแวดล้อม IDN DISABLE ; การสนับสนุน IDN ต่อไปนี้ถูกปิดใช้งาน ถ้าตัวแปร ถูกเซตเมื่อคำสั่ง **dig** รัน

## **ตัวอย่าง**

การร้องขอตามปกติของ **dig** มีลักษณะดังนี้:

```
dig @server name type
```

โดยที่:

**server** ชื่อหรือ IP แอดเดรสของเซิร์ฟเวอร์ชื่อที่เคียวรี ซึ่งสามารถเป็น IPv4 แอดเดรสในรูปแบบเลขฐานสิบที่มีจุด หรือ IPv6 แอดเดรสที่มี การแยกด้วยโคลอน เมื่อเซิร์ฟเวอร์อาร์กิวเมนต์ที่ระบุเป็นชื่อโฮสต์, คำสั่ง **dig** วิเคราะห์ชื่อนั้น

ก่อนทำการเคียวรี เซิร์ฟเวอร์ชื่อ ถ้าไม่มีการกำหนดเซิร์ฟเวอร์อาร์กิวเมนต์ คำสั่ง **dig** ค้นหาข้อมูลในไฟล์ `/etc/resolv.conf` และเคียวรี เซิร์ฟเวอร์ชื่อตามรายการในนั้น การตอบกลับจากเซิร์ฟเวอร์ชื่อที่ตอบสนองถูกแสดง

*name* ชื่อของรีซอร์สเร็กคอร์ดที่ถูกค้นหา

*type* ระบุชนิดของเคียวรีที่ต้องการ — **ANY, A, MX, SIG**, และอื่นๆ ค่าอาร์กิวเมนต์ *type* ที่ใช้สามารถเป็น ชนิดเคียวรีที่ถูกต้องทุกแบบ ถ้าไม่มีการกำหนดอาร์กิวเมนต์ *type* คำสั่ง **dig** ทำการค้นหา เร็กคอร์ด **A**

## ไฟล์

ไอเท็ม คำอธิบาย  
`/etc/resolv.conf`  
`${HOME}/.digrc`

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `host9`” ในหน้า 794

“คำสั่ง `dnssec-keygen`” ในหน้า 178

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `named9`

คำสั่ง `named-checkconf`

---

## คำสั่ง `digest`

### Purpose

แปลงรูปแบบ ASCII ของไฟล์ `/etc/qconfig` เป็นไฟล์ `/etc/qconfig.bin` ไบนารี เวอร์ชันของคิว configuration ที่ใช้โดยคำสั่ง `qdaemon` คำสั่งนี้ไม่ควรถูกป้อนบนบรรทัดคำสั่ง; คำสั่งนี้ถูกเรียกโดยคำสั่ง `qdaemon`

### ไวยากรณ์

`/usr/lib/lpd/digest ASCIIFile BinaryFile`

### คำอธิบาย

คำสั่ง `digest` ยอมรับ อินพุตไฟล์ที่เป็นอักขระ ASCII และแปลงไปเป็นไบนารีไฟล์ คำสั่งนี้ ถูกใช้เฉพาะโดยคำสั่ง `qdaemon` เพื่อแปลงไฟล์ `/etc/qconfig` เป็นไบนารีเวอร์ชันของไฟล์, ไฟล์ `/etc/qconfig.bin`

## ไฟล์

ไอเท็ม	คำอธิบาย
/etc/qconfig	มีไฟล์คอนฟิกูเรชันคิว
/usr/sbin/qdaemon	มีการคิว daemon
/etc/qconfig.bin	มีเวอร์ชันไบนารีแบบย่อของไฟล์/etc/qconfig

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง qdaemon

## คำสั่ง dircmp

### วัตถุประสงค์

เปรียบเทียบสองไดเรกทอรีและเนื้อหาของไฟล์ปกติ

### ไวยากรณ์

```
dircmp[ -d ][ -s ][ -w num ] Directory1 Directory2
```

### คำอธิบาย

คำสั่ง **dircmp** เปรียบเทียบ สองไดเรกทอรีที่ระบุโดยพารามิเตอร์ *Directory1* และ *Directory2* และเขียนข้อมูลเกี่ยวกับเนื้อหาของไดเรกทอรีไปที่เอาต์พุตมาตรฐาน อันดับแรก คำสั่ง **dircmp** เปรียบเทียบชื่อไฟล์ในแต่ละไดเรกทอรี ถ้าชื่อไฟล์เหมือนกัน แสดงใน ทั้งสองไดเรกทอรี, คำสั่ง **dircmp** เปรียบเทียบเนื้อหาของทั้งสอง ไฟล์

ในเอาต์พุตคำสั่ง **dircmp** แสดง ไฟล์ที่ไม่ซ้ำในแต่ละไดเรกทอรี จากนั้นแสดงไฟล์ที่มีชื่อเหมือนกัน ในทั้งสองไดเรกทอรี แต่มีเนื้อหาต่างกัน ถ้าไม่มีการระบุแฟล็ก จะแสดงไฟล์ที่มีเนื้อหาเหมือนกันเช่นเดียวกับชื่อที่เหมือนกัน ในทั้งสองไดเรกทอรี

คำสั่ง **diff -r** มีฟังก์ชันเหมือนกับคำสั่ง **dircmp**

### แฟล็ก

ไอเท็ม	คำอธิบาย
-d	แสดงเวอร์ชันของเนื้อหาไฟล์ที่ต่างกัน สำหรับแต่ละชื่อไฟล์ทั่วไปรูปแบบที่แสดงเหมือนกับในคำสั่ง <b>diff</b>
-s	ไม่แสดงชื่อไฟล์ที่เหมือนกัน
-w	เปลี่ยนความกว้างของเอาต์พุตเป็นจำนวนของอักขระ <i>num</i>
<i>num</i>	

### สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

- ไอเท็ม คำอธิบาย  
0 ดำเนินการเป็นผลสำเร็จ  
>0 มีข้อผิดพลาดเกิดขึ้น

หมายเหตุ: ความแตกต่าง ในเนื้อหาไดเรกทอรีไม่ถือว่าเป็นข้อผิดพลาด

## ตัวอย่าง

1. เมื่อต้องการทำสรุปความแตกต่างระหว่างไฟล์ใน สองไดเรกทอรี ให้พิมพ์ดังต่อไปนี้:

```
dircmp proj.ver1 proj.ver2
```

นี้จะแสดงสรุปของความแตกต่างระหว่างไดเรกทอรีproj.ver1 และ proj.ver2 รายการสรุปแยกไฟล์ที่พบเฉพาะในหนึ่งไดเรกทอรีหรืออีกไดเรกทอรี และที่พบในทั้งสองไดเรกทอรี ถ้าไฟล์พบในทั้งสองไดเรกทอรี คำสั่ง `dircmp` จะทำหมายเหตุว่าเหมือนกันหรือไม่

2. เมื่อต้องการแสดงรายละเอียดของความแตกต่างระหว่าง ไฟล์ ให้พิมพ์ดังนี้:

```
dircmp -d -s proj.ver1 proj.ver2
```

แฟล็ก `-s` ยับยั้งข้อมูลเกี่ยวกับไฟล์ที่เหมือนกัน แฟล็ก `-d` แสดง `diff` แสดงแต่ละไฟล์ที่ต่างกัน ที่พบในทั้งสองไดเรกทอรี

3. เมื่อต้องการแสดงรายละเอียดของความแตกต่างระหว่างไฟล์ด้วยความกว้างของ บรรทัดเอาต์พุตเซ็ทเป็น 90 อักขระ ให้พิมพ์ดังนี้:

```
$dircmp -w 90 dir1 dir2
```

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/dircmp</code>	มีคำสั่ง <code>dircmp</code>

## ข้อมูลที่เกี่ยวข้อง

คำสั่ง `cmp`, คำสั่ง `diff`

ไดเรกทอรี ใน *Operating system and device management* อธิบาย โครงสร้างและคุณสมบัติของไดเรกทอรีในระบบไฟล์

การเปลี่ยนทิศทางอินพุตและเอาต์พุต ใน *Operating system and device management* อธิบายวิธี ที่ระบบปฏิบัติการประมวลผลอินพุตและเอาต์พุต

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `diff`” ในหน้า 150

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `cmp`

คำสั่งไดเรกทอรี

การเปลี่ยนทิศทางอินพุตและเอาต์พุต

---

## คำสั่ง `dirname`

### วัตถุประสงค์

เขียนข้อมูลทั้งหมดนอกจากส่วนสุดท้ายของ พาทที่ระบุไปที่เอาต์พุตมาตรฐาน

### ไวยากรณ์

`dirname Path`

### คำอธิบาย

คำสั่ง `dirname` อ่านชื่อพาทที่ระบุ ลบทั้งหมดยกเว้น `/` (slash) สุดท้ายและอักขระที่ต่อท้าย, และเขียนผลลัพธ์ไปที่เอาต์พุตมาตรฐาน ถ้าไม่มีอักขระที่ตามด้วย `/` สุดท้าย, คำสั่ง `dirname` จะใช้ `/` ที่อยู่ถัดจากสุดท้าย แล้วละเว้นอักขระทั้งหมดที่ตามมา คำสั่ง `dirname` ใช้กฎต่อไปนี้ในการสร้างชื่อพาท:

1. ถ้าพารามิเตอร์ `Path` เป็น `//` (สแลชคู่) หรือถ้าพารามิเตอร์ `Path` ประกอบด้วยอักขระ สแลชทั้งหมด ให้เปลี่ยน สตริงเป็น `/` (สแลช) เดียวข้าม ขั้นตอน 2 ถึง 7
2. ลบอักขระ `/` ท้ายจากพาท ที่ระบุ
3. ถ้าไม่มีอักขระ `/` เหลือในพารามิเตอร์ `Path` ให้เปลี่ยนพาทเป็น `.` (จุด) เดียวข้ามขั้นตอน 4 ถึง 7
4. ลบส่วนท้ายที่ไม่ใช่อักขระ slash ออกจาก พาท
5. ถ้าพาทส่วนที่เหลือคือ `//` (slash คู่), ให้ไปที่ขั้นตอนที่ 6
6. ลบอักขระ slash ส่วนท้ายออกจาก พาท
7. ถ้าพาทที่เหลืออยู่ว่าง ให้เปลี่ยน พาทเป็น `/` เดียว

ตัวอย่าง ป้อน:

```
dirname //
```

ผลลัพธ์ใน `/` (slash) เดียว ป้อน:

```
dirname /a/b/
```

ผลลัพธ์ใน `/a` ป้อน:

```
dirname a
```

ผลลัพธ์ใน `.` (จุด) ป้อน:

```
dirname a/b
```

ผลลัพธ์ในชื่อพาท `a`

คำสั่ง `dirname` และ `basename` โดยปกติถูกใช้ภายในการแทนที่คำสั่ง ภายในเชลล์โปรแกรมเมอร์เพื่อระบุชื่อไฟล์เอาต์พุตที่มีความผันแปรของชื่อไฟล์อินพุตที่ระบุ

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

เมื่อต้องการสร้างชื่อของไฟล์ที่อยู่ในไดเรกทอรีเดียวกัน กับไดเรกทอรีอื่น ให้ป้อน:

```
AOUTFILE=`dirname $TEXTFILE`/a.out
```

นี่เป็นการเซตตัวแปรเชลล์ AOUTFILE ให้เป็นชื่อของไฟล์ a.out ที่อยู่ใน ไดเรกทอรีเดียวกับ TEXTFILE ถ้า TEXTFILE คือ /home/fran/prog.c, ค่าของ dirname \$TEXTFILE คือ /home/fran และ AOUTFILE กลายเป็น /home/fran/a.out

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/dirname	มีคำสั่ง dirname

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง basename

คำสั่ง sh

---

## คำสั่ง disable

คำสั่ง disable มีข้อมูลสำหรับ AIX Print Subsystem disable และ System V Print Subsystem disable

## วัตถุประสงค์

ปิดใช้งานอุปกรณ์พริ้นเตอร์คิว

## ไวยากรณ์

```
disable [ -c ] [ -rReason ] PrinterName ...
```

## คำอธิบาย

คำสั่ง disable ปิดใช้งาน หรือออฟไลน์อุปกรณ์พริ้นเตอร์คิวที่ระบุโดยพารามิเตอร์ PrinterName

หมายเหตุ: คุณต้องมีสิทธิ์ผู้ใช้ root หรือเป็นสมาชิกกลุ่ม group ในการรันคำสั่งนี้

## แฟล็ก

ไอเท็ม	คำอธิบาย
-c	ยกเลิกการร้องของานทั้งหมด การใช้แฟล็กนี้เหมือนกับ การป้อน คำสั่ง <code>enq -K</code>
-rReason	ระบุเหตุผลสำหรับการปิดใช้งานอุปกรณ์พริ้นเตอร์ด้วย ตัวแปร Reason แฟล็กนี้เป็นแฟล็กที่ “ไม่มีการดำเนินการ” ซึ่งหมายความว่าระบบจะข้ามแฟล็กนี้

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ตัวอย่าง

- เมื่อต้องการออฟไลน์พริ้นเตอร์คิว lp0 โดยไม่ต้องรอกงานพิมพ์ปัจจุบันให้เสร็จสิ้น พิมพ์:

```
disable -c lp0
```

- เมื่อต้องการออฟไลน์พริ้นเตอร์คิว lp0 หลังจากงานพิมพ์ทั้งหมดเสร็จสิ้น ให้พิมพ์:

```
disable lp0
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/sbin/qdaemon	daemon การจัดการคิว
/etc/qconfig	ไฟล์คอนฟิกูเรชันคิว
/etc/qconfig.bin	มีส่วนย่อย, ไบนารีเวอร์ชันของไฟล์ /etc/qconfig
/var/spool/lpd/qdir/*	การร้องขอคิว
/var/spool/lpd/stat/*	ข้อมูลบนสถานะของอุปกรณ์
/var/spool/qdaemon/*	สำเนาชั่วคราวของไฟล์ที่คิว

## คำสั่ง System V Print Subsystem disable

### วัตถุประสงค์

ปิดทำงานพริ้นเตอร์ LP

### ไวยากรณ์

```
disable [flags] printers
```

### คำอธิบาย

คำสั่ง `disable` หยุดการทำงาน *พริ้นเตอร์* ที่มีการกำหนดชื่อ, ปิดทำงานจากการพิมพ์การร้องขอที่ส่งโดย lp โดยดีฟอลต์ การร้องขอที่กำลังพิมพ์อยู่บนพริ้นเตอร์ที่กำหนด จะถูกพิมพ์ทั้งหมดบนพริ้นเตอร์เดียวกันหรือ บนสมาชิกของพริ้นเตอร์คลาสเดียวกัน ถ้าพริ้นเตอร์เป็นแบบรีโมต คำสั่งนี้จะหยุดการส่งข้อมูลงานไปที่ระบบรีโมตเท่านั้น คำสั่ง `disable` ต้องถูกรันบนระบบรีโมต เพื่อปิดการทำงานพริ้นเตอร์ (รัน `lpstat -p` เพื่อรับสถานะของพริ้นเตอร์)

ชื่อพริ้นเตอร์เป็น *ค่าที่กำหนดโดยระบบ* ดังนั้น ควรถูกจำกัดเป็นอักขระ ASCII ตัวพิมพ์ใหญ่และตัวพิมพ์เล็ก

ถ้าคุณป้อน `disable -?`, ระบบจะแสดงข้อความการใช้ คำสั่งและส่งกลับ 0

## แฟล็ก

`-c` ยกเลิกการร้องขอที่พิมพ์อยู่ขณะนี้ บนพรินเตอร์ที่มีการกำหนด แฟล็กนี้ไม่สามารถถูกใช้กับแฟล็ก `-W` ถ้าพรินเตอร์เป็นแบบรีโมต แฟล็ก `-c` จะถูกละเว้น

`-r reason`

กำหนด *reason* สำหรับการปิดพรินเตอร์ *reason* นี้ใช้กับ *พรินเตอร์* ทั้งหมดที่ระบุ *reason* นี้ถูกรายงานโดย `lpstat -p reason` ต้องอยู่ในเครื่องหมายคำพูด ถ้ามีช่องว่าง เหตุผลที่พอลต์คือ *unknown reason* สำหรับพรินเตอร์ที่มีอยู่และ *new printer* สำหรับพรินเตอร์ที่เพิ่งเพิ่มให้กับระบบ แต่ยังไม่ได้อ่าน

`-W` ระงับการร้องขอที่ถูกพิมพ์ในขณะนี้เสร็จสิ้นก่อนการปิดทำงาน พรินเตอร์ที่ระบุ แฟล็กนี้ไม่สามารถใช้กับแฟล็ก `-c` ถ้าพรินเตอร์เป็นแบบรีโมต แฟล็ก `-W` จะถูกละเว้น

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ไฟล์

`/var/spool/lp/*`

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `enable`” ในหน้า 382

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `cancel`

คำสั่ง `lp`

การเริ่มทำงานและหยุดทำงานคิวงานพิมพ์

---

## คำสั่ง `diskusg`

### วัตถุประสงค์

สร้างข้อมูลแอดเคาต์ดิสก์ตาม ID ผู้ใช้

### ไวยากรณ์

```
diskusg [-X] [-U MaxUsers] [ -i FileListName ] [ -p File ] [ -u File ] [ -v ] {  
-s [ File ... ] | FileSystem ... }
```

## คำอธิบาย

คำสั่ง **diskusg** สร้าง ข้อมูลแคคเคาต์ดิสก์ จากข้อมูลในไฟล์ที่ระบุ ด้วยพารามิเตอร์ *File* หรือ *FileSystem* หรือจากอินพุตมาตรฐาน คำสั่ง **diskusg** เขียน หนึ่งเรีกคอร์ดต่อผู้ใช้ไปที่เอาต์พุตมาตรฐาน คำสั่งนี้ถูกเรียกโดยคำสั่ง **odisk** ซึ่งรันได้ภายใต้ **cron daemon** เอาต์พุตอยู่ในรูปแบบต่อไปนี้:

ไอเท็ม	คำอธิบาย
UID	มี ID ผู้ใช้เป็นตัวเลขของผู้ใช้
Login	มีชื่อล็อกอินของผู้ใช้
Blocks	มีจำนวนรวมบล็อกดิสก์ 512 ไบต์ที่จัดสรรให้กับ ผู้ใช้

เอาต์พุตของคำสั่งนี้กลายเป็นอินพุตของคำสั่ง **acctdisk** ซึ่งแปลง ข้อมูลไปเป็นเรีกคอร์ดแคคเคาต์รวม เรีกคอร์ดแคคเคาต์รวม ถูกผนวกกับเรีกคอร์ดแคคเคาต์รวมอื่นเพื่อสร้างรายงานประจำวัน

ถ้าคุณระบุพารามิเตอร์ *FileSystem* คำสั่ง **diskusg** จะอ่าน i-nodes ของระบบไฟล์ ที่ระบุเพื่อสร้างข้อมูลการใช้ พารามิเตอร์ *FileSystem* ต้องเป็นชื่อไฟล์พิเศษของอุปกรณ์ระบบไฟล์ ตัวอย่าง ใช้อุปกรณ์ **/dev/hd4** แทนไดเร็กทอรี **/** (root) เพื่อสร้างข้อมูลการใช้สำหรับ ระบบไฟล์ root

ถ้าคุณระบุพารามิเตอร์ *File* , อินพุตต้องเป็นรูปแบบเอาต์พุต **diskusg**

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการใช้งานดิสก์ ดูที่คำสั่ง **acctdusg**

หมายเหตุ: คำสั่งนี้ ใช้สำหรับอุปกรณ์โลคัลเท่านั้น

## แฟล็ก

ไอเท็ม	คำอธิบาย
-i <i>FileListName</i>	ละเว้นข้อมูลในระบบไฟล์ <i>FileListName</i> ตัวแปร <i>FileListName</i> ระบุรายการของ ชื่อระบบไฟล์ที่แยกโดยคอมมา หรือถูกปิดอยู่ในเครื่องหมายคำพูด
-p <i>File</i>	ใช้ไฟล์รหัสผ่านที่ระบุโดยตัวแปร <i>File</i> เพื่อสร้างชื่อล็อกอิน ดีพอลต์คือไฟล์ <b>/etc/passwd</b>
-s [ <i>File</i> ]	รวมเรีกคอร์ดทั้งหมดจากไฟล์อินพุตหรือจากอินพุตมาตรฐาน ลงในเรีกคอร์ดเดียว ข้อมูลอินพุตมีอยู่แล้วในรูปแบบเอาต์พุต <b>diskusg</b>
-U <i>MaxUsers</i>	เซตจำนวนผู้ใช้มากที่สุดที่สามารถถูกประมวลผลได้โดยคำสั่ง <b>diskusg</b> คุณจำเป็นต้องใช้แฟล็กนี้เฉพาะเมื่อจำนวนของผู้ใช้มากกว่าค่าดีพอลต์ 5000
-u <i>File</i>	เขียนเรีกคอร์ดไปที่ตัวแปร <i>File</i> ที่ระบุ สำหรับแต่ละไฟล์ที่ถูกซาร์จให้กับ ID ผู้ใช้ที่ไม่มีอยู่ แต่ละเรีกคอร์ดประกอบด้วย ชื่อไฟล์พิเศษ หมายเลข i-node และ ID ผู้ใช้
-v	เขียนรายการของไฟล์ที่ถูกซาร์จให้กับผู้ใช้ที่ไม่มีอยู่กับเอาต์พุตข้อผิดพลาด มาตรฐาน
-X	พิมพ์และประมวลผลอักขระที่มีทั้งหมดสำหรับ แต่ละชื่อผู้ใช้ แทนการตัดให้เหลือ 8 อักขระแรก

## ความปลอดภัย

การควบคุมสิทธิ์เข้าถึง: คำสั่งนี้ควรให้สิทธิ์ในการเรียกใช้งาน (x) สำหรับการเข้าถึงให้กับสมาชิกของกลุ่ม **adm**

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

เมื่อต้องการสร้างข้อมูลแอดเคาต์ดิสก์รายวัน ให้เพิ่ม บรรทัดเหมือนดังต่อไปนี้ให้กับไฟล์ `/var/spool/cron/crontab/root` :

```
0 2 * * 4 /usr/sbin/acct/dodisk
```

คำสั่งนี้บอก cron daemon ให้รัน คำสั่ง `dodisk` เวลา 2 a.m. (02) ทุกวันพฤหัสบดี (4) คำสั่ง `dodisk` เรียกทั้งคำสั่ง `diskusg` และ `acctdisk`

หมายเหตุ: เมื่อต้องการดำเนินการตัวอย่างนี้ คุณ ต้องมีสิทธิ์ root

## ไฟล์

### ไอเท็ม

`/usr/sbin/acct/diskusg`  
`/etc/passwd`

### คำอธิบาย

มีคำสั่ง `diskusg`  
มีแอตทริบิวต์ระดับต้นของผู้ใช้

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `acctdisk`

รูทีนย่อย `acct`

รูปแบบไฟล์ `acct`

การจัดทำแอดเคาต์ระบบ

---

## คำสั่ง `dispgid`

### วัตถุประสงค์

แสดงรายการของชื่อกลุ่มที่ใช้ได้ทั้งหมด

### ไวยากรณ์

`dispgid`

### คำอธิบาย

คำสั่ง `dispgid` สามารถใช้เพื่อแสดงรายการของชื่อกลุ่มทั้งหมดบนระบบ (หนึ่งชื่อต่อบรรทัด) ไม่มีพารามิเตอร์สำหรับคำสั่งนี้ ไฟล์ต่อไปนี้ถูกเข้าถึงในโหมดอ่านอย่างเดียวเพื่อเรียก ข้อมูล:

- `/etc/passwd`
- `/etc/group`
- `/etc/security/user`
- `/etc/security/limits`
- `/etc/security/group`
- `/etc/security/envron`

## สถานะออก

0 คำสั่งดำเนินการเสร็จด้วยผลสำเร็จ

>0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการแสดงรายการกลุ่มที่ใช้ได้ทั้งหมดในเครื่อง ให้ป้อนคำสั่ง `dispgid` ดังนี้:

```
dispgid
```

เอาต์พุตคล้ายกับตัวอย่างต่อไปนี้:

```
system
staff
bin
sys
adm
uucp
mail
security
cron
printq
audit
ecs
nobody
usr
perf
```

## ไฟล์

ไอเท็ม

`/usr/sbin/dispgid`

`/etc/group`

คำอธิบาย

มีคำสั่ง `dispgid`

มีข้อมูลกลุ่ม

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dispgid`”

---

## คำสั่ง `dispgid`

### วัตถุประสงค์

แสดงรายการของชื่อผู้ใช้ทั้งหมดที่ใช้ได้

### ไวยากรณ์

`dispgid`

## คำอธิบาย

คำสั่งนี้สามารถใช้เพื่อแสดงรายการของชื่อผู้ใช้ทั้งหมดบนระบบ (หนึ่งบรรทัดต่อชื่อ) ไม่มีพารามิเตอร์สำหรับคำสั่งนี้ ไฟล์ต่อไปนี้ถูกเข้าถึงในโหมดอ่านอย่างเดียวเพื่อเรียกข้อมูลผู้ใช้:

- /etc/passwd
- /etc/security/user
- /etc/security/user.roles
- /etc/security/limits
- /etc/security/environ
- /etc/group
- /etc/group

## สถานะออก

- 0 คำสั่งดำเนินการเสร็จด้วยผลสำเร็จ  
>0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการแสดงรายการของผู้ใช้ที่ใช้ได้ในเครื่องของคุณให้ป้อนคำสั่ง dispuid ดังนี้:

```
dispuid
```

เอาต์พุตคล้ายกับตัวอย่างต่อไปนี้:

```
root
daemon
bin
sys
adm
uucp
guest
nobody
lpd
invscout
imnadm
user1
```

## ไฟล์

ไอเท็ม

/usr/sbin/dispuid  
/etc/passwd

คำอธิบาย

มีคำสั่ง dispuid  
มีข้อมูลรหัสผ่าน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dispuid” ในหน้า 171

---

## คำสั่ง dist

### วัตถุประสงค์

จัดสรรข้อความใหม่ไปที่แอดเดรสเพิ่มเติม

### ไวยากรณ์

```
dist [ +Folder ] [ -nodraftfolder | -draftfolder +Folder ] [ Message | -draftmessage Message ] [
-annotate [ -inplace | -notinplace ] | -noannotate ] [ -form FormFile ] [ -editor Editor | -noedit ] [
-nowhatnowproc | -whatnowproc Program ]
```

### คำอธิบาย

คำสั่ง `dist` มีอินเตอร์เฟซสำหรับการจัดสรรข้อความที่มีอยู่ใหม่ ไปที่รายการของแอดเดรสใหม่ โดยดีฟอลต์คำสั่ง `dist` คัดลอกข้อความปัจจุบัน ในโฟลเดอร์ปัจจุบันไปที่ไฟล์ `UserMHDDirectory/draft` และเริ่มต้นเอดิเตอร์ เมื่อต้องการระบุข้อความในโฟลเดอร์ปัจจุบัน ที่ไม่ใช่ค่าดีฟอลต์ ให้ใช้พารามิเตอร์ `Message`

เมื่อเริ่มต้น เอดิเตอร์จะพร้อมดีให้คุณป้อนค่า สำหรับแต่ละฟิลด์ส่วนตัว คำสั่ง `dist` ใช้รูปแบบส่วนตัว ที่กำหนดในไฟล์ `UserMHDDirectory/distcomps` (ถ้าไฟล์นี้ไม่มีอยู่ ระบบจะใช้ไฟล์ `/etc/mh/distcomps`) เนื่องจากเนื้อหาของข้อความคือข้อความที่คุณส่งออกไปใหม่ อย่าป้อนข้อมูลในเนื้อหาของข้อความ เมื่อต้องการกำหนดไฟล์รูปแบบ ที่ไม่ใช่ไฟล์ `UserMHDDirectory/distcomps` ให้ใช้แฟล็ก `-form`

เมื่อต้องการเปลี่ยนดีฟอลต์เอดิเตอร์ ให้ใช้แฟล็ก `-editor` หรือกำหนดรายการ `Editor:` ในไฟล์ `$HOME/.mh_profile` ของคุณ

กดคีย์ `Ctrl-D` เพื่อออกจาก เอดิเตอร์ เมื่อออกจาก เอดิเตอร์คำสั่ง `dist` เริ่มต้น Message Handler (MH) What Now? กด คีย์ `Enter` เพื่อดูรายการของคำสั่งย่อย `whatnow` ที่มี คำสั่งย่อยเหล่านี้ทำให้คุณสามารถดำเนินการแก้ไขส่วนตัวข้อความ แสดงส่วนตัวข้อความ กำหนดการจัดการ ข้อความหรือจบการประมวลผลของคำสั่ง `dist`

หมายเหตุ: บรรทัดของเส้นประ หรือบรรทัดว่างต้องอยู่ ระหว่างส่วนตัวและเนื้อหาของข้อความสำหรับข้อความที่ ระบุเมื่อถูกส่ง

ข้อความที่ส่งออกไปประกอบด้วยส่วนตัวต้นฉบับ และเนื้อหาของข้อความแนบไปกับส่วนตัวใหม่ไฟล์ `draft` ที่คุณ แก้ไขโดยใช้คำสั่ง `dist` โดยมีเพียงฟิลด์ ส่วนหัวเท่านั้น สำเนาของข้อความต้นฉบับที่มีข้อความแบบร่างใหม่จะไม่ถูกเก็บ โดยอัตโนมัติ

หากต้องการเพิ่มความคิดเห็นให้กับข้อความต้นฉบับด้วยข้อมูลที่แจกจ่ายใหม่ ให้ใช้แฟล็ก `-annotate` แฟล็กนี้ผนวกเข้ากับข้อความต้นฉบับด้วยฟิลด์ `Resent:`, และ วันที่และเวลาปัจจุบัน

### แฟล็ก

ไอเท็ม  
-annotate

#### คำอธิบาย

เพิ่มความคิดเห็นให้กับข้อความที่กำลังถูกจัดสรรด้วยบรรทัด:

Resent: date  
Resent: address

เนื่องจากแฟล็ก -annotate จะไม่คงอยู่เมื่อมีการเรียกใช้คำสั่งหลายครั้ง การเพิ่มความคิดเห็นจะสมบูรณ์เฉพาะถ้าข้อความถูกส่งโดยตรงจากคำสั่ง dist แฟล็ก -inplace บังคับให้ การเพิ่มความคิดเห็นเสร็จสิ้นในจุดที่กำหนด เพื่อรักษาลิงก์ไปยังข้อความ ที่เพิ่มความคิดเห็นวางข้อความแบบร่างในโฟลเดอร์ที่ระบุเฉพาะ ถ้าแฟล็ก -draftfolder +Folder ตามด้วยตัวแปร Message จะเหมือนกับการใช้แฟล็ก -draftmessage ถ้าไม่ระบุ +Folder ข้อความแบบร่างจะถูกวางไว้ใน Current-Folder

-draftfolder +Folder

-draftmessage Message

ระบุข้อความแบบร่าง โดยตีฟอลด์ระบบสร้างข้อความแบบร่างใหม่ ในโฟลเดอร์ปัจจุบัน ข้อความแบบร่างกลายเป็นข้อความปัจจุบัน

-editor Editor

+Folder

-form FormFile

-help

ระบุเอดิเตอร์เริ่มต้นสำหรับการเตรียมข้อความสำหรับการกระจายข้อมูล ระบุโฟลเดอร์ที่มีข้อความที่จะจัดสรรใหม่ ถ้าไม่ระบุโฟลเดอร์จะใช้ Current-Folder กำหนดฟอร์มข้อความ คำสั่ง dist จะทำงาน กับแต่ละบรรทัดในฟอร์มไฟล์ที่ระบุ แสดงรายการไวยากรณ์คำสั่ง การสับเปลี่ยน (สลับ) ที่มีอยู่ และข้อมูลเวอร์ชัน

-inplace

หมายเหตุ: สำหรับ MH ชื่อของแฟล็กนี้ ต้องถูกสะกดคำ บังคับให้การเพิ่มความคิดเห็นที่ต้องทำในตำแหน่งที่เรียงลำดับเพื่อสแกนลิงก์ กับข้อความการเพิ่มความคิดเห็น

Message

ระบุข้อความที่จะจัดสรรใหม่ ใช้การอ้างอิงต่อไปนี้ เพื่อระบุข้อความ:

Number จำนวนของข้อความ

#### cur or . (จุด)

ข้อความ ปัจจุบัน นี้คือ คำตีฟอลด์

first ข้อความแรกในโฟลเดอร์

last ข้อความสุดท้ายในโฟลเดอร์

next ข้อความหลังจากข้อความปัจจุบัน

prev ข้อความที่นำหน้าข้อความปัจจุบัน  
ยับยั้งการเพิ่มความคิดเห็น แฟล็กนี้คือ คำตีฟอลด์

-noannotate

-nodraftfolder

-noedit

-notinplace

-nowhatnowproc

-whatnowproc Program

วางแบบร่างในไฟล์ UserMHDdirectory/draft

ไม่แสดงการแก้ไขเริ่มต้น

ป้องกันการเพิ่มความเห็นเห็นในตำแหน่ง แฟล็กนี้คือ คำตีฟอลด์

ยับยั้งการประมวลผลแบบโต้ตอบของคำสั่ง dist แฟล็ก -nowhatnowproc ป้องกันการแฟล็ก

สตาร์ทโปรแกรมที่ระบุเพื่อแนะนำคุณผ่านงาน การกระจายข้อมูล ถ้าคุณระบุคำสั่ง whatnow เป็นตัวแปร Program คำสั่ง dist สตาร์ทไพโรซีเดอร์ whatnow ภายในแทนโปรแกรม ที่มีชื่อไฟล์ whatnow

## รายการโปรไฟล์

รายการต่อไปนี้ถูกป้อนในไฟล์ UserMHDdirectory/.mh\_profile:

#### ไอเท็ม

Current-Folder:

Draft-Folder:

Editor:

fileproc:

Path:

whatnowproc:

#### คำอธิบาย

ตั้งค่าโฟลเดอร์ปัจจุบันสำหรับคำตีฟอลด์

ตั้งคำตีฟอลด์โฟลเดอร์สำหรับแบบร่าง

เซตตีฟอลด์เอดิเตอร์

ระบุโปรแกรมที่ใช้เพื่อรีไฟล์ข้อความ

ระบุไดเรกทอรี MH ของผู้ใช้

ระบุโปรแกรมที่ใช้เพื่อประมวลคำถาม What now?

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

### ตัวอย่าง

1. เมื่อต้องการกระจายข้อความปัจจุบันใหม่จากโพลเดอร์ปัจจุบันป้อน:

```
dist
```

ระบบพร้อมที่เพื่อขอค่าฟิลด์ส่วนตัว หลังจากป้อนค่าให้กดคีย์ Enter เมื่อต้องการข้ามรายการให้กดคีย์ Enter โดยไม่ต้องป้อนค่า คุณต้องกรอกข้อมูลในฟิลด์ Resent-to: หลังจากเสร็จสิ้นกับส่วนตัว อย่าแก้ไขเนื้อหาของข้อความ กดคีย์ Ctrl-D เพื่อออกจากเอดิเตอร์ พร้อมที่ระบบแสดง:

```
What now?
```

กดคีย์ Enter เพื่อดูรายการของอีพซันที่มี ถ้าคุณต้องการกระจายข้อความนี้ใหม่ให้ป้อน send. ข้อความของคุณถูกส่งออกไปใหม่ไปยังรายการแอดเดรสใหม่

2. เมื่อต้องการกระจายข้อความไปที่รายการแอดเดรสใหม่ เมื่อมีแบบร่างข้อความอยู่ให้ป้อน:

```
dist
```

ระบบตอบกลับด้วยข้อความที่คล้ายกับต่อไปนี้:

```
Draft "$HOME/Mail/draft" exists (43 bytes).  
Disposition? _
```

เมื่อต้องการกระจายแบบร่างนี้ให้ป้อน:

```
replace
```

ระบบพร้อมที่เพื่อขอค่าฟิลด์ส่วนตัว หลังจากป้อนค่าให้กดคีย์ Enter เมื่อต้องการข้ามรายการให้กดคีย์ Enter โดยไม่ต้องป้อนค่า คุณต้องกรอกข้อมูลในฟิลด์ Resent-to: หลังจากเสร็จสิ้นกับส่วนตัว อย่าแก้ไขเนื้อหาของข้อความ กดคีย์ Ctrl-D เพื่อออกจากเอดิเตอร์ พร้อมที่ระบบแสดง:

```
What now?
```

กดคีย์ Enter เพื่อดูรายการของอีพซันที่มี ถ้าคุณต้องการกระจายแบบร่างนี้ใหม่ให้ป้อน send. ข้อความของคุณถูกส่งออกไปใหม่ไปยังรายการแอดเดรสใหม่

3. เมื่อต้องการส่งข้อความ 15 จากโพลเดอร์ schedules ให้ป้อน:

```
dist +schedules 15
```

ระบบพร้อมที่เพื่อขอค่าฟิลด์ส่วนตัว หลังจากป้อนค่าให้กดคีย์ Enter เมื่อต้องการข้ามรายการให้กดคีย์ Enter โดยไม่ต้องป้อนค่า คุณต้องกรอกข้อมูลในฟิลด์ Resent-to: หลังจากเสร็จสิ้นกับส่วนตัว อย่าแก้ไขเนื้อหาของข้อความ กดคีย์ Ctrl-D เพื่อออกจากเอดิเตอร์ พร้อมที่ระบบแสดง:

```
What now?
```

กดคีย์ Enter เพื่อดูรายการของ อีพซันที่มี เมื่อต้องการแจกจ่าย ข้อความอีกครั้ง พิมพ์ send และกดปุ่ม Enter

## ไฟล์

### ไอเท็ม

/etc/mh/distcomps  
UserMHDirectory/distcomps  
UserMHDirectory/draft  
/usr/bin/dist

### คำอธิบาย

มีรูปแบบข้อความดีฟอลต์ระบบ  
มีรูปแบบข้อความดีฟอลต์ผู้ใช้  
มีไฟล์แบบร่างปัจจุบัน  
มีรูปแบบที่รันได้ของคำสั่ง dist

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง forw” ในหน้า 598

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง whatnow

คำสั่ง mh\_alias

แอ็พพลิเคชันเมลล์

---

## คำสั่ง dmpuncompress

### วัตถุประสงค์

เรียกคืนไฟล์บีบอัดดัมพ์

### ไวยากรณ์

```
/usr/bin/dmpuncompress [-f] [-p] [File]
```

### คำอธิบาย

คำสั่ง dmpuncompress เรียกคืนไฟล์ดัมพ์ต้นฉบับที่ถูก บีบอัดขณะเวลาดัมพ์

แต่ละไฟล์บีบอัดที่ระบุโดยพารามิเตอร์ *File* ถูกเอาออก และแทนที่โดยสำเนาส่วนเพิ่มเติม ไฟล์ที่คล้ายบีบอัดมีชื่อเหมือนกับเวอร์ชันบีบอัด แต่ไม่มีส่วนขยาย .BZ ถ้าผู้ใช้มีสิทธิแบบผู้ใช้ root ไฟล์ที่คล้ายบีบอัดจะยังคงรักษา เจ้าของ กลุ่ม โหมด และเวลาแก้ไข เหมือนกับไฟล์ต้นฉบับไว้ ถ้าผู้ใช้ไม่มีสิทธิ root, ไฟล์จะรักษา โหมดและเวลาแก้ไข เดิมไว้ แต่จะรับค่า เจ้าของและกลุ่มใหม่

### แฟล็ก

#### ไอเท็ม

-f *File*

-p *File*

#### คำอธิบาย

บังคับส่วนขยาย เขียนทับไฟล์ ถ้าไฟล์มีอยู่แล้ว ระบบไม่พร้อมต่อผู้ใช้ว่าไฟล์ที่มีอยู่จะถูกเขียนทับ ขนาดไฟล์อาจไม่ถูกต้องจริง  
สแกนไฟล์ .BZ ต้นฉบับและไฟล์ดัมพ์ ที่ไม่ได้บีบอัดไว้ การเขียนทับนี้ลบไฟล์ที่บีบอัด เมื่อการรู้คือไฟล์ดัมพ์ ต้นฉบับเสร็จสมบูรณ์ ถ้าการเรียกคืนไฟล์ดัมพ์ ต้นฉบับ ไม่สมบูรณ์ เนื่องจากข้อผิดพลาด อีพซันจะปิดใช้งานการลบ ดัมพ์บางส่วน

### สถานะออก

ไอเท็ม  
0  
>0

คำอธิบาย  
ดำเนินการเป็นผลสำเร็จ  
มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการยกเลิกการบีบอัดไฟล์ **dump.BZ** ให้ป้อน:

```
/usr/bin/dmpuncompress dump.BZ
```

ไฟล์ **dump.BZ** ถูกยกเลิกการบีบอัดและเปลี่ยนชื่อเป็น **dump**

2. เมื่อต้องการรักษาไฟล์ **dump.BZ** และไฟล์ดัมพ์ที่สร้างขึ้นใหม่ในระบบไฟล์หลังดำเนินการสมบูรณ์แล้ว ให้ป้อน:

```
/usr/bin/dmpuncompress -p dump.BZ
```

## Location

`/usr/bin/dmpuncompress`

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `savecore`

คำสั่ง `snap`

คำสั่ง `uncompress`

สิ่งอำนวยความสะดวกที่รวมอยู่ในระบบ

---

## คำสั่ง `dnssec-keygen`

### วัตถุประสงค์

เครื่องมือการสร้างคีย์ส่วนขยายการรักษาความปลอดภัยระบบชื่อโดเมน (DNSSEC)

### ไวยากรณ์

```
dnssec-keygen [-a algorithm] [-b keysize] [-n nametype] [-c class] [-e] [-f flag] [-g generator] [-h] [-k] [-p protocol] [-r randomdev] [-s strength] [-t type] [-v level] [name]
```

### คำอธิบาย

คำสั่ง `dnssec-keygen` สร้างคีย์สำหรับ DNSSEC (Secure DNS) และยังสามารถสร้างคีย์เพื่อใช้กับ Transaction Signatures (TSIG)

### แฟล็ก

## ไอเท็ม

-a *algorithm*

### คำอธิบาย

เลือกอัลกอริทึมการเข้ารหัส อัลกอริทึม มีค่าหนึ่งในค่าต่อไปนี้:

- RSAMD5
- DSA
- DH (Diffie–Hellman)
- HMAC-MD5

ค่าเหล่านี้เป็นแบบสนใจขนาดตัวพิมพ์

### Notes:

1. สำหรับ DNSSEC, RSASHA1 เป็นอัลกอริทึม mandatory-implement และต้องการ DSA สำหรับ TSIG, HMAC-MD5 เป็นสิ่งที่จำเป็น
  2. HMAC-MD5 และ DH จะตั้งค่าแฟล็ก -k โดยอัตโนมัติ  
ระบุจำนวนบิตในคีย์ อ็อพชัน ของขนาดคีย์ขึ้นกับอัลกอริทึมที่ใช้ คีย์ RSAMD5 และ RSASHA1 ต้องมีขนาด 512 – 4096 บิต คีย์ DH ต้องมีขนาด 128 – 4096 บิต คีย์ DSA ต้องมีขนาด 512 – 1024 บิตและเป็นผลคูณของ 64 คีย์ HMAC-MD5 ต้องเป็นขนาด 1 – 512 บิต
- b *keysize* ระบุชนิดเจ้าของของคีย์ ค่าของ *nametype* ต้องเป็น ZONE (สำหรับ คีย์โซน DNSSEC), HOST หรือ ENTITY (สำหรับ คีย์ที่เชื่อมโยงกับโฮสต์), USER (สำหรับ คีย์ที่เชื่อมโยงกับผู้ใช้) หรือ OTHER (DNSKEY) ค่าเหล่านี้เป็นแบบไม่สนใจขนาดตัวพิมพ์
- c *class* ระบุว่าเร็กคอร์ดของ Domain Name Server (DNS) ที่มีคีย์ต้องมีคลาสที่ระบุ ถ้าไม่ได้ระบุ, จะใช้คลาส IN
- e หากคุณสร้างคีย์ RSAMD5 หรือ RSASHA1 ให้ใช้การยกกำลังจำนวนมาก
- f *flag* เช็ตแฟล็กที่ระบุในฟิลด์ *flag* ของเร็กคอร์ด KEY หรือ DNSKEY แฟล็กที่รู้จักคือ KSK (Key Signing Key) DNSKEY
- g *generator* หากคุณกำลังสร้างคีย์ DH ให้ใช้ตัวสร้างนี้ ค่าที่ใช้ได้คือ 2 และ 5 ถ้าไม่มีการระบุ generator, ไพรม์ที่รู้จักจาก RFC 2539 ถูกใช้ ถ้าเป็นไปได้; มิฉะนั้นค่าดีฟอลต์ เป็น 2
- h พิมพ์ข้อมูลสรุปย่อ ของอ็อพชันและอาร์กิวเมนต์ ของคำสั่ง `dnssec-keygen`
- k สร้างเร็กคอร์ด KEY ที่ไม่ใช่เร็กคอร์ด DNSKEY
- p *protocol* เช็ตค่าโปรโตคอลสำหรับคีย์ที่สร้าง โปรโตคอลคือหมายเลข 0 – 255 ค่าดีฟอลต์คือ 3 (DNSSEC)
- r *randomdev* ระบุชื่อของอุปกรณ์แรนดอม ถ้า ระบบปฏิบัติการไม่มีไฟล์ `/dev/random` หรืออุปกรณ์ที่เทียบเคียงกัน ดีฟอลต์ชื่อของอุปกรณ์แรนดอมคือคีย์บอร์ด อินพุต อาร์กิวเมนต์ `randomdev` ระบุชื่อ ของอุปกรณ์อักขระหรือไฟล์ที่มีข้อมูลแรนดอมที่จะถูก ใช้แทนค่าดีฟอลต์ คีย์บอร์ดค่าพิเศษซ้ำๆ คีย์บอร์ดอินพุตต้องถูกใช้
- s *strength* ระบุค่าความ strength ของคีย์ อาร์กิวเมนต์ *strength* คือหมายเลข 0 – 15, และขณะนี้ ไม่มีวัตถุประสงค์ที่กำหนดไว้ใน DNSSEC
- t *type* บ่งชี้การใช้คีย์ *type* ต้องเป็น AUTHCONF, NOAUTHCONF, NOAUTH หรือ NOCONF ค่าใดค่าหนึ่ง ค่าดีฟอลต์คือ AUTHCONF AUTH อ้างถึง ความสามารถในการพิสูจน์ตัวตนข้อมูล และ CONF เป็นความสามารถในการเข้ารหัสข้อมูล ไม่มีคีย์ที่ถูกสร้างสำหรับอัลกอริทึมเหล่านี้ (DH, HMAC-MD5, HMAC-SHA1, HMAC-SHA224, HMAC-SHA256, HMAC-SHA384, HMAC-SHA512) ที่มีชนิดคีย์เป็น NOAUTHCONF
- v *level* เช็ตระดับการดีบั๊ก

## พารามิเตอร์

### ไอเท็ม

name

### คำอธิบาย

ชื่อของคีย์ที่ระบุไว้ใน บรรทัดรับคำสั่ง สำหรับคีย์ DNSSEC ชื่อนี้ต้องตรงกับชื่อของโซน ที่คีย์ถูกสร้างขึ้น

## คีย์ที่สร้าง

เมื่อคำสั่ง `dnssec-keygen` เสร็จ สมบูรณ์ จะพิมพ์สตริงฟอร์ม `Knnnnn.+aaa+iiii` ไปที่เอาต์พุตมาตรฐาน ซึ่งเป็นสตริงการระบุสำหรับคีย์ที่สร้างขึ้น

- nnnn เป็นชื่อคีย์
- aaa เป็นตัวเลขแสดงถึง อัลกอริทึม
- iiii เป็นคีย์ identifier (หรือ ฟุตพริ้นต์)

คำสั่ง `dnssec-keygen` สร้างไฟล์สองไฟล์ โดยมีชื่อจากสตริงที่พิมพ์: `Knnnn.+aaa+iiii.key` มี คีย์พับลิค และ `Knnnn.+aaa+iiii.private` มี คีย์ไพรเวต

ไฟล์ `.key` มีเร็กคอร์ด DNSKEY ที่ สามารถถูกแทรกในโซนไฟล์ (โดยตรงหรือด้วยประโยค `$INCLUDE`) ไฟล์ `.private` มีฟิลด์จำเพาะ อัลกอริทึม สำหรับเหตุผลด้านความปลอดภัย ไฟล์นี้ไม่มีสิทธิ การอ่านทั่วไป ทั้งไฟล์ `.key` และไฟล์ `.private` ถูกสร้างขึ้นสำหรับอัลกอริทึมการเข้ารหัสแบบสมมาตร เช่น HMAC-MD5 แม้ว่าพับลิคคีย์และไพรเวตคีย์จะเท่ากัน

## ตัวอย่าง

เมื่อต้องการสร้าง 768 - บิต DSA คีย์สำหรับโดเมน `example.com`, พิมพ์คำสั่งต่อไปนี้:

```
dnssec-keygen -a DSA -b 768 -n ZONE example.com
```

คำสั่ง พิมพ์สตริงของฟอร์ม:

```
Kexample.com.+003+26160
```

ใน ตัวอย่างนี้, `dnssec-keygen` สร้างไฟล์ `Kexample.com.+003+26160.key` และ `Kexample.com.+003+26160.private`

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dig`” ในหน้า 157

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `named9`

คำสั่ง `nsupdate9`

คำสั่ง `rndc-confgen`

---

## คำสั่ง `dnssec-makekeyset`

### วัตถุประสงค์

เครื่องมือการลงชื่อโซนส่วนขยายการรักษาความปลอดภัยระบบชื่อโดเมน (DNSSEC)

### ไวยากรณ์

```
dnssec-makekeyset [-a] [-s start-time] [-e end-time] [-h] [-p] [-r randomdev] [-t ttl] [-v level] {key...}
```

### คำอธิบาย

คำสั่ง `dnssec-makekeyset` จะสร้างชุดคีย์จากคีย์หนึ่งคีย์หรือมากกว่าที่สร้างโดยคำสั่ง `dnssec-keygen` ซึ่งจะสร้างไฟล์ที่มีเร็กคอร์ด KEY สำหรับแต่ละคีย์ และชุดคีย์ self-signs ที่มี แต่ละโซนคีย์ไฟล์เอาต์พุตจะมีรูปแบบ `keyset-nnnn`. โดยที่ `nnnn` เป็นชื่อโซน

### แฟล็ก

ไอเท็ม	คำอธิบาย
-a	ตรวจสอบลายเซ็นที่สร้างทั้งหมด
-s <i>start-time</i>	ระบุวันที่และเวลาที่เรีกคอร์ด SIG ที่สร้างขึ้น จะใช้ได้ ซึ่งสามารถเป็นเวลาสัมบูรณ์หรือสัมพัทธ์ เวลาเริ่มต้น สัมบูรณ์ระบุได้โดยตัวเลขในรูปแบบ YYYYMMDDHHMMSS; 20000530144500 denotes 14:45:00 UTC on May 30th, 2000 เวลาเริ่มต้นสัมพัทธ์ ถูกระบุโดย +N ซึ่งเป็น N วินาที จากเวลาปัจจุบัน ถ้าไม่มีการระบุ <i>start-time</i> จะใช้เวลาปัจจุบัน
-e <i>end-time</i>	ระบุวันที่และเวลาที่เรีกคอร์ด SIG ที่สร้างขึ้น หมดอายุ โดยการใช้ค่า <i>start-time</i> เวลาสัมบูรณ์จะถูกระบุในรูปแบบ YYYYMMDDHHMMSS เวลาที่สัมพันธ์กับเวลาเริ่มต้นถูกระบุด้วย +N ซึ่งเป็น N วินาทีจากเวลาเริ่มต้น เวลาที่สัมพันธ์กับ เวลาปัจจุบันถูกระบุด้วย now+N หากไม่ได้รับ <i>end-time</i> จะใช้เวลา 30 วันจาก เวลาเริ่มต้นโดยดีฟอลต์
-h	พิมพ์ข้อมูลสรุปย่อ ของอ็อพชันและอาร์กิวเมนต์ ของคำสั่ง <code>dnssec-makekeyset</code>
-p	ใช้ข้อมูล pseudo-random เมื่อคุณลงชื่อโซน ซึ่งเร็วกว่า แต่ปลอดภัยน้อยกว่าการใช้ข้อมูลแบบสุ่มจริง อ็อพชันอาจมีประโยชน์เมื่อคุณลงชื่อโซนขนาดใหญ่หรือเมื่อ entropy source ถูกจำกัด
-r <i>randomdev</i>	ระบุซอร์สของการแรนดอม ถ้า ระบบปฏิบัติการไม่มี /dev/random หรืออุปกรณ์ที่เทียบเคียงกัน ดีฟอลต์ซอร์สของการแรนดอมคือคีย์บอร์ดอินพุต <i>randomdev</i> ระบุชื่อของอุปกรณ์หรือไฟล์อักขระที่มีข้อมูลแบบสุ่มที่ใช้แทน ค่าดีฟอลต์ คีย์บอร์ดคีย์พิเศษชื่อว่า คีย์บอร์ดอินพุตต้องถูกใช้
-t <i>tll</i>	ระบุ TTL (time to live) ของเรีกคอร์ด KEY และ SIG ค่าดีฟอลต์คือ 3600 วินาที
-v <i>level</i>	เซตระดับการดีบั๊ก

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
key	รายการของคีย์ที่จะถูกรวมใน ไฟล์ชุดคีย์ คีย์เหล่านี้จะถูกแสดงในรูปแบบ Knnnn.+aaa+iiii ตามที่สร้าง โดยคำสั่ง <code>dnssec-keygen</code>

## ตัวอย่าง

คำสั่งต่อไปนี้ จะสร้าง ชุดคีย์ที่มีคีย์ DSA สำหรับ example.com ที่สร้างขึ้นใน เพจ `dnssec-keygen man`

```
dnssec-makekeyset -t 86400 -s 20000701120000 -e +2592000 Kexample.com.+003+26160
```

ในตัวอย่างนี้ คำสั่ง `dnssec-makekeyset` จะสร้าง ไฟล์ `keyset-example.com`. ไฟล์นี้มีคีย์ที่ระบุ และลายเซ็นที่สร้างด้วยตัวเอง ผู้ดูแลระบบ DNS สำหรับ example.com สามารถส่ง `keyset-example.com`. ไปยัง ผู้ดูแลระบบ DNS สำหรับ .com สำหรับการลงชื่อหากโซน .com เป็นแบบ DNSSEC-aware และผู้ดูแลระบบของสองโซนมีกลไกบางอย่าง สำหรับการพิสูจน์ตัวตนกัน และกันและการแลกเปลี่ยนคีย์และลายเซ็นตัวอย่างปลอดภัย

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dnssec-keygen`” ในหน้า 178

“คำสั่ง `dnssec-signkey`”

## คำสั่ง `dnssec-signkey`

### วัตถุประสงค์

เครื่องมือการลงชื่อชุดคีย์ส่วนขยายการรักษาความปลอดภัยระบบชื่อโดเมน (DNSSEC)

### ไวยากรณ์

```
dnssec-signkey [-a] [-c class] [-s start-time] [-e end-time] [-h] [-p] [-r randomdev] [-v level] keyset key
```

## คำอธิบาย

คำสั่ง `dnssec-signkey` จะลงชื่อชุดคีย์ โดยทั่วไปชุดคีย์ใช้สำหรับโซนชายน์ และสร้างขึ้นโดย คำสั่ง `dnssec-makekeyset` ชุดคีย์ของโซนชายน์ ถูกลงชื่อด้วยคีย์โซนสำหรับโซนพารেন্ট ไฟล์เอาต์พุตจะอยู่ในรูปแบบ `signedkey-nnnn`. โดยที่ `nnnn` เป็นชื่อโซน

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a	ตรวจสอบลายเซ็นที่สร้างทั้งหมด
-c class	ระบุดคลาส DNS ของชุดคีย์
-s start-time	ระบุวันที่และเวลาที่เรีกคอร์ด SIG ที่สร้าง จะใช้ได้ ซึ่งสามารถเป็นเวลาสัมบูรณ์หรือ สัมพัทธ์ เวลาเริ่มต้น สัมบูรณ์ระบุได้โดยตัวเลขในรูปแบบ YYYYMMDDHHMMSS; 20000530144500 denotes 14:45:00 UTC on May 30th, 2000 เวลาเริ่มต้นแบบสัมพัทธ์ ถูกระบุด้วย +N ซึ่งเป็น N วันที่ จากเวลาปัจจุบัน ถ้าไม่มีการระบุ start-time จะใช้เวลาปัจจุบัน
-e end-time	ระบุวันที่และเวลาที่เรีกคอร์ด SIG ที่สร้าง หหมดอายุ โดยมี start-time, เวลาสัมบูรณ์ถูกบ่งชี้ในรูปแบบ YYYYMMDDHHMMSS เวลาที่สัมพันธ์กับเวลาเริ่มต้น ถูกระบุด้วย +N ซึ่งเป็น N วันที่ จากเวลาเริ่มต้น เวลาที่สัมพันธ์กับ เวลาปัจจุบันถูกระบุด้วย now+N หากไม่ได้ระบุ end-time จะใช้เวลา 30 วันจาก เวลาเริ่มต้นโดยดีฟอลต์
-h	พิมพ์การสรุปแบบย่อของอ็อปชันและอาร์กิวเมนต์ สำหรับคำสั่ง <code>dnssec-signkey</code>
-p	ใช้ข้อมูล pseudo-random เมื่อคุณลงชื่อโซน ซึ่งเร็วกว่า แต่ปลอดภัยน้อยกว่าการใช้ข้อมูลแบบสุ่มจริง อ็อปชันอาจมีประโยชน์เมื่อคุณลงชื่อโซนขนาดใหญ่หรือเมื่อ entropy source ถูกจำกัด
-r randomdev	ระบุซอร์สของการแรนดอม ถ้า ระบบปฏิบัติการไม่มี /dev/random หรืออุปกรณ์ที่เทียบเคียงกัน ดีฟอลต์ซอร์สของการแรนดอมคือคีย์บอร์ดอินพุต randomdev ระบุ ชื่อของอุปกรณ์หรือไฟล์อักขระที่มีข้อมูลแบบสุ่มที่ใช้แทน ค่าดีฟอลต์ คีย์บอร์ดค่าพิเศษชื่อว่า คีย์บอร์ดอินพุตต้องถูกใช้
-v level	เซตระดับการติบัก

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
keyset	ไฟล์ที่มีชุดคีย์ของชายน์
key	คีย์ที่ใช้เพื่อลงชื่อชุดคีย์ของ ชายน์

## ตัวอย่าง

ผู้ดูแลระบบ DNS สำหรับโซน DNSSEC-aware .com ใช้คำสั่งต่อไปนี้เพื่อลงชื่อไฟล์ชุดคีย์สำหรับ example.com ที่สร้างโดย คำสั่ง `dnssec-makekeyset` ที่มีคีย์ที่สร้างโดย คำสั่ง `dnssec-keygen` :

```
dnssec-signkey keyset-example.com. Kcom.+003+51944
```

ในตัวอย่างนี้ `dnssec-signkey` สร้างไฟล์ `signedkey-example.com.`, ซึ่งมีคีย์ `example.com` และลายเซ็นโดยคีย์ `.com` สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dnssec-keygen`” ในหน้า 178

“คำสั่ง `dnssec-makekeyset`” ในหน้า 180

“คำสั่ง `dnssec-signzone`” ในหน้า 183

# คำสั่ง `dnssec-signzone`

## วัตถุประสงค์

เครื่องมือการลงชื่อโซนส่วนขยายการรักษาความปลอดภัยระบบชื่อโดเมน (DNSSEC)

## ไวยากรณ์

```
dnssec-signzone [-a] [-c class] [-d directory] [-e end-time] [-f output-file] [-g] [-h] [-k key] [-l domain] [-i interval] [-I input-format] [-j jitter] [-N soa-serial-format] [-o origin] [-O output-format] [-p] [-r randomdev] [-s start-time] [-t] [-v level] [-z zonefile [key...]]
```

## คำอธิบาย

คำสั่ง `dnssec-signzone` ลงชื่อโซน โดยสร้างเร็กคอร์ด NSEC และ RRSIG และสร้างเวอร์ชันที่ลงชื่อ ของโซน การมีหรือไม่มีไฟล์ชุดคีย์สำหรับ แต่ละโซนชายนจะกำหนดสถานะความปลอดภัยของการมอบหมายจาก โซนที่ลงชื่อ (นั่นคือ จะกำหนดว่าโซนชายนปลอดภัยหรือไม่)

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a	ตรวจสอบลายเซ็นที่สร้างทั้งหมด
-c class	ระบุคลาส DNS ของโซน
-d directory	ค้นหาไฟล์ชุดคีย์ในไดเรกทอรีที่ระบุโดย อาร์กิวเมนต์ <code>directory</code>
-k key	ปฏิบัติต่อคีย์ที่ระบุเป็นคีย์ key-signing โดยละเว้นแฟล็กคีย์ทั้งหมด คุณสามารถระบุชื่อคีย์นี้ได้หลายครั้ง
-l domain	สร้างชุด DLV เพิ่มจาก key (DNSKEY) และชุด DS โดเมนถูกผนวกกับชื่อของเร็กคอร์ด
-g	สร้างเร็กคอร์ด DS สำหรับโซนชายนจาก ไฟล์ชุดคีย์ แฟล็กนี้ลบเร็กคอร์ด DS ที่มีอยู่
-s start-time	ระบุวันที่และเวลาที่เร็กคอร์ด RRSIG ที่สร้าง จะใช้ได้ ซึ่งสามารถเป็นเวลาสัมพันธ์หรือสัมพันธ์ เวลาเริ่มต้นสัมพันธ์ระบุได้โดยตัวเลขในรูปแบบ YYYYMMDDHHMMSS; 20000530144500 denotes 14:45:00 UTC on May 30th, 2000 เวลาเริ่มต้นสัมพันธ์สัมพันธ์ ถูกระบุด้วย +N ซึ่งเป็น N วินาทีจากเวลาปัจจุบัน ถ้าคุณไม่ได้ระบุอาร์กิวเมนต์ <code>start-time</code> คำสั่งจะใช้เวลาปัจจุบันลบด้วย 1 ชั่วโมง (เพื่อสำหรับเวลา ไม่เที่ยงตรง)
-e end-time	ระบุวันที่และเวลาที่เร็กคอร์ด RRSIG ที่สร้าง หมดอายุ ด้วยอาร์กิวเมนต์ <code>start-time</code> , เวลาสัมพันธ์สัมพันธ์ถูกแสดง ในรูปแบบ YYYYMMDDHHMMSS เวลาที่สัมพันธ์สัมพันธ์กับเวลาเริ่มต้น ถูกระบุด้วย +N ซึ่งเป็น N วินาทีจากเวลาเริ่มต้น เวลาที่สัมพันธ์สัมพันธ์กับ เวลาปัจจุบันถูกระบุด้วย now+N หากคุณไม่ได้ระบุอาร์กิวเมนต์ <code>end-time</code> คำสั่งจะใช้ 30 วันจากเวลาเริ่มต้นเป็นค่าดีฟอลต์
-f output-file	ระบุชื่อของไฟล์เอาต์พุตที่มี โซนที่ลงชื่อ ค่าดีฟอลต์คือ <code>signed</code> กับชื่ออินพุตไฟล์
-h	พิมพ์ข้อมูลสรุปย่อ ของอ็อปชันและอาร์กิวเมนต์ ของคำสั่ง <code>dnssec-signzone</code>
-i interval	เมื่อโซนที่ลงชื่อก่อนหน้าถูกผ่านเป็นอินพุต เร็กคอร์ดอาจถูกลงชื่ออีกครั้ง อ็อปชัน <code>interval</code> ระบุระยะเวลารอบ เป็นออฟเซตจากเวลาปัจจุบัน (เป็นวินาที) ถ้าเร็กคอร์ด RRSIG หมดอายุหลังจากระยะเวลารอบ, เร็กคอร์ดจะถูกคงไว้ ไม่เช่นนั้น จะถูกพิจารณา ว่าจะหมดอายุเร็ว ๆ นี้ และจะถูกแทนที่ ระยะเวลารอบดีฟอลต์ คือหนึ่งในสี่ของผลต่างระหว่าง signature end และ start times ถ้าคุณไม่ได้ระบุอาร์กิวเมนต์ <code>end-time</code> หรืออาร์กิวเมนต์ <code>start-time</code> , คำสั่ง <code>dnssec-signzone</code> สร้างลายเซ็นที่ใช้ได้ 30 วัน โดยมีระยะเวลารอบ เป็น 7.5 วัน ดังนั้น ถ้ามีเร็กคอร์ด RRSIG ถึงกำหนดหมดอายุ ในเวลาน้อยกว่า 7.5 วัน เร็กคอร์ดจะถูกแทนที่
-I input-format	ระบุรูปแบบของอินพุตโซนไฟล์ รูปแบบที่เป็นไปได้คือ text (ดีฟอลต์) และ raw
-j jitter	เมื่อคุณลงชื่อโซนด้วยลายเซ็นที่คงที่ตลอดอายุใช้งาน เร็กคอร์ด RRSIG ทั้งหมดที่ถูกสร้างเมื่อเวลาการลงชื่อจะหมดอายุพร้อมกัน ถ้าโซนถูกลงชื่อเพิ่มขึ้น ตัวอย่าง โซนที่ลงชื่อก่อนหน้า ถูกส่งเป็นอินพุตไปที่ตัวลงชื่อ และลายเซ็นที่หมดอายุทั้งหมดต้องถูก สร้างใหม่ในเวลาเดียวกัน อาร์กิวเมนต์ <code>jitter</code> ระบุหน้าต่าง <code>jitter</code> ที่ถูกใช้เพื่อแรนดอมเวลาหมดอายุ ของลายเซ็น ดังนั้นเป็นการกระจายการสร้างลายเซ็นใหม่เพิ่มขึ้น ตลอดเวลา ลายเซ็นตลอดอายุใช้งาน <code>jitter</code> ยังมีประโยชน์ต่อตัวตรวจสอบและเซิร์ฟเวอร์ โดยการกระจายการหมดอายุของแคช ตัวอย่างเช่น หาก RRSIGs จำนวนมากไม่หมดอายุพร้อมกันจากแคชทั้งหมด จะมีความคับคั่งน้อยกว่า หากตัวตรวจสอบทั้งหมดต้องถูกดึงใหม่เกือบจะในเวลาเดียวกัน
-n ncpu	ระบุจำนวนของ threads ที่จะใช้โดยดีฟอลต์ คำสั่งสตาร์ทหนึ่ง thread สำหรับแต่ละตัวประมวลผลที่พบ

ไอเท็ม	คำอธิบาย
-N soa-serial-format	ระบุรูปแบบหมายเลขลำดับ SOA ของ โซนที่ลงชื่อ อาร์กิวเมนต์ soa-serial-format สามารถเป็นค่าหนึ่งในค่าต่อไปนี้:  keep ไม่แก้ไขหมายเลขลำดับ SOA นี้เป็นค่าดีฟอลต์  increment เพิ่มหมายเลขลำดับ SOA โดยใช้การคำนวณ RFC 1982  unixtime ใช้หมายเลขลำดับ SOA เป็นจำนวนวินาทีตั้งแต่ epoch
-o origin	ระบุแหล่งกำเนิดโซน ถ้าไม่ระบุ ชื่อของโซนไฟล์จะถือว่าเป็นแหล่งกำเนิด
-O output-format	ระบุรูปแบบของไฟล์เอาต์พุตที่มีโซนที่ลงชื่อ รูปแบบที่เป็นไปได้คือ text (ดีฟอลต์) และ raw
-p	ใช้ข้อมูล pseudo-random เมื่อคุณลงชื่อโซน ซึ่งเร็วกว่า แต่ปลอดภัยน้อยกว่าการใช้ข้อมูลแบบสุ่มจริง อีพซัน นี้มีประโยชน์เมื่อคุณลงชื่อโซนขนาดใหญ่ หรือเมื่อเอนโทรปีซอร์ส ถูกจำกัด
-r randomdev	ระบุซอร์สของการแรนดอม ถ้าระบบปฏิบัติการไม่มีไฟล์ /dev/random หรืออุปกรณ์ที่เทียบเคียงกัน ดีฟอลต์ซอร์สของการแรนดอมคือคีย์บอร์ด อินพุต randomdev ระบุ ชื่อของอุปกรณ์หรือไฟล์อักขระที่มีข้อมูลแบบสุ่มที่ใช้แทน ค่าดีฟอลต์คีย์บอร์ดค่าพิเศษว่า คีย์บอร์ดอินพุตต้องใช้
-t	พิมพ์สถิติเมื่อเสร็จสมบูรณ์
-V level	ใช้ระดับการดีบั๊ก
-Z	ข้ามแฟล็ก KSK บนคีย์เมื่อคุณกำหนด สิ่งที่จะลงชื่อ

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
zonefile	ไฟล์ที่มีโซนที่ต้องการลงชื่อ
key	คีย์ที่ใช้เพื่อลงชื่อชุดไฟล์ ถ้าไม่ระบุคีย์ ค่าดีฟอลต์คือโซนคีย์ที่มีไพรเวตคีย์ไฟล์ ในไดเรกทอรีปัจจุบัน

## ตัวอย่าง

ต่อไปนี้เป็นตัวอย่างของโซน example.com ที่มีคีย์ DSA ที่สร้างโดยคำสั่ง dnssec-keygen คีย์ของโซนต้องอยู่ในโซน หากมีไฟล์ชุดคีย์ที่เชื่อมโยงกับโซนนี้หรือโซนชายนี้อะไรๆ ไฟล์ต้องอยู่ใน ไดเรกทอรีปัจจุบัน example.com คุณสามารถใช้คำสั่งต่อไปนี้:

```
dnssec-signzone -o example.com db.example.com Kexample.com.+003+26160
```

ในตัวอย่างนี้คำสั่ง dnssec-signzone สร้างไฟล์ db.example.com.signed ไฟล์ นี้ต้องเป็นการอ้างอิงในคำสั่งโซนในไฟล์ named.conf

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง named9

คำสั่ง named-checkconf

คำสั่ง nsupdate9

คำสั่ง rndc-confgen

## คำสั่ง dodisk

### วัตถุประสงค์

เริ่มแอคเคาต์การใช้งานดิสก์

## ไวยากรณ์

```
/usr/sbin/acct/dodisk [-X] [-o ][ File ... ]
```

### คำอธิบาย

คำสั่ง **dodisk** เริ่มแอคเคาต์ **disk-usage** โดยเรียกคำสั่ง **diskusg** และคำสั่ง **acctdisk** ถ้าคุณระบุแฟล็ก **-o** ด้วยคำสั่ง **dodisk**, ละเอียดยกกว่าแต่ซ้ากว่า เวอร์ชันของแอคเคาต์ **disk** โดยไตรีกทอรี **login** ถูกเริ่มโดยใช้คำสั่ง **acctdusg** โดยปกติ **cron daemon** รันคำสั่ง **dodisk**

โดยดีฟอลต์คำสั่ง **dodisk** ทำแอคเคาต์ดิสก์เฉพาะในการกำหนดไฟล์ด้วย stanzas ในไฟล์ **/etc/filesystems** ที่มีแอตทริบิวต์ **account=true** ถ้าคุณระบุชื่อไฟล์ด้วยพารามิเตอร์ **File** การทำแอคเคาต์ดิสก์เสร็จสิ้นบนไฟล์เหล่านั้นเท่านั้น

ถ้าคุณไม่ระบุแฟล็ก **-o**, พารามิเตอร์ **File** ควรมีชื่อไฟล์พิเศษของระบบไฟล์ที่เมาท์ได้ ถ้าคุณระบุทั้งแฟล็ก **-o** และพารามิเตอร์ **File** ไฟล์ควรเป็น จุดเมาท์ของระบบไฟล์ที่เมาท์

หมายเหตุ: คุณไม่ควรแบ่งใช้ไฟล์ แอคเคาต์ระหว่างโหนดในสภาวะแวดล้อมแบบกระจาย โหนดแต่ละโหนด ควรมีสำนเนาไฟล์แอคเคาต์ที่แตกต่างกัน

### แฟล็ก

ไอเท็ม คำอธิบาย

**-o** เรียกคำสั่ง **acctdusg** แทนคำสั่ง **diskusg** ในการเริ่มการทำแอคเคาต์ดิสก์โดยล็อกอินไตรีกทอรี

**-X** ประมวลผลอักขระที่มีทั้งหมดของแต่ละชื่อผู้ใช้ แทนการตัดให้เหลือ 8 อักขระแรก

### ความปลอดภัย

การควบคุมสิทธิ์เข้าถึง: คำสั่งนี้ควรให้สิทธิ์ในการเรียกใช้งาน (x) สำหรับการเข้าถึงให้กับสมาชิกของกลุ่ม **adm**

### ตัวอย่าง

1. เมื่อต้องการเริ่มการทำแอคเคาต์ **disk-usage** อัตโนมัติ ให้เพิ่มค่าต่อไปนี้ให้กับไฟล์ **/var/spool/cron/crontabs/root** :  

```
0 2 * * 4 /usr/sbin/acct/dodisk
```

ตัวอย่างนี้แสดงคำสั่งที่ **cron daemon** จะอ่านและทำงาน คำสั่ง **dodisk** จะรันในเวลา 2 a.m. (0 2) ทุกวันพฤหัสบดี (4) คำสั่งนี้คือหนึ่งในคำสั่งแอคเคาต์ปกติที่กำหนดไว้ให้กับ **cron daemon** ดูที่ "การตั้งค่า ระบบแอคเคาต์" ใน *Operating system and device management* สำหรับข้อมูลเพิ่มเติมเกี่ยวกับรายการแอคเคาต์ **cron** ทั่วไป

2. เมื่อต้องการรันแอคเคาต์ **disk-usage** บนระบบที่มีชื่อผู้ใช้มากกว่า 8 อักขระ ให้เพิ่มบรรทัดต่อไปนี้ให้กับไฟล์ **/var/spool/cron/crontabs/root** :

```
0 2 * * 4 /usr/sbin/acct/dodisk -X
```

### ไฟล์

ไอเท็ม  
/usr/sbin/acct  
/etc/filesystems

คำอธิบาย  
พาธไปยังคำสั่งแอคเคาต์  
มีข้อมูลเกี่ยวกับระบบไฟล์

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง diskusg” ในหน้า 169

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง acctdisk

คำสั่ง cron

การจัดทำแอคเคาต์ระบบ

---

## คำสั่ง domainname

### วัตถุประสงค์

แสดงหรือเซตชื่อของโดเมน Network Information Service (NIS) ปัจจุบัน

### ไวยากรณ์

`/usr/bin/domainname [ DomainName ]`

### คำอธิบาย

คำสั่ง `domainname` แสดง หรือเซตชื่อของโดเมน NIS ปัจจุบัน ถ้าคุณไม่ระบุ พารามิเตอร์ คำสั่ง `domainname` แสดงชื่อ ของโดเมน NIS ปัจจุบัน โดยปกติโดเมนรวมกลุ่ม ของโฮสต์ภายใต้การดูแลเหมือนกัน

ผู้ใช้ `root` เท่านั้นที่สามารถเซตชื่อของโดเมนโดย กำหนดดออาร์กิวเมนต์ให้กับคำสั่ง `domainname`

### ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

### ตัวอย่าง

1. เมื่อต้องการเข้าร่วมโดเมนใหม่ให้ป้อน:

```
domainname caesar
```

ในตัวอย่างนี้คำสั่ง `domainname` เซตชื่อโดเมน NIS เป็น `caesar`

2. เมื่อต้องการค้นหาชื่อของโดเมนที่เครื่องของคุณ เป็นสมาชิก ให้ป้อน:

```
domainname
```

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง ypinit

คำสั่ง ypserv

ภาพรวม Network Information Service (NIS) สำหรับการจัดการระบบ

---

## คำสั่ง domlist

### วัตถุประสงค์

แสดงข้อมูลโดเมนสำหรับผู้ใช้หรือกระบวนการ

### ไวยากรณ์

`domlist -p pid`

### คำอธิบาย

คำสั่ง `domlist` จัดเตรียมข้อมูลโดเมนให้กับผู้เรียกเกี่ยวกับโดเมนที่กำหนดไว้ในปัจจุบัน หากระบุแฟล็ก หรืออาร์กิวเมนต์ไว้ คำสั่ง `domlist` จะแสดงรายการของโดเมนที่กำหนดให้กับผู้เรียกด้วยคำอธิบายข้อความ ของแต่ละโดเมนหากจัดเตรียมไว้ในฐานข้อมูลโดเมน

คำสั่ง `domlist` ยังยอมให้ผู้ใช้ที่มีสิทธิใช้งาน เพื่อแสดงข้อมูลโดเมนสำหรับกระบวนการ การระบุ ID กระบวนการ ด้วยแฟล็ก `-p` ยอมให้ผู้ใช้ที่มีสิทธิใช้งาน แสดงโดเมนที่เชื่อมโยงกับกระบวนการ

### แฟล็ก

ไอเท็ม  
`-pPID`

คำอธิบาย  
แสดงข้อมูลโดเมนของกระบวนการที่ระบุเฉพาะ

### ตัวอย่าง

1. หากต้องการแสดงรายการของโดเมนที่กำหนดให้กับคุณและ คำอธิบายข้อความ ให้ใช้คำสั่งต่อไปนี้:

```
domlist
```

2. หากต้องการแสดงรายการของโดเมนที่กำหนดให้กับกระบวนการ ให้ใช้คำสั่งต่อไปนี้:

```
domlist -p <pid>
```

### ไฟล์ที่เข้าถึง

ไอเท็ม  
ไฟล์  
`/etc/security/domains`

คำอธิบาย  
Mode  
r

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `mkdom`

คำสั่ง `lsdom`

คำสั่ง `rmdom`

---

## คำสั่ง dosdel

### วัตถุประสงค์

ลบไฟล์ DOS

### ไวยากรณ์

```
dosdel [-v ][ -D Device ] File ...
```

### คำอธิบาย

คำสั่ง `dosdel` ลบไฟล์ DOS ที่ระบุโดยพารามิเตอร์ *File* ใช้แฟล็ก `-v` เพื่อรับข้อมูลรูปแบบ เกี่ยวกับดิสก์

DOS ระเบียบ file-naming ถูกใช้โดย มีหนึ่งข้อยกเว้น เนื่องจากอักขระ \ (backslash) มีความหมายพิเศษได้ต่อ ระบบปฏิบัติการ ให้ใช้อักขระ / (slash) เป็นตัวคั่นเพื่อระบุชื่อไดเรกทอรีย่อยในชื่อพาธ DOS คำสั่ง `dosdel` แปลงอักขระตัวพิมพ์เล็กในชื่อไฟล์หรือไดเรกทอรีเป็นตัวพิมพ์ใหญ่ก่อนทำการตรวจสอบดิสก์ เนื่องจากชื่อไฟล์ทั้งหมดถือว่าเป็นชื่อพาธแบบเต็ม (ไม่ใช่แบบสัมพัทธ์) คุณไม่จำเป็นต้องเพิ่ม / (เครื่องหมายทับ) เริ่มต้น

### แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-D Device</code>	ระบุชื่อของอุปกรณ์ DOS เป็น <code>/dev/fd0</code> หรือ <code>/dev/fd1</code> อุปกรณ์ดีฟอลต์คือ <code>/dev/fd0</code>
<code>-v</code>	เขียนข้อมูลไปที่เอาต์พุตมาตรฐานเกี่ยวกับรูปแบบของดิสก์ ใช้แฟล็กนี้เพื่อตรวจสอบว่าอุปกรณ์ดิสก์ DOS

### ตัวอย่าง

เมื่อต้องการลบไฟล์ DOS บนอุปกรณ์ดีฟอลต์ให้ป้อน:

```
dosdel file.ext
```

### ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/dosdel</code>	มีคำสั่ง <code>dosdel</code>

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dosdir`” ในหน้า 189

“คำสั่ง `dosformat`” ในหน้า 190

“คำสั่ง `dosread`” ในหน้า 192

“คำสั่ง `doswrite`” ในหน้า 194

---

# คำสั่ง dosdir

## วัตถุประสงค์

แสดงไดเรกทอรีสำหรับไฟล์ DOS

## ไวยากรณ์

```
dosdir [ -l [ -e ] ] [ -a ] [ -d ] [ -t ] [ -v ] [ -D Device ] [ File ... | Directory ... ]
```

## คำอธิบาย

คำสั่ง dosdir แสดงข้อมูลเกี่ยวกับ ไฟล์หรือไดเรกทอรี DOS ที่ระบุ ถ้าคุณระบุไดเรกทอรี โดยไม่ได้ระบุแฟล็ก -d ด้วย คำสั่ง dosdir จะแสดงข้อมูลเกี่ยวกับไฟล์ในไดเรกทอรีนั้น

DOS ระเบียบ file-naming ถูกใช้โดย มีหนึ่งข้อยกเว้น เนื่องจากอักขระ \ (backslash) มีความหมายพิเศษได้ต่อ ระบบปฏิบัติการ การใช้อักขระ / (slash) เป็นตัวคั่นเพื่อระบุ ชื่อไดเรกทอรีย่อยในชื่อพาธ DOS คำสั่ง dosdir แปลงอักขระตัวพิมพ์เล็กในชื่อไฟล์หรือไดเรกทอรี เป็นตัวพิมพ์ใหญ่ก่อนทำการตรวจสอบดีสก์ เนื่องจากชื่อไฟล์ทั้งหมดถือว่าเป็นชื่อพาธแบบเต็ม (ไม่ใช่แบบสัมพัทธ์) คุณไม่จำเป็นต้องเพิ่ม / (เครื่องหมายทับ) เริ่มต้น

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a	เขียนข้อมูลเกี่ยวกับไฟล์ทั้งหมด ซึ่งรวมถึงไฟล์ซ่อนและไฟล์ระบบ รวมทั้ง . (จุด) และ .. (จุด-จุด)
-d	ปฏิบัติต่อค่า File เป็นไฟล์ แม้ว่า ระบุเป็นไดเรกทอรี เมื่อมีการระบุไดเรกทอรีด้วยพารามิเตอร์ Directory ข้อมูลเกี่ยวกับตัวไดเรกทอรีเองถูกแสดง แทนข้อมูลเกี่ยวกับไฟล์ที่อยู่ภายใน
-D Device	ระบุชื่อของอุปกรณ์ DOS เป็น /dev/fd0 หรือ /dev/fd1 อุปกรณ์ดีฟอลต์คือ /dev/fd0
-e	ใช้แฟล็ก -l เพื่อเขียนรายการของคลัสเตอร์ที่จัดสรรให้กับไฟล์
-l	สร้างรายการของคลัสเตอร์ที่รวม วันที่สร้าง ขนาด เป็นไบต์ และแอดทริบิวต์ของไฟล์ ขนาดของไดเรกทอรีย่อยถูกระบุ เป็น 0 ไบต์ แอดทริบิวต์มีความหมายดังต่อไปนี้:  A (Archive) ไฟล์ไม่ได้ถูกลบข้อมูล ตั้งแต่ถูกแก้ไขครั้งสุดท้าย  D (Directory) ไฟล์เป็นไดเรกทอรีย่อยและไม่ถูกรวมไว้ในการค้นหาไดเรกทอรี DOS ปกติ  H (Hidden) ไฟล์ไม่ถูกรวมอยู่ในการค้นหาไดเรกทอรี DOS ปกติ  R (Read-only) ไม่สามารถแก้ไขไฟล์ได้  S (System) ไฟล์เป็นไฟล์ระบบและไม่ถูกรวมไว้ในการค้นหาไดเรกทอรี DOS ปกติ
-t	แสดงผังไดเรกทอรีทั้งหมด เริ่มที่ไดเรกทอรีตามชื่อที่กำหนด
-v	เขียนข้อมูลไปที่เอาต์พุตมาตรฐานเกี่ยวกับรูปแบบของดีสก์ ใช้แฟล็กนี้เพื่อตรวจสอบว่าอุปกรณ์ดีสก์ DOS

## ตัวอย่าง

เมื่อต้องการอ่านไดเรกทอรีของไฟล์ DOS บน /dev/fd0, ให้ป้อน:

```
dosdir
```

คำสั่งส่งคืนข้อมูลชื่อของไฟล์และพื้นที่ดิสก์

```
PG3-25.TXT  
PG4-25.TXT  
PG5-25.TXT  
PG6-25.TXT  
Free space: 312320 bytes
```

เมื่อต้องการอ่านไดเรกทอรีของไฟล์ DOS บน /dev/fd1, ให้ป้อน:

```
dosdir -D/dev/fd1
```

คำสั่งส่งคืนข้อมูลชื่อของไฟล์และพื้นที่ดิสก์

```
PG7-25.TXT  
PG8-25.TXT  
PG9-25.TXT  
PG10-25.TXT  
Free space: 312320 bytes
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/dosdir	มีคำสั่ง dosdir

สิ่งอ้างอิงที่เกี่ยวข้อง:

- “คำสั่ง dosdel” ในหน้า 188
- “คำสั่ง dosformat”
- “คำสั่ง dosread” ในหน้า 192
- “คำสั่ง doswrite” ในหน้า 194

---

## คำสั่ง dosformat

### วัตถุประสงค์

ฟอร์แมตดิสเก็ต DOS diskett

### ไวยากรณ์

```
dosformat[ -V Label ][-D Device | -4 ]
```

### คำอธิบาย

คำสั่ง dosformat ฟอร์แมต ดิสเก็ตด้วยฟอร์แมต DOS

อุปกรณ์ดีฟลอปต์และฟอร์แมตดิสเก็ตไดรฟ์ DOS คือ `/dev/fd0` สำหรับดิสเก็ต 3.5-นิ้ว ความจุโดยปกติจะเป็น 1.44M-byte หรือ 2.88M-byte, ขึ้นกับความจุที่ไดรฟ์สนับสนุน รูปแบบดิสเก็ตไดรฟ์ DOS อื่นถูกนำมาใช้โดยใช้แฟล็ก `-D` or `-4`

เมื่อต้องการรวมวลุ่มเลเบล ให้ใช้แฟล็ก `-V`

หมายเหตุ: วัตถุประสงค์ของคำสั่งนี้เพื่อช่วยอำนวยความสะดวกในการถ่ายโอนไฟล์ระหว่างระบบปฏิบัติการนี้และระบบ DOS การใช้คำสั่งเพื่อจัดรูปแบบดิสเก็ตที่จำเป็น ต้องมีไฟล์เริ่มระบบ DOS นั้นไม่แนะนำ

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-V</code>	เขียนพารามิเตอร์ <i>Label</i> ให้กับดิสเก็ต เป็นวลุ่มเลเบล DOS
<code>-DDevice</code>	ระบุชนิดและขนาดดิสเก็ตไดรฟ์ พารามิเตอร์ <i>Device</i> สามารถถูกระบุเป็น:  สำหรับไดรฟ์ 3.5-inch, 1.44M:  <code>/dev/fd0</code> 1.44MB (ดีฟลอปต์)  <code>/dev/fd0h</code> 1.44MB  <code>/dev/fd0l</code> 720KB  <code>/dev/fd0.18</code> 1.44MB  <code>/dev/fd0.9</code> 720KB  สำหรับไดรฟ์ 3.5-นิ้ว, 2.88M:  <code>/dev/fd0</code> 2.88MB (ดีฟลอปต์)  <code>/dev/fd0h</code> 2.88MB  <code>/dev/fd0l</code> 720KB  <code>/dev/fd0.36</code> 2.88MB  <code>/dev/fd0.18</code> 1.44MB  <code>/dev/fd0.9</code> 720KB  สำหรับไดรฟ์ 5.25-นิ้ว, 1.2M:  <code>/dev/fd0</code> 1.2MB (ดีฟลอปต์)  <code>/dev/fd0.15</code> 1.2MB  <code>/dev/fd0.9</code> 360KB

ไอเท็ม คำอธิบาย  
-4 ระบุความจุที่ต่ำกว่าสำหรับขนาดดิสเก็ต

## ตัวอย่าง

1. เมื่อต้องการฟอร์แมตดิสเก็ต 3.5-นิ้ว, 1.44M-ไบต์ที่มีวอลุ่มเลเบล "homework," ให้พิมพ์ดังนี้:

```
dosformat -V homework
```

2. เมื่อต้องการฟอร์แมตดิสเก็ต 5.25-นิ้ว, 360K-ไบต์ให้พิมพ์ ดังนี้:

```
dosformat -D /dev/fd1.9
```

OR

```
dosformat -D /dev/fd1 -4
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/dosformat	มีคำสั่ง dosformat

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dosdel” ในหน้า 188

“คำสั่ง dosdir” ในหน้า 189

“คำสั่ง dosread”

“คำสั่ง doswrite” ในหน้า 194

---

## คำสั่ง dosread

### วัตถุประสงค์

คัดลอกไฟล์ DOS

### ไวยากรณ์

```
dosread[ -a ][ -v ][ -D Device ]File1 [ File2 ]
```

### คำอธิบาย

คำสั่ง **dosread** ก๊อปปี้ไฟล์ DOS ที่ระบุโดยตัวแปร *File1* ไปที่เอาต์พุตมาตรฐาน หรือไปที่ไฟล์ที่ระบุโดยตัวแปร *File2* ถ้าไม่มีการระบุชื่อพารสำหรับตัวแปร *File2* ไฟล์ DOS จะถูกก๊อปปี้ไปที่ไดเร็กทอรี *root*

นอกจากจะมีการระบุ, คำสั่ง **dosread** ก๊อปปี้จำนวนไบต์ตามที่ระบุในรายการไดเร็กทอรี สำหรับไฟล์ที่ระบุโดยตัวแปร *File1* ซึ่งหมายความว่าโดยเฉพาะ คุณไม่สามารถก๊อปปี้ไดเร็กทอรีได้เนื่องจาก โดยระเบียบ ไดเร็กทอรีมีขนาดเรกคอร์ดเป็น 0

คุณสามารถใช้รูปแบบ DOS file-naming โดยมีหนึ่งข้อยกเว้น: \ (backslash) เนื่องจากอักขระ \ มีความหมายพิเศษได้ใน DOS, ใช้อักขระ / (slash) เป็นตัวคั่นเพื่อระบุชื่อไดเร็กทอรีย่อยในชื่อพาร DOS คำสั่ง **dosdir** แปลงอักขระตัวพิมพ์เล็กในชื่อ

ไฟล์หรือไดเรกทอรีเป็นตัวพิมพ์ใหญ่ก่อนทำการตรวจสอบดิสก์ เนื่องจากชื่อไฟล์ทั้งหมดถือว่าเป็นชื่อพาธแบบเต็ม (ไม่ใช่แบบสัมพัทธ์) คุณไม่จำเป็นต้องเพิ่ม / (เครื่องหมายทับ) เริ่มต้น

#### Notes:

1. คำสั่ง `dosread` ไม่แปล \* และ ? (เครื่องหมายดอกจันและเครื่องหมายคำถาม) อักขระ wildcard เนื่องจากมีความหมายพิเศษ ถ้าคุณไม่ระบุส่วนขยายชื่อไฟล์ ชื่อไฟล์จะถูกจับคู่เหมือนกับว่า คุณได้ระบุส่วนขยายว่าง
2. คุณไม่สามารถกำหนดชื่อของคำสั่งนี้เองได้ คำสั่งต้องมีชื่อ `dosread`
3. คำสั่ง `dosread` อ่านไฟล์จากดีฟอลต์ไดเรกทอรีที่มีดิสก์เกิด DOS จากนั้นคำสั่ง `dosread` ก็อปปีไฟล์ไปที่ไดเรกทอรีปัจจุบันเป็นไฟล์ที่ระบบปฏิบัติการนี้รู้จัก ถ้าดิสก์เกิด DOS มีไดเรกทอรีย่อยคำสั่ง `dosread` จะไม่สร้างไดเรกทอรีย่อยใหม่ที่ตรงกัน ในระบบปฏิบัติการนี้ คุณต้องสร้างไดเรกทอรีย่อยและระบุแต่ละไฟล์ DOS ที่คุณต้องการคัดลอกไปที่ไดเรกทอรีย่อยใหม่

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a	แทนที่แต่ละลำดับคีย์ CR-LF (ปัดแคร่, ขึ้นบรรทัดใหม่) ด้วย อักขระบรรทัดใหม่และแปลลำดับคีย์ Ctrl-Z (ASCII SUB) เป็น อักขระสิ้นสุดบรรทัด
-DDevice	ระบุชื่อของอุปกรณ์ DOS เป็น /dev/fd0 หรือ /dev/fd1 ค่าดีฟอลต์ของตัวแปร Device คือ /dev/fd0 อุปกรณ์นี้ต้องมีรูปแบบดิสก์ DOS
-v	เขียนข้อมูลไฟล์ไปที่เอาต์พุตมาตรฐานเกี่ยวกับรูปแบบของ ดิสก์ ใช้แฟล็กนี้เพื่อตรวจสอบว่าอุปกรณ์ดิสก์ DOS

## ตัวอย่าง

1. เมื่อต้องการคัดลอกไฟล์ข้อความจาก DOS, พิมพ์:

```
dosread -a chap1.doc chap1
```

ลำดับคำสั่งนี้คัดลอกไฟล์ข้อความ DOS \CHAP1.DOC บนอุปกรณ์ดีฟอลต์ /dev/fd0 ไปยัง chap1 ในไดเรกทอรี ปัจจุบัน

2. เมื่อต้องการคัดลอกไบนารีไฟล์จากดิสก์เกิด DOS พิมพ์:

```
dosread -D/dev/fd1 /survey/test.dta /home/fran/testdata
```

ลำดับคำสั่งนี้คัดลอกไฟล์ข้อมูล DOS \SURVEY\TEST.DTA บน /dev/fd1 ไปยัง /home/fran/testdata

3. เมื่อต้องการก็อปปีทุกไฟล์ DOS บนดิสก์เกิด พิมพ์:

```
dosdir | awk '!/There are/ {print $1}' | xargs -t -i dosread {} {}
```

คำสั่ง นี้ นำไฟล์ตามลำดับจากดีฟอลต์ไดเรกทอรีที่มีดิสก์ DOS แล้วก็อปปีไปที่ไดเรกทอรีปัจจุบัน

## ไฟล์

ไอเท็ม  
/usr/bin/dosread  
/dev/fd0

คำอธิบาย  
มีคำสั่ง dosread  
มีชื่ออุปกรณ์สำหรับดิสเก็ตไดรฟ์

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dosdel” ในหน้า 188

“คำสั่ง doswrite”

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง awk

ชนิดของไฟล์

---

## คำสั่ง doswrite

### วัตถุประสงค์

คัดลอกไฟล์ไปที่ไฟล์ DOS

### ไวยากรณ์

```
doswrite[ -a][ -v ][ -DDevice ]File1 File2
```

### คำอธิบาย

คำสั่ง doswrite คัดลอกไฟล์ที่ระบุโดยพารามิเตอร์ *File1* ไปที่ไฟล์ DOS ที่ระบุโดยพารามิเตอร์ *File2* คำสั่ง doswrite คัดลอกไฟล์ไปที่ดิสเก็ต DOS เดียว คำสั่ง doswrite ไม่สามารถคัดลอกไฟล์ข้ามดิสเก็ต DOS หลายดิสเก็ต

คำสั่ง doswrite เขียนไฟล์ที่ระบุโดยพารามิเตอร์ *File2* ไปที่อุปกรณ์ DOS โดยใช้ระเบียบการตั้งชื่อ DOS มาตรฐาน เนื่องจากอักขระ DOS\ (backslash) มีความหมายพิเศษได้สำหรับระบบปฏิบัติการ DOS, โปรดอย่าใช้\ (backslash) เมื่อระบุถึงชื่อไดเรกทอรีย่อยในพารามิเตอร์ *File2* ให้ใช้อักขระ / (slash) แทน

คำสั่ง doswrite แปลง อักขระตัวพิมพ์เล็กที่ระบุในพารามิเตอร์ *File1* ไปเป็นตัวพิมพ์ใหญ่ ก่อนทำการตรวจสอบอุปกรณ์ DOS เนื่องจากชื่อไฟล์ทั้งหมดถือว่าเป็นชื่อพาธแบบเต็ม (ไม่ใช่แบบสัมพัทธ์) คุณไม่จำเป็นต้องเพิ่ม / (slash) เริ่มต้น

ถ้าไฟล์ที่ระบุในพารามิเตอร์ *File2* มี / (slash), แต่ละคอมโพเนนต์แทรกเข้ามาต้อง มีอยู่เป็นไดเรกทอรีและคอมโพเนนต์สุดท้าย (ไฟล์ที่มีชื่อ) ต้องไม่มีอยู่ไฟล์ใดที่มีอยู่โดยมีชื่อเหมือนกันจะถูกเขียนทับ

### Notes:

1. อักขระ wildcard \* และ ? (เครื่องหมายดอกจัน และเครื่องหมายคำถาม) ไม่ถูกปฏิบัติเป็นพิเศษโดยคำสั่งนี้ (แม้ว่า จะมีความหมายกับเชลล์) ถ้าคุณไม่ระบุส่วนขยายชื่อไฟล์ ชื่อไฟล์จะถูกจับคู่เหมือนกับว่า คุณได้ระบุส่วนขยายว่าง
2. คำสั่งนี้ต้องมีชื่อ doswrite
3. ไดเรกทอรี DOS เก็บได้สูงสุด 244 ไฟล์

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a	แทนที่อักขระ NL (new-line) ด้วยลำดับ CR-LF (carriage return, line-feed) Ctrl-Z ถูกเพิ่มให้กับเอาต์พุตที่ท้ายไฟล์
-DDevice	ระบุชื่อของอุปกรณ์ DOS เป็น /dev/fd0 หรือ /dev/fd1 อุปกรณ์ที่พอลต์คือ /dev/fd0 อุปกรณ์นี้ต้องมีรูปแบบดิสก์ DOS
-v	เขียนข้อมูลไปที่เอาต์พุตมาตรฐานเกี่ยวกับรูปแบบของดิสก์ ใช้แฟล็กนี้เพื่อตรวจสอบว่าอุปกรณ์ดิสก์ DOS

## ตัวอย่าง

1. เมื่อต้องการคัดลอกไฟล์ข้อความไปที่ดิสก์เกิด DOS, ป้อน:

```
doswrite -a chap1 chap1.doc
```

คำสั่งนี้คัดลอกไฟล์ chap1 ในไดเรกทอรีปัจจุบันไปยังไฟล์ข้อความ DOS \CHAP1.DOC บนอุปกรณ์ดีพอลต์ /dev/fd0

2. เมื่อต้องการคัดลอกไบนารีไฟล์ไปที่ดิสก์เกิด DOS, ป้อน:

```
doswrite -D/dev/fd1 /home/fran/testdata /survey/test.dta
```

คำสั่งนี้คัดลอกไฟล์ /home/fran/testdata ไปยังไฟล์ DOS \SURVEY\TEST.DTA บน /dev/fd1

3. เมื่อต้องการคัดลอกทุกไฟล์ในไดเรกทอรีปัจจุบันไปที่ดิสก์เกิด DOS ในดีพอลต์ไดเรกทอรีของคุณ ให้ป้อน:

```
for i in *
do
doswrite $i $i
done
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/doswrite	มีคำสั่ง doswrite
/dev/fd0	มีชื่ออุปกรณ์สำหรับดิสก์เกิดไดเรก

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dosdir” ในหน้า 189

“คำสั่ง dosformat” ในหน้า 190

“คำสั่ง dosread” ในหน้า 192

---

## คำสั่ง dp

### วัตถุประสงค์

วิเคราะห์ค่าและจัดรูปแบบวันที่ใหม่

### ไวยากรณ์

```
dp[ -form File| -format String ][ -width Number ]Date
```

## คำอธิบาย

คำสั่ง **dp** วิเคราะห์ค่าและจัดรูปแบบวันที่ใหม่ คำสั่ง **dp** ไม่ได้ถูกเริ่มโดยผู้ใช้ คำสั่ง **dp** ถูกเรียกโดยโปรแกรมอื่น โดยปกติตามชื่อพารามิเตอร์, `/usr/lib/mh/dp`

คำสั่ง **dp** วิเคราะห์ค่าแต่ละสตริงส่วนหัว เมลที่ระบุเป็นวันที่และพยายามจัดรูปแบบสตริงใหม่ รูปแบบดีฟอลต์เอาต์พุตสำหรับคำสั่ง **dp** คือมาตรฐาน ARPA RFC 822 สำหรับแต่ละสตริงที่ไม่สามารถวิเคราะห์ได้ คำสั่ง **dp** จะแสดงข้อความแสดงความผิดพลาด

## พารามิเตอร์

ไอเท็ม คำอธิบาย  
Date ระบุวันที่ที่ต้องการวิเคราะห์

## แฟล็ก

ไอเท็ม คำอธิบาย  
-form *File* จัดรูปแบบใหม่ให้กับวันที่ที่ระบุในพารามิเตอร์ *Date* ให้เป็นรูปแบบทางเลือกที่อธิบายโดยตัวแปร *File*  
-format *String* จัดรูปแบบใหม่ให้กับวันที่ที่ระบุในพารามิเตอร์ *Date* ให้เป็นรูปแบบทางเลือกที่ระบุโดยตัวแปร *String* รูปแบบสตริงดีฟอลต์เป็นดังนี้:  

```
%<(nodate{text})error:%{text}%|%(putstr(pretty{text}))%>
```

  
-help แสดงรายการไวยากรณ์คำสั่ง การสับเปลี่ยน (สลับ) ที่มีอยู่ และข้อมูลเวอร์ชัน  
หมายเหตุ: สำหรับ Message Handler (MH), ชื่อของแฟล็กนี้ระบบเพิ่มเติม  
-width *Number* เซ็ตจำนวนสูงสุดของคอลัมน์ ที่คำสั่ง **dp** command ใช้เพื่อแสดงวันที่และข้อความแสดงความผิดพลาด ค่าดีฟอลต์คือความกว้างของจอแสดงผล

## ไฟล์

ไอเท็ม คำอธิบาย  
\$HOME/.mh\_profile มีโปรไฟล์ผู้ใช้ MH  
/etc/mh/mtstailor มินิยามคำสั่ง MH

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **ap**

แอ็พพลิเคชันเมลล์

---

## dpid2 Daemon

### วัตถุประสงค์

เริ่มต้น **dpid2** Distributed Protocol Interface – SNMP multiplexer protocol (DPI–SMUX) converter daemon เป็นกระบวนการที่อยู่เบื้องหลัง

### ไวยากรณ์

**dpid2** [-d [*Level*]]

## คำอธิบาย

คำสั่ง **dpid2** สตาร์ท **dpid2** DPI-SMUX converter daemon คำสั่งนี้ สามารถเรียกใช้โดยผู้ใช้ที่มีสิทธิ์ root หรือโดยสมาชิก ของกลุ่มระบบ

**dpid2** DPI-SMUX converter daemon จะคอมไพล์ด้วย Simple Network Management Protocol (SNMP) DPI เวอร์ชัน 2.0 มาตรฐานที่กำหนดโดยโปรโตคอล RFC 1592 และ SNMP SMUX และ Management Information Base (MIB) ที่กำหนดโดย RFC 1227

**dpid2** daemon ทำงานเหมือนกับ DPI 2.0 กับ SMUX converter ซึ่งใช้เพื่ออนุญาตให้ DPI subagents เช่น /usr/sbin/hostmibd สามารถสื่อสารกับเอเจนต์ AIX SNMP เวอร์ชัน 1 คอนเวอเตอร์จะเปลี่ยนข้อความ DPI2 เป็นข้อความโปรโตคอล SMUX และกลับกัน ตัว **dpid2** daemon เอง จะถูกใช้เป็นที่เชื่อมต่อกับ TCP พอร์ต 199 ของ เซิร์ฟเวอร์ SMUX ที่เป็นส่วนหนึ่งของเอเจนต์ **snmpd** สำหรับ DPI2 subagent (เช่น /usr/sbin/hostmibd) **dpid2** จะทำหน้าที่คล้ายกับเอเจนต์ DPI2 ซึ่งจะรับฟัง คำร้องขอการสื่อสารบนพอร์ต TCP ที่ไม่ได้กำหนดจาก DPI2 subagent หมายเลขพอร์ตนี้ถูกลบทะเบียนโดย **dpid2** daemon กับเอเจนต์ **snmpd** ผ่านตัวแปร MIB dpiPortForTCP (1.3.6.1.4.1.2.2.1.1.1) DPI2 subagent จะรู้จักหมายเลขพอร์ตนี้จากเอเจนต์ **snmpd** โดยการส่งเคียวรี get-request สำหรับอินสแตนซ์ dpiPortForTCP.0 (1.3.6.1.4.1.2.2.1.1.1.0) ไปยังเอเจนต์ **snmpd** หลังจาก DPI2 subagent รู้จักหมายเลขพอร์ต TCP ซึ่งเอเจนต์ DPI2 กำลังรอการสื่อสาร เอเจนต์จะพยายามเชื่อมต่อกับพอร์ต

โดยทั่วไป **dpid2** daemon จะรันระหว่างการเริ่มต้นระบบเมื่อเชลล์สคริปต์ /etc/rc.tcpip ถูกเรียกใช้

**dpid2** daemon ต้องถูกควบคุมโดยใช้ System Resource Controller (SRC) ไม่แนะนำให้ป้อน **dpid2** ที่บรรทัดคำสั่ง

ใช้คำสั่ง SRC ต่อไปนี้เพื่อจัดการ **dpid2** daemon:

### startsrc

เริ่มต้นระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย

stopsrc หยุดระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย

refresh ทำให้ระบบย่อยหรือกลุ่มของระบบย่อยอ่าน ไฟล์คอนฟิกูเรชันอีกครั้ง

lssrc รับข้อมูลสถานะของระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย

หมายเหตุ: ตัวเอเจนต์ **snmpdv3** จะทำงานเป็นเหมือนกับ DPI2 และรอการสื่อสารบนพอร์ต dpiPortForTCP.0 TCP ดังนั้นคุณไม่จำเป็นต้องใช้ **dpid2** daemon เมื่อคุณใช้เอเจนต์ **snmpdv3** ดังนั้น **dpid2** daemon จะไม่รันเมื่อระบบเริ่มทำงาน และบรรทัด **dpid2** ใน /etc/rc.tcpip จะถูกใส่เครื่องหมายหมายเหตุ

## แฟล็ก

ไอเท็ม	คำอธิบาย
-d Level	ระบุระดับการติดตามหรือดีบั๊ก
8	DPI level 1
16	DPI level 2
32	Internal level 1
64	Internal level 2
128	Internal level 3

เพิ่มตัวเลขสำหรับระดับการติดตามหลายระดับ  
หมายเหตุ: หากระบุแฟล็ก -d ไม่ได้ระบุหมายเลขระดับ ระดับดีฟอลต์จะเป็น 56 ถ้าไม่ได้ระบุแฟล็ก -d ระดับดีฟอลต์คือ 0

## ตัวอย่าง

1. เมื่อต้องการเริ่มต้น **dpid2** daemon ให้ป้อนคำสั่ง คล้ายกับคำสั่งต่อไปนี้:

```
startsrc -s dpid2 -a "-f /tmp/dpid2.log"
```

คำสั่งนี้ เริ่ม **dpid2** daemon และบันทึกข้อมูลไว้ในไฟล์ /tmp/dpid2.log ที่ระดับดีบั๊ก 0

2. เมื่อต้องการหยุด **dpid2** daemon ตามปกติ ให้ป้อนคำสั่งต่อไปนี้:

```
stopsrc -s dpid2
```

คำสั่งนี้หยุด **dpid2** daemon แฟล็ก -s ระบุระบบย่อยที่ติดตามเพื่อให้หยุดทำงาน

3. เมื่อต้องการรับสถานะแบบย่อจาก **dpid2** daemon ให้ป้อนคำสั่งต่อไปนี้:

```
lssrc -s dpid2
```

คำสั่งนี้ ส่งกลับชื่อของ daemon ID กระบวนการของ daemon และสถานะของ daemon (แอ็คทีฟหรือไม่แอ็คทีฟ)

## ไฟล์

ไอเท็ม	คำอธิบาย
/etc/snmpd.conf	ระบุรายการ SMUX peer ในไฟล์คอนฟิกูเรชันเอเจนต์ snmpd เวอร์ชัน 1
/etc/snmpd.peers	ระบุคอนฟิกูเรชันสำหรับ SMUX peer
/etc/mib.defs	กำหนดตัวแปร MIB ที่เอเจนต์ SNMP และตัวจัดการต้องรู้จัก และจัดการ

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“hostmibd Daemon” ในหน้า 799

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง snmpdv1

---

## คำสั่ง dping

### วัตถุประสงค์

ping โหนดหรืออุปกรณ์แบบขนาน

# ไวยากรณ์

`dping [-h] [-v] [-a] [-s] [-S] [-r] [-i interface...] [-w "selectstr"] [-H host_list] [-f filename] [-N nodegroup...] [-d devicename...] [-D devicegroup] [[-n] node_list]`

## คำอธิบาย

คำสั่ง `dping ping` เชิร์ฟเวอร์ที่ระบุเฉพาะ คำสั่งสามารถใช้เพื่อเรียกข้อมูลสถานะโหนด หรือเมื่อคุณสงสัยปัญหาเกี่ยวกับ Rational® Method Composer (RMC) และ `heartbeating` คำสั่ง `dping` ถูกใช้เพื่อ ping โหนดหรืออุปกรณ์ แบบขนาน - Ping โหนด pings ใดๆ อินเทอร์เน็ตโหนดที่สองในชุด ตัวอย่างเช่น `eth1` หรือ `mryi0` และ - Direct โหนด pings โหนดอื่นๆ

## คีย์เวิร์ด

ไอเท็ม	คำอธิบาย
-a	ระบุเพื่อ ping โหนดทั้งหมด แฟล็กนี้ไม่สามารถรวมกับแฟล็ก -n, -N, -f, -d, -H หรือ -w ได้
-c	ยุบเอาต์พุตที่เหมือนกันจากโหนดมากกว่า 1 โหนด และแสดงเพียง 1 ครั้ง
-d devicename...	ระบุอุปกรณ์ตั้งแต่หนึ่งอุปกรณ์ขึ้นไปที่จะค้นด้วยเครื่องหมายคอมมาเพื่อ ping อีกชื่อ เครื่องหมายดอกจัน (*) บ่งชี้อุปกรณ์ทั้งหมด
-D devicegroup...	ระบุกลุ่มของอุปกรณ์ตั้งแต่หนึ่งกลุ่มขึ้นไปที่จะค้นด้วยเครื่องหมายคอมมาเพื่อ ping
-f filename	ระบุไฟล์ที่มีรายการของโหนด หากชื่อไฟล์ที่ระบุไว้คืออักขระขีดเส้น (-) รายการจะถูกอ่านจากอินพุตมาตรฐาน ไฟล์สามารถมีบรรทัดจำนวนมาก และแต่ละบรรทัดสามารถแสดงชื่อโฮสต์ของโหนดหรือช่วงของโหนด ตั้งแต่หนึ่งโหนดขึ้นไปที่จะค้นด้วยเครื่องหมายคอมมา
-h	แสดงข้อมูลการใช้งานคำสั่ง
-H host_list	ระบุชื่อโฮสต์ที่ค้นด้วยเครื่องหมายคอมมา หรือช่องว่างที่ต้อง ping ชื่อโฮสต์เหล่านี้ต้องไม่เป็นโหนด NIM ที่นิยามไว้ ช่องว่างที่ค้นชื่อโฮสต์ต้องถูกระบุอยู่ใน เครื่องหมายอัญประกาศคู่ แฟล็ก -H ไม่สามารถระบุพร้อมกับแฟล็ก -n, -N, -f, -d, -w, หรือ -a
-i interface..	ระบุอินเทอร์เน็ตเครือข่ายตั้งแต่หนึ่งอินเทอร์เน็ตเฟสขึ้นไปที่จะค้นด้วยเครื่องหมายคอมมา เพื่อ ping สำหรับแต่ละโหนดที่ระบุไว้ แฟล็กจะสมมติว่า <code>nodename-interface.domain</code> แกไข IP แอดเดรสของเน็ตเวิร์กอะแดปเตอร์นั้นบนโหนด คุณต้องตั้งค่าการแก้ไขปัญหากับชื่อโฮสต์ ก่อนที่จะรัน <code>dping</code> หากหนึ่งในชื่อโฮสต์ เป็นสตริงว่างเปล่า ตัวอย่างเช่น " <code>eth1,eth2</code> " แฟล็กยัง ping ชื่อโฮสต์หลัก
-n node_list	ระบุชื่อโฮสต์ของโหนดตั้งแต่หนึ่งชื่อขึ้นไปที่จะค้นด้วยเครื่องหมายคอมมา หรือช่วงของโหนดที่ต้องการ ping แฟล็กนี้สามารถใช้พร้อมกับแฟล็ก -N และ -f ค่าของชื่อโฮสต์ สามารถระบุได้โดยไม่มี -n หากระบุอาร์กิวเมนต์ล่าสุด โปรดดูไฟล์ <code>noderange</code> สำหรับข้อมูลเกี่ยวกับช่วงของโหนด
-N nodegroup..	ระบุกลุ่มของโหนดตั้งแต่หนึ่งกลุ่มขึ้นไปที่จะค้นด้วยเครื่องหมายคอมมา ที่ต้องการรันคำสั่ง
-r	ping แบบเรียกซ้ำ แฟล็กนี้รันคำสั่ง <code>dsh</code> กับโหนดที่ ping เป็นผลสำเร็จ ซึ่งจะ ping โหนดที่เหลืออยู่ทั้งหมด ที่ระบุด้วยคำสั่ง <code>dping</code> จากโหนดเหล่านั้น
-s	ping โหนดตามลำดับแทนการ ping แบบขนาน แฟล็กนี้ไม่สามารถใช้พร้อมกับแฟล็ก -S ได้
-S	แสดงสรุปของผลลัพธ์ของการ ping เท่านั้น แฟล็กนี้ไม่สามารถใช้กับแฟล็ก -s ได้
-v	ระบุโหมดรายละเอียด
-w selectstr	แสดงโหนดที่ตรงกับ "where" clause ของสตริง select การระบุสตริงทั้งหมดภายในเครื่องหมายอัญประกาศคู่ จะช่วยคุณระบุสตริงค่าแอตทริบิวต์ภายใน เครื่องหมายอัญประกาศเดี่ยว อักขระเครื่องหมายดอกจัน (*) บ่งชี้ถึงโหนดทั้งหมด หาก "where" clause ไม่ได้ระบุไว้ แฟล็ก -w ไม่สามารถระบุพร้อมกับแฟล็ก -n, -N, -f, -d, -H, or -a

## ความปลอดภัย

คำสั่งต้องการการเข้าถึงเซิร์ฟเวอร์การจัดการคลัสเตอร์แบบ root

## ตัวอย่าง

1. หากต้องการ ping โหนดทั้งหมด ให้อ่าน:

```
dping -a
```

เอาต์พุต จะคล้ายกับ:

```
node1.localdomain: ping (alive)
```

node2.localdomain: noping (unreachable)

node3.localdomain: ping (alive)

2. หากต้องการ ping **group1** *nodegroup* และอินเทอร์เฟซเน็ต **eth1** ให้ป้อน:

```
dping -N group1 -i eth1
```

เอาต์พุตจะคล้ายกับ:

node1-eth1.localdomain: ping (alive)

node2-eth1.localdomain: noping (unreachable)

3. หากต้องการ ping ชื่อโฮสต์ ping **node1-eth2.clusters.com** ให้ป้อน

```
dping -i eth2 node1.clusters.com
```

เอาต์พุตจะคล้ายกับ: Exit Status 0 คำสั่งดำเนินการเป็นผลสำเร็จ 1 คำสั่งล้มเหลว 10 ไม่ได้ระบุโหนดหรืออุปกรณ์ไว้

## คำสั่ง **drmgr**

### วัตถุประสงค์

คำสั่ง **drmgr** ใช้เพื่อติดตั้ง และกำหนดค่าสคริปต์ dynamic logical partitioning (DLPAR)

### ไวยากรณ์

```
drmgr { -i script_name [-w minutes] [-f] | -u script_name } [-D hostname ]
```

```
drmgr [-b ]
```

```
drmgr [-R script_install_root_directory ]
```

```
drmgr [-S syslog_ID ]
```

```
drmgr [-I ]
```

### คำอธิบาย

สคริปต์ DLPAR ถูกจัดเตรียม โดยผู้ดูแลระบบและผู้จำหน่ายเพื่อประสานงานการใช้รีซอร์ส (เช่น ระบุตัวประมวลผลและหน่วยความจำที่ใช้จำนวนมาก) โดยแอฟพลิเคชั่นและมิดเดิลแวร์ที่มีการเพิ่มหรือการลบ รีซอร์สเหล่านั้นตามการกำหนดของระบบปฏิบัติการ สคริปต์ DLPAR จะถูกรันทั้งก่อนและหลังการดำเนินการ DLPAR DLPAR สคริปต์ถูกจัดเตรียมเพื่อที่แอฟพลิเคชั่นสามารถถูกปิดและ รีสตาร์ทอย่างเรียบร้อย

หมายเหตุ: แฟล็กแอ็คชันที่ระบุ ไม่สามารถรวมได้ นั่นคือ ผู้ใช้ไม่สามารถ รวมแฟล็ก **-R** และ **-S** แฟล็ก **-I** และ **-R** และอื่นๆ

### แฟล็ก

แฟล็ก	คำอธิบาย
-b	สร้างไฟล์ข้อมูลของสคริปต์ใหม่ที่ถูกจัดการโดย คำสั่ง <code>drmgr</code> โดยทั่วไป อีอ็อปชันนี้ต้องใช้เมื่อเรียกคืนสคริปต์จากระบบอื่นเท่านั้น
-D <i>hostname</i>	ระบุชื่อโฮสต์ของระบบที่สคริปต์สามารถเริ่มทำงาน
-f	บังคับการแทนที่สคริปต์ที่มีอยู่
-i <i>script_name</i>	ติดตั้งสคริปต์ <i>script_name script_name</i> ต้องมีพารามิเตอร์ที่สมบูรณ์ ถ้าไม่ระบุพารามิเตอร์จะใช้ไดเรกทอรีปัจจุบัน หากชื่อขัดแย้งกัน คำสั่ง <code>drmgr</code> จะแสดงค่าเตือนและไม่ติดตั้งสคริปต์ สคริปต์ใดๆ ที่มีอยู่แล้ว สามารถเขียนทับได้โดยการระบุแฟล็ก -f
-l	แสดงรายละเอียดเกี่ยวกับสคริปต์ DLPAR ที่สร้างขึ้นในปัจจุบัน
-R <i>base_script_directory</i>	เปลี่ยนไดเรกทอรีการติดตั้งสคริปต์ฐาน
-S <i>syslog_ID</i>	บันทึกข้อความ syslog ที่มีสตริง <i>syslog ID</i> ที่ระบุ สตริง ID นี้จะผนวกกับมีรายการที่บันทึกไว้ใน syslog โดยคำสั่ง <code>drmgr</code>
-u <i>script_name</i>	ยกเลิกการติดตั้งสคริปต์ DLPAR ถ้าสคริปต์ ถูกติดตั้งด้วยอีอ็อปชัน -D ต้องใช้พารามิเตอร์เดียวกันเพื่อถอนการติดตั้งสคริปต์ หากไม่ไดระบุไดเรกทอรี คำสั่ง <code>drmgr</code> จะลบสคริปต์ DLPAR ออกจากไดเรกทอรีการติดตั้ง "all"
-w <i>minutes</i>	เขียนทับค่าจำกัดเวลาที่ระบุโดย คำสั่งสำหรับสคริปต์ สคริปต์จะหยุดทำงานหากเกิน เวลาจำกัดที่ระบุ

## สถานะออก

- 0      ดำเนินการที่ร้องขอเป็นผลสำเร็จ
- >0     คำสั่งล้มเหลว สาเหตุของความล้มเหลวสามารถเป็น:
- ไม่มีไฟล์หรือไดเรกทอรี
  - ความยาวของพารามิเตอร์เกินข้อจำกัดของระบบ (PATH\_MAX)
  - มีการระบุอาร์กิวเมนต์มากเกินไป
  - คุณไม่มีสิทธิ์ root เพื่อรันคำสั่งนี้

## ข้อมูลที่เกี่ยวข้อง:

การแบ่งพาร์ติชันแบบไดนามิกโลจิคัล

## คำสั่ง `drslot`

### วัตถุประสงค์

จัดการสล็อตที่ปรับแต่งซ้ำได้แบบไดนามิกเช่น สล็อต hot plug

### ไวยากรณ์

เมื่อต้องการระบุ Hot Plug Slot

```
drslot -i { -s Slot | -I DeviceName } -c ConnectorType
```

เมื่อต้องการจัดเตรียม Hot Plug Slot สำหรับอุปกรณ์ที่ปรับแต่ง

```
drslot -a -s slot -c ConnectorType [ -I ]
```

เมื่อต้องการจัดเตรียม Hot Plug Slot สำหรับการลบอุปกรณ์

```
drslot -r { -s slot | -I DeviceName } -c ConnectorType [ -I ]
```

# เมื่อต้องการจัดเตรียม Hot Plug Slot สำหรับการลบ และการแทนที่อุปกรณ์

```
drslot -R { -s slot | -I DeviceName } -c ConnectorType [ -I ]
```

## คำอธิบาย

คำสั่ง **drslot** จัดการสล็อตที่ปรับแต่งได้แบบไดนามิก นั่นคือสล็อต hot plug สล็อต Hot plug เป็นตำแหน่งปลั๊กอินสำหรับ entities การเชื่อมต่อซึ่งสามารถถูกปรับแต่งโดยไม่ต้องมีการปิดระบบ หรือรีบูตระบบปฏิบัติการ สำหรับการดำเนินการเพิ่มเติม (**-a**) สล็อตต้องถูกระบุโดยตรงโดยใช้แฟล็ก **-s** กำหนด identifier เฉพาะสำหรับสล็อต สำหรับการดำเนินการ identify (**-i**), remove (**-r**), และ replace (**-R**) สล็อตอาจถูกระบุได้โดยตรง ด้วยแฟล็ก **-s** หรือทางอ้อม สล็อตอาจถูกระบุได้ ทางอ้อมโดยใช้แฟล็ก **-I** กำหนดชื่อโลจิคัล สำหรับอุปกรณ์ที่เชื่อมต่อกับสล็อต คำสั่ง **drslot** command กำหนดสล็อตที่อุปกรณ์ที่ระบุถูกเชื่อมต่อและจัดการ สล็อตนั้น

## Notes:

1. การดำเนินการลบและแทนที่จะล้มเหลว นอกจากอุปกรณ์ที่เชื่อมต่อกับสล็อตที่ระบุถูกยกเลิกการปรับแต่ง สำหรับข้อมูลเกี่ยวกับวิธียกเลิกการปรับแต่งอุปกรณ์ให้เสร็จสมบูรณ์ ดูที่ การจัดการตัวเชื่อมต่อ Hot Plug ใน *Operating system and device management*
2. หลังการดำเนินการจากเพิ่มหรือลบ คุณต้องรันคำสั่ง **cfgmgr** เพื่อให้อุปกรณ์ให้แฉ็คทีฟ และพร้อมใช้โดยระบบปฏิบัติการ

## แฟล็ก

หมายเหตุ: อย่าใช้แฟล็ก **-a**, **-i**, **-r**, **-R** ร่วมกัน

ไอเท็ม	คำอธิบาย
<b>-a</b>	เตรียมสล็อต hot plug สำหรับการปรับแต่งอุปกรณ์ที่เชื่อมต่อกับ สล็อต สล็อตมีการระบุแก่คุณก่อน และคุณจะได้รับพร้อมดีเพื่อยืนยัน สล็อต ต่อมา คุณจะได้รับพร้อมดีเพื่อยืนยันว่าอุปกรณ์ได้ถูก เชื่อมต่อกับสล็อต เมื่อมีการยืนยันว่าอุปกรณ์ได้ถูกเชื่อมต่อแล้ว สล็อตจะถูกเตรียมการและอุปกรณ์ถูกจัดเตรียมให้พร้อมสำหรับการปรับแต่ง
<b>-c ConnectorType</b>	ระบุ ConnectorType ของ สล็อต ซึ่งคุณกำลังดำเนินการอยู่ ตัวอย่าง ConnectorType สำหรับสล็อต hot plug PCI คือ pci
<b>-i</b>	แฟล็กนี้ต้องถูกระบุด้วยแฟล็ก <b>-a</b> , <b>-i</b> , <b>-r</b> และแฟล็ก <b>-R</b>
	ระบุสล็อต hot plug การระบุสล็อตขึ้นกับ ฮาร์ดแวร์ ตัวอย่าง ถ้าสล็อตมี LED การเรียกคำสั่ง <b>drslot -i</b> อาจทำให้ LED กระพริบ
<b>-I</b>	ระบุว่าขั้นตอน identification ควรถูกข้ามเมื่อใช้ แฟล็ก <b>-a</b> (add), <b>-r</b> (remove) และ <b>-R</b> (replace) แฟล็กนี้ควรถูกใช้เฉพาะเมื่อ คุณแน่ใจว่าคุณได้ระบุสล็อตที่ถูกต้องแล้ว
<b>-I DeviceName</b>	ระบุ DeviceName, ซึ่งเป็นชื่อ อุปกรณ์โลจิคัลของอุปกรณ์ที่เชื่อมต่อกับสล็อตที่จะถูกจัดการ แฟล็กนี้ต้อง ถูกใช้สำหรับแฟล็ก <b>-i</b> (identify), <b>-r</b> (remove) or <b>-R</b> (replace) ถ้าแฟล็ก <b>-s</b> ไม่ถูกใช้
<b>-r</b>	เตรียมสล็อต hot plug สำหรับการถอดอุปกรณ์ที่ยกเลิกการกำหนดค่าไว้ ก่อนหน้านั้นโดยใช้คำสั่ง <b>rmdev</b> หรือ <b>SMIT</b> ที่เทียบเท่า สล็อตถูกระบุและคุณ ได้รับพร้อมดีเพื่อยืนยันสล็อต ถ้ามีตัวบ่งชี้ที่มองเห็นได้สัมพันธ์กับ สล็อต จะถูกปิด สุดท้าย สล็อตถูกจัดเตรียมสำหรับการลบอุปกรณ์ และคุณจะได้รับพร้อมดีเพื่อยืนยันว่าอุปกรณ์ได้ถูกลบออกจาก สล็อต
<b>-R</b>	เตรียมสล็อต hot plug สำหรับการลบอุปกรณ์ที่ได้ถูก ยกเลิกการปรับแต่งก่อนหน้านี และแทนที่ด้วยอุปกรณ์ที่เท่าเทียมกัน อุปกรณ์ต้องถูกยกเลิกการกำหนดค่าโดยใช้คำสั่ง <b>rmdev</b> หรือ <b>SMIT</b> ที่เทียบเท่า <b>drslot</b> ระบุสล็อตและคุณได้รับพร้อมดีเพื่อยืนยัน สล็อต ถัดมา สล็อตถูกจัดเตรียมสำหรับการแทนที่อุปกรณ์ จากนั้น คุณจะได้รับพร้อมดีเพื่อยืนยันว่าอุปกรณ์ได้ถูกแทนที่ เมื่อมีการยืนยันว่าอุปกรณ์ได้ถูกแทนที่แล้ว ในสล็อต hot plug สล็อตจะถูกเตรียมการและอุปกรณ์ถูกจัดเตรียมให้พร้อมสำหรับการปรับแต่ง
<b>-s Slot</b>	ระบุ Slot ซึ่ง <b>drslot</b> ควรดำเนินการ แฟล็กนี้จำเป็นสำหรับการดำเนินการเพิ่มเติม ( <b>-a</b> ) แฟล็กนี้ ต้องใช้สำหรับการดำเนินการเพิ่ม ( <b>-i</b> ), ลบ ( <b>-r</b> ) หรือแทนที่ ( <b>-R</b> ) หากไม่ได้ใช้แฟล็ก <b>-I</b> รูปแบบของ Slot ขึ้นกับ Platform/ConnectorType

## ตัวอย่าง

1. เมื่อต้องการระบุสล็อต PCI hot plug จะแจ้งให้ป้อน:

```
drslot -i -c pci -s U0.1-P1-I3
```

ในตัวอย่างนี้ สล็อตมี LED ระบบอาจแสดงข้อความเหมือนดังต่อไปนี้:

ตัวบ่งชี้ที่มองเห็นได้ของ สล็อต PCI ที่ระบุได้ถูกขีดเป็นสถานะ ระบุ กด Enter เพื่อดำเนินการต่อหรือป้อน x เพื่อออก LED สำหรับสล็อตที่ระบุโดย U0.1-P1-I3 กระพริบจนกว่าคุณกดคีย์ Enter

2. เมื่อต้องการเพิ่มอะแดปเตอร์ Ethernet ที่ใส่ได้กับสล็อต hot plug โดยไม่มีการยืนยันสล็อต ให้ป้อน:

```
drslot -a -I -c pci -s U0.1-P1-I3
```

ไม่มีพร้อมต์การยืนยัน ถูกกำหนดสำหรับการระบุสล็อต จะมีพร้อมต์การยืนยันแสดง เมื่อถึงเวลาใส่อะแดปเตอร์ใหม่ลงในสล็อต และข้อความเหมือนดังต่อไปนี้แสดง:

ตัวบ่งชี้ที่มองเห็นได้สำหรับสล็อต PCI ที่ระบุได้ถูกขีดเป็นสถานะ ดำเนินการ ใส่การ์ด PCI ลงในสล็อตที่ระบุ เชื่อมต่ออุปกรณ์ที่จะถูกปรับแต่ง และกด Enter เพื่อดำเนินต่อ กด x เพื่อออก

หลังจากการเชื่อมต่ออะแดปเตอร์ กด Enter สล็อตจะถูกเตรียมพร้อม

3. เมื่อต้องการระบุสล็อต PCI เฉพาะก่อนการแทนที่การ์ด scsi ลงไป ให้ป้อน:

```
drslot -R -c pci -s U0.2-P1-I3
```

ระบบแสดงข้อความเหมือนดังต่อไปนี้:

ตัวบ่งชี้ที่มองเห็นได้ของ สล็อต PCI ที่ระบุได้ถูกขีดเป็นสถานะ ระบุ กด Enter เพื่อดำเนินการต่อหรือป้อน x เพื่อออก

LED สำหรับสล็อต PCI กระพริบเพื่อระบุสล็อต การกดคีย์ที่ไม่ใช่ คีย์ Enter จะออกจากคำสั่ง กด Enter ดำเนินการต่อกับสล็อตนี้ ถ้าดำเนินต่อ LED สำหรับสล็อต PCI ถูกเปลี่ยนเป็นสถานะ ดำเนินการ และ ระบบแสดงข้อความเหมือนดังต่อไปนี้:

ตัวบ่งชี้ที่มองเห็นได้สำหรับสล็อต PCI ที่ระบุได้ถูกขีดเป็นสถานะ ดำเนินการ แทนที่การ์ด PCI ลงในสล็อตที่ระบุ เชื่อมต่ออุปกรณ์ที่จะถูกปรับแต่งอีกครั้ง และกด Enter เพื่อดำเนินต่อ กด x เพื่อออก การออกในขณะนี้ จะปล่อยสล็อต PCI ให้อยู่ในสถานะ เอาออกแล้ว

## ไฟล์

/usr/sbin/drslot

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง lsslot

คำสั่ง rmdev

คำสั่ง cfgmgr

การจัดการฮอตปลั๊ก PCI

---

## คำสั่ง dscrctl

### วัตถุประสงค์

ตั้งค่าคุณสมบัติการดึงข้อมูลล่วงหน้าดีฟอลต์ของระบบปฏิบัติการ

### ไวยากรณ์

เมื่อต้องการเคียวรีคุณสมบัติของ สตรีมฮาร์ดแวร์บนคอมพิวเตอร์:

```
dscrctl -q
```

เมื่อต้องการ ตั้งค่าความลึกในการดึงข้อมูลล่วงหน้าดีฟอลต์ของระบบปฏิบัติการบนคอมพิวเตอร์ แบบชั่วคราว (สำหรับเซสชันปัจจุบัน) หรือแบบถาวร (หลังจากการดำเนินการรีสตาร์ท แต่ละครั้ง):

```
dscrctl [-n] [-b] -s dscr_value
```

เมื่อต้องการ ยกเลิกการตั้งค่าถาวรของความลึกการดึงข้อมูลล่วงหน้าดีฟอลต์ของระบบปฏิบัติการ ในเวลาเริ่มทำงาน:

```
dscrctl -c
```

### คำอธิบาย

คำสั่งย่อย **dscrctl -q** แสดงจำนวนสตรีมฮาร์ดแวร์ และความลึกการดึงข้อมูลล่วงหน้าดีฟอลต์ของแพลตฟอร์ม และระบบปฏิบัติการ ผู้ใช้ใดๆ สามารถรันคำสั่งย่อยนี้

คำสั่งย่อย **dscrctl -s** ตั้งค่าความลึกการดึงข้อมูลล่วงหน้าดีฟอลต์ของ ระบบปฏิบัติการ คุณต้องมีสิทธิ์ root เพื่อรันคำสั่งย่อยนี้ ค่าดีฟอลต์นี้สามารถเปลี่ยนแปลงสำหรับเซสชันปัจจุบันโดยใช้แฟล็ก **-n** ในเวลาเริ่มทำงานโดยใช้แฟล็ก **-b** หรือสำหรับทั้งเซสชันปัจจุบัน และในเวลาเริ่มทำงานโดยใช้แฟล็ก **-n -b** ร่วมกับคำสั่ง **dscrctl**

อ็อปชัน **dscrctl -c** ยกเลิกค่าติดตั้งความลึกการดึงข้อมูลดีฟอลต์ของระบบปฏิบัติการ ในเวลาเริ่มทำงาน อ็อปชันนี้ลบคำสั่ง **dscrctl** ออกจากไฟล์ **/etc/inittab** ดังนั้นจึงจะมีผลหลัง การดำเนินการรีสตาร์ทถัดไป

### แฟล็ก

- q แสดงจำนวนสตรีมฮาร์ดแวร์ที่ได้รับการสนับสนุนโดย แพลตฟอร์ม และยังแสดงค่าของความลึกการดึงข้อมูลล่วงหน้าดีฟอลต์ของเฟิร์มแวร์และระบบปฏิบัติการ
- c ยกเลิกค่าติดตั้งถาวรของความลึกการดึงข้อมูลล่วงหน้าในเวลา เริ่มทำงานโดยการลบคำสั่ง **dscrctl** ออกจากไฟล์ **/etc/inittab**
- n เปลี่ยนค่าเวลาเริ่มของความลึกการดึงข้อมูลล่วงหน้าดีฟอลต์ของ ระบบปฏิบัติการ แฟล็กนี้ใช้ร่วมกับแฟล็ก **-s** การเปลี่ยนแปลงไม่คงอยู่ในการดำเนินการบูตครั้งหนึ่งถึงครั้งถัดไป
- b ทำให้การเปลี่ยนแปลงคงอยู่ตลอดการดำเนินการบูตโดยการเพิ่มคำสั่ง **dscrctl** ในไฟล์ **/etc/inittab** แฟล็กนี้ใช้ร่วมกับแฟล็ก **-s**

-s *dscr\_value*

กำหนดค่าสำหรับความลึกการดึงข้อมูลล่วงหน้าดีฟอลต์ของ ระบบปฏิบัติการ ค่าถูกถือเป็นเลขทศนิยม ยกเว้นว่าขึ้นต้นด้วย 0x ซึ่งกรณีเช่นนั้นจะถือว่าเป็นจำนวน เลขฐานสิบหก

## ตัวอย่าง

1. เมื่อต้องการตั้งค่าของความลึกการดึงข้อมูลล่วงหน้าดีฟอลต์ของ ระบบปฏิบัติการเป็น 13 สำหรับเซสชันปัจจุบัน ให้ป้อน:

```
# dscrctl -n -s 13
```

2. เมื่อต้องการแสดงค่าติดตั้งปัจจุบันของกลไกสตรีมฮาร์ดแวร์ให้ป้อน:

```
# dscrctl -q
```

เอาต์พุตต่อไปนี้จะแสดง:

ค่าติดตั้ง DSCR ปัจจุบัน:

Data Streams Version = V2.06

number\_of\_streams = 16

platform\_default\_pd = 0x5 (DPFD\_DEEP)

os\_default\_pd = 0xd (DSCR\_SSE | DPFD\_DEEP)

ข้อมูลที่เกี่ยวข้อง:

/etc/inittab file

---

## คำสั่ง dscreen

### วัตถุประสงค์

สตาร์ท ยูทิลิตี้ Dynamic Screen

### ไวยากรณ์

```
dscreen [ -i InfoFile ] [ -t TermType ]
```

### คำอธิบาย

คำสั่ง **dscreen** สตาร์ท ยูทิลิตี้ Dynamic Screen ซึ่งอนุญาตให้ฟิลิคัลเทอร์มินัลเดี่ยวถูกเชื่อมต่อ กับหลายเซสชันเสมือน หรือจอภาพ ในหนึ่งครั้ง

ถ้าไม่มีการระบุแฟล็ก คำสั่ง **dscreen** อ่านรายละเอียดสำหรับเทอร์มินัลที่ระบุในตัวแปรสถานะแวดล้อม **TERM** จากไฟล์ที่ระบุในตัวแปรสถานะแวดล้อม **DSINFO** ถ้าตัวแปรสถานะแวดล้อม **DSINFO** ไม่ถูกระบุ รายละเอียดเทอร์มินัลถูกอ่าน จากไฟล์ **/etc/dsinfo** รายละเอียดเทอร์มินัลโดยปกติมีข้อมูล configuration ต่อไปนี้

- คีย์ที่ใช้กับยูทิลิตี้ Dynamic Screen และฟังก์ชัน
- จำนวนเพจของหน่วยความจำจอภาพที่เทอร์มินัล มีอยู่
- ลำดับโค้ดต้องถูกส่งหรือรับ เพื่อเข้าถึงและใช้คุณลักษณะ Dynamic Screen

### แฟล็ก

ไอเท็ม	คำอธิบาย
-i InfoFile	ระบุไฟล์ที่มีการแก้ไขที่เลือกสำหรับใช้กับ ยูทิลิตี้ Dynamic Screen อีพซันนี้มีประโยชน์เมื่อใช้ Dynamic Screen ที่กำหนดเริ่มแรก ชัดแย้งกับหนึ่งในแอปพลิเคชันของคุณ
-t TermType	ถ้าแฟล็กนี้ไม่ถูกระบุ ข้อมูล configuration ของเทอร์มินัลถูกอ่านจากไฟล์ที่ระบุในตัวแปรสถานะแวดล้อม DSINFO ถ้าเซตไว้ มิฉะนั้น ข้อมูลถูกอ่านจากไฟล์ /etc/dsinfo ระบุรายละเอียดเทอร์มินัลที่จะถูกอ่านจากไฟล์ที่มีการแก้ไข อีพซันนี้มีประโยชน์เมื่อชนิดเทอร์มินัลที่ต้องการ ไม่ตรงกับ การตั้งค่าของตัวแปรสถานะแวดล้อม TERM

## ตัวอย่าง

- เมื่อต้องการสตาร์ทยูทิลิตี้ Dynamic Screen โดยใช้ การแก้ไขที่ฟอลต์ให้ป้อน:

```
dscreen
```

นี่เป็นการเซตตัวแปรสถานะแวดล้อม DSINFO และ TERM ตามที่กำหนดในไฟล์ /etc/dsinfo ดีฟอลต์

- เมื่อต้องการเริ่มยูทิลิตี้ Dynamic Screen และระบุ ไฟล์ที่มีคีย์ทางเลือก การแก้ไขยังระบุรายละเอียดเทอร์มินัล ที่จะถูกอ่านจากไฟล์ให้ป้อน:

```
dscreen -i myfile -t myterm
```

นี่เป็นการใช้ข้อมูลจากชนิดไฟล์ dsinfo-ที่ผู้ใช้สร้างชื่อ myinfo เพื่อจัดการความจำเป็นการแก้ไขที่ไม่ปกติ ไฟล์ myinfo ยังมีข้อกำหนดเทอร์มินัลชื่อ myterm

- เมื่อต้องการสตาร์ทยูทิลิตี้ Dynamic Screen และระบุ เซ็ตอัพเทอร์มินัลทางเลือก ให้ป้อน:

```
dscreen -t wy60-wp
```

นิยามเทอร์มินัลนี้ (อยู่ในไฟล์ /etc/dsinfo) เซตการดำเนินการคีย์ที่กำหนด dscreen เพื่อที่ไม่ให้ขัดแย้งกับลำดับคำสั่งคีย์ควบคุมในแอปพลิเคชันการประมวลผลคำ ที่ใช้

## ไฟล์

ไอเท็ม	คำอธิบาย
/etc/dsinfo	มีรายละเอียดเทอร์มินัลสำหรับยูทิลิตี้ Dynamic Screen

### ข้อมูลที่เกี่ยวข้อง:

ยูทิลิตี้ Dynamic screen

## คำสั่ง dshbak

คำสั่งนี้เป็นส่วนหนึ่งของซอฟต์แวร์ IBM Distributed Shell Management (DSM) คำสั่งจะวางอยู่ที่ตำแหน่ง /opt/ibm/sysmgmt/dsm/bin/dshbak

### วัตถุประสงค์

แสดงเอาต์พุตที่จัดรูปแบบแล้วจากคำสั่ง dsh

### ไวยากรณ์

```
dshbak [-c | -x]
```

206 AIX เวอร์ชัน 7.2: ข้อมูลอ้างอิงคำสั่ง วอลุ่ม 2, d - h

## คำอธิบาย

คำสั่ง **dshbak** จัดรูปแบบเอาต์พุตจากคำสั่ง **dsh** ไวยการณ์ของคำสั่ง **dshbak** เป็นดังนี้:

*host\_name*: บรรทัดของเอาต์พุตจากคำสั่ง **remote**

คำสั่ง **dshbak** จัดรูปแบบบรรทัดและ เขียนบรรทัดเหล่านั้นลงในเอาต์พุตมาตรฐานดังนี้ ข้อสรุปคือ เอาต์พุตจาก *host\_name3* and *host\_name4* เป็นเอาต์พุตที่เหมือนกันและระบุแฟล็ก **-c** ไว้

```
HOSTS -----
      host_name1
      -----
      .
      .
      lines from dsh with host_names stripped off
      .
      .
      HOSTS -----
      host_name2
      -----
      .
      .
      lines from dsh with host_names stripped off
      .
      .
      HOSTS -----
      host_name3          host_name4
      -----
      .
      .
      lines from dsh with host_names stripped off
      .
      .
```

ชื่อโฮสต์จะแสดงตามตัวอักษร หากเอาต์พุตถูกแสดงจากโหนดมากกว่าหนึ่งโหนดในรูปแบบที่ถูกยุบไว้ เอาต์พุตจะเรียงลำดับตามตัวอักษร ตามชื่อโฮสต์หากเอาต์พุตไม่ได้ถูกยุบไว้ คำสั่ง **dshbak** จะเขียน "." สำหรับแต่ละ 1000 บรรทัดเอาต์พุตที่กรองแล้ว

หากระบุแฟล็ก **-x** ไว้ บรรทัดส่วนหัวพิเศษ ซึ่งคำสั่ง **dshbak** แสดงสำหรับแต่ละโหนดที่แยกออก คำสั่ง **dshbak** เรียงลำดับเอาต์พุตโดยใช้ชื่อโหนดเพื่อค้นหา:

```
host_name1: lines from dsh started
      .
      .
      lines from dsh continued
      .
      .
      lines from dsh ended
      host_name2: lines from dsh started
      .
      .
```

```
lines from dsh continued
.
.
lines from dsh ended
```

## แฟล็ก

ไอเท็ม	คำอธิบาย
-c	ยุบเอาต์พุตที่เหมือนกันจากโหนดมากกว่าหนึ่งโหนด เพื่อแสดงเอาต์พุตในหนึ่งครั้ง
-x	แยกบรรทัดส่วนหัวพิเศษที่ dshbak แสดงสำหรับแต่ละโหนด แฟล็กนี้จะเตรียมเอาต์พุตที่บีบอัด และคำสั่ง dshbak จะเรียงลำดับเอาต์พุตตามชื่อโหนดเพื่อดูเนื้อหา แฟล็กต้องไม่ถูกใช้พร้อมกับ -c

## ความปลอดภัย

หมายเหตุ: คุณต้องรันคำสั่ง kinit เพื่อขอรับ ticket-grant-ticket ก่อนที่จะรัน Kerberos เวอร์ชัน 5 คำสั่ง remote ข้อควรพิจารณาความปลอดภัยเพิ่มเติม จะคล้ายกับคำสั่ง remote shell

## ตัวอย่าง

- เมื่อต้องการแสดงผลลัพธ์ของคำสั่งที่ใช้บนโหนดต่างๆ ในรูปแบบที่ใช้ใน คำอธิบาย ให้ป้อนคำสั่งต่อไปนี้:  

```
dsh -n node1,node2,node3 cat /etc/passwd | dshbak
```
- เมื่อต้องการแสดงผลลัพธ์ของคำสั่งที่ใช้บนโหนดต่างๆ พร้อมกับเอาต์พุตที่เหมือนกัน ให้ป้อนคำสั่งต่อไปนี้:  

```
dsh -w host1,host2,host3 pwd | dshbak -c
```
- เมื่อต้องการแสดงผลลัพธ์ของคำสั่งที่ใช้บนโหนดต่างๆ พร้อมกับเอาต์พุตที่บีบอัด ให้ป้อนคำสั่งต่อไปนี้:

หมายเหตุ: เอาต์พุต จะถูกเรียงลำดับตามตัวอักษรตามชื่อโฮสต์

```
dsh -w host1,host2,host3 date | dshbak -x
```

## ข้อผิดพลาดมาตรฐาน

ข้อความแสดงความผิดพลาดบนข้อผิดพลาดมาตรฐานจะถูกแสดงก่อนข้อความ เอาต์พุตมาตรฐานทั้งหมดหากใช้ตัวกรอง dshbak ลักษณะการทำงานนี้เป็นจริงและไม่มีแฟล็ก -c

---

## คำสั่ง dsh

### วัตถุประสงค์

รันคำสั่งพร้อมกันบนโหนดและอุปกรณ์ฮาร์ดแวร์จำนวนมาก

### ไวยากรณ์

```
dsh -h dsh -V dsh -q dsh [-a] [--all-nodes context_list]
```

```
[-A] [--all-devices context_list] [-n
```

```
node_list] [-N nodegroups] [-d device_list]
```

`[-D devicegroups] [-C context]`

`[-c] [-e] [-E environment_file]`

`[-f fanout] [-F output_path]`

`[-i] [-l user_ID] [-L]`

`[--log log_file] [-m] [-o`

`node_options] [-O device_options] [-Q]`

`[-r node_remote_shell] [--device-rsh`

`device_remote_shell] [-s] [-S`

`csch | ksh] [-t timeout]`

`[-T] [-v] [-X`

`env_list] [-z] [--report`

`report_path] [--report-name report_name] [command_list]`

## คำอธิบาย

คำสั่ง `dsh` จะรันคำสั่งบน เป้าหมายแบบรีโมตอย่างพร้อมเพียงกัน - โหนด อุปกรณ์ฮาร์ดแวร์ หรือทั้งสอง เป้าหมาย สามารถเลือกจากคอนเท็กซ์จำนวนมากได้ คอนเท็กซ์คือฐานข้อมูลเป้าหมาย ที่มีนิยามของโหนดและอุปกรณ์ เช่น ฐานข้อมูล NIM คำสั่ง `dsh` ใช้คำสั่ง `remote shell` สำหรับแต่ละเป้าหมายที่ระบุ ซึ่งจะส่งคืนเอาต์พุตจาก เป้าหมายทั้งหมดด้วยวิธีการจัดรูปแบบ เพื่อเปิดใช้งานผลลัพธ์ของคำสั่งจาก โหนดทั้งหมดที่ต้องถูกจัดการได้อย่างง่ายดาย `/usr/bin/rsh` คือโมเลสำหรับไวยากรณ์ และความปลอดภัยและคำสั่ง `dsh` คือ DSM Distributed Shell Utility

## พารามิเตอร์

### ไอเท็ม

#### TARGET CONTEXT

#### คำอธิบาย

คอนเท็กซ์เป้าหมายของคำสั่ง `dsh` คือฐานข้อมูลที่นิยามเป้าหมายหรือกลุ่มของเป้าหมาย คอนเท็กซ์ดีฟอลต์สามารถกำหนดคอนเท็กซ์ได้โดยใช้แฟล็กคอนเท็กซ์ `-C` หรือตัวแปรสถานะแวดล้อม `DSH_CONTEXT` หากไม่ได้ระบุพารามิเตอร์ใดๆ ไว้ คอนเท็กซ์ดีฟอลต์คือ NIM เมื่อคำสั่ง `dsh` รันจากเซิร์ฟเวอร์การจัดการ NIM คอนเท็กซ์ดีฟอลต์คือ DSH (โปรดดู `DSH_CONTEXT`) คอนเท็กซ์ ถูกใช้กับคำสั่ง `DSH Utilities` โดยติดตั้งส่วนขยายไฟล์คอนเท็กซ์ ในไดเรกทอรี `/opt/ibm/sysmgt/dsm/pm/Context` คอนเท็กซ์เป้าหมายหรือกลุ่มเป้าหมายสามารถระบุได้โดยการตรวจสอบ ชื่อเป้าหมายกับชื่อคอนเท็กซ์ หรือนิยามโดยคอนเท็กซ์ดีฟอลต์ สำหรับชื่อเป้าหมายที่ไม่ผ่านการตรวจสอบ (โปรดดู `Target list`)

#### DSH CONTEXT

`DSH_CONTEXT` คือคอนเท็กซ์ในตัว สำหรับคำสั่ง `DSH Utilities` ทั้งหมด ซึ่งอนุญาตให้ใช้ฐานข้อมูลกลุ่มของโหนดที่นิยามโดยผู้ใช้ซึ่งมีอยู่ในระบบโลคัลไฟล์ ตัวแปรสถานะแวดล้อม `DSH_NODEGROUP_PATH` ระบุพาธไปยังฐานข้อมูลกลุ่มของโหนด แต่ละไฟล์ในไดเรกทอรีนี้ แสดงถึงกลุ่มของโหนด และมีชื่อโฮสต์หนึ่งชื่อหรือแอดเดรส TCP/IP สำหรับแต่ละโหนดที่เป็นสมาชิกของกลุ่ม บรรทัดว่างและบรรทัดข้อคิดเห็น ที่ขึ้นต้นด้วยสัญลักษณ์ `#` จะถูกข้าม หากโหนดทั้งหมดถูกร้องขอสำหรับ `DSH_CONTEXT` รายการโหนดแบบเต็มจะถูก build จากกลุ่มทั้งหมดในไดเรกทอรี `DSH_NODEGROUP_PATH` และแคชใน `/var/ibm/sysmgt/dsm/dsh/$DSH_NODEGROUP_PATH/AllNodes` ไฟล์นี้ถูกสร้างขึ้นใหม่ในแต่ละครั้งที่ไฟล์กลุ่มถูกแก้ไขหรือเพิ่มให้กับไดเรกทอรี `DSH_NODEGROUP_PATH` เป้าหมายของอุปกรณ์ ไม่ได้รับการสนับสนุนในคอนเท็กซ์ `DSH`

ไอเท็ม  
TARGET  
SPECIFICATION

คำอธิบาย

เป้าหมายคือ โหนด หรืออุปกรณ์ฮาร์ดแวร์ที่ออกใช้คำสั่ง remote เป้าหมายของโหนดถูกระบุไว้ โดยใช้แฟล็ก `-a`, `--all-nodes context_list`, `-n node_list` และ `-N nodegroups` หรือตัวแปรสถานะแวดล้อม `DSH_NODE_LIST` ถ้าใช้ทั้งแฟล็ก `-N` และตัวแปรสถานะแวดล้อม `DSH_NODE_LIST` กลุ่มและรายการจะรวมเข้าด้วยกันเพื่อตัดปัญหาความซ้ำซ้อน

หมายเหตุ: ตัวแปรสถานะแวดล้อม `DSH_NODE_LIST` แทนที่ `WCOLL` เป้าหมายของอุปกรณ์ถูกระบุไว้โดยใช้ `-A`, `--all-devices context_list`, `-d device_list` และ `-D devicegroups flags` หรือตัวแปรสถานะแวดล้อม `DSH_DEVICE_LIST` หากโลคัลโฮสต์ถูกรวมเป็นส่วนหนึ่งของเป้าหมาย `command_list` จะรันโดยตรงบนโลคัลโฮสต์ และไม่ผ่าน remote shell ที่กำหนดคอนฟิกไว้ เว้นเสียแต่ว่า `user_ID` ถูกระบุไว้เพื่อเรียกทำงานกับโลคัลโฮสต์ (โปรดดู ผู้ใช้แบบรีโมต) ตัวแปรสถานะแวดล้อม

`DSH_NODE_LIST` และ `DSH_DEVICE_LIST` ระบุไฟล์ที่แสดงโหนดและอุปกรณ์ของเป้าหมาย รูปแบบไฟล์คือ 1 เป้าหมายต่อบรรทัด บรรทัดว่างและบรรทัดขอลัดขึ้น ที่ขึ้นต้นด้วยสัญลักษณ์ `#` จะถูกข้าม ทั้งเป้าหมายของโหนดและอุปกรณ์สามารถระบุไว้พร้อมกัน แต่ชื่อเป้าหมายเดียวกันไม่สามารถใช้สำหรับทั้ง โหนดและอุปกรณ์ หากชื่อที่คล้ายกันถูกใช้ โปรแกรมจะข้ามเป้าหมายที่ซ้ำกัน และดำเนินการเรียกทำงานต่อไปบนเป้าหมายอื่น เป้าหมายของโหนด และอุปกรณ์ยังสามารถถูกระบุโดยใช้ช่วงของโหนด โปรดดู ไฟล์ `noderange` สำหรับรายละเอียด หากระบุเป้าหมายเดียวกัน มากกว่าหนึ่งครั้ง คำสั่ง `remote` จะรันเพียงหนึ่งครั้งบน

TARGET LIST

เป้าหมายที่ระบุ  
เป้าหมายและกลุ่มของเป้าหมายถูกระบุไว้โดยใช้รูปแบบต่อไปนี้:

```
[context:][user_ID@]target[, [context:][user_ID@]target]...
```

โดยที่ `context` คือข้อกำหนดคุณลักษณะของคอนเท็กซ์สำหรับเป้าหมาย `user_ID` คือชื่อผู้ใช้เพื่อเลือกที่ต้องการใช้ เมื่อรันคำสั่งแบบรีโมตบนเป้าหมายและ `target` คือชื่อหรือ TCP/IP แอดเดรสของเป้าหมายตามที่อนุญาตให้ใช้โดยคอนเท็กซ์ของ เป้าหมาย สำหรับนิพจน์ `noderange` นั้น `user_ID` ถูกใช้สำหรับปลายทางแต่ละค่าในรายการ ที่เป็นผลมาจากการประเมินผลของนิพจน์ `noderange` หากรายการเป้าหมายถูกระบุไว้เฉพาะกับ `dash (-)` เป้าหมายจะสามารถระบุแบบโต้ตอบได้ สำหรับโหนด พร้อมต์คือ `dsh node>` และสำหรับอุปกรณ์ พร้อมต์คือ `dsh device>` ระบุรายการเป้าหมายที่ใช้ไวยากรณ์ต่อไปนี้ บนหนึ่งบรรทัดในหนึ่งครั้ง:

```
[context:][user_ID@]target
```

โดยที่ `context` คือข้อกำหนดคุณลักษณะของคอนเท็กซ์สำหรับเป้าหมาย `user_ID` คือชื่อผู้ใช้เพื่อเลือกที่ต้องการใช้ เมื่อรันคำสั่งแบบรีโมตบนเป้าหมาย และเป้าหมายคือชื่อ หรือ TCP/IP แอดเดรสของเป้าหมายตามที่อนุญาตไว้โดยคอนเท็กซ์ของ เป้าหมาย สำหรับนิพจน์ `noderange` นั้น `user_ID` ถูกใช้สำหรับปลายทางแต่ละค่าในรายการ ที่เป็นผลมาจากการประเมินผลของนิพจน์ `noderange` หากรายการเป้าหมายถูกระบุไว้เฉพาะกับเครื่องหมายขีดคั่น `(-)` เป้าหมายสามารถถูกระบุไว้แบบโต้ตอบ สำหรับโหนด พร้อมต์คือ `dsh node>` สำหรับอุปกรณ์ พร้อมต์คือ `dsh device>` ระบุรายการเป้าหมายโดยใช้ไวยากรณ์ต่อไปนี้บนหนึ่งบรรทัดในหนึ่งครั้ง:

COMMAND  
SPECIFICATION

```
[context:][user_ID@]target
```

เมื่อคุณทำเสร็จสิ้นแล้ว ให้กด `Ctrl-d` เพื่อดำเนินการต่อ คำสั่งที่ต้องการรันบนเป้าหมายแบบรีโมตถูกระบุไว้โดย `command_list` พารามิเตอร์ `dsh` การป้อนคำสั่ง บนบรรทัดรับคำสั่งในโหมดการโต้ตอบ การจัดเตรียม `command_list` ผ่านอินพุตมาตรฐาน หรือการรันสคริปต์แบบโลคัลโดยใช้แฟล็ก `-e` ไวยากรณ์สำหรับ `command_list` พารามิเตอร์ `dsh` เป็นดังต่อไปนี้: `"command[; command]..."` โดยที่ คำสั่งคือคำสั่งที่ต้องการรันบนเป้าหมายแบบรีโมต เครื่องหมายคำถาม ถูกบังคับไว้ต้องมีเพื่อให้มั่นใจว่า คำสั่งทั้งหมดในรายการถูกรันแบบรีโมต และอักขระพิเศษใดๆ ถูกตีความไว้อย่างถูกต้องบน เป้าหมายแบบรีโมต สคริปต์ไฟล์บนโลคัลโฮสต์ถูกรันอยู่บนแต่ละเป้าหมายแบบรีโมต โดยใช้แฟล็ก `-e` หาก `-e` ถูกระบุไว้ `command_list` คือชื่อสคริปต์และอาร์กิวเมนต์กับสคริปต์ ตัวอย่างเช่น:

```
dsh -e[flags] script_filename [arguments]...
```

REMOTE USER

ไฟล์ `script_filename` ถูกคัดลอกไปยังไฟล์สุม ที่มีชื่ออยู่ในไดเรกทอรี `/tmp` บนแต่ละเป้าหมายแบบรีโมต จากนั้นรันบนเป้าหมาย ถ้าพารามิเตอร์ `command_list` ไม่ถูกระบุ `dsh` เข้าสู่โหมดบรรทัดรับคำสั่งแบบโต้ตอบ และพร้อมต์แสดงพร้อมต์ `dsh>` ป้อนคำสั่งที่พร้อมต์ `dsh>` โดยใช้ไวยากรณ์ ต่อไปนี้: `[!] "command"` โดย `command` คือคำสั่งที่จะรัน บนปลายทางรีโมต เครื่องหมายอุทธาน (!) ที่นำหน้าคำสั่ง เป็นสาเหตุทำให้คำสั่งรันอยู่บนโลคัลโฮสต์เท่านั้น และไม่ได้อยู่บน เป้าหมายแบบรีโมต คำสั่ง `dsh` รันคำสั่งบนเป้าหมายที่แก้ปัญหาไว้ และผลลัพธ์จะถูกแสดง จากนั้นกลับไปพร้อมต์ `dsh>` เมื่อต้องการออกจากโหมด บรรทัดรับคำสั่ง ป้อน `exit` ที่พร้อมต์ `dsh>` คำสั่ง `dsh` ไม่ทำงานกับคำสั่ง แบบโต้ตอบใดๆ ซึ่งประกอบด้วยคำสั่งเหล่านั้นที่อ่านจาก อินพุตมาตรฐาน `user_ID` ที่ต้องการใช้สำหรับ เป้าหมายแบบรีโมตสามารถระบุไว้เป็นส่วนหนึ่งของไวยากรณ์เป้าหมาย (โปรดดู Target lists) หรือใช้แฟล็ก `-I` (ตัวพิมพ์เล็ก L) หากทั้งสองเมธอดถูกใช้ `user_ID` ถูกกำหนดไว้ดังต่อไปนี้:

1. สำหรับปลายทางที่ระบุเป็น `user_ID@target`, `user_ID` ถูกใช้สำหรับการเรียกใช้งานรีโมตบนปลายทาง และแฟล็ก `-I` ถูกข้าม
2. สำหรับปลายทางที่ไม่ได้ระบุโดยใช้ `user_ID@target` จะใช้ `user_ID` สำหรับการเรียกใช้งานรีโมตบนปลายทางโดยพิจารณา ดังนี้: `- user_ID` ที่ระบุด้วยแฟล็ก `-I` หาก `-I` ไม่ได้ถูกระบุไว้ ผู้ใช้ปัจจุบันที่รันคำสั่งจะถูกระบุไว้ หาก `user_ID` ถูกระบุไว้สำหรับโลคัลโฮสต์ ซึ่งรวมในรายการเป้าหมาย คำสั่ง `remote shell` รัน `command_list` บนโลคัลโฮสต์ เพื่อทำให้มั่นใจว่า ID ล็อกอินมีความปลอดภัย

ไอเท็ม

REMOTE SHELL  
COMMAND

คำอธิบาย

คำสั่งที่ต้องการรันบนเป้าหมายแบบรีโมตถูกระบุไว้โดย `command_list` พารามิเตอร์ `dsh` การป้อนคำสั่ง บนบรรทัดรับคำสั่งใน โหมดการโต้ตอบ การจัดเตรียม `command_list` ถึงขั้นพจนานุกรม หรือการเรียกทำงานโลคัลสคริปต์โดยใช้แฟล็ก `-e` ไวยากรณ์ สำหรับ `command_list` พารามิเตอร์ `dsh` เป็นดังต่อไปนี้: "`command[; command]...`" โดยที่ คำสั่งคือคำสั่งที่ต้องการรันบนเป้าหมายแบบรีโมต เครื่องหมายคำถาม ถูกบังคับไว้ต้องมีเพื่อให้เห็นใจว่า คำสั่งทั้งหมดในรายการถูกรันบนเป้าหมายรีโมต และอักขระพิเศษใดๆ ถูกตีความไว้อย่างถูกต้องบน เป้าหมายแบบรีโมต สคริปต์ไฟล์บนโลคัลโฮสต์ถูกรันอยู่บนแต่ละเป้าหมายแบบรีโมต โดยใช้แฟล็ก `-e` หาก `-e` ถูกระบุไว้ `command_list` คือชื่อสคริปต์และอาร์กิวเมนต์กับสคริปต์ ตัวอย่างเช่น: `dsh -e[flags] script_filename [arguments]...` ไฟล์ `script_filename` ถูกคัดลอกไปยังชื่อไฟล์สุ่ม ในไดเรกทอรี `/tmp` บนแต่ละเป้าหมายแบบรีโมต จากนั้นรันบนเป้าหมาย ถ้าพารามิเตอร์ `command_list` ไม่ถูกระบุ `dsh` จะสุ่มโหมตบรรทัดรับคำสั่ง แบบโต้ตอบ และแสดงพร้อม `dsh>` ป้อนคำสั่งที่พร้อม `dsh>` โดยใช้ไวยากรณ์ต่อไปนี้: `[!] "command"` โดย `command` คือคำสั่งที่จะรันบนปลายทางรีโมต เครื่องหมายอุทธาน (!) ที่นำหน้าคำสั่งเป็นสาเหตุทำให้คำสั่งรันอยู่บน โลคัลโฮสต์เท่านั้น และไม่ได้อยู่บนเป้าหมายแบบรีโมต คำสั่ง `dsh` รันแต่ละคำสั่งกับ แต่ละปลายทางที่ระบุ ผลลัพธ์ถูกแสดง และส่งคืนไปยัง พร้อม `dsh>` เมื่อต้องการออกจากโหมดบรรทัดรับคำสั่ง ป้อน `exit` ที่พร้อม `dsh>` คำสั่ง `dsh` ไม่ทำงานกับคำสั่งแบบโต้ตอบใดๆ ซึ่งประกอบด้วยคำสั่งเหล่านั้นที่อ่านจาก อินพุตมาตรฐาน

REMOTE USER

`user_ID` ที่ต้องการใช้สำหรับ เป้าหมายแบบรีโมตสามารถระบุไว้เป็นส่วนหนึ่งของไวยากรณ์เป้าหมาย (โปรดดู Target lists) หรือใช้แฟล็ก `-I` (ตัวพิมพ์เล็ก L) หากทั้งสองเมธอดถูกใช้ `user_ID` ถูกกำหนดไว้ดังต่อไปนี้:

1. สำหรับปลายทางที่ระบุเป็น `user_ID@target`, `user_ID` ถูกใช้สำหรับการเรียกใช้งานรีโมตบนปลายทาง และแฟล็ก `-I` ถูกข้าม
2. สำหรับปลายทางที่ไม่ได้ระบุโดยใช้ `user_ID@target` จะใช้ `user_ID` สำหรับการเรียกใช้งานรีโมตบนปลายทางโดยพิจารณา ดังนี้: `-user_ID` ที่ระบุด้วยแฟล็ก `-I` หากไม่ได้ระบุ `-I` ไว้ ผู้ใช้ปัจจุบันจะรันคำสั่ง หากระบุ `user_ID` ไว้สำหรับโลคัลโฮสต์ถูกรวม อยู่ในรายการเป้าหมาย คำสั่ง `remote shell` รัน `command_list` บนโลคัลโฮสต์เพื่อให้เห็นใจว่า ล็อกอินปลอดภัย

REMOTE SHELL  
ENVIRONMENT

สภาวะแวดล้อม shell ที่ใช้บนเป้าหมายแบบรีโมต ดีฟอลต์ไปยัง shell ซึ่งนิยามไว้สำหรับ `user_ID` ถูกใช้สำหรับการเรียกทำงานคำสั่งแบบรีโมต ไวยากรณ์คำสั่งที่ใช้สำหรับการเรียกทำงานคำสั่งแบบรีโมต สามารถระบุไว้โดยใช้แฟล็ก `-S` หากไม่ได้ระบุ `-S` ไว้ ไวยากรณ์จะดีฟอลต์ไปเป็น ไวยากรณ์ `ksh` เมื่อคำสั่งถูกรันบนเป้าหมายแบบรีโมต พาธที่ถูกกำหนดไว้โดยตัวแปรสภาวะแวดล้อม `DSH_PATH` ซึ่งนิยามอยู่ใน shell ของผู้ใช้ปัจจุบัน หากไม่ได้ตั้งค่า `DSH_PATH` ไว้ พาธที่ถูกใช้คือพาธดีฟอลต์ `remote shell` ตัวอย่างเช่น หากต้องการตั้งค่าโลคัลพาธสำหรับเป้าหมายแบบรีโมต ให้ใช้: `DSH_PATH=$PATH` แฟล็ก `-E` เลิกซ์พอร์ตไฟล์นิยามสภาวะแวดล้อมแบบโลคัล ไปยังแต่ละเป้าหมายแบบรีโมต ตัวแปรสภาวะแวดล้อมที่ระบุในไฟล์นี้ ถูกนิยามไว้ในสภาวะแวดล้อม `remote shell` ก่อนที่ `command_list` จะรัน

COMMAND  
EXECUTION

การประมวลผลคำสั่ง `remote shell` พร้อมกันสูงสุด (fanout) ถูกระบุไว้กับแฟล็ก `-f` หรือกับตัวแปรสภาวะแวดล้อม `DSH_FANOUT` fanout ถูกจำกัดไว้โดยจำนวนของคำสั่ง `remote shell` ที่สามารถรันแบบขนานได้ คุณอาจมีประสบการณ์กับค่า `DSH_FANOUT` บนเซิร์ฟเวอร์การจัดการของคุณ เพื่อดูค่าที่สูงกว่าที่เหมาะสม ค่าหมดเวลาใช้งานสำหรับการเรียกทำงานคำสั่งแบบรีโมตสามารถระบุได้พร้อมทั้งแฟล็ก `-t` หรือกับตัวแปรสภาวะแวดล้อม `DSH_TIMEOUT` หากเป้าหมายแบบรีโมตใดๆ ไม่ได้จัดเตรียมเอาต์พุตให้กับเอาต์พุตมาตรฐาน หรือข้อผิดพลาดมาตรฐานภายในค่าหมดเวลา ข้อความแสดงข้อผิดพลาดถูกแสดง โดยคำสั่ง `dsh` และออก หากโหมดการสตรีมถูกระบุไว้กับแฟล็ก `-s` เอาต์พุตจะถูกส่งคืน เนื่องจากกลับสู่สภาพพร้อมใช้งานจากแต่ละเป้าหมาย กระบวนการนี้ไม่ได้ออให้ `command_list` เสร็จสิ้นบนเป้าหมายทั้งหมด ก่อนที่จะส่งคืนเอาต์พุต ซึ่งสามารถปรับปรุงผลการทำงานได้ แต่เป็นสาเหตุทำให้ เอาต์พุตไม่ได้ถูกเรียงลำดับ แฟล็ก `-z` ถูกใช้ เพื่อแสดงได้ค้อออกจากคำสั่งล่าสุดที่ใช้บนโหมดแบบรีโมตใน `command_list`

**หมายเหตุ:** OpenSSH ส่งคืนสถานะออกของแบบรีโมตล่าสุดที่ใช้เป็นสถานะออกซึ่งจะมีผลต่อลักษณะการทำงานของ `dsh` และต้องการใช้แฟล็ก `-c` หากคำสั่งถูกใช้บนโหมดแบบรีโมต ถูกรันอยู่ในส่วนหลัง คำสั่งต้องไม่แสดงสถานะออก แฟล็ก `-m` มอนิเตอร์การเรียกทำงานของคำสั่ง `dsh` โดยพิมพ์ข้อความสถานะไปยังเอาต์พุตมาตรฐาน ข้อความสถานะแต่ละข้อความนำหน้าด้วย `dsh>` แฟล็ก `-T` จัดเตรียมข้อมูลการติดตาม การวินิจฉัยสำหรับการเรียกทำงานของคำสั่ง `dsh` ค่ากำหนดดีฟอลต์ และคำสั่ง `remote shell` จริงที่รันบน เป้าหมายแบบรีโมตถูกแสดง ซึ่งไม่ได้จัดเตรียมการตรวจหาข้อผิดพลาดหรือกลไกการกู้คืน สำหรับเป้าหมายแบบรีโมต คำสั่ง `dsh` เอาต์พุตไปยังข้อผิดพลาดมาตรฐาน และเอาต์พุตมาตรฐานสามารถวิเคราะห์เพื่อกำหนดคอร์สของการดำเนินการที่เหมาะสม ในโหมดการโต้ตอบ หากคำสั่งไม่สามารถรันไดบน เป้าหมายแบบรีโมต (ตัวอย่างเช่น คำสั่ง `remote shell`) ส่งผลลัพธ์ของโค้ดส่งคืนที่ไม่ใช่ศูนย์) คำสั่งต่อมา ไม่ได้ถูกส่งไปยังโหมดนี้บนการเรียกของคำสั่ง `dsh` นี้ เว้นเสียแต่ว่าระบุแฟล็ก `-c` ไว้

ไอเท็ม COMMAND OUTPUT	คำอธิบาย คำสั่ง <code>dsh</code> จะรอเพื่อแสดงเอาต์พุตจากแต่ละกระบวนการ <code>remote shell</code> จากนั้น เริ่มต้นกระบวนการ <code>remote shell</code> ใหม่ ลักษณะการทำงานดีฟอลต์นั้นถูกเขียนทับโดย แฟล็ก <code>-s</code> เอาต์พุตคำสั่ง <code>dsh</code> ประกอบด้วย ข้อผิดพลาดมาตรฐานและเอาต์พุตมาตรฐานจากคำสั่งแบบรีโมต เอาต์พุตมาตรฐาน <code>dsh</code> คือเอาต์พุตมาตรฐานจากคำสั่ง <code>remote shell</code> ข้อผิดพลาดมาตรฐาน <code>dsh</code> คือข้อผิดพลาดมาตรฐานจากคำสั่ง <code>remote shell</code> แต่ละบรรทัดถูกนำหน้าด้วยชื่อโฮสต์ของโหนด ที่สร้างเอาต์พุต ชื่อโฮสต์จะตามด้วย: อักขระและบรรทัดเอาต์พุตคำสั่ง ตัวกรองที่ต้องการแสดงเอาต์พุตเฉพาะ ที่จัดกลุ่มตามโหนดจะถูกแสดงแยกต่างหาก โปรดดูคำสั่ง <code>dshbak</code> สำหรับข้อมูลเพิ่มเติม เอาต์พุตสำหรับแต่ละเป้าหมาย สามารถคัดลอกไปยังไฟล์ได้โดยใช้แฟล็ก <code>-F output_path</code> เอาต์พุตมาตรฐานสำหรับแต่ละเป้าหมาย ถูกเขียนไปยังไฟล์ <code>target.output</code> ในไดเรกทอรี <code>output_path</code> และข้อผิดพลาดมาตรฐานสำหรับ แต่ละเป้าหมายถูกเขียนไปยังไฟล์ <code>target.error</code> ในไดเรกทอรี <code>output_path</code> แฟล็ก <code>-F</code> ไม่ได้ระงับเอาต์พุตบนคอนโซล คำสั่งสามารถรันแบบ <code>silent</code> ได้โดยใช้แฟล็ก <code>-Q</code> ไม่มีเอาต์พุตจากเอาต์พุตมาตรฐานหรือข้อผิดพลาดมาตรฐานของ 'เป้าหมาย' ถูกแสดง หากระบุแฟล็ก <code>-F</code> เอาต์พุตจะดำเนินการเขียนลงในไฟล์เอาต์พุตต่อไป
REPORTING	เอาต์พุตจากคำสั่ง <code>dsh</code> สามารถบันทึกไว้ใน รายงานบนโลคัลโฮสต์ แฟล็ก <code>--report report_path</code> เปิดใช้งานการสร้างรายงานไปยังไดเรกทอรี <code>report_path</code> ที่ระบุไว้ การรายงาน ถูกเรียกทำงานโดยนิยามตัวแปรสภาวะแวดล้อม <code>DSH_REPORT</code> กับ <code>report_path</code> <code>--report flag</code> เขียนทับตัวแปรสภาวะแวดล้อม <code>DSH_REPORT</code> แฟล็ก <code>--report-name</code> นิยามชื่อรายงาน หากเรียกทำงานการสร้างรายงาน <code>report name</code> ยังเป็นไดเรกทอรีย่อยของ <code>report_path</code> ที่มีไฟล์รายงาน ดัชนีตัวเลขถูกต่อท้ายกับชื่อไดเรกทอรีย่อย เพื่อยอมให้มีรายงานจำนวนมากที่มีชื่อเดียวกัน หากไม่ได้ใช้แฟล็ก <code>--report-name</code> ชื่อจะดีฟอลต์ไปเป็น Unspecified ไฟล์รายงานสรุป HTML และ XML จะถูกสร้างขึ้น นอกเหนือจากไฟล์ผลลัพธ์ XML SIGNALS: Signal 2 (INT), Signal 3 (QUIT) และ Signal 15 (TERM) ถูกแพร่ไปยังคำสั่งที่เรียกทำงานบน เป้าหมายแบบรีโมต Signal 19 (CONT), Signal 17 (STOP) และ Signal 18 (TSTP) ดีฟอลต์เป็น <code>dsh</code> คำสั่ง <code>dsh</code> จะตอบกลับสัญญาณเหล่านี้ตามปกติ แต่สัญญาณไม่กระทบกับการเรียกทำงานคำสั่ง แบบรีโมต สัญญาณอื่นๆ ถูกกำหนดไว้โดย <code>dsh</code> และมีผลกับคำสั่ง <code>dsh</code> การประมวลผล <code>child</code> ปัจจุบันทั้งหมดผ่านการแพร่ไปยังการรันคำสั่งแบบรีโมตถูกยกเลิก (SIGTERM) พารามิเตอร์ <code>command_list</code> ระบุรายการของคำสั่ง เพื่อเรียกทำงานบนเป้าหมายแบบรีโมต ไวยากรณ์สำหรับพารามิเตอร์ <code>command_list</code> มีดังต่อไปนี้: " <code>command[; command...</code> "

## คีย์เวิร์ด

ไอเท็ม -a	คำอธิบาย รวมในเป้าหมายที่แสดงโหนดทั้งหมดที่ถูกนิยามใน คอนเท็กซ์ดีฟอลต์ คอนเท็กซ์ดีฟอลต์สามารถตั้งค่าได้โดยใช้แฟล็ก <code>-C</code> หรือตัวแปรสภาวะแวดล้อม <code>DSH_CONTEXT</code>
-A	รวมในเป้าหมายที่แสดงอุปกรณ์ทั้งหมดที่ถูกนิยามใน คอนเท็กซ์ดีฟอลต์ คอนเท็กซ์ดีฟอลต์สามารถตั้งค่าได้โดยใช้แฟล็ก <code>-C</code> หรือตัวแปรสภาวะแวดล้อม <code>DSH_CONTEXT</code> แฟล็กนี้ถูกปิดใช้งานบน HMCs
--all-nodescontext_list	รวมในเป้าหมายที่แสดงโหนดทั้งหมดซึ่งถูกนิยามในคอนเท็กซ์ ที่แสดงอยู่ใน <code>context_list</code> คอนเท็กซ์ดีฟอลต์ไม่ได้รวมอยู่ในรายการนี้ แฟล็กนี้ถูกปิดใช้งานบน HMC
--all-devices context_list	รวมในเป้าหมายที่แสดงอุปกรณ์ทั้งหมดซึ่งถูกนิยามในคอนเท็กซ์ ที่แสดงอยู่ใน <code>context_list</code> คอนเท็กซ์ดีฟอลต์ไม่ได้รวมอยู่ในรายการนี้ แฟล็กนี้ถูกปิดใช้งานบน HMCs
-C   --continue	ในโหมดแบบโต้ตอบเท่านั้น ให้เก็บโหนดไว้ในรายการเป้าหมาย แม้ว่าคำสั่ง <code>remote shell</code> สำหรับโฮสต์จะมีค่าออกที่เป็นค่าที่ไม่ใช่ศูนย์
-C   --contextcontext	คอนเท็กซ์ดีฟอลต์ที่ต้องการใช้เมื่อแก้ปัญหาที่เกิดขึ้นกับชื่อเป้าหมาย ค่าคอนเท็กซ์ต้องสอดคล้องกับโมดูลส่วนขยายคอนเท็กซ์ที่ถูกต้อง ในไดเรกทอรี <code>/opt/ibm/sysmgmt/dsm/pm/Context</code> ตัวอย่างเช่น ไฟล์ <code>/opt/ibm/sysmgmt/dsm/pm/Context/DSH.pm</code> คือโมดูลสำหรับคอนเท็กซ์ DSH
-d   --devices device_list	ระบุรายชื่อเป้าหมายของอุปกรณ์ เพื่อรวมในรายชื่อเป้าหมาย ไวยากรณ์ <code>device_list</code> คือ:  [context:][user_ID@]device_name[, [context:]\n [user_ID@]device_name]...  แฟล็กนี้ถูกปิดใช้งานบน HMCs
--device-rshdevice_remote_shell	ระบุพารามิเตอร์ของคำสั่ง <code>remote shell</code> ที่ถูกใช้เพื่อเรียกทำงานคำสั่งแบบรีโมตบนเป้าหมายของอุปกรณ์ <code>remote shell</code> สำหรับคอนเท็กซ์ที่ระบุเฉพาะสามารถนิยามได้โดยรวม <code>context</code> : ก่อนพาร ไวยากรณ์ <code>device_remote_shell</code> คือ:  [context:]path[, [context:]path]... แฟล็กนี้ ถูกปิดใช้งานบน HMCs -
-D   --devicegroups devicegroups	รวมในเป้าหมายที่แสดงอุปกรณ์ทั้งหมดซึ่งถูกนิยามอยู่ใน กลุ่มอุปกรณ์ที่ระบุอยู่ในรายการ <code>devicegroups</code> ไวยากรณ์ <code>devicegroups</code> คือ:  [context:] [user_ID@]devicegroup[, [context:]\n [user_ID@]devicegroup]... แฟล็กนี้ถูกปิดใช้งานบน HMCs

**ไอลเอ็ม**

**-e | --execute** คำอธิบาย  
 บังชี้ว่า *command\_list* ระบุชื่อไฟล์สคริปต์แบบโลคัลและอาร์กิวเมนต์ที่ต้องถูกเรียกทำงานบนเป้าหมายแบบรีโมต ไฟล์สคริปต์ถูกคัดลอกไปยังเป้าหมายแบบรีโมต จากนั้นเรียกทำงานแบบรีโมตด้วยอาร์กิวเมนต์ที่กำหนดไว้ ตัวแปรสถานะแวดล้อม *DSH\_NODE\_RCP* และ *DSH\_DEVICE\_RCP* ระบุคำสั่ง remote copy ที่ต้องการใช้เพื่อคัดลอกไฟล์สคริปต์ไปยังโหนด และเป้าหมายของอุปกรณ์ ตามลำดับ

**-E | --environment environment\_file** ระบุว่า *environment\_file* มีนิยามตัวแปรสถานะแวดล้อมที่ต้องการเอ็กซ์พอร์ตไปยังเป้าหมายก่อนที่จะเรียกทำงาน *command\_list* ตัวแปรสถานะแวดล้อม *DSH\_NODE\_RCP* และ *DSH\_DEVICE\_RCP* ระบุคำสั่ง remote copy ที่ต้องการใช้เพื่อเอ็กซ์พอร์ตไฟล์ไปยังโหนดและเป้าหมายของอุปกรณ์ ตามลำดับ

**-f | --fanout fanout\_value** ระบุค่า fanout สำหรับจำนวนสูงสุดของการเรียกทำงานการประมวลผล remote shell แบบพร้อมเพียงกัน การเรียกทำงานตามลำดับ สามารถระบุได้โดยบ่งชี้ถึงค่า fanout ที่เป็นค่า 1 หากไม่ได้ระบุ -f ไว้ ค่า fanout ที่เป็นค่าดีฟอลต์ของ 64 จะถูกใช้

**-F | --output output\_path** คัดลอกเอาต์พุตมาตรฐานไปยัง *output\_path/target\_name.output* และข้อผิดพลาดมาตรฐานไปยัง *output\_path/target\_name.error* เอาต์พุตยังคงถูกส่งไปยังเอาต์พุตมาตรฐาน หรือข้อผิดพลาดมาตรฐาน ใช้แฟล็ก -Q เพื่อระงับเอาต์พุตมาตรฐาน และข้อผิดพลาดมาตรฐาน

**-i | --notify** บังชี้ว่า เป้าหมายไม่ได้ตอบสนอง และพร้อมเพื่อดำเนินการเรียกทำงานแบบรีโมตต่อไปสำหรับเป้าหมาย

**-l (lowercase L) | --user user\_ID** ระบุชื่อผู้ใช้แบบรีโมตที่ต้องการใช้ สำหรับการเรียกทำงานคำสั่งแบบรีโมต

**-h | --help** แสดงข้อมูลการใช้งานคำสั่ง

**-n | --nodes node\_list** ระบุรายชื่อเป้าหมายของโหนดเพื่อรวม ในรายชื่อเป้าหมาย ไวยากรณ์ *node\_list* คือ:

[context:] [user\_ID@]node\_name[, [context:] \

[user\_ID@]node\_name]...

**-L | --no-locale** ระบุเพื่อไม่ให้เอ็กซ์พอร์ตนิยามโลคัลของโลคัลไฮสตร์ไปยังเป้าหมายแบบรีโมต นิยามโลคัลของโลคัลไฮสตร์ถูกเอ็กซ์พอร์ตตามค่าดีฟอลต์ไปยังแต่ละเป้าหมายแบบรีโมต เอาต์พุตถูกต่อท้าย ไฟล์สำหรับแต่ละการเรียกทำงานของคำสั่ง dsh

**--log log\_file** เปิดใช้งานการบันทึกไปยัง *log\_file* ที่ระบุเฉพาะ

**-m | --monitor** มอนิเตอร์การเรียกทำงาน remote shell โดยแสดงข้อความสถานะ ในระหว่างการเรียกทำงานบนแต่ละเป้าหมายรวมในเป้าหมายที่แสดงโหนดทั้งหมดซึ่งถูกนิยามอยู่ใน กลุ่มของโหนดที่ระบุอยู่ในรายการ *nodegroups* ไวยากรณ์ของ *nodegroups* คือ:

[context:] [user\_ID@] nodegroup [, [context:] \

[user\_ID@]nodegroup]...

**-o --node-options node\_options** ระบุอ็อปชันที่ต้องการส่งไปยัง คำสั่ง remote shell สำหรับเป้าหมายของโหนด อ็อปชันต้องถูกระบุอยู่ ภายในเครื่องหมายอัญประกาศคู่ ("" ) เพื่อแยกอ็อปชันเหล่านั้นออกจากอ็อปชัน dsh อ็อปชันสำหรับโหนดในคอนเท็กซ์ที่ระบุเฉพาะสามารถนิยามได้โดยรวม context: ก่อนรายการอ็อปชัน ไวยากรณ์สำหรับ *node\_options* คือ:

[context:] " options "[, [context:] " options" ]...

**-O | --device-options device\_options** ระบุอ็อปชันที่ต้องการส่งไปยัง คำสั่ง remote shell สำหรับเป้าหมายของอุปกรณ์ อ็อปชันต้องถูกระบุอยู่ภายในเครื่องหมายอัญประกาศคู่เพื่อแยกอ็อปชันเหล่านั้นออกจากอ็อปชัน dsh อ็อปชันสำหรับอุปกรณ์ในคอนเท็กซ์ที่ระบุเฉพาะสามารถนิยามได้โดย รวม context: ก่อนรายการอ็อปชัน ไวยากรณ์สำหรับ *device\_options* คือ

[context:] " options "[, [context:] " options" ]...แฟล็กนี้ถูกปิดใช้งานบน HMCs

**-Q | --silent** ระบุโหมด silent ไม่มีเอาต์พุตเป้าหมายที่เขียนไปยัง เอาต์พุตมาตรฐานหรือข้อผิดพลาดมาตรฐาน ข้อความการมอนิเตอร์ ถูกเขียนไปยังเอาต์พุตมาตรฐาน

**-q | --show-config** แสดงค่าติดตั้งสถานะแวดล้อมปัจจุบัน ที่เกี่ยวข้องกับคำสั่ง DSH Utilities ทั้งหมด ซึ่งรวมแทรกค่าของตัวแปรสถานะแวดล้อมทั้งหมด และค่าติดตั้งสำหรับคอนเท็กซ์ที่ติดตั้งไว้และคอนเท็กซ์ที่ถูกต้องในปัจจุบัน แต่ละค่าติดตั้งจะมีคอนเท็กซ์นำหน้า: เพื่อระบุ คอนเท็กซ์ต้นฉบับของค่าติดตั้ง

**-r | --node-rsh node\_remote\_copy** ระบุพารามิเตอร์ของ คำสั่ง remote shell ที่ถูกใช้เพื่อคัดลอกไฟล์ไปหรือจากเป้าหมายของโหนด คำสั่ง remote shell สำหรับคอนเท็กซ์ที่ระบุเฉพาะ สามารถถูกนิยามได้โดยรวม context: ก่อนพารามิเตอร์ *node\_remote\_copy* คือ:

[context:] [path][, [context:] [path]]... หาก พารามิเตอร์ rsync พารามิเตอร์นั้นจะสมมติว่า คำสั่ง rsync ดำเนินการกับ remote copy

**--report report\_path** เปิดใช้งานการสร้างรายงานและระบุ พารามิเตอร์ไปยังไดเรกทอรีที่บันทึกรายงาน --report-name report\_name ระบุชื่อที่ต้องการใช้เมื่อสร้าง รายงาน หากไม่ได้ระบุไว้ ชื่อจะดีฟอลต์เป็น Unspecified แฟล็กนี้ สามารถใช้ได้กับแฟล็ก

**--report**

**-s | --stream** ระบุเพื่อส่งคืนเอาต์พุตซึ่งกลับมารวมใช้งาน จากแต่ละเป้าหมาย ซึ่งจะไม้ออให้ *command\_list* เสร็จสิ้นบนเป้าหมายก่อนที่จะส่งคืนเอาต์พุต

## ไอเอ็ม

-S|--syntax csh | ksh  
-t|--timeout timeout

-T|--trace

-v|--verify

-X env\_list

-V|--version

## คำอธิบาย

ระบุไวยากรณ์ shell ที่ต้องการใช้บน เป้าหมายแบบรีโมต หากไม่ได้ระบุไว้ ไวยากรณ์ ksh จะถูกนำมาใช้แทน ระยะเวลาในหน่วยวินาที เพื่อรอเอาต์พุตจากการเรียกทำงานเป้าหมายแบบรีโมตแบบพร้อมเพียงกัน หากไม่มี เอาต์พุต ที่พร้อมใช้งานจากเป้าหมายใดๆ ในช่วงของค่าหมดเวลาใช้งานที่ระบุไว้ ข้อความแสดงข้อผิดพลาดจะ แสดงขึ้นโดยคำสั่ง dsh และยกเลิกการเรียกทำงานกับเป้าหมายแบบรีโมต ซึ่งลมเหลวในการตอบกลับ หากไม่ได้ ระบุค่าหมดเวลาใช้งานไว้ dsh จะรอเพื่อดำเนินการประมวลผลเอาต์พุตจากเป้าหมายแบบรีโมตทั้งหมดต่อไป เมื่อระบุด้วยแฟล็ก -i ผู้ใช้จะได้รับพร้อมท์ สำหรับช่วงหมดเวลาใช้งานเพิ่มเติมเพื่อรอเอาต์พุต เปิดใช้งานโหมดการติดตาม คำสั่ง dsh จะพิมพ์ข้อความวินิจฉัย ไปยังเอาต์พุตมาตรฐานในระหว่างการเรียกทำงาน กับแต่ละเป้าหมาย

ตรวจสอบแต่ละเป้าหมายก่อนที่จะเรียกทำงาน คำสั่งแบบรีโมตใดๆ บนเป้าหมาย หากเป้าหมายไม่ตอบสนอง การเรียกทำงาน ของคำสั่งแบบรีโมตสำหรับเป้าหมายจะถูกยกเลิก เมื่อระบุด้วยแฟล็ก -i ผู้ใช้จะได้รับพร้อมท์เพื่อ ลองการร้องขอการตรวจสอบความถูกต้องอีกครั้ง

ข้ามตัวแปรสภาวะแวดล้อม dsh อีพซันนี้ ใช้อาร์กิวเมนต์ที่เป็นรายชื่อตัวแปรสภาวะแวดล้อมที่ค้นด้วยเครื่อง หมายคอมมา ซึ่งต้องไม่ถูกข้าม หากไม่มีอาร์กิวเมนต์ในอีพซันนี้ หรืออาร์กิวเมนต์คือสตริงว่างเปล่า ตัวแปร สภาวะแวดล้อม dsh ทั้งหมดจะถูกข้าม แฟล็กนี้ไม่สามารถระบุเป็นแฟล็กสุดท้ายได้

แสดงข้อมูลเวอร์ชันของคำสั่ง dsh

## ไอเอ็ม

-z|--exit-status

## คำอธิบาย

แสดงสถานะออกสำหรับคำสั่งที่ไม่ใช่เซ็กโครันซ์ซึ่งเรียกทำงานแบบรีโมตล่าสุด บนแต่ละเป้าหมาย หากคำสั่งที่ใช้ บนโหนดแบบรีโมต ถูกรันอยู่ในส่วนหลัง สถานะออก จะไม่ถูกแสดง ค่า Exit Status Exit สำหรับการเรียกทำงาน remote shell แต่ละครั้งถูกแสดงอยู่ในข้อความจากคำสั่ง dsh หากค่าออกของ remote shell ไม่ใช่ค่าศูนย์ โค้ดส่งคืนที่ ไม่ใช่ค่าศูนย์จาก remote shell บ่งชี้ว่า เกิดข้อผิดพลาดขึ้นใน remote shell โค้ดส่งคืนนี้ ไม่เกี่ยวข้องกับโค้ดออกของคํ าสั่งที่ใช้งานแบบรีโมต หาก remote shell พบข้อผิดพลาด การเรียกทำงานของคำสั่งแบบรีโมต บนเป้าหมายจะถูกส่ง ผ่าน โค้ดออกของคำสั่ง dsh มีค่า 0 หากคำสั่งที่เรียกทำงานโดยไม่มีข้อผิดพลาดและคำสั่ง remote shell ทั้งหมดเสร็จ ลิน ด้วยโค้ดออก 0

หากเกิดข้อผิดพลาดของ dsh ภายในหรือคำสั่ง remote shell ไม่เสร็จสิ้น ค่าออกของคำสั่ง dsh จะมากกว่า 0 ค่าออกจะ ถูกเพิ่มขึ้น 1 สำหรับแต่ละอินสแตนซ์ ของการเรียกทำงานคำสั่งแบบรีโมตที่ไม่เป็นผลสำเร็จ หากคำสั่งที่ใช้งานแบบรี โมต ถูกรันอยู่ในส่วนหลัง โค้ดออกของคำสั่งที่ใช้งานแบบรีโมต คือ 0 ตัวแปรสภาวะแวดล้อม DSH\_CONTEXT ระบุ คอนเท็กซ์ที่ฟอลต์ที่ต้องการใช้เมื่อแก้ปัญหาเกี่ยวกับเป้าหมาย ตัวแปรนี้ ถูกเขียนทับโดยแฟล็ก -C

**DSH\_DEVICE\_LIST** ระบุไฟล์ที่มีรายการเป้าหมายของอุปกรณ์

ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -d ตัวแปรสภาวะแวดล้อมนี้ถูกข้ามบน HMCs **DSH\_DEVICE\_OPTS** ระบุอีพซันที่ ต้องการใช้สำหรับคำสั่ง remote shell กับเป้าหมายของอุปกรณ์ เท่านั้น ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -O ตัวแปร สภาวะแวดล้อมนี้ถูกข้ามบน HMCs **DSH\_DEVICE\_RCP** ระบุพารามิเตอร์ของคำสั่ง remote copy ที่ถูกใช้เพื่อคัดลอกสคริปต์ไคล และไฟล์คอนฟิกูเรชันสภาวะแวดล้อมแบบไคลกับเป้าหมายของอุปกรณ์

ตัวแปรสภาวะแวดล้อมนี้ถูกข้ามบน HMCs **DSH\_DEVICE\_RSH** ระบุพารามิเตอร์ของ remote shell ที่ต้องการใช้สำหรับ การเรียกทำงานคำสั่งแบบรีโมตบนเป้าหมายของอุปกรณ์ ตัวแปรนี้ ถูกเขียนทับโดยแฟล็ก --device-rsh ตัวแปร สภาวะแวดล้อมนี้ถูกข้ามบน HMCs **DSH\_ENVIRONMENT** ระบุไฟล์ที่มีนิยามตัวแปรสภาวะแวดล้อมที่ต้องการ เอ็กซ์พอร์ต ไปยังเป้าหมายก่อนที่จะเรียกทำงานคำสั่งแบบรีโมต ตัวแปรนี้ถูกเขียนทับ โดยแฟล็ก -E **DSH\_FANOUT** ระบุค่า fanout

ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -f **DSH\_LOG** ระบุพารามิเตอร์ของไฟล์ที่ต้องการใช้ เพื่อบันทึกการทำงาน ตัวแปรนี้ถูก เขียนทับโดยแฟล็ก --log **DSH\_NODE\_LIST** ระบุไฟล์ที่มีรายการของ เป้าหมายของโหนด ตัวแปร

**DSH\_NODE\_LIST** โดแทนที่ WCOLL.HeI **DSH\_NODE\_OPTS** ระบุอีพซันที่ต้องการใช้ สำหรับคำสั่ง remote shell ที่มีเป้าหมายของโหนดเท่านั้น ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -o **DSH\_NODE\_RCP** ระบุพารามิเตอร์ของคำสั่ง remote copy ที่ต้องการใช้เพื่อคัดลอกสคริปต์ไคลและไฟล์คอนฟิกูเรชันสภาวะแวดล้อม แบบไคล ไปยังเป้าหมายของโหนด **DSH\_NODE\_RSH** ระบุพารามิเตอร์ของ remote shell ที่ต้องการใช้สำหรับการเรียก ทำงานคำสั่งแบบรีโมต บนเป้าหมายของโหนด ตัวแปรนี้ถูกแทนที่ด้วยแฟล็ก -r **DSH\_NODEGROUP\_PATH** ระบุ รายการไตรีกทอรี ที่ค้นด้วยเครื่องหมายโคลอนซึ่งมีไฟล์กลุ่มสำหรับคอนเท็กซ์ DSH เมื่อระบุแฟล็ก -a ไว้ใน คอนเท็กซ์ DSH รายชื่อโหนดที่ไม่ซ้ำกันจะถูกรวบรวมจากไฟล์กลุ่มของโหนดทั้งหมด ในพาร **DSH\_OUTPUT** ระบุชื่อไฟล์พื้นฐานสำหรับเอาต์พุตมาตรฐาน และข้อผิดพลาดมาตรฐานที่คัดลอก เอาต์พุตยังคงถูกส่งไปยังเอาต์พุต มาตรฐาน หรือข้อผิดพลาดมาตรฐาน ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก -F **DSH\_PATH** ตั้งค่าพารามิเตอร์ของคำสั่งเพื่อใช้บน เป้าหมาย หากไม่ได้ตั้งค่า **DSH\_PATH** ไว้ พารตึฟอลต์ที่นิยามไว้ในโปรไฟล์ของ user\_ID แบบรีโมตจะถูกใช้แทน **DSH\_PATH** ไม่สามารถใช้เพื่อรันคำสั่ง dsh ไปยัง HMC **DSH\_REPORT**

## ไอเท็ม

### คำอธิบาย

เปิดใช้งานการสร้างรายงานเมื่อตั้งค่าพารามิเตอร์ ของไต่เร็กทอรีที่บันทึกรายงาน ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก `--report DSH_SYNTAX` ระบุไวยากรณ์ shell ที่ต้องการใช้บนเป้าหมายแบบรีโมต เช่น ksh หรือ csh หากไม่ได้ระบุไว้ ไวยากรณ์ ksh จะถูกสมมติขึ้น ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก `-S DSH_TIMEOUT` ระบุเวลาในหน่วยวินาทีเพื่อรอเอาต์พุตจากแต่ละเป้าหมายแบบรีโมต ตัวแปรนี้ถูกเขียนทับโดยแฟล็ก `-t` ความปลอดภัย คำสั่ง dsh ไม่มีข้อกำหนด คอนฟิกูเรชันความปลอดภัย ข้อกำหนดความปลอดภัยของคำสั่งแบบรีโมตทั้งหมด - คอนฟิกูเรชัน การพิสูจน์ตัวตน และการให้สิทธิ์ - จะถูกกำหนดไว้โดยคำสั่งแบบรีโมตที่สำคัญ ที่กำหนดคอนฟิกูเรชันไว้สำหรับ dsh คำสั่งอนุมัติว่าการพิสูจน์ตัวตน และการให้สิทธิ์จะถูกกำหนดคอนฟิกูเรชันระหว่างไคลโฮสต์และ เป้าหมายแบบรีโมต การพร้อมตรัสผ่านการโต้ตอบไม่ได้รับการสนับสนุน ข้อผิดพลาดจะถูกแสดง และการเรียกทำงานจะถูกส่งผ่าน สำหรับเป้าหมายแบบรีโมต หากมีพร้อมแสดงหรือหากการพิสูจน์ตัวตนหรือการให้สิทธิ์กับเป้าหมายแบบรีโมต อย่างไม่อย่างหนึ่งล้มเหลว คอนฟิกูเรชันความปลอดภัยตามที่เกี่ยวข้องกับ สภาวะแวดล้อมแบบรีโมตและคำสั่ง remote shell เป็นแบบผู้ใช้กำหนดเอง เมื่อคำสั่งแบบรีโมตถูกกำหนดคอนฟิกูเรชันไว้เป็น `/usr/bin/rsh` และคำสั่งนี้ถูกกำหนดค่าเพื่อใช้ Kerberos เวอร์ชัน 5 คุณต้องรัน Kerberos คำสั่ง `kinit` เป็นอันโบแรกเพื่อขอรับตั๋ว ตัวการให้สิทธิ์ และคุณต้องมั่นใจว่า Kerberos หลักของคุณอยู่ในไฟล์ `k5login` ในไต่เร็กทอรีโฮมของผู้ใช้แบบรีโมตบนเป้าหมาย

## ตัวอย่าง

1. หากต้องการรันคำสั่ง `ps` บนเป้าหมายของโหนด `node1` และ `node2` ให้ป้อน:  

```
dsh -n node1,node2 "ps"
```
2. หากต้องการรันคำสั่ง `ps` บนเป้าหมายของโหนดแต่ละโหนดที่แสดงอยู่ในไฟล์ `myhosts` ให้ป้อน:  

```
DSH_NODE_LIST=./myhosts; dsh ps
```
3. หากต้องการป้อนคำสั่งในโหมดแบบโต้ตอบสำหรับการเรียกทำงานบนเป้าหมายของโหนด ที่นิยามอยู่ใน `NodeGroup1` ให้ป้อน:  

```
dsh -N NodeGroup1
```
4. หากต้องการแสดงจำนวนของผู้ใช้บนโหนด NIM Managed ทั้งหมด และในกลุ่มโหนดคอนเท็กซ์ `DSH NodeGroup2` ให้ป้อน:  

```
dsh --all-nodes NIM -N DSH:NodeGroup2 "who | wc -l"
```
5. หากต้องการป้อนรายการของเป้าหมายของโหนดและเป้าหมายของอุปกรณ์แบบโต้ตอบ จากนั้นเรียกทำงานคำสั่ง `date` ในโหมดแบบโต้ตอบ ให้ป้อน:  

```
dsh -n - -d -
```

อินพุต และเอาต์พุตเพิ่มเติมจะคล้ายกับที่แสดงต่อไปนี้:

```
dsh node> node1
dsh node> gregb@node2
dsh node>
dsh device> CSM:kathyc@device1
dsh device>
dsh> date node1: Wed Apr 13 17:15:59 EDT 2005
gregb@node2: Wed Apr 13 17:15:59 EDT 2005
kathyc@device1: Wed Apr 13 17:15:59 EDT 2005
dsh> exit #
```

6. หากต้องการรันคำสั่ง `ls` บนโหนดทั้งหมดใน คลัสเตอร์และข้ามตัวแปรสภาวะแวดล้อม dsh ทั้งหมด ให้ป้อน:  

```
dsh -X -a ls
```
7. หากต้องการรันคำสั่ง `ps` บน `node1` และข้ามตัวแปรสภาวะแวดล้อม dsh ทั้งหมด ยกเว้น `DSH_NODE_OPTS` ให้ป้อน:  

```
dsh -n node1 -X 'DSH_NODE_OPTS' ps
```

---

## คำสั่ง `dsldapaccept`

### วัตถุประสงค์

รับการร้องขอคิวการพิมพ์สำหรับระบบการพิมพ์ System V ที่เปิดใช้ไดเร็กทอรี

### ไวยากรณ์

`dsldapaccept PrintQueueName`

### คำอธิบาย

คำสั่ง `dsldapaccept` และ `dsldapreject` ถูกใช้เพื่อเซตคิวงานพิมพ์ เพื่อที่ยอมรับ หรือปฏิเสธการร้องขอการพิมพ์ที่อยู่ในคิวไม่เหมือนกับคำสั่ง `accept` และ `reject` คำสั่ง `directory-enabled` สามารถควบคุมระบบพิมพ์รีโมต ตรวจจับที่เป็น `directory-enabled` เนื่องจากเป็นการเขียนโดยตรง ไปที่อ็อบเจกต์คิวงานพิมพ์บนไดเร็กทอรีเซิร์ฟเวอร์

ผู้ใช้คำสั่งนี้ต้องเป็น `directory-enabled` และมีชุดสิทธิในการ `write`, `modify`, `search` และ `read` กับไดเร็กทอรีในบริบทไดเร็กทอรี ซึ่งผู้ใช้เป็นผู้ดูแลระบบ

### พารามิเตอร์

ไอเท็ม

`PrintQueueName`

คำอธิบาย

พารามิเตอร์ `PrintQueueName` คือ distinguished name (RDN) สัมพันธ์ของอ็อบเจกต์คิวการพิมพ์ ชื่อคิวงานพิมพ์หลายชื่อ ถูกระบุได้ใน รายการที่ค้นด้วยคอมมา

### สถานะออก

- 0 หมายถึงสำเร็จ
- 1 หมายถึงอ็อบชันไม่ถูกต้อง
- 2 หมายถึงไม่รู้จักคิวงานพิมพ์ที่ระบุ
- 3 หมายถึงผู้ใช้นี้ไม่มีสิทธิ `modify`
- 4 ระบุว่าค่า RDN ที่ไม่ถูกต้องถูกระบุ
- 5 หมายถึงค่าถูกเซตแล้ว
- 6 หมายถึงคำสั่งไม่สามารถติดต่อกับไดเร็กทอรีเซิร์ฟเวอร์ได้
- 7 หมายถึงข้อผิดพลาดอื่น

### ตัวอย่าง

1. เมื่อต้องการเซตคิวงานพิมพ์ "hpcolor" ให้ยอมรับการร้องขอ:

```
dsldapaccept hpcolor
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dsldapadmin`” ในหน้า 218

“คำสั่ง `dslopenable`” ในหน้า 223

“คำสั่ง `dslpsearch`” ในหน้า 227

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `lpstat`

---

## คำสั่ง `dslpaccess`

### วัตถุประสงค์

อนุญาตหรือปฏิเสธผู้ใช้ที่เปิดใช้ non-directory และการเข้าถึงระบบกับคิวงานพิมพ์ สำหรับระบบย่อย System V print

### ไวยากรณ์

```
dslpaccess -q QueueName -a AllowList | -d DenyList
```

### คำอธิบาย

คำสั่ง `dslpaccess` อนุญาตหรือปฏิเสธผู้ใช้ และระบบ เข้าถึงคิวงานพิมพ์ directory-enabled โดยถูกโมเดลบนอ็อปชันของคำสั่ง `lpadmin` คืออ็อปชัน `-u`

รายการอนุญาตและปฏิเสธประกอบด้วยรายการ entry ที่คั่นด้วยคอมมา แต่ละ รายการอาจระบุ ล็อกอิน ID หรือชื่อระบบและ ล็อกอิน ID ดังนี้:

```
[[LoginID]|[System!LoginID]], [[LoginID]|[System!Login-ID]], ...
```

`LoginID` หรือ `System` หรือทั้งคู่ สามารถถูกใช้เป็นตัว wildcard `all`, อนุญาตหรือปฏิเสธรายการที่จัดสรรทั้งหมด ใช้ `all` ด้วยความระมัดระวัง เมื่อ entry `all` ถูกเพิ่มในหนึ่งรายการ entry ทั้งหมดที่ไม่ใช่ `all` จะถูกลบ ออกจากรายการ, สำหรับค่าที่เหมาะสมของ `LoginID` หรือ `System` ดีพอลต์สำหรับ `System` คือ local host

ผู้ใช้คำสั่งนี้ต้องเป็น directory-enabled และมีชุดสิทธิ ในการ write, modify, search และ read กับไดเรกทอรีในบริบทไดเรกทอรี ซึ่งผู้ใช้เป็นผู้ดูแลระบบ

### แฟล็ก

#### ไอเท็ม

`-a AllowList`

`-d DenyList`

`-q QueueName`

#### คำอธิบาย

ระบุรายการของผู้ใช้เพื่อเพิ่มลงในรายการอนุญาต ถ้ามีอยู่ ผู้ใช้จะถูกลบออกจากรายการปฏิเสธ อ็อปชันนี้ใช้ไม่ได้กับ อ็อปชัน `-d` ระบุรายการของผู้ใช้เพื่อเพิ่มลงในรายการปฏิเสธ ถ้ามีอยู่ ผู้ใช้จะถูกลบออกจากรายการอนุญาต อ็อปชันนี้ใช้ไม่ได้กับ อ็อปชัน `-a` พารามิเตอร์ `queue-name` เป็น Relative Distinguished Name (RDN) ของคิวงานพิมพ์ ถ้าชื่อคิวงานพิมพ์ไม่มีอยู่ในบริบทไดเรกทอรี คำสั่งจะ ลมเหลว

### สถานะออก

- 0 หมายถึงสำเร็จ
- 1 หมายถึงอ็อปชันไม่ถูกต้อง
- 2 หมายถึงไม่รู้จักคิวงานพิมพ์ที่ระบุ

- 3 หมายถึงผู้ใช้ไม่มีสิทธิค่าควบคุมการเข้าถึงที่เหมาะสม
- 4 ระบุว่าค่าRDNที่ไม่ถูกต้องถูกระบุ
- 5 หมายถึงค่าถูกเซตแล้ว
- 6 หมายถึงข้อผิดพลาดอื่น

## ตัวอย่าง

1. คำสั่งต่อไปนี้ให้สิทธิการเข้าถึง fredb แก่ผู้ใช้กับคิวงานพิมพ์ printq1 บนโฮสต์ systemX:  
`dsldapaccess -q printq1 -a systemX!fredb`
2. คำสั่งต่อไปนี้ปฏิเสธการเข้าถึงคิวงานพิมพ์ printq1 กับผู้ใช้ tomt สำหรับ โฮสต์ทั้งหมด:  
`dsldapaccess -q printq1 -d all!tomt`

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dsldapprotocol” ในหน้า 224

“คำสั่ง dsldapreject” ในหน้า 226

“คำสั่ง dsldapsearch” ในหน้า 227

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง lpstat

## คำสั่ง dsldapadmin

### วัตถุประสงค์

ปรับแต่งเซิร์ฟเวอร์การพิมพ์ directory-enabled สำหรับ System V print subsystem

### ไวยากรณ์

```
dsldapadmin [ [-q PrintQueueName [-D Queueคำอธิบาย] [-n LocalQueueName] [-o banner | nobanner] [-A mail | none]
] [-F FaultRecovery] [-P PhysicalPrinterName] [-s NetworkEntityName] ] [-P PhysicalPrinterName [-T
PrinterType] [-I Location] [-L PDLLList] [-q PrintQueueName -P PhysicalPrinterName [-I ContentType] [-i
InterfaceScript] [-m [ Standard | PS ] ] [-o PrintOptions] ] [-q PrintQueueName [-I ContentType] ] [-q
PrintQueueName -s NetworkEntityName [-a PrintSystemDNSName | PrinterSystemAddress] [-t BSD | HPNP ]]
```

```
dsldapadmin [ -q PrintQueueName [-u PhysicalPrinterName] [-U ObjectRDN ]]
```

```
dsldapadmin [ -x PrintQueueName] [-X PhysicalPrinterName] [-r NetworkEntityName ]
```

```
dsldapadmin [ -h ]
```

### คำอธิบาย

คำสั่ง dsldapadmin ถูกใช้เพื่อดำเนินฟังก์ชันต่อไปนี้ เพื่อปรับแต่งเซิร์ฟเวอร์การพิมพ์ directory-enabled:

- เพิ่มคิวงานพิมพ์และฟิสิกส์พรินเตอร์ให้กับระบบ

- ดัดแปลงคิวงานพิมพ์และฟิลิ์คัลพรินเตอร์
- ลบคิวงานพิมพ์และฟิลิ์คัลพรินเตอร์จากระบบ
- เพิ่มและลบอ็อบเจ็กต์ entity เน็ตเวิร์กสำหรับเน็ตเวิร์กพรินเตอร์

คำสั่ง `dsldapadmin` จัดเตรียมเวอร์ชัน directory-aware ของการทำงานที่ระบุโดย `lpadmin` (ซึ่งไม่ใช่ directory-aware), และดำเนินการใช้ระบบคอนฟิกูเรชัน ``flat file`` ตามปกติ หมายเหตุ เมื่อทั้งสองระบบถูกใช้อยู่ ระบบย่อยพรินเตอร์ใช้ข้อมูลที่อยู่ในรายการไดเร็กทอรี เป็นความรับผิดชอบของผู้ดูแลระบบ ในการรับประกันว่าจะไม่มีการกำหนดชื่อที่ขัดแย้งกันเกิดขึ้นระหว่างระบบคอนฟิกูเรชัน สองระบบ

คำสั่ง directory-enabled ใช้ Relative Distinguished Names (RDNs), แทน Distinguished Names (DNs) ตัวอย่างเช่น เมื่อต้องการสร้างคิวที่เปิดใช้โดยไดเร็กทอรีด้วย DN ของ "cn=test,ou=printq,ou=print,cn=aixdata" RDN "test" เท่านั้นที่จะถูกใช้สำหรับ `PrintQueueName`

เมื่อปรับแต่งคิวงานพิมพ์ ซึ่งผู้ดูแลระบบไม่ได้อยู่ในระบบ เพื่อโฮสต์คิวงานพิมพ์ พารามิเตอร์ `InterfaceScript` ของ `-i` และพารามิเตอร์ `PrinterType` ของ `-T` ไม่ถูกเลือก เนื่องจากระบบรีโมต ไม่สามารถถูกเข้าถึงเพื่อเลือกได้ ดังนั้นจึงเป็นหน้าที่ของผู้ดูแลระบบ เพื่อตรวจสอบว่า `InterfaceScript` และ `PrinterType` ที่ระบุมีอยู่บนระบบ ที่เป็นโฮสต์รีโมต

บรรทัดคำสั่งมี การใช้ร่วมกันของแฟล็ก `-q`, `-P` และ `-s` ได้, หรือ การรวมกันของแฟล็ก `-x`, `-X` และ `-r`, แต่เฉพาะหนึ่งแฟล็กของแต่ละแฟล็กเท่านั้น When multiple directory objects are simultaneously created or modified, appropriate links are set up between the three object types (printers, print queues and network entities).

## แฟล็ก

### ไอเท็ม

`-a PrinterSystemDNSName | PrinterSystemAddress`

`-A [ mail | none ]`

`-D Queue` คำอธิบาย

### คำอธิบาย

เชื่อมโยงชื่อ DNS หรือเน็ตเวิร์กแอดเดรสกับระบบ ถ้าอาร์กิวเมนต์ที่กำหนดสามารถถูกแปลงเป็นแอดเดรส IPv4 หรือ IPv6 ได้ จะเป็น แอดเดรส ถ้าไม่ได้จะถือว่าเป็นชื่อ DNS แฟล็ก `-a` ทำให้เน็ตเวิร์ก entity อ็อบเจ็กต์ที่ระบุโดย `-s` ถูกแก้ไข หรือถูกสร้างขึ้น ถ้ายังไม่มีอยู่ ผู้ดูแลระบบควรตรวจสอบ ว่าเน็ตเวิร์ก entity อ็อบเจ็กต์ถูกกำหนดชื่อเฉพาะ เพื่อหลีกเลี่ยงการแก้ไข อ็อบเจ็กต์ระบบ UNIX ที่มีอยู่ แทนที่จะเป็นการเพิ่มอ็อบเจ็กต์ระบบการพิมพ์ใหม่ แฟล็กนี้ต้องการแฟล็ก `-s` สั่งให้ระบบการพิมพ์สร้างข้อความเมล ถ้าการร้องขอการพิมพ์ล้มเหลว เมลถูกส่งไปที่เจ้าของฟิลิ์คัลพรินเตอร์ หรือไปที่ผู้ใช้ root ของระบบที่โฮสต์คิวงานพิมพ์ ถ้าพรินเตอร์ ไม่มีเจ้าของหรือผู้ใช้ไม่มีเมลแอดเดรส ค่าดีฟอลต์คือ `none` แฟล็กนี้ต้องการแฟล็ก `-q` กำหนดหมายเหตุรายละเอียดสำหรับอ็อบเจ็กต์คิวงานพิมพ์ ที่ระบุด้วยแฟล็ก `-q` รายละเอียดนี้ถูกแสดงเมื่อผู้ใช้ขอรายละเอียดแบบเต็มของคิวงานพิมพ์โดยใช้คำสั่ง `lpstat` สตริงที่มีช่องว่างควรรีไ้อัญประกาศ แฟล็กนี้ต้องการแฟล็ก `-q`

## ไอเท็ม

-F *FaultRecovery*

### คำอธิบาย

กำหนดมาตรการกู้คืนความผิดพลาดของคิวงานพิมพ์ แฟล็กนี้ ระบุการกู้คืนที่จะถูกใช้ถ้าพริเตอร์ในคิวงานพิมพ์ล้มเหลว ขณะพิมพ์การร้องขอการพิมพ์ ค่าของ *FaultRecovery* มีค่าได้ดังต่อไปนี้:

**continue** ทำการพิมพ์บนที่หน้าบนสุดที่การพิมพ์หยุดทำงานไป ซึ่ง ต้องการตัวกรองเพื่อรอให้ความผิดพลาดถูกเคลียร์ก่อนดำเนินต่อโดยอัตโนมัติ

### beginning

เริ่มการพิมพ์ตามการร้องขออีกครั้งจากจุดเริ่มต้น

**wait** ปิดใช้งานการพิมพ์บน *PhysicalPrinterName* และ รอผู้ดูแลระบบหรือผู้ใช้ให้เปิดใช้งานการพิมพ์อีกครั้ง

ระหว่าง รอผู้ดูแลระบบหรือผู้ใช้ ซึ่งส่งการร้องขอการพิมพ์ที่หยุดไป สามารถส่งการร้องขอการเปลี่ยนแปลงที่ระบุจุดที่ควรพิมพ์ต่อ ถ้า ไม่มีการร้องขอเปลี่ยนแปลง ก่อนการพิมพ์ถูกเปิดใช้ การพิมพ์จะดำเนินต่อที่ ด้านบนของเพจที่หยุด ถ้าตัวกรองอนุญาต มิฉะนั้นการร้องขอ ถูกพิมพ์จากจุดเริ่มต้น

ค่าดีฟอลต์ของ *FaultRecovery* คือ **beginning** แฟล็กนี้ต้องการแฟล็ก -q

แสดงจอแสดงผลวิธีใช้แบบย่อ

ชื่อพารสำหรับ *InterfaceScript* พริเตอร์เมื่อเข้าถึงผ่านคิวงานพิมพ์ที่ระบุ แฟล็กนี้ใช้ไม่ได้ ถ้าแฟล็ก -P ไม่ถูกระบุ อินเตอร์เฟซ สคริปต์ปกติถูกกำหนดโดยผู้ใช้ แฟล็กนี้ไม่สามารถใช้ได้เมื่อระบุ -m แฟล็กนี้ต้องการแฟล็ก -q และ -P

ระบุชนิดเนื้อหาของคิวงานพิมพ์ อนุญาตให้ คิวงานพิมพ์จัดการการร้องขอการพิมพ์ที่มีชนิดเนื้อหาในรายการ ถ้า รายการมีมากกว่าหนึ่งพารามิเตอร์ *ContentType*, พารามิเตอร์ *ContentType* ต้องถูกแยกโดยคอมมา ดูที่ หน้าในคู่มือ *lpadmin* สำหรับรายละเอียดแบบสมบูรณ์ของ รูปแบบ และยัง

ต้องการแฟล็ก -P และแฟล็ก -q

กำหนดตำแหน่งของพริเตอร์ นี้เป็นสตริงระบุ ตำแหน่งที่พริเตอร์อยู่ทางกายภาพ ตัวอย่าง "Building X, Room 6" สามารถ ถูกค้นหาได้โดยคำสั่ง *dsldapsearch* เมื่อเซต คำนึงถูกเขียนทับได้เท่านั้น ไม่ใช้การลบออก แฟล็กนี้ต้องการแฟล็ก -P

ระบุรายการของ Page Description Languages (PDLs) ที่สนับสนุนโดยพริเตอร์ ซึ่งถูกใช้เพื่อแจ้งการสนับสนุนพริเตอร์ PDL, สามารถค้นหาได้โดยใช้คำสั่ง *dsldapsearch* สนับสนุน AUTOSW, PCL, PCLXL, POSTSCRIPT, TEXT, ESCP, PJI, SIMPLE, และ OTHER PDLs ถ้าแฟล็ก -L ถูกใช้เพื่อ

แก้ไขฟิลิ์คพริเตอร์อ็อบเจกต์ที่มีอยู่ รายการ จะแทนที่รายการที่มีอยู่ แฟล็กนี้ต้องการแฟล็ก -P โมเดลอินเตอร์เฟซโปรแกรมสำหรับพริเตอร์เมื่อเข้าถึง ผ่านคิวงานพิมพ์จำเพาะ มีการเลือกโมเดลอินเตอร์เฟซสคริปต์ที่จะ ใช้โดยคิวงานพิมพ์ เมื่อฟิลิ์คพริเตอร์อ็อบเจกต์ถูกสร้าง และไม่มีการระบุทั้งแฟล็ก -m หรือ -i ค่าดีฟอลต์คือ *standard* แฟล็กนี้ไม่สามารถใช้ได้เมื่อระบุ -i แฟล็กนี้ต้องการทั้งแฟล็ก -q และ -P

กำหนดชื่อไลคัลของคิวงานพิมพ์ โดยปกติ ชื่อนี้แตกต่างจาก RDN ของคิวต่อเมื่อคิวอยู่บนโฮสต์ที่ไม่ได้เปิดใช้งานแบบ ไดร็กทอรี ซึ่งถูกใช้โดยการเชื่อมต่อเน็ตเวิร์กโมเดิตาเขา เพื่อระบุคิวงานพิมพ์ บนระบบที่รับ ค่าดีฟอลต์คือ RDN ของคิวการพิมพ์ แฟล็กนี้ต้องการแฟล็ก -q

กำหนดไว้ ถ้าเพจแถบป้ายจะถูกสร้างโดย คิวงานพิมพ์นี้เสมอ ค่าดีฟอลต์, *banner*, บังคับ ให้เพจแถบป้ายถูกพิมพ์สำหรับการร้องขอการพิมพ์ทั้งหมด โดยที่ *nobanner* อนุญาตให้ผู้ใช้ส่งงานพิมพ์ที่ระบุว่าไม่มีเพจแถบป้าย ที่จะถูกพิมพ์ แฟล็กนี้ต้องการแฟล็ก -q

ระบุค่าสำหรับอ็อบชันการพิมพ์ ดูที่เอกสาร *lpadmin* สำหรับคำอธิบายละเอียดของอ็อบชันการพิมพ์ที่ใช้ได้กับแฟล็ก -o แฟล็กนี้ต้องการทั้งแฟล็ก -q และ -P

สร้างหรือแก้ไขฟิลิ์คพริเตอร์อ็อบเจกต์ อาร์กิวเมนต์ *PhysicalPrinterName* ระบุ RDN ของอ็อบเจกต์เครื่องพิมพ์ ถ้าอ็อบเจกต์ไม่มีอยู่ *dsldapadmin* จะสร้าง ขึ้นมา

*dsldapadmin* สร้างหรือแก้ไข อ็อบเจกต์คิวงานพิมพ์ อาร์กิวเมนต์ *PrintQueueName* ระบุ RDN ของอ็อบเจกต์คิวการพิมพ์ เมื่อเพิ่มคิวงานพิมพ์ใหม่ คุณต้องระบุแฟล็ก -s และ -P เพื่อที่คำสั่งจะรู้จัก *NetworkEntityName* และ *PhysicalPrinterName* สำหรับคิวงานพิมพ์ที่ถูกเพิ่ม ถ้า อ็อบเจกต์คิวงานพิมพ์ไม่มีอยู่ *dsldapadmin* จะสร้าง ขึ้น

บรรทัดคำสั่งนี้ มีการใช้ร่วมกันของแฟล็ก -q, -P และ -s ได้, หรือ การรวมกันของแฟล็ก -x, -X และ -r, แต่เฉพาะหนึ่งแฟล็กของแต่ละแฟล็กเท่านั้น When multiple directory objects are simultaneously created or modified, appropriate links are set up between the three object types (printers, print queues and network entities).

-h

-i *InterfaceScript*

-I *ContentType*[, *ContentType*, ...]

-l *Location*

-L *PDL*[, *PDL*, ...]

-m [*standard* | *PS*]

-n *LocalQueueName*

-o [*banner* | *nobanner*]

-o *PrintOption*=*Value*[, ...]

-P *PhysicalPrinterName*

-q *PrintQueueName*

## ไอเอ็ม

-r *NetworkEntityName*

-s *NetworkEntityName*

-t [BSD|HPNP]

-T *PrinterType* [, *PrinterType*, ...]

-u *PhysicalPrinterName*

-U *ObjectRDN*

-x *PrintQueueName*

-X *PhysicalPrinterName*

## คำอธิบาย

ลบอ็อบเจกต์ระบบ entity เน็ตเวิร์ก ต้องใช้ความระมัดระวัง ที่จะไม่ลบอ็อบเจกต์ระบบ non-printer เป็นความรับผิดชอบของผู้ดูแลระบบ เพื่อประกันว่าอ็อบเจกต์ที่ถูกต้องถูกลบ ระบบอ็อบเจกต์ระบบ entity เน็ตเวิร์กที่โฮสต์ คิวงานพิมพ์ ถ้ามีการกำหนด -a อ็อบเจกต์ ถูกสร้างหรือแก้ไข อาร์กิวเมนต์ *NetworkEntityName* ระบบ RDN ของอ็อบเจกต์ในบริบท ไตรีกทอรีปัจจุบัน เน็ตเวิร์ก entity อ็อบเจกต์กำหนดเน็ตเวิร์กแอดเดรสที่รีโมตไคลเอ็นต์จำเป็นต้องใช้ในการเข้าถึงคิวงานพิมพ์ กำหนดโปรโตคอลการพิมพ์ที่ใช้โดยคิวงานพิมพ์ "networked printer" นี้ ค่าทำซ้ำและหมดเวลาใช้งาน ถูกเซตเป็นค่าดีฟอลต์สำหรับ เน็ตเวิร์กพริ้นเตอร์ เมื่อต้องการเปลี่ยนค่าเหล่านี้ ควรใช้คำสั่ง **dslpprotocol** หมายถึง แฟล็กนี้ควรใช้เฉพาะสำหรับ เน็ตเวิร์กพริ้นเตอร์ที่สนับสนุนโปรโตคอล BSD หรือ HPNP แฟล็กนี้ต้องการแฟล็ก -q รายการของชนิดพริ้นเตอร์ ระบบพริ้นเตอร์เป็น ชนิดพริ้นเตอร์ ตัวอย่าง "hplaserjet" ดูที่คู่มือ **lpadmin** สำหรับรายละเอียด แฟล็กนี้ต้องการแฟล็ก -P ยกเลิกลิงก์ฟิสิกัลพริ้นเตอร์ที่กำหนดชื่อจากคิวงานพิมพ์ (ระบบด้วยแฟล็ก -q) โดยไม่ต้องลบอ็อบเจกต์ แฟล็กนี้ต้องการแฟล็ก -q ยกเลิกลิงก์ฟิสิกัลพริ้นเตอร์หรืออ็อบเจกต์คิวงานพิมพ์ (ที่ระบุโดย *ObjectRDN*) จากคิวงานพิมพ์ (ที่ระบบด้วยแฟล็ก -q) โดยไม่ลบอ็อบเจกต์ แฟล็กนี้ต้องการแฟล็ก -q ลบอ็อบเจกต์คิวงานพิมพ์ ลบฟิสิกัลพริ้นเตอร์อ็อบเจกต์

## สถานะออก

0 หมายถึงสำเร็จ

255 (หรือ -1)

หมายถึงมีข้อผิดพลาดใน configuration ข้อความแสดงความผิดพลาดถูกแสดง เพื่ออธิบายข้อผิดพลาดหรือความล้มเหลว

## ตัวอย่าง

ตัวอย่างต่อไปนี้แสดงการใช้คำสั่ง **dslpadmin** เมื่อ ผู้ใช้เข้าสู่ระบบ directory-enabled UNIX

1. คำต่อไปนี้เพิ่มเครื่องพิมพ์เครือข่าย HP LaserJet ที่ใช้โปรโตคอลการพิมพ์รีโมต BSD กับ RDN คิวการพิมพ์ของ "denlj5n" และเครื่องพิมพ์ฟิสิกัล RDN ของ "denplj5n" โดยให้ข้อมูลแก่คิวงานพิมพ์เป็นรายละเอียดของ "HP JetDirect (PostScript)", ชนิดพริ้นเตอร์ "PS-b", และโมเดล อินเทอร์เน็ตเฟสสคริปต์เป็น "PS" พริ้นเตอร์มีเน็ตเวิร์กแอดเดรสเป็น "p\_hplj.ibm.com":

```
dslpadmin -q denlj5n -P denplj5n -T PS-b -D "HP JetDirect (Postscript)" \  
-I PS -m PS -A mail -o nobanner -s denslj5n -a p_hplj.ibm.com -t BSD
```

ระบบ การพิมพ์จะอนุญาตการร้องขอการพิมพ์ชนิดเนื้อหา PS สำหรับคิวงานพิมพ์นี้ และอนุญาตการปิดใช้เพจแถบป้าย

2. คำต่อไปนี้เพิ่มเครื่องพิมพ์เครือข่าย HP LaserJet PostScript โดยใช้โปรโตคอลการพิมพ์รีโมต HPNP กับ RDN คิวการพิมพ์เป็น "dehpnnp" และเครื่องพิมพ์ฟิสิกัล RDN เป็น "dehpnnp" โดยให้ข้อมูลแก่คิวงานพิมพ์เป็นรายละเอียดของ "HPNP (PCL)", ชนิดพริ้นเตอร์ "hplaserjet" และโมเดลอินเทอร์เน็ตเฟสสคริปต์เป็น "standard" พริ้นเตอร์มีเน็ตเวิร์กแอดเดรสเป็น "p\_hplj.ibm.com":

```
dslpadmin -q dehpnnp -P dehpnnp -T hplaserjet -D "HPNP (PCL)" -I pcl \  
-m standard -A mail -s deshpnnp -a p_hplj.ibm.com -t HPNP
```

ระบบ การพิมพ์จะอนุญาตการร้องขอการพิมพ์ชนิดเนื้อหา PCL สำหรับคิวงานพิมพ์นี้ และปฏิเสธการร้องขอ ถ้าไม่มีการร้องขอเพจแถบป้าย ถ้ามีความผิดพลาดของพริ้นเตอร์เกิดขึ้น ระบบการพิมพ์จะส่งเมลไปที่เจ้าของพริ้นเตอร์

3. คำสั่งต่อไปนี้ลบพริ้นเตอร์ HP LaserJet PostScript:

```
dsldapadmin -x delj5n -X deplj5n
```

#### 4. คำสั่งต่อไปนี้ลบพริเตอร์HPNP:

```
dsldapadmin -x dehnp -X dehpnp -r deshnp
```

#### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dsldapaccept” ในหน้า 216

“คำสั่ง dsldapaccess” ในหน้า 217

#### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง lpstat

---

## คำสั่ง dsldapdisable

### วัตถุประสงค์

ปิดใช้การร้องขอคิวงานพิมพ์สำหรับ System V print subsystem

### ไวยากรณ์

```
dsldapdisable [ -r Reason ] PrintQueueName
```

### คำอธิบาย

คำสั่ง dsldapenable และ dsldapdisable ถูกใช้เพื่อเปิดใช้หรือปิดใช้งานคิวงานพิมพ์ไม่ให้ประมวลผลการร้องขอการพิมพ์ที่ได้ถูกจัดคิวไว้ ไม่เหมือนกับคำสั่ง enable และ disable คำสั่ง directory-enabled สามารถควบคุมระบบพิมพ์รีโมต ตรวจจับที่เป็น directory-enabled เนื่องจากการเขียนโดยตรงไปที่อ็อบเจกต์คิวงานพิมพ์บนไดเร็กทอรีเซิร์ฟเวอร์

### แฟล็ก

ไอเท็ม  
-r Reason

คำอธิบาย  
กำหนดเหตุผลสำหรับการปิดใช้คิวงานพิมพ์ สตรีงที่มีช่องว่างควรใส่อัฒประกาศ

Reason เป็นสตรีงที่ถูกแสดงโดยคำสั่ง lpstat ไม่มีการเซตเหตุผล ตีฟอลต์เมื่อไม่มีการระบุไว้

### พารามิเตอร์

ไอเท็ม  
PrintQueueName

คำอธิบาย  
พารามิเตอร์ PrintQueueName คือ RDN ของคิวการพิมพ์ นี้ควรเป็นรายการของคิวงานพิมพ์ ถ้าชื่อคิวงานพิมพ์ไม่มีอยู่ในบริบทไดเร็กทอรี คำสั่งจะ ล้มเหลว

### สถานะออก

- 0 หมายถึงสำเร็จ
- 1 หมายถึงอ็อบชันไม่ถูกต้อง
- 2 หมายถึงไม่รู้จักรหัสคิวงานพิมพ์ที่ระบุ

- 3 หมายถึงผู้ใช้นี้ไม่มีสิทธิ modify
- 4 ระบุว่าค่า RDN ที่ไม่ถูกต้องถูกระบุ
- 5 หมายถึงค่าถูกเซตแล้ว
- 6 หมายถึงคำสั่งไม่สามารถติดต่อกับไดเรกทอรีเซิร์ฟเวอร์ได้
- 7 หมายถึงข้อผิดพลาดอื่น

## ตัวอย่าง

เมื่อต้องการปิดใช้งานคิวงานพิมพ์ "printer1", โดยระบุเหตุผล "routine maintenance", ให้ป้อนดังต่อไปนี้:

```
dsldapdisable -r "routine maintenance" printer1
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dsldapprotocol” ในหน้า 224

“คำสั่ง dsldapreject” ในหน้า 226

“คำสั่ง dsldapsearch” ในหน้า 227

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง lpstat

## คำสั่ง dsldapenable

### วัตถุประสงค์

เปิดใช้การร้องขอคิวงานพิมพ์สำหรับ System V print subsystem

### ไวยากรณ์

**dsldapenable** *PrintQueueName*

### คำอธิบาย

คำสั่ง **dsldapenable** และ **dsldapdisable** ถูกใช้เพื่อเปิดใช้หรือปิดใช้งานคิวงานพิมพ์ไม่ให้ประมวลผลการร้องขอการพิมพ์ ที่ได้ถูกจัดคิวไว้ ไม่เหมือนกับคำสั่ง **enable** และ **disable** คำสั่ง **directory-enabled** สามารถควบคุมระบบพิมพ์รีโมต トラบที่เป็น **directory-enabled** เนื่องจากการเขียนโดยตรงไปที่อ็อบเจกต์คิวงานพิมพ์บนไดเรกทอรีเซิร์ฟเวอร์

### พารามิเตอร์

ไอเท็ม  
*PrintQueueName*

คำอธิบาย

พารามิเตอร์ *PrintQueueName* คือ RDN ของคิวการพิมพ์ นี้ควรเป็นรายการของคิวงานพิมพ์ ถ้าชื่อคิวงานพิมพ์ไม่มีอยู่ในบริบทไดเรกทอรี คำสั่งจะล้มเหลว

## คำสั่งย่อย

### สถานะออก

- 0 หมายถึงสำเร็จ
- 1 หมายถึงอีพชันไม่ถูกต้อง
- 2 หมายถึงไม่รู้จักคิวงานพิมพ์ที่ระบุ
- 3 หมายถึงผู้ใช้นี้ไม่มีสิทธิ modify
- 4 ระบุว่าค่า RDN ที่ไม่ถูกต้องถูกระบุ
- 5 หมายถึงค่าถูกเซตแล้ว
- 6 หมายถึงคำสั่งไม่สามารถติดต่อกับไดเรกทอรีเซิร์ฟเวอร์ได้
- 7 หมายถึงข้อผิดพลาดอื่น

## ตัวอย่าง

1. เมื่อต้องการเปิดใช้คิวงานพิมพ์ "hpcolor" ให้ป้อนดังต่อไปนี้:

```
dslopenable hpcolor
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dslpadmin” ในหน้า 218

“คำสั่ง dslpprotocol”

“คำสั่ง dslpreject” ในหน้า 226

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง lpstat

---

## คำสั่ง dslpprotocol

### วัตถุประสงค์

ปรับแต่งโปรโตคอลการพิมพ์รีโมตของคิวงานพิมพ์สำหรับ System V print subsystem

### ไวยากรณ์

```
dslpprotocol -t RemoteProtocol [ -T TimeOut ] [ -R Retry ] [ -r PrintQueueName
```

```
dslpprotocol -l [ -S ] PrintQueueName
```

## คำอธิบาย

คำสั่ง `dspprotocol` ถูกใช้เพื่อปรับแต่ง "remote print protocol" ที่ไคลเอ็นต์การพิมพ์รีโมตสามารถใช้เมื่อส่งการร้องขอ การพิมพ์ไปที่คิวงานพิมพ์

ในการพิมพ์ directory-enabled เมื่อต้องการพิมพ์ไปที่คิวงานพิมพ์รีโมต ไคลเอ็นต์ต้องรับ โปรโตคอลการพิมพ์รีโมตที่สามารถใช้ได้มาก่อน ซึ่งรับได้จาก อ็อบเจ็กต์คิวงานพิมพ์ในไดเรกทอรี ซึ่งเป็นหนึ่งใน BSD และ HPNP หรือทั้งคู่ เมื่อมีโปรโตคอลถูกปรับแต่งมากกว่าหนึ่งโปรโตคอลสำหรับคิวงานพิมพ์ ระบบการพิมพ์ UNIX จะใช้ค่าแรกที่อ่าน ดังนั้นโดยปกติควรมีเพียง หนึ่งโปรโตคอลที่ตั้งค่าไว้

พารามิเตอร์ `PrintQueueName` เป็น Relative Distinguished Name (RDN) ของคิวการพิมพ์ ถ้าค่าที่กำหนดให้กับ `PrintQueueName` ไม่มีอยู่ คำสั่งจะล้มเหลว

ผู้ใช้คำสั่งนี้ต้องเป็น directory-enabled และมีชุดสิทธิในการ write, modify, search และ read กับไดเรกทอรีในบริบทไดเรกทอรี ซึ่งผู้ใช้เป็นผู้ดูแลระบบ

## แฟล็ก

ไอเท็ม	คำอธิบาย
-l	พิมพ์รายละเอียดของพารามิเตอร์โปรโตคอลการพิมพ์รีโมต ที่เชื่อมโยงกับคิวงานพิมพ์
-t <i>RemoteProtocol</i>	ระบุโปรโตคอลการพิมพ์รีโมตที่สามารถถูกใช้ เมื่อส่งการร้องขอการพิมพ์ไปที่คิวการพิมพ์นี้ ค่าชนิดโปรโตคอล ที่สนับสนุน มี <code>bsd</code> และ <code>hpnp</code> ค่าดีฟอลต์คือ <code>bsd</code>
-T <i>TimeOut</i>	เซตค่าหมดเวลาใช้งานการเชื่อมต่อเน็ตเวิร์กสำหรับ โปรโตคอลที่ระบุ คือเวลาที่การเชื่อมต่อเน็ตเวิร์กควรมีอยู่ในสถานะไม่ได้ทำงาน ก่อนการตัดการเชื่อมต่อ ค่า n ยังสามารถถูกระบุเพื่อ ปิดใช้การหมดเวลาใช้งาน ค่า 0 ทำให้การเชื่อมต่อหยุดทันทีที่อยู่ในสถานะไม่ได้ทำงาน ค่าดีฟอลต์คือ 10 นาที และไม่มีค่า จำกัดระดับบนในทางปฏิบัติ ดูที่เพจคู่มือ <code>Ipsystem</code> สำหรับนิยามแบบเต็มของอ็อปชัน -T
-r	อ็อปชันนี้ถูกใช้เพื่อลบโปรโตคอลที่ระบุออกจาก อ็อบเจ็กต์คิวงานพิมพ์ อ็อปชันนี้ต้องการให้มีการระบุอ็อปชัน -t เช่นกัน
-R <i>Retry</i>	เซตเวลาการทำการเชื่อมต่อเน็ตเวิร์กซ้ำสำหรับโปรโตคอล ที่ระบุ คือเวลาเป็นนาทีเพื่อรอ ก่อนที่จะพยายามทำการเชื่อมต่อเน็ตเวิร์กอีกครั้ง หลังความล้มเหลว ค่าดีฟอลต์คือ 2 นาที ค่า 0 ทำให้การเชื่อมต่อถูกทำซ้ำทันที หมายเหตุ ค่านี้ ต้องน้อยกว่าค่าหมดเวลาใช้งานที่ระบุโดยใช้อ็อปชัน -T ค่า n สามารถถูกระบุเพื่อป้องกันการหยุดลงของ การเชื่อมต่อที่ซ้ำซ้ำได้เช่นกัน เมื่อไม่มีงานอยู่ ไม่มีค่าจำกัดระดับบน ในทางปฏิบัติ สำหรับ "เน็ตเวิร์กพรีนเตอร์" เวลาทำซ้ำควรถูก เซตเป็น 0 ดูที่เพจคู่มือ <code>Ipsystem</code> สำหรับนิยาม แบบเต็มของอ็อปชัน -R
-S	ที่ให้กับอ็อปชัน -l เพื่อแสดง โปรโตคอลเซตอ็อปของคิวงานพิมพ์ในรูปแบบธรรมดา

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
<code>PrintQueueName</code>	พารามิเตอร์ <code>PrintQueueName</code> เป็น Relative Distinguished Name (RDN) ของคิวการพิมพ์ ถ้าค่าที่กำหนดให้กับ <code>PrintQueueName</code> ไม่มีอยู่ คำสั่งจะล้มเหลว

## สถานะออก

- 0 หมายถึงสำเร็จ
- 1 หมายถึงอ็อปชันไม่ถูกต้อง
- 2 หมายถึงไม่รู้จักคิวงานพิมพ์ที่ระบุ
- 3 หมายถึงผู้ใช้ไม่มีสิทธิ modify

- 4 ระบุว่าค่า RDN ที่ไม่ถูกต้องถูกระบุ
- 5 หมายถึงค่าถูกเซตแล้ว
- 6 หมายถึงข้อผิดพลาดอื่น

## ตัวอย่าง

1. เมื่อต้องการเซตคิวงานพิมพ์ "printq1" ให้อนุญาตโปรโตคอลการพิมพ์รีโมต BSD ให้อ่าน คำสั่งต่อไปนี้:

```
dsllpprotocol -t BSD printq1
```

2. เมื่อต้องการลบโปรโตคอล BSD จากคิวงานพิมพ์ "hpcolor" ให้อ่านดังต่อไปนี้:

```
dsllpprotocol -r -t BSD hpcolor
```

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dslpaccept” ในหน้า 216

“คำสั่ง dslpaccess” ในหน้า 217

“คำสั่ง dslpadmin” ในหน้า 218

“คำสั่ง dslpenable” ในหน้า 223

## คำสั่ง dslpreject

### วัตถุประสงค์

ปฏิเสธการร้องขอคิวการพิมพ์สำหรับระบบการพิมพ์ System V ที่เปิดใช้ไดเร็กทอรี

### ไวยากรณ์

```
dslpreject [ -r Reason ] PrintQueueName
```

### คำอธิบาย

คำสั่ง `dslpaccept` และ `dslpreject` ถูกใช้เพื่อเซตคิวงานพิมพ์ เพื่อที่ยอมรับ หรือปฏิเสธการร้องขอการพิมพ์ที่อยู่ในคิวไม่เหมือนกับคำสั่ง `accept` และ `reject` คำสั่ง `directory-enabled` สามารถควบคุมระบบพิมพ์รีโมต トラบที่เป็น `directory-enabled` เนื่องจากการเขียนโดยตรงไปที่อ็อบเจ็กต์คิวงานพิมพ์บนไดเร็กทอรีเซิร์ฟเวอร์ การร้องขอการพิมพ์ที่อยู่ในคิวแล้วจะไม่ได้รับผลจากคำสั่ง `dslpreject`

ผู้ใช้งานคำสั่งนี้ต้องเป็น `directory-enabled` และมีชุดสิทธิในการ `write`, `modify`, `search` และ `read` กับไดเร็กทอรีในบริบทไดเร็กทอรี ซึ่งผู้ใช้งานเป็นผู้ดูแลระบบ

### แฟล็ก

ไอเท็ม  
-r Reason

คำอธิบาย  
กำหนดเหตุผลสำหรับการปฏิเสธ สตริงมี ช่องว่างควรรออยู่ภายในอัญประกาศ Reason คือ สตริงที่ถูกแสดงโดยคำสั่ง  
Ipstat ไม่มีการเซตเหตุผล ตีพอลต์เมื่อไม่มีการระบุไว้

## พารามิเตอร์

ไอเท็ม  
PrintQueueName

คำอธิบาย  
พารามิเตอร์ PrintQueueName คือ RDN ของอ็อบเจกต์คิวการพิมพ์ ชื่อคิวงานพิมพ์หลายชื่อ ถูกระบุได้ใน รายการที่ค้น  
ด้วยคอมมา

## สถานะออก

- 0 หมายถึงสำเร็จ
- 1 หมายถึงอ็อบชันไม่ถูกต้อง
- 2 หมายถึงไม่รู้จักคิวงานพิมพ์ที่ระบุ
- 3 หมายถึงผู้ใช้นี้ไม่มีสิทธิ modify
- 4 ระบุว่าค่า RDN ที่ไม่ถูกต้องถูกระบุ
- 5 หมายถึงค่าถูกเซตแล้ว
- 6 หมายถึงคำสั่งไม่สามารถติดต่อกับไดเรกทอรีเซอร์วิสได้
- 7 หมายถึงข้อผิดพลาดอื่น

## ตัวอย่าง

1. เมื่อต้องการเซตคิวงานพิมพ์ให้ปฏิเสธการร้องขอ และระบุเหตุผลว่า no toner, ให้ป้อนดังต่อไปนี้:

```
dsldapreject -r "no toner" printer1
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dsldapdisable” ในหน้า 222

“คำสั่ง dsldapprotocol” ในหน้า 224

“คำสั่ง dsldapsearch”

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง lpstat

---

## คำสั่ง dsldapsearch

### วัตถุประสงค์

ค้นหาไดเรกทอรีเพื่อหาอ็อบเจกต์ระบบการพิมพ์ใน System V print subsystem

### ไวยากรณ์

```
dsldapsearch [-q [-p]] [-P] [-o SearchOptions]
```

## คำอธิบาย

คำสั่ง `dslpsearch` อนุญาตให้ผู้ใช้และผู้ดูแลระบบ ค้นหาไดเรกทอรีเพื่อหาอ็อบเจกต์ระบบการพิมพ์ ตัวอย่าง ผู้ใช้สามารถ ค้นหาพรินเตอร์ที่สามารถพิมพ์งานไฟล์ PostScript ได้ การใช้งานหลักของคำสั่งนี้ จะเป็นการค้นหาคิวงานพิมพ์ที่ตรงกับสตริงการค้นหา

คำสั่ง `dslpsearch` ส่งคืน Distinguished Name (DN) ของอ็อบเจกต์ที่ตรงกับสตริงการค้นหา อย่างไรก็ตาม ต้องการ Relative Distinguished Name (RDN) สำหรับใช้ใน คำสั่งที่ไดเรกทอรีอื่นเปิดใช้งาน ตัวอย่างเช่น ถ้า DN "cn=testqueue, ou=printq,ou=print,cn=aixdata" ถูกส่งคืน โดยคำสั่ง `dslpsearch` เฉพาะ RDN "testqueue" เท่านั้นที่ถูกใช้เพื่ออ้างอิงคิวการพิมพ์

## แฟล็ก

### ไอเท็ม

-q

#### คำอธิบาย

ค้นหาคิวงานพิมพ์ที่ตรงกับอ็อบเจกต์การค้นหา การค้นหาขึ้นในฟิลิคัลพรินเตอร์อ็อบเจกต์ แต่คิวงานพิมพ์ที่บริการพรินเตอร์เหล่านั้นถูกแสดง นี่เป็นชนิดการค้นหาดีฟอลต์ อ็อบชัน -q ไม่สามารถถูกระบุกับ -P

-p

อ็อบชันนี้ถูกใช้กับอ็อบชัน -q, และทำให้รายการของฟิลิคัลพรินเตอร์บริการคิวงานพิมพ์ถูก แสดงเช่นกัน

-P

ค้นหาฟิลิคัลพรินเตอร์ที่ตรงกับสตริงการค้นหา อ็อบชัน -P ไม่สามารถถูกระบุกับ -q

-o SearchOptions

อ็อบชันการค้นหาหลายอ็อบชันสามารถสร้างรายการที่ค้นหาด้วยคอมมาได แต่ละอ็อบชันอาจถูกใช้ดังนี้:

- Page Description Languages (PDLs) ดังต่อไปนี้: AUTOSW, PCL, PCLXL, POSTSCRIPT, TEXT, ESCP, PJJ, SIMPLE, OTHER

- ส่วนอำนวยความสะดวกพรินเตอร์ต่อไปนี้: COLOR, DUPLEX, TRAYS, FINISH

- ตำแหน่งฟิลิคัลพรินเตอร์ ที่ระบุโดย location=xxxxxxx หรือ location='aaaa bbbbbb'

- คำสตริงที่กำหนดโดย location= ถูกค้นหาด้วย wildcards ที่กำหนด ที่ตำแหน่งสิ้นสุดทั้งสองด้านของสตริง ดังนั้น location=Room1 จะค้นหาพรินเตอร์โดยมี "Room1" เป็นตำแหน่ง เช่น "Building X, Room1, Bay6" คำสตริงยังสามารถเป็น wildcards (\*) ที่ฝังตัวในสตริง เช่น location="Building X\*Bay6" คำตำแหน่งหลายคำใช้ OR ในการค้นหา

- บรรทัดคำสั่งต่อไปนี้มีสตริงการค้นหาที่ใช้ได้:

```
dslpsearch -q -o PCL,ESCP,location=room2,COLOR
```

```
dslpsearch -q -p -o "PS, location='Building 1, Room1', DUPLEX"
```

## สถานะออก

0 หมายถึงสำเร็จ

1 หมายถึงอ็อบชันไม่ถูกต้อง

2 หมายถึงการค้นหาในแผนยังไดเรกทอรีล้มเหลว

3 หมายถึงบริบทไดเรกทอรีไม่ถูกต้อง

4 หมายถึงคำสั่งไม่สามารถติดต่อกับไดเรกทอรีเซอวิสได้

## ตัวอย่าง

1. บรรทัดคำสั่งต่อไปนี้ค้นหาคิงานพิมพ์ที่ตรงกับ อีพซันการค้นหา:

```
dsllpsearch -q -o search-options
```

2. การค้นหาต่อไปนี้ค้นหาฟิลิคัลพรินเตอร์ที่ตรงกับอีพซัน การค้นหา:

```
dsllpsearch -P -o search-options
```

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dsllpaccept” ในหน้า 216

“คำสั่ง dsllpaccess” ในหน้า 217

“คำสั่ง dsllpadmin” ในหน้า 218

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง lpstat

---

## คำสั่ง dspcat

### วัตถุประสงค์

แสดงแค็ตตาล็อกข้อความ

### ไวยากรณ์

เมื่อต้องการแสดงข้อความในแค็ตตาล็อก

```
dspcat CatalogName [ SetNumber [ MessageNumber ] ]
```

เมื่อต้องการจัดรูปแบบเอาต์พุตสำหรับคำสั่ง gencat

```
dspcat -g CatalogName [ SetNumber ]
```

### คำอธิบาย

คำสั่ง **dspcat** แสดง ข้อความเฉพาะ ข้อความทั้งหมดในเซต หรือข้อความทั้งหมดในแค็ตตาล็อก คำสั่ง **dspcat** กำหนดข้อความไปที่เอาต์พุต มาตรฐาน

หมายเหตุ: คำสั่ง **dspcat** จะค้นหาไฟล์แค็ตตาล็อกภายใต้ตัวแปรสถานะแวดล้อม **NLSPATH** หากตั้งค่าแ็ตทริบิวต์ **LC\_FASTMSG** เป็น **False** ในสถานะแวดล้อม **โลแคล C** หรือ **POSIX**

แ็ตทริบิวต์ **LC\_FASTMSG** ระบุข้อความดีฟอลต์ที่ใช้สำหรับ **โลแคล C** และ **POSIX** และระบุว่าตัวแปรสถานะแวดล้อม **NLSPATH** จะถูกข้ามหรือไม่เมื่อตั้งค่าแ็ตทริบิวต์ **LC\_FASTMSG** เป็น **True**

ค่าดีฟอลต์สำหรับแ็ตทริบิวต์ **LC\_FASTMSG** คือ **True** ในพาธ **/etc/environment**

พารามิเตอร์ **CatalogName** ระบุ แค็ตตาล็อกข้อความ พารามิเตอร์ **SetNumber** ระบุ เซตในแค็ตตาล็อกที่ระบุโดยพารามิเตอร์ **CatalogName** พารามิเตอร์ **MessageNumber** ระบุข้อความ เฉพาะในเซตที่ระบุโดยพารามิเตอร์ **SetNumber** ถ้าคุณรวมพารา

มีเตอร์ทั้งสาม คำสั่ง `dspcat` แสดง ข้อความที่ระบุ ถ้าคุณไม่รวมพารามิเตอร์ `MessageNumber` คำสั่ง `dspcat` จะแสดงข้อความทั้งหมดในเซต ถ้าคุณระบุค่า `nonexistent` สำหรับพารามิเตอร์ `SetNumber` หรือ `MessageNumber` คำสั่ง `dspcat` แสดงข้อความแสดงความผิดพลาดและส่งกลับ คำส่งกลับที่ไม่ใช่ศูนย์ ถ้าคุณระบุเพียงพารามิเตอร์ `CatalogName` คำสั่ง `dspcat` แสดงข้อความทั้งหมด ในแค็ตตาล็อก คุณต้องรวมพารามิเตอร์ `SetNumber` ถ้าคุณรวมพารามิเตอร์ `MessageNumber`

คำสั่ง `dspcat` ใช้ตัวแปรสภาวะแวดล้อม `NLSPATH` และหมวดหมู่ `LC_MESSAGES` ในการค้นหาแค็ตตาล็อกข้อความที่ระบุ ถ้าคุณไม่ใช่ชื่อ / (slash) ในค่าของพารามิเตอร์ `CatalogName`

## แฟล็ก

ไอเท็ม คำอธิบาย

-g จัดรูปแบบเอาต์พุตที่จะถูกใช้เป็นอินพุตกับคำสั่ง `gencat` พารามิเตอร์ `MessageNumber` ใช้ไม่ได้ เมื่อคุณใช้แฟล็ก -g

## ตัวอย่าง

เมื่อต้องการแสดงหมายเลขข้อความ 2 ในหมายเลขเซต 1 ของไฟล์ `test.cat` ป้อน:

```
dspcat test.cat 1 2
```

## ไฟล์

ไอเท็ม

`/usr/bin/dspcat`

คำอธิบาย

มีคำสั่ง `dspcat`

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `gencat`” ในหน้า 673

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `runcat`

คำสั่ง `catopen`

Message Facility

---

## คำสั่ง `dspmsg`

### วัตถุประสงค์

แสดงข้อความที่เลือกจากแค็ตตาล็อกข้อความ

### ไวยากรณ์

```
dspmsg [ -s SetNumber ] CatalogName MessageNumber [ 'DefaultMessage' [ Arguments ] ]
```

### คำอธิบาย

คำสั่ง `dspmsg` แสดง ข้อความเฉพาะจากแค็ตตาล็อกข้อความที่สร้างด้วยคำสั่ง `gencat` หรือ ถ้าข้อความเรียกคืนไม่ได้ จะแสดงข้อความดีฟอลต์ ที่กำหนดเป็นพารามิเตอร์ให้กับคำสั่ง คำสั่ง `dspmsg` กำหนดข้อความไปที่เอาต์พุต มาตรฐาน คำสั่งนี้ถูกกำหนด เพื่อใช้ในเชลล์สคริปต์เป็นการแทนที่คำสั่ง `echo`

หมายเหตุ: คำสั่ง `dspmsg` ค้นหาแค็ตตาล็อกไฟล์ภายใต้ `NLSPATH` ถ้า `LC_FASTMSG` ถูกเซตเป็น `False` ในสภาวะแวดล้อม `C` หรือโลแคล `POSIX`

`LC_FASTMSG` ระบุว่าข้อความดีฟอลต์ถูกใช้สำหรับ `C` และโลแคล `POSIX` และ `NLSPATH` นั้นถูกละเว้นเมื่อ `LC_FASTMSG` ถูกเซตเป็น `True`

ค่าดีฟอลต์สำหรับ `LC_FASTMSG` จะเป็น `True` ใน `/etc/environment`

ตัวแปรสภาวะแวดล้อม `NLSPATH` และหมวดหมู่ `LC_MESSAGES` ถูกใช้เพื่อค้นหาแค็ตตาล็อกข้อความที่ระบุถ้าไม่มีอักขระ `/` (slash) ที่ใช้ในค่าของพารามิเตอร์ `CatalogName` ถ้าไม่พบแค็ตตาล็อกที่กำหนดชื่อโดยพารามิเตอร์ `CatalogName` หรือถ้าไม่พบข้อความที่กำหนดชื่อโดยพารามิเตอร์ `MessageNumber` (และค่า `SetNumber` ทางเลือก) ค่า `DefaultMessage` ที่ระบุจะถูกแสดง ถ้าไม่ระบุค่า `DefaultMessage` ข้อความแสดงความผิดพลาด `system-generated` จะถูกแสดง

คำสั่ง `dspmsg` อนุญาตให้ใช้ได้ถึงสิบสตริง อาร์กิวเมนต์เพื่อใช้แทนที่ลงในข้อความ ถ้ามี ค่ากำหนดการแปลงรูทึนย่อย `%s`, `%n$s`, `%ld` หรือ `%n$ld` `printf` อาร์กิวเมนต์ที่หายไป สำหรับค่ากำหนดการแปลงมีผลให้เกิดข้อความแสดงความผิดพลาด `dspmsg` อักขระหลักตัวควบคุมรูทึนย่อย `printf` ปกติ (ตัวอย่าง `\n`) เป็นที่รู้จัก

ขอแนะนำให้ใช้การใช้สตริงการจัดรูปแบบรูทึนย่อย `printf` ในแค็ตตาล็อก รูปแบบนี้จัดเตรียมสำหรับการแทรกอาร์กิวเมนต์ที่ถูกต้อง แม้ว่าสตริงการจัดรูปแบบในข้อความอยู่ในลำดับต่างจาก ข้อความดีฟอลต์ คุณต้องปิดข้อความดีฟอลต์ ในเครื่องหมายคำพูดเดี่ยว ถ้าใช้รูปแบบ `%n$s` สำหรับการแทรกข้อความ

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-s SetNumber</code>	ระบุหมายเลขเซตทางเลือก ค่าดีฟอลต์สำหรับตัวแปร <code>SetNumber</code> คือ 1

## ตัวอย่าง

เมื่อต้องการแสดงหมายเลขเซต 1, หมายเลขข้อความ 2 ของแค็ตตาล็อก `test.cat` ให้ป้อน:

```
dspmsg -s 1 test.cat 2 'message %s not found' 2
```

ถ้าไม่พบข้อความ `message 2 not found` จะถูกแสดง

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/dspmsg</code>	มีคำสั่ง <code>dspmsg</code>

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dspcat`” ในหน้า 229

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `mkcatdefs`

คำสั่ง `catclose`

Message Facility

---

## คำสั่ง dtaction

### วัตถุประสงค์

เรียกการดำเนินการ CDE ด้วยอาร์กิวเมนต์ที่ระบุ

### ไวยากรณ์

```
dtaction [-contextDir context_dir]\
```

```
[-execHost host_name] [-termOpts terminal_arguments]
```

```
[-user user_name] action_name
```

```
[action_arg] ...
```

### คำอธิบาย

คำสั่ง **dtaction** อนุญาตให้แอปพลิเคชันหรือเซลล์สคริปต์ ซึ่ง มิฉะนั้นแล้วจะไม่เชื่อมต่อกับสถานะแวดล้อมการพัฒนา CDE เพื่อเรียกขอ การเรียกขอการดำเนินการ

การดำเนินการที่เรียกว่า *action\_name* ถูกเรียกด้วย *action\_arg* ที่จัดเตรียมบนบรรทัดคำสั่ง

*action\_name* เดี่ยวจำเป็น ผู้ใช้สามารถระบุจำนวนของ *action\_args* ได้ตามต้องการ

การแปล *action\_name* และ *action\_args* ขึ้นกับนิยาม ของการดำเนินการใช้ฐานข้อมูลการดำเนินการ

การดำเนินการอาจถูกกำหนดในหนึ่ง ไฟล์ฐานข้อมูลการดำเนินการระบบ หรือในหนึ่งในไฟล์ฐานข้อมูลการดำเนินการโปรเวนต์ของผู้ใช้

*action\_args* เป็นชื่อพารของไฟล์สั้มบูรณ์หรือสัมพัทธ์ คำสั่ง **dtaction** ผ่านรายการของไฟล์นี้ไปที่การดำเนินการ ที่ระบุ

ไดอะล็อกข้อผิดพลาดถูกแสดง เมื่อพบเงื่อนไขต่อไปนี้:

- ไม่สามารถกำหนดค่าเริ่มต้นสถานะแวดล้อมเดสก์ท็อปได้
- ผู้ใช้หรือรหัสผ่านไม่ถูกต้อง
- ไม่สามารถเปลี่ยน ID เป็นผู้ใช้ที่ร้องขอ
- ไม่มีการระบุชื่อการดำเนินการ

### แฟล็ก

## ไอเท็ม

`contextDir context_dir`

`execHost host_name`

`termOpts terminal_arguments`

`user user_name`

## คำอธิบาย

ระบุบริบทดีโพลต์ไตรีทอรี ถ้านิยามของ `action_name` ไม่ได้กำหนดไตรีทอรีทำงานปัจจุบันสำหรับการดำเนินการ คำสั่ง

ระบุไฮสตรักเจอร์ทางเลือก `host_name`, สำหรับการดำเนินการคำสั่ง ถ้าการดำเนินการไม่ใช่การดำเนินการคำสั่ง คำสั่ง `dtaction` จะละเว้นอ็อพชันนี้ การดำเนินการถูกปฏิบัติที่ `host_name` แทนไฮสตรักเจอร์ในค่ากำหนด EXEC\_HOST ของการดำเนินการ มีการแสดงข้อผิดพลาดถ้าเป็นไปได้ที่จะเรียกการดำเนินการที่ระบุบนไฮสตรักเจอร์ที่เหมาะสม ระบุอาร์กิวเมนต์ที่กำหนดไว้สำหรับเทอร์มินัลอิมูเลเตอร์ ที่ถูกจัดเตรียมสำหรับการดำเนินการคำสั่งที่ไม่ใช่ชนิด NO\_STDIO ถ้ามีอักขระ white-space ในสตริง `terminal_arguments` สตริงนั้นต้องถูกใส่เครื่องหมายคำพูดเพื่อป้องกันจากเซลล์อาร์กิวเมนต์เหล่านี้ถูกส่งผ่าน โดยไม่เปลี่ยนแปลงไปที่เทอร์มินัลอิมูเลเตอร์ เพื่อผู้ใช้ประกันได้ว่าสตริง เหมาะสม โดยเฉพาะ, `terminal_arguments` ไม่รับ อาร์กิวเมนต์ที่ระบุคำสั่งที่จะถูกรันในหน้าต่างเทอร์มินัลอิมูเลเตอร์ (ซึ่งคือการใช้ `dtterm1` กับแฟล็ก `-e`)

ระบุชื่อผู้ใช้ ถ้า `dtaction` ไม่ได้รับเป็นผู้ใช้นั้น ในขณะนี้พร้อมดีโพลต์ไตรีทอรีที่สลับเก็บรหัสผ่านผู้ใช้ หรือรหัสผ่านผู้ใช้ `root` ที่ระบุ หลังจากการรหัสผ่านที่ถูกต้องถูกป้อน คำสั่ง `dtaction` เปลี่ยนแปลงเพื่อรันเป็นผู้ใช้ที่ร้องขอ แล้ว สตาร์ทการดำเนินการที่ร้องขอ

## พารามิเตอร์

### ไอเท็ม

`action_name`

`action_arg`

### คำอธิบาย

ระบุชื่อของการดำเนินการที่ต้องเรียกใช้  
ระบุไฟล์ของชื่อไฟล์สั้มบรูณ์หรือสั้มพัทธ์

## ตัวแปรสภาวะแวดล้อม

### ไอเท็ม

`DTDATABASESEARCHPATH`

### คำอธิบาย

รายการที่ค้นด้วยคอมมาของไตรีทอรี (โดยมีไฮสตรักเจอร์เลือก: นำหน้า) ซึ่งแจ้งตำแหน่งที่จะค้นหาฐานข้อมูลการดำเนินการแก้ไขรหัสผ่าน

## สถานะออก

ค่าออกต่อไปนี้จะถูกส่งคืน:

### ไอเท็ม

0

>0

### คำอธิบาย

ดำเนินการเป็นผลสำเร็จ  
มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

คำสั่ง `dtaction` เป็นแอ็พพลิเคชันที่เปิดใช้โดย PAM โดยมีชื่อเซอร์วิส `dtaction`

หากชื่อผู้ใช้ที่ระบุโดยอ็อพชัน `user user_name` แตกต่างจากชื่อล็อกอินผู้ใช้

คำสั่ง `dtaction` พิสูจน์ตัวตนผู้ใช้ ก่อนเรียกใช้การดำเนินการที่ระบุเฉพาะ โดยสามารถทำ การพิสูจน์ตัวตน PAM และการพิสูจน์ตัวตนตามปกติ

เมื่อต้องการใช้ PAM เพื่อการพิสูจน์ตัวตนแบบ system-wide, ให้สร้างสิทธิผู้ใช้ `root` และแก้ไขค่าของแอ็ททริบิวต์ `auth_type` ใน `usw stanza` ของไฟล์ `/etc/security/login.cfg` เป็น `PAM_AUTH`

กลไกการพิสูจน์ตัวตนที่ใช้เมื่อ PAM ถูกเปิดใช้ขึ้นกับ configuration สำหรับเซอวิสเซอ์กอนใน `/etc/pam.conf`

คำสั่ง `dtaction` ต้องการ `/etc/pam.conf` entry สำหรับชนิดโมดูล `auth`

configuration ต่อไปนี้แนะนำให้ใช้ใน `/etc/pam.conf` สำหรับเซอวิสเซอ์กอน `dtaction`:

```
dtaction      auth                required      /usr/lib/security/pam_aix
```

## ตัวอย่าง

1. เมื่อต้องการเรียกการดำเนินการ ป้อน:

```
dtaction Xterm
```

คำนี้เรียกทำงานเทอร์มินัลอิมูเลเตอร์ X Windows (Xterm)

2. เมื่อต้องการเรียกการดำเนินการบนรีโมตโฮสต์ให้ป้อน:

```
dtaction -execHost hostname Xterm
```

นี้จะเรียกใช้งาน Xterm บนรีโมตโฮสต์ที่ระบุ

3. เมื่อต้องการเรียกการดำเนินการในฐานะผู้ใช้อื่น ให้ป้อน:

```
dtaction -user username Xterm
```

นี้จะเรียกใช้งาน Xterm ตามผู้ใช้ที่ระบุ

## Location

`/usr/dt/bin/dtaction`

## ขอผิดพลาดมาตรฐาน

คำสั่ง `dtaction` เขียนข้อความแสดงความผิดพลาดของการวินิจฉัยไปที่ข้อผิดพลาด มาตรฐาน ซึ่งถูกเปลี่ยนทิศทางไปที่ `HOME/.dt/errorlog`

## ไฟล์

ไอเท็ม

`/etc/pam.conf`

`/etc/security/login.cfg`

คำอธิบาย

กำหนดกลไกการพิสูจน์ตัวตน PAM

กำหนด PAM authentication system-wide

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dtlogin`” ในหน้า 236

“คำสั่ง `dtsession`” ในหน้า 265

---

## คำสั่ง `dtappintegrate`

## วัตถุประสงค์

เครื่องมือการรวมแอพลิเคชัน Common Desktop Environment

## ไวยากรณ์

`dtappintegrate -s ApplicationRoot [ -t TargetPath ] [ -l Language ] [ -u ]`

### คำอธิบาย

คำสั่ง `dtappintegrate` ลิงก์ ไฟล์คอนฟิกูเรชันแอปพลิเคชัน CDE จากตำแหน่งจำเพาะแอปพลิเคชัน ไปที่ตำแหน่งระบบ และอัปเดตวอลุ่มวิธีใช้เบราว์เซอร์ของระบบสำหรับ ภาษาที่ได้รับผลกระทบ คำสั่ง `dtappintegrate` ถูกใช้ ระหว่างกระบวนการติดตั้งแอปพลิเคชัน สคริปต์การติดตั้งแอปพลิเคชัน ควรเรียกคำสั่ง `dtappintegrate` ที่จุด ลื่นสุด

มีสี่คีย์ไคเร็กทอรีย่อยใต้แอปพลิเคชัน root (อ้างอิงถึงโดย `$APP_ROOT`) ที่ควบคุมโดยนโยบาย CDE ไคเร็กทอรีมีดังนี้:

#### ไอเท็ม

`$APP_ROOT/dt/appconfig/types/ Language`

`$APP_ROOT/dt/appconfig/appmanager/ Language`

`$APP_ROOT/dt/appconfig/icons/ Language`

`$APP_ROOT/dt/appconfig/help/ Language`

#### คำอธิบาย

สำหรับ filetype, Front Panel และไฟล์การดำเนินการ

สำหรับไฟล์กลุ่มแอปพลิเคชัน

สำหรับไอคอนที่ใช้โดยตัวจัดการ CDE

สำหรับวิธีใช้แอปพลิเคชัน ตัวอย่าง แอปพลิเคชัน default-language

SpreadSheet จะโหลดเดสก์ทอปไอคอนภายใต้: `/opt/SpreadSheet/dt/`

`appconfig/icons/C/*.bm` และ `/opt/SpreadSheet/dt/appconfig/icons/C/`

`*.pm` โดยที่ `/opt/SpreadSheet` คือค่าของ `$APP_ROOT`

หมายเหตุ: `$APP_ROOT` เป็นข้อตกลงโครงสร้างไวยากรณ์ของ เอกสารนี้ และไม่ใช่โดยสภาวะแวดล้อมรันไทม์) ทั้งหมดของไฟล์คอนฟิกูเรชัน CDE และไคเร็กทอรีย่อยเหล่านี้ถูกกำหนดไว้ที่ระดับบนสุดทั่วไป และควรมีไคเร็กทอรีย่อยภาษาดีฟอลต์ C เสมอ

ในกรณีธรรมดาที่สุด คำสั่งรับ แอปพลิเคชัน root เป็นอินพุต ตัวอย่าง, `/opt/thisapp` เอาต์พุตจาก การดำเนินการนี้เป็นไคเร็กทอรีย่อยและไฟล์ที่ตรงกันบนแอปพลิเคชันเซิร์ฟเวอร์ ที่มีลิงก์สัญลักษณ์สัมพันธ์กับไฟล์คอนฟิกูเรชันแอปพลิเคชัน CDE ที่อธิบายด้านบน ภายใต้ตำแหน่งระบบต่อไปนี้:

#### ไอเท็ม

`/etc/dt/appconfig`

#### คำอธิบาย

ไคเร็กทอรีย่อยคอนฟิกูเรชันระดับบนสุด ประกอบด้วยไคเร็กทอรีย่อย ดังต่อไปนี้:

`/etc/dt/appconfig/types/ Language/`

มี `*.dt` และลิงก์ `*.fp`

`/etc/dt/appconfig/appmanager/ Language/`

มีลิงก์ไปที่ไคเร็กทอรีย่อยกลุ่มแอปพลิเคชันและไฟล์สคริปต์การดำเนินการเพื่อแสดงเป็นการดำเนินการใต้ Application Manager

`/etc/dt/appconfig/help/ Language/`

มีลิงก์สัญลักษณ์ไปที่ไฟล์วิธีใช้ที่ติดตั้งใต้ root ของแอปพลิเคชัน

`/etc/dt/appconfig/icons/ Language/`

มีลิงก์สัญลักษณ์ไปที่ไอคอน CDE สำหรับแอปพลิเคชัน

## แฟล็ก

## ไอเท็ม

-s ApplicationRoot  
-t TargetPath

## คำอธิบาย

รวมแอพลิเคชันที่ ApplicationRoot แฟล็กนี้จำเป็นต้องมี  
ลิงก์ไฟล์คอนฟิกูเรชันแอพลิเคชัน CDE จากตำแหน่งจำเพาะแอพลิเคชัน ไปที่ TargetPath แทน  
ตำแหน่งระบบ แฟล็ก นี้เป็นทางเลือก

ถ้าแฟล็ก -t ถูกระบุ ไฟล์ถูกลิงก์ภายใต้ไต่เร็กทอรีย่อยที่ระบุ ตัวอย่าง การระบุ -t /etc/dt/private จะทำให้  
ไฟล์วิธีใช้แอพลิเคชัน ถูกลิงก์ทางสัญลักษณ์ภายใต้ /etc/dt/private/help/Language แฟล็กนี้โดยปกติ  
ถูกใช้เฉพาะโดย ผู้ดูแลระบบ ซึ่งต้องการสร้างแอพลิเคชันแยก และไม่ใช่ แอพลิเคชัน post-installation  
สคริปต์ โดยดีฟอลต์ (ไม่ระบุ -t) root ไต่เร็กทอรีย่อยของแอพลิเคชันเป็นค่าโกลบอลต่อแอพลิเคชัน  
โฮสต์ แอพลิเคชันทั้งหมดที่ติดตั้งบนโฮสต์จะคัดลอกไฟล์คอนฟิกูเรชัน ไปที่ตำแหน่งเดียวกันเพื่อรวม  
กับไฟล์คอนฟิกูเรชัน ของแอพลิเคชันอื่น

-L Language

ระบุภาษาที่จะรวม โดยทั่วไป แฟล็กนี้ระบุ ไต่เร็กทอรีที่จะค้นหาไฟล์คอนฟิกูเรชันแอพลิเคชัน CDE ถ้า  
ไม่ระบุพารามิเตอร์นี้ ภาษาทั้งหมดจะถูกรวมไว้ พารามิเตอร์ นี้เป็นทางเลือก  
การรวมแอพลิเคชันถูกยกเลิก แฟล็กนี้คือแฟล็กเพื่อเลือก

-u

---

## คำสั่ง dtlogin

### วัตถุประสงค์

ดำเนิน CDE ล็อกอินเซอวิสเซ

### ไวยากรณ์

```
dtlogin [ -config configuration_file ] [ -daemon ] [ -debug debug_level ] [ -error error_log_file ] [ -nodaemon ] [ -resources resource_file ] [ -server server_entry ] [ -session session_program ] [ -udpPort port_number ]
```

### คำอธิบาย

คำสั่ง dtlogin สนับสนุนงานหลักต่อไปนี้:

- Launching dtgreet login screen for explicitly managed local and remote displays and XDMCP-managed remote displays.
- เข้าถึงล็อกอินเทอร์มินัล (อักขระ) ปกติจากจอภาพล็อกอิน GUI
- การพิสูจน์ตัวตนและล็อกอินในผู้ใช้ system-dependent
- การเรียกทำงานเซสชันที่เลือก

คำสั่ง dtlogin มีเซอวิสเซเหมือนกับที่มีใน init, getty และ login บนเทอร์มินัลอักขระ, ซึ่งรวมถึงการพร้อมท์เพื่อของล็อกอินและรหัสผ่าน, การพิสูจน์ตัวตนผู้ใช้ และการรันเซสชัน session ถูกกำหนดโดยช่วงชีวิต ของกระบวนการ ในโลกของเทอร์มินัลที่ใช้อักขระปกติ เซสชันคือกระบวนการเซลล์ล็อกอินของผู้ใช้; ในบริบท DT คือ DT Session Manager ถ้าไม่ใช่ DT Session Manager, โดยปกติ จะแทนด้วย window manager ที่มีอ็อปชัน exit, หรือเทอร์มินัลอิมูเลเตอร์ รันเซลล์, โดยที่ช่วงชีวิตของเทอร์มินัลอิมูเลเตอร์คือช่วงชีวิต ของกระบวนการเซลล์ที่เทอร์มินัลอิมูเลเตอร์รันอยู่ นี่เป็นการลดเซสชัน X ต่อ อิมูเลชันของ character-based เทอร์มินัลเซสชัน เมื่อเซสชันถูกจบการทำงาน, dtlogin รีเซ็ต X server และ (เป็นทางเลือก) รีเซ็ตาร์ททั้งกระบวนการ

คำสั่ง dtlogin สนับสนุนการจัดการจอแสดงผลรีโมทโดยใช้ X Display Manager Control Protocol, Version 1.0 (XDMCP) เมื่อ dtlogin ได้รับเคียวรีทางอ้อมจาก XDMCP, คำสั่งสามารถรันกระบวนการ chooser เพื่อดำเนิน XDMCP BroadcastQuery

(หรือ XDMCP Query กับโฮสต์ที่ระบุ) ในแบบของจอแสดงผลและเสนอเมนูของโฮสต์ที่เป็นไปได้ที่เสนอการจัดการ จอแสดงผล XDMCP คุณลักษณะนี้มีประโยชน์กับ X terminals ที่ไม่สนับสนุน โฮสต์เมนู

เนื่องจาก **dtlogin** มีอินเตอร์เฟซแรกที่ใช้เห็น จึงถูก ออกแบบให้ง่ายต่อการใช้งานและการปรับแต่งตามความต้องการของ ไซต์

## หน้าต่างล็อกอิน

หน้าต่างล็อกอินอนุญาตให้ผู้ใช้ป้อน ID ผู้ใช้ และรหัสผ่าน เลือก เซสชันเริ่มต้น และเลือกโลแคลเริ่มต้น ผู้ใช้ยังสามารถรีเซ็ต X server หรือหยุด X server ชั่วคราวเพื่อเข้าถึงพร้อมต์ล็อกอิน อักขระ

เนื้อหาของหน้าต่างล็อกอินมีดังนี้:

### ล็อกอินฟิลด์

มี entry ฟิลด์ซึ่งผู้ใช้ป้อน ID

### ฟิลด์รหัสผ่าน

มี entry ฟิลด์ซึ่งผู้ใช้ป้อนรหัสผ่าน(no-echo)

### ปุ่มตกลง

พิสูจน์ตัวตนผู้ใช้และเรียกใช้เซสชัน

### ปุ่มเคลียร์

เคลียร์ฟิลด์ล็อกอินและรหัสผ่าน

### อ็อปชัน

ให้ผู้ใช้เลือกชื่อโลแคล และชนิดล็อกอินเซสชัน และยังให้ ผู้ใช้รีสตาร์ท X server หรือสลับไปที่พร้อมต์ล็อกอินแบบ อักขระ (สำหรับจอแสดงผล โลคัล) เนื้อหาของเมนูอ็อปชันมีดังนี้:

**ภาษา** แสดงเมนูภาษา เลือกภาษาจากเมนูอ็อปชันของจอภาพ ล็อกอิน จะแปลภาษาของจอภาพล็อกอินทันที และเซตตัวแปร LANG สำหรับเซสชันต่อไป การแปลจอภาพล็อกอินและ LANG กลับไปเป็นค่าดีฟอลต์ ขณะการเสร็จสิ้นเซสชัน เนื้อหาของเมนูนี้ต่างกันไปขึ้นกับโลแคลที่ติดตั้งบน ระบบ ซึ่งถูกเขียนทับได้โดยใช้ รีซอร์ส languageList ดีฟอลต์โลแคลของ C สามารถถูกเขียนทับโดยใช้รีซอร์สภาษา ระบบ หรือ languageList โลแคลที่ระบุถูกแสดงเป็นรายการเมนู ในเมนูภาษา ข้อความทางเลือกที่จะถูกแสดงสามารถ ระบุได้สำหรับ ชื่อโลแคลที่กำหนดโดยใช้รีซอร์ส languageName

### No-windows

แสดงพร้อมต์ล็อกอินแบบอักขระ (จอแสดงผลโลคัลเท่านั้น)

### Reload Login

รีสตาร์ท X Server และกลับไปจอภาพล็อกอิน

**รีซอร์ส** รายการรีซอร์สที่จะถูกใช้

**เซสชัน** แสดงเมนูเซสชัน อนุญาตให้ผู้ใช้เลือกชนิดเซสชันที่ควรถูกสตาร์ท เมื่อล็อกอิน รายการเมนูรวมถึง:

### DT Session

สตาร์ทเดสก์ท็อปเซสชันปกติ (Xsession)

### Fail-safe Session

สตาร์ท fail-safe เซสชัน (Xfailsafe)

## วิธีใช้ แสดงข้อความวิธีใช้

### การควบคุมเซิร์ฟเวอร์

คำสั่ง **dtlogin** ควบคุมโวลคัลเซิร์ฟเวอร์โดยใช้สัญญาณ POSIX ต้องการสัญญาณ SIGHUP เมื่อรีเซ็ตเซิร์ฟเวอร์, ปิดการเชื่อมต่อไคลเอ็นต์ทั้งหมด และดำเนินหน้าที่ล้างค่าอื่นๆ ต้องการสัญญาณ SIGTERM ในการจบการทำงาน เซิร์ฟเวอร์ ถ้าสัญญาณเหล่านี้ไม่ทำงานตามที่คาดหวัง รีซอร์ส **resetSignal** และ **termSignal** สามารถระบุสัญญาณทางเลือก

เมื่อต้องการควบคุมรีโมตเซิร์ฟเวอร์ที่ไม่ใช่ XDMCP, **dtlogin** ค้นหา ลำดับชั้นหน้าต่างบนจอแสดงผลและใช้การร้องขอโปรโตคอล KillClient X ในการพยายามล้างค่าเทอร์มินัลสำหรับเซสชันต่อไป นี่อาจไม่จบการทำงานของไคลเอ็นต์จริง เนื่องจากเฉพะไคลเอ็นต์ที่สร้างหน้าต่างเท่านั้น ที่มีการรับทราบ XDMCP มีกลไกที่แน่นอนกว่า; เมื่อ **dtlogin** ปิด การเชื่อมต่อเริ่มต้นเซสชันจบการทำงานและจำเป็นต้องใช้เทอร์มินัลในการ ปิดการเชื่อมต่ออื่นทั้งหมด

### การควบคุม dtlogin

คำสั่ง **dtlogin** ตอบสนองกับสองสัญญาณ: SIGHUP และ SIGTERM เมื่อส่ง SIGHUP, **dtlogin** อ่านไฟล์คอนฟิกูเรชัน และไฟล์ที่ระบุโดยเซิร์ฟเวอร์รีซอร์ส และกำหนดว่ารายการถูกเพิ่มหรือลบ ถ้ารายการใหม่ถูกเพิ่ม, **dtlogin** เริ่ม เซสชันบนจอแสดงผลที่เกี่ยวข้องของ รายการที่ถูกลบออกถูกปิดใช้งานทันที, หมายความว่าเซสชันที่กำลังดำเนินอยู่จะถูกจบการทำงานโดยไม่มี การเตือน และไม่มีเซสชันใหม่ถูกเริ่มทำงาน เมื่อส่ง SIGTERM, **dtlogin** จบการทำงาน เซสชันทั้งหมดที่กำลังดำเนินอยู่และออก นี้สามารถใช้ได้เมื่อทำการปิดระบบ

### การทำให้เป็นสากล

เลเบลและข้อความทั้งหมดแปลให้เป็นภาษาต่างๆ ได้ แค็ตตาล็อกข้อความ **dtlogin.cat** มีการแสดงข้อมูลที่แปลแล้วของเลเบลและข้อความดีฟอลต์ คำสั่ง **dtlogin** อ่านแค็ตตาล็อกข้อความที่เกี่ยวข้องที่ระบุโดย ตัวแปรสภาวะแวดล้อม LANG และแสดงสตริงที่แปล อีอพชันในจอภาพการพิสูจน์ตัวตน อนุญาตให้ผู้ใช้เขียนทับภาษา ดีฟอลต์สำหรับเซสชันที่ตามมา ถ้าจอภาพการพิสูจน์ตัวตนได้ถูก แปลสำหรับภาษาที่เลือก จอภาพจะถูกแสดงใหม่ในภาษานั้น; หรือไม่แล้วจะแสดงภาษาดีฟอลต์ ไม่ว่าในกรณีใด ตัวแปรสภาวะแวดล้อม LANG ถูกเซ็ตัวอย่างเหมาะสมสำหรับเซสชันที่เป็นผลลัพธ์

ภาษารีซอร์สพร้อมใช้ในไฟล์คอนฟิกูเรชัน **dtlogin** เพื่อ เปลี่ยนภาษาดีฟอลต์สำหรับการแสดงผล รีซอร์ส **languageList** มีในไฟล์คอนฟิกูเรชัน **dtlogin** เพื่อเขียนทับชุดดีฟอลต์ ของภาษาที่แสดงบนจอภาพการพิสูจน์ตัวตน รีซอร์ส **languageName** พร้อมใช้ในการเตรียมการแม้พจากชื่อโลแคลไปเป็น ข้อความที่แสดงบนเมนูภาษา

### การพิสูจน์ตัวตนและการตรวจสอบ

คำสั่ง **dtlogin** คือล็อกอินเซอร์วิสที่เปิดใช้โดย PAM พร้อมกับชื่อเซอร์วิส **dtlogin** ไคลเอ็นต์ **dtlogin** สนับสนุนการพิสูจน์ตัวตน PAM นอกเหนือจากไลคัล UNIX ล็อกอินและการตรวจสอบ ปกติ ฟังก์ชัน การพิสูจน์ตัวตนหรือการตรวจสอบเพิ่มเติม เช่น Kerberos หรือ B1 สามารถถูกเพิ่ม โดยคู่ค่าแต่ละราย

เมื่อต้องการใช้ PAM สำหรับการพิสูจน์ตัวตน system-wide, ให้สร้างสิทธิผู้ใช้ root และแก้ไขค่าของแอ็ททริบิวต์ **auth\_type** ใน **usw stanza** ของไฟล์ **/etc/security/login.cfg** เป็น PAM\_AUTH

กลไกการพิสูจน์ตัวตนที่ใช้เมื่อ PAM ถูกเปิดใช้ขึ้นกับ configuration สำหรับเซอร์วิสล็อกอินใน **/etc/pam.conf** คำสั่ง **dtlogin** ต้องการ **/etc/pam.conf** entry สำหรับชนิดโมดูล **auth**, **account**, **password** และ **session** configuration ต่อไปนี้ แนะนำให้ใช้ใน **/etc/pam.conf** สำหรับเซอร์วิส **dtlogin**:

dtlogin	auth	required	/usr/lib/security/pam_aix
dtlogin	account	required	/usr/lib/security/pam_aix
dtlogin	password	required	/usr/lib/security/pam_aix
dtlogin	session	required	/usr/lib/security/pam_aix

## การรักษาความปลอดภัย X Server

X server มีค่าควบคุมการเข้าถึงทั้งแบบ user-based และ host-based โดย ดีฟอลต์, dtlogin ใช้ค่าควบคุมการเข้าถึง user-based กับ X server (MIT-MAGIC-COOKIE-1) ระดับความปลอดภัยนี้อนุญาตค่าควบคุมการเข้าถึงในแบบต่อผู้ใช้ซึ่งมาจาก รูปแบบถ้าไคลเอ็นต์ส่งข้อมูลการพิสูจน์ตัวตนที่ตรงกับ ที่เซิร์ฟเวอร์มี ไคลเอ็นต์จะได้รับอนุญาตให้เข้าถึง เมื่อผู้ใช้ล็อกอิน ข้อมูลการพิสูจน์ตัวตน นี้โดยดีฟอลต์ถูกเก็บและป้องกันในไฟล์ \$HOME/.Xauthority

อย่างไรก็ตาม การใช้กลไกค่าควบคุมการเข้าถึง host-based อาจเหมาะสม ในสภาวะแวดล้อมที่เน็ตเวิร์กไม่ปลอดภัย เนื่องจากค่าควบคุมการเข้าถึง user-based อนุญาตให้ทุกโฮสต์เชื่อมต่อ ถ้าโฮสต์มีไพรเวตคีย์ ข้อเสีย อีกอย่างของค่าควบคุมการเข้าถึง user-based คือไคลเอ็นต์ R2 หรือ R3 ไม่สามารถ เชื่อมต่อกับเซิร์ฟเวอร์

รีซอร์ส authorize ควบคุมว่า ค่าควบคุมการเข้าถึง user-based หรือ host-based ค่าใดที่ถูกใช้โดย dtlogin ดูที่คำสั่ง xhost และ xauth สำหรับข้อมูลเพิ่มเติม

## รีซอร์ส

คำสั่ง dtlogin ถูกควบคุมโดยเนื้อหาของไฟล์คอนฟิกูเรชัน dtlogin ซึ่งเป็นดีฟอลต์กับ /usr/dt/config/Xconfig บางรีซอร์สควบคุมการทำงานของ dtlogin โดยทั่วไป, และสามารถระบุรีซอร์สอื่นสำหรับการแสดงผลเฉพาะ

## รีซอร์สทั่วไป

รีซอร์สทั่วไปของ dtlogin ต่อไปนี้ไม่ใช่ display-specific และใช้กับจอแสดงผลทั้งหมดเมื่อเหมาะสม

### ไอเท็ม

#### accessFile

#### คำอธิบาย

คลาส: AccessFile

#### ชนิดคลาส:

String

#### ดีฟอลต์:

Null

#### คำอธิบาย:

เพื่อป้องกันเซิร์ฟเวอร์ XDMCP ที่ไม่ได้รับอนุญาตและเพื่อให้สามารถทำการส่งต่อการร้องขอ XDMCP IndirectQuery ไฟล์นี้มีฐานข้อมูลของชื่อโฮสต์ที่ได้รับอนุญาตให้เข้าถึง เครื่องนี้ได้โดยตรงหรือมีรายการของโฮสต์ซึ่งควรส่ง เคียวรีไปให้อ้างถึงส่วน Xaccess file สำหรับรายละเอียดของรูปแบบ ถ้ารีซอร์สนี้ไม่ถูก เซ็ต โฮสต์ทั้งหมดจะอนุญาตเซิร์ฟเวอร์ XDMCP

#### authDir

คลาส: AuthDir

#### ชนิดคลาส:

String

#### ดีฟอลต์:

/var/dt

#### คำอธิบาย:

ชื่อไดเรกทอรีที่ dtlogin ใช้เพื่อเก็บไฟล์การอนุญาตชั่วคราว สำหรับการแสดงผลโดยใช้ XDMCP

ไอเท็ม autoRescan	<p><b>คำอธิบาย:</b></p> <p><b>คลาส:</b> AutoRescan</p> <p><b>ชนิดคลาส:</b> Boolean</p> <p><b>ดีฟอลต์:</b> True</p> <p><b>คำอธิบาย:</b> ควบคุมว่า dtlogin รีสแกนไฟล์คอนฟิกูเรชัน และเซิร์ฟเวอร์ไฟล์หรือไม่หลังจากเซสชันสิ้นสุด และไฟล์มีการเปลี่ยนแปลง คุณสามารถ บังคับ dtlogin ให้อ่านไฟล์เหล่านี้ซ้ำโดยการส่งสัญญาณ SIGHUP ไปที่กระบวนการหลัก</p>
daemonMode	<p><b>คลาส:</b> DaemonMode</p> <p><b>ชนิดคลาส:</b> Boolean</p> <p><b>ดีฟอลต์:</b> False</p> <p><b>คำอธิบาย:</b> คำสั่ง dtlogin สามารถทำให้ตัวเองเป็นกระบวนการ daemon ที่ไม่เกี่ยวข้อง ซึ่งทำได้โดยแยกและปล่อยให้กระบวนการพารেন্টจบการทำงาน แล้วปิดไฟล์ descriptors และปลดการควบคุมเทอร์มินัล ซึ่งไม่สะดวกเมื่อจะทำการดีบั๊ก dtlogin การเซตริชอร์ส เป็น False จะปิด daemonMode</p>
debugLevel	<p><b>คลาส:</b> DebugLevel</p> <p><b>ชนิดคลาส:</b> Int</p> <p><b>ดีฟอลต์:</b> 0</p> <p><b>คำอธิบาย:</b> ค่าที่ไม่ใช่ศูนย์ที่ระบุสำหรับริชอร์สจำนวนเต็มนี้ทำให้พิมพ์ ข้อมูลการดีบั๊กได้ และยังปิดโหมด daemon, ซึ่งเปลี่ยนทิศทาง ข้อมูลไปที่ bit-bucket ที่ไม่มีประโยชน์ตามปกติ</p>
errorLogFile	<p><b>คลาส:</b> ErrorLogFile</p> <p><b>ชนิดคลาส:</b> String</p> <p><b>ดีฟอลต์:</b> NULL</p> <p><b>คำอธิบาย:</b> เอาต์พุตข้อผิดพลาดโดยปกติส่งไปที่คอนโซลระบบ เมื่อต้องการเปลี่ยนทิศทาง ให้เซตริชอร์สนี้เป็นชื่อไฟล์ ไฟล์นี้มีเอาต์พุตที่กำหนดไปที่ stderr โดย Xsetup, Xstartup และ Xreset</p>
errorLogSize	<p><b>คลาส:</b> errorLogSize</p> <p><b>ชนิดคลาส:</b> Int</p> <p><b>ดีฟอลต์:</b> 50</p> <p><b>คำอธิบาย:</b> ริชอร์สนี้ระบุขนาดสูงสุดของไฟล์บันทึกข้อผิดพลาดเป็นกิโลไบต์ เมื่อถึงค่าจำกัด, dtlogin จะลบรายการเก่าที่สุดในไฟล์ออกจนกว่าขนาดไฟล์ถูกลดลงเหลือ 75 เปอร์เซ็นต์ของค่ามากที่สุด หลีกเลี่ยง ไฟล์ถูกลดขนาด ผู้ใช้ที่กำลังเข้าถึงไฟล์บันทึกข้อผิดพลาด (ตัวอย่าง เช่นการใช้ cat หรือ tail) จะจำเป็นต้องปิดไฟล์ และเปิดใหม่ เพื่อเข้าถึงเพื่อ ดูข้อมูลตามลำดับที่ถูกรับบันทึกในไฟล์</p>

ไอเท็ม  
exportList

คำอธิบาย

คลาส: ExportList

ชนิดคลาส:

String

ดีฟอลต์: NULL

คำอธิบาย:

มีชุดของชื่อตัวแปรคั่นโดยช่องว่างหรือแท็บ แต่ละตัวแปรได้ชื่อจากสภาวะแวดล้อม dtlogin และโหลดลง  
ในสภาวะแวดล้อมของเซิร์ฟเวอร์และเซสชัน ดูที่ส่วน สภาวะแวดล้อม สำหรับรายละเอียด

fontPathHead

คลาส: FontPathHead

ชนิดคลาส:

String

ดีฟอลต์: NULL

คำอธิบาย:

ค่าที่นำหน้าให้กับดีฟอลต์ X server ฟอนต์พาธ

fontPathTail

คลาส: fontPathTail

ชนิดคลาส:

String

ดีฟอลต์: NULL

คำอธิบาย:

ค่าที่ถูกต่อท้ายให้กับดีฟอลต์ X server ฟอนต์พาธ

keyFile

คลาส: KeyFile

ชนิดคลาส:

String

ดีฟอลต์: /usr/dt/config/Xkeys

คำอธิบาย:

การพิสูจน์ตัวตน XDM-AUTHENTICATION-1 style XDMCP ต้องการให้โปรเวตคีย์ ถูกแบ่งใช้  
ระหว่าง dtlogin และเทอร์มินัล รีซอร์สรีระบุไฟล์ที่มีค่าเหล่านั้น แต่ละ entry ในไฟล์ประกอบด้วยชื่อจอแสด  
ดงผล และคีย์ที่แบ่งใช้โดยดีฟอลต์ dtlogin ไม่รวมการสนับสนุน XDM-AUTHENTICATION-1 เนื่อง  
จากต้องการ DES, ซึ่งโดยทั่วไปจะแจกจ่ายไม่ได้

lockPidFile

คลาส: LockPidFile

ชนิดคลาส:

Boolean

ดีฟอลต์: True

คำอธิบาย:

ควบคุมว่า dtlogin ใช้การล็อกไฟล์เพื่อป้องกันหลาย อินสแตนซ์ของ dtlogin จากการทำงานพร้อมกันหรือไม่

ไอเท็ม networkDevice	<p><b>คำอธิบาย</b></p> <p><b>คลาส:</b> NetworkDevice</p> <p><b>ชนิดคลาส:</b> String</p> <p><b>ดีฟอลต์:</b> /dev/dtremote</p> <p><b>คำอธิบาย:</b> สำหรับการเชื่อมต่อรีโมต ค่าสำหรับ line ใน /etc/utmp ต้องมีอยู่เช่นเดียวกับอุปกรณ์ในไดเรกทอรี /dev สำหรับคำสั่ง เช่น finger เพื่อให้ทำงานอยู่ถูกต้อง รีซอร์สรีบูตชื่อพาธ ของไฟล์ /dev ที่ dtlogin สร้าง เมื่อจอแสดงผลรีโมต เชื่อมต่อสำหรับแพลตฟอร์มส่วนใหญ่ ไฟล์ถูกสร้างเป็นลิงก์สัญลักษณ์ไปที่ /dev/null ค่าที่ระบุต้องเริ่มต้นด้วย /dev/, หรือไม่แล้ว ค่าจะถูกละทิ้งและไม่มีการสร้างไฟล์</p>
pidFile	<p><b>คลาส:</b> PidFile</p> <p><b>ชนิดคลาส:</b> SString</p> <p><b>ดีฟอลต์:</b> NULL</p> <p><b>คำอธิบาย:</b> ชื่อไฟล์ที่ระบุถูกสร้างเพื่อเก็บการแสดงค่า ASCII ของ process-ID ของกระบวนการหลัก dtlogin ซึ่งสามารถถูกใช้เมื่อทำการส่งสัญญาณไปที่ dtlogin ไคลเอ็นต์ dtlogin ยังใช้การล็อกไฟล์เพื่อจะป้องกันไม่ให้ dtlogin รันมากกว่าหนึ่ง dtlogin บน เครื่องเดียวกัน ดูที่รีซอร์ส lockPidFile สำหรับข้อมูลเพิ่มเติม</p>
removeDomainname	<p><b>คลาส:</b> RemoveDomainname</p> <p><b>ชนิดคลาส:</b> Boolean</p> <p><b>ดีฟอลต์:</b> True</p> <p><b>คำอธิบาย:</b> เมื่อคำนวณชื่อจอแสดงผลสำหรับไคลเอ็นต์ XDMCP, dtlogin โดยปกติ จะสร้างชื่อโฮสต์ที่มีการระบุแบบเต็มสำหรับเทอร์มินัล เนื่องจากอาจสร้างความสับสนในบางครั้ง, dtlogin ลบส่วนของชื่อโดเมนของชื่อโฮสต์ ถ้าเหมือนกับชื่อโดเมนสำหรับโลคัลโฮสต์ เมื่อตัวแปรนี้ ถูกเซต</p>
requestPort	<p><b>คลาส:</b> RequestPort</p> <p><b>ชนิดคลาส:</b> int</p> <p><b>ดีฟอลต์:</b> 177</p> <p><b>คำอธิบาย:</b> ระบุหมายเลขพอร์ต UDP ที่ dtlogin ใช้เพื่อรอ การร้องขอ XDMCP ขาเข้า นอกจากระบบจำเป็นต้องถูกดีบั๊ก ค่าดีฟอลต์สำหรับรีซอร์สนี้ควรเหมือนเดิม</p>

คำอธิบาย

คลาส: Servers

ชนิดคลาส:

String

ดีฟอลต์: :0 Local local /system\_dependent\_path/X :0

คำอธิบาย:

ระบุชื่อไฟล์เต็มของรายการเซิร์ฟเวอร์, หนึ่งชื่อต่อหนึ่งบรรทัด (ถ้า คำเริ่มต้นด้วย slash) หรือรายการเซิร์ฟเวอร์เดี่ยว แต่ละรายการหมายถึง จอแสดงผลที่ควรถูกจัดการอย่างคงที่ และไม่ใช่ XDMCP ไวยากรณ์ทั่วไปสำหรับรายการเป็นดังนี้:

*DisplayName DisplayClass DisplayType[@ite] [Command [options]]*

โดยที่:

*DisplayName*

ค่าที่สามารถถูกส่งในอ็อปชัน **-display** ไปที่ X program สตริงนี้ถูกใช้ในรีซอร์ส display-specific เพื่อระบุจอแสดงผลเฉพาะ ดังนั้นควรระมัดระวังในการจับคู่ชื่อ ตัวอย่าง, ใช้ :0 local /usr/bin/X11/X :0 แทน localhost:0 local /usr/bin/X11/X :0 ถ้ารีซอร์สอื่นของคุณถูกระบุเป็น Dtlogin.\_0.session เครื่องหมายดอกจัน (\*) ในฟิลด์นี้ขยายเป็น hostname:0 โดย dtlogin

*DisplayClass*

ส่วนของคลาสจอแสดงผลยังถูกใช้ในรีซอร์ส display-specific เป็นส่วนของคลาสของรีซอร์ส ซึ่งมีประโยชน์ ถ้าคุณมีคอลเล็กชันจำนวนมาก ของจอแสดงผลที่เหมือนกัน (กลุ่มของ X terminals, เป็นต้น) และต้องการเซ็ทรีซอร์สสำหรับกลุ่ม เมื่อใช้ XDMCP, จอแสดงผลจำเป็นต้อง ระบุคลาสจอแสดงผลอ้างอิงถึงเอกสารคู่มือ X terminal ของคุณสำหรับข้อมูล เกี่ยวกับ สตริงคลาสจอแสดงผลมาตรฐานที่เหมาะสมสำหรับอุปกรณ์ของคุณ

*DisplayType*

ถ้าระบุเป็น local, หมายความว่า X server ควร ถูกสตาร์ทสำหรับ entry นี้ ค่าของ remote หมายถึง X server ที่มีอยู่ควรถูกเชื่อมต่อ

@ite

บนโลคัลบิตแม็พ ผู้ใช้สามารถเลือกอ็อปชัน ล็อกอินด้วยบรรทัดคำสั่ง โดยใช้จอภาพล็อกอิน ซึ่งจะหยุด X-server ชั่วคราวและแสดงพร้อมต์ login: อีกประเภทติ จากนั้นผู้ใช้สามารถล็อกอิน และดำเนินงานที่เกี่ยวข้องที่เป็นงาน non-X เมื่อผู้ใช้เสร็จสิ้นการล็อกเอาต์ X-server จะถูกรีสตาร์ทและจอภาพล็อกอินถูกแสดงอีกครั้ง เพื่อสนับสนุนโหมด **Command Line Login** จอแสดงผลต้องมีอุปกรณ์ Internal Terminal Emulator (ITE) ที่สัมพันธ์กัน โดยดีฟอลต์, dtlogin เชื่อมโยง อุปกรณ์ ITE "console" (/dev/console) กับจอแสดงผล :0 ถ้า configuration ของคุณไม่ตรงกับค่าดีฟอลต์นี้, ระบุ @device สำหรับ จอแสดงผล ITE ที่เชื่อมโยง, และระบุ @none สำหรับ จอแสดงผลอื่นที่มีในเซิร์ฟเวอร์ไฟล์

**Command [options]**

สตริงที่สตาร์ทเซิร์ฟเวอร์ X โคลเอ็นต์ dtlogin จะเชื่อมต่อกับ X server เสมอโดยใช้ *DisplayName* ที่ระบุ ดังนั้น คุณอาจจำเป็นต้องระบุหมายเลขการเชื่อมต่อเป็นอ็อปชันกับ X server ของคุณ (:0 ในตัวอย่างก่อนหน้า)

ไอเท็ม  
sysParmsFile

คำอธิบาย

คลาส: SysParmsFile

ชนิดคลาส:

String

ดีฟอลต์: /system\_dependent\_path

คำอธิบาย:

ระบุไฟล์ที่มีคำสั่งเซลล์ซึ่งเซตตัวแปรสถานะแวดล้อม เขตเวลา (TZ) ให้กับระบบ ถ้าเขตเวลาถูกเซตโดยใช้ไวยากรณ์เซลล์ TZ=, dtlogin สามารถใช้ข้อมูลนี้เพื่อเซตเขตเวลาสำหรับเซสชันผู้ใช้

timeZone

คลาส: TimeZone

ชนิดคลาส:

String

ดีฟอลต์: NULL

คำอธิบาย:

ระบุเขตเวลาสำหรับ dtlogin ซึ่งถูกโหลดใน สภาวะแวดล้อมของ dtlogin เป็นค่าของตัวแปร TZ และที่สืบทอดโดยเซสชันต่อมา บางระบบรักษาไฟล์คอนฟิกูเรชัน ที่มีการตั้งค่าเขตเวลา (ตัวอย่าง, /etc/src.sh) ดูเพิ่มเติมที่รีจอร์ส sysParmsFile

wakeupInterval

คลาส: WakeupInterval

ชนิดคลาส:

Int

ดีฟอลต์: 10

คำอธิบาย:

ถ้าผู้ใช้เลือกโหมด Command Line Login จากหน้าจอล็อกอิน, dtlogin จบการทำงาน X-server และอนุญาตให้พร้อมท์ล็อกอิน character-based ปกติ login: แสดงได้ถ้าผู้ใช้ไม่มิล็อกอินภายใน 2 คุณ wakeupInterval วินาที, X-server จะถูกรีสตาร์ท หลังจากผู้ใช้ล็อกอิน, dtlogin ตรวจสอบทุก wakeupInterval วินาทีเพื่อดูว่าผู้ใช้ล็อกเอาต์หรือไม่ ถ้าเป็นเช่นนั้น X-server ถูกรีสตาร์ทและหน้าจอล็อกอินถูกแสดง

## รีจอร์สจอแสดงผล

รีจอร์สจอแสดงผลคำสั่ง dtlogin สามารถถูกระบุสำหรับจอแสดงผลทั้งหมด หรือสำหรับเฉพาะจอแสดงผล เมื่อต้องการระบุจอแสดงผล ชื่อจอแสดงผล ถูกแทรกลงในชื่อรีจอร์สระหว่าง Dtlogin และ เชกเมนต์ชื่อรีจอร์สสุดท้าย ตัวอย่าง, Dtlogin.expo\_0.startup เป็นชื่อของรีจอร์สที่กำหนดไฟล์เซลล์เริ่มต้นบนจอแสดงผล expo:0 Resource manager แยกชื่อของรีจอร์สจากค่าด้วยเครื่องหมายโคลอน และแยกส่วนของชื่อรีจอร์สด้วยจุด ดังนั้น dtlogin จึงใช้ขีดเส้นใต้ (\_) สำหรับจุด (.) และเครื่องหมายโคลอน (:) เมื่อสร้างชื่อรีจอร์ส

รีจอร์สยังสามารถถูกระบุสำหรับคลาสของจอแสดงผลโดยการแทรกชื่อ คลาสแทนชื่อจอแสดงผล จอแสดงผลที่ไม่ได้ถูกจัดการโดย XDMCP สามารถมีคลาสที่ติดต่อยู่ที่ระบุในไฟล์ที่อ้างอิงโดยเซิร์ฟเวอร์รีจอร์ส จอแสดงผลใช้ XDMCP กำหนดคลาสที่ติดต่เป็นส่วนหนึ่งของ แพ็กเก็ต XDMCP

รีจอร์สทั่วไปของ dtlogin ต่อไปนี้ไม่ใช่ display-specific และใช้กับจอแสดงผลทั้งหมดเมื่อเหมาะสม

ไอเท็ม  
authorize

คำอธิบาย

ClassClass:

Authorize

ชนิดข้อมูล:

Boolean

ดีฟอลต์: False

คำอธิบาย:

Authorize เป็นรีซอร์ส Boolean ที่ควบคุมว่า dtlogin สร้าง และใช้การอนุญาตสำหรับการเชื่อมต่อเซิร์ฟเวอร์หรือไม่ อ้างอิงเพิ่มเติมที่รีซอร์ส authName

authName

ClassClass:

AuthName

ชนิดข้อมูล:

String

ดีฟอลต์: MIT-MAGIC-COOKIE-1

คำอธิบาย:

ถ้าใช้รีซอร์ส authorize , authName ระบุ ชนิดของการอนุญาตที่จะถูกใช้ ขณะนี้ dtlogin สนับสนุน การอนุญาต MIT-MAGIC-COOKIE-1 เท่านั้น สามารถสนับสนุน XDM-AUTHORIZATION-1 ได้ แต่ DES โดยทั่วไปไม่สามารถแจกจ่ายได้ การเชื่อมต่อ XDMCP ระบุชนิดการอนุญาต ที่สนับสนุนแบบไดนามิก, ดังนั้น authName จึงถูกละเว้นไปในกรณีนี้ อ้างอิงเพิ่มเติมที่รีซอร์ส authorize)

authFile

ClassClass:

AuthFile

ชนิดข้อมูล:

String

ดีฟอลต์: NULL

คำอธิบาย:

สื่อสารข้อมูลการอนุญาตจาก dtlogin ไปที่เซิร์ฟเวอร์โดยใช้ -auth อ็อปชันบรรทัดคำสั่งของเซิร์ฟเวอร์ เก็บรีซอร์สนี้ไว้ใน ไดเรกทอรีป้องกันการบันทึกเพื่อป้องกันการลบ ซึ่งจะปิดใช้กลไก การอนุญาตในเซิร์ฟเวอร์ ถ้าเป็น NULL, dtlogin จะสร้าง ชื่อไฟล์

chooser

ClassClass:

Chooser

ชนิดข้อมูล:

ดีฟอลต์:

คำอธิบาย:

ระบุการรันของโปรแกรมเพื่อสร้างเมนูโฮสต์สำหรับเคียวรีทางอ้อม ที่เปลี่ยนทิศทางไปที่ชื่อโฮสต์พิเศษ CHOOSER ดีฟอลต์คือ /usr/dt/bin/dtchooser ดูที่ส่วน Xaccess file

ไอเท็ม  
cpp

คำอธิบาย

ClassClass:

Cpp

ชนิดข้อมูล:

String

ดีฟอลต์: system dep.

คำอธิบาย:

ระบุพารามิเตอร์ของ C preprocessor ที่ถูกใช้โดย **xrdb**

environment

ClassClass:

Environment

ชนิดข้อมูล:

String

ดีฟอลต์: system dep.

คำอธิบาย:

มีชุดของคู่ *name=value* pairs ที่แยกโดย ช่องว่างหรือแท็บ แต่ละรายการถูกโหลดลงในสภาวะแวดล้อมของเซิร์ฟเวอร์และเซสชัน คู่ที่ส่วน Environment สำหรับข้อมูลเพิ่มเติม

failsafeClient

ClassClass:

FailsafeClient

ชนิดข้อมูล:

String

ดีฟอลต์: /system\_dep./xterm

คำอธิบาย:

ถ้าดีฟอลต์เซสชันล้มเหลวในการทำงาน, **dtlogin** จะย้อนกลับ มาที่โปรแกรมนี้ โปรแกรมนี้ถูกเรียกใช้โดยไม่มีอาร์กิวเมนต์แต่ทำงาน โดยใช้ตัวแปรสภาวะแวดล้อมเหมือนกับที่เซสชันมี

grabServer

ClassClass:

GrabServer

ชนิดข้อมูล:

Boolean

ดีฟอลต์: True

คำอธิบาย:

เพื่อเพิ่มความปลอดภัย, **dtlogin** ยึดข้อมูลเซิร์ฟเวอร์และคีย์บอร์ดไว้ขณะอ่านชื่อและรหัสผ่าน รีซอร์ส **grabServer** ระบุว่าเซิร์ฟเวอร์ควรถูกพักไว้ขณะทำการอ่านชื่อและรหัสผ่านหรือไม่ เมื่อเป็น FALSE, เซิร์ฟเวอร์จะไม่ถูกยึดไว้หลังจากการยึดคีย์บอร์ดสำเร็จ มิฉะนั้นเซิร์ฟเวอร์ จะถูกยึดไว้จนถึงก่อนการเริ่มต้นเซสชัน

ไอเท็ม  
grabTimeout

คำอธิบาย

ClassClass:

GrabTimeout

ชนิดข้อมูล:

Int

ดีฟอลต์: 3 seconds

คำอธิบาย:

ระยะเวลามากที่สุดที่ dtlogin จะรอให้การยึดสำเร็จ การยึดจะล้มเหลวได้ถ้าโคลเอ็นต์อื่นมีเซิร์ฟเวอร์ที่ยึดไว้ หรือเป็นไปได้ที่เวลาแฝงของเน็ตเวิร์กสูงมาก รีซอร์ส grabTimeout มีค่าดีฟอลต์ 3 วินาที ให้ใช้รีซอร์สนี้ด้วยความระมัดระวัง เนื่องจากผู้ใช้ถูก หลอกได้โดยหน้าต่างเลียนแบบบนจอแสดงผล ถ้าการยึดล้มเหลว, dtlogin จะจบการทำงานเซิร์ฟเวอร์และรีสตาร์ทเซิร์ฟเวอร์ (ถ้าเป็นไปได้) รวบรวมเซสชัน บาง X-terminals ไม่สามารถแสดงจอภาพลือกอื่น ขณะที่เซิร์ฟเวอร์ถูกยึด การตั้งค่า grabServer เป็น FALSE ทำให้จอภาพถูกแสดงได้ แต่ ี่มีความเป็นไปได้ที่ชื่อลือกอื่นของผู้ใช้จะถูกขโมยโดยการตัดลอก เนื้อหาของจอภาพลือกอื่น เนื่องจากคีย์บอร์ดยังคงถูกยึดและ รหัสผ่านไม่ถูกส่งกลับ, จึงไม่สามารถขโมยรหัสผ่านได้

language

ClassClass:

Language

ชนิดข้อมูล:

String

ดีฟอลต์: system dep.

คำอธิบาย:

ระบุค่ากำหนดดีฟอลต์สำหรับตัวแปรสภาวะแวดล้อม LANG ถ้าจอภาพ dtlogin ถูกแปลสำหรับภาษานั้น จะแสดงผลอย่างเหมาะสม มิฉะนั้น จะแสดงในภาษา C ผู้ใช้สามารถ เขียนทับค่าติดตั้งนี้ได้ชั่วคราวโดยใช้จอภาพลือกอื่น เมื่อเซสชันที่ตามมาจบการทำงาน ตัวแปร LANG กลับไปเป็น ค่าติดตั้งนี้

languageList

ClassClass:

LanguageList

ชนิดข้อมูล:

String

ดีฟอลต์: NULL

คำอธิบาย:

อนุญาตให้ผู้ใช้เขียนทับดีฟอลต์เซตของภาษาที่แสดงใน เมนู Language ของจอภาพลือกอื่น มีประโยชน์ ถ้าชุดของภาษา ถูกใช้จริงๆ กับจอแสดงผล มีค่าน้อยกว่าเซตที่ติดตั้งบน ระบบ ค่ารีซอร์สคือรายการของค่าที่ใช้ได้สำหรับตัวแปรสภาวะแวดล้อม LANG ค่า Language ควรถูกแยก โดยช่องว่างหรือแท็บ

ไอเท็ม  
languageName

คำอธิบาย

ClassClass:

LanguageName

ชนิดข้อมูล:

String

ดีฟอลต์: NULL

คำอธิบาย:

อนุญาตให้ผู้ใช้เขียนทับชื่อโลแคลดีฟอลต์ที่แสดงใน เมนู Language ของจอภาพล็อกอินด้วยข้อความอื่น ด้วยวิธีนี้ แทนที่ผู้ใช้จะเห็นรายการ En\_US ผู้ใช้จะเห็นรายการ English (United States) แทน รีซอร์สนี้ ถูกระบุเป็น `Dtlogin *local_name languageName: text` ดังนี้:

`Dtlogin*En_US.languageName: English (United States)`

`Dtlogin*Fr_CA.languageName: French (Canadian)`

openDelay

ClassClass:

OpenDelay

ชนิดข้อมูล:

Int

ดีฟอลต์: 5 seconds

คำอธิบาย:

ระบุระยะเวลา (เป็นวินาที) ระหว่างความพยายามครั้งต่อไปในการเปิด เซิร์ฟเวอร์ที่ไม่ตอบสนอง

openRepeat

ClassClass:

OpenRepeat

ชนิดข้อมูล:

Int

ดีฟอลต์: 5 seconds

คำอธิบาย:

ระบุจำนวนการพยายามต่อไปในการเปิดเซิร์ฟเวอร์ที่ไม่ตอบสนอง

openTimeout

ClassClass:

OpenTimeout

ชนิดข้อมูล:

Int

ดีฟอลต์: 30 seconds

คำอธิบาย:

ระบุระยะเวลาเพื่อรอ ในขณะที่มีการพยายามจริงในการเปิด เซิร์ฟเวอร์ที่ไม่ตอบสนอง เวลานี้เหมือนกับ เวลามากที่สุดที่ใช้ในการเรียกการทำงานระบบ `connect`

ไอเท็ม  
pingInterval

คำอธิบาย

ClassClass:

PingInterval

ชนิดข้อมูล:

Int

ดีฟอลต์: 5 minutes

คำอธิบาย:

เมื่อต้องการทราบเวลาที่จอแสดงผลรีโมตหายไป dtlogin จะทำการ ping เป็นระยะ, โดยใช้ X connection และส่งการร้องขอ XSync รีซอร์ส pingInterval ระยะเวลา (เป็นนาที) ระหว่างความพยายาม ping ครั้งต่อไป

pingTimeout

ClassClass:

PingTimeout

ชนิดข้อมูล:

int

ดีฟอลต์: 5 minutes

คำอธิบาย:

ระยะเวลาสูงสุด (เป็นนาที) สำหรับการตอบสนองของเทอร์มินัลต่อการร้องขอ ถ้าเทอร์มินัลไม่ตอบสนอง เซสชันจะถูกจบการทำงาน โคลเอ็นต์ dtlogin ไม่ ping ไปที่จอแสดงผลโลคัล เซสชันโลคัล ไม่ควรถูกจบการทำงาน เนื่องจากการรอของเซิร์ฟเวอร์ (สำหรับเซิร์ฟเวอร์ระบบ ไฟล์รีโมต, เป็นต้น) และไม่ตอบสนองต่อไป

reset

ClassClass:

Reset

ชนิดข้อมูล:

String

ดีฟอลต์: NULL

คำอธิบาย:

ระบุโปรแกรมที่ถูกรีบ (เป็น root) หลีกจากเซสชันสิ้นสุด ถ้ารีซอร์สนี้ไม่ถูกเซต จะไม่มีโปรแกรมรัน ชื่อทั่วไปคือ Xreset ดูที่ Xreset File

resetForAuth

ClassClass:

ResetForAuth

ชนิดข้อมูล:

Boolean

ดีฟอลต์: False

คำอธิบาย:

ระหว่างการนำการอนุญาตมาใช้ครั้งแรกในเซิร์ฟเวอร์ตัวอย่าง, ไฟล์การอนุญาตถูกอ่านซ้ำ เมื่อเซิร์ฟเวอร์รีเซต ไม่ใช่เมื่อมีการตรวจสอบ การเชื่อมต่อเริ่มต้น เนื่องจาก dtlogin จากข้อมูลการอนุญาต ก่อนเชื่อมต่อไปที่จอแสดงผล เซิร์ฟเวอร์เก่าไม่ได้รับ ข้อมูลการอนุญาตปัจจุบัน รีซอร์สนี้ทำให้ dtlogin ส่ง SIGHUP ไปที่เซิร์ฟเวอร์หลังจากการตั้งค่าไฟล์ ทำให้มีการรีเซตเซิร์ฟเวอร์เพิ่มเติม ระยะเวลาที่ข้อมูลการอนุญาต ถูกอ่าน

ไอเท็ม  
resetSignal

คำอธิบาย

ClassClass:

Signal

ชนิดข้อมูล:

Int

ดีฟอลต์: 1 SIGHUP

คำอธิบาย:

ระบุสัญญาณที่ dtlogin ส่งเพื่อรีเซ็ตเซิร์ฟเวอร์

resources

ClassClass:

Resource

ชนิดข้อมูล:

String

ดีฟอลต์: NULL

คำอธิบาย:

ระบุชื่อไฟล์ที่จะถูกโหลดโดย xrdp เป็น รีซอร์สฐานข้อมูลกับหน้าต่าง root ของจอภาพ 0 ของจอแสดงผล รีซอร์สฐานข้อมูลนี้ ถูกโหลดก่อนขั้นตอนการพิสูจน์ตัวตนเริ่มขึ้น ดังนั้น จึงสามารถควบคุมการแสดงผลของหน้าต่างสื่ออื่น ดูส่วนจอภาพ การพิสูจน์ตัวตน ซึ่งอธิบายรีซอร์สต่างๆ ที่เหมาะสมกับตำแหน่งในไฟล์นี้ ไม่มีค่าดีฟอลต์สำหรับรีซอร์สนี้ แต่ชื่อปกติคือ Xresources

session

ClassClass:

Session

ชนิดข้อมูล:

String

ดีฟอลต์: /usr/dt/bin/Xsession

คำอธิบาย:

ระบุระบุที่จะถูกดำเนินการสำหรับผู้ใช้ที่ได้รับการพิสูจน์ตัวตน โดย ดีฟอลต์ไฟล์ /usr/dt/bin/Xsession จะถูกรัน ชื่อทั่วไปคือ Xsession อ้างอิงถึงไฟล์ the Xsession

setup

ClassClass:

Setup

ชนิดข้อมูล:

String

ดีฟอลต์: NULL

คำอธิบาย:

ระบุโปรแกรมที่ถูกรัน (เป็น root) ก่อนการแสดงจอภาพ การพิสูจน์ตัวตน โดยค่าดีฟอลต์ไม่มีโปรแกรมถูกรัน ชื่อทั่วไปคือ Xsetup อ้างอิงถึงไฟล์ the Xsetup

ไอเท็ม  
startAttempts

คำอธิบาย

ClassClass:

StartAttempts

ชนิดข้อมูล:

Int

ดีฟอลต์: 4

คำอธิบาย:

รีซอร์สตัวเลขควบคุมการทำงานของ dtlogin เมื่อพยายาม เปิดเซิร์ฟเวอร์ที่ไม่ตอบสนอง: **openDelay**, **openRepeat**, **openTimeout**, และ **startAttempts** รีซอร์สนี้ระบุจำนวนครั้งที่กระบวนการทั้งหมดเกิดขึ้น ก่อนการหยุดติดต่อเซิร์ฟเวอร์ หลังจาก **openRepeat** ทำงานแล้ว หรือถ้า **openTimeout** วินาทีผ่านไปในการพยายาม, **dtlogin** จบการทำงานและรีสตาร์ทเซิร์ฟเวอร์ แล้วพยายามเชื่อมต่ออีกครั้ง กระบวนการนี้ถูกทำซ้ำ **startAttempts** ครั้ง, ในจุดที่จอสถงผลถูกประกาศว่าไม่ตอบสนองหรือถูกปิดใช้งาน

startup

ClassClass:

Startup

ชนิดข้อมูล:

String

ดีฟอลต์: NULL

คำอธิบาย:

ระบุโปรแกรมที่ถูกรัน (เป็น root) หลังจากกระบวนการการพิสูจน์ตัวตน สำเร็จ โดยดีฟอลต์ ไม่มีโปรแกรมถูกรัน ชื่อทั่วไปสำหรับ ไฟล์ที่ใช้ที่นี่คือ **Xstartup** ดูที่ส่วน **Xstartup file**

systemPath

ClassClass:

SystemPath

ชนิดข้อมูล:

String

ดีฟอลต์: system\_dep.\_path

คำอธิบาย:

โคลเอ็นต์ **dtlogin** เช็ตตัวแปรสภาวะแวดล้อม **PATH** สำหรับการเริ่มต้นและรีเช็ตสคริปต์เป็นค่าของรีซอร์สนี้ หมายเหตุ ไม่มี ". " จาก entry นี้เป็นแนวปฏิบัติที่ดีในการติดตาม root เนื่องจากหลีกเลี่ยงรูปแบบการเจาะระบบ

systemShell

ClassClass:

SystemShell

ชนิดข้อมูล:

String

ดีฟอลต์: /bin/sh

คำอธิบาย:

โคลเอ็นต์ **dtlogin** เช็ตตัวแปรสภาวะแวดล้อม **SHELL** สำหรับการเริ่มต้นและรีเช็ตสคริปต์เป็นค่าของรีซอร์สนี้

ไอเท็ม  
terminateServer

คำอธิบาย

ClassClass:

TerminateServer

ชนิดข้อมูล:

Boolean

ดีฟอลต์: False

คำอธิบาย:

ระบุว่า X ควรถูกจบการทำงานหรือไม่ เมื่อเซสชันสิ้นสุด (แทนการรีเซ็ต) อีพซันนี้สามารถใช้ถ้าเซิร์ฟเวอร์มีแนวโน้มจะขยายอย่างไม่จำกัดเมื่อเวลาผ่านไป เพื่อจำกัดจำนวนของเวลาที่เซิร์ฟเวอร์ถูกรันอย่างต่อเนื่อง

termSignal

ClassClass:

Signal

ชนิดข้อมูล:

Int

ดีฟอลต์: 15 (SIGTERM)

คำอธิบาย:

ระบุสัญญาณ dtlogin ที่ส่งไปเพื่อจบการทำงานเซิร์ฟเวอร์

userAuthDir

ClassClass:

UserAuthDir

ชนิดข้อมูล:

String

ดีฟอลต์: /var/dt

คำอธิบาย:

เมื่อ dtlogin ไม่สามารถเขียนไปที่ไฟล์การอนุญาตของผู้ใช้ (\$HOME/.Xauthority), จะสร้างชื่อไฟล์เฉพาะในไดเรกทอรีนี้ และชี้ไปที่ตัวแปรสภาวะแวดล้อม XAUTHORITY ที่ไฟล์ที่สร้าง

userPath

ClassClass:

UserPath

ชนิดข้อมูล:

String

ดีฟอลต์: system\_dep.\_path

คำอธิบาย:

โคลเอ็นต์ dtlogin เซ็ตตัวแปรสภาวะแวดล้อม PATH สำหรับเซสชันเป็นค่านี้ ซึ่งควรเป็นรายการไดเรกทอรีที่ค้นด้วยโคลอน

ไอเท็ม  
xdmMode

คำอธิบาย

ClassClass:

XdmMode

ชนิดข้อมูล:

Boolean

ดีฟอลต์: False

คำอธิบาย:

ถ้าเป็น True, ไฟล์ `$HOME/.xsession` จะถูกเรียกใช้จาก `Xsession` ขณะการพิสูจน์ตัวตนผู้ใช้ แทนการเรียกใช้จาก `dtssession`

xrdb

ClassClass:

Xrdb

ชนิดข้อมูล:

String

ดีฟอลต์: `/system_dep./xrdb`

คำอธิบาย:

ระบุโปรแกรมที่ใช้โหลดรีซอร์ส จอภาพการพิสูจน์ตัวตน อ่านค่า `name-password` จากคีย์บอร์ด เนื่องจากนี้เป็นโคลเอ็นต์ทุกคิต Motif, ซี, ฟอนต์ และอ็อปชันโครงร่างบางส่วนสามารถถูกควบคุมได้ด้วยรีซอร์สรีซอร์สที่ส่วไปสำหรับจอภาพควรถูกนำไปไว้ในไฟล์ตั้งชื่อโดยรีซอร์ส resources (Xresources เป็นดีฟอลต์) ระบุค่า language-specific เช่นข้อความ หรือฟอนต์ ในไฟล์ `Dtlogin app-defaults`

## Logo Resources

ดีฟอลต์ตราสัญลักษณ์บนจอภาพการพิสูจน์ตัวตนสามารถถูกแทนที่ด้วยบิตแม็พ หรือพิกซ์แม็พของอ็อปชันของผู้ใช้รีซอร์ส ควรถูกนำหน้าด้วยสตริง `Dtlogin*logo*` เมื่อมีการระบุ

ไอเท็ม  
bitmapFile

คำอธิบาย

ClassClass:

BitmapFile

ชนิดข้อมูล:

String

ดีฟอลต์: NULL

คำอธิบาย:

ระบุชื่อพารามิเตอร์ไปที่ไฟล์บิตแม็พหรือพิกซ์แม็พที่จะถูกใช้ เป็นตราสัญลักษณ์

แบ็กกราวน์

ClassClass:

Background

ชนิดข้อมูล:

Pixel

ดีฟอลต์: `#a8a8a8`

คำอธิบาย:

ระบุสีแบ็กกราวน์สำหรับตราสัญลักษณ์

ไอเท็ม  
topShadowPixmap

คำอธิบาย  
ClassClass:  
topShadowPixmap  
ชนิดข้อมูล:  
String  
ดีฟอลต์: 25\_foreground  
คำอธิบาย:  
ระบุพิกซ์แมปที่จะใช้สำหรับเงาเส้นขอบตราสัญลักษณ์

รีซอร์สต่อไปนี้อธิบายสตริงต้อนรับที่ขึ้นหน้าจอ ล็อกอิน รีซอร์สควรถูกนำหน้าด้วยสตริง Dtlogin\*greeting\* เมื่อมีการ  
ระบุ

ไอเท็ม  
พื้นหน้า

คำอธิบาย  
ClassClass:  
Foreground  
ชนิดข้อมูล:  
Pixel  
ดีฟอลต์: black  
คำอธิบาย:  
ระบุสีพื้นหน้าสำหรับข้อความยินดีต้อนรับ

แบ็กกราวนด์

ClassClass:  
Background  
ชนิดข้อมูล:  
Pixel  
ดีฟอลต์: dynamic  
คำอธิบาย:  
ระบุสีส่วนหลังสำหรับข้อความยินดีต้อนรับ สีดีฟอลต์ คือสีเทาจางสำหรับระบบสี หรือสีขาวสำหรับระบบ  
โมนโอโครม

fontlist

ClassClass:  
FontList  
ชนิดข้อมูล:  
FontList  
ดีฟอลต์: -\*-\*schoolbook-medium-i-normal--18-\*  
คำอธิบาย:  
ระบบฟอนต์ที่จะใช้สำหรับข้อความยินดีต้อนรับ

ไอเท็ม  
labelString

คำอธิบาย

ClassClass:

LabelString

ชนิดข้อมูล:

String

ดีฟอลต์: Welcome to %LocalHost%

คำอธิบาย:

ระบสตริงที่จะใช้สำหรับข้อความยินดีต้อนรับ สามารถระบุบรรทัด หลายบรรทัดได้โดยการรวมอักขระขึ้นบรรทัดใหม่ (0 ในข้อความ ถ้าโทเค็น %LocalHost% ถูกรวมไว้ในข้อความ จะถูกแทนที่ด้วยชื่อของโฮสต์ที่จัดเตรียมล็อกอินเซอร์วิส ถ้าโทเค็น %DisplayName% ถูกรวมไว้ในข้อความ จะถูกแทนที่ด้วยชื่อจอแสดงผล

perLabelString

ClassClass:

LabelString

ชนิดข้อมูล:

String

ดีฟอลต์: Welcome %s

คำอธิบาย:

ระบสตริงที่จะใช้สำหรับข้อความยินดีต้อนรับที่เป็นส่วนบุคคลนี้เป็นข้อความที่แสดงหลังจากชื่อผู้ใช้ได้ถูกป้อน %s จะถูกแทนที่ด้วยชื่อผู้ใช้ที่ป้อน

alignment

ClassClass:

Alignment

ชนิดข้อมูล:

String

ดีฟอลต์: ALIGNMENT\_CENTER

คำอธิบาย:

ระบสตริงที่จะใช้สำหรับการจัดตำแหน่งข้อความยินดีต้อนรับ ค่าที่ใช้ได้คือ ALIGNMENT\_BEGINNING, ALIGNMENT\_CENTER และ ALIGNMENT\_END

## รีซอร์ส Matte

รีซอร์สต่อไปนี้อธิบายโครงสร้าง matte ที่ใช้บนหน้าจอล็อกอิน รีซอร์สควรถูกนำหน้าด้วย Dtlogin\*matte. string เมื่อมีการระบุ

ไอเท็ม  
ความกว้าง

คำอธิบาย

ClassClass:

Width

ชนิดข้อมูล:

Int

ดีฟอลต์: 806 สำหรับการแสดงผลความละเอียดสูง 755 สำหรับการแสดงผลความละเอียดปานกลาง 585 สำหรับการแสดงผลความละเอียดต่ำ

คำอธิบาย:

ระบุความกว้างที่ใช้สำหรับ login\_matte

height

ClassClass:

Height

ชนิดข้อมูล:

Int

ดีฟอลต์: 412 สำหรับการแสดงผลความละเอียดสูง 385 สำหรับการแสดงผลความละเอียดปานกลาง 300 สำหรับการแสดงผลความละเอียดต่ำ

คำอธิบาย:

ระบุความสูงที่ใช้สำหรับ login\_matte

## รีซอร์สเลเบล

รีซอร์สต่อไปนี้คืออธิบายโครงสร้างพอนต์ที่ใช้บนหน้าจอล็อกอิน รีซอร์สควรถูกนำหน้าด้วย string Dtlogin\*. เมื่อมีการระบุ

ไอเท็ม  
labelFont

คำอธิบาย

ClassClass:

LabelFont

ชนิดข้อมูล:

String

ดีฟอลต์: \*-swiss 742-medium-r-normal-\* 140-\* -p-110-\* for high-resolution displays \*-swiss 742-bold-r-normal-\* 140-\* -p-100-\* for low-resolution displays

คำอธิบาย:

ระบุ labelFont ที่จะใช้สำหรับปุ่มกดและเลเบล

textFont

ClassClass:

TextFont

ชนิดข้อมูล:

String

ดีฟอลต์: \*-prestige-medium-r-normal-\* 128-72-\* for high-resolution displays \*-helvetica-bold-r-normal-\* 100-\* for low-resolution displays

คำอธิบาย:

ระบุ textFont ที่จะใช้สำหรับปุ่มกดและเลเบล

## แฟล็ก

แฟล็กทั้งหมดยกเว้น `-config`, ระบุค่าที่สามารถถูกระบุได้เช่นกัน ในไฟล์คอนฟิกูเรชันเป็นรีซอร์ส โดยปกติ การกำหนดเองทำได้โดยใช้ไฟล์คอนฟิกูเรชันไม่ใช่ข้อบกพร่องคำสั่ง แฟล็กเหล่านี้มีประโยชน์ มากที่สุดสำหรับการดีบั๊กและทดสอบ one-shot

### ไอเท็ม

`-config configuration_file`

`-daemon`

`-debug debug_level`

`-error error_log_file`

`-nodaemon`

`-resources resource_file`

`-server server_entry`

`-udpPort port_number`

`-session session_program`

### คำอธิบาย

ระบุรีซอร์สไฟล์ที่ระบุ พารามิเตอร์คอนฟิกูเรชันที่เหลือ ซึ่งแทนที่ไฟล์ `dtlogin default Xconfig` ดูที่ส่วน ไฟล์ Xconfig สำหรับข้อมูลเพิ่มเติม

ระบุ `true` เป็นค่าสำหรับรีซอร์ส `daemonMode` ทำให้ `dtlogin` ปิดไฟล์ descriptors ทั้งหมด, ยกเลิกการเชื่อมโยงเทอร์มินัลการควบคุม และกำหนดตัวเองไปอยู่ในแบ็กกราวนด์เมื่อเริ่มต้นครั้งแรก (เหมือนโฮสต์ของ daemons อื่น)

ระบุค่าตัวเลขสำหรับรีซอร์ส `debug_level` ค่าที่ไม่ใช่ศูนย์ทำให้ `dtlogin` พิมพ์ข้อความการดีบั๊ก ไปที่เทอร์มินัล; และยังปิดใช้งานรีซอร์ส `daemonMode` บังคับให้ `dtlogin` รันพร้อมกัน

ระบุค่าสำหรับรีซอร์ส `error_log_file` ดูที่ส่วนไฟล์ Xerrors สำหรับข้อมูลเพิ่มเติม

ระบุ `false` เป็นค่าสำหรับรีซอร์ส

ระบุค่าสำหรับรีซอร์ส `resource_log_file` ดูที่ส่วน Xresources file สำหรับข้อมูลเพิ่มเติม

ระบุค่าสำหรับรีซอร์ส `server_entry` ดูที่ส่วน Xservers file สำหรับข้อมูลเพิ่มเติม

ระบุค่าสำหรับรีซอร์ส `requestPort` ซึ่งจะเซตหมายเลขพอร์ตที่ `dtlogin` มอนิเตอร์สำหรับการร้องขอ XDMCP เนื่องจาก XDMCP ใช้ udp port 177 ลง

ทะเบียนที่รู้จักดี, ใหญ่เสี่ยง การเปลี่ยนรีซอร์สนั้นนอกจากเพื่อการดีบั๊ก

ระบุค่าสำหรับรีซอร์ส `session_program` ดูที่ส่วน Xconfig file สำหรับข้อมูลเพิ่มเติม

## ตัวแปรสภาวะแวดล้อม

คำสั่ง `dtlogin` เรียกเซสชันของผู้ใช้ด้วยสภาวะแวดล้อมดีฟอลต์ ต่อไปนี้:

### ไอเท็ม

`DISPLAY`

`EDITOR`

`HOME`

`KBD_LANG`

`LANG`

`LC_ALL`

`LC_MESSAGES`

`LOGNAME`

`MAIL`

`PATH`

`USER`

`SHELL`

`TERM`

`TZ`

`XAUTHORITY`

### คำอธิบาย

เซตเป็นชื่อจอแสดงผลที่เชื่อมโยง

เซตเป็น `/usr/dt/bin/dtpad`

เซตเป็นโฮมไดเรกทอรีของผู้ใช้

เซตเป็นค่าของ `LANG` สำหรับภาษาที่ใช้ได้

เซตเป็นภาษา NLS ปัจจุบัน (ถ้ามี)

เซตเป็นภาษา NLS ปัจจุบัน (ถ้ามี)

เซตเป็นภาษา NLS ปัจจุบัน (ถ้ามี)

เซตเป็นชื่อผู้ใช้

เซตเป็น `/usr/mail/$USER` (ขึ้นกับระบบ)

เซตเป็นค่าของรีซอร์ส `userPath`

เซตเป็นชื่อผู้ใช้

เซตเป็นเชลล์ดีฟอลต์ของผู้ใช้ (จาก `/etc/passwd`)

เซตเป็น `dtterm`

เซตเป็นค่าของรีซอร์ส `timeZone` หรือ ค่าดีฟอลต์ระบบ

เซตเป็นไฟล์ลิวทิ

## การเพิ่มข้อมูลในรายการสภาวะแวดล้อม

มีสี่เมธอดในการแก้ไขหรือเพิ่มรายการก่อนหน้าขึ้นอยู่กับขอบเขตที่ต้องการของตัวแปรสภาวะแวดล้อมผลลัพธ์:

- รีซอร์ส `exportList` มีเพื่ออนุญาตการเอ็กซ์พอร์ต ตัวแปรที่เตรียมไว้ให้กับกระบวนการ `dtlogin` โดย `parent` ของ `dtlogin` ตัวแปรที่ระบุโดยเมธอดนี้พร้อมใช้กับทั้งกระบวนการ `X server` ของ จอแสดงผลและเซสชันผู้ใช้ และเขียนทับค่ากำหนดดีฟอลต์ที่มี รีซอร์สรับสตริงของ `name=value` ที่ค้นโดยอย่างน้อยหนึ่งช่องว่างหรือแท็บ
- รีซอร์ส `environment` มีในไฟล์คอนฟิกูเรชัน `dtlogin` เพื่ออนุญาตการตั้งค่าตัวแปรสภาวะแวดล้อมในแบบโกลบอลหรือต่อจอแสดงผล ตัวแปรที่ระบุโดยเมธอดนี้พร้อมใช้กับทั้งกระบวนการ `X server` ของ จอแสดงผลและเซสชันผู้ใช้ และเขียนทับค่ากำหนดดีฟอลต์ที่มี รีซอร์สรับสตริงของ `name=value` ที่ค้นโดยอย่างน้อยหนึ่งช่องว่างหรือแท็บ ค่าที่ระบุต้องเป็นค่าคงที่ เนื่องจาก ไม่มีการใช้เซลล์ในการวิเคราะห์ค่าสตริง ตัวอย่างเช่น:

```
Dtlogin*environment:MAIL_HOST=blanco MAIL_SERVER=pablo
```

**หมายเหตุ:** ตัวแปรสภาวะแวดล้อม `LANG` และ `TZ` มีรีซอร์สกำหนดให้เป็นของตัวเอง ในไฟล์คอนฟิกูเรชัน และไม่ควรถูกเซตโดยสภาวะแวดล้อม

- ตัวแปรสภาวะแวดล้อมที่ต้องการการประมวลผลโดยเซลล์หรือขึ้นกับ ค่าของตัวแปรสภาวะแวดล้อมอื่น สามารถถูกระบุได้ในสคริปต์ เริ่มทำงาน `Xsession` ตัวแปรถูกโหลกลงในสภาวะแวดล้อม ของผู้ใช้ทั้งหมดบนจอแสดงผล แต่ไม่ใช้กับกระบวนการ `X server` โดยเขียนทับ ค่าที่ตั้งก่อนหน้าของตัวแปรเดียวกัน สคริปต์ `Xsession` รับไวยากรณ์ `ksh` สำหรับการตั้งค่าตัวแปรสภาวะแวดล้อม ตัวอย่างเช่น:

```
MAIL=/usr/mail/$USER
```

- ตัวแปรสภาวะแวดล้อมส่วนบุคคลสามารถถูกเซตในแบบต่อผู้ใช้ในสคริปต์ไฟล์ `$HOME/.dtprofile` คำสั่ง `dtlogin` รับไวยากรณ์ `sh`, `ksh` หรือ `csh` สำหรับคำสั่งในไฟล์นี้ คำสั่งควรเป็นเพียงคำสั่งที่เซตตัวแปรสภาวะแวดล้อม ไม่ใช้คำสั่งที่ดำเนินการเทอร์มินัล I/O, ยกเว้น `tset` หรือ `stty` ถ้าบรรทัดแรกของ `.dtprofile` คือ `#!/bin/sh`, `#!/bin/ksh` หรือ `#!/bin/csh`, `dtlogin` ใช้เซลล์ที่เหมาะสม ในการวิเคราะห์ค่า `.dtprofile` มิฉะนั้น จะใช้ดีฟอลต์เซลล์ของผู้ใช้ (`$SHELL`)

## สถานะออก

ค่าออกต่อไปนี้ถูกส่งคืน:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการสตาร์ท CDE ล็อกอินเซอร์วิสเป็น daemon, ให้ป้อน:
 

```
/usr/dt/bin/dtlogin -daemon
```
2. เมื่อต้องการสตาร์ท CDE ล็อกอินเซอร์วิสในดีบั๊กโหมด ให้ป้อน:
 

```
/usr/dt/bin/dtlogin -debug 1
```

## Location

`/usr/dt/bin/dtlogin`

## ข้อผิดพลาดมาตรฐาน

คำสั่ง `dtlogin` ส่งคืนข้อความแสดงความผิดพลาดต่อไปนี้:

- ล็อกอินไม่ถูกต้อง โปรดลองอีกครั้ง

- ไม่สามารถเปลี่ยนโฮมไดเรกทอรี
- ขอบข่าย จำนวนสูงสุดของผู้ใช้ได้ล็อกอินอยู่แล้ว
- ข้อผิดพลาดการล็อกอิน ID ผู้ใช้ไม่ถูกต้อง
- ข้อผิดพลาดการล็อกอิน ID กลุ่มไม่ถูกต้อง
- ข้อผิดพลาดการล็อกอิน audit ID ไม่ถูกต้อง
- ข้อผิดพลาดการล็อกอินแฟล็ก audit ไม่ถูกต้อง
- ล็อกอินถูกปิดใช้งานในขณะนี้
- รหัสผ่านปัจจุบันของคุณหมดอายุแล้ว

## ไฟล์

คำสั่ง **dtlogin** ถูกออกแบบให้ทำงานในสภาวะแวดล้อมที่หลากหลายและจัดเตรียมชุดของไฟล์คอนฟิกูเรชันที่เปลี่ยนแปลงได้ เพื่อให้เหมาะสมกับเฉพาะระบบ ไฟล์คอนฟิกูเรชัน **dtlogin** ดีฟอลต์ พบได้ใน `/usr/dt/config` ยกเว้น `Xsession`, ซึ่งถูกเก็บใน `/usr/dt/bin` มีดังต่อไปนี้:

### ไอเท็ม

`/usr/dt/config/Xconfig`  
`/usr/dt/config/Xaccess`  
`/usr/dt/config/Xservers`  
`/usr/dt/config/Xresources`  
`/usr/dt/config/Xsetup`  
`/usr/dt/config/Xstartup`  
`/usr/dt/bin/Xsession`  
`/usr/dt/config/Xfailsafe`  
`/usr/dt/config/Xreset`

### คำอธิบาย

ระบุไฟล์คอนฟิกูเรชัน **dtlogin** และการทำงาน **dtlogin** อื่น  
 ควบคุมการเข้าถึงจากเซอวิส XDMCP ของการร้องขอแสดงผล  
 มีรายการของจอแสดงผลสำหรับ **dtlogin** เพื่อจัดการ  
 มินิยามรีซอร์สที่ระบุลักษณะที่ปรากฏ ของจอภาพล็อกอิน  
 สคริปต์ทำงานในฐานะ root ก่อนที่จะแสดงจอภาพล็อกอิน  
 สคริปต์ทำงานเป็น root หลังจากผู้ใช้ได้รับการรับรองสมบูรณ์  
 สคริปต์ทำงานในฐานะผู้ใช้ที่ได้รับการรับรองซึ่งเริ่มต้น เซสชันของผู้ใช้  
 สคริปต์ทำงานในฐานะผู้ใช้ที่ได้รับการรับรองซึ่งเริ่มต้น เซสชัน fail-safe  
 สคริปต์ทำงานในฐานะ root หลังจากเซสชันผู้ใช้ สิ้นสุด

## ไฟล์ Xconfig

ไฟล์ **Xconfig** มีรีซอร์สทั่วไปสำหรับ **dtlogin** และอยู่ที่ด้านบนของแผนผังไฟล์คอนฟิกูเรชัน **dtlogin Xconfig** ระบุตำแหน่งของไฟล์คอนฟิกูเรชันและล็อกไฟล์ **dtlogin** อื่น และระบุลักษณะการทำงาน **dtlogin** ตำแหน่งของคอนฟิกูเรชัน **dtlogin** และล็อกไฟล์อื่นถูกระบุโดยมินิยามรีซอร์ส ดีฟอลต์มีดังนี้:

### **Dtlogin.errorLogFile**

`/var/dt/Xerrors`

### **Dtlogin.pidFile**

`/var/dt/Xpid`

### **Dtlogin.accessFile**

`Xaccess`

### **Dtlogin.servers**

`Xservers`

### **Dtlogin\*resources**

`%L/Xresources`

**Dtlogin\*setup**

Xsetup

**Dtlogin\*startup**

Xstartup

**Dtlogin\*reset**

Xreset

**Dtlogin\*failsafeClient**

Xfailsafe

**Dtlogin\*session**

/usr/dt/bin/Xsession

ถ้าพาดที่ระบุสำหรับรีซอร์ส **accessFile**, **servers**, **resources**, **setup**, **startup**, **reset**, **failsafeClient**, หรือ **session** สัมพันธ์กันชั้นแรก **dtlogin** จะค้นหา ไฟล์ในไดเรกทอรี **/etc/dt/config** จากนั้น **/usr/dt/config**

หมายเหตุ: รีซอร์สบางส่วนถูกระบุโดยมี \* แยก คอมโพเนนต์รีซอร์สเหล่านี้สามารถทำให้ไม่ซ้ำกันได้สำหรับแต่ละจอแสดงผลที่ต่างกัน โดยการแทนที่ \* ด้วยชื่อจอแสดงผล อ้างอิงถึง รีซอร์สจอแสดงผล สำหรับข้อมูลเพิ่มเติม

ดีโฟลต์ไฟล์ **Xconfig** คือ **/usr/dt/config/Xconfig** ผู้ดูแลระบบ สามารถกำหนด **Xconfig** เองได้คัดลอก **/usr/dt/config/Xconfig** ไปที่ **/etc/dt/config/Xconfig** และแก้ไข **/etc/dt/config/Xconfig** ดีโฟลต์ไฟล์ **Xconfig** มีรายการคอนฟิกูเรชัน และลิสต์ออกไฟล์ก่อนหน้าเพิ่มเติมจากนิยามรีซอร์ส และตัวอย่างที่จำเพาะกับคู่คีย์บางส่วน

## ไฟล์ Xaccess

ไฟล์ฐานข้อมูลที่ระบุโดยรีซอร์ส **accessFile** มีข้อมูล ซึ่ง **dtlogin** ใช้เพื่อควบคุมการเข้าถึงจากจอแสดงผลร้องขอ เซอร์วิส XDMCP ไฟล์นี้มี entry สามชนิด: entry ซึ่งควบคุม การตอบสนองกับเคียวรี Direct และ Broadcast, entry ซึ่งควบคุมการตอบสนอง กับเคียวรี Indirect และ macro definitions

รูปแบบของ Direct entry เป็นชื่อโฮสต์หรือแพตเทิร์น แพตเทิร์น ถูกแยกออกจากชื่อโฮสต์โดยการรวมอักขระ meta (\* จับคู่กับลำดับ 0 หรืออักขระอื่นๆ , และ ? จับคู่กับอักขระเดียว) ซึ่งถูกเปรียบเทียบกับ ชื่อโฮสต์ของอุปกรณ์จอแสดงผล ถ้า entry เป็นชื่อโฮสต์ การเปรียบเทียบทั้งหมด กระทำโดยใช้เน็ตเวิร์กแอดเดรส ดังนั้นชื่อที่แปลงไปเป็นเน็ตเวิร์กแอดเดรส ที่ถูกต้องสามารถนำมาใช้ได้ สำหรับแพตเทิร์น ชื่อโฮสต์ซึ่งเป็นที่ยอมรับเท่านั้น ที่ใช้ได้ในการเปรียบเทียบ ดังนั้นตรวจสอบว่าคุณไม่ได้พยายามจับคู่กับนามแฝง การใส่ เครื่องหมายตกใจ (!) ก่อนชื่อโฮสต์หรือแพตเทิร์น ทำให้โฮสต์ที่ตรงกับ entry นั้นถูกแยกออก

entry ทางอ้อมยังมีชื่อโฮสต์หรือแพตเทิร์น แต่ตามด้วย รายการของชื่อโฮสต์หรือแมโครซึ่งเคียวรีทางอ้อมควรถูกส่งไป entry ทางอ้อมยังสามารถระบุ **dtlogin** รัน **dtchooser** เพื่อเสนอเมนูของโฮสต์กับจอภาพล็อกอินที่แสดงได้

macro definition มีชื่อแมโครและรายการของชื่อโฮสต์และ แมโครอื่นที่แมโครขยายไปถึง เมื่อต้องการแยกแมโครจากชื่อโฮสต์ ให้กำหนดชื่อแมโครเริ่มต้นด้วยอักขระ % แมโครสามารถถูกซ้อนกันได้

เมื่อการเข้าถึงโฮสต์จ้อแสดงผลถูกตรวจสอบ แต่ละ entry ถูก สแกนเป็นลำดับ และ entry แรกที่ตรงจะกำหนดการตอบสนอง Direct และ Broadcast entries ถูกละเว้นเมื่อทำการสแกนหา Indirect entry และในทางกลับกัน บรรทัดเปล่าถูกละเว้น, # จะถือ ว่าเป็นตัวคั่นหมายเหตุ ทำให้ส่วนที่เหลือของบรรทัดถูกละเว้น และ \newline ทำให้ newline ถูกละเว้น, ทำให้โฮสต์ทางอ้อม ขยายหลายบรรทัดได้

ตัวอย่างต่อไปนี้แสดงไฟล์ Xaccess:

```
#
# Xaccess - XDMCP access control file
#

#
# Direct/Broadcast query entries
#
!extra.lcs.mit.edu # disallow direct/broadcast service for extra
bambi.ogi.edu     # allow access from this particular display
*.lcs.mit.edu     # allow access from any display in LCS

#
# Indirect query entries
#

#define %HOSTS macro
%HOSTS             expo.lcs.mit.edu xenon.lcs.mit.edu \
                   excess.lcs.mit.edu kanga.lcs.mit.edu

#force extract to contact xenon
extract.lcs.mit.edu xenon.lcs.mit.edu

#disallow indirect access by extra
!extra.lcs.mit.edu dummy

#all others get to choose among %HOSTS
*.lcs.mit.edu      %HOSTS
```

ถ้าได้รับการเข้าถึง XDMCP ไฟล์ชั่วคราวสามารถถูกสร้างได้ในไคเร็กทอรีที่ระบุโดย `authDir` ซึ่งมีข้อมูลการอนุญาตสำหรับ X-terminal จะถูกลบเมื่อเซสชันสตาร์ท

สำหรับ X terminals ที่ไม่มีโฮสต์เมนูเพื่อใช้กับ Broadcast Indirect เคียววี, โปรแกรม `chooser` สามารถดำเนินการให้ได้ในไฟล์ `Xaccess` ระบุ `CHOOSE` เป็น รายการแรกในรายการโฮสต์ทางอ้อม โปรแกรม `chooser` ส่งการร้องขอ Query ไปที่แต่ละชื่อโฮสต์ ที่เหลือในรายการและแสดงเมนูของโฮสต์ทั้งหมดที่ตอบสนอง รายการอาจมีคำว่า `BROADCAST`, ซึ่ง `chooser` จะส่ง Broadcast แทน, และแสดงเมนูของ โฮสต์ทั้งหมดที่ตอบสนองอีกครั้ง ในบางระบบปฏิบัติการ, แพ็กเก็ต UDP ไม่สามารถถูก broadcast ได้, ดังนั้นคุณลักษณะจะไม่ทำงาน

ตัวอย่างของไฟล์ `Xaccess` ที่ใช้โปรแกรม `chooser` มีดังนี้:

```
#offer a menu of these hosts to extract
extract.lcs.mit.edu CHOOSE %HOSTS

#offer a menu of all hosts to extra
extra.lcs.mit.edu    CHOOSE BROADCAST
```

โปรแกรมที่จะใช้สำหรับ chooser ถูกระบุโดยรีซอร์ส chooser รีซอร์สสำหรับโปรแกรมนี้สามารถนำไปไว้ในไฟล์ที่กำหนดชื่อ โดยรีซอร์ส ดีฟอลต์ Xaccess ไฟล์คือ /usr/dt/config/Xaccess ผู้ดูแลระบบสามารถกำหนด Xaccess เองโดยคัดลอก /usr/dt/config/Xaccess ไปที่ /etc/dt/config/Xaccess แล้วแก้ไข /etc/dt/config/Xaccess ดีฟอลต์ Xaccess ไฟล์ไม่มี entry

### ไฟล์ Xservers

ไฟล์ Xservers มีรายการของจอแสดงผลที่จะจัดการ ดีฟอลต์ Xservers ไฟล์คือ /usr/dt/config/Xservers ผู้ดูแลระบบ สามารถ กำหนด Xservers เองโดยคัดลอก /usr/dt/config/Xservers ไปที่ /etc/dt/config/Xservers แล้วแก้ไข /etc/dt/config/Xservers ดีฟอลต์ Xservers ไฟล์มี entry สำหรับหนึ่งจอแสดงผล โลคัล

### ไฟล์ Xresources

ไฟล์ Xresources มีนิยามรีซอร์สที่ระบุ ลักษณะที่ปรากฏของจอภาพลือกอิน ดีฟอลต์ Xresources ไฟล์คือ /usr/dt/config/Xresources. ผู้ดูแลระบบสามารถกำหนด Xresources เองโดยคัดลอก /usr/dt/config/Xresources ไปที่ /etc/dt/config/Xresources แล้วแก้ไข /etc/dt/config/Xresources

### ไฟล์ Xsetup

ไฟล์ Xsetup ปกติคือเชลล์สคริปต์ เฉพาะผู้ใช้ root เท่านั้นที่สามารถ รันได้ และควรระมัดระวังเกี่ยวกับการรักษาความปลอดภัย สคริปต์นี้ถูกรัน ก่อนจอภาพลือกอินถูกแสดง ไม่มีอาร์กิวเมนต์ถูกส่งให้ สคริปต์ คำสั่ง dtlogin รอจนสคริปต์นี้จบการทำงาน ก่อนแสดงจอภาพลือกอิน

ดีฟอลต์ Xsetup ไฟล์คือ /usr/dt/config/Xsetup ผู้ดูแลระบบ สามารถกำหนด Xsetup เองโดยคัดลอก /usr/dt/config/Xsetup ไปที่ /etc/dt/config/Xsetup แล้วแก้ไข /etc/dt/config/Xsetup ดีฟอลต์ Xsetup ไฟล์มีโค้ดจำเพาะคู่ค่า แต่โดยปกติจะมีโค้ดที่เซตอัพ X server ก่อนที่จะแสดงจอภาพ ลือกอิน เช่นการเซตคีย์บอร์ดแม่พิมพ์

### ไฟล์ Xstartup

ไฟล์ Xstartup ปกติคือเชลล์สคริปต์ เฉพาะผู้ใช้ root เท่านั้นที่สามารถ รันได้ และควรระมัดระวังเกี่ยวกับการรักษาความปลอดภัย นี่เป็นที่เก็บ คำสั่งที่แสดงข้อความประจำวันหรือดำเนินฟังก์ชันระดับระบบ อื่นในนามของผู้ใช้ ตัวแปรสภาวะแวดล้อมต่อไปนี้ถูกเซต สำหรับการใช้สคริปต์นี้:

#### DISPLAY

เซตเป็นชื่อจอแสดงผลที่เชื่อมโยง

HOME เซตเป็นโฮมไดเรกทอรีของผู้ใช้

PATH เซตเป็นค่าของรีซอร์ส systemPath

USER เซตเป็นชื่อผู้ใช้

#### SHELL

เซตเป็นค่าของรีซอร์ส systemShell

ไม่มีอาร์กิวเมนต์ถูกส่งให้ สคริปต์ คำสั่ง dtlogin รอจนสคริปต์นี้จบการทำงาน ก่อนเริ่มเซสชันผู้ใช้ ถ้าค่า exit ของสคริปต์นี้ไม่ใช่ศูนย์ dtlogin หยุดเซสชัน ทันทีและเริ่มวงรอบการพิสูจน์ตัวตนอื่น

ดีฟอลต์ Xstartup ไฟล์ /usr/dt/config/Xstartup ผู้ดูแลระบบสามารถกำหนด Xstartup เองโดยคัดลอก /usr/dt/config/Xstartup ไปที่ /etc/dt/config/Xstartup แล้วแก้ไข /etc/dt/config/Xstartup ดีฟอลต์ Xstartup ไฟล์มีโค้ดที่จะเปลี่ยนความเป็นเจ้าของ /dev/console ไปเป็นผู้ใช้ซึ่งเซสชันกำลังรันบนคอนโซล

## ไฟล์ Xsession

สคริปต์ Xsession กำหนดเซสชันของผู้ใช้และเรียก desktop session manager โดยถูกรันด้วยสิทธิของผู้ใช้ที่ได้รับอนุญาต และมีตัวแปรสถานะแวดล้อมบางส่วนเซตไว้ก่อน ดูที่ ตัวแปรสถานะแวดล้อม สำหรับรายการของตัวแปรที่เซตไว้ก่อน

ดีฟอลต์ Xsession ไฟล์คือ /usr/dt/bin/Xsession ผู้ดูแลระบบ สามารถกำหนด Xsession เองโดยคัดลอก /usr/dt/bin/Xsession ไปที่ /etc/dt/config/Xsession แล้วแก้ไข /etc/dt/config/Xsession เซสชันรีซอร์สที่กำหนดใน Xconfig ต้องถูกเปลี่ยนให้อ้างอิงไฟล์ Xsession ที่กำหนดเองเช่นกัน ดูที่ ไฟล์ Xconfig สำหรับข้อมูลเกี่ยวกับวิธีอัปเดตไฟล์ Xconfig ดีฟอลต์ Xsession ไฟล์มีโค้ดการกำหนดค่าเริ่มต้นเซสชัน โดยมี โค้ดจำเพาะคู่ค่าบางส่วน แต่ฟังก์ชันทั่วไปมีดังนี้:

- ซอร์ส \$HOME/.dtprofile ของผู้ใช้
- ซอร์สของสคริปต์ /etc/dt/config/Xsession.d/\*
- ซอร์สของสคริปต์ /usr/dt/config/Xsession.d/\*
- เรียกใช้ไคลเอ็นต์ต้อนรับของเดสก์ท็อป, dthello ในแบ็กกราวนด์
- ซอร์สของสคริปต์เซตอัปพาทค้นหาแอปพลิเคชัน, dtsearchpath
- เรียกใช้ไคลเอ็นต์เซตอัปวิธีใช้ dthelpgen ในแบ็กกราวนด์
- เรียกใช้ไคลเอ็นต์เซตอัปไดเร็กทอรีตัวจัดการแอปพลิเคชัน dtappgather ในแบ็กกราวนด์
- Execs ตัวจัดการเซสชันเดสก์ท็อป dtsession
- 

ผู้ดูแลระบบถูกป้องกันไม่ให้ปรับแต่งไฟล์ Xsession

## ไฟล์ Xreset

สมมาตรกับ Xstartup, Xreset สคริปต์ถูกรันหลังจาก เซสชันผู้ใช้จบการทำงาน เนื่องจากถูกรันโดยผู้ใช้ root Xreset สคริปต์ควรมีคำสั่งที่ยกเลิกผลของคำสั่ง ใน Xstartup, เช่นการยกเลิกการเมาท์ไดเร็กทอรีจากไฟล์เซิร์ฟเวอร์ คอลเล็กชัน ของตัวแปรสถานะแวดล้อมที่ถูกส่งให้กับ Xstartup ถูก กำหนดให้กับ Xreset เช่นกัน

ดีฟอลต์ Xreset ไฟล์คือ /usr/dt/config/Xreset ผู้ดูแลระบบ สามารถกำหนด Xreset เองโดยคัดลอก /usr/dt/config/Xreset ไปที่ /etc/dt/config/Xreset แล้วแก้ไข /etc/dt/config/Xreset ดีฟอลต์ Xreset ไฟล์มีโค้ดการเปลี่ยนเจ้าของ /dev/console กลับไปเป็น root

## ไฟล์ Xerrors

สคริปต์ Xerrors มีข้อความแสดงความผิดพลาดจาก dtlogin และ ข้อมูลที่เอาต์พุตไปที่ stderr โดย Xsetup, Xstartup หรือ Xreset ผู้ดูแลระบบสามารถใช้เนื้อหาของไฟล์นี้สำหรับการแก้ปัญหา dtlogin errorLogSize รีซอร์สจำกัด ขนาดของไฟล์ Xerrors และสามารถป้องกันไฟล์ไม่ให้มีขนาดเกิน ขอบเขต ถ้าไฟล์ขยายใหญ่กว่าขนาดที่ร้องขอและถูกลดขนาด โดย dtlogin, ผู้ใช้ที่กำลังเข้าถึงไฟล์อยู่ (ตัวอย่างเช่น โดยใช้ cat หรือ tail) จะจำเป็นต้องปิดไฟล์ (หลังจากไฟล์ถูก ลดขนาด) และเปิดใหม่เพื่อเข้าถึง เพื่อให้เห็นข้อมูลซึ่งต่อมาถูกบันทึกลงในไฟล์

ผู้ดูแลระบบสามารถเปลี่ยนชื่อพารของ Xerrors โดย เซ็ตรีชอร์ส errorLogFile ในไฟล์ Xconfig

## ไฟล์ Xpid

สคริปต์ Xpid มี process ID ของกระบวนการ dtlogin หลักซึ่งสามารถถูกใช้เมื่อส่งสัญญาณไปที่ dtlogin ผู้ดูแลระบบสามารถเปลี่ยนชื่อพารของ Xpid โดย เซ็ตรีชอร์ส pidFile ในไฟล์ Xconfig

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dtaction” ในหน้า 232

“คำสั่ง dtsession” ในหน้า 265

---

## คำสั่ง dtscript

### วัตถุประสงค์

สร้างไดอะล็อกอย่างง่ายที่ใช้ในสภาวะแวดล้อม X Window System

### ไวยากรณ์

```
dtscript [-xrm options] [-dir Path] [-file FileName] [-workspace WorkspaceName]
```

หมายเหตุ: อ็อพชัน `-xrm` ต้อง ถูกระบุ ถ้าจะนำไปใช้ ก่อนแฟล็กอื่นๆ

### คำอธิบาย

Desktop Script สนับสนุนเซ็ตย่อยของ Motif widgets ที่คุณ ลากแล้วปล่อยจากถาดค่าอ็อพชันลงในไดอะล็อกของคุณ คุณสามารถย้ายหรือปรับขนาด widget ในไดอะล็อก คุณยังสามารถคุณสมบัติแก้ไข widget โดยใช้เอดิเตอร์พิเศษที่เตรียมไว้

คุณสามารถป้อนการเรียกกลับ เพื่อให้ widgets ทำงานตามต้องการ เมื่อไดอะล็อกสมบูรณ์, Desktop Script จะสร้างโค้ด dtksh ให้

### แฟล็ก

ไอเท็ม

`-dir Path`

`-file FileName`

`-workspace WorkspaceName`

`-xrm options`

คำอธิบาย

เซ็ตไดเรกทอรีปัจจุบันของ Desktop Script ที่แสดงในไดอะล็อก File Select ไปที่ `Path`  
โหนดไดอะล็อกที่มีอยู่ชื่อ: `FileName` อาร์กิวเมนต์ `FileName` เป็นได้ทั้งชื่อพารสมบูรณ์ ชื่อพารที่สัมพันธ์กับไดเรกทอรีปัจจุบัน หรือชื่อพารที่สัมพันธ์กับ ค่า `-dir`  
โหนด Desktop Script ลงในพื้นที่ใช้งาน CDE ที่ตรงกัน  
ช่วยให้คุณป้อนค่ากำหนด (`options`) ที่ปกติคุณจะกำหนดไว้ในริชอร์สไฟล์

### ตัวอย่าง

เมื่อต้องการเรียก Desktop Script จากหน้าต่างให้ป้อน:

```
dtscript
```

# ไฟล์

ไอเท็ม  
/usr/dt/bin/dtscript

คำอธิบาย  
มีคำสั่ง dtscript

---

## คำสั่ง dtsession

### วัตถุประสงค์

จัดการเซสชัน CDE

### ไวยากรณ์

`dtsession [options] ...`

### คำอธิบาย

คำสั่ง `dtsession` จัดเตรียมฟังก์ชันการจัดการเซสชัน ที่สอดคล้องกับ ICCCM 1.1 ระหว่างเซสชันผู้ใช้ จากล็อกอินถึงล็อกเอาต์ โดยจะเริ่มต้นตัวจัดการหน้าต่าง และอนุญาตให้ผู้ใช้บันทึกเซสชัน เรียกคืนเซสชัน ล็อกเซสชัน เริ่มต้นโปรแกรม screen savers และจัดสรรสีสำหรับไคลเอ็นต์ desktop-compatible

หมายเหตุ: ตัวจัดการล็อกอินของเดสก์ท็อป `dtlogin` เรียกไคลเอ็นต์ `dtsession` โดยอัตโนมัติผ่านสคริปต์ `Xsession` ไคลเอ็นต์ `dtsession` ยังสามารถถูกสตาร์ทผ่านสคริปต์ `Xsession` บน X server ที่มีอยู่ ตัวจัดการเซสชัน `dtsession` เริ่มต้นตัวจัดการหน้าต่างโดยอัตโนมัติ

คำสั่ง `dtsession` สนับสนุนงานต่อไปนี้:

- เตรียมข้อมูลเบื้องต้นเซสชัน
- การเริ่มต้นตัวจัดการหน้าต่าง
- เรียกคืนเซสชันโฮมหรือเซสชันปัจจุบัน
- จัดเตรียมการล็อกเซสชันตามคำสั่งหรือเมื่อหมดเวลา
- จัดเตรียม screen saver ของเซสชันตามคำสั่งหรือเมื่อหมดเวลา
- ทำหน้าที่เหมือนเซิร์ฟเวอร์การจัดสรรสีสำหรับเดสก์ท็อปไคลเอ็นต์อื่น
- บันทึกเซสชันโฮมหรือเซสชันปัจจุบัน
- แสดงไดอะล็อกการยืนยันเมื่อล็อกเอาต์
- แสดงไดอะล็อกการเลือกเซสชันเมื่อล็อกเอาต์
- จบการทำงานเซสชัน

### เซสชัน

เซสชันเป็นคอลเล็กชันของแอปพลิเคชัน การตั้งค่าและรีซอร์สที่มีอยู่บนเดสก์ท็อปของผู้ใช้ การจัดการเซสชันคือชุดของหลัก การ และโปรโตคอลที่อนุญาตให้ตัวจัดการเซสชันพิเศษ เช่น `dtsession` ทำการบันทึกและเรียกคืนเซสชันของผู้ใช้ ผู้ใช้สามารถ

อกอินเข้าสู่ระบบ และได้รับชุดของแอ็พพลิเคชันที่รันอยู่ ค่าติดตั้ง และ รีซอร์สที่มีอยู่เหมือนกับเมื่อผู้ใช้ล็อกอ้อฟ เมื่อผู้ใช้ล็อกอิน เข้าสู่เดสก์ทอปในครั้งแรก เซสชันเริ่มต้นดีฟอลต์จะถูกโหลดขึ้นมา ในภายหลัง, dtsession สนับสนุนรูปแบบของเซสชันปัจจุบันและเซสชันโฮม

เซสชันต่อไปนี้จะถูกกำหนดไว้:

### เซสชันเริ่มต้น

เมื่อผู้ใช้ล็อกอินเข้าสู่เดสก์ทอปในครั้งแรก dtsession จะสร้างเซสชันเริ่มต้นของผู้ใช้โดยใช้ค่าดีฟอลต์ของระบบสำหรับข้อมูลเพิ่มเติม โปรดดูที่ Session Resource Management และ Session Application Management

### เซสชันปัจจุบัน

เซสชันของผู้ใช้ที่รันอยู่ ถือว่าเป็นเซสชันปัจจุบันเสมอ ไม่ว่าจะเป็นที่เรียกคืนเมื่อล็อกอินจากเซสชันหลักที่บันทึกไว้ เซสชันปัจจุบัน ที่บันทึกไว้ หรือเซสชันเริ่มต้นดีฟอลต์ของระบบ จากค่าติดตั้ง Style Manager Startup ของผู้ใช้ เมื่อผู้ใช้ออกจากเซสชัน เซสชันปัจจุบัน จะถูกบันทึกโดยอัตโนมัติ เมื่อผู้ใช้ล็อกอินเข้าสู่เดสก์ทอปในครั้งต่อมา เซสชันปัจจุบัน ที่บันทึกไว้ก่อนหน้านี้จะถูกเริ่มต้นอีกครั้ง เดสก์ทอปถูกคืนค่าเป็นสถานะเดิม ที่เป็นอยู่เมื่อผู้ใช้ล็อกเอาต์ครั้งสุดท้าย

### เซสชันหลัก

อีกอ้อพชั่นที่เรียกคืนเดสก์ทอปกลับเป็นสถานะเดิมทุกครั้ง ที่ผู้ใช้ล็อกอิน ไม่ว่าจะสถานะเมื่อผู้ใช้ล็อกเอาต์จะเป็นอย่างไร ผู้ใช้สามารถบันทึก สถานะของเซสชันปัจจุบัน จากนั้นเซต Style Manager Startup เพื่อให้เดสก์ทอปเริ่มต้นเซสชันนั้นทุกครั้งที่ใช้ล็อกอิน

### เซสชันเฉพาะการแสดงผล

เมื่อต้องการรันเซสชันที่เจาะจงสำหรับจอแสดงผล ผู้ใช้สามารถสร้าง เซสชัน display-specific เมื่อต้องการทำดังกล่าว ผู้ใช้สามารถตัดลอกไดเรกทอรี \$HOME/.dt/sessions ไปยัง \$HOME/.dt/display โดยที่ display เป็นชื่อโฮสต์ที่ไม่ใช่ชื่อแบบเต็มจริง (เช่น pablo:0 สามารถใช้ได้ แต่ pablo.gato.com:0 หรือ local:0 ใช้ไม่ได้) เมื่อผู้ใช้ล็อกอินบนจอแสดงผล pablo:0, เซสชัน display-specific นั้น มีความสำคัญกว่า

## โปรโตคอล ICCCM Session Management

สำหรับแอ็พพลิเคชันที่จะถูกบันทึกเมื่อล็อกเอาต์ และรีสตาร์ทเมื่อล็อกอิน จะต้อง รวมอยู่ในโปรโตคอล session management คำสั่ง dtsession สนับสนุนโปรโตคอล ICCCM 1.1 Session Management

แอ็พพลิเคชันที่ต้องการบันทึกสถานะ สามารถเข้าร่วมในโปรโตคอล WM\_SAVE\_YOURSELF เมื่อต้องการทำดังกล่าว แอ็พพลิเคชันต้องการตั้งค่าคุณสมบัติ WM\_SAVE\_YOURSELF บนหนึ่งในหน้าต่างระดับบนสุด เมื่อเซสชันถูกบันทึก dtsession จะส่งข้อความไคลเอ็นต์ WM\_SAVE\_YOURSELF ไปที่หน้าต่างระดับบนสุดของแอ็พพลิเคชัน แอ็พพลิเคชันดำเนินการต่อเพื่อบันทึกสถานะของตัวเองอย่างเสถียร แอ็พพลิเคชันไม่สามารถโต้ตอบกับผู้ใช้ได้ ขณะทำการบันทึกสถานะอยู่ เนื่องจากแอ็พพลิเคชันจะบันทึกสถานะของตัวเองลงในไฟล์ ตัวจัดการเซสชันมีฟังก์ชันที่อำนวยความสะดวก DtSessionSavePath ซึ่งส่งคืนชื่อพาธแบบเต็มของไฟล์ซึ่งแอ็พพลิเคชันสามารถ บันทึกสถานะของตัวเอง ขณะที่แอ็พพลิเคชันกำลังบันทึกสถานะของตัวเอง, dtsession รอ การแจ้งจากแอ็พพลิเคชันว่าบันทึกเสร็จสิ้นแล้ว เพื่อแจ้ง dtsession ว่าการบันทึกเสร็จสมบูรณ์แล้ว แอ็พพลิเคชันต้องอัปเดตคุณสมบัติ WM\_COMMAND บนหน้าต่างระดับบนสุดของตัวเอง

คุณสมบัติ WM\_COMMAND ในหน้าต่างระดับบนของแอ็พพลิเคชันทำ หน้าที่สองประการหนึ่ง แจ้งการเปลี่ยนแปลงของคุณสมบัติกับ dtsession ว่า แอ็พพลิเคชันเสร็จสิ้นการบันทึกสถานะแล้วและ dtsession สามารถดำเนินการต่อ ไปยังแอ็พพลิเคชันต่อไป สอง ค่าคุณสมบัติ WM\_COMMAND ถูกคาดหวังว่ามี บรรทัดคำสั่งที่ dtsession ใช้ในการรีสตาร์ทแอ็พพลิเคชัน เมื่อเซสชันเริ่มทำงาน ถ้าแอ็พพลิเคชันถูกเริ่มต้น ด้วยชื่อพาธแบบเต็ม แอ็พพลิเคชันต้องใช้ชื่อพาธแบบเต็มเมื่อตั้งค่า

WM\_COMMAND แอ็พพลิเคชันที่ไม่ต้องการบันทึกสถานะของตัวเอง แต่ต้องการรีสตาร์ท สามารถตั้งค่า WM\_COMMAND หนึ่งครั้งระหว่างการเริ่มทำงานของแอ็พพลิเคชัน

## การเรียกคืนเซสชัน

เวลาเริ่มทำงานเซสชัน, dtsession กำหนดเซสชันที่จะเรียกคืน รายการต่อไปนี้อธิบายลำดับของการมาก่อน:

1. แสดงเฉพาะปัจจุบันของเซสชันหลัก
2. เซสชัน ปัจจุบัน หรือเซสชันหลัก
3. เซสชันเริ่มต้น

## การจัดการรีซอร์สเซสชัน

ตัวจัดการเซสชันใช้คุณสมบัติ X Server RESOURCE\_MANAGER เพื่อให้รีซอร์สเดสก์ทอปพร้อมใช้สำหรับแอ็พพลิเคชันทั้งหมด ตัวจัดการเซสชัน โหลด RESOURCE\_MANAGER ในรูปแบบต่อไปนี้:

1. โหลดรีซอร์สดีฟอลต์ของระบบ
2. รวมรีซอร์สที่ระบุของผู้ดูแลระบบ
3. รวมรีซอร์สที่ระบุของผู้ใช้

รีซอร์สดีฟอลต์ของเดสก์ทอปพบได้ในไฟล์ `/usr/dt/config/$LANG/sys.resources` รีซอร์สเหล่านี้จะถูกทำให้พร้อมใช้กับเซสชันผู้ใช้แต่ละเซสชัน โดยใช้คุณสมบัติ RESOURCE\_MANAGER อย่าแก้ไขไฟล์เนื่องจาก ไฟล์ ถูกเขียนทับแบบไม่มีเงื่อนไข ระหว่างการติดตั้งเดสก์ทอปที่จะตามมา

โดยการสร้างไฟล์ `/etc/dt/config/$LANG/sys.resources` ผู้ดูแลระบบ สามารถแทนที่รีซอร์สดีฟอลต์ของระบบหรือระบุรีซอร์สเพิ่มเติม เนื่องจากไฟล์นี้ถูกรวมไว้ในรีซอร์สดีฟอลต์ของเดสก์ทอป ระหว่างการเริ่มต้นเซสชัน เฉพาะข้อมูลจำเพาะใหม่หรือที่อัปเดตเท่านั้นที่ควร อยู่ในไฟล์นี้ เป็นการดีกว่าที่จะทำสำเนาของไฟล์รีซอร์สดีฟอลต์ของเดสก์ทอป รีซอร์สที่ระบุในไฟล์นี้ถูกทำให้พร้อมใช้ กับแต่ละเซสชันของผู้ใช้ผ่านคุณสมบัติ RESOURCE\_MANAGER รีซอร์สที่ระบุในไฟล์นี้มีลำดับมาก่อน รีซอร์สที่ระบุในไฟล์รีซอร์สดีฟอลต์ของ เดสก์ทอป

โดยการแก้ไขไฟล์ `$HOME/.Xdefaults` ผู้ใช้สามารถเขียนทับ รีซอร์สดีฟอลต์ของเดสก์ทอปและรีซอร์สของผู้ดูแลระบบ รีซอร์สที่ระบุในไฟล์นี้ ถูกทำให้พร้อมใช้เฉพาะเซสชันของผู้ใช้ผ่านคุณสมบัติ RESOURCE\_MANAGER และมีลำดับเหนือกว่า รีซอร์สที่ระบุในไฟล์ รีซอร์สดีฟอลต์ของเดสก์ทอปหรือ ไฟล์รีซอร์สของผู้ดูแลระบบ

**หมายเหตุ:** X Toolkit Intrinsics ระบุว่าจะโหลดแอ็พพลิเคชันรีซอร์ส จาก RESOURCE\_MANAGER หรือจาก `$HOME/.Xdefaults` อย่างใดอย่างหนึ่ง โดยปกติ นี้หมายความว่าไฟล์ `$HOME/.Xdefaults` ของผู้ใช้จะถูกข้าม อย่างไรก็ตาม ตัวจัดการเซสชันจัดเตรียม `$HOME/.Xdefaults` โดยรวมเข้ากับ RESOURCE\_MANAGER เมื่อเซสชันเริ่มทำงาน เมื่อผู้ใช้เปลี่ยนไฟล์ `$HOME/.Xdefaults`, การเปลี่ยนแปลงที่ว่าจะไม่แสดงกับแอ็พพลิเคชันใหม่จนกว่าผู้ใช้จะเรียกการดำเนินการ

### ReloadResources

แอ็พพลิเคชัน `ReloadResources` สั่งงานตัวจัดการเซสชันให้รีโหลด RESOURCE\_MANAGER ด้วยรีซอร์สที่ระบุระบบระบุ ผู้ดูแลระบบระบุ และผู้ใช้ระบุ ซึ่งทำให้พร้อมใช้กับการเปลี่ยนแปลง แอ็พพลิเคชันใหม่ที่ถูกดำเนินการกับไฟล์รีซอร์สที่ผู้ดูแลระบบระบุ หรือที่ ผู้ใช้ระบุ

## การจัดการแอ็พพลิเคชันเซสชัน

เมื่อเซสชันเริ่มทำงาน ตัวจัดการเซสชันรีสตาร์ทแอ็พพลิเคชัน ที่ถูกบันทึกเป็นส่วนหนึ่งของเซสชัน ชุดดีฟอลต์ของแอ็พพลิเคชันของระบบ ที่จะถูกเรียกคืนเป็นส่วนหนึ่งของเซสชันเริ่มต้นของผู้ใช้สามารถพบได้ในไฟล์ `/usr/dt/config/$LANG/sys.session` อย่าแก้ไขไฟล์เนื่องจาก ไฟล์ ถูกเขียนทับแบบไม่มีเงื่อนไข ระหว่างการติดตั้งเดสก์ท็อปที่จะตามมา

ผู้ดูแลระบบสามารถแทนที่ชุดของแอ็พพลิเคชันที่ถูกเรียกคืน เป็นส่วนหนึ่งของเซสชันเริ่มต้นของผู้ใช้โดยการสร้างไฟล์ `/etc/dt/config/$LANG/sys.session` ไม่เหมือนกับรีซอร์สไฟล์ ไฟล์นี้ถูกใช้เป็นการแทนที่อย่างสมบูรณ์ สำหรับไฟล์ดีฟอลต์ของเดสก์ท็อป ดังนั้นคุณสามารถสร้างสำเนาของไฟล์ดีฟอลต์ของระบบ และทำการเปลี่ยนแปลงที่จำเป็น

## ตัวจัดการหน้าต่าง

คำสั่ง `dtsession` สตาร์ทตัวจัดการหน้าต่าง โดยดีฟอลต์ `/usr/dt/bin/dtwm` ถูกสตาร์ท สามารถระบุตัวจัดการหน้าต่างที่เป็นทางเลือกได้โดยใช้รีซอร์ส `wmStartupCommand` สำหรับข้อมูลเพิ่มเติม โปรดดูที่ข้อมูลจำเพาะ Workspace Manager

## ตัวจัดการลักษณะ

ตัวจัดการลักษณะมีอินเตอร์เฟซซึ่งผู้ใช้สามารถเปลี่ยนค่าติดตั้ง เดสก์ท็อปและ X server ต่างๆ สำหรับเซสชันปัจจุบันได้ สำหรับข้อมูลเพิ่มเติม โปรดดูที่ข้อมูลจำเพาะ Style Manager

## สีเซิร์ฟเวอร์

คำสั่ง `dtsession` ทำงานเป็นเซิร์ฟเวอร์สีสำหรับเดสก์ท็อป และจัดเตรียมชุดของรีซอร์สต่อไปนี้ที่สามารถใช้เพื่อปรับแต่ง :

### foregroundColor

ควบคุมว่าพิกเซลถูกจัดสรรสำหรับสีพื้นหน้าหรือไม่

### dynamicColor

ระบุว่าสีแบบอ่านอย่างเดียวถูกจัดสรรหรือไม่

### shadowPixmap

ระบุว่าสีถูกจัดสรรสำหรับแสงเงาด้านบนหรือแสงเงาด้านล่าง

### colorUse

จำกัดการจัดสรรสี

### writeXrdbColors

ระบุว่ารีซอร์ส `*background` และ `*foreground` ถูกกำหนดไว้ในฐานข้อมูลรีซอร์สหรือไม่

สำหรับข้อมูลเพิ่มเติม โปรดดูที่ส่วน Color Server Resources

## การล็อกเซสชัน

คำสั่ง `dtsession` จัดเตรียมการล็อกเซสชัน เซสชันปัจจุบัน สามารถถูกล็อกได้โดยตรงโดยการกดไอคอนล็อกบนพาเนลด้านหน้า ถ้าสนับสนุน โดย X server, เซสชันปัจจุบันสามารถถูกล็อกหลังจากช่วงเวลาที่ระบุของการไม่มีกิจกรรม เมื่อต้องการปลดล็อกเซสชัน ผู้ใช้ต้องป้อนรหัสผ่านล็อกอิน รหัสผ่านล็อกอินสำหรับผู้ใช้ `root` หรือรหัสผ่านล็อกอินสำหรับผู้ใช้ ที่ระบุโดยรีซอร์ส `keys` ดูที่ การล็อกจอภาพ และรีซอร์สการบันทึกจอภาพ สำหรับข้อมูลเพิ่มเติมเกี่ยวกับรีซอร์ส คีย์

คำสั่ง `dtssession` เป็นตัวจัดการเซสชัน PAM-enabled ที่มีชื่อเซสชัน `dtssession` ซึ่งสนับสนุนการพิสูจน์ตัวตน UNIX โคลนแบบเดิม เช่นเดียวกับโคลน PAM สำหรับการปลดล็อกเซสชัน การทำงาน การพิสูจน์ตัวตนซ้ำเพิ่มเติม, เช่นที่ต้องการโดย DCE, สามารถถูกเพิ่ม โดยค่าแต่ละราย

System-wide configuration เพื่อใช้ PAM สำหรับการพิสูจน์ตัวตนถูกเซตโดย การสร้างสิทธิผู้ใช้ `root` และแก้ไขค่าแอตทริบิวต์ `auth_type` ใน `usw stanza` ของไฟล์ `/etc/security/login.cfg` เป็น `PAM_AUTH`

กลไกการพิสูจน์ตัวตนที่ใช้ เมื่อ PAM ถูกเปิดใช้ขึ้นกับคอนฟิกูเรชัน สำหรับเซสชันล็อกอินใน `/etc/pam.conf` คำสั่ง `dtssession` ต้องการ `/etc/pam.conf` entry สำหรับชนิดโมดูล `auth` configuration ต่อไปนี้แนะนำให้ใช้ใน `/etc/pam.conf` สำหรับเซสชัน `dtssession`:

```
dtssession      auth          required      /usr/lib/security/pam_aix
```

## โปรแกรมรักษาจอภาพ

คำสั่ง `dtssession` จัดเตรียมการสนับสนุนสำหรับการเรียกใช้โปรแกรม รักษาจอภาพภายนอกเป็นส่วนหนึ่งของการล็อกเซสชัน จากพาเนลด้านหน้า หรือถ้ามีการสนับสนุน โดย X server, หลังจากระยะที่ระบุของการไม่มีกิจกรรม อ้างอิงถึงค่ากำหนด โปรแกรมรักษา จอภาพสำหรับข้อมูลเกี่ยวกับวิธีที่โปรแกรมรักษาจอภาพถูกรวมไว้ใน เดสก์ท็อป

## ส่วนขยายโปรแกรมรักษาจอภาพ X Server

ความสามารถของคำสั่ง `dtssession` ในการจัดเตรียมการล็อกเซสชันหรือ การเรียกใช้โปรแกรมรักษาจอภาพหลังจากช่วงเวลา ที่ระบุของการไม่มีกิจกรรมขึ้นกับ ความสามารถของส่วนขยายโปรแกรมรักษาจอภาพ X server คำสั่ง `dtssession` สนับสนุน X Consortium Sample X11 Screen Saver Extension 1.0 และ HP X Screen Saver Extension ความสามารถของคำสั่ง `dtssession` ในการจดจำส่วนขยายเหล่านี้ทั้งสองตัว ตัวใดตัวหนึ่ง หรือไม่จดจำขึ้นอยู่กับผู้จำหน่าย

## การเริ่มต้นตัวจัดการเซสชัน

คำสั่ง `dtssession` ต้องเริ่มต้นจากสคริปต์ `Xsession Xsession` ถูกอธิบายไว้ในค่ากำหนดตัวจัดการการล็อกอิน แม้ว่า จะแนะนำให้ เริ่มต้น `Xsession` จาก `dtlogin` เป็นส่วนหนึ่งของลำดับการล็อกอินดีฟอลต์ บางระบบจะอนุญาตให้โปรแกรมพรีอ็อกซี เช่น `xinit`, `x11start` หรือ `startx` เพื่อเริ่มต้น `Xsession`

## รีซอร์สเชิร์ฟเวอร์

ไอเท็ม

คำอธิบาย

ไอเท็ม  
colorUse

คำอธิบาย

ClassClass:

ColorUse

ชนิดข้อมูล:

String

ดีฟอลต์: DEFAULT

คำอธิบาย:

ระบุจำนวนของสีที่จะใช้สำหรับส่วนการติดต่อกับผู้ใช้ เซิร์ฟเวอร์จะกำหนดชนิดของมอนิเตอร์ขึ้นอยู่กับจำนวนของระนาบการแสดงผลของหน้าจอ ดังต่อไปนี้:

#### 1, 2 หรือ 3 ระนาบ (B\_W)

ระบุระบบขาวดำ พาเลตสีใช้เซลล์สี 2 เซลล์ สำหรับส่วนติดต่อกับผู้ใช้ ในคอนฟิกูเรชันนี้ จะมีพาเลตสีที่พร้อมใช้งานเพียง 2 พาเลต คือ BlackWhite หรือ WhiteBlack ภาคค่าอ็อพชันเหล่านี้ไม่สามารถเปลี่ยนได้แบบไดนามิก เมื่อต้องการเปลี่ยนพาเลต แอปพลิเคชันทั้งหมดที่ใช้ พาเลตสีต้องถูกรีเซ็ต ค่ารีซอร์สนี้บังคับ ShadowPmaps ให้เป็น True และ ForegroundColor เป็นสีดำหรือขาว (ขึ้นกับการเลือกภาคค่าอ็อพชัน)

#### 4 หรือ 5 planes (LOW\_COLOR)

ระบุระบบสีระดับต่ำ ภาคค่าอ็อพชันสีมีสองชุดสี และใช้ค่าสูงสุด 12 เซลล์สีสำหรับส่วนการติดต่อกับผู้ใช้ รวมทั้งสีดำ และสีขาว (เซลล์สี 0 และ 1) จำนวนของเซลล์สีสามารถถูกลดโดยใช้รีซอร์ส ShadowPmaps และ ForegroundColor

#### 6 planes (MEDIUM\_COLOR)

ระบุระบบสีระดับกลาง ภาคค่าอ็อพชันสีมีสี่ชุดสี และใช้ค่าสูงสุด 22 เซลล์สีสำหรับส่วนการติดต่อกับผู้ใช้ รวมทั้งสีดำ และสีขาว (เซลล์สี 0 และ 1) จำนวนของเซลล์สีสามารถถูกลดโดยใช้รีซอร์ส ShadowPmaps และ ForegroundColor

#### 7+ planes (HIGH\_COLOR)

ระบุระบบสีระดับสูง ภาคค่าอ็อพชันสีมีแปดชุดสี และใช้ค่าสูงสุด 42 เซลล์สีสำหรับส่วนการติดต่อกับผู้ใช้ รวมทั้งสีดำ และสีขาว (เซลล์สี 0 และ 1) จำนวนของเซลล์สีสามารถถูกลดโดยใช้รีซอร์ส ShadowPmaps และ ForegroundColor

dynamicColor

ClassClass:

DynamicColor

ชนิดข้อมูล:

Boolean

ดีฟอลต์: True

คำอธิบาย:

รีซอร์สนี้มีค่าได้ทั้ง True หรือ False รีซอร์ส dynamicColor ถูกใช้เพื่อลดจำนวนของเซลล์สีที่ใช้ หลังจากเลือกพาเลต และไม่มี การเปลี่ยนแปลง dynamicColor สามารถ ตั้งค่าเป็น False หากตั้งค่าเป็น False สีจะไม่สามารถเปลี่ยนได้แบบไดนามิกโดยใช้ตัวจัดการลักษณะเดสก์ท็อป ภาคค่าอ็อพชันที่เลือกมีผลในเซสชันต่อไป ครั้งถัดไปที่เซสชันทำงาน เซิร์ฟเวอร์จะใช้ เซลล์สีแบบอ่านอย่างเดียวที่สามารถแบ่งใช้โดยโคลเอ็นต์ทั้งหมด เพื่อลดจำนวนของเซลล์สีที่ใช้

ไอเท็ม  
foregroundColor

คำอธิบาย

ClassClass:

ForegroundColor

ชนิดข้อมูล:

String

ดีฟอลต์: DYNAMIC

คำอธิบาย:

รีซอร์สนี้มีค่า White, Black หรือ Dynamic รีซอร์ส foregroundColor จะทำให้ข้อความทั้งหมด (ส่วนหน้า) ใช้พิกเซล 0 หรือ 1 (Black หรือ White) หรือเพื่อให้มีเซลล์สีสำหรับส่วนหน้าโดยเฉพาะ และ เปลี่ยนเพื่อตอบสนองกับสีพื้นหลัง (Dynamic) สำหรับแต่ละ ColorSet หากตั้งค่าเป็น White หรือ Black จำนวนของเซลล์สีที่ใช้อ่ต่อ ColorSet จะลดลง 1

shadowPixmap

ClassClass:

ShadowPixmap

ชนิดข้อมูล:

String

ดีฟอลต์: DEFAULT

คำอธิบาย:

สำหรับระบบสี รีซอร์สนี้มีค่าได้ทั้ง True หรือ False ถ้าเป็น True, topShadowColor และ bottomShadowColor ใช้พิกเซลเหมือนกับ background และ topShadowPixmap และ bottomShadowPixmap ถูกระบุแทนสีที่บในการสร้างลักษณะ 3-D ซึ่งจะลด จำนวนของเซลล์สีต่อ ColorSet ลง 2 รีซอร์สนี้จะมีค่าดีฟอลต์เป็น True สำหรับระบบที่มีระนาบสีี่ระนาบหรือน้อยกว่า (16 เซลล์สี หรือน้อยกว่า) และ False สำหรับระบบที่มีระนาบสีี่มากกว่าสี่ระนาบ

writeXrdbColors

ClassClass:

WriteXrdbColors

ชนิดข้อมูล:

Boolean

ดีฟอลต์: True

## รีซอร์สการล็อกจอภาพและการรักษาจอภาพ

ไอเท็ม

คำอธิบาย

ไอเท็ม  
keys

คำอธิบาย

ClassClass:

Keys

ชนิดข้อมูล:

unsigned char

ดีฟอลต์: NULL

คำอธิบาย:

แสดงผู้ถือคีย์ซึ่งสามารถปลดล็อกจอภาพ เมื่อ ถูกล็อกโดยผู้ใช้ รายการนี้เป็นรายการของ ID ผู้ใช้  
ที่ค้นโดยคอมพิวเตอร์ ตัวอย่างเช่น หากผู้ใช้ kim มีรีซอร์สต่อไปนี้ที่แอ็คทีฟระหว่าง เซสชัน ผู้ใช้ fred และ  
keith สามารถ ปลดล็อกเมื่อ kim ล็อกหน้าจอ:

Dtsession\*keys: fred,keith

passwordTimeout

ClassClass:

passwordTimeout

ชนิดข้อมูล:

unsigned int

ดีฟอลต์: 10

คำอธิบาย:

ระบุ (เป็นวินาที) จำนวนครั้งก่อนที่ไดอะล็อกที่ซ่อน จะถูกลบออกจากจอภาพ เมื่อจอแสดงผล  
ถูกล็อก ตัวชี้แสดงเคอร์เซอร์ล็อก และไดอะล็อกถูกแสดงเพื่อขอรหัสผ่านผู้ใช้ ถ้าไม่มีกิจกรรมจากตัว  
ชี้หรือคีย์บอร์ดถูกตรวจพบเป็นเวลา passwordTimeout วินาที ไดอะล็อกจะถูกลบออกจากจอภาพ  
ไดอะล็อกถูกแสดงอีกครั้ง ทันทีที่เหตุการณ์ตัวชี้หรือคีย์บอร์ดถูกตรวจพบ passwordTimeout เท่ากับ  
0 จะคงไดอะล็อกที่ซ่อนไว้ในตำแหน่ง ตลอดเวลาที่จอแสดงผล ถูกล็อก ค่าดีฟอลต์คือ 10 วินาที

## รีซอร์สเบ็ดเตล็ด

ไอเท็ม  
queryServerSettings

คำอธิบาย

ClassClass:

QueryServerSettings

ชนิดข้อมูล:

Boolean

ดีฟอลต์: False

คำอธิบาย:

ระบุว่าคำสั่ง dtsession จะเคียวรีเซิร์ฟเวอร์ที่ล็อกเอาต์สำหรับค่าติดตั้งทั้งหมด หรือระบุว่าระบบที่  
เฉพาะค่าติดตั้งที่ตั้งค่าโดยใช้ Style Manager บนเดสก์ท็อป การเคียวรีเซิร์ฟเวอร์ประกันว่าค่าติด  
ตั้งทั้งหมด ถูกบันทึก; อย่างไรก็ตามมี ผลการทำงานจะลดลงเมื่อการเคียวรีเต็มรูปแบบ เสร็จสิ้น  
ค่าดีฟอลต์ คือ False ซึ่งหมายความว่าเซิร์ฟเวอร์ไม่ถูกเคียวรี

ไอเท็ม  
saveFontPath

คำอธิบาย  
ClassClass:  
SaveFontPath

ชนิดข้อมูล:  
Boolean

wmStartupCommand

ดีฟอลต์: False  
ClassClass:  
WmStartupCommand

ชนิดข้อมูล:  
executable path

ดีฟอลต์: NULL

คำอธิบาย:

อนุญาตให้ตัวจัดการหน้าต่างที่เป็นทางเลือกเพื่อเริ่มต้นเมื่อล็อกอิน ถ้ารีซอร์สนี้เป็น NULL, dtsession สตาร์ท /usr/dt/bin/dtwm การเริ่มต้นที่เป็นทางเลือกอาจมีลักษณะดังนี้:

```
Dtsession*wmStartupCommand: /usr/bin/X11/mwm
```

คำสั่ง ต้องไม่มีคำสั่งใดๆ ไปยังเซลล์ในตัวเอง และต้องไม่อยู่ในเครื่องหมายคำพูด หากใช้ตัวจัดการหน้าต่างอื่นที่ไม่ใช่ /usr/dt/bin/dtwm โคลเอ็นต์จะถูกเรียกคืน แต่อาจไม่ถูกเรียกคืนไปยังตำแหน่งที่ถูกต้อง โดยดีฟอลต์ รีซอร์สนี้มีค่า NULL

## สถานะออก

ค่าออกต่อไปนี้ถูกส่งคืน:

ไอเท็ม  
0  
>0

คำอธิบาย  
ดำเนินการเป็นผลสำเร็จ  
มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการเริ่มตัวจัดการเซสชันจากบรรทัดคำสั่งโดยไม่เรียกคือ เซสชันก่อนหน้าให้ป้อน:

```
dtsession -norestore
```

## Location

/usr/dt/bin/dtsession

ไฟล์

## ไอเท็ม

```

/usr/dt/config/ $LANG/sys.session
/etc/dt/config/ $LANG/sys.session
/usr/dt/config/ $LANG/sys.resources
/etc/dt/config/ $LANG/sys.resources
$HOME/.Xdefaults

```

## คำอธิบาย

ชุดดีฟอลต์เดสก์ทอปของแอฟพลิเคชันสำหรับ เซสชันเริ่มต้นของผู้ใช้  
ชุดแอฟพลิเคชันที่ผู้ดูแลระบบระบุ สำหรับเซสชันเริ่มต้นของผู้ใช้  
รีซอร์สดีฟอลต์ของเดสก์ทอป  
รีซอร์สที่ระบุผู้ดูแลระบบ  
รีซอร์สที่ระบุผู้ใช้

หมายเหตุ: คำสั่ง `dtsession` เก็บข้อมูลเซสชันใน `$HOME/.dt/display` หรือ `$HOME/.dt/sessions` เนื้อหาของไดเรกทอรีเหล่านี้ไม่ควรถูกแก้ไขโดยผู้ใช้โดยตรง

```

/usr/dt/app-defaults/ $LANG/Dtsession

```

รีซอร์ส `dtsession` ดีฟอลต์

## สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dtaction`” ในหน้า 232

“คำสั่ง `dtlogin`” ในหน้า 236

# คำสั่ง `dtterm`

## วัตถุประสงค์

จัดเตรียมการสนับสนุนรันไทม์ของ แอฟพลิเคชันที่มีอยู่

## ไวยากรณ์

`dtterm` [Flags...]

## คำอธิบาย

โคลเอ็นต์ `dtterm` จัดเตรียม การสนับสนุนรันไทม์ของแอฟพลิเคชันที่มีอยู่ที่เขียนขึ้นสำหรับเทอร์มินัลอักขระตามมาตรฐาน X3.64-1979 และ ISO 6429:1992(E)

## แฟล็ก

หมายเหตุ: เทอร์มินัลอิมูเลเตอร์ `dtterm` ยอมรับแฟล็กบรรทัดรับคำสั่ง X Toolkit มาตรฐานทั้งหมดพร้อมกับ แฟล็กเพิ่มเติม โดยทั้งหมดถูกแสดงด้านล่าง (ถ้าแฟล็กขึ้นต้น ด้วย + แทน - แฟล็กถูกเรียกคืนเป็น ค่าดีฟอลต์):

## ไอเท็ม

-132

## คำอธิบาย

ทำให้ลำดับหลัก DECCOLM เป็นที่จดจำ และหน้าต่าง `dtterm` จะปรับขนาดอย่างเหมาะสม โดยปกติลำดับหลัก DECCOLM ที่สลับระหว่างโหมดคอลัมน์ 80 และ 132 จะถูกละเว้น รีซอร์สที่เกี่ยวข้อง: `c132`

+132

ทำให้ลำดับหลัก DECCOLM ถูกละเว้น นี่เป็นการทำงานดีฟอลต์ รีซอร์สที่เกี่ยวข้อง: `c132`

-aw

หมายถึงควรอนุญาต `auto-wraparound` ซึ่งอนุญาตให้เคอร์เซอร์ ตัดไปที่ตำแหน่งเริ่มต้นของบรรทัดถัดไปโดยอัตโนมัติ เมื่อเคอร์เซอร์อยู่ที่ตำแหน่ง ขวาสุดของบรรทัดและข้อความถูกเอาต์พุต นี่เป็นการทำงานดีฟอลต์ รีซอร์สที่เกี่ยวข้อง: `autoWrap`

+aw

หมายถึงว่า `auto-wraparound` ควรได้รับอนุญาต รีซอร์สที่เกี่ยวข้อง: `autoWrap`

-background *background\_color*

ระบุแบ็กกราวนด์ของหน้าต่างเทอร์มินัลและดีฟอลต์แบ็กกราวนด์ที่ใช้สำหรับแถบเลื่อนและเคอร์เซอร์ตัวชี้ X11 ภายใต้ CDE, แฟล็กนี้เป็นค่าดีฟอลต์กับฟังก์ชันการเลือกชุดสีหลักหรือฟังก์ชันแบ็กกราวนด์, ดูที่ `-bs` โดยปราศจาก CDE, แฟล็กนี้เป็นดีฟอลต์กับ `*background/*Background` โดยมี `ultimate fallback` สีค่า `background_color` อธิบายสีแบ็กกราวนด์ที่จะใช้ รีซอร์สที่เกี่ยวข้อง: `background`

-bd *border\_color*

ระบุสีเส้นขอบสำหรับหน้าต่างทั้งหมด เส้นขอบของเซลล์ widget อาจมองไม่เห็น เมื่อ `reparenting` ตัวจัดการหน้าต่าง เช่น `dtwm` และ `mwm` ถูกใช้ สีดีฟอลต์คือสีค่า `border_color` อธิบายสีเส้นขอบที่จะใช้ รีซอร์สที่เกี่ยวข้อง: `borderColor`

## ไอเท็ม

-bg background\_color  
-bordercolor border\_color  
-borderwidth border\_width

-bs

+bs

-bw border\_width

-C

## คำอธิบาย

เหมือนกับ -background background\_color อธิบายสีแบ็กกราวนด์ที่จะใช้ รีซอร์สที่เกี่ยวข้อง: background

เหมือนกับ -bd ด้านบน border\_color อธิบายสีเส้นขอบที่จะใช้ รีซอร์สที่เกี่ยวข้อง: borderColor

ระบุความกว้างเส้นขอบของหน้าต่างของเซลล์ widget ค่านี้ อาจถูกเขียนทับโดยการ reparenting ตัวจัดการหน้าต่าง เช่น dtwm และ mwm ค่าดีฟอลต์คือ 0 border\_width ระบุความกว้างของเส้นขอบหน้าต่างเป็นพิกเซล รีซอร์สที่เกี่ยวข้อง: borderWidth

ระบุว่าหน้าต่างเทอร์มินัลควรใช้วิธีการเลือก Motif แทนสีแบ็กกราวนด์สำหรับสีแบ็กกราวนด์ของหน้าต่างเทอร์มินัล นี่เป็นการทำงานดีฟอลต์ รีซอร์สที่เกี่ยวข้อง: backgroundIsSelect

ระบุว่าหน้าต่างเทอร์มินัลไม่ควรใช้วิธีการเลือก Motif แทนสีแบ็กกราวนด์สำหรับสีแบ็กกราวนด์ของหน้าต่าง เทอร์มินัล รีซอร์สที่เกี่ยวข้อง: backgroundIsSelect

เหมือนกับ -borderwidth รีซอร์สที่เกี่ยวข้อง: borderWidth

ระบุเอาต์พุตที่กำหนดที่ /dev/console ควรถูก ส่งไปที่หน้าต่างเทอร์มินัลแทน เป็นการจัดเตรียมเพื่อป้องกัน เอาต์พุตซึ่งปกติจะถูกแสดงบน ITE ไม่ให้เขียนทับจอแสดงผลของ X server โดยไม่ได้ถูกจัดเตรียมไว้เป็นกลไกทั่วไป ในการกำหนด เอาต์พุตจาก /dev/console ของระบบที่ไม่ได้กำหนด ไปที่ X server ที่ไม่ได้กำหนด

**หมายเหตุ:** คุณต้องมีสิทธิ์ เป็นเจ้าของ และ read/write กับ /dev/console เพื่อให้แฟล็กนี้ทำงาน

## ไอเท็ม

-display display\_name

-e program\_argument...

-fb fontset

-fg foreground\_color

-fn fontset

-font fontset

-foreground foreground

-geometry geometry\_string

-help

-iconic

+iconic

-j

+j

-kshMode

+kshMode

## คำอธิบาย

ระบุเซิร์ฟเวอร์จอแสดงผล X11 ที่จะถูกใช้โดย dtterm นี่เป็นค่าดีฟอลต์สำหรับค่าในตัวแปรสถานะแวดล้อม \$DISPLAY display\_name ระบุเซิร์ฟเวอร์ X11 ที่จะเชื่อมต่อ

ระบุโปรแกรมเรียกทำงานที่จะถูกเรียกใช้ เป็น subprocess เมื่อ dtterm ถูกสตาร์ท แฟล็กนี้ต้องเป็นแฟล็กสุดท้าย ใน บรรทัดคำสั่ง program\_argument ระบุโปรแกรม และอาร์กิวเมนต์บรรทัดคำสั่งที่จะรัน

ระบุ XFontSet ที่จะถูกใช้เมื่อแสดงข้อความเทอร์มินัลตัวหนา ซึ่งควรถูกระบุเป็น Motif XmFontList สนับสนุน เฉพาะอักขระหรือฟอนต์ mono space การทำงานเมื่อใช้ฟอนต์ตามสัดส่วนไม่มีกำหนดไว้ ฟอนต์ตัวหนาดีฟอลต์ จะถูกสร้างจากชื่อ XLFD ของ userFont ถ้าฟอนต์นั้นไม่มีอยู่ ข้อความตัวหนาจะถูกสร้างโดยขีดเส้นทับ (ด้วย

หนึ่งพิกเซลออฟเซต) userFont fontset ระบุ XFontSet เทอร์มินัลตัวหนาที่จะใช้ รีซอร์สที่เกี่ยวข้อง: userFont

ระบุพื้นหน้าของหน้าต่างเทอร์มินัลและพื้นหน้าดีฟอลต์ ที่ใช้สำหรับแถบเลื่อนและเคอร์เซอร์ตัวชี้ X11 ภายใต้ CDE รีซอร์สจะเป็นค่าดีฟอลต์กับพิกเซลพื้นหน้าชุด สีหลัก โดยปราศจาก CDE รีซอร์สจะเป็นค่าดีฟอลต์กับ \*foreground หรือ \*Foreground โดยมี ultimate fallback สีขาว foreground\_color ระบุสีพื้นหน้าที่จะใช้ รีซอร์สที่เกี่ยวข้อง: พื้นหน้า

ระบุ XFontSet ที่จะถูกใช้เมื่อแสดงข้อความเทอร์มินัล ซึ่งควรถูกระบุเป็น Motif XmFontList สนับสนุนเฉพาะ อักขระหรือฟอนต์ mono space การทำงานเมื่อใช้ฟอนต์ตามสัดส่วนไม่มีกำหนดไว้ ฟอนต์ นี้จะไม่ถูกใช้เพื่อแสดง ข้อความที่ไม่ใช่เทอร์มินัล (เมนูบาร์ ป๊อปอัพเมนู ไอคอน ล็อก และอื่นๆ). ดีฟอลต์คือใช้ค่า XmNtextFontList ของ กระดานข่าวสารพาเรนต์ (โปรดดู XmBulletinBoard) ในลักษณะเดียวกันกับวิดเจ็ต XmText fontset ระบุ XFontSet เทอร์มินัลที่จะใช้ รีซอร์สที่เกี่ยวข้อง: userFont

เหมือนกับ -fn fontset ระบุ XFontSet เทอร์มินัลที่จะใช้ รีซอร์สที่เกี่ยวข้อง: userFont

เหมือนกับ -fg foreground ระบุสีพื้นหน้าที่จะใช้ รีซอร์สที่เกี่ยวข้อง: พื้นหน้า

ระบุขนาดและตำแหน่งที่ตั้งของหน้าต่างเทอร์มินัล ขนาด ดีฟอลต์คือ 24 บรรทัด 80 อักขระในแต่ละบรรทัด ไม่มีตำแหน่งดีฟอลต์ geometry\_string ระบุภูมิศาสตร์ของเทอร์มินัล ที่จะใช้ รีซอร์สที่เกี่ยวข้อง: geometry แสดงข้อความสรุปการใช้ของ dtterm

ระบุว่าเทอร์มินัลอิมูเลเตอร์ควรถูกกำหนดเมื่อเริ่มต้นให้เป็น ไอคอนจอแสดงผล รีซอร์สที่เกี่ยวข้อง: iconic

ระบุว่าเทอร์มินัลอิมูเลเตอร์ควรถูกกำหนดเมื่อเริ่มต้นบน จอแสดงผลเป็นหน้าต่างปกติ นี่เป็นการทำงานดีฟอลต์ รีซอร์สที่เกี่ยวข้อง: iconic

ระบุว่าการเล่นแบบข้ามควรมานำมาใช้ ภายใต้การเล่นแบบข้าม, จอภาพอาจถูกเลื่อนมากกว่าหนึ่งบรรทัดต่อครั้ง ซึ่งมีไว้เพื่อ การอัปเดตจอภาพที่เร็วขึ้น เมื่อมีข้อความหลายบรรทัดถูกส่งไปที่เทอร์มินัล จำนวนมากที่สุดของ บรรทัดที่อาจถูกข้ามถูกจำกัดโดย จำนวนบรรทัดของหน้าต่างเทอร์มินัล บรรทัดทั้งหมดถูกแสดง นี่เป็นการทำงาน ดีฟอลต์ รีซอร์สที่เกี่ยวข้อง: jumpScroll

ระบุว่าการเล่นแบบข้ามที่ไม่ควรมานำมาใช้ สำหรับรายละเอียด ของการเล่นแบบข้าม ดูที่ -j รีซอร์สที่เกี่ยวข้อง: jumpScroll

ระบุว่าโหมด ksh ควรถูกเปิดใช้งาน ภายใต้โหมด ksh การกดคีย์โดยมีชุดปิด extend modifier จะสร้าง escape character ที่ตามด้วยอักขระที่สร้างโดยการกดคีย์ un-extended แฟล็กนี้ถูกจัดเตรียมสำหรับการใช้กับ emacs และ เอดิเตอร์โหมด บรรทัดคำสั่ง emacs ของ ksh หรือ ied ซึ่งขัดแย้งกับ \ การใช้ปกติของ คีย์ meta สำหรับการสร้าง อักขระไบต์เดี่ยวแบบขยาย และสำหรับการสร้างอักขระ multi-byte Asian รีซอร์สที่เกี่ยวข้อง: kshMode

ระบุว่าโหมด ksh ไม่ควรถูกเปิดใช้งาน นี่เป็นการทำงานดีฟอลต์ รีซอร์สที่เกี่ยวข้อง: kshMode

## ไอเท็ม

-l	<b>คำอธิบาย</b> เปิดใช้การบันทึกเอาต์พุต เมื่อการบันทึกถูกเปิดใช้งาน เอาต์พุตทั้งหมดที่ได้รับ จาก subprocess ถูกบันทึกไปที่ไฟล์หรือไปที่ไฟล์ไลน์คำสั่ง (ตามที่ระบุผ่านแฟล็ก -If) เนื่องจากข้อมูลกำลัง ถูกบันทึกโดยตรงจาก subprocess, ซึ่งจะรวม escape characters และคู่ ปิดแคร์/newline ที่ส่งโดยระเบียบบรรทัดของเทอร์มินัล เอาต์พุตอาจถูกเปิดใช้งานและถูกแสดงผ่าน escape sequences ริชอร์สที่เกี่ยวข้อง: logging
+l	ปิดใช้งานการบันทึกเอาต์พุต สำหรับรายละเอียดของการบันทึกเอาต์พุต ดูที่ -l แฟล็กนี้เป็นดีฟอลต์ ริชอร์สที่เกี่ยวข้อง: logging
-If file_name	ระบุชื่อของไฟล์ซึ่งบันทึกเอาต์พุตที่อธิบายใน แฟล็ก -l ถ้า file_name เริ่มต้น ด้วยสัญลักษณ์ไพพ์ (!), ส่วนที่เหลือของสตริงถูกยอมรับว่าเป็นคำสั่ง ที่จะถูกใช้เป็นจุดสิ้นสุดของไพพ์ ชื่อไฟล์ดีฟอลต์คือ DtermLogXXXXX (โดยที่ XXXXX คือ process id ของ dterm) และถูกสร้างในไดเรกทอรีซึ่ง dterm ถูกเริ่มทำงาน ถ้าทำอักขระสุดท้าย คือ XXXXX, จะถูกแทนที่ด้วย process ID file_name ระบุชื่อล็อกไฟล์ที่จะใช้ ริชอร์สที่เกี่ยวข้อง: logFile
-ls	ระบุว่าเชลล์ที่เริ่มทำงานควรเป็นเชลล์ ล็อกอิน (เช่น อักขระแรกของ argv[0] จะเป็นเสนประ เพื่อระบุ เชลล์ว่าควรอ่านไปรไฟล์ของระบบ และ \$HOME/.profile ของผู้ใช้ (สำหรับ ksh และ sh) หรือ csh.login ของระบบ และ \$HOME.login ของผู้ใช้ (สำหรับ csh) ริชอร์สที่เกี่ยวข้อง: loginShell
+ls	ระบุว่าเชลล์ (non-login) ปกติควรถูกสตาร์ท นี่เป็นการทำงานดีฟอลต์ ริชอร์สที่เกี่ยวข้อง: loginShell
-map	บ่งชี้ว่า dterm ควรแม่พ (de-iconify) ตัวเองเมื่อ subprocess เอาต์พุต ถ้าไม่ถูกแม่พ (iconified) ระยะเริ่มต้นของเวลาระหว่างที่ dterm จะไม่แม่พตัวเอง เมื่อ subprocess เอาต์พุตอาจถูกระบุผ่านริชอร์ส mapOnOutputDelay ริชอร์สที่เกี่ยวข้อง: mapOnOutput
+map	ระบุว่าการมีการทำแม่พพิเศษ นี่เป็นการทำงานดีฟอลต์ ริชอร์สที่เกี่ยวข้อง: mapOnOutput
-mb	ระบุว่าการแจ้งเตือนถึงขีดแจ้งเตือน เมื่อผู้ใช้พิมพ์ไกลถึงขอบขวา ระยะจริงที่เกี่ยวข้องของถูก ระบุโดยแฟล็ก -nb ริชอร์สที่เกี่ยวข้อง: marginBell
+mb	ระบุว่าการแจ้งเตือนถึงขีดแจ้งเตือนไม่ควรตั้ง เมื่อผู้ใช้พิมพ์ไกล ขอบขวา นี่เป็นค่าดีฟอลต์ ริชอร์สที่เกี่ยวข้อง: marginBell
-ms pointer_color	ระบุสีพื้นหน้าที่จะใช้สำหรับตัวชี้เคอร์เซอร์ (X11) ของ หน้าต่าง ดีฟอลต์คือเพื่อใช้สีพื้นหน้าของหน้าต่างเทอร์มินัล ดูที่ foreground pointer_color ระบุสีพื้นหน้าตัวชี้ที่จะใช้ ริชอร์สที่เกี่ยวข้อง: pointerColor

## ไอเท็ม

-name prog_name	<b>คำอธิบาย</b> ระบุชื่อ X11 ของหน้าต่าง dterm prog_name ชื่อที่จะใช้
-nb number	ระบุจำนวนอักขระจากขอบขวา ซึ่งสัญญาณเตือนขีดขอบจะตั้ง ถ้าเปิดใช้ ค่าดีฟอลต์คือ 10 ริชอร์สที่เกี่ยวข้อง: nMarginBell
-r	ทำให้หน้าต่าง dterm ถูกแสดงพร้อมกับ สีพื้นหน้าและแบ็กกราวด์กลับตรงกันข้าม นี่เหมือนกับแฟล็ก -rv และ -reverse
+r	ทำให้หน้าต่าง dterm ถูกแสดงพร้อมกับ สีพื้นหน้าและแบ็กกราวด์ปกติ นี่เป็นดีฟอลต์ และยังเหมือนกับ แฟล็ก +rv
-reverse	ทำให้หน้าต่าง dterm ถูกแสดงพร้อมกับ สีพื้นหน้าและแบ็กกราวด์กลับตรงกันข้าม นี่เหมือนกับแฟล็ก -r และ -rv
-rv	ทำให้หน้าต่าง dterm ถูกแสดงพร้อมกับ สีพื้นหน้าและแบ็กกราวด์กลับตรงกันข้าม นี่เหมือนกับการเลือก Options   Global Options, แล้วเปลี่ยนอ็อปชันเมนู ``windowBackground`` options เป็น ``Inverse`` หน้าต่าง dterm ที่เริ่มต้นด้วยแฟล็กนี้ มีอ็อปชันเมนู ``Window Background`` เซ็ตเป็น ``Inverse`` ดูที่ ``Global Options``
+rv	ทำให้หน้าต่าง dterm ถูกแสดงพร้อมกับ สีพื้นหน้าและแบ็กกราวด์ปกติ นี่คือ ค่าดีฟอลต์
-rw	ระบุว่าการ reverse-wraparound ควรถูกเปิดใช้งาน ริชอร์สที่เกี่ยวข้อง: reverseWrap
+rw	หมายถึงว่า reverse-wraparound ไม่ควรเปิดใช้ นี่เป็นค่าดีฟอลต์ ริชอร์สที่เกี่ยวข้อง: reverseWrap
-Scn	ระบุว่าการเทอร์มินัลมีเลเตอร์ควรถูกรันกับ pre-opened pty หรืออุปกรณ์ STREAMS แฟล็กนี้ถูกจัดเตรียมสำหรับใช้โดยที่ pty หรือชื่อ slave ของอุปกรณ์ STREAMS มีรูปแบบ tty?? (ตัวอย่าง, อักขระสองตัวตามด้วย tty) แฟล็กนี้มีไว้เพื่อใช้เมื่อ dterm ถูก เรียกใช้ทางโปรแกรมจากแอ็พพลิเคชันอื่น cc ระบุ สองอักขระสุดท้ายของ pty หรือชื่อ slave ของอุปกรณ์ STREAMS ซึ่ง ชื่อ slave มีรูปแบบ tty?? ค่านี้ถูกละเว้น แต่ต้องมีค่านัย สองอักขระเท่านั้น n
-Sc.n	ระบุจำนวนของ ไฟล์ descriptor ที่ตรงกับ pty หรือด้านหลักที่เปิดแล้ว ของอุปกรณ์ STREAMS แฟล็กนี้เหมือนกับ -Scn ตามบน แต่ถูกจัดเตรียม สำหรับระบบที่มี pty name space ขนาดใหญ่กว่า c ระบุ คอมโพเนนต์สุดท้ายของชื่อ pty slave ค่านี้ถูกละเว้นและอาจ วางเปล่า n ระบุจำนวนของไฟล์ descriptor ที่ตรงกับด้านหลักที่เปิดแล้วของ pty
-sb	หมายถึงว่าแถบเลื่อนควรถูกแสดง นี่เป็นค่าดีฟอลต์ ริชอร์สที่เกี่ยวข้อง: scrollbar
+sb	หมายถึงว่าแถบเลื่อนไม่ควรถูกแสดง ริชอร์สที่เกี่ยวข้อง: scrollbar
-sf	หมายถึงว่าโค้ด Sun Function Key escape ควรถูกสร้างสำหรับ ฟังก์ชันคีย์แทน VT220 escape sequences มาตรฐาน ริชอร์สที่เกี่ยวข้อง: sunFunctionKeys

ไอเท็ม	คำอธิบาย
+sf	หมายถึงว่า escape sequences มาตรฐานควรถูกสร้างสำหรับ ฟังก์ชันคีย์แทนโค้ด Sun Function Key escape นี้เป็นการทำงานดีฟอลต์ รีซอร์สที่เกี่ยวข้อง: sunFunctionKeys
-sl screens[sl]	ระบุจำนวนบรรทัดในเทอร์มินัลบัฟเฟอร์ นอกเหนือความยาวของหน้าต่าง ค่าแฟล็กประกอบด้วยหมายเลขตามด้วยคำต่อท้าย เป็นทางเลือก ถ้าไม่มีคำต่อท้าย หรือคำต่อท้ายเป็น l (ell), ความยาวรวมของเทอร์มินัลบัฟเฟอร์จะเป็น screens บวกความยาวของหน้าต่างเทอร์มินัล ถ้าคำต่อท้ายคือ s (ess), ความยาวรวมของเทอร์มินัลบัฟเฟอร์จะเป็น (screens บวกหนึ่ง) คูณ ความยาวของหน้าต่างเทอร์มินัล dterm จะพยายามรักษา อัตราบัฟเฟอร์-ต่อ-หน้าต่าง เดิมไว้ เมื่อหน้าต่างถูกปรับให้ใหญ่ขึ้น ดีฟอลต์คือ 4s screens ระบุจำนวนของหน้าจอหรือบรรทัดที่จะบันทึก รีซอร์สที่เกี่ยวข้อง: saveLines
-ti term_id	กำหนดชื่อที่ใช้เพื่อเลือกการตอบสนองที่ถูกต้องกับ terminal ID เดียวรี ค่าที่ใช้ได้คือ vt100, vt101, vt102 และ vt220 ดีฟอลต์คือ vt220 term_id ระบุ ID เทอร์มินัลที่จะใช้
-title title_string	ระบุชื่อหน้าต่าง ถ้าแฟล็ก -e ถูกใช้ ดีฟอลต์จะเป็นคอมโพเนนต์สุดท้ายของพารามิเตอร์ ถ้าไม่ใช้แฟล็ก -e ดีฟอลต์จะเป็น คอมโพเนนต์สุดท้ายของชื่อที่ใช้เพื่อรัน dterm (เช่น, argv[0]) title_string ระบุหัวเรื่องที่จะใช้ รีซอร์สที่เกี่ยวข้อง: title
-tm term_modes	ระบุสตริงที่มีคีย์เวิร์ด terminal-setting และอักขระ ซึ่งค้นหาคีย์เวิร์ดที่ใช้ได้ประกอบด้วย intr, quit, erase, kill, eof, eol, swtch, start, stop, brk, susp, dsusp, rpmt, flush, weras, และ lnext คีย์เวิร์ดที่ไม่ได้ใช้กับสถาปัตยกรรมจำเพาะจำถูกวิเคราะห์ค่าและละเว้นอย่างถูกต้อง อักขระควบคุมอาจถูกระบุเป็น ^ ตามด้วย อักขระ (เช่น ^c หรือ ^u) และ ^? อาจถูกใช้เพื่อระบุการลบ ซึ่งมีประโยชน์ สำหรับการเขียนทับการตั้งค่าดีฟอลต์เทอร์มินัลโดยไม่ต้องทำ stty ทุกครั้งที่กระบวนการของเทอร์มินัลถูกสตาท์ ดีฟอลต์คือ NULL term_modes ระบุเทอร์มินัลโหมดสตริง รีซอร์สที่เกี่ยวข้อง: ttyModes
-tn term_name	ระบุชื่อเพื่อเซตตัวแปรสภาวะแวดล้อม \$TERM ดีฟอลต์คือ vt220. term_name ระบุชื่อเทอร์มินัลที่จะใช้ รีซอร์สที่เกี่ยวข้อง: termName
-usage	พิมพ์ข้อความการใช้งานบนจอภาพ
-vb	หมายถึงว่าต้องการสัญญาณเตือนที่มองเห็นได้มากกว่าการเตือนด้วยเสียง แทนการ ส่งเสียงสัญญาณเทอร์มินัลเมื่อได้รับ Control-G, หน้าต่าง จะกระพริบ รีซอร์สที่เกี่ยวข้อง: visualBell
+vb	หมายถึงว่าต้องการสัญญาณเตือนด้วยเสียงมากกว่าสัญญาณที่มองเห็นได้ นี้เป็นการทำงานดีฟอลต์ รีซอร์สที่เกี่ยวข้อง: visualBell
-w border_width	เหมือนกับ -borderwidth border_width ระบุความกว้างของขอบหน้าต่างเป็นพิกเซล
-xrm resource_string	อนุญาตให้ระบุรีซอร์ส X11 Resource Manager-style บน บรรทัดคำสั่ง resource_string ระบุ X11 รีซอร์ส สตริง

## รีซอร์ส

ไอเท็ม	คำอธิบาย
allowSendEvents	ระบุว่าเทอร์มินัลอิมูเลเตอร์ควรอนุญาตเหตุการณ์ที่ไม่ได้เกิดขึ้นเอง (ที่สร้างและส่งโดยแอปพลิเคชันอื่น) การเปิดใช้รีซอร์สนี้จะมีความเสี่ยงด้านความปลอดภัย ดีฟอลต์คือ False
appCursorDefault	ถ้าเป็น True เคอร์เซอร์คีย์อยู่ในโหมดแอฟพลิเคชันเมื่อเริ่มต้น ถ้าเป็น False, จะอยู่ในโหมดเคอร์เซอร์เมื่อเริ่มต้น ดีฟอลต์คือ False
appKeypadDefault	ถ้าเป็น True เคอร์เซอร์คีย์แพดอยู่ในโหมดแอฟพลิเคชันเมื่อเริ่มต้น ถ้าเป็น False, จะอยู่ในโหมดตัวเลขเมื่อเริ่มต้น ดีฟอลต์คือ False
autoWrap	ระบุว่า auto-wraparound ถูกเปิดใช้งานตอนเริ่มต้นหรือไม่ ดีฟอลต์คือ True
background	ระบุสีแบ็กกราวนด์ของหน้าต่างเทอร์มินัลและสีแบ็กกราวนด์ดีฟอลต์ที่ใช้สำหรับแถบเลื่อน ภายใต้ CDE, ดีฟอลต์ของรีซอร์ส นี้เป็นดีฟอลต์กับพิกเซลการเลือกชุดสีหรือพิกเซลแบ็กกราวนด์ชุดสี ดูที่ backgroundsSelect ดีฟอลต์เป็นพิกเซลแบ็กกราวนด์ชุดสีหลัก ถ้าไม่มี CDE, รีซอร์สนี้ ดีฟอลต์เป็นสีดำ
backgroundsSelect	เมื่อเป็น True, รีซอร์สนี้ระบุว่าหน้าต่างเทอร์มินัลควรใช้สีการเลือก Motif แทนสีแบ็กกราวนด์สำหรับสีแบ็กกราวนด์ของหน้าต่างเทอร์มินัล ดีฟอลต์คือ False
blinkRate	ระบุจำนวนมิลลิวินาทีที่เคอร์เซอร์อยู่ในสภาวะ เปิดและปิด ขณะกระพริบ ค่า 250 จะกระพริบเคอร์เซอร์สองครั้งต่อวินาที ค่า 0 จะปิดการกระพริบ ดีฟอลต์คือ 250
borderColor	กำหนดสีขอบสำหรับหน้าต่าง เส้นขอบหน้าต่าง อาจมองไม่เห็น เมื่อ reparenting ตัวจัดการหน้าต่างเช่น dtwm และ mwm ถูกใช้ ดีฟอลต์คือ "black"
borderWidth	ระบุความกว้างเส้นขอบของหน้าต่างของเซลล์ widget คำนี้ อาจถูกเขียนทับโดยการ reparenting ตัวจัดการหน้าต่างเช่น dtwm และ mwm ดีฟอลต์คือ 0
c132	ระบุว่า DECCOLM escape sequence ที่สลับ ไปที่หน้าต่างระหว่าง 80 และ 132 คอลัมน์ควรถูก honored ดีฟอลต์คือ False

ไอเท็ม	คำอธิบาย
charCursorStyle	ระบุบุปทรวงของเคอร์เซอร์ข้อความ ค่าของ char_cursor_box ระบุเคอร์เซอร์ ด้วยความกว้างและความสูงของกรอบขอบเขตของฟอนต์หลัก ค่า char_cursor_bar ระบุเคอร์เซอร์ด้วยความกว้างของกรอบขอบเขตของ ฟอนต์หลัก ความสูงสองฟีกเซล และแสดงโดยมีจุดสูงสุดบนเส้นบรรทัด ดีฟอลต์คือ char_cursor_box
consoleMode	ระบุเอาต์พุตที่กำหนดที่ /dev/console ควรถูก ส่งไปที่หน้าต่างเทอร์มินัลแทน เป็นการจัดเตรียมเพื่อป้องกันเอาต์พุตซึ่งปกติจะถูกแสดงบน ITE ไม่ให้เขียนทับจอแสดงผลของ X server โดยไม่ได้ถูกจัดเตรียมไว้เป็นกลไกทั่วไปในการกำหนด เอาต์พุตจาก /dev/console ของระบบที่ไม่ได้กำหนด ไปที่ X server ที่ไม่ได้กำหนด หมายถึงคุณต้องมีสิทธิ เป็นเจ้าของ และ read/write กับ /dev/console เพื่อให้แฟล็กนี้ทำงาน ดีฟอลต์คือ False
foreground	ระบุพื้นหนาของหน้าต่างเทอร์มินัลและพื้นหนาดีฟอลต์ที่ใช้สำหรับแถบเลื่อนและสีที่ใช้สำหรับ เคอร์เซอร์ตัวชี้ ภายใต้ CDE รีซอร์สนี่จะเป็นค่าดีฟอลต์กับพื้นหน้าชุด สีหลัก มีฉะนั้น, จะเป็นค่าดีฟอลต์ ``white``
geometry	ระบุขนาดและตำแหน่งที่ต้องการของหน้าต่างเทอร์มินัล ขนาด ดีฟอลต์คือ 24 บรรทัด 80 อักขระในและบรรทัด ไม่มีตำแหน่งดีฟอลต์
iconGeometry	ระบุตำแหน่งที่ต้องการของไอคอนของเทอร์มินัลอีมูเลเตอร์ ตัวจัดการ หน้าต่างอาจจะเว้นค่านี ไม่มีดีฟอลต์
iconic	ถ้าเป็น true, ระบุว่าเทอร์มินัลอีมูเลเตอร์ควรถูกกำหนดเมื่อเริ่มต้นให้เป็น ไอคอนจอแสดงผล ตัวจัดการหน้าต่าง (รวมถึง dtwm และ mwm อาจจะเว้นค่านี ดีฟอลต์คือ False
iconicName	ระบุชื่อสำหรับไอคอน ถ้าแฟล็ก -c ถูกใช้ ดีฟอลต์จะเป็นคอมโพเนนต์สุดท้ายของพาธโปรแกรม ถ้าไม่ใช้แฟล็ก -c ดีฟอลต์จะเป็น ชื่อหลักของชื่อที่ใช้เพื่อรัน dtterm (เช่น, argv[0])
jumpScroll	ระบุว่าการเลื่อนแบบข้ามควรนำมาใช้ ภายใต้การเลื่อนแบบข้าม, จอภาพอาจถูกเลื่อนมากกว่าหนึ่งบรรทัดต่อครั้ง ซึ่งมีไว้เพื่อ การอัปเดตจอภาพที่เร็วขึ้น เมื่อมีข้อความหลายบรรทัดถูกส่งไปที่เทอร์มินัล จำนวนมากที่สุดของบรรทัดที่อาจถูกข้ามถูกจำกัดโดย จำนวนบรรทัดในจอแสดงผล เป็นการก้ารันที่ว่าทุกบรรทัดจะถูกแสดง ดีฟอลต์คือ True
kshMode	ระบุว่าโหมด ksh ควรถูกเปิดใช้งาน ภายใต้โหมด ksh การกดคีย์โดยมีชุดบิต extend modifier จะสร้าง escape character ที่ตามด้วยอักขระที่สร้างโดยการกดคีย์ un-extended แฟล็กนี้มีให้สำหรับใช้กับโหมดเอ็ดิเตอร์ บรรทัดรับคำสั่ง emacs และ emacs ของ ksh หรือ ied ซึ่งขัดแย้งกับการใช้ปกติของคีย์ meta สำหรับการสร้างอักขระ โบทเดี่ยวแบบขยาย และสำหรับการสร้างอักขระ multi-byte Asian ดีฟอลต์คือ False
logFile	ระบุชื่อของไฟล์ซึ่งบันทึกเอาต์พุตที่อธิบายด้านล่าง ถูกเขียน ถ้าชื่อไฟล์เริ่มต้นด้วยสัญลักษณ์โพเพ์ (!), ส่วนที่เหลือของสตริงถูกยอมรับว่าเป็นคำสั่ง ที่จะถูกใช้เป็นจุดสิ้นสุดของโพเพ์ ชื่อไฟล์ ดีฟอลต์คือ DtermLogXXXXX (โดยที่ XXXXX คือสตริงอักขระเฉพาะ) และถูกสร้างในไดเรกทอรีซึ่ง subprocess ถูกสตาร์ท ถ้าหากอักขระสุดท้าย คือ XXXXX, จะถูกแทนที่ด้วยสตริงอักขระเฉพาะ
logging	เปิดใช้การบันทึกเอาต์พุต เมื่อการบันทึกถูกเปิดใช้งาน เอาต์พุตทั้งหมดที่ได้รับ จาก subprocess ถูกบันทึกไปที่ไฟล์หรือไปที่โพเพ์ไลน์คำสั่ง (ตาม ที่ระบุผ่านแฟล็ก logFile) เนื่องจากข้อมูลกำลัง ถูกบันทึกโดยตรงจาก subprocess, ซึ่งจะรวม escape characters และคู่ บัดแคร/newline ที่ส่งโดยระเบียบบรรทัดของเทอร์มินัล เอาต์พุตอาจถูกเปิดใช้งานและถูกแสดงผ่าน escape sequences ดีฟอลต์คือ False
logInhibit	ระบุว่าการบันทึกถูกอุปกรณ์และไฟล์ควรถูกห้าม ดีฟอลต์คือ False
loginShell	ระบุว่าเชลล์ที่ถูกเริ่มต้นควรเป็นลิสต์อีนเชลล์ (ตัวอย่าง อักขระแรกของ argv[0] จะเป็นเส้นประ บ่งชี้ถึงเชลล์ที่ควรอ่านไปรไฟล์ระบบ และ \$HOME/.profile ของผู้ใช้ (สำหรับ ksh และ sh) หรือ csh.login ของระบบและ \$HOME.login ของผู้ใช้ (สำหรับ csh) ดีฟอลต์คือ False
ไอเท็ม	คำอธิบาย
mapOnOutput	หมายถึงว่าเทอร์มินัลอีมูเลเตอร์ควรมีพ (de-iconify) ตัวเองเมื่อ subprocess เอาต์พุต ถ้ายังไม่ถูกแม็พ (iconified) ระยะเวลาเริ่มต้น ของเวลาระหว่างที่จะไม่แม็พตัวเอง เมื่อ subprocess เอาต์พุตอาจถูกระบุผ่านรีซอร์ส mapOnOutputDelay ดีฟอลต์คือ False
mapOnOutputDelay	ระบุเวลาเป็นวินาทีหลังจากการเริ่มทำงาน ซึ่ง dtterm จะไม่ honor รีซอร์ส mapOnOutput นี่เป็นการอนุญาตเอาต์พุตเริ่มต้น (เช่น เชลล์พร้อมต์) ให้ถูกส่งไปที่เทอร์มินัล โดยไม่มีกรรมแม็พหน้าต่างอัตโนมัติ ดีฟอลต์คือ 0 (ไม่มีหนวง)
marginBell	ระบุว่าสัญญาณควรถูกรันหรือไม่เมื่อผู้ใช้พิมพ์ใกล้ ขอบด้านขวา ดีฟอลต์คือ False
menuBar	ระบุว่าเมนู pulldown ควรถูกแสดง ดีฟอลต์ คือ True
menuPopup	ระบุว่าป๊อปอัพเมนูควรถูกเปิดใช้ ดีฟอลต์ คือ True
nMarginBell	ระบุจำนวนอักขระจากขอบขวาซึ่ง สัญญาณเตือนขีดขอบควรจะตั้ง เมื่อเปิดใช้ ดีฟอลต์คือ 10
pointerBlank	ระบุว่าเคอร์เซอร์ตัวชี้ควรถูกกำหนดไว้ในโหมด blanking ในโหมดนี้ เคอร์เซอร์จะเปิด เมื่อตัวชี้ถูกเคลื่อนย้าย และจะ หายไป ตามเวลาที่กำหนดเป็นวินาที หรือหลังจากมีคีย์บอร์ดอินพุต เกิดขึ้น การหนวงถูกเซ็ทผ่านรีซอร์ส pointerBlankDelay ดีฟอลต์คือ False
pointerBlankDelay	กำหนดจำนวนวินาทีเพื่อรอก่อนที่จะทำให้เคอร์เซอร์ตัวชี้หายไป หลังจากในตัวชี้ถูกย้ายตำแหน่ง ค่า 0 เรียกให้ตัวชี้หายไปเฉพาะ กับคีย์บอร์ดอินพุต ค่าดีฟอลต์คือ 2 วินาที

ไอเท็ม	คำอธิบาย
pointerColor	ระบุสีพื้นหน้าที่จะใช้สำหรับเคอร์เซอร์ตัวชี้ (X11) ของหน้าต่าง ดีฟอลต์คือเพื่อใช้สีพื้นหน้าของหน้าต่างเทอร์มินัล ดูที่ foreground
pointerColorBackground	ระบุสีแบ็กกราวนด์ที่จะใช้สำหรับตัวชี้เคอร์เซอร์ (X11) ของหน้าต่างเทอร์มินัล ดีฟอลต์คือใช้สีแบ็กกราวนด์ของหน้าต่างเทอร์มินัล ดูที่ background
pointerShape	ระบุอักขระฟอนต์เคอร์เซอร์ X ที่จะใช้เป็นตัวชี้เคอร์เซอร์ ควรถูกระบุเป็นสตริงจาก include file ที่มีส่วนนำหน้า XC_removed ค่าดีฟอลต์คือ xterm
reverseVideo	ระบุว่าการแสดงผลกลับสีควรถูกใช้หรือไม่ ดีฟอลต์คือ False
reverseWrap	ระบุว่า reverse-wraparound ควรถูกเปิดใช้หรือไม่ ดีฟอลต์คือ False
saveLines	ระบุจำนวนบรรทัดในเทอร์มินัลบัฟเฟอร์ นอกเหนือความยาวของหน้าต่าง ค่าประกอบด้วยหมายเลขตามด้วยคำต่อท้ายเป็นทางเลือก ถ้าไม่มีคำต่อท้าย หรือคำต่อท้ายเป็น l (ell), ความยาวรวมของเทอร์มินัลบัฟเฟอร์จะเป็น screens บวกความยาวของหน้าต่างเทอร์มินัล ถ้าคำต่อท้ายคือ s (ess), ความยาวรวมของเทอร์มินัลบัฟเฟอร์จะเป็น (screens บวกหนึ่ง) คูณ ความยาวของหน้าต่างเทอร์มินัล dterm จะพยายามรักษา อัตราบัฟเฟอร์ต่อหน้าต่าง เดิมไว้ เมื่อหน้าต่างถูกปรับให้ใหญ่ขึ้น ดีฟอลต์คือ 4s
scrollBar	ระบุว่าแถบเลื่อนควรมองเห็นได้หรือไม่ ค่าดีฟอลต์คือ True
sunFunctionKeys	ระบุว่าโค้ด Sun Function Key escape ควรถูกสร้าง สำหรับฟังก์ชันคีย์ VT220 escape sequences มาตรฐานหรือไม่ ดีฟอลต์คือ False
termId	กำหนดชื่อที่ใช้เพื่อเลือกการตอบสนองที่ถูกต้องกับ terminal ID เคียวีรี ค่าที่ใช้ได้คือ vt100, vt101, vt102 และ vt220 ดีฟอลต์คือ vt220
termName	กำหนดชื่อสำหรับตัวแปรสภาวะแวดล้อม \$TERM ดีฟอลต์คือ vt220
title	ระบุชื่อหน้าต่าง ถ้าแฟล็ก -c ถูกใช้ ดีฟอลต์จะเป็นคอมโพเนนต์สุดท้ายของพารามิเตอร์ ถ้าไม่ใช้แฟล็ก -c ดีฟอลต์จะเป็น คอมโพเนนต์สุดท้ายของชื่อที่ใช้เพื่อรัน dterm (เช่น, argv[0])
ttyModes	ระบุสตริงที่มีคีย์เวิร์ด terminal-setting และอักขระ ซึ่งคนหาพบ คีย์เวิร์ดที่ใช้ได้ประกอบด้วย: intr, quit, erase, kill, eof, eol, swtch, start, stop, brk, susp, dsusp, rprnt, flush, weras, และ Inext คีย์เวิร์ดที่ไม่ได้ใช้กับสถาปัตยกรรมจำเพาะจำถูกวิเคราะห์ค่าและเลขอย่างถูกต้อง อักขระควบคุมอาจถูกระบุเป็น ^ ตามด้วยอักขระ (เช่น ^c หรือ ^u) และ ^? อาจถูกใช้เพื่อระบุการลบ ซึ่งมีประโยชน์มาก สำหรับการเขียนทับการตั้งค่า ดีฟอลต์เทอร์มินัลโดยไม่ต้องทำ stty ทุกครั้งที่กระบวนการของเทอร์มินัลถูกสตาร์ท ดีฟอลต์คือ NULL
userBoldFont	ระบุ XFontSet ที่จะถูกใช้เมื่อแสดงข้อความเทอร์มินัลตัวหนา ซึ่งควรถูกระบุเป็น Motif XmFontList สนับสนุนเฉพาะอักขระหรือฟอนต์ mono space การทำงานเมื่อใช้ฟอนต์ตามสัดส่วนไม่มีกำหนดไว้ ฟอนต์ตัวหนาดีฟอลต์จะถูกสร้างจากชื่อ XLFD ของ userFont ถ้าฟอนต์นั้นไม่มีอยู่ข้อความตัวหนาจะถูกสร้างโดยขีดเส้นทับ (ด้วยหนึ่งพิกเซลออฟเซต) userFont
userFont	ระบุ XFontSet ที่จะถูกใช้เมื่อแสดงข้อความเทอร์มินัล ซึ่งควรถูกระบุเป็น Motif XmFontList สนับสนุนเฉพาะอักขระหรือฟอนต์ mono space การทำงานเมื่อใช้ฟอนต์ตามสัดส่วนไม่มีกำหนดไว้ ฟอนต์ นี้จะไม่ถูกใช้เพื่อแสดงข้อความที่ไม่ใช่เทอร์มินัล (เมนูบาร์ ป็อปอัพเมนู ไดอะล็อก และอื่นๆ). ค่าดีฟอลต์คือใช้ค่า XmNtextFontList ของกระดานข่าวสาร parent (ดูที่ XmBulletinBoard(3X)) ในแบบเดียวกันกับ XmText widget.
visualBell	ระบุว่าต้องการสัญญาณเตือนที่มองเห็นได้มากกว่าการเตือนด้วยเสียง แทนการ ส่งเสียงสัญญาณเทอร์มินัล เมื่อได้รับ CTRL-G หน้าต่าง จะกระพริบ ดีฟอลต์คือ False

## การใช้ตัวชี้

หมายเหตุ: dterm อนุญาตให้คุณเลือกขอบเขตของข้อความ การเลือกใช้หลักจากโมเดลที่ระบุใน Inter-Client Communication Conventions Manual (ICCCM) dterm สนับสนุนการเลือกหลักเท่านั้น คุณสามารถคัดลอกหรือวางข้อความที่เลือกโดยใช้การถ่ายโอนหลัก อินพุตถูกปฏิบัติเป็นคีย์บอร์ดอินพุตและถูกแทรกที่เคอร์เซอร์ การดำเนินการ เลือก/แทรก และการกำหนดดีฟอลต์อธิบายอยู่ทางด้านล่าง

ไอเท็ม	คำอธิบาย
select	ปุ่มซ้ายถูกใช้เพื่อเลือกข้อความที่จะถูกคัดลอก ย้ายตัวชี้ไปที่จุดเริ่มต้นของข้อความที่จะคัดลอก กดปุ่มซ้ายค้างไว้ย้าย เคอร์เซอร์ไปที่จุดสิ้นสุดของข้อความที่จะคัดลอก และปล่อยปุ่ม ข้อความที่เลือกปัจจุบัน ยกเลิกการเลือกได้โดยคลิกปุ่มซ้ายหนึ่งครั้งโดยไม่ต้องขยับเมาส์
insert	ปุ่มกลางวางข้อความจากการเลือกหลัก ทำงานเหมือนกับ เป็นคีย์บอร์ดอินพุต

## การดำเนินการ

ไอเท็ม	คำอธิบาย
bell ( <i>[Percentage]</i> )	การดำเนินการนี้ส่งสัญญาณเตือนของคีย์บอร์ด ตามเปอร์เซ็นต์ที่ระบุสูงกว่า หรือต่ำกว่าวอลุ่มหลัก
break ( )	การดำเนินการนี้ส่งสัญญาณ break ไปที่กระบวนการ child
cancel ( )	การดำเนินการนี้ส่งอักขระ CAN (cancel) ไปที่ กระบวนการ child
do ( )	การดำเนินการนี้ส่ง escape sequence ที่สัมพันธ์กับคีย์ Do ไปที่กระบวนการ child
edit-key ( <i>string</i> )	การดำเนินการนี้ส่ง escape sequence ที่สัมพันธ์กับคีย์ แยกไขที่ตรงกันไปที่กระบวนการ child การแปลคีย์เหล่านี้ขึ้นกับ แต่ละแอฟพลิเคชัน ค่าที่ใช้ได้สำหรับคีย์คีย์คือ find, insert, next, prior, remove และ select
extend-start ( )	สตาร์ทส่วนขยายของข้อความที่เลือกในขณะนี้ extend-end ( )
function-key-execute ( <i>num</i> [ <i>,type</i> ])	<b>หมายเหตุ:</b> ขยายการเลือกปัจจุบัน จำนวนของข้อความที่เลือกขึ้นกับ จำนวนการคลิกเมาส์ การดำเนินการนี้ส่ง escape sequence ที่สัมพันธ์กับฟังก์ชันคีย์ <i>num</i> ที่ตรงกันไปที่กระบวนการ child ค่าที่ใช้ได้สำหรับ <i>num</i> คือ 1 ถึง 35 ถ้านิตถูกเซตเป็นฟังก์ชัน (หรือไม่ได้เซตเลย), escape sequence ที่สัมพันธ์กับฟังก์ชันคีย์ <i>num</i> ถูกส่งไปที่กระบวนการ child ถ้า <i>type</i> ถูกเซตเป็น UDK, จากนั้นสตริงที่สัมพันธ์กับคีย์ที่กำหนดโดยผู้ใช้ <i>num</i> ถูกส่งไปที่กระบวนการ child
grab-focus ( )	การดำเนินการนี้ตั้งต่อนี้ขึ้นกับจำนวนของ การคลิกเมาส์ หนึ่งคลิกจะยกเลิกการเลือกข้อความที่เลือกไว้ และเซตจุดการเลือกที่ตำแหน่งตัวชี้ สองคลิกจะเลือกหนึ่งคำ สามคลิก จะเลือกหนึ่งบรรทัดข้อความ และสี่คลิกจะเลือกข้อความทั้งหมด
hard-reset ( )	การดำเนินการนี้จะทำการฮาร์ดรีเซ็ตบนเทอร์มินัลอิมูเลเตอร์
help ( )	การดำเนินการนี้ส่ง escape sequence ที่สัมพันธ์กับคีย์ DEC VT220 Help ไปที่กระบวนการ child การแปลคีย์นี้ขึ้นกับ แอปพลิเคชัน
keymap ( <i>name</i> )	การดำเนินการนี้กำหนดตารางการแปลใหม่แบบไดนามิก ซึ่งชื่อรีซอร์ส เป็นชื่อที่มีคำต่อท้าย Keymap (ตัวพิมพ์มีความสำคัญ) ชื่อ "None" เรียกคืนตารางการแปลต้นฉบับ
keypad-key-execute ( <i>string</i> )	การดำเนินการนี้ส่ง escape sequence ที่สัมพันธ์กับคีย์ คีย์แปดที่ตรงกันไปที่กระบวนการ child การแปลคีย์เหล่านี้ขึ้นกับ แต่ละแอปพลิเคชัน ค่าที่ใช้ได้สำหรับ <i>string</i> มี: f1 -f4, space, tab, enter, equal, multiply, add, separator, subtract, decimal, divide, และ 0 - 9
move-cursor ( <i>direction</i> )	การดำเนินการนี้ส่ง escape sequence ที่สัมพันธ์กับ การเคลื่อนที่ของเคอร์เซอร์ที่ตรงกันไปที่กระบวนการ child การแปลคีย์เหล่านี้ขึ้นกับ แต่ละแอปพลิเคชัน ค่าที่ใช้ได้สำหรับ <i>direction</i> include: up, down, backward และ forward
redraw-display ( )	การดำเนินการนี้แสดงเนื้อหาของหน้าต่างข้อความใหม่
scroll ( <i>count</i> [ <i>,units</i> ])	การดำเนินการนี้จะเลื่อนหน่วยความจำจอแสดงผลลง ถ้า <i>count</i> น้อยกว่า ศูนย์ หรือเลื่อนขึ้นถ้า <i>count</i> มากกว่าศูนย์ จำนวนบรรทัด ที่เลื่อน ใช้ข้อมูลจาก <i>count</i> และ <i>units</i> ค่าที่ใช้ได้สำหรับ <i>units</i> คือ page, halfpage, หรือ line ดีฟอลต์สำหรับ <i>units</i> คือบรรทัด
select-adjust ( )	การดำเนินการนี้ขยายการเลือก จำนวนของข้อความที่เลือกขึ้นกับ จำนวนการคลิกเมาส์: <ol style="list-style-type: none"> <li>1 คลิก = อักขระ</li> <li>2 คลิก = คำ</li> <li>3 คลิก = บรรทัด</li> <li>4 คลิก = บัพเฟอร์</li> </ol>
select-all ( )	การดำเนินการนี้เลือกข้อความทั้งหมด
select-page ( )	การดำเนินการนี้เลือกข้อความทั้งหมดบนจอภาพ
self-insert ( )	การดำเนินการนี้ส่งอักขระที่เชื่อมโยงกับคีย์ที่กดไปที่ กระบวนการ child
soft-reset ( )	การดำเนินการนี้ทำการซอฟต์แวร์รีเซ็ตเทอร์มินัล
stop ( <i>state</i> )	การดำเนินการนี้ สลับ หรือ สตาร์ท หรือ หยุด กระบวนการอ่าน ข้อมูลจากกระบวนการ child ค่าที่ใช้ได้สำหรับ <i>state</i> คือ toggle, on และ off
string ( <i>string</i> )	การดำเนินการนี้แทรกข้อความที่ระบุ <i>string</i> เหมือนกับ ว่าถูกพิมพ์ <i>string</i> ต้องอยู่ในเครื่องหมายคำพูด ถ้ามีช่องว่างหรืออักขระที่ไม่ใช่ตัวเลขตัวอักษร <i>string</i> ถูกแปลเป็นค่าคงที่อักขระ hex ถ้าเริ่มต้นด้วย อักขระ 0x
tab ( )	การดำเนินการนี้ส่งแท็บไปที่กระบวนการ child
visual-bell ( )	การดำเนินการนี้กระพริบหน้าต่างอย่างรวดเร็ว

ไอเท็ม  
Virtual Bindings

คำอธิบาย  
การเชื่อมคีย์เสมือนขึ้นกับคีย์ คำ การเชื่อมเสมือน ไม่นำมาใช้เมื่อ dterm widget ได้รับอินพุตโฟกัส สำหรับข้อมูล  
เกี่ยวกับการเชื่อมสำหรับปุ่มและคีย์เสมือน ดูที่ VirtualBindings

## ไฟล์

ไอเท็ม  
/usr/bin/diff

คำอธิบาย  
มีคำสั่ง diff

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง Files

การเปลี่ยนทิศทางอินพุตและเอาต์พุต

---

## คำสั่ง du

### วัตถุประสงค์

สรุปการใช้งานดิสก์

### ไวยากรณ์

```
du [-a|-s] [-k] [-m] [-g] [-l] [-r] [-x] [-H|-L] [File ...]
```

### คำอธิบาย

คำสั่ง `du` แสดงจำนวนบล็อกที่ใช้สำหรับไฟล์ ถ้าพารามิเตอร์ `File` ที่ระบุเป็นไดเรกทอรี ไฟล์ทั้งหมดภายในไดเรกทอรีจะถูก  
รายงาน ถ้าไม่มีการกำหนดพารามิเตอร์ `File` คำสั่ง `du` จะใช้ไฟล์ในไดเรกทอรีปัจจุบัน

ถ้าพารามิเตอร์ `File` เป็นไดเรกทอรี แล้ว จำนวนบล็อกที่รายงานเป็นผลรวมของบล็อกที่จัดสรรสำหรับไฟล์ในไดเรกทอรีและ  
บล็อกที่จัดสรรสำหรับตัวไดเรกทอรีเอง

ถ้าอ็อปเจกต์ของคำสั่ง `du` เป็นไฟล์หรือไดเรกทอรีที่มีอยู่ภายใน JFS2 snapshot, คำสั่ง `du` จะให้ข้อมูลสำหรับอ็อปเจกต์  
point-in-time เมื่อ snapshot ถูกสร้าง ข้อมูลนี้ไม่รวมจำนวนที่วางที่ถูกกู้คืน ถ้าตัว snapshot เอง ถูกลบ

ระบุแฟล็ก `-a` รายงานจำนวนของบล็อกในแต่ละไฟล์ ไม่ว่าแฟล็ก `-a` จะถูกใช้หรือไม่ แต่ละไฟล์ที่ระบุโดยพารามิเตอร์ `File`  
จะถูกแสดงเสมอ

การระบุแฟล็ก `-s` รายงานจำนวนรวมบล็อกสำหรับไฟล์ทั้งหมดที่ระบุหรือไฟล์ทั้งหมดในไดเรกทอรี

จำนวนบล็อกรวมถึงบล็อกทางอ้อมของแต่ละไฟล์ จำนวนบล็อกถูกคำนวณในหน่วย 512-ไบต์ไม่ขึ้นกับขนาดคลัสเตอร์ที่ใช้  
โดยระบบ การระบุแฟล็ก `-k` คำนวณจำนวน บล็อกในหน่วย 1024-ไบต์

### Notes:

1. ไฟล์ที่มีหลายลิงก์ถูกนับและเขียนข้อมูลเป็นหนึ่งรายการเท่านั้น
2. จำนวนบล็อกขึ้นอยู่กับขนาดไฟล์เท่านั้น ดังนั้น บล็อกที่ไม่ได้จัดสรร จะไม่ถูกนำมาคำนวณในจำนวนบล็อกที่รายงาน

- ถ้า `du` ไม่สามารถรับไฟล์แอ็ททริบิวต์หรือไม่สามารถ อ่านไดเร็กทอรี จะรายงานข้อผิดพลาดและสถานะจบการทำงานของคำสั่งที่จะได้รับผล

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a	สำหรับแต่ละไฟล์ที่ระบุ แสดงการใช้งานดิสก์ของไฟล์สำหรับ แต่ละไดเร็กทอรีที่ระบุ แสดงการใช้งานดิสก์ของแต่ละไฟล์ภายในไดเร็กทอรี รวมทั้งไดเร็กทอรีย่อยทั้งหมด ตรงกันข้ามกับ แฟล็ก <code>-s</code>
-g	คำนวณจำนวนบล็อกเป็นหน่วย GB แทนหน่วยดีฟอลต์ 512-ไบต์ ค่าเอาต์พุตสำหรับการใช้งานดิสก์จะเป็นตัวเลข อดังดัชนีเนื่องจากค่าของแต่ละหน่วยเป็นไบต์มีค่าสูงมาก
-H	ถ้าลิงก์สัญลักษณ์ถูกระบุบนบรรทัดคำสั่ง คำสั่ง <code>du</code> จะนับขนาดของไฟล์หรือลำดับชั้นไฟล์ที่อ้างอิงถึง โดยลิงก์
-k	คำนวณจำนวนบล็อกเป็นหน่วย 1024-ไบต์ แทนหน่วยดีฟอลต์ 512-ไบต์
-l	จัดสรรบล็อกอย่างเท่ากันระหว่างลิงก์สำหรับไฟล์ที่มีหลายลิงก์ โดยดีฟอลต์ ไฟล์ที่มีลิงก์มากกว่าสองลิงก์ถูกนับเป็นหนึ่งเท่านั้น
-L	ถ้าลิงก์สัญลักษณ์ถูกระบุบนบรรทัดคำสั่งหรือพบระหว่าง ผ่านลำดับชั้นไฟล์ คำสั่ง <code>du</code> จะนับขนาดของไฟล์หรือลำดับชั้นไฟล์ที่อ้างอิงถึง โดยลิงก์
-m	คำนวณจำนวนบล็อกเป็นหน่วย MB แทนหน่วยดีฟอลต์ 512-ไบต์ ค่าเอาต์พุตสำหรับการใช้งานดิสก์จะเป็นตัวเลข อดังดัชนีเนื่องจากค่าของแต่ละหน่วยเป็นไบต์มีค่าสูงมาก
-r	รายงานชื่อของไฟล์และไดเร็กทอรีที่เข้าถึงไม่ได้ นี่คือน่าดีฟอลต์
-s	สำหรับแต่ละไฟล์ที่ระบุ แสดงการใช้งานดิสก์ของไฟล์สำหรับ แต่ละไดเร็กทอรีที่ระบุ แสดงการใช้งานดิสก์โดยรวมของไฟล์ทั้งหมดภายใน ไดเร็กทอรี รวมทั้งไดเร็กทอรีย่อยทั้งหมด แฟล็กนี้ตรงกันข้ามกับแฟล็ก <code>-a</code>
-x	เมื่อหาขนาดไฟล์ หากเฉพาะไฟล์ที่อยู่ใน อุปกรณ์เดียวกันกับไฟล์หรือไดเร็กทอรีที่ระบุโดยพารามิเตอร์ <code>File</code> ตัวอย่าง คุณสามารถระบุไดเร็กทอรีที่มีไฟล์บน อุปกรณ์ต่างๆ ในกรณีนี้แฟล็ก <code>-x</code> แสดงขนาดบล็อกของไฟล์ทั้งหมดที่อยู่ในอุปกรณ์เดียวกันเป็นไดเร็กทอรี

### Notes:

- ถ้ามีการระบุแฟล็กใดแฟล็กหนึ่ง `-k`, `-m` และ `-g` หรือทั้งสองแฟล็ก แฟล็กที่ระบุสุดท้าย จะมีผล เอาต์พุตของการใช้งานดิสก์ที่มีแฟล็ก `-m` และ `-g` จะถูกปิดเศษเป็นค่าใกล้เคียง ตัวเลขทศนิยมตำแหน่งที่สองมากที่สุด
- ถ้าอ็อปชันที่ต้องไม่เกิดร่วมกันทั้งสองคือ `-H` และ `-L` ถูกระบุพร้อมกัน คำสั่งไม่รายงานข้อผิดพลาด อ็อปชันที่ระบุสุดท้าย จะกำหนดลักษณะการทำงานของยูทิลิตี้

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

- เมื่อต้องการสรุปการใช้งานดิสก์ของผังไดเร็กทอรีและแต่ละ subtrees ให้ป้อน:

```
du /home/fran
```

ซึ่งแสดงจำนวนบล็อกดิสก์ในไดเร็กทอรี `/home/fran` และแต่ละ ไดเร็กทอรีย่อยที่อยู่ภายใน

- เมื่อต้องการสรุปการใช้งานดิสก์ของผังไดเร็กทอรีและแต่ละ subtrees ในแบบบล็อก 1024-ไบต์ ให้ป้อน:

```
du -k /home/fran
```

ซึ่งแสดงจำนวนบล็อกดิสก์ 1024-ไบต์ในไดเร็กทอรี `/home/fran` และแต่ละ ไดเร็กทอรีย่อยที่อยู่ภายใน

- เมื่อต้องการสรุปการใช้งานดิสก์ของผังไดเร็กทอรีและแต่ละ subtrees ในแบบบล็อก MB ให้ป้อน:

```
du -m /home/fran
```

ซึ่งจะแสดงจำนวน บล็อกดิสก์เป็น MB โดยปัดเป็นตัวเลขทศนิยมสองตำแหน่งที่เคียงที่สุดในไดเรกทอรี /home/fran และแต่ละไดเรกทอรีย่อยที่อยู่ภายใน

4. เมื่อต้องการสรุปการใช้งานดิสก์ของฝั่งไดเรกทอรีและแต่ละ subtrees ในแบบบล็อก GB ให้ป้อน:

```
du -g /home/fran
```

ซึ่งจะแสดงจำนวน บล็อกดิสก์เป็น MB โดยปัดเป็นตัวเลขทศนิยมสองตำแหน่งที่เคียงที่สุดในไดเรกทอรี /home/fran และแต่ละไดเรกทอรีย่อยที่อยู่ภายใน

5. เมื่อต้องการแสดงการใช้งานดิสก์ของแต่ละไฟล์ ให้ป้อน:

```
du -a /home/fran
```

ซึ่งจะแสดงจำนวนบล็อกดิสก์ที่มีในแต่ละไฟล์และ ไดเรกทอรีย่อยของไดเรกทอรี /home/fran ตัวเลขข้างไดเรกทอรีคือ การใช้งานดิสก์ของฝั่งไดเรกทอรีนั้น หมายเลขข้างไฟล์ปกติคือการใช้งานดิสก์ของไฟล์นั้นเท่านั้น

6. เมื่อต้องการแสดงเฉพาะการใช้งานดิสก์ทั้งหมดของฝั่งไดเรกทอรี ให้ป้อน:

```
du -s /home/fran
```

แฟล็ก -s กำหนดให้คำสั่ง **du** แสดงเฉพาะผลรวมการใช้งานดิสก์ของไดเรกทอรี /home/fran และไฟล์ที่อยู่ภายใน โดยตีฟอลต์คำสั่ง **du** แสดงข้อความแสดงความผิดพลาด ถ้าไม่สามารถอ่านไฟล์หรือไดเรกทอรี

7. เมื่อต้องการแสดงการใช้งานดิสก์ของไฟล์และลำดับชั้นของไฟล์ที่อ้างอิง โดยลิงก์สัญลักษณ์ทั้งหมด เพิ่มเติมจากไฟล์ปกติที่พบระหว่างผ่าน ไดเรกทอรี /home/fran ให้พิมพ์:

```
du -L /home/fran
```

8. เมื่อต้องการรายการการใช้งานดิสก์ของไฟล์หรือลำดับชั้นไฟล์ที่อ้างอิงโดย ลิงก์สัญลักษณ์ mylink, ให้พิมพ์:

```
du -H mylink
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/du	มีคำสั่ง <b>du</b>

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง **df**” ในหน้า 118

ข้อมูลที่เกี่ยวข้อง:

คำสั่งไดเรกทอรี

คำสั่ง Files

---

## คำสั่ง **dump**

### วัตถุประสงค์

ดัมพ์ส่วนที่เลือกของอ็อบเจกต์ไฟล์

# ไวยากรณ์

```
dump { -a -c -d -g -h -l -n -o -p -r -s -t -u -v -H -R -T } [ -zName [,Number] [+zNumber] ] [-tIndex [ +tIndex ] ] [ -X { 32|64|32_64|d64|any } ] File ...
```

หมายเหตุ: อย่าใส่ช่องว่างระหว่างแฟล็ก `-zName` และพารามิเตอร์ `Number`

## คำอธิบาย

คำสั่ง `dump` ดัมพ์ส่วนที่เลือก ของพารามิเตอร์ `File` ที่ระบุ คำสั่ง `dump` ยอมรับอ็อบเจกต์ไฟล์ เก็บอ็อบเจกต์ไฟล์ถาวร และไฟล์เรียกทำงาน

## แฟล็ก

### ไอเท็ม

	คำอธิบาย										
<code>-a</code>	ดัมพ์ส่วนหัวไฟล์เก็บถาวรของแต่ละสมาชิกของแต่ละไฟล์เก็บถาวรที่ระบุ										
<code>-c</code>	ดัมพ์ตารางสตริง										
<code>-d</code>	ดัมพ์ข้อมูลดิบสำหรับแต่ละส่วน										
<code>-g</code>	ดัมพ์สัญลักษณ์โกลบอลในตารางสัญลักษณ์ไฟล์เก็บถาวร										
<code>-h</code>	ดัมพ์ส่วนหัวของส่วน										
<code>-l</code>	ดัมพ์ข้อมูลหมายเลขบรรทัด										
<code>-n</code>	ดัมพ์ข้อมูลส่วนตัวโหนดทั้งหมด										
<code>-o</code>	ดัมพ์แต่ละส่วนหัวที่เป็นทางเลือก										
<code>-p</code>	หยุดการพิมพ์ส่วนหัว										
<code>-r</code>	ดัมพ์ข้อมูลการเปลี่ยนตำแหน่ง										
<code>-s</code>	ดัมพ์ข้อมูลดิบสำหรับการเลือก										
<code>-t</code>	ดัมพ์รายการตารางสัญลักษณ์										
<code>-tIndex</code>	ดัมพ์รายการตารางสัญลักษณ์ดัชนีที่ระบุด้วยพารามิเตอร์ <code>Index</code> ใช้แฟล็ก <code>-t</code> กับแฟล็ก <code>+t</code> ในการระบุช่วงของรายการตารางสัญลักษณ์										
<code>+tIndex</code>	ดัมพ์รายการสัญลักษณ์ในขอบเขตที่จับด้วยพารามิเตอร์ <code>Index</code> ขอบเขตเริ่มที่รายการตารางสัญลักษณ์แรก หรือที่รายการที่ระบุโดยแฟล็ก <code>-t</code>										
<code>-u</code>	ขีดเส้นใต้ชื่อของพารามิเตอร์ <code>File</code>										
<code>-v</code>	ดัมพ์ข้อมูลในการสัญลักษณ์แทนตัวเลข แฟล็กทั้งหมดยกเว้นแฟล็ก <code>-o</code> และแฟล็ก <code>-s</code> สามารถถูกใช้กับแฟล็ก <code>-v</code>										
<code>-zName[,Number]</code>	ดัมพ์รายการหมายเลขบรรทัดสำหรับพารามิเตอร์ <code>Name</code> หรือขอบเขตของรายการหมายเลขบรรทัดที่หมายเลขที่ระบุ										
<code>+zNumber</code>	ดัมพ์หมายเลขบรรทัดทั้งหมดถึงพารามิเตอร์ <code>Number</code>										
<code>-H</code>	ดัมพ์ส่วนหัวของส่วนตัวโหนด แฟล็ก <code>-H</code> ใช้เฉพาะกับไฟล์เรียกทำงาน										
<code>-R</code>	ดัมพ์รายการการเปลี่ยนตำแหน่งสำหรับส่วนลิตเตอร์ แฟล็ก <code>-R</code> ใช้เฉพาะกับไฟล์เรียกทำงาน										
<code>-T</code>	ดัมพ์รายการตารางสัญลักษณ์สำหรับส่วนตัวโหนด แฟล็ก <code>-T</code> ใช้เฉพาะกับไฟล์เรียกทำงาน										
<code>-X mode</code>	ระบุชนิดของอ็อบเจกต์ไฟล์ที่ <code>dump</code> ควรตรวจสอบ <code>mode</code> ต้องเป็นหนึ่งในสิ่งต่อไปนี้:  <table><tbody><tr><td>32</td><td>ประมวลผลเฉพาะอ็อบเจกต์ไฟล์ 32-บิต</td></tr><tr><td>64</td><td>ประมวลผลเฉพาะอ็อบเจกต์ไฟล์ 64-บิต</td></tr><tr><td>32_64</td><td>ประมวลผลอ็อบเจกต์ไฟล์ทั้ง 32-บิตและ 64-บิต</td></tr><tr><td>d64</td><td>ตรวจสอบไฟล์ 64-บิต XCOFF ที่ไม่ต่อเนื่อง (magic number = U803XTOCMAGIC)</td></tr><tr><td>อื่นๆ</td><td>ประมวลผลอ็อบเจกต์ไฟล์ที่สนับสนุนทั้งหมด</td></tr></tbody></table>	32	ประมวลผลเฉพาะอ็อบเจกต์ไฟล์ 32-บิต	64	ประมวลผลเฉพาะอ็อบเจกต์ไฟล์ 64-บิต	32_64	ประมวลผลอ็อบเจกต์ไฟล์ทั้ง 32-บิตและ 64-บิต	d64	ตรวจสอบไฟล์ 64-บิต XCOFF ที่ไม่ต่อเนื่อง (magic number = U803XTOCMAGIC)	อื่นๆ	ประมวลผลอ็อบเจกต์ไฟล์ที่สนับสนุนทั้งหมด
32	ประมวลผลเฉพาะอ็อบเจกต์ไฟล์ 32-บิต										
64	ประมวลผลเฉพาะอ็อบเจกต์ไฟล์ 64-บิต										
32_64	ประมวลผลอ็อบเจกต์ไฟล์ทั้ง 32-บิตและ 64-บิต										
d64	ตรวจสอบไฟล์ 64-บิต XCOFF ที่ไม่ต่อเนื่อง (magic number = U803XTOCMAGIC)										
อื่นๆ	ประมวลผลอ็อบเจกต์ไฟล์ที่สนับสนุนทั้งหมด										

ดีฟอลต์คือการประมวลผลอ็อบเจกต์ไฟล์ที่มีขนาด 32 บิต (ละเว้นไฟล์ที่มีขนาด 64 บิต) `mode` ยังสามารถถูกเซตด้วยตัวแปรสภาวะแวดล้อม `OBJECT_MODE` ตัวอย่าง, `OBJECT_MODE=64` ทำให้ `dump` ประมวลผลอ็อบเจกต์ 64-บิตและ ละเว้นอ็อบเจกต์ 32-บิต แฟล็ก `-X` แทนที่ค่าตัวแปร `OBJECT_MODE`

## ตัวอย่าง

1. เมื่อต้องการดัมพ์ตารางสตริงของไฟล์ a.out ให้ป้อน:  

```
dump -c a.out
```
2. เมื่อต้องการดัมพ์เนื้อหาของส่วนข้อมูล XCOFF ไปที่เอาต์พุตมาตรฐาน ให้ป้อน:  

```
dump -d a.out
```
3. เมื่อต้องการดัมพ์ส่วนหัวอ็อบเจกต์ไฟล์ ให้ป้อน:  

```
dump -o a.out
```
4. เมื่อต้องการดัมพ์ข้อมูลหมายเลขบรรทัดสำหรับไฟล์ a.out ให้ป้อน:  

```
dump -l a.out
```
5. เมื่อต้องการดัมพ์ข้อมูลการเปลี่ยนตำแหน่งสำหรับไฟล์ a.out ให้ป้อน:  

```
dump -r a.out
```
6. เมื่อต้องการดัมพ์เนื้อหาของส่วนข้อความอ็อบเจกต์ไฟล์ a.out ให้ป้อน:  

```
dump -s a.out
```
7. เมื่อต้องการดัมพ์ข้อมูลตารางสัญลักษณ์สำหรับอ็อบเจกต์ไฟล์ a.out ให้ป้อน:  

```
dump -t a.out
```
8. เมื่อต้องการพิมพ์รายการตารางสัญลักษณ์ 20 ถึง 31 โดยไม่มีข้อมูลส่วนหัว ให้ป้อน:  

```
dump -p -t20 +t30 a.out
```
9. เมื่อต้องการดัมพ์ส่วนหัวอ็อบเจกต์ไฟล์เฉพาะจากอ็อบเจกต์ 64-บิตใน lib.a, ให้ป้อน:  

```
dump -X64 -o lib.a
```

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง ar

คำสั่ง size

คำสั่ง a.out

คำสั่ง ar

---

## คำสั่ง dumpcheck

### วัตถุประสงค์

ตรวจสอบเพื่อดูว่าอุปกรณ์ดัมพ์และไดเรกทอรีคัดลอกสามารถที่จะรับ ดัมพ์ของระบบข้อผิดพลาดถูกบันทึกโดยดีฟอลต์ ถ้าจะมีริชอร์สไม่เพียงพอ ในการบรรจุดัมพ์

### ไวยากรณ์

```
/usr/lib/ras/dumpcheck [[-l][ -p ][ -t TimeParameters ][ -P ] ] | [ -r ]
```

## คำอธิบาย

คำสั่ง `/usr/lib/ras/dumpcheck` ถูกใช้เพื่อตรวจสอบ รีซอร์สดีสก์ที่ใช้โดยดัมพ์ของระบบ คำสั่งบันทึกข้อผิดพลาด ถ้าอุปกรณ์ดัมพ์ที่ใหญ่ที่สุดเล็กเกินไปที่จะรับดัมพ์หรือมีที่ว่างไม่เพียงพอ ในไดเรกทอรีคัดลอก เมื่อทำการเพจที่ว่าง

`dumpcheck` โดยปกติรันโดย cron เวลา 3:00 pm ตาม เวลาท้องถิ่นของแต่ละวัน ซึ่งแตกต่างกันไปในการใช้แฟล็ก `-r` เพื่อลบออกจาก `crontab` ของ root หรือ `-t TimeParameters` เพื่อเปลี่ยนเวลาที่ `dumpcheck` ถูกปฏิบัติการ และยังอาจถูกตั้งค่าจาก SMIT `dumpcheck` ถูกเพิ่มให้กับ `crontab` ของ root โดยอัตโนมัติเมื่อการช่วยเหลือเซอวิสถูกติดตั้ง

เพื่อประสิทธิภาพสูงสุด, `dumpcheck` ควรถูกรัน เมื่อระบบถูกโหลดอย่างหนักที่สุดในเวลาดังกล่าว ดัมพ์ของระบบจะมีขนาดสูงสุดเป็นส่วนใหญ่ นอกจากนี้ แม้ว่า `dumpcheck` ฝ้าดูขนาดดัมพ์ยังเป็นเรื่องที่เกิดขึ้นได้ที่ดัมพ์จะไม่พอดีกับอุปกรณ์ดัมพ์หรือไดเรกทอรีคัดลอกเมื่อมีการดำเนินการ ซึ่ง อาจเกิดขึ้นได้ ถ้ามีการใช้งานสูงสุดในโหลดระบบขณะเวลาดัมพ์

ฟังก์ชัน `dumpcheck` ถูกติดตั้งเป็นส่วนหนึ่งของ ชุดไฟล์การช่วยเหลือเซอวิส ที่ติดตั้งโดยอัตโนมัติ

## แฟล็ก

### ไอเท็ม

	คำอธิบาย
<code>-l</code>	บันทึกค่าเตือนที่มีลงในบันทึกข้อผิดพลาด นี่เป็นดีฟอลต์ ถ้าไม่มี การระบุพารามิเตอร์พิมพ์ค่าเตือนที่สร้างทั้งหมดไปที่ <code>stdout</code>
<code>-p</code>	หมายถึงว่าการเปลี่ยนแปลงถูกทำให้เป็นการถาวร นั่นคือมีการ ใช้การปฏิบัติการต่อมาของระบบช่วยเหลือ <code>dumpcheck</code>
<code>-P</code>	แฟล็ก <code>-P</code> ไม่จำเป็นกับแฟล็ก <code>-t</code> และแฟล็ก <code>-r</code> ถ้าแฟล็ก <code>-P</code> ถูกระบุ, <code>dumpcheck</code> เพียงเปลี่ยนรายการ <code>crontab</code> โดยไม่ต้องทำการตรวจสอบใดๆ
<code>-r</code>	ลบรายการ <code>crontab</code> สำหรับฟังก์ชันนี้ ยกเลิกการตั้งค่าอย่างมีประสิทธิภาพ คำสั่งนี้โดยปกติถูกรันโดย cron แฟล็ก <code>-r</code> ต้องถูกระบุเพียงแฟล็กเดียว แฟล็กใช้ไม่ได้กับแฟล็กอื่น
<code>-t TimeParameters</code>	เปลี่ยนเวลาที่ <code>dumpcheck</code> ถูกเรียกใช้งาน แฟล็ก <code>TimeParameters</code> ต้องถูกปิดภายในอัญประกาศเดี่ยว หรือคู่ซึ่งระบุพารามิเตอร์เวลา <code>crontab</code> , หัวพารามิเตอร์แรกของบรรทัดในไฟล์ <code>crontab</code> ดูที่คำสั่ง <code>crontab</code> สำหรับรูปแบบของพารามิเตอร์เวลา แฟล็ก <code>-t</code> ใช้ไม่ได้ กับแฟล็ก <code>-r</code> ถ้าแฟล็ก <code>-t</code> ถูกระบุ, <code>dumpcheck</code> เพียงเปลี่ยนรายการ <code>crontab</code> โดยไม่ต้องทำการตรวจสอบใดๆ

## ความปลอดภัย

คำสั่งนี้สามารถถูกเรียกใช้งานได้โดยผู้ใช้ root เท่านั้น

## ตัวอย่าง

1. เมื่อต้องการตรวจสอบดัมพ์รีซอร์สและให้พิมพ์ผลลัพธ์ไปที่เอาต์พุตมาตรฐาน แทนการบันทึก ให้พิมพ์:

```
/usr/lib/ras/dumpcheck -p
```

เมื่อต้องการ ให้การเปลี่ยนแปลงนี้ถาวร คือให้สร้างในรายการ `crontab` ให้พิมพ์:

```
/usr/lib/ras/dumpcheck -p -P
```

2. เมื่อต้องการให้ `dumpcheck` รันที่เวลา 9:00 am และ 3:00 pm Monday ถึง Friday ให้พิมพ์:

```
/usr/lib/ras/dumpcheck -t "0 9,15 * * 1-5"
```

เมื่อต้องการกลับไปใช้ค่าดีฟอลต์ให้พิมพ์:

```
/usr/lib/ras/dumpcheck -t "0 15 * * *"
```

คุณอาจต้องใช้ SMIT ด้วยในการตั้งค่าเวลา เมื่อ `dumpcheck` ดำเนินการ

3. เมื่อต้องการรันคุณลักษณะนี้ต่อ ให้พิมพ์:

/usr/lib/ras/dumpcheck -r

คุณอาจใช้ SMIT ร่วมกับงานนี้

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง sysdumpdev

สิ่งอำนวยความสะดวกที่มัพระบบ y

---

## คำสั่ง dumpctrl

### Purpose

จัดการดัมพ์ของระบบและดัมพ์ live

### ไวยากรณ์

**dumpctrl -k**

**dumpctrl -R [ l | s ] [-P]**

**dumpctrl -s [-c | -C *comp-path-list*] [-l | -L *comp-alias-list*] [-t | -T *type\_subtype*] [-r] [-u]**

**dumpctrl -qc [-c *comp-path-list*] [-l *comp-alias-list*] [-t *type\_subtype*] [-r] [-u] [-p | -P]**

**dumpctrl -q | [-p | -P]**

**dumpctrl -qs [-p | -P]**

**dumpctrl [-P] [*global\_attribute*]**

**dumpctrl [-c *comp-path-list*] [l *comp-alias-list*] [-t *type\_subtype*] [-r] [-u] [-n | -p | -P | -x] [*per-component\_attribute*]**

### คำอธิบาย

มีดัมพ์คอมโพเนนต์อยู่สองชนิด:

#### คอมโพเนนต์

อ้างอิงถึงคอมโพเนนต์ที่ระบุกับโครงสร้างพื้นฐาน RAS (ที่สร้างด้วยเคอร์เนลเซอร์วิส `ras_register()`)

#### คอมโพเนนต์ legacy

อ้างอิงถึงคอมโพเนนต์ดัมพ์ที่ระบุด้วยเคอร์เนลเซอร์วิส `dmp_add()` หรือ `dmp_ctl()`

คำสั่ง `dumpctrl` ถูกใช้เพื่อรับข้อมูลเกี่ยวกับ คอมโพเนนต์ที่ถูกริจิสเตอร์สำหรับดัมพ์ live หรือดัมพ์ระบบ และ เพื่อเคียวรี และ เปลี่ยนคุณสมบัติดัมพ์

คอมโพเนนต์ถูกระบุด้วยชื่อพารามิเตอร์เต็ม, alias โลจิคัลของอุปกรณ์, ชนิดหรือชนิดย่อย คุณสามารถใช้หลายแฟล็กเพื่อระบุหลายคอมโพเนนต์หรือรายการคอมโพเนนต์

# แฟล็ก

## ต้องมีการระบุอย่างน้อยหนึ่งแฟล็ก

### ไอเท็ม

#### คำอธิบาย

-c *comp-path-*

ระบุคอมโพเนนต์ตามชื่อพาธ ใช้ Wildcards ได้ ใช้คำสั่ง `-c all` เพื่อระบุคอมโพเนนต์ทั้งหมด

*list*

-k

รีเฟรชรายการของดัมพ์ของเคอร์เนล แฟล็กนี้จะรันทุก 5 นาทีโดยดีฟอลต์ ช่วงเวลานี้สามารถเปลี่ยนได้โดยการแก้ไขคำสั่ง `crontab` สำหรับผู้ใช้ `root` และเปลี่ยนรายการสำหรับ `/usr/sbin/dumpctrl -k` สำหรับข้อมูลเพิ่มเติม ดูที่คำสั่ง `crontab` คุณต้องรันคำสั่ง `dumpctrl -k` หลังจากคุณเพิ่มหรือลบดัมพ์ด้วยตัวเอง

หากระบบยังเก็บดัมพ์ใดๆ ไว้ในฮาร์ดไดรฟ์ที่ไม่สามารถเขียนลงในระบบไฟล์ได้อีกก่อนหน้านี้ ระบบจะพยายามเขียนดัมพ์เหล่านั้นและเรียกคืนพื้นที่หน่วยเก็บข้อมูลของตนตอนนี้

-l *comp-alias-*

ระบุคอมโพเนนต์ตาม `alias` ใช้ Wildcards ได้

*list*

-r

ดัมพ์คอมโพเนนต์ย่อยของคอมโพเนนต์ที่ระบุ

-q *cmd*

เคียวรีแอ็คทีวิตีสำหรับดัมพ์ `live` หรือดัมพ์ระบบ

- แฟล็ก `-qc` แสดงแอ็คทีวิตีดัมพ์ `live` เฉพาะคอมโพเนนต์และแอ็คทีวิตีดัมพ์ระบบ แฟล็ก `-qc` สามารถถูกใช้กับแฟล็ก `-p` หรือ `-P` เพื่อเคียวรีแอ็คทีวิตี `persistent` ต่อคอมโพเนนต์ แฟล็ก `-qc` แสดงแอ็คทีวิตีสำหรับคอมโพเนนต์ทั้งหมด หากไม่ระบุแฟล็ก `-c`, `-l` หรือ `-t` หรือ `-c all` เป็นดีฟอลต์

ข้อมูลต่อไปนี้เป็นตัวอย่าง เอาต์พุตสำหรับคำสั่งนี้:

```
dumpctrl -qc -r -l vmm -l proc
```

Component name	Have   Live Dump   System Dump
	Alias   /level   /level
vmm	no   on/3   on/3
.pft	no   on/3   on/3
...	
proc	no   on/4   on/3
...	

- แฟล็ก `-q1` แสดงรายการค่าติดตั้งดัมพ์ `live` แบบโกลบอล แฟล็ก `-q1` สามารถใช้กับแฟล็ก `-p` หรือ `-P` เพื่อเคียวรีค่าติดตั้งดัมพ์ `live` แบบโกลบอลที่ต่อเนื่อง

- แฟล็ก `-qs type` แสดงแอ็คทีวิตีดัมพ์ระบบแบบโกลบอล `-qs` สามารถใช้กับแฟล็ก `-p` หรือ `-P` เพื่อเคียวรีแอ็คทีวิตีดัมพ์ระบบแบบโกลบอล

-r

รวมคอมโพเนนต์ทางด้านล่าง คอมโพเนนต์ที่ระบุในลำดับชั้นคอมโพเนนต์

-Rx

เรียกคืนค่าติดตั้งดัมพ์ไปเป็นดีฟอลต์ `x` สามารถเป็น `l` สำหรับค่าติดตั้งดัมพ์แบบ `live` หรือ `s` สำหรับค่าติดตั้งดัมพ์ระบบ ซึ่งจะรีเซ็ตค่าติดตั้งดัมพ์โกลบอลเท่านั้น คอมโพเนนต์ไม่สามารถถูกแยกระบุได้ จำเป็นต้องใช้แฟล็ก `-P` และบูตอิมเมจใหม่เพื่อให้แน่ใจว่าค่าติดตั้งเหล่านั้นทั้งหมดจะยังใช้ได้ หลังจากรีสตาร์ท

-t *type\_subtype*

ระบุคอมโพเนนต์ตามชื่อ `type_subtype`

-s

แสดงชื่อพาธและชื่อของดัมพ์ `live` ทั้งหมดในที่เก็บดัมพ์ หากระบุคอมโพเนนต์ด้วยแฟล็ก `-c`, `-l` หรือ `-t` รายการของดัมพ์ที่แสดง จะมีดัมพ์ที่มีคอมโพเนนต์ที่ระบุเท่านั้น หากระบุคอมโพเนนต์ด้วยแฟล็ก `-C`, `-L` หรือ `-T` รายการของดัมพ์ที่แสดง จะมีดัมพ์ที่มีคอมโพเนนต์ที่ล้มเหลวที่ระบุเท่านั้น

-C *comp-path-*

ระบุคอมโพเนนต์ตามชื่อพาธ ใช้ Wildcards ได้ ยังอนุญาตให้ใช้ชื่อสแกน `all` เพื่อระบุคอมโพเนนต์ทั้งหมด แฟล็ก `-C` ใช้ได้เฉพาะกับแฟล็ก `-s`

*list*

-L *comp-alias-*

ระบุคอมโพเนนต์ตาม `alias` ใช้ Wildcards ได้ แฟล็ก `-L` ใช้ได้เฉพาะกับแฟล็ก `-s`

*list*

-T *type\_subtype*

ระบุคอมโพเนนต์ตามชื่อ `type_subtype` แฟล็ก `-T` ใช้ได้กับแฟล็ก `-s` เท่านั้น

-u

รวมคอมโพเนนต์ทางด้านบน คอมโพเนนต์ที่ระบุในลำดับชั้นคอมโพเนนต์

## แฟล็กถาวร

ไอเอ็ม	คำอธิบาย
-p	การเปลี่ยนแปลงใช้เฉพาะกับคอมพิวเตอร์ที่สร้างใหม่ ซึ่งเป็นคอมพิวเตอร์โครงสร้างพื้นฐาน RAS ที่สร้างหลังจากคำสั่ง <code>dumpctrl</code> รัน
-P	ทำการเปลี่ยนแปลงที่ระบุให้เป็นถาวร การเปลี่ยนแปลง มียังคงมีผลหลังการรีสตาร์ท หากต้องการ boot image ใหม่ ข้อความจะถูกสร้างขึ้นเพื่อให้คุณทราบ แฟล็ก -P ใช้กับแอตทริบิวต์ คอมพิวเตอร์ การเปิดหรือปิดแบบโกลบอลของดัมพ์ live, ระดับดัมพ์ live โกลบอล, การเปิดใช้หรือปิดใช้งานคอมพิวเตอร์ legacy, และค่ากำหนดอุปกรณ์ดัมพ์ระบบ
-n	การเปลี่ยนแปลงที่ใช้กับคอมพิวเตอร์ที่มีอยู่ แฟล็ก -n เป็นค่าดีฟอลต์ถ้าไม่มีการระบุ -p หรือ -P เมื่อต้องการใช้การเปลี่ยนแปลงกับคอมพิวเตอร์ปัจจุบันและที่สร้างใหม่ ให้ใช้ แฟล็ก -n และ -p
-x	ลบค่ากำหนด persistence นี้ แฟล็ก -x ลบค่ากำหนด (-P) persistence ถาวร ต้องระบุข้อมูลจำเพาะในแบบเดียวกับที่ระบุไว้ดั้งเดิม ด้วยแฟล็ก -P

การกำหนดเอง Recursive-down (ที่ระบุโดยแฟล็ก -r) มีความสำคัญเหนือกว่าการกำหนดเองอื่นทั้งหมดโดยไม่คำนึงถึงลำดับที่ถูกระบุที่สัมพันธ์กับการกำหนดเองที่ไม่ใช่แบบ recursive-down อื่นๆ

ถ้าคุณไม่รู้จักการกำหนดเองที่ได้สร้างขึ้น แต่ต้องการ เรียกคืนค่าติดตั้งระบบดีฟอลต์ คุณสามารถเลือกทำหนึ่งในแอตชันต่อไปนี้:

- ในไฟล์ `/var/adm/ras/rasptune` ให้ลบ บรรทัดที่เกี่ยวข้องกับการกำหนดเองและรันคำสั่ง `bosboot` เพื่อเริ่มต้น AIX
- อ่านไฟล์ `/var/adm/ras/rasptune` เพื่อหา แฟล็กและพารามิเตอร์ที่เหมาะสมที่ระบุไว้ จากนั้น ใช้แฟล็ก -x เพื่อลบการกำหนดเอง รันคำสั่ง `bosboot` และรีสตาร์ท AIX

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับวิธีที่แอตทริบิวต์ดัมพ์ต่างๆ โต้ตอบกับ persistence, ดูที่ตารางแอตทริบิวต์ ดัมพ์ live และดัมพ์ระบบ ใน “แอตทริบิวต์”

## แอตทริบิวต์

แอตทริบิวต์ดัมพ์สามารถนำมาจาก attribute=value ตัวอย่างเช่น

```
dumpctrl dir=/usr/dumps freespc=20
```

ตัวอย่างนี้ เซ็ตไดเรกทอรีดัมพ์เป็น `/usr/dumps`, และทำให้มีขอบเขตพื้นที่ เป็น 20%

มีการจัดบางข้อตัดให้ เช่นแอตทริบิวต์ `ldmpon`, ซึ่งเหมือนกับ `ldmpenable=yes`

ถ้ามีการกำหนดคอมพิวเตอร์แอตทริบิวต์ที่ไม่รู้จักจะถูกผ่านไปยัง callbacks ของคอมพิวเตอร์เหล่านั้นโดยใช้ `RASCD_DMP_PASS_THROUGH`

ตารางต่อไปนี้แสดงแอตทริบิวต์ดัมพ์ live

ตารางที่ 1. แอตทริบิวต์ดัมพ์ Live และดีฟอลต์

แอตทริบิวต์	ค่ากำหนด	ค่าดีฟอลต์
ldmpenable	ระบุว่าดัมพ์ live ถูกเปิดใช้งานหรือไม่ ค่าที่เป็นไปได้คือ yes และ no คุณสามารถใช้แอตทริบิวต์ <code>ldmpon</code> แทน <code>ldmpenable=yes</code> , และแอตทริบิวต์ <code>ldmpoff</code> แทน <code>ldmpenable=no</code>	ใช่ สำหรับข้อมูลเพิ่มเติม โปรดดูที่บันทึก ต่อไปนี้ 1 ในหน้า 291
dir	ระบุชื่อไดเรกทอรี live ดัมพ์	<code>/var/adm/ras/livedump</code>

ตารางที่ 1. แอ็ตทริบิวต์ดัมพ์ Live และดีฟอลต์ (ต่อ)

แอ็ตทริบิวต์	ค่ากำหนด	ค่าดีฟอลต์
freespc	ระบุขีดจำกัดพื้นที่ว่างดัมพ์แบบ live โดยใช้ค่าเลขฐานสิบจาก 0 ถึง 99	25 (หมายถึง 25%)
ldmplevel	ระบุระดับดัมพ์แบบ live โดยใช้ค่าเลขฐานสิบจาก 0 ถึง 9 คุณสามารถระบุแอ็ตทริบิวต์ ldmplevel minimal, ldmplevel normal หรือ ldmplevel detail แทน ldmplevel=1, 3, 7	3 (ปกติ) สำหรับข้อมูลเพิ่มเติม โปรดดูที่บันทึก ต่อไป นี้ 1 ในหน้า 291
heapsz	ระบุขนาดสปีดัมพ์แบบ live โดยใช้ค่าเลขฐานสิบ เป็นเมกะไบต์	0 สำหรับข้อมูลเพิ่มเติม โปรดดูที่บันทึก ต่อไป นี้ 2 ในหน้า 291
duptype	ระบุชนิดการย้ายยังดัมพ์ซ้ำซ้อน ค่าต่อไปนี้ เป็นค่าที่เป็นไปได้: • ทั้งหมด • pre • post • none	ทั้งหมด
maxfreeze	ระบุระยะเวลาหยุดการทำงานระบบที่แนะนำ สูงสุด โดยใช้เลขฐานสิบเป็น มิลลิวินาที	100 ms

ตารางต่อไปนี้แสดงแอ็ตทริบิวต์ดัมพ์ของระบบ

ตารางที่ 2. แอ็ตทริบิวต์ดัมพ์ของระบบและดีฟอลต์

แอ็ตทริบิวต์	ค่ากำหนด	ค่าดีฟอลต์
sdmpenable	ระบุว่าดัมพ์ระบบถูกเปิดใช้งาน ค่าที่เป็นไปได้คือ yes และ no คุณยังสามารถระบุ sdmp on หรือ sdmp off แทน sdmpenable=yes หรือ sdmpenable=no	ใช่ สำหรับข้อมูลเพิ่มเติม โปรดดูที่บันทึก ต่อไป นี้ 3 ในหน้า 291
legacyenable	ระบุว่าคอมโพเนนต์ดัมพ์ legacy ถูก เปิดใช้งาน ค่าที่เป็นไปได้คือ yes และ no คุณยังสามารถ ระบุ legacy on หรือ legacy off แทน legacyenable=yes หรือ legacyenable=no	ใช่
sdmplevel	ระบุระดับดัมพ์ระบบโดยใช้ค่าเลขฐานสิบจาก 0 ถึง 9 คุณสามารถระบุแอ็ตทริบิวต์ sdmplevel minimal, sdmplevel normal หรือ sdmplevel detail แทน sdmplevel=1, 3, 7	3 (ปกติ) สำหรับข้อมูลเพิ่มเติม โปรดดูที่บันทึก ต่อไป นี้ 4 ในหน้า 291
copydir	ระบุชื่อไดเรกทอรีที่คัดลอก	/var/adm/ras

ตารางที่ 2. แอ็ตทริบิวต์ดัมพ์ของระบบและดีฟอลต์ (ต่อ)

แอ็ตทริบิวต์	คำกำหนด	ค่าดีฟอลต์
forcecopy	ระบุว่าแอ็ตทริบิวต์ forcecopy ถูก เปิดใช้งานหรือไม่ ค่าที่เป็นไปได้คือ yes และ no ถ้าดัมพ์ ต้องถูกคัดลอกจากพื้นที่การเพจขณะที่บูต และมีพื้นที่ว่างไม่เพียงพอ ในไดเรกทอรีที่คัดลอก คุณจะได้รับพร้อมท์ให้คัดลอกดัมพ์ไปที่สื่อบันทึกแบบถอดได้ ถ้าค่า forcecopy เป็น yes ถ้าค่าเป็น no ดัมพ์จะไม่ถูกคัดลอกและระบบบูตตามปกติ แม้วาดัมพ์อาจสูญหาย	yes
keyseq	ระบุว่าลำดับคีย์ทำให้เกิดดัมพ์เสมอหรือไม่ ค่าที่เป็นไปได้คือ yes และ no	no
primary	ระบุชื่อพารามิเตอร์ดัมพ์หลัก	/dev/hd6 หรือ /dev/lg_dump1v
secondary	ระบุชื่อพารามิเตอร์ดัมพ์รอง	/dev/sysdumpnu11

**Notes:**

1. แอ็ตทริบิวต์ `ldmpenable` และ `ldmplevel` สามารถถูกระบุโดยมีหรือไม่มีคอมโพเนนต์ ถ้าระบุโดยไม่มี คอมโพเนนต์ แอ็ตทริบิวต์จะใช้กับแอ็ตทริบิวต์โกลบอลที่ตรงกัน
2. แอ็ตทริบิวต์ `heapsz` (ขนาดฮีพ) สามารถตั้งค่าเป็น 0 หมายความว่า ขณะเตรียมข้อมูลเบื้องต้นสำหรับการดัมพ์ ระบบจะคำนวณขนาดฮีพดัมพ์แบบ live ที่ขึ้นอยู่กับจำนวนของหน่วยความจำจริง ซึ่งเป็นค่าต่ำสุด 64 MB และ 1/64 ของขนาดของหน่วยความจำจริง
3. แต่ละคอมโพเนนต์ต้องถูกระบุเมื่อมีการกำหนดแอ็ตทริบิวต์ `sdmpenable` ถ้าไม่มีการกำหนดคอมโพเนนต์ จะไม่สามารถระบุแอ็ตทริบิวต์ `sdmpenable` ได้ เนื่องจากไม่สามารถเปิดใช้งานดัมพ์ระบบได้
4. แอ็ตทริบิวต์ `sdmplevel` สามารถ ถูกระบุโดยมีหรือไม่มีคอมโพเนนต์ ถ้าระบุโดยไม่มีคอมโพเนนต์ จะใช้กับระดับดีฟอลต์ระบบ คอมโพเนนต์ที่มี `sdmplevel` ที่มากกว่าค่า `sdmplevel` โกลบอลไม่ถูกรวมไว้ในดัมพ์ระบบ

ตารางต่อไปนี้จะแสดงแอ็ตทริบิวต์ดัมพ์ live และ persistence

ตารางที่ 3. แอ็ตทริบิวต์ดัมพ์ Live และ persistence

แอ็ตทริบิวต์	รายละเอียด	Persistence
ldmpenable	ดัมพ์แบบ live ถูกเปิดใช้งาน	ควบคุมโดยแฟล็ก persistence ต้องการบูตอิมเมจใหม่ กับแฟล็ก -P
dir	ไดเรกทอรีดัมพ์แบบ live	มีผลทันทีและเมื่อระบบรีสตาร์ท
freespace	ขีดจำกัดพื้นที่ว่างดัมพ์แบบ live	มีผลทันทีและเมื่อระบบรีสตาร์ท
ldmplevel	ระดับดัมพ์แบบ live	ควบคุมโดยแฟล็ก persistence ต้องการบูตอิมเมจใหม่ กับแฟล็ก -P
heapsz	ขนาดฮีปดัมพ์แบบ live	มีผลทันทีและเมื่อระบบรีสตาร์ท
duptype	ชนิดการยับยั้งดัมพ์ที่ซ้ำกัน	มีผลทันทีและเมื่อระบบรีสตาร์ท
maxfreeze	ระยะเวลาหยุดทำงานสูงสุดของระบบที่แนะนำ	มีผลทันทีและเมื่อระบบรีสตาร์ท

หมายเหตุ: Persistence มีผลกับแอตทริบิวต์เฉพาะเมื่อใช้กับ คอมโพเนนต์โครงสร้างพื้นฐาน RAS Persistence ยังควบคุมระดับ ดัมพ์ live โกลบอลที่เปิดใช้หรือปิดใช้สถานะ

ตารางต่อไปนี้จะแสดงแอตทริบิวต์ดัมพ์ของระบบ และ persistence

ตารางที่ 4. แอตทริบิวต์ดัมพ์ของระบบและ persistence

แอตทริบิวต์	คำอธิบาย	Persistence
sdmpenable	ดัมพ์ระบบที่เปิดใช้	ควบคุมโดยแฟล็ก persistence ต้องการบูตอิมเมจใหม่กับแฟล็ก -P
legacyenable	คอมโพเนนต์เวลาแฝงดัมพ์	มีผลทันที, และเมื่อระบบรีสตาร์ท ด้วยแฟล็ก -P ไม่ต้องการบูตอิมเมจใหม่กับแฟล็ก -P
sdmplevel	ระดับดัมพ์ระบบ	ควบคุมโดยแฟล็ก persistence ต้องการบูตอิมเมจใหม่กับแฟล็ก -P
copydir	คัดลอกไดเรกทอรี	มีผลทันทีและเมื่อระบบรีสตาร์ท
forcemove	แสดงเมนูเวลาบูตหากไม่สามารถคัดลอก	มีผลทันทีและเมื่อระบบรีสตาร์ท
keyseq	ลำดับคีย์จะทำให้เกิดดัมพ์เสมอ	มีผลทันทีและเมื่อระบบรีสตาร์ท
primary	อุปกรณ์ดัมพ์หลัก	มีผลทันที, และเมื่อระบบรีสตาร์ท ด้วยแฟล็ก -P ไม่ต้องการบูตอิมเมจใหม่กับแฟล็ก -P
secondary	อุปกรณ์ดัมพ์รอง	มีผลทันที, และเมื่อระบบรีสตาร์ท ด้วยแฟล็ก -P ไม่ต้องการบูตอิมเมจใหม่กับแฟล็ก -P

หมายเหตุ: Persistence มีผลกับแอตทริบิวต์ เมื่อนำไปใช้กับ คอมโพเนนต์

แอตทริบิวต์ copydir, forcemove, keyseq, primary และ secondary ทำงานเหมือนกับคำสั่ง sysdumpdev ซึ่งเหมือนกับที่ระบุด้วยแฟล็ก -d, -D, -k, -K, -p และ -s สำหรับข้อมูลเพิ่มเติม ดูที่คำสั่ง sysdumpdev ใน *Commands Reference, Volume 5*

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
ไม่เป็นศูนย์	มีข้อผิดพลาดเกิดขึ้น คำสั่งนี้ล้มเหลวเนื่องจากเงื่อนไขต่อไปนี้: <ul style="list-style-type: none"> <li>มีพารามิเตอร์ไม่ถูกต้องอย่างน้อยหนึ่งพารามิเตอร์</li> <li>มีแอตทริบิวต์ไม่ถูกต้องอย่างน้อยหนึ่งแอตทริบิวต์</li> <li>ไม่สามารถระบุคอมโพเนนต์ได้</li> <li>ต้องมีการระบุอย่างน้อยหนึ่งคอมโพเนนต์</li> <li>ไม่พบค่ากำหนด persistent (ซึ่งสามารถเกิดขึ้นกับ แฟล็ก -x)</li> </ul>

## ความปลอดภัย

เฉพาะผู้ใช้ root เท่านั้นที่สามารถใช้คำสั่งนี้

---

## คำสั่ง `dumpfs`

### วัตถุประสงค์

ดัมพ์ข้อมูลระบบไฟล์

### ไวยากรณ์

```
dumpfs { FileSystem | Device }
```

### คำอธิบาย

คำสั่ง `dumpfs` พิมพ์ข้อมูล superblock, i-node map, และดิสก์แม็พ สำหรับระบบไฟล์หรือ อุปกรณ์พิเศษที่ระบุ รายการนี้ถูกใช้เพื่อค้นหาข้อมูลระบบ โดยหลักแล้วคำสั่ง `dumpfs` ใช้สำหรับวัตถุประสงค์การดีบั๊ก

คำสั่ง `dumpfs` ยังสามารถรับกับ JFS2 snapshot คำสั่ง `dumpfs` พิมพ์ superblock, snapshot map, และ block map xtree copy สำหรับ snapshot ที่ระบุ

หมายเหตุ: คำสั่ง `dumpfs` จะไม่ทำงานบนดิสก์เก็ต UDF, NFS หรือ JFS

### ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

### ตัวอย่าง

เมื่อต้องการพิมพ์ข้อมูลสำหรับ `/dev/hd4` ให้ป้อน:

```
dumpfs /dev/hd4
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `fsck`” ในหน้า 610

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `mkfs`



---

## e

คำสั่ง AIX ต่อไปนี้ขึ้นต้นด้วยตัวอักษร e

---

### คำสั่ง echo

#### วัตถุประสงค์

เขียนสตริงอักขระไปที่เอาต์พุตมาตรฐาน

#### ไวยากรณ์

echo [ *String* ... ]

#### คำอธิบาย

คำสั่ง echo เขียนสตริงอักขระไปยังเอาต์พุตมาตรฐาน *String* ถูกแบ่ง ด้วยช่องว่าง และอักขระบรรทัดใหม่ตามหลังพารามิเตอร์ *String* สุดท้ายที่ระบุ ถ้าไม่มีการระบุพารามิเตอร์ *String* บรรทัดว่าง (อักขระขึ้นบรรทัดใหม่) จะถูกแสดง

โดยปกติคุณสามารถแยกแยะระหว่างแฟล็กและสตริงที่ขึ้นต้นด้วยเครื่องหมายชิดคั่นโดยใช้ — (เครื่องหมายชิดคั่นคู่) เนื่องจากมีการสนับสนุนการไม่ระบุแฟล็กด้วยคำสั่ง echo — (เครื่องหมายชิดคั่นคู่) ถูกดำเนินการเป็นตัวอักษร

คำสั่ง echo รู้จัก ระเบียบ escape ต่อไปนี้:

ไอเท็ม	คำอธิบาย
\a	แสดงอักขระแฉะเตือน
\b	แสดงอักขระถอยกลับ
\c	ยับยั้งอักขระขึ้นบรรทัดใหม่ซึ่งมีฉะนั้นจะตามอาร์กิวเมนต์สุดท้าย ในเอาต์พุต อักขระทั้งหมดที่ตามด้วยลำดับ \c ถูกละเว้น
\f	แสดงอักขระป้อนกระดาษ
\n	แสดงอักขระขึ้นบรรทัดใหม่
\r	แสดงอักขระปิดแคร์
\t	แสดงอักขระแท็บ
\v	แสดงอักขระแท็บแนวตั้ง
\\	แสดงอักขระ backslash
\O <i>Number</i>	แสดงอักขระ 8-บิต ซึ่งค่า ASCII คือ 0-, 1-, 2- หรือ ตัวเลขฐานแปด 3-หลัก

**หมายเหตุ:** คำสั่ง bsh, ksh และ csh แต่ละคำสั่งมีคำสั่งย่อย echo ในตัว คำสั่ง echo และ bsh และคำสั่งย่อย ksh echo ทำงานเหมือนกัน คำสั่งย่อย csh echo ทำงานไม่เหมือนกับคำสั่ง echo

\" (backslash) คืออักขระ quote ในเชลล์ ซึ่งหมายความว่ามันนอกจาก\" ถูกใช้กับ escape character หรือที่ปิดด้วย เครื่องหมายคำพูด ตัวอย่าง \"\" หรือ ' ' เชลล์จะลบ backslashes ออกเมื่อคำสั่งถูกขยาย

หลังจากส่วนขยายเชลล์ คำสั่ง echo เขียนเอาต์พุตจากลำดับ escape ในอินพุต อ้างอิงถึงตาราง Backslash Reduction สำหรับการเปรียบเทียบตัวอย่างของวิธีที่ backslashes ใน คำสั่งถูกลดออกในครั้งแรกโดยเชลล์ และจากนั้นโดยคำสั่ง echo:

#### การลด Backslash

คำสั่งที่ป้อน	หลังการขยายเซลล์	หลังการประมวลผลคำสั่ง echo
echo hi\\there	echo hi\there	hi\there
echo 'hi\\there'	echo 'hi\\there'	hi\there
echo "hi\\there"	echo "hi\there"	hi\there

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย

0 ดำเนินการเป็นผลสำเร็จ

>0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการเขียนข้อความไปที่เอาต์พุตมาตรฐานให้ป้อน:

```
echo Please insert diskette . . .
```

2. เมื่อต้องการแสดงข้อความที่มีอักขระพิเศษให้ป้อน:

```
echo "\n\nI'm at lunch.\nI'll be back at 1:00."
```

ซึ่งจะข้ามสามบรรทัดและแสดงข้อความ:

```
I'm at lunch.
```

```
I'll be back at 1:00.
```

**หมายเหตุ:** คุณต้องนำข้อความไปไว้ในเครื่องหมายคำพูดถ้ามีลำดับ escape มิฉะนั้น เซลล์จะแปล\`(backslash)` เป็น metacharacter และปฏิบัติกับ\`ต่างกัน`

3. เมื่อต้องการใช้คำสั่ง `echo` กับอักขระการจับคู่รูปแบบให้ป้อน:

```
echo The back-up files are: *.bak
```

การใช้งานนี้แสดงข้อความ The back-up files are: ตามด้วยชื่อไฟล์ในไดเรกทอรีปัจจุบันปิดท้ายด้วย `.bak`

4. เมื่อต้องการเพิ่มหนึ่งบรรทัดข้อความให้กับไฟล์ให้ป้อน:

```
echo Remember to set the shell search path to $PATH. >>notes
```

การใช้งานนี้เพิ่มข้อความที่จุดสิ้นสุดของหมายเหตุไฟล์ หลังเซลล์แทนค่าตัวแปรเซลล์ `PATH`

5. เมื่อต้องการเขียนข้อความไปที่เอาต์พุตข้อผิดพลาดมาตรฐานให้ป้อน:

```
echo Error: file already exists. >&2
```

คำสั่งนี้เปลี่ยนทิศทางข้อความแสดงข้อผิดพลาดไปที่ข้อผิดพลาดมาตรฐาน ถ้า `>&2` ถูกละเว้น ข้อความจะถูกเขียนไปที่เอาต์พุตมาตรฐาน

## File

ไอเท็ม  
/usr/bin/echo

คำอธิบาย  
มีคำสั่ง echo

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง bsh

คำสั่ง ksh

คำสั่ง printf

การเปลี่ยนทิศทางอินพุตและเอาต์พุต

---

## คำสั่ง ed หรือ red

### วัตถุประสงค์

เอดิเตอร์รายบรรทัดสำหรับไฟล์ข้อความ

### ไวยากรณ์

`ed [-p String] [-s|-] [File]`

`red [-pString] [-s|-] [File]`

### คำอธิบาย

คำสั่ง `ed` สตาร์ทโปรแกรมการแก้ไขรายบรรทัดเอดิเตอร์ `ed` เอดิเตอร์ `ed` ทำงานเพียงหนึ่งไฟล์ต่อหนึ่งครั้ง โดยการคัดลอกไฟล์ลงในบัฟเฟอร์แก้ไขชั่วคราวและทำการเปลี่ยนแปลงกับสำเนา นั้น เอดิเตอร์ `ed` เป็นส่วนหนึ่งของกลุ่มเอดิเตอร์ที่รวมอยู่ในเอดิเตอร์ `edit`, เอดิเตอร์ `ex`, และเอดิเตอร์ `vi` เอดิเตอร์ `ed` ทำการเปลี่ยนแปลงที่คุณระบุในบัฟเฟอร์ ไม่มี การเปลี่ยนแปลงกับตัวไฟล์ จนกว่าคุณจะใช้คำสั่งย่อย `write (w)`

คุณสามารถระบุชื่อไฟล์ที่คุณต้องการแก้ไข เมื่อคุณสตาร์ทเอดิเตอร์ `ed` ด้วยคำสั่ง `ed` หรือคุณสามารถใช้คำสั่งย่อย `e` เมื่อคำสั่ง `ed` อ่านบรรทัดใหม่ลงในบัฟเฟอร์, เนื้อหาของไฟล์แทนที่เนื้อหาก่อนหน้านี้ของบัฟเฟอร์

คำสั่ง `red` เป็นเวอร์ชันที่จำกัด ของคำสั่ง `ed` สำหรับใช้กับเชลล์ที่จำกัด (`rsh`) ด้วยคำสั่ง `red`, คุณแก้ไขเฉพาะไฟล์ที่อยู่ในไดเรกทอรีปัจจุบันหรือในไดเรกทอรี `/tmp`; คุณไม่สามารถใช้ ! ก่อน

คำสั่งย่อยของเอดิเตอร์ `ed` ประกอบด้วย `zero`, `one` หรือ `two` addresses, ตามด้วยคำสั่งย่อยอักขระเดียว ต่อด้วยพารามิเตอร์เป็นทางเลือกกับคำสั่งย่อยนั้น แอดเดรสระบุบรรทัด ในบัฟเฟอร์ เนื่องจากทุกคำสั่งย่อยมีดีฟอลต์แอดเดรส บ่อยครั้ง จึงไม่จำเป็นต้องระบุแอดเดรส

เอดิเตอร์ `ed` อนุญาตการแก้ไขเฉพาะบรรทัดปัจจุบัน นอกจากคุณแอดเดรสบรรทัดอื่นในบัฟเฟอร์ คุณสามารถย้ายและคัดลอกเฉพาะ บรรทัดข้อมูลที่สมบูรณ์ เอดิเตอร์ `ed` มีประโยชน์สำหรับการแก้ไขไฟล์ขนาดใหญ่หรือ สำหรับการแก้ไขภายในโปรแกรมเชลล์

เอดิเตอร์ `ed` ทำงานหนึ่งในสองโหมด:

ไอเท็ม	คำอธิบาย
โหมดคำสั่ง	ในโหมดคำสั่ง เดีเตออร์ ed รู้จักและรันคำสั่งย่อย เมื่อคุณสตาร์ทเดเตออร์ ed จะทำงานในโหมดคำสั่ง พิมพ์ . (จุด) และกด Enter เพื่อยืนยันว่าคุณอยู่ในโหมดคำสั่ง
โหมดอินพุตข้อความ	ในโหมดอินพุตข้อความ เดีเตออร์ ed อนุญาตให้คุณป้อนข้อความลงใน ไฟล์บัฟเฟอร์แต่ไม่รู้จักคำสั่งย่อย คุณเข้าสู่โหมดอินพุตข้อความ โดยใช้คำสั่งย่อย a, คำสั่งย่อย c หรือคำสั่งย่อย i คุณออกจากโหมดอินพุตข้อความและกลับไปโหมดคำสั่งโดยการพิมพ์ . (จุด) จุดเดียว ที่จุดเริ่มต้นบรรทัด เมื่อต้องการกำหนด . (จุด) ลงใน บัฟเฟอร์ขณะอยู่ในโหมดอินพุตข้อความ ให้ป้อนอักขระที่ตามด้วย . (จุด) จากนั้น ออกโหมดอินพุตข้อความและใช้คำสั่งย่อย s เพื่อลบอักขระ

รายการต่อไปนี้มีข้อจำกัดสูงสุดของเดเตออร์ ed edito

- 64 อักขระต่อชื่อไฟล์
- 256 อักขระต่อรายการคำสั่งย่อยโกลบอล
- ขนาดบัฟเฟอร์ 128,000 อักขระ

หมายเหตุ: บัฟเฟอร์ มีไฟล์ต้นฉบับและข้อมูลการแก้ไข

จำนวนบรรทัดสูงสุดขึ้นกับจำนวน ของหน่วยความจำที่มี ขนาดไฟล์สูงสุดขึ้นกับจำนวนที่เก็บข้อมูล ฟิสิคัล (ดิสก์หรือเทป ไดรฟ์) ที่มีหรือจำนวนบรรทัดสูงสุด ที่มีได้ในหน่วยความจำผู้ใช้

## แฟล็ก

ไอเท็ม	คำอธิบาย
-p <i>String</i>	เซตพร้อมเดเตออร์เป็นพารามิเตอร์ <i>String</i> ดีฟอลต์สำหรับ <i>String</i> คือค่า null (ไม่มีพร้อม)
-s	ยับยั้งจำนวนอักขระที่เดเตออร์แสดงกับคำสั่งย่อย e, คำสั่งย่อย r, และคำสั่งย่อย w แฟล็กนี้ยังยับยั้งข้อความวินิจฉัยสำหรับ คำสั่งย่อย e และคำสั่งย่อย q, และยับยั้ง ! (เครื่องหมายตกใจ) พร้อมหลังจาก ! ก่อน
-	มีการทำงานเหมือนกับแฟล็ก -s

## การจับคู่รูปแบบ

เดเตออร์ ed สนับสนุนฟอร์มที่จำกัดของอักขระการจับคู่รูปแบบ พิเศษคุณสามารถใช้เป็น regular expressions (REs) เพื่อสร้างสตริงรูปแบบ คุณสามารถใช้รูปแบบเหล่านี้ในแอตเตรสเพื่อระบุบรรทัดและใน บางคำสั่งย่อยเพื่อระบุส่วนของบรรทัด

## นิพจน์ทั่วไป

RE ต่อไปนี้จับคู่อักขระเดี่ยวหรือเปรียบเทียบ ส่วนประกอบต่อไปนี้:

ไอเท็ม	คำอธิบาย
<i>Character</i>	จับคู่กับอักขระธรรมดา (ที่ไม่ใช่หนึ่งในสัญลักษณ์การจับคู่รูปแบบพิเศษ)
.	จับคู่อักขระเดี่ยวทั้งหมดยกเว้นอักขระขึ้นบรรทัดใหม่

ไอเท็ม  
[String]

คำอธิบาย

จับคู่หนึ่งอักขระในสตริง ในบางอักขระการจับคู่รูปแบบมีความหมายพิเศษเมื่ออยู่ในวงเล็บ ดังนี้:

- ^ จับคู่อักขระทั้งหมดยกเว้นอักขระในพารามิเตอร์ String และอักขระขึ้นบรรทัดใหม่ ถ้าอักขระแรกของพารามิเตอร์ String คือ ^ (circumflex) เื่อนไซน์เป็น true เฉพาะเมื่อ ^ เป็นอักขระแรกในสตริง, [^String]
- ระบุขอบเขตของอักขระ ASCII ต่อเนื่องตาม ลำดับการเรียงปัจจุบัน ตัวอย่าง, [a-f] อาจเท่ากับ [abcdef] หรือ [aAbBcCdDeEfF] หรือ [abcdef] และอาจรวมถึง อักขระ a และ e ที่มีการเน้นเสียง ลำดับการเรียงสามารถกำหนดคลาส equivalence สำหรับอักขระ  
เครื่องหมายลบเสียความสำคัญ ถ้าเป็นอักขระแรก ในสตริง, [-String]; ถ้าตามหลัง circumflex เริ่มต้น, [^String]; หรือถ้าเป็นอักขระสุดท้ายในสตริง, [String-]
- ] ฟังก์ชันเป็นส่วนหนึ่งของสตริง มากกว่าที่จะเป็นตัวปิดท้ายสตริง, เมื่อ ] (วงเล็บเหลี่ยมขวา) เป็นอักขระแรก ในสตริง, [ ]String], หรือเมื่อตาม circumflex เริ่มต้น, [^]String]

## การฟอร์มรูปแบบ

กฎต่อไปนี้อธิบายวิธีฟอร์มรูปแบบจาก RE:

- RE ที่ประกอบด้วยอักขระ เดียว, ธรรมดา จับคู่อักขระที่เหมือนกันในสตริง
- RE ตามด้วย \* (เครื่องหมายดอกจัน) จับคู่ตั้งแต่ศูนย์อักขระหรือมากกว่า ที่ตรงกับ RE ตัวอย่าง, รูปแบบต่อไปนี้:  
ab\*cd

ตรงกับสตริง แต่ละสตริงต่อไปนี้:

acd  
abcd  
abbcd  
abbbcd

แต่ไม่ตรงกับสตริงต่อไปนี้:

abd

ถ้า มีอ็อพชัน สตริงทางซ้ายที่ตรงกันยาวที่สุดจะถูกเลือก ตัวอย่าง, กำหนดสตริงต่อไปนี้:

122333444

รูปแบบ .\* ตรงกับ 122333444, รูปแบบ .\*3 ตรงกับ 122333 และรูปแบบ .\*2 ตรงกับ 122

- RE ตามด้วย:

ไอเท็ม

คำอธิบาย

\{m\}

ตรงกัน *อย่างน้อย*  $m$  ครั้ง กับอักขระที่จับคู่โดย RE

\{m,\}

ตรงกัน *อย่างน้อย*  $m$  ครั้งกับอักขระ ที่จับคู่โดย RE

\{m,n\}

ตรงกัน *ไม่จำกัด* ครั้งกับ อักขระที่จับคู่โดย RE จาก  $m$  ถึง  $n$  อย่างครอบคลุม

ตัวเลข  $m$  และ  $n$  ต้อง เป็นจำนวนเต็มจาก 0 ถึง 255, อย่างครอบคลุม เมื่อใดก็ตามที่มีอ็อพชัน รูปแบบนี้ จับคู่มากเท่าที่พบ

- คุณสามารถรวม RE ลงในรูปแบบที่จับคู่สตริงที่มี ลำดับอักขระเหมือนกัน ตัวอย่าง รูปแบบ  $AB^*CD$  จับคู่กับสตริง  $AB^*CD$  และรูปแบบ  $[A-Za-z]^*[0-9]^*$  จับคู่กับสตริงที่มีการรวม กันของอักขระแบบตัวอักษร (รวมทั้ง ไม่มี), ตามด้วยการรวมกันของ ตัวเลข (รวมทั้ง ไม่มี)
- ลำดับอักขระ  $\backslash(\text{Pattern})$  ทำเครื่องหมายรูปแบบย่อย ที่ตรงกับสตริงเดียวกัน ที่ลำดับจะจับคู่ถ้าไม่อยู่ในวงเล็บ

- อักขระ `\Number` จับคู่สตริงอักขระเดียวกัน ที่รูปแบบย่อยจับคู่ ก่อนหน้าในรูปแบบ (ดูที่กฎ ลำดับความสำคัญ) รูปแบบของพารามิเตอร์ `Number` แสดง ดิจิต รูปแบบ `\Number` จับคู่สตริงที่ตรงกัน ตามรูปแบบย่อยที่พบซึ่งระบุโดยพารามิเตอร์ `Number` นับจากซ้ายไปขวา

ตัวอย่าง, รูปแบบต่อไปนี้:

`\(A)\(B)C\2\1`

ตรงกับสตริง ABCBA คุณสามารถซ่อนรูปแบบย่อย

### การจำกัดสิ่งที่รูปแบบจับคู่

คุณสามารถจำกัดรูปแบบที่จะจับคู่เฉพาะเชกเมนต์แรกของบรรทัด เชกเมนต์สุดท้าย หรือทั้งบรรทัด รูปแบบ `null`, `//` (สอง slashes), ทำซ้ำรูปแบบก่อนหน้า

### การจับคู่เชกเมนต์แรกของบรรทัด

พารามิเตอร์ `^Pattern` จับคู่ เฉพาะสตริงที่เริ่มต้นในตำแหน่งอักขระแรกของบรรทัด

### การจับคู่เชกเมนต์สุดท้ายของบรรทัด

พารามิเตอร์ `Pattern$` จับคู่ เฉพาะสตริงที่จบด้วยอักขระสุดท้าย (ไม่รวมอักขระ ขึ้นบรรทัดใหม่) บนบรรทัด

### การจับคู่ทั้งบรรทัด

พารามิเตอร์ `^Pattern$` จำกัดรูปแบบ ที่จะจับคู่บรรทัดทั้งหมด

### การกำหนดแอดเดรสบรรทัด

เอดิเตอร์ `ed` ใช้ชนิดแอดเดรสสามชนิด: แอดเดรส หมายเลขบรรทัด, แอดเดรสที่สัมพันธ์กับบรรทัดปัจจุบัน และแอดเดรสรูปแบบ บรรทัดปัจจุบัน (ปกติคือบรรทัดสุดท้ายที่ได้รับผลจากคำสั่งย่อย) คือ จุดอ้างอิงในบัพเพอร์

คุณสามารถใช้การกำหนดแอดเดรสบรรทัดเพื่อทำดังต่อไปนี้:

- กำหนดบรรทัดปัจจุบันใหม่
- แสดงบรรทัดที่แอดเดรส
- ทำให้คำสั่งทำงานบนบรรทัดหรือหลายบรรทัดที่เจาะจง

คำสั่งย่อยที่ไม่รับแอดเดรสจะพิจารณา การมีอยู่ของแอดเดรสเป็นข้อผิดพลาด คำสั่งย่อยที่รับแอดเดรสสามารถใช้ แอดเดรสที่กำหนดหรือแอดเดรสดีฟอลต์ เมื่อกำหนดแอดเดรสให้มากกว่าที่จะรับได้ คำสั่งใช้แอดเดรสสุดท้าย (ขวาสุด)

ในกรณีส่วนใหญ่ คอมมา (,) แยกแอดเดรส (ตัวอย่าง 2,8) เซมิโคลอน (;) ยังสามารถแยกแอดเดรส เซมิโคลอนระหว่างแอดเดรส ทำให้เอดิเตอร์ `ed` ทำการเซตบรรทัดปัจจุบัน เป็นแอดเดรสแรก แล้วคำนวณแอดเดรสที่สอง (ตัวอย่าง, เมื่อต้องการเซตบรรทัดเริ่มต้นเพื่อการค้นหา) ในคู่ของแอดเดรส แอดเดรสแรก ต้องมีจำนวนทางตัวเลขน้อยกว่าแอดเดรสที่สอง

คุณสามารถใช้หมายเลขบรรทัดและการกำหนดแอดเดรสเชิงสัญลักษณ์ เพื่อดำเนินงานต่อไปนี้:

- การกำหนดแอดเดรสบรรทัดปัจจุบัน
- การกำหนดแอดเดรสบรรทัดตามตัวเลข

- การกำหนดแอดเดรสบรรทัดก่อนบรรทัดแรก
- การกำหนดแอดเดรสบรรทัดสุดท้าย
- การกำหนดแอดเดรสบรรทัดเหนือบรรทัดที่แอดเดรส
- การกำหนดแอดเดรสบรรทัดใต้บรรทัดที่แอดเดรส
- การกำหนดแอดเดรสบรรทัดแรกผ่านบรรทัดสุดท้าย
- การกำหนดแอดเดรสบรรทัดปัจจุบันผ่านบรรทัดสุดท้าย
- การกำหนดแอดเดรสกลุ่มบรรทัด
- การกำหนดแอดเดรสบรรทัดถัดไปที่มีรูปแบบที่ระบุ
- การกำหนดแอดเดรสบรรทัดก่อนหน้าที่มีรูปแบบที่ระบุ
- การกำหนดแอดเดรสบรรทัดที่ทำเครื่องหมาย

### การระบุบรรทัด ปัจจุบัน

. (จุด) แอดเดรส บรรทัดปัจจุบัน . (จุด) เป็นดีฟอลต์สำหรับคำสั่งย่อยเอดิเตอร์ ed ส่วนใหญ่ และไม่จำเป็นต้องถูกระบุ

### การกำหนดแอดเดรสบรรทัดตามตัวเลข

เมื่อต้องการแอดเดรสบรรทัดของบัฟเฟอร์ที่ระบุให้พิมพ์:

Number

โดยที่พารามิเตอร์ *Number* แสดงหมายเลขบรรทัด ตัวอย่างเช่น :

2253

แอดเดรสหมายเลขบรรทัด 2253 เป็นบรรทัดปัจจุบัน

### การกำหนดแอดเดรสบรรทัดก่อนบรรทัดแรก

เมื่อต้องการแอดเดรสบรรทัดก่อนบรรทัดแรกของบัฟเฟอร์ให้พิมพ์:

0

### การกำหนดแอดเดรสบรรทัดสุดท้าย

เมื่อต้องการแอดเดรสบรรทัดสุดท้ายของบัฟเฟอร์ให้พิมพ์:

\$

### การกำหนดแอดเดรสบรรทัดเหนือบรรทัดที่แอดเดรส

เมื่อต้องการระบุแอดเดรสที่เป็นจำนวนบรรทัด ที่ระบุเหนือบรรทัดปัจจุบัน ให้พิมพ์:

-Number

โดยที่พารามิเตอร์ *Number* คือจำนวนบรรทัดที่ระบุเหนือบรรทัดปัจจุบัน ที่คุณต้องการที่จะแอดเดรส ตัวอย่างเช่น :

-5

แอดเดรสห้าบรรทัดเหนือบรรทัดปัจจุบัน เป็นบรรทัดปัจจุบัน

คุณยังสามารถระบุเพียง – เพื่อแอดเดรสบรรทัดเหนือ บรรทัดปัจจุบันได้ทันที เครื่องหมายลบมีเครื่องหมายลบมีผลสะสม ตัวอย่าง address – – (เครื่องหมายลบสองเครื่อง) แอดเดรสสองบรรทัดเหนือ บรรทัดปัจจุบัน

การกำหนดแอดเดรสบรรทัดใต้บรรทัดที่แอดเดรส

เมื่อต้องการระบุแอดเดรสที่เป็นจำนวนบรรทัด ที่ระบุใต้บรรทัดปัจจุบัน ให้พิมพ์:

+Number

โดยที่พารามิเตอร์ *Number* คือจำนวนบรรทัดที่ระบุใต้บรรทัดปัจจุบัน ที่คุณต้องการที่จะแอดเดรส + (เครื่องหมายบวก) เป็นทางเลือก ตัวอย่างเช่น :

+11

แอดเดรส 11 บรรทัดใต้บรรทัดปัจจุบัน เป็นบรรทัดปัจจุบัน

คุณยังสามารถระบุเพียง + เพื่อแอดเดรสบรรทัดใต้ บรรทัดปัจจุบันได้ทันที + มีผลสะสม ตัวอย่าง address + + (เครื่องหมายบวกสองเครื่อง) แอดเดรสสองบรรทัดใต้บรรทัดปัจจุบัน

การกำหนดแอดเดรสบรรทัดแรกถึงบรรทัดสุดท้าย

เมื่อต้องการแอดเดรสบรรทัดแรกถึงบรรทัดสุดท้าย ให้พิมพ์:

,

, (คอมมา) หมายถึงคูแอดเดรส 1,\$ (บรรทัดแรก ถึงบรรทัดสุดท้าย) บรรทัดแรกกลายเป็นบรรทัดปัจจุบัน

การระบุบรรทัด ปัจจุบัน จนถึงบรรทัด สุดท้าย

เมื่อต้องการแอดเดรสบรรทัดปัจจุบันถึงบรรทัดสุดท้าย ให้พิมพ์:

;

; (เซมิโคลอน) หมายถึงคูแอดเดรส .,\$ (บรรทัดปัจจุบัน ถึงบรรทัดสุดท้าย)

การกำหนดแอดเดรสกลุ่มบรรทัด

เมื่อต้องการแอดเดรสกลุ่มบรรทัด ให้พิมพ์:

FirstAddress,LastAddress

โดยที่พารามิเตอร์ *FirstAddress* คือหมายเลขบรรทัด (หรือแอดเดรสเชิงสัญลักษณ์) ของบรรทัดแรกในกลุ่มที่ คุณต้องการแอดเดรส และพารามิเตอร์ *LastAddress* คือ หมายเลขบรรทัด (หรือแอดเดรสเชิงสัญลักษณ์) ของบรรทัดสุดท้ายในกลุ่มบรรทัด แรกในกลุ่มกลายเป็นบรรทัดปัจจุบัน ตัวอย่างเช่น:

3421,4456

แอดเดรสบรรทัด 3421 ถึง 4456 บรรทัด 3421 กลายเป็น บรรทัดปัจจุบัน

การกำหนดแอดเดรสบรรทัดถัดไปที่มีรูปแบบที่ระบุ

**302** AIX เวอร์ชัน 7.2: ข้อมูลอ้างอิงคำสั่งวอลุ่ม 2, d – h

เมื่อต้องการแอดเดรสบรรทัดถัดไปที่มีสตริงที่ตรงกัน ให้พิมพ์:

/Pattern/

โดยที่พารามิเตอร์ *Pattern* คือสตริงอักขระหรือ นิพจน์ทั่วไป การค้นหาเริ่มต้นที่บรรทัดหลังจากบรรทัดปัจจุบันและหยุดเมื่อพบรูปแบบที่ตรง ถ้าจำเป็น การค้นหาจะย้ายไปที่จุดสิ้นสุดบัพเฟอร์, ย้อนกลับไปจุดเริ่มต้นของบัพเฟอร์, และดำเนินการต่อจนพบข้อมูลที่ตรงกันหรือกลับไปบรรทัดปัจจุบัน ตัวอย่างเช่น:

/Austin, Texas/

แอดเดรสบรรทัดถัดไปที่มี Austin, Texas เป็น บรรทัดปัจจุบัน

การกำหนดแอดเดรสบรรทัดก่อนหน้านี้ที่มีรูปแบบที่ระบุ

เมื่อต้องการแอดเดรสบรรทัดก่อนหน้านี้ที่มีข้อมูลตรงกับ รูปแบบ ให้พิมพ์:

?Pattern?

โดยที่พารามิเตอร์ *Pattern* คือสตริงอักขระหรือ นิพจน์ทั่วไป *?Pattern?* คอนสตรัคชัน เช่น */Pattern/*, สามารถค้นหาบัพเฟอร์ทั้งหมด แต่ ค้นหาในทิศทางตรงกันข้าม ตัวอย่างเช่น:

?Austin, Texas?

แอดเดรสบรรทัดก่อนหน้านี้ที่มี Austin, Texas เป็น บรรทัดปัจจุบัน

การกำหนดแอดเดรสบรรทัดที่ทำเครื่องหมาย

เมื่อต้องการแอดเดรสบรรทัดที่ทำเครื่องหมายด้วยคำสั่งย่อย *k* ให้พิมพ์:

'x

โดยที่พารามิเตอร์ *x* คืออักขระตัวพิมพ์เล็ก *a* ถึง *z* ตัวอย่างเช่น:

'c

แอดเดรสบรรทัดที่ทำเครื่องหมาย เป็น *c* ด้วยคำสั่งย่อย *k*

## คำสั่งย่อย

ใช้คำสั่งย่อยเอดิเตอร์ *ed* เพื่อดำเนินการ ต่อไปนี้:

- แก้ไขไฟล์
- การจัดการไฟล์
- ดำเนินฟังก์ชันเบ็ดเตล็ด
  - เปลี่ยนพร้อมต์สตริง
  - ป้อนคำสั่งระบบ
  - ออกจากเอดิเตอร์ *ed*
  - ร้องขอวิธีใช้

ในกรณีส่วนใหญ่ คุณสามารถป้อนได้เพียงหนึ่งคำสั่งย่อยเอดิเตอร์ ed บนหนึ่งบรรทัด อย่างไรก็ตามคุณสามารถเพิ่มคำสั่งย่อย I (list) และ p (print) ให้กับคำสั่งย่อยยกเว้นคำสั่งย่อย e (edit), E (Edit), f (file), q (quit), Q (Quit), r (read), w (write) และ ! (คำสั่งระบบปฏิบัติการ)

คำสั่งย่อย e, f, r, และ w รับชื่อไฟล์เป็น พารามิเตอร์ เอดิเตอร์ ed เก็บชื่อไฟล์สุดท้ายที่ใช้กับคำสั่งย่อย เป็นชื่อไฟล์ดีฟอลต์ คำสั่งย่อย e, E, f, r, หรือ w ถัดไปที่กำหนดโดยไม่มีชื่อไฟล์ใช้ชื่อไฟล์ดีฟอลต์

เอดิเตอร์ ed ตอบสนองกับเงื่อนไขข้อผิดพลาดด้วยหนึ่งในสองข้อความ: ? (เครื่องหมายคำถาม) หรือ ?File เมื่อเอดิเตอร์ ed ได้รับสัญญาณ อินเทอร์รัปต์ (ลำดับคีย์ Ctrl-C), จะแสดง ? และกลับไปโหมดคำสั่ง เมื่อเอดิเตอร์ ed อ่านไฟล์ จะละเว้นอักขระ ASCII null และอักขระทั้งหมดหลังจากอักขระขึ้นบรรทัดใหม่สุดท้าย

## การแก้ไขไฟล์

คุณสามารถใช้คำสั่งย่อยเอดิเตอร์ ed ในการดำเนินงาน ดังต่อไปนี้:

- เพิ่มข้อความ
- เปลี่ยนข้อความ
- คัดลอกข้อความ
- ลบข้อความ
- แสดงข้อความ
- รวมและแยกบรรทัด
- ทำการเปลี่ยนแปลงระดับโกลบอล
- ทำเครื่องหมายข้อความ
- ย้ายข้อความ
- บันทึกข้อความ
- ค้นหาข้อความ
- แทนที่ข้อความ
- ยกเลิกการเปลี่ยนแปลงข้อความ

หมายเหตุ: ในรายละเอียดต่อไปนี้ คำสั่งย่อยเอดิเตอร์ ed, ดีฟอลต์ แอดเดรสถูกแสดงในวงเล็บ อย่าพิมพ์วงเล็บ แอดเดรส. (จุด) อ้างอิงถึงบรรทัดปัจจุบัน. (จุด) ในตำแหน่งแรก ของบรรทัดที่วางถ้าเป็นกรณีอื่น คือสัญญาณกลับไปโหมดคำสั่ง

## การเพิ่มข้อความ

## ไอเท็ม

(.)a [l] [n] [p] Text.

### คำอธิบาย

คำสั่งย่อย a (append) เพิ่มข้อความให้กับบัพเฟอร์ หลัง บรรทัดที่แอดเดรส คำสั่งย่อย a เชื่อมบรรทัดปัจจุบันเป็นบรรทัดที่แทรกล่าสุด หรือถ้าไม่มีบรรทัด ถูกแทรก, ไปที่บรรทัดที่แอดเดรส 0 แอดเดรสเพิ่มข้อความให้กับจุดเริ่มต้นของบัพเฟอร์

พิมพ์ l (list), n (number) หรือ p (print) ตัวเลือกคำสั่งย่อยถ้าคุณต้องการแสดงข้อความเพิ่มเติม

พิมพ์ข้อความของคุณ กดคีย์ Enter ที่จุดสิ้นสุดของแต่ละบรรทัด ถ้าคุณไม่ กด Enter ที่ตำแหน่งสุดท้ายของแต่ละบรรทัด เอ็ดดิเตอร์ ed ย้ายเคอร์เซอร์ของคุณโดยอัตโนมัติไปที่บรรทัดต่อไป หลังจากคุณป้อนอักขระในบรรทัด เอ็ดดิเตอร์ ed ปฏิบัติต่อข้อมูลที่พิมพ์ก่อนกด Enter เป็นหนึ่งบรรทัด ไม่ว่าจะใช้ บรรทัดไปเท่าไรบนจอภาพ

(.)i [l] [n] [p] Text.

พิมพ์ . (จุด) ที่จุดเริ่มต้นของบรรทัดใหม่ หลังจากคุณได้พิมพ์ข้อความของคุณทั้งหมด

คำสั่งย่อย i (insert) แทรกข้อความ ก่อน บรรทัดที่แอดเดรสและเชื่อมบรรทัดปัจจุบันเป็นบรรทัดที่แทรก สุดท้าย ถ้าไม่มีการแทรกบรรทัด คำสั่งย่อย i เชื่อม บรรทัดปัจจุบันเป็นบรรทัดที่แอดเดรส คุณไม่สามารถใช้ 0 แอดเดรสสำหรับ คำสั่งย่อยนี้

พิมพ์ l (list), n (number) หรือ p (print) คำสั่งย่อย ที่เป็นทางเลือก ถ้าคุณต้องการแสดงข้อความที่แทรก

พิมพ์ข้อความของคุณ กดคีย์ Enter ที่จุดสิ้นสุดของแต่ละบรรทัด ถ้าคุณไม่ กด Enter ที่ตำแหน่งสุดท้ายของแต่ละบรรทัด เอ็ดดิเตอร์ ed ย้ายเคอร์เซอร์ ของคุณโดยอัตโนมัติไปที่บรรทัดต่อไป หลังจากคุณป้อนอักขระในบรรทัด เอ็ดดิเตอร์ ed ปฏิบัติต่อข้อมูลที่พิมพ์ก่อนกด Enter เป็นหนึ่งบรรทัด ไม่ว่าจะใช้ บรรทัดไปเท่าไรบนจอภาพ

พิมพ์ . (จุด) ที่จุดเริ่มต้นของบรรทัดใหม่ หลังจากคุณได้พิมพ์ข้อความของคุณทั้งหมด

หมายเหตุ: คำสั่งย่อย i ต่างจากคำสั่งย่อย a เพียงแค่ตำแหน่งการวางข้อความ

คุณสามารถใช้คำสั่งย่อยเอ็ดดิเตอร์ ed อื่นเพื่อเพิ่ม ข้อความในตำแหน่งที่ต่างกัน ใช้รูปแบบนำหน้าเพื่อดำเนินงานแก้ไข ต่อไปนี้:

- การเพิ่มข้อความหลังจากบรรทัดปัจจุบัน
- การเพิ่มข้อความก่อนบรรทัดปัจจุบัน
- การเพิ่มข้อความหลังจากบรรทัดที่แอดเดรส
- การเพิ่มข้อความก่อนบรรทัดที่แอดเดรส
- การเพิ่มข้อความหลังบรรทัดที่มีรูปแบบ การค้นหา
- การเพิ่มข้อความก่อนบรรทัดที่มีรูปแบบ การค้นหา
- การเพิ่มข้อความหลังบรรทัดที่ไม่มี รูปแบบการค้นหา
- การเพิ่มข้อความก่อนบรรทัดที่ไม่มี รูปแบบการค้นหา

### เมื่อต้องการเพิ่มข้อความหลังบรรทัด ปัจจุบัน

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
a[l][n][p]
```

โดยที่ l, n และ p เป็น คำสั่งย่อยทางเลือกที่แสดงข้อความที่เพิ่ม

2. พิมพ์ข้อความ และกด Enter
3. พิมพ์ . (จุด) และกด Enter อีกครั้งเพื่อกลับไปโหมดคำสั่ง

### เมื่อต้องการเพิ่มข้อความก่อนบรรทัด ปัจจุบัน

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
i[l][n][p]
```

โดยที่ *l*, *n* และ *p* เป็น คำสั่งย่อยทางเลือกที่แสดงข้อความที่เพิ่ม

2. พิมพ์ข้อความ และกด Enter
3. พิมพ์ . (จุด) และกด Enter อีกครั้งเพื่อกลับไปโหมดคำสั่ง

#### เมื่อต้องการเพิ่มข้อความหลังบรรทัดที่แอดเดรส

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
Addressa[l][n][p]
```

โดยที่ พารามิเตอร์ *Address* คือหมายเลขบรรทัดของบรรทัด ที่ข้อความที่แทรกควรติดตาม ตัวเลือกคำสั่งย่อย *l*, *n* และ *p* แสดงข้อความที่เพิ่ม

2. พิมพ์ข้อความ และกด Enter
3. พิมพ์ . (จุด) และกด Enter อีกครั้งเพื่อกลับไปโหมดคำสั่ง

#### เมื่อต้องการเพิ่มข้อความก่อนบรรทัดที่แอดเดรส

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
Addressi[l][n][p]
```

โดยที่ พารามิเตอร์ *Address* คือหมายเลขบรรทัดของบรรทัด ที่ข้อความที่แทรกควรนำหน้า ตัวเลือกคำสั่งย่อย *l*, *n* และ *p* แสดงข้อความที่เพิ่ม

2. พิมพ์ข้อความ และกด Enter
3. พิมพ์ . (จุด) และกด Enter อีกครั้งเพื่อกลับไปโหมดคำสั่ง

#### เมื่อต้องการเพิ่มข้อความหลังจากบรรทัดที่มีรูปแบบการค้นหา

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]g/Pattern/a[l][n][p]
```

โดยที่ *Address* เป็นพารามิเตอร์ทางเลือกที่ระบุช่วงของบรรทัด เพื่อค้นหาในรูปแบบที่ระบุในพารามิเตอร์ *Pattern* พารามิเตอร์ *Pattern* คือสตริงอักขระหรือ นิพจน์ทั่วไป ถ้าคุณเว้นพารามิเตอร์ *Address* เอดิเตอร์ ed จะค้นหาทั้งไฟล์เพื่อหาบรรทัดที่มีรูปแบบ ตัวเลือกคำสั่งย่อย *l*, *n* และ *p* แสดงข้อความที่เพิ่ม

2. พิมพ์ backslash:

```
\
```

3. พิมพ์ข้อความ เมื่อต้องการเริ่มบรรทัดใหม่ภายในข้อความที่เพิ่ม ให้พิมพ์ backslash:

```
\
```

และกด Enter ข้อความที่คุณพิมพ์จะถูกเพิ่มหลังจากทุกๆ บรรทัดที่มีรูปแบบที่ระบุในคำสั่ง

4. เมื่อต้องการกลับไปโหมดคำสั่ง ให้กด Enter

#### เมื่อต้องการเพิ่มข้อความก่อนบรรทัดที่มีรูปแบบการค้นหา

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]g/Pattern/i[l][n][p]
```

โดยที่ *Address* เป็นพารามิเตอร์ทางเลือกที่ระบุช่วงของบรรทัด เพื่อค้นหารูปแบบที่ระบุในพารามิเตอร์ *Pattern* พารามิเตอร์ *Pattern* คือสตริงอักขระหรือ นิพจน์ทั่วไป ถ้าคุณเว้นพารามิเตอร์ *Address* เอดิเตอร์ *ed* จะค้นหาทั้งไฟล์เพื่อหาบรรทัดที่มีรูปแบบ ตัวเลือกคำสั่งย่อย *l*, *n* และ *p* แสดง ข้อความที่เพิ่ม

2. พิมพ์ backslash:

\

3. พิมพ์ข้อความ เมื่อต้องการเริ่มบรรทัดใหม่ภายในข้อความที่เพิ่ม ให้พิมพ์ backslash:

\

และกด Enter ข้อความที่คุณพิมพ์จะถูกเพิ่มก่อนทุกๆ บรรทัดที่มีรูปแบบที่ระบุในคำสั่ง

4. เมื่อต้องการกลับไปทีเอนด์คำสั่ง ให้กด Enter

เมื่อต้องการเพิ่มข้อความหลังจากบรรทัดที่ไม่มีรูปแบบการค้นหา

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]g/Pattern/a[1][n][p]
```

โดยที่ *Address* เป็นพารามิเตอร์ทางเลือกที่ระบุช่วงของบรรทัด เพื่อค้นหาบรรทัดที่ไม่มีรูปแบบที่ระบุในพารามิเตอร์ *Pattern* พารามิเตอร์ *Pattern* คือสตริงอักขระหรือ นิพจน์ทั่วไป ถ้าคุณเว้น *Address* เอดิเตอร์ *ed* จะค้นหาทั้งไฟล์เพื่อหาบรรทัดที่ไม่มีรูปแบบนั้น ตัวเลือกคำสั่งย่อย *l*, *n* และ *p* แสดง ข้อความที่เพิ่ม

2. พิมพ์ backslash:

\

3. พิมพ์ข้อความ เมื่อต้องการเริ่มบรรทัดใหม่ภายในข้อความที่เพิ่ม ให้พิมพ์ backslash:

\

และกด Enter ข้อความที่คุณพิมพ์จะถูกเพิ่มหลังจากทุกๆ บรรทัดที่ไม่มีรูปแบบที่ระบุในคำสั่ง

4. เมื่อต้องการกลับไปทีเอนด์คำสั่ง ให้กด Enter

เมื่อต้องการเพิ่มข้อความก่อนบรรทัดที่ไม่มีรูปแบบการค้นหา

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]g/Pattern/i[1][n][p]
```

โดยที่ *Address* เป็นพารามิเตอร์ทางเลือกที่ระบุช่วงของบรรทัด เพื่อค้นหาบรรทัดที่ไม่มีรูปแบบที่ระบุในพารามิเตอร์ *Pattern* พารามิเตอร์ *Pattern* คือสตริงอักขระหรือ นิพจน์ทั่วไป ถ้าคุณเว้นพารามิเตอร์ *Address* เอดิเตอร์ *ed* จะค้นหาทั้งไฟล์เพื่อหา บรรทัดที่มีรูปแบบ ตัวเลือกคำสั่งย่อย *l*, *n* และ *p* แสดง ข้อความที่เพิ่ม

2. พิมพ์ backslash:

\

3. พิมพ์ข้อความ เมื่อต้องการเริ่มบรรทัดใหม่ภายในข้อความที่เพิ่ม ให้พิมพ์ backslash:

\

และกด Enter ข้อความที่คุณพิมพ์จะถูกเพิ่มก่อนทุกๆ บรรทัดที่ไม่มีรูปแบบที่ระบุในคำสั่ง

4. เมื่อต้องการกลับไปทีเอนด์คำสั่ง ให้กด Enter

## การเปลี่ยนข้อความ

### ไวยากรณ์

(.,.)c[l][n][p]Text.

### คำอธิบาย

คำสั่งย่อย c (change) ลบบรรทัดที่แอดเดรส ที่คุณต้องการแทนที่แล้วแทนที่ด้วยบรรทัดใหม่ที่คูป้อน คำสั่งย่อย c เช็ดบรรทัดปัจจุบันเป็นบรรทัดใหม่ล่าสุดของอินพุต หรือถ้าไม่มีอินพุต จะเป็นบรรทัดแรกที่ไม่ถูกลบ

พิมพ์ l (list), n (number) หรือ p (print) คำสั่งย่อย ที่เป็นทางเลือก ถ้าคุณต้องการแสดงข้อความที่แทรก

พิมพ์ข้อความใหม่ แล้วกด Enter ที่จุดสิ้นสุดของแต่ละบรรทัด เมื่อคุณได้ป้อนข้อความใหม่ทั้งหมดให้พิมพ์ . (จุด) บนตัวบรรทัดเอง

คุณสามารถเปลี่ยนข้อความได้หลายวิธีด้วย เอดิเตอร์ ed ใช้รูปแบบนำหน้าเพื่อดำเนินงานแก้ไข ต่อไปนี้:

- การเปลี่ยนข้อความของบรรทัดปัจจุบัน
- การเปลี่ยนข้อความของบรรทัดหรือกลุ่มของบรรทัด
- การเปลี่ยนข้อความของบรรทัดที่มีรูปแบบ ที่ระบุ
- การเปลี่ยนข้อความของบรรทัดที่ไม่มีรูปแบบ ที่ระบุ

### เมื่อต้องการเปลี่ยนข้อความของบรรทัด ปัจจุบัน

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
c[l][n][p]
```

โดยที่ l, n และ p เป็น คำสั่งย่อยทางเลือกที่แสดงข้อความที่เปลี่ยนแปลง

2. พิมพ์ข้อความ และกด Enter
3. พิมพ์ . (จุด) และกด Enter อีกครั้งเพื่อกลับไปโหมดคำสั่ง

### เมื่อต้องการเปลี่ยนข้อความของบรรทัดหรือกลุ่มบรรทัด

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
Addressc[l][n][p]
```

โดยที่พารามิเตอร์ Address คือแอดเดรสของบรรทัดหรือ กลุ่มบรรทัดที่ต้องการเปลี่ยนแปลง คำสั่งย่อยทางเลือก l, n และ p แสดงข้อความที่เปลี่ยนแปลง

2. พิมพ์ข้อความ และกด Enter
3. พิมพ์ . (จุด) และกด Enter อีกครั้งเพื่อกลับไปโหมดคำสั่ง

### เมื่อต้องการเปลี่ยนข้อความของบรรทัดที่มีรูปแบบที่ระบุ

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
Addressg/Pattern/c[l][n][p]
```

โดยที่พารามิเตอร์ Address เป็นแอดเดรส ของกลุ่มบรรทัดที่คุณต้องการค้นหาสำหรับรูปแบบที่ระบุด้วย พารามิเตอร์ Pattern คำสั่งย่อยทางเลือก l, n และ p แสดงข้อความที่เปลี่ยนแปลง

2. พิมพ์ backslash:

```
\
```

3. พิมพ์ข้อความใหม่ เมื่อต้องการเริ่มบรรทัดใหม่ภายในข้อความใหม่ ให้พิมพ์ backslash:

\

และกด Enter

4. เมื่อต้องการกลับไปโหมดคำสั่ง กด Enter อีกครั้ง พิมพ์ . (จุด) และกด Enter อีกครั้ง

เมื่อต้องการเปลี่ยนข้อความของบรรทัดที่ไม่มีรูปแบบที่ระบุ

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
Addressv/Pattern/c[1][n][p]
```

โดยที่พารามิเตอร์ *Address* เป็นแอดเดรสของกลุ่มบรรทัดที่คุณต้องการค้นหาสำหรับรูปแบบที่ระบุด้วย พารามิเตอร์ *Pattern* คำสั่งย่อยทางเลือก *l*, *n* และ *p* แสดงข้อความที่เปลี่ยนแปลง

2. พิมพ์ backslash:

\

3. พิมพ์ข้อความใหม่ เมื่อต้องการเริ่มบรรทัดใหม่ภายในข้อความใหม่ ให้พิมพ์ backslash:

\

และกด Enter

4. เมื่อต้องการกลับไปโหมดคำสั่ง กด Enter อีกครั้ง พิมพ์ . (จุด) และกด Enter อีกครั้ง

**การคัดลอกข้อความ**

ไอเท็ม

```
(.,.)tAddress [p] [l] [n]
```

คำอธิบาย

คำสั่งย่อย *t* (transfer) แทรกลำเนาของ บรรทัดที่แอดเดรสหลังบรรทัดที่ระบุโดยพารามิเตอร์ *Address* คำสั่งย่อย *t* ยอมรับ 0 แอดเดรสเพื่อแทรกบรรทัดที่จุดเริ่มต้นของบัฟเฟอร์

คำสั่งย่อย *t* เชื่อบรรทัดปัจจุบันเป็นบรรทัดสุดท้าย ที่คัดลอก

พิมพ์ *l* (list), *n* (number) หรือ *p* (print) คำสั่งย่อย ที่เป็นทางเลือก หากคุณต้องการแสดงข้อความที่ถ่ายโอน

การคัดลอกบรรทัดหรือชุดของบรรทัดคงบรรทัด ที่ระบุไว้ที่ตำแหน่งดั้งเดิม และนำสำเนาไปไว้ในตำแหน่งใหม่ คุณสามารถเลือกบรรทัดที่จะคัดลอกโดยระบุแอดเดรสหรือรูปแบบ ใช้รูปแบบนำหน้าเพื่อดำเนินงานแก้ไข ต่อไปนี้:

- การคัดลอกบรรทัดปัจจุบัน
- การคัดลอกบรรทัดที่ระบุโดยแอดเดรส
- การคัดลอกบรรทัดที่มีรูปแบบที่ระบุ
- การคัดลอกบรรทัดที่ไม่มีรูปแบบที่ระบุ

เมื่อต้องการคัดลอกบรรทัด ปัจจุบัน

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
tAddress[1][n][p]
```

โดยที่ *Address* เป็นหมายเลขบรรทัด หรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของบรรทัดที่คุณต้องการให้สำเนาของบรรทัดปัจจุบันต่อท้าย คำสั่งย่อยทางเลือก *l*, *n* และ *p* แสดงบรรทัดที่คัดลอก

2. พิมพ์ข้อความ และกด Enter

- พิมพ์ (จุด) และกด Enter อีกครั้งเพื่อกลับไปโหมดคำสั่ง

### เมื่อต้องการคัดลอกบรรทัดที่ระบุโดยแอดเดรส

- พิมพ์คำสั่งย่อยต่อไปนี้:

```
LineNumbertDestinationAddress[l][n][p]
```

โดยที่พารามิเตอร์ *LineNumber* คือแอดเดรสของบรรทัดที่คุณต้องการคัดลอก และพารามิเตอร์ *DestinationAddress* คือบรรทัดที่คุณต้องการให้สำเนาต่อท้าย คำสั่งย่อยทางเลือก l, n และ p แสดงบรรทัดที่คัดลอก

- พิมพ์ข้อความ และกด Enter
- พิมพ์ (จุด) และกด Enter อีกครั้งเพื่อกลับไปโหมดคำสั่ง

### เมื่อต้องการคัดลอกบรรทัดที่มีรูปแบบที่ระบุ

พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]g/Pattern/t[DestinationAddress][l][n][p]
```

โดยที่ *Address* เป็นพารามิเตอร์ ทางเลือกที่ระบุช่วงของบรรทัดเพื่อค้นหาบรรทัดที่มีรูปแบบที่ระบุ พารามิเตอร์ *Pattern* คือข้อความที่คุณกำลังค้นหา และ *DestinationAddress* เป็นพารามิเตอร์ ทางเลือกที่ระบุบรรทัดที่คุณต้องการให้สำเนา ต่อท้าย คำสั่งย่อยทางเลือก l, n และ p แสดงบรรทัดที่คัดลอก

ถ้าคุณเว้นพารามิเตอร์ *Address* เดีเตอร์ ed จะค้นหาทั้งไฟล์เพื่อหาบรรทัดที่มีรูปแบบ ถ้าคุณเว้นพารามิเตอร์ *DestinationAddress* ข้อความที่ คัดลอกจะถูกวางไว้หลังจากบรรทัดปัจจุบัน

### เมื่อต้องการคัดลอกบรรทัดที่ไม่มีรูปแบบที่ระบุ

พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]v/Pattern/t[DestinationAddress][l][n][p]
```

โดยที่ *Address* เป็นพารามิเตอร์ ทางเลือกที่ระบุช่วงของบรรทัดเพื่อค้นหาบรรทัดที่ไม่มี รูปแบบที่ระบุ พารามิเตอร์ *Pattern* คือข้อความ และ *DestinationAddress* เป็นพารามิเตอร์ ทางเลือกที่ระบุบรรทัดที่คุณต้องการให้สำเนาต่อท้าย คำสั่งย่อยทางเลือก l, n และ p แสดงบรรทัดที่คัดลอก

ถ้าคุณเว้นพารามิเตอร์ *Address* เดีเตอร์ ed จะค้นหาทั้งไฟล์เพื่อหา บรรทัดที่มีรูปแบบ ถ้าคุณเว้นพารามิเตอร์ *DestinationAddress* ข้อความที่ คัดลอกจะถูกวางไว้หลังจากบรรทัดปัจจุบัน

### การลบข้อความ

ไอเท็ม

(.,.)d [l] [n] [p]

คำอธิบาย

คำสั่งย่อย d (delete) ลบบรรทัดที่แอดเดรส ออกจากบัฟเฟอร์ บรรทัดหลังจากบรรทัดสุดท้ายที่ลบกลายเป็น บรรทัด ปัจจุบัน ถ้าบรรทัดที่ลบในตอนแรกอยู่ที่จุดเริ่มต้นของบัฟเฟอร์ บรรทัดสุดท้ายใหม่ จะกลายเป็น บรรทัดปัจจุบัน

พิมพ์ l (list), n (number) หรือ p (print) คำสั่งย่อยที่เป็นทางเลือก ถ้าคุณต้องการแสดงการลบ

เอดิเตอร์ ed จัดเตรียมหลายวิธีเพื่อลบข้อความ ใช้รูปแบบนำหน้าเพื่อดำเนินงานแก้ไข ต่อไปนี้:

- การลบบรรทัดปัจจุบัน
- การลบบรรทัดหรือกลุ่มของบรรทัด
- การลบบรรทัดหรือกลุ่มของบรรทัดที่มีรูปแบบที่ระบุ
- การลบบรรทัดหรือกลุ่มของบรรทัดที่ไม่มีรูปแบบที่ระบุ
- การลบข้อความจากบรรทัดปัจจุบัน
- การลบข้อความภายในบรรทัดที่เลือก
- การลบข้อความจากบรรทัดที่แอดเดรส
- การลบข้อความจากบรรทัดที่มีรูปแบบ ที่ระบุ
- การลบรูปแบบจากบรรทัดที่มีรูปแบบ ที่ระบุต่างไป
- การลบรูปแบบจากบรรทัดที่ไม่มีรูปแบบ ที่ระบุต่างไป

เมื่อต้องการลบบรรทัด ปัจจุบัน

พิมพ์คำสั่งย่อยต่อไปนี้:

```
d[1][n][p]
```

โดยที่ l, n และ p เป็น คำสั่งย่อยทางเลือกที่แสดงบรรทัดที่ลบ

เมื่อต้องการลบบรรทัดหรือกลุ่มของบรรทัด

พิมพ์คำสั่งย่อยต่อไปนี้:

```
Addressd[1][n][p]
```

โดยที่พารามิเตอร์ *Address* คือหมายเลขบรรทัด หรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของบรรทัดที่คุณต้องการลบ และ l, n, และ p เป็นคำสั่งย่อย ทางเลือกที่แสดงบรรทัดที่ลบ

เมื่อต้องการลบบรรทัดหรือกลุ่มบรรทัดที่มีรูปแบบที่ระบุ

พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]g/Pattern/d[1][n][p]
```

โดยที่ *Address* เป็นพารามิเตอร์ทางเลือกที่ ระบุหมายเลขบรรทัดหรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของบรรทัดที่คุณ ต้องการค้นหา และพารามิเตอร์ *Pattern* เป็นสตริงอักขระหรือ นิพจน์ทั่วไป ที่แสดงข้อความที่คุณ ต้องการค้นหา ถ้าคุณเว้น พารามิเตอร์ *Address* เอดิเตอร์ ed จะค้นหาทั้งไฟล์เพื่อหา บรรทัดที่มีรูปแบบที่ระบุ คำสั่งย่อยทางเลือก l, n และ p แสดง บรรทัดที่ลบ

## เมื่อต้องการลบบรรทัดหรือกลุ่มบรรทัดที่ไม่มีรูปแบบที่ระบุ

พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]v/Pattern/d[1][n][p]
```

โดยที่ *Address* เป็นพารามิเตอร์ทางเลือกที่ ระบุหมายเลขบรรทัดหรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของบรรทัดที่คุณต้องการค้นหา และพารามิเตอร์ *Pattern* เป็นสตริงอักขระหรือ นิพจน์ทั่วไป ที่แสดงข้อความที่คุณ ต้องการค้นหา ถ้าคุณเว้นพารามิเตอร์ *Address* เอดีเตอร์ *ed* จะค้นหาทั้งไฟล์เพื่อหา บรรทัดที่ไม่มีรูปแบบที่ระบุ คำสั่งย่อยทางเลือก *l*, *n* และ *p* แสดงบรรทัดที่ลบ

## เมื่อต้องการลบข้อความออกจาก บรรทัด ปัจจุบัน

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
s/Pattern
```

โดยที่ พารามิเตอร์ *Pattern* คือสตริงอักขระหรือ นิพจน์ทั่วไป ที่แสดงข้อความที่คุณ ต้องการลบ

2. เมื่อต้องการลบอินสแตนซ์แรกของรูปแบบจาก บรรทัด ให้พิมพ์:

```
//
```

หรือ

เพื่อลบ *ทุกอินสแตนซ์* ของรูปแบบจากบรรทัด ให้พิมพ์:

```
//g
```

3. ถ้าคุณต้องการแสดงการลบ ให้พิมพ์หนึ่งในคำสั่งย่อยทางเลือก ต่อไปนี้:

```
l
```

```
n
```

```
p
```

4. กด Enter

## เมื่อต้องการลบข้อความภายในบรรทัดที่เลือก

1. พิมพ์แอดเดรสของกลุ่มบรรทัดเพื่อเลือก (หรือข้ามขั้นตอนนี้เพื่อ เลือกบรรทัดทั้งหมด)

2. เมื่อต้องการเลือกบรรทัดที่ระบุโดยพารามิเตอร์ *Pattern* ในขั้นตอนที่ 4 ให้พิมพ์:

```
g
```

หรือ

เมื่อต้องการเลือกบรรทัดที่ *ไม่* ระบุโดยพารามิเตอร์ *Pattern* ในขั้นตอนที่ 4 พิมพ์:

```
v
```

3. เมื่อต้องการป้อนข้อความที่คุณต้องการค้นหา ให้พิมพ์คำสั่งย่อยต่อไปนี้:

```
/Pattern/s
```

โดยที่พารามิเตอร์ *Pattern* เป็นข้อความที่คุณต้องการค้นหา

4. พิมพ์หนึ่งในคำสั่งต่อไปนี้เพื่อทำการลบที่ต้องการ:

เมื่อต้องการลบอินสแตนซ์แรกของพารามิเตอร์ *Pattern* ภายในแต่ละบรรทัดที่เลือกให้พิมพ์:

```
///
```

เมื่อต้องการลบทุกอินสแตนซ์ของพารามิเตอร์ *Pattern* ภายในแต่ละบรรทัดที่เลือกให้พิมพ์:

```
///g
```

เมื่อต้องการลบจำนวนที่พบของพารามิเตอร์ *Pattern* ที่ระบุครั้งแรกบนแต่ละบรรทัดที่เลือก (โดยที่พารามิเตอร์ *Number* เป็นจำนวนเต็ม) ให้พิมพ์:

```
///Number
```

เมื่อต้องการลบสตริงอักขระแรกที่ระบุโดยพารามิเตอร์ *OtherPattern* ภายในแต่ละบรรทัดที่เลือกโดยพารามิเตอร์ *Pattern* (โดยที่พารามิเตอร์ *OtherPattern* คือรูปแบบที่คุณต้องการค้นหา) ให้พิมพ์:

```
/OtherPattern//
```

เมื่อต้องการลบทุกอินสแตนซ์ของพารามิเตอร์ *OtherPattern* ภายในแต่ละบรรทัดที่เลือกโดยพารามิเตอร์ *Pattern* ให้พิมพ์:

```
/OtherPattern//g
```

เมื่อต้องการลบจำนวนครั้งที่พบของพารามิเตอร์ *OtherPattern* ที่ระบุครั้งแรกบนแต่ละบรรทัดที่เลือกโดยพารามิเตอร์ *Pattern* (โดยที่พารามิเตอร์ *Number* เป็นจำนวนเต็ม) ให้พิมพ์:

```
/OtherPattern//Number
```

5. ถ้าคุณต้องการแสดงการลบ ให้พิมพ์หนึ่งในคำสั่งย่อยทางเลือกต่อไปนี้:

```
l
```

```
n
```

```
p
```

6. กด Enter

ตัวอย่าง เมื่อต้องการลบอินสแตนซ์ของรูปแบบทั้งหมดจาก ช่วงบรรทัด ให้พิมพ์:

```
38,$g/tmp/s/gn
```

ตัวอย่างก่อนหน้านี้นั้นค้นหาบรรทัดทั้งหมดจากบรรทัด 38 ไปถึงบรรทัดสุดท้าย (38,\$) สำหรับสตริงอักขระ tmp และลบทุกอินสแตนซ์ (/g) ของสตริงอักขระนั้นภายในบรรทัดดังกล่าว จากนั้นจะแสดง บรรทัดที่มีข้อความที่ลบออกไปและหมายเลขบรรทัด (n)

เมื่อต้องการลบทุกอินสแตนซ์ของรูปแบบจาก บรรทัดทั้งหมด ที่มีรูปแบบนั้น พิมพ์:

```
g/rem/s///g|
```

ตัวอย่างก่อนหน้าค้นหาทั้งไฟล์ (พารามิเตอร์ address ถูกเว้น) สำหรับบรรทัดทั้งหมดที่มีสตริงอักขระ (g) rem โดยลบ อินสแตนซ์ (///g) ของสตริงอักขระ rem ทั้งหมดจากแต่ละบรรทัดแล้วแสดงบรรทัด ที่ได้ลบข้อความออกไป รวมถึงอักขระที่ไม่พิมพ์ออกใน บรรทัดเหล่านั้น (1)

#### เมื่อต้องการลบข้อความจากบรรทัดที่แอดเดรส

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
Addresss/Pattern
```

หมายเหตุ: พารามิเตอร์ Address ถูกติดตามด้วย คำสั่งย่อย s โดยที่พารามิเตอร์ Address คือหมายเลขบรรทัด, ช่องของหมายเลขบรรทัด, หรือการกำหนดแอดเดรสเชิงสัญลักษณ์ ของบรรทัดซึ่งคุณต้องการลบรูปแบบออก และพารามิเตอร์ Pattern คือสตริงอักขระหรือ นิพจน์ทั่วไป ที่แสดงข้อความที่คุณต้องการลบ

2. เมื่อต้องการลบอินสแตนซ์แรกของรูปแบบจาก แต่ละบรรทัด ให้พิมพ์:

```
//
```

หรือ

เพื่อลบ ทุกอินสแตนซ์ของรูปแบบจากแต่ละบรรทัด ให้พิมพ์:

```
//g
```

3. ถ้าคุณต้องการแสดงการลบ ให้พิมพ์หนึ่งในคำสั่งย่อยทางเลือก ต่อไปนี้:

```
|
```

```
n
```

```
p
```

4. กด Enter

#### เมื่อต้องการลบข้อความจากบรรทัดที่มีรูปแบบที่ระบุ

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]g/Pattern/s
```

โดยที่ Address เป็นพารามิเตอร์ทางเลือกที่ ระบุหมายเลขบรรทัด, ช่องของหมายเลขบรรทัด หรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของ บรรทัดที่มีรูปแบบที่ระบุ และพารามิเตอร์ Pattern คือสตริงอักขระหรือ นิพจน์ ทั่วไป ที่แสดงข้อความที่คุณต้องการค้นหาและลบ ถ้าคุณ เว้นพารามิเตอร์ Address เอดิเตอร์ ed ค้นหา บรรทัดทั้งหมดในไฟล์เพื่อหารูปแบบ

2. เมื่อต้องการลบอินสแตนซ์แรกของรูปแบบจาก แต่ละบรรทัด ให้พิมพ์:

```
///
```

หรือ

เมื่อต้องการลบ ทุกอินสแตนซ์ของรูปแบบจาก แต่ละบรรทัด ให้พิมพ์:

```
///g
```

3. ถ้าคุณต้องการแสดงการลบ ให้พิมพ์หนึ่งในคำสั่งย่อยทางเลือก ต่อไปนี้:

|

n

p

4. กด Enter

เมื่อต้องการลบรูปแบบจากบรรทัดที่มีรูปแบบที่ระบุต่างกัน

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]g/SearchPattern/s
```

โดยที่ *Address* เป็นพารามิเตอร์ทางเลือกที่ ระบุหมายเลขบรรทัด, ช่วงของหมายเลขบรรทัด หรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของ บรรทัดที่มีรูปแบบที่ระบุ และพารามิเตอร์ *SearchPattern* คือสตริงอักขระหรือ นิพจน์ ทั่วไป ที่แสดงข้อความที่อยู่ในบรรทัดที่คุณต้องการเปลี่ยนแปลง ถ้าคุณเว้นพารามิเตอร์ *Address* เอดิเตอร์ ed ค้นหา บรรทัดทั้งหมดในไฟล์เพื่อหารูปแบบที่ระบุ

2. เมื่อต้องการระบุข้อความที่คุณต้องการลบ ให้พิมพ์:

```
/DeletePattern/
```

3. เมื่อต้องการลบอินสแตนซ์แรกของรูปแบบจาก แต่ละบรรทัด ให้พิมพ์:

```
/
```

หรือ

เพื่อลบ *ทุกอินสแตนซ์*ของรูปแบบจากแต่ละบรรทัด ให้พิมพ์:

```
/g
```

หมายเหตุ: สตริงคำสั่งย่อยทั้งหมด มีลักษณะดังนี้:

```
[Address]g/SearchPattern/s/DeletePattern//[g]
```

4. ถ้าคุณต้องการแสดงการลบ ให้พิมพ์หนึ่งในคำสั่งย่อยทางเลือก ต่อไปนี้:

|

n

p

5. กด Enter

ตัวอย่าง เมื่อต้องการลบอินสแตนซ์แรกของรูปแบบ จากบรรทัดที่มีรูปแบบที่ระบุต่างกัน ให้พิมพ์:

```
1,.g/rem/s/tmp//1
```

ตัวอย่างก่อนหน้านี้นค้นหาจากบรรทัดแรกถึง บรรทัดปัจจุบัน (1, .) สำหรับบรรทัดทั้งหมดที่มี (g) สตริงอักขระ rem โดยลบอินสแตนซ์แรกของสตริงอักขระ tmp จากแต่ละบรรทัด (/), แล้วแสดง บรรทัดที่มีข้อความที่ลบไป รวมถึงอักขระที่ไม่พิมพ์ออกใน บรรทัดเหล่านั้น (1)

## เมื่อต้องการลบรูปแบบจากบรรทัดที่ไม่มีรูปแบบที่ระบุ ต่างกัน

### 1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]v/SearchPattern/s
```

โดยที่ *Address* เป็นพารามิเตอร์ทางเลือกที่ ระบุหมายเลขบรรทัด, ช่วงของหมายเลขบรรทัด หรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของ บรรทัดที่มีรูปแบบที่ระบุ และพารามิเตอร์ *SearchPattern* คือสตริงอักขระหรือ นิพจน์ ทั่วไป ที่แสดงข้อความที่อยู่ในบรรทัดที่คุณต้องการค้นหาและเปลี่ยนแปลง ถ้าคุณเว้นพารามิเตอร์ *Address* เอดิเตอร์ ed ค้นหา บรรทัดทั้งหมดในไฟล์เพื่อหารูปแบบที่ระบุ

### 2. เมื่อต้องการระบุข้อความที่คุณต้องการลบให้พิมพ์:

```
/DeletePattern/
```

### 3. เมื่อต้องการลบอินสแตนซ์แรกของรูปแบบให้พิมพ์:

```
/
```

หรือ

เพื่อลบ **ทุกอินสแตนซ์** ของรูปแบบจากแต่ละบรรทัด ให้พิมพ์:

```
/g
```

**หมายเหตุ:** สตริงคำสั่งย่อยทั้งหมด มีลักษณะดังนี้:

```
[Address]v/SearchPattern/s/DeletePattern//[g]
```

### 4. ถ้าคุณต้องการแสดงการลบให้พิมพ์หนึ่งในคำสั่งย่อยทางเลือก ต่อไปนี้:

```
l
```

```
n
```

```
p
```

### 5. กด Enter

ตัวอย่าง เมื่อต้องการลบอินสแตนซ์แรกของรูปแบบ จากบรรทัดที่ไม่มีรูปแบบที่ระบุให้พิมพ์:

```
1,.v/rem/s/tmp//l
```

ตัวอย่างก่อนหน้านี้ค้นหาจากบรรทัดแรกถึง บรรทัดปัจจุบัน (1, .) สำหรับบรรทัดทั้งหมดที่ไม่มี (v) สตริงอักขระ rem โดยลบอินสแตนซ์แรกของสตริงอักขระ tmp จากแต่ละบรรทัด (/), แล้วแสดง บรรทัดที่มีข้อความที่ลบไป รวมถึงอักขระที่ไม่พิมพ์ออกใน บรรทัดเหล่านั้น (l)

## การแสดงข้อความ

ไอเท็ม (.,.)l	คำอธิบาย คำสั่งย่อย l (list) เขียนบรรทัดที่แอดเดรส ไปที่เอ็ดฟุตมาตรฐานในรูปแบบที่เห็นได้ชัดและเขียนอักขระ \\, \\a, \\b, \\f, \\r, \\t และ \\v ในลำดับ escape ที่ตรงกัน คำสั่งย่อย เขียน อักขระที่ไม่พิมพ์ออกเป็นหนึ่งตัวเลขฐานแปด 3-หลัก, พร้อมกับ \ (backslash) นำหน้าสำหรับแต่ละไบต์ในอักขระ (ไบต์ที่สำคัญที่สุดก่อน)
(.,.)n	คำสั่งย่อย n ย้อนผ่านบรรทัดยาว และคุณสามารถระบุจุดย้อนกลับโดยเขียนลำดับอักขระ \ (backslash)/new-line การย้อนกลับเกิดขึ้นที่ตำแหน่งคอลัมน์ที่ 72 \$ (เครื่องหมายดอลลาร์) กำหนดเป็น จุดสิ้นสุดของแต่ละบรรทัด คุณสามารถผนวกคำสั่งย่อย l กับ คำสั่งย่อยเอ็ดเตอร์ ed ใดๆ ยกเว้น e, E, f, q, Q, r, w หรือ ! หมายเลขบรรทัดปัจจุบันถูกเซตเป็นแอดเดรสของบรรทัดสุดท้ายที่เขียน คำสั่งย่อย n (number) แสดงบรรทัดที่ แอดเดรส แต่ละบรรทัดนำหน้าโดยหมายเลขบรรทัดและอักขระแท็บ (ที่แสดงเป็น พื้นที่ว่าง); n เซตบรรทัดปัจจุบันเป็นบรรทัดปัจจุบัน ที่แสดง คุณสามารถผนวกคำสั่งย่อย n กับคำสั่งย่อยเอ็ดเตอร์ ed ใดๆ ยกเว้น e, f, r หรือ w ตัวอย่าง คำสั่งย่อย dn ลบบรรทัดปัจจุบัน และแสดงบรรทัดปัจจุบันใหม่และหมายเลขบรรทัด
(.,.)p	คำสั่งย่อย p (print) แสดงบรรทัดที่แอดเดรส และเซตบรรทัดปัจจุบันเป็นบรรทัดสุดท้ายที่แสดง คุณสามารถผนวก คำสั่งย่อย p ท้ายคำสั่งย่อยของเอ็ดเตอร์ ed ใดๆ ยกเว้น e, f, r หรือ w ตัวอย่างเช่น คำสั่งย่อย dp ลบบรรทัดปัจจุบันและแสดงบรรทัดปัจจุบันใหม่
(.)=	โดยไม่มีแอดเดรสคำสั่งย่อย = (เครื่องหมายเท่ากับ) แสดงหมายเลขบรรทัดปัจจุบัน เมื่อนำหน้าด้วย \$ แอดเดรส คำสั่งย่อย = แสดงจำนวนของบรรทัดสุดท้ายใน บัฟเฟอร์ คำสั่งย่อย = ไม่เปลี่ยนบรรทัดปัจจุบัน และไม่สามารถถูกผนวกกับคำสั่งย่อย g หรือคำสั่งย่อย v

เมื่อคุณค้นหาบรรทัดที่มีหรือไม่มี รูปแบบที่ระบุ คุณสามารถเลือกช่วงของหมายเลขบรรทัดที่จะค้นหา คุณสามารถ เลือกและแสดงหนึ่งบรรทัดหรือหนึ่งกลุ่มของบรรทัดในไฟล์เอ็ดเตอร์ ed ได้หลายวิธีต่างกัน ใช้รูปแบบนำหน้าเพื่อดำเนินงานแก้ไข ต่อไปนี้:

- การแสดงบรรทัดหรือกลุ่มของบรรทัดที่แอดเดรส
- การแสดงบรรทัดหรือกลุ่มของบรรทัดที่แอดเดรส และอักขระที่ไม่พิมพ์ออก
- การแสดงบรรทัดหรือกลุ่มของบรรทัดที่แอดเดรส และหมายเลขบรรทัด
- การแสดงบรรทัดที่มีรูปแบบการค้นหา
- การแสดงบรรทัดที่มีรูปแบบการค้นหา รวมทั้งอักขระที่ไม่พิมพ์ออก
- การแสดงบรรทัดที่มีรูปแบบการค้นหา รวมทั้งหมายเลขบรรทัด
- การแสดงบรรทัดที่ไม่มีรูปแบบการค้นหา
- การแสดงบรรทัดที่ไม่มีรูปแบบการค้นหา รวมทั้งอักขระที่ไม่พิมพ์ออก
- การแสดงบรรทัดที่ไม่มีรูปแบบการค้นหา รวมทั้งหมายเลขบรรทัด

### เมื่อต้องการแสดงบรรทัดหรือกลุ่มของบรรทัดที่แอดเดรส

พิมพ์คำสั่งย่อยต่อไปนี้:

Addressp

โดยที่พารามิเตอร์ Address คือหมายเลขบรรทัด หรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของบรรทัดที่คุณต้องการแสดง

บรรทัดที่แอดเดรสถูกแสดงบนจอภาพ ถ้ากลุ่มของบรรทัดยาวไม่พอดีบนจอภาพ เอ็ดเตอร์ ed จะแสดง เท่าที่แสดงได้พอดี เริ่มต้นด้วยบรรทัดแรกที่แอดเดรส

### เมื่อต้องการแสดงบรรทัดหรือกลุ่มของบรรทัดที่แอดเดรสและอักขระที่ไม่พิมพ์ออก

พิมพ์คำสั่งย่อยต่อไปนี้:

Addressl

โดยที่พารามิเตอร์ Address คือหมายเลขบรรทัด หรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของบรรทัดที่คุณต้องการแสดง

บรรทัดที่แอดเดรสและอักขระที่ไม่พิมพ์ออก ถูกแสดงบนจอภาพ ถ้ากลุ่มของบรรทัดยาวไม่พอดีบนจอภาพ เอดิเตอร์ ed จะแสดง เท่าที่แสดงได้พอดี เริ่มต้นด้วยบรรทัดแรกที่แอดเดรส

**เมื่อต้องการแสดงบรรทัดหรือกลุ่มของบรรทัดที่แอดเดรสและหมายเลขบรรทัด**

พิมพ์คำสั่งย่อยต่อไปนี้:

Addressn

โดยที่พารามิเตอร์ *Address* คือหมายเลขบรรทัด หรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของบรรทัดที่คุณต้องการแสดง

บรรทัดที่แอดเดรสถูกแสดงบนจอภาพ หมายเลขบรรทัดสำหรับแต่ละบรรทัดถูกแสดงข้างบรรทัด ถ้ากลุ่มของบรรทัดยาวไม่พอดีบนจอภาพ เอดิเตอร์ ed จะแสดง เท่าที่แสดงได้พอดี เริ่มต้นด้วยบรรทัดแรกที่แอดเดรส

**เมื่อต้องการแสดงบรรทัดที่มีรูปแบบการค้นหา**

พิมพ์คำสั่งย่อยต่อไปนี้:

Addressg/Pattern/p

โดยที่พารามิเตอร์ *Address* คือช่วงของบรรทัด และพารามิเตอร์ *Pattern* คือสตริงอักขระ หรือ นิพจน์ทั่วไปที่คุณต้องการ ค้นหา

บรรทัดที่มีรูปแบบที่ระบุ ถูกแสดงบนจอภาพ ถ้ากลุ่มของบรรทัดยาวไม่พอดีบนจอภาพ เอดิเตอร์ ed จะแสดง เท่าที่แสดงได้พอดี เริ่มต้นด้วยบรรทัดแรกที่แอดเดรส

**เมื่อต้องการแสดงบรรทัดที่มีรูปแบบการค้นหา รวมทั้งอักขระที่ไม่พิมพ์ออก**

พิมพ์คำสั่งย่อยต่อไปนี้:

[Address]g/Pattern/l

โดยที่ *Address* คือพารามิเตอร์ทางเลือก ที่ระบุช่วงของบรรทัดและพารามิเตอร์ *Pattern* คือสตริงอักขระ หรือ นิพจน์ทั่วไปที่คุณต้องการ ค้นหา ถ้าคุณเว้นพารามิเตอร์ *Address* เอดิเตอร์ ed จะค้นหาทั้งไฟล์

บรรทัดที่มีรูปแบบที่ระบุ ถูกแสดงบนจอภาพ อักขระที่ไม่พิมพ์ออกแสดงในจอแสดงผล ถ้ากลุ่มของบรรทัดยาวไม่พอดีบนจอภาพ เอดิเตอร์ ed จะแสดง เท่าที่แสดงได้พอดี เริ่มต้นด้วยบรรทัดแรกที่แอดเดรส

**เมื่อต้องการแสดงบรรทัดที่มีรูปแบบการค้นหา รวมทั้ง หมายเลขบรรทัด**

พิมพ์คำสั่งย่อยต่อไปนี้:

[Address]g/Pattern/n

โดยที่ *Address* คือพารามิเตอร์ทางเลือก ที่ระบุช่วงของบรรทัดและพารามิเตอร์ *Pattern* คือสตริงอักขระ หรือ นิพจน์ทั่วไปที่คุณต้องการ ค้นหา ถ้าคุณเว้นพารามิเตอร์ *Address* เอดิเตอร์ ed จะค้นหาทั้งไฟล์

บรรทัดที่มีรูปแบบที่ระบุ ถูกแสดงบนจอภาพ หมายเลขบรรทัดสำหรับแต่ละบรรทัดถูกแสดงข้างบรรทัด ถ้ากลุ่มของบรรทัดยาวไม่พอดีบนจอภาพ เอดิเตอร์ ed จะแสดง เท่าที่แสดงได้พอดี เริ่มต้นด้วยบรรทัดแรกที่แอดเดรส

**เมื่อต้องการแสดงบรรทัดที่ไม่มีรูปแบบการค้นหา**

พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]v/Pattern/p
```

โดยที่ *Address* คือพารามิเตอร์ทางเลือก ที่ระบุช่วงของบรรทัดและพารามิเตอร์ *Pattern* คือสตริงอักขระ หรือ นิพจน์ทั่วไป ที่คุณต้องการ ค้นหา ถ้าคุณเว้นพารามิเตอร์ *Address* เดีเตอร์ *ed* จะค้นหาทั้งไฟล์

บรรทัดที่ไม่มีรูปแบบที่ระบุ ถูกแสดงบนจอภาพ ถ้ากลุ่มของบรรทัดยาวไม่พอดีบนจอภาพ เดีเตอร์ *ed* จะแสดง เท่าที่แสดงได้พอดี เริ่มต้นด้วยบรรทัดแรกที่แอดเดรส

เมื่อต้องการแสดงบรรทัดที่ไม่มีรูปแบบการค้นหา รวมทั้งอักขระที่ไม่พิมพ์ออก

พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]v/Pattern/l
```

โดยที่ *Address* คือพารามิเตอร์ทางเลือก ที่ระบุช่วงของบรรทัดและพารามิเตอร์ *Pattern* คือสตริงอักขระ หรือ นิพจน์ทั่วไป ที่คุณต้องการ ค้นหา ถ้าคุณเว้นพารามิเตอร์ *Address* เดีเตอร์ *ed* จะค้นหาทั้งไฟล์

บรรทัดที่ไม่มีรูปแบบที่ระบุ ถูกแสดงบนจอภาพ รวมทั้งอักขระที่ไม่พิมพ์ออก ถ้ากลุ่มของบรรทัดยาวไม่พอดีบนจอภาพ เดีเตอร์ *ed* จะแสดง เท่าที่แสดงได้พอดี เริ่มต้นด้วยบรรทัดแรกที่แอดเดรส

เมื่อต้องการแสดงบรรทัดที่ไม่มีรูปแบบการค้นหา รวมทั้ง หมายเลขบรรทัด

พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]v/Pattern/n
```

โดยที่ *Address* คือพารามิเตอร์ทางเลือก ที่ระบุช่วงของบรรทัดและพารามิเตอร์ *Pattern* คือสตริงอักขระ หรือ นิพจน์ทั่วไป ที่คุณต้องการ ค้นหา ถ้าคุณเว้นพารามิเตอร์ *Address* เดีเตอร์ *ed* จะค้นหาทั้งไฟล์

บรรทัดที่ไม่มีรูปแบบที่ระบุ ถูกแสดงบนจอภาพ ตามด้วยหมายเลขบรรทัด ถ้ากลุ่มของบรรทัดยาวไม่พอดีบนจอภาพ เดีเตอร์ *ed* จะแสดง เท่าที่แสดงได้พอดี เริ่มต้นด้วยบรรทัดแรกที่แอดเดรส

## การรวมและการแยกบรรทัด

ไอเท็ม

```
(.,+1)j [l] [n] [p]
```

คำอธิบาย

คำสั่งย่อย *j* (join) รวมบรรทัดต่อเนื่อง โดยการลบอักขระ new-line ที่แทรกออก ถ้ากำหนดเพียงหนึ่งแอดเดรส คำสั่งย่อย *j* จะไม่ทำอะไร

พิมพ์คำสั่งย่อย *l* (list), *n* (number), หรือ *p* (print) ถ้าคุณต้องการแสดง บรรทัดที่รวม คำสั่งย่อยเหล่านี้เป็นทางเลือก

เอดิเตอร์ *ed* มีหลายวิธีในการรวมหรือแยก บรรทัด ใช้รูปแบบนำหน้าเพื่อดำเนินงานแก้ไข ต่อไปนี้:

- การรวมบรรทัดปัจจุบันและบรรทัดถัดไป
- การรวมบรรทัดที่แอดเดรส
- การแยกบรรทัดปัจจุบัน
- การแยกบรรทัดที่แอดเดรส

## เมื่อต้องการรวมบรรทัด ปัจจุบัน และถัดไป

พิมพ์คำสั่งย่อยต่อไปนี้:

```
j[1][n][p]
```

โดยที่ *l*, *n* และ *p* เป็นคำสั่งย่อยทางเลือกที่แสดงบรรทัดที่รวม

## เมื่อต้องการรวมบรรทัดที่แอดเดรส

พิมพ์คำสั่งย่อยต่อไปนี้:

```
Addressj[1][n][p]
```

โดยที่พารามิเตอร์ *Address* เป็นชื่อของบรรทัดต่อเนื่องที่จะฟอร์มหนึ่งบรรทัด และ *l*, *n* และ *p* เป็นคำสั่งย่อยทางเลือกที่แสดงบรรทัดที่รวม

## เมื่อต้องการแบ่ง บรรทัด ปัจจุบัน

1. เมื่อต้องการแยกบรรทัดปัจจุบันหลังรูปแบบที่ระบุให้พิมพ์คำสั่งย่อยต่อไปนี้:

```
s/Pattern/Pattern\
```

โดยที่พารามิเตอร์ *Pattern* คือสตริงอักขระที่คุณต้องการแยก ไปต่อท้าย

หมายเหตุ: ตรวจสอบว่าทั้งสองสตริงที่แสดงโดยพารามิเตอร์ *Pattern* เหมือนกัน

2. กด Enter
3. พิมพ์ backslash ต่อไปนี้:

```
/
```

4. เมื่อต้องการแสดงบรรทัดที่แยกให้พิมพ์หนึ่งในคำสั่งย่อยทางเลือกต่อไปนี้:

```
l
```

```
n
```

```
p
```

5. กด Enter

## เมื่อต้องการแยกบรรทัดที่แอดเดรส

1. เมื่อต้องการแยกบรรทัดที่แอดเดรสหลังรูปแบบที่ระบุให้พิมพ์คำสั่งย่อยต่อไปนี้:

```
Addresss/Pattern/Pattern\
```

โดยที่พารามิเตอร์ *Address* คือแอดเดรสของบรรทัดที่จะแยก และพารามิเตอร์ *Pattern* คือสตริงอักขระที่จะแยกบรรทัดไปต่อท้าย

หมายเหตุ: ตรวจสอบว่าทั้งสองสตริงที่แสดงโดยพารามิเตอร์ *Pattern* เหมือนกัน

2. กด Enter

3. พิมพ์ backslash ต่อไปนี้:

/

4. เมื่อต้องการแสดงบรรทัดที่แยกให้พิมพ์หนึ่งในคำสั่งย่อยทางเลือกต่อไปนี้:

l

n

p

5. กด Enter

## การทำการเปลี่ยนแปลงระดับโกลบอล

### ไอเท็ม

(1,\$)g/Pattern/SubcommandList [l] [n] [p]

#### คำอธิบาย

คำสั่งย่อย g (global) ชั้นแรกทำเครื่องหมายทุกบรรทัด ที่ตรงกับพารามิเตอร์ *Pattern* รูปแบบเป็นได้ทั้งสตริงอักขระคงที่หรือ นิพจน์ทั่วไป จากนั้น สำหรับแต่ละบรรทัดที่ทำเครื่องหมาย คำสั่งย่อยนี้ใช้บรรทัดปัจจุบันเป็น บรรทัดที่ทำเครื่องหมายและรันพารามิเตอร์ *SubcommandList* ป้อนคำสั่งย่อยเดียวหรือคำสั่งย่อยแรกของรายการคำสั่งย่อย บนบรรทัดเดียวกับคำสั่งย่อย g; ป้อนคำสั่งย่อย ต่อมาบนบรรทัดแยก ยกเว้นสำหรับบรรทัดสุดท้าย แต่ละบรรทัด ควรจบด้วย \ (backslash)

พารามิเตอร์ *SubcommandList* สามารถมีคำสั่งย่อย a, i, และ c และ อินพุต ถ้าคำสั่งสุดท้ายในพารามิเตอร์ *SubcommandList* โดยปกติจะเป็น . (จุด) ที่จบโหมด อินพุต . (จุด) เป็นทางเลือก ถ้าไม่มีพารามิเตอร์ *SubcommandList* อยู่บรรทัดปัจจุบันจะถูกแสดง พารามิเตอร์ *SubcommandList* ไม่อาจมีคำสั่งย่อย g, G, v หรือ V

พิมพ์คำสั่งย่อย l (list), n (number) หรือ p (print) ถ้าคุณต้องการแสดงการเปลี่ยนแปลง คำสั่งย่อยเหล่านี้เป็นทางเลือก

หมายเหตุ: คำสั่งย่อย g เหมือนกับคำสั่งย่อย v ซึ่งรันพารามิเตอร์ *SubcommandList* สำหรับทุกบรรทัดที่ไม่มีข้อมูลตรงกับรูปแบบ

(1,\$)G/Pattern/ [l] [n] [p]

คำสั่งย่อย G (Global) แบบโต้ตอบทำเครื่องหมาย ทุกบรรทัดที่ตรงพารามิเตอร์ *Pattern*, แสดง บรรทัดที่ทำเครื่องหมายแรก ใช้บรรทัดปัจจุบันเป็นบรรทัดนั้น แล้วรอ คำสั่งย่อย รูปแบบ เป็นได้ทั้งสตริงอักขระคงที่หรือ นิพจน์ทั่วไป

คำสั่งย่อย G ไม่ยอมรับคำสั่งย่อย a, i, c, g, G, v และ V หลังจาก คำสั่งย่อยเสร็จสิ้น คำสั่งย่อย G แสดง บรรทัดที่ทำเครื่องหมายถัดไป และต่อไป คำสั่งย่อย G ปฏิบัติ ต่ออักขระ new-line เหมือนคำสั่งย่อย null :& (โคลนแอมเปอร์แซนด์) ทำให้ คำสั่งย่อย G รับคำสั่งย่อยก่อนหน้านี้อีกครั้ง คุณสามารถหยุด คำสั่งย่อย G โดยการกด Ctrl+C

พิมพ์คำสั่งย่อย l (list), n (number) หรือ p (print) ถ้าคุณต้องการแสดงการเปลี่ยนแปลง คำสั่งย่อยเหล่านี้เป็นทางเลือก

(1,\$)v/Pattern/SubcommandList [l] [n] [p]

คำสั่งย่อย v รันคำสั่งย่อยในพารามิเตอร์ *SubcommandList* สำหรับแต่ละบรรทัดที่ไม่มีข้อมูลที่ตรงกับพารามิเตอร์ *Pattern* รูปแบบ เป็นได้ทั้งสตริงอักขระคงที่หรือ นิพจน์ทั่วไป

พิมพ์คำสั่งย่อย l (list), n (number) หรือ p (print) ถ้าคุณต้องการแสดงการเปลี่ยนแปลง คำสั่งย่อยเหล่านี้เป็นทางเลือก

คำสั่งย่อย v ไม่ยอมรับ คำสั่งย่อย a, i, c, g, G และ V

หมายเหตุ: คำสั่งย่อย v ทำให้คำสั่งย่อย g สมบูรณ์, ซึ่งรันพารามิเตอร์ *SubcommandList* สำหรับทุกบรรทัดที่มีข้อมูลตรงกับรูปแบบ

## ไอเท็ม

(1,\$)V/Pattern/ [l] [n] [p]

### คำอธิบาย

คำสั่งย่อย V ทำเครื่องหมายทุกบรรทัดที่ไม่ตรงกับพารามิเตอร์ *Pattern* แสดง บรรทัดที่ทำเครื่องหมายแรก เซ็ดบรรทัดปัจจุบันเป็นบรรทัดนั้น แล้วรอคำสั่งย่อย รูปแบบ เป็นได้ทั้งสตริงอักขระคงที่หรือนิพจน์ทั่วไป

พิมพ์คำสั่งย่อย l (list), n (number) หรือ p (print) ถ้าคุณต้องการแสดงการเปลี่ยนแปลง คำสั่งย่อยเหล่านี้เป็นทางเลือก

คำสั่งย่อย V ไม่ยอมรับ คำสั่งย่อย a, i, c, g, G และ v

หมายเหตุ: คำสั่งย่อย V ทำให้คำสั่ง G สมบูรณ์ ซึ่งทำเครื่องหมาย บรรทัดที่ตรงรูปแบบ

## การทำเครื่องหมายข้อความ

### ไอเท็ม

(.)kx [l] [n] [p]

### คำอธิบาย

คำสั่งย่อย k (mark) ทำเครื่องหมายบรรทัดที่แอดเดรส ด้วยชื่อที่ระบุโดยพารามิเตอร์ x, ซึ่งต้องเป็นตัวอักษร ASCII ตัวพิมพ์เล็ก แอดเดรส 'x' (เครื่องหมายคำพูดเดี่ยวก่อนอักขระการทำเครื่องหมาย) จากนั้นแอดเดรสบรรทัดนี้ คำสั่งย่อย k ไม่เปลี่ยนบรรทัด ปัจจุบัน

พิมพ์ l (list), n (number) หรือคำสั่งย่อย p (print) ถ้าคุณต้องการแสดงข้อความที่ทำเครื่องหมาย คำสั่งย่อยเหล่านี้เป็นทางเลือก

## เมื่อต้องการทำเครื่องหมายบรรทัด ปัจจุบัน

พิมพ์คำสั่งย่อยต่อไปนี้:

```
kLetter[l][n][p]
```

โดยที่พารามิเตอร์ *Letter* คือตัวอักษร a ถึง z สำหรับ เครื่องหมาย, และ l, n, และ p เป็น คำสั่งย่อยทางเลือกที่แสดงข้อความที่ทำเครื่องหมาย

## เมื่อต้องการทำเครื่องหมายบรรทัดที่แอดเดรส

พิมพ์คำสั่งย่อยต่อไปนี้:

```
AddresskLetter[l][n][p]
```

โดยที่พารามิเตอร์ *Address* เป็นหมายเลขบรรทัดหรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของบรรทัดที่คุณต้องการทำเครื่องหมาย และ พารามิเตอร์ *Letter* คือตัวอักษร a ถึง z สำหรับเครื่องหมาย คำสั่งย่อยทางเลือก l, n, และ p แสดงข้อความที่ทำเครื่องหมาย

## การย้ายข้อความ

ไอเท็ม  
(..)mA [l] [n] [p]

#### คำอธิบาย

คำสั่งย่อย *m* (move) จัดตำแหน่งบรรทัด ที่แอดเดรสใหม่ บรรทัดแรกที่ย้ายต่อท้ายบรรทัดที่แอดเดรส โดยพารามิเตอร์ *A* พารามิเตอร์ *O* ย้ายบรรทัดที่แอดเดรส ไปที่จุดเริ่มต้นของไฟล์ แอดเดรสที่ระบุโดย พารามิเตอร์ *A* ต้องไม่ใช่บรรทัดที่จะถูกย้าย คำสั่งย่อย *m* เชื่อบรรทัดปัจจุบันเป็นบรรทัดที่ย้ายสุดท้าย

พิมพ์คำสั่งย่อย *l* (list), *n* (number) หรือ *p* (print) ถ้าคุณต้องการแสดงการลบ คำสั่งย่อยเหล่านี้เป็นทางเลือก

การย้ายบรรทัดหรือชุดของบรรทัดลบบรรทัด ที่ระบุจากตำแหน่งเดิม และนำไปไว้ที่ตำแหน่งใหม่ คุณสามารถ เลือกบรรทัดที่จะย้ายตามแอดเดรสหรือรูปแบบ ใช้รูปแบบนำหน้าเพื่อดำเนินงานแก้ไข ต่อไปนี้:

- การย้ายบรรทัดปัจจุบัน
- การย้ายบรรทัดที่ระบุตามแอดเดรส
- การย้ายบรรทัดที่มีรูปแบบที่ระบุ
- การย้ายบรรทัดที่ไม่มีรูปแบบที่ระบุ

#### เมื่อต้องการย้ายบรรทัด ปัจจุบัน

พิมพ์คำสั่งย่อยต่อไปนี้:

```
mAddress[l][n][p]
```

โดยที่พารามิเตอร์ *Address* คือหมายเลขบรรทัดหรือการกำหนดแอดเดรสเชิงสัญลักษณ์ของบรรทัดที่คุณต้องการให้บรรทัดปัจจุบันต่อท้าย และ *l*, *n* และ *p* เป็น คำสั่งย่อยทางเลือกที่แสดงบรรทัดที่ย้าย

#### เมื่อต้องการย้ายบรรทัดที่ระบุโดยแอดเดรส

พิมพ์คำสั่งย่อยต่อไปนี้:

```
LineNumbermDestinationAddress[l][n][p]
```

โดยที่พารามิเตอร์ *LineNumber* คือแอดเดรสของบรรทัดที่คุณต้องการย้าย และพารามิเตอร์ *DestinationAddress* คือบรรทัดที่คุณต้องการให้ย้ายไปต่อท้าย คำสั่งย่อยทางเลือก *l*, *n* และ *p* แสดงบรรทัดที่ย้าย

#### เมื่อต้องการย้ายบรรทัดที่มีรูปแบบที่ระบุ

พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]g/Pattern/m[DestinationAddress][l][n][p]
```

โดยที่ *Address* เป็นพารามิเตอร์ ทางเลือกที่ระบุช่วงของบรรทัดเพื่อค้นหาบรรทัดที่มี รูปแบบที่ระบุ พารามิเตอร์ *Pattern* คือข้อความที่คุณกำลังค้นหา และ *DestinationAddress* เป็นพารามิเตอร์ทางเลือกที่แสดงบรรทัดที่คุณต้องการย้ายไปต่อท้าย คำสั่งย่อยทางเลือก *l*, *n* และ *p* แสดงบรรทัดที่ย้าย

ถ้าคุณเว้นพารามิเตอร์ *Address* เอดิเตอร์ *ed* จะค้นหาทั้งไฟล์เพื่อหาบรรทัดที่มีรูปแบบ ถ้าคุณเว้นพารามิเตอร์ *DestinationAddress* ข้อความที่ ย้ายจะถูกวางไว้หลังจากบรรทัดปัจจุบัน

#### เมื่อต้องการย้ายบรรทัดที่ไม่มีรูปแบบที่ระบุ

พิมพ์คำสั่งย่อยต่อไปนี้:

```
[Address]v/Pattern/m[DestinationAddress][l][n][p]
```

โดยที่ *Address* เป็นพารามิเตอร์ ทางเลือกที่ระบุช่วงของบรรทัดเพื่อค้นหาบรรทัดที่ไม่มีรูปแบบที่ระบุ พารามิเตอร์ *Pattern* คือข้อความ และ *DestinationAddress* เป็นพารามิเตอร์ทางเลือกที่แสดงบรรทัดที่คุณต้องการย้ายข้อความไปต่อท้าย คำสั่งย่อยทางเลือก *l*, *n* และ *p* แสดงบรรทัดที่ย้าย

ถ้าคุณเว้นพารามิเตอร์ *Address* เดีเตอร์ *ed* จะค้นหาทั้งไฟล์เพื่อหา บรรทัดที่มีรูปแบบ ถ้าคุณเว้นพารามิเตอร์ *DestinationAddress* ข้อความที่ย้ายจะถูกวางไว้หลังจากบรรทัดปัจจุบัน

## การบันทึกข้อความ

### ไอเท็ม

(1,\$)w File

### คำอธิบาย

คำสั่งย่อย *w* (write) คัดลอกบรรทัดที่แอดเดรสจากบัฟเฟอร์ไปที่ไฟล์ที่ระบุโดยพารามิเตอร์ *File* ถ้าไฟล์ไม่มีอยู่คำสั่งย่อย *w* จะสร้าง ขึ้นมาโดยมีไดรัลทิการใช้งาน 666 (สิทธิ *read* และ *write* สำหรับทุกคน), นอกจากการตั้งค่า *umask* ระบุโหมด การสร้างไฟล์อื่น

คำสั่งย่อย *w* ไม่เปลี่ยน ชื่อไฟล์ดีฟอลต์ (นอกจากพารามิเตอร์ *File* เป็นชื่อไฟล์แรกที่ใช้ตั้งแต่คุณเริ่มเดีเตอร์ *ed*) ถ้าคุณไม่ได้ระบุชื่อไฟล์ คำสั่งย่อย *w* จะใช้ชื่อไฟล์ ดีฟอลต์ คำสั่งย่อย *w* ไม่เปลี่ยนบรรทัด ปัจจุบัน

ถ้าเดีเตอร์ *editor* เขียนไฟล์จาก บัฟเฟอร์สำเร็จ จะแสดงจำนวนอักขระที่เขียน ถ้าคุณระบุคำสั่งย่อย ! *Command* แทนชื่อไฟล์ คำสั่งย่อย *w* จะอ่านเอาต์พุต ของคำสั่งระบบปฏิบัติการที่ระบุด้วยพารามิเตอร์ *Command* คำสั่งย่อย *w* ไม่บันทึกชื่อของคำสั่งระบบปฏิบัติการ ที่คุณระบุเป็นชื่อไฟล์ดีฟอลต์

หมายเหตุ: เนื่องจาก 0 ไม่ใช่แอดเดรสที่ถูกต้องสำหรับคำสั่งย่อย *w* คุณไม่สามารถ สร้างไฟล์เปล่าด้วยคำสั่ง *ed*

คุณสามารถบันทึกการเปลี่ยนแปลงกับไฟล์ได้หลายวิธี ใช้รูปแบบนำหน้าเพื่อดำเนินการต่อไปนี้:

- การบันทึกไฟล์เป็นไฟล์ปัจจุบัน
- การบันทึกส่วนของไฟล์เป็นไฟล์ปัจจุบัน
- การบันทึกไฟล์เป็นไฟล์อื่น
- การบันทึกส่วนของไฟล์เป็นไฟล์อื่น

## เมื่อต้องการบันทึกไฟล์ลงในไฟล์ ปัจจุบัน

พิมพ์คำสั่งย่อยต่อไปนี้:

```
w
```

ไฟล์ปัจจุบันถูกบันทึกด้วยชื่อปัจจุบัน และ เดีเตอร์ *ed* แสดงจำนวนอักขระที่เขียน

## เมื่อต้องการบันทึกส่วนของไฟล์ไปยังไฟล์ ปัจจุบัน

พิมพ์คำสั่งย่อยต่อไปนี้:

```
Addressw
```

โดยที่พารามิเตอร์ *Address* ระบุ บรรทัดหรือกลุ่มบรรทัดที่จะเขียน เดีเตอร์ *ed* แสดงจำนวนอักขระ ที่เขียน

## เมื่อต้องการบันทึกไฟล์เป็นไฟล์อื่น

พิมพ์คำสั่งย่อต่อไปนี้:

w File

โดยที่พารามิเตอร์ *File* คือชื่อของไฟล์ที่จะเขียน

ไฟล์ปัจจุบันถูกบันทึกเป็นไฟล์ที่ระบุโดย พารามิเตอร์ *File* เอดิเตอร์ ed แสดงจำนวนอักขระที่เขียน

เมื่อต้องการบันทึกส่วนของไฟล์เป็นไฟล์อื่น

พิมพ์คำสั่งย่อต่อไปนี้:

Addressw File

โดยที่พารามิเตอร์ *Address* ระบุบรรทัดหรือกลุ่มบรรทัดที่จะเขียนและพารามิเตอร์ *File* ระบุไฟล์ที่จะเขียน

บรรทัดที่ระบุถูกบันทึกเป็นไฟล์ที่ระบุโดย พารามิเตอร์ *File* เอดิเตอร์ ed แสดงจำนวนอักขระที่เขียน

**การค้นหาข้อความ**

คุณสามารถค้นหาไปยังหน้าหรือย้อนกลับจากบรรทัดปัจจุบัน เพื่อค้นหารูปแบบข้อความ รูปแบบเป็นได้ทั้งสตริงอักขระหรือนิพจน์ทั่วไปที่กำหนดจากอักขระตามตัวอักษร และอักขระพิเศษ ^ (circumflex), \$ (เครื่องหมายดอลลาร์), . (จุด), [ (วงเล็บเหลี่ยมซ้าย), ] (วงเล็บเหลี่ยมขวา), \* (เครื่องหมายดอกจัน), \ (backslash), % (เครื่องหมายเปอร์เซ็นต์) และ & คีย์

คุณสามารถใช้เอดิเตอร์ ed เพื่อดำเนินการค้นหาข้อความ ดังต่อไปนี้:

- การค้นหาไปยังหน้า
- การค้นหาย้อนกลับ
- การทำซ้ำการค้นหาในทิศทางเดิม
- การทำซ้ำการค้นหาในทิศทางตรงข้าม

เมื่อต้องการค้นหาไปยังหน้า

พิมพ์คำสั่งย่อต่อไปนี้:

/Pattern

โดยที่พารามิเตอร์ *Pattern* คือสตริงอักขระหรือนิพจน์ทั่วไปที่ระบุข้อความที่จะค้นหา

เคอร์เซอร์ย้ายไปที่อักขระแรกของข้อความ ที่ระบุโดยรูปแบบ

เมื่อต้องการค้นหาย้อนกลับ

พิมพ์คำสั่งย่อต่อไปนี้:

?Pattern

โดยที่พารามิเตอร์ *Pattern* คือสตริงอักขระหรือนิพจน์ทั่วไปที่ระบุข้อความที่จะค้นหา

เคอร์เซอร์ย้ายไปที่อักขระแรกของข้อความ ที่ระบุโดยรูปแบบ

## เมื่อต้องการทำซ้ำการค้นหาในทิศทางเดียวกัน

พิมพ์คำสั่งย่อยต่อไปนี้:

/

เคอร์เซอร์ย้ายไปที่อักขระแรกของ อินสแตนซ์ที่ไกลที่สุดของข้อความที่ระบุโดยรูปแบบในคำสั่งค้นหาสุดท้าย

## เมื่อต้องการทำซ้ำการค้นหาในทิศทางตรงกันข้าม

พิมพ์คำสั่งย่อยต่อไปนี้:

?

เคอร์เซอร์ย้ายไปที่อักขระแรกของ อินสแตนซ์ที่ไกลที่สุดของข้อความที่ระบุโดยรูปแบบในคำสั่งค้นหาสุดท้าย

## การแทนที่ข้อความ

### ไอเท็ม

(.,.)s/Pattern/Replacement/ [l] [n] [p] (.,.)  
s/Pattern/Replacement/ng [l] [n] [p]

### คำอธิบาย

คำสั่งย่อย s (substitute) ค้นหาแต่ละ บรรทัดที่แอดเดรสสำหรับสตริงที่ตรงกับพารามิเตอร์ *Pattern* และแทนที่สตริงด้วยพารามิเตอร์ *Replacement* ที่ระบุ รูปแบบ เป็นได้ทั้งสตริงอักขระคงที่หรือ นิพจน์ทั่วไป โดยไม่มีคำสั่งย่อยโกลบอล (g), คำสั่งย่อย s แทนที่ เฉพาะสตริงแรกที่ตรง บนแต่ละบรรทัดที่แอดเดรส ด้วยคำสั่งย่อย g คำสั่งย่อย s แทนที่ทุกตำแหน่งที่พบ สตริงที่ตรงกัน บนแต่ละบรรทัดที่แอดเดรส ถ้าคำสั่งย่อย s ไม่ค้นหาข้อมูลที่ตรงกับรูปแบบ จะมีการส่งกลับ ข้อความแสดงความผิดพลาด? (เครื่องหมายคำถาม)

พิมพ์คำสั่งย่อย l (list), n (number) หรือ p (print) เพื่อแสดงข้อความ ที่แทนที่ คำสั่งย่อยเหล่านี้เป็นทางเลือก

หมายเหตุ: อักขระทั้งหมดยกเว้น อักขระช่องว่างหรืออักขระขึ้นบรรทัดใหม่สามารถแยก (ค้น) พารามิเตอร์ *Pattern* และ *Replacement* คำสั่งย่อย s เช็ตบรรทัดปัจจุบันเป็นบรรทัดสุดท้ายที่เปลี่ยนแปลง

ถ้าพารามิเตอร์ *Number* (จำนวนเต็ม) ถูกระบุ หมายเลขแรกที่ตรงกับสตริงในแต่ละบรรทัด ที่แอดเดรสจะถูกแทนที่

อักขระ & (แอมเปอร์แซนด์) ที่ใช้ในพารามิเตอร์ *Replacement* มีค่าเดียวกับ พารามิเตอร์ *Pattern* ตัวอย่าง คำสั่งย่อย s/are/&n't/ มีผลเหมือนกับคำสั่งย่อย s/are/aren't/ และแทนที่ are ด้วย aren't บนบรรทัด ปัจจุบัน เครื่องหมาย & (แอมเปอร์แซนด์) ลบความหมาย พิเศษของอักขระ & ในพารามิเตอร์ *Replacement*

รูปแบบย่อยคือส่วนของรูปแบบปิดด้วยสตริง \ (backslash, วงเล็บ ซ้าย) และ \ (backslash, วงเล็บขวา); รูปแบบทำงานเหมือน ว่าอักขระที่ปิดไว้ไม่มีอยู่ในพารามิเตอร์ *Replacement*, \Number อ้างอิงถึงสตริง ที่ตรงกับรูปแบบย่อย ตัวอย่างคำสั่งย่อย s/(t)(h)\(e)/t1\2ose/ แทนที่ the ด้วย those ถ้าข้อมูลตรงกับรูปแบบ the มีอยู่ในบรรทัดปัจจุบัน ไม่ว่ารูปแบบย่อยถูกซ่อนหรืออยู่ในชุด, \Number อ้างอิงถึงข้อมูลที่พบที่ระบุโดยพารามิเตอร์ *Number* นับจากซ้ายของอักขระที่ค้น \ (backslash, วงเล็บขวา)

% (เครื่องหมายเปอร์เซ็นต์), เมื่อใช้เป็นพารามิเตอร์ *Replacement* ทำให้ คำสั่งย่อย s ทำซ้ำพารามิเตอร์ *Replacement* ก่อนหน้านี้ % ไม่มีความหมายพิเศษ ถ้าเป็น ส่วนหนึ่งของพารามิเตอร์ *Replacement* ที่ยาวกว่าหรือถ้าถูก นำหน้าด้วย \ (backslash)

คุณสามารถแยกบรรทัด โดยการแทนที่อักขระขึ้นบรรทัดใหม่ลงไปในพารามิเตอร์ *Replacement* การกด ลำดับคีย์ \+Enter ปิด อักขระขึ้นบรรทัดใหม่ (ไม่แสดง) ด้วยเครื่องหมายคำพูดและย้ายเคอร์เซอร์ไปที่ บรรทัดถัดไป สำหรับส่วนที่เหลือของสตริง อักขระขึ้นบรรทัดใหม่ไม่สามารถถูกแทนที่เป็น ส่วนหนึ่งของคำสั่งย่อย g หรือรายการคำสั่งย่อย v

เอดิเตอร์ ed จัดเตรียมหลายวิธีเพื่อแทนที่ข้อความ ใช้รูปแบบนำหน้าเพื่อดำเนินงานแก้ไข ต่อไปนี้:

- การแทนที่ข้อความภายใน บรรทัดปัจจุบัน
- การแทนที่ข้อความภายใน บรรทัดที่แอตเดรสหรือกลุ่มบรรทัด
- การแทนที่รูปแบบที่ระบุภายในบรรทัดที่มีรูปแบบนั้น
- การแทนที่รูปแบบภายใน บรรทัดที่มีรูปแบบต่างไป
- การแทนที่รูปแบบภายใน บรรทัดที่ไม่มีรูปแบบอื่น

#### เมื่อต้องการแทนข้อความภายในบรรทัด ปัจจุบัน

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
s/OldString/NewString
```

โดยที่พารามิเตอร์ *OldString* คือข้อความที่มีอยู่ และพารามิเตอร์ *NewString* เป็นข้อความที่คุณต้องการ แทนที่

2. พิมพ์หนึ่งในการดำเนินการต่อไปนี้:

เมื่อต้องการแทนที่ พารามิเตอร์ *NewString* สำหรับอินสแตนซ์แรกของ พารามิเตอร์ *OldString* ภายในบรรทัดปัจจุบัน ให้พิมพ์:

```
/
```

เมื่อต้องการแทนที่พารามิเตอร์ *NewString* สำหรับทุกอินสแตนซ์ของ พารามิเตอร์ *OldPattern* ภายในบรรทัดปัจจุบัน ให้พิมพ์:

```
/g
```

3. เมื่อต้องการแสดงข้อความที่เปลี่ยนแปลง ให้พิมพ์ หนึ่งในคำสั่งย่อยทางเลือกต่อไปนี้:

```
l
```

```
n
```

```
p
```

4. กด Enter

#### เมื่อต้องการแทนที่ข้อความภายในบรรทัดที่แอตเดรสหรือกลุ่มบรรทัด

1. พิมพ์คำสั่งย่อยต่อไปนี้:

```
Addresss/OldPattern/NewString
```

โดยที่พารามิเตอร์ *Address* เป็นแอตเดรส ของบรรทัดหรือกลุ่มบรรทัดที่ต้องการแทนที่ข้อความ พารามิเตอร์ *OldPattern* เป็นข้อความที่มีอยู่ และพารามิเตอร์ *NewString* เป็นข้อความที่คุณต้องการแทนที่

2. พิมพ์หนึ่งในการดำเนินการต่อไปนี้:

เมื่อต้องการแทนที่ พารามิเตอร์ *NewString* สำหรับอินสแตนซ์แรกของ พารามิเตอร์ *OldPattern* ภายในแต่ละบรรทัด ให้พิมพ์:

```
/NewString/
```

เมื่อต้องการแทนที่พารามิเตอร์ *NewString* สำหรับทุกอินสแตนซ์ของ พารามิเตอร์ *OldPattern* ภายในแต่ละบรรทัด ให้พิมพ์:

/NewString/g

เมื่อต้องการแทนที่พารามิเตอร์ *NewString* สำหรับอินสแตนซ์แรกของ พารามิเตอร์ *NumberOldPattern* บนแต่ละบรรทัด แอดเดรสให้พิมพ์:

/NewString/Number

3. เมื่อต้องการแสดงข้อความที่เปลี่ยนแปลง ให้พิมพ์ หนึ่งในคำสั่งย่อยทางเลือกต่อไปนี้:

|

n

p

4. กด Enter

เมื่อต้องการแทนที่รูปแบบที่ระบุภายในบรรทัดที่มีรูปแบบนั้น

1. พิมพ์คำสั่งย่อยต่อไปนี้:

Addressg/Pattern/s//NewString

โดยที่พารามิเตอร์ *Address* คือแอดเดรสของกลุ่มบรรทัดที่คุณต้องการค้นหารูปแบบที่ระบุด้วย พารามิเตอร์ *Pattern* และพารามิเตอร์ *NewString* เป็นข้อความที่คุณต้องการแทนที่สำหรับพารามิเตอร์ *Pattern*

2. พิมพ์หนึ่งในการดำเนินการต่อไปนี้:

เมื่อต้องการแทนที่ พารามิเตอร์ *NewString* สำหรับอินสแตนซ์แรกของ พารามิเตอร์ *Pattern* ภายในแต่ละบรรทัดให้พิมพ์:

/

เมื่อต้องการแทนที่พารามิเตอร์ *NewString* สำหรับทุกอินสแตนซ์ของ พารามิเตอร์ *Pattern* ภายในแต่ละบรรทัดให้พิมพ์:

/g

3. เมื่อต้องการแสดงข้อความที่เปลี่ยนแปลง ให้พิมพ์ หนึ่งในคำสั่งย่อยทางเลือกต่อไปนี้:

|

n

p

4. กด Enter

เมื่อต้องการแทนที่รูปแบบภายในบรรทัดที่มีรูปแบบต่างไป

1. พิมพ์คำสั่งย่อยต่อไปนี้:

Addressg/Pattern/s/OldString/NewString

โดยที่พารามิเตอร์ *Address* คือแอดเดรสของกลุ่มบรรทัดที่คุณต้องการค้นหารูปแบบที่ระบุด้วย พารามิเตอร์ *Pattern* พารามิเตอร์ *OldString* เป็นข้อความที่คุณต้องการแทนที่ และพารามิเตอร์ *NewString* คือข้อความที่คุณต้องการแทนที่ในตำแหน่งของพารามิเตอร์ *OldString*

2. พิมพ์หนึ่งในการดำเนินการต่อไปนี้:

เมื่อต้องการแทนที่ พารามิเตอร์ *NewString* สำหรับอินสแตนซ์แรกของ พารามิเตอร์ *OldString* ภายในแต่ละบรรทัดที่มีพารามิเตอร์ *Pattern* ให้พิมพ์:

/

เมื่อต้องการแทนที่ พารามิเตอร์ *NewString* สำหรับทุกอินสแตนซ์ของ พารามิเตอร์ *OldString* ภายในแต่ละบรรทัดที่มีพารามิเตอร์ *Pattern* ให้พิมพ์:

/g

3. เมื่อต้องการแสดงข้อความที่เปลี่ยนแปลง ให้พิมพ์ หนึ่งในคำสั่งย่อยทางเลือกต่อไปนี้:

l

n

p

4. กด Enter

เมื่อต้องการแทนที่รูปแบบภายในบรรทัดที่ไม่มีรูปแบบ ต่างไป

1. พิมพ์คำสั่งย่อยต่อไปนี้:

Addressv/Pattern/s/OldString/NewString

โดยที่พารามิเตอร์ *Address* คือแอดเดรสของกลุ่มบรรทัดที่คุณต้องการค้นหารูปแบบที่ระบุด้วย พารามิเตอร์ *Pattern* พารามิเตอร์ *OldString* เป็นข้อความที่คุณต้องการแทนที่ และพารามิเตอร์ *NewString* คือข้อความที่คุณต้องการแทนที่ในตำแหน่งของพารามิเตอร์ *OldString*

2. พิมพ์หนึ่งในการดำเนินการต่อไปนี้:

เมื่อต้องการแทนที่ พารามิเตอร์ *NewString* สำหรับอินสแตนซ์แรกของ พารามิเตอร์ *OldString* ภายในแต่ละบรรทัดที่ไม่มีพารามิเตอร์ *Pattern* ให้พิมพ์:

/

เมื่อต้องการแทนที่ พารามิเตอร์ *NewString* สำหรับทุกอินสแตนซ์ของ พารามิเตอร์ *OldString* ภายในแต่ละบรรทัดที่ไม่มีพารามิเตอร์ *Pattern* ให้พิมพ์:

/g

3. เมื่อต้องการแสดงข้อความที่เปลี่ยนแปลง ให้พิมพ์ หนึ่งในคำสั่งย่อยทางเลือกต่อไปนี้:

l

n

p

#### 4. กด Enter

### การยกเลิกการเปลี่ยนแปลงข้อความ

ไอเท็ม u [l] [n] [p]	คำอธิบาย คำสั่งย่อย u (undo) เรียกคืนบัฟเฟอร์ กลับสู่สถานะที่เป็นก่อนการแก้ไขล่าสุดโดยคำสั่งย่อยเอดิเตอร์ ed คำสั่งย่อย u ไม่สามารถเลิกทำคำสั่งย่อย e, f และ w  พิมพ์คำสั่งย่อย l (list), n (number) หรือ p (print) ถ้าคุณต้องการแสดงการเปลี่ยนแปลง คำสั่งย่อยเหล่านี้เป็นทางเลือก
-------------------------	---

### เมื่อต้องการยกเลิกการเปลี่ยนแปลงข้อความ

พิมพ์คำสั่งย่อยต่อไปนี้:

```
u[l][n][p]
```

โดยที่ l, n และ p เป็นคำสั่งย่อยทางเลือกที่แสดงการเปลี่ยนแปลง ฟังก์ชันการแก้ไข add, change, move, copy หรือ delete ทั้งหมดที่ทำงานกับข้อความหลังการบันทึกล่าสุดถูกยกเลิก

### การจัดการไฟล์

คุณสามารถใช้คำสั่งย่อย ed เพื่อจัดการไฟล์เพื่อดำเนินงาน ต่อไปนี้:

- การเพิ่มไฟล์อื่นให้กับไฟล์ ปัจจุบัน
- การเปลี่ยนชื่อไฟล์ดีฟอลต์
- แก้ไขไฟล์เพิ่มเติม

#### การเพิ่มไฟล์อื่นในไฟล์ ปัจจุบัน

ไอเท็ม (@)r File	คำอธิบาย คำสั่งย่อย r (read) อ่านไฟล์ลงใน บัฟเฟอร์หลังบรรทัดที่แอดเดรส คำสั่งย่อย r ไม่ลบ เนื้อหาก่อนหน้าของบัฟเฟอร์ เมื่อป้อนโดยไม่มีพารามิเตอร์ File คำสั่งย่อย r อ่านไฟล์ดีฟอลต์ ถ้ามีลงในบัฟเฟอร์ คำสั่งย่อย r ไม่เปลี่ยนชื่อไฟล์ดีฟอลต์
---------------------	---

0 แอดเดรสทำให้คำสั่งย่อย r อ่านไฟล์ที่จุดเริ่มต้น ของบัฟเฟอร์ หลังจากอ่านไฟล์สำเร็จ คำสั่งย่อย r แสดงจำนวนอักขระที่อ่านลงในบัฟเฟอร์ และเซตบรรทัดปัจจุบันเป็นบรรทัดสุดท้ายที่อ่าน

ถ้า ! (เครื่องหมายตกใจ) แทนพารามิเตอร์ File ใน คำสั่งย่อย r ส่วนที่เหลือของบรรทัดถูกรับเป็น คำสั่งเซลล์ระบบปฏิบัติการซึ่งเอาต์พุตจะถูกอ่าน คำสั่งย่อย r ไม่เก็บชื่อของคำสั่งระบบปฏิบัติการเป็น ชื่อไฟล์ดีฟอลต์

#### เมื่อต้องการแทรกไฟล์หลังบรรทัด ปัจจุบัน

พิมพ์คำสั่งย่อยต่อไปนี้:

```
r File
```

โดยที่พารามิเตอร์ File คือชื่อของ ไฟล์ที่จะถูกแทรก

เอดิเตอร์ ed อ่านไฟล์ที่ระบุโดยพารามิเตอร์ *File* ลงในไฟล์ปัจจุบัน หลังจากบรรทัดปัจจุบันและแสดงจำนวนอักขระที่อ่านลงในบรรทัดปัจจุบัน

### เมื่อต้องการแทรกไฟล์หลังบรรทัดที่ระบุโดยแอดเดรส

พิมพ์คำสั่งย่อยต่อไปนี้:

Addressr File

โดยที่พารามิเตอร์ *Address* ระบุบรรทัดที่คุณต้องการให้ไฟล์ที่แทรกไปต่อท้าย และพารามิเตอร์ *File* คือชื่อของไฟล์ที่จะถูกแทรก

เอดิเตอร์ ed อ่านไฟล์ที่ระบุโดยพารามิเตอร์ *File* ลงในไฟล์ปัจจุบัน หลังจากบรรทัดที่ระบุและแสดงจำนวนอักขระที่อ่านลงในบรรทัดปัจจุบัน

### การเปลี่ยนชื่อไฟล์ดีฟอลต์

ไอเท็ม	คำอธิบาย
f [ <i>File</i> ]	คำสั่งย่อย f (file name) เปลี่ยนชื่อไฟล์ดีฟอลต์ (ชื่อที่เก็บของไฟล์สุดท้ายที่ใช้) กับชื่อที่ระบุโดย พารามิเตอร์ <i>File</i> ถ้าพารามิเตอร์ <i>File</i> ไม่ถูกระบุ คำสั่งย่อย f แสดงชื่อไฟล์ ดีฟอลต์ (คำสั่งย่อย e เก็บชื่อไฟล์ดีฟอลต์)

### เมื่อต้องการแสดงชื่อไฟล์

พิมพ์คำสั่งย่อยต่อไปนี้:

f

เอดิเตอร์ ed แสดงชื่อของไฟล์ใน บัฟเฟอร์แก้ไข

### เมื่อต้องการตั้งชื่อไฟล์

พิมพ์คำสั่งย่อยต่อไปนี้:

f File

โดยที่พารามิเตอร์ *File* คือชื่อใหม่ สำหรับไฟล์ในบัฟเฟอร์แก้ไข

ไฟล์ในบัฟเฟอร์แก้ไขถูกเปลี่ยนชื่อ

### การแก้ไขไฟล์เพิ่มเติม

ไอเท็ม  
e File

**คำอธิบาย**

คำสั่งย่อย e (edit) ชั้นแรกจะลบเนื้อหาจากบัฟเฟอร์ เซ็ทบรรทัดปัจจุบันเป็นบรรทัดสุดท้ายของบัฟเฟอร์ และ แสดงจำนวนอักขระที่อ่านลงในบัฟเฟอร์ ถ้าบัฟเฟอร์ได้ถูก เปลี่ยนแปลง ตั้งแต่ที่เนื้อหาถูกบันทึก (ด้วยคำสั่งย่อย w) เอดิเตอร์ ed แสดง ? (เครื่องหมายคำถาม) ก่อนที่จะเคลียร์บัฟเฟอร์

คำสั่งย่อย e เก็บพารามิเตอร์ File เป็นชื่อไฟล์ดีฟอลต์ที่จะถูกใช้ ถ้าจำเป็น โดยคำสั่งย่อย e, r หรือ w ต่อมา (เมื่อต้องการเปลี่ยนชื่อชื่อไฟล์ดีฟอลต์ให้ใช้คำสั่งย่อย f)

เมื่อ ! (เครื่องหมายตกใจ) แทนที่พารามิเตอร์ File คำสั่งย่อย e รับส่วนที่เหลือของบรรทัดเป็นคำสั่ง เซลล์ระบบปฏิบัติการและอ่านเอาต์พุตคำสั่ง คำสั่งย่อย e ไม่เก็บชื่อของคำสั่งเซลล์เป็นชื่อไฟล์ ดีฟอลต์

E File

คำสั่งย่อย E (Edit) ทำงานเหมือนกับคำสั่งย่อย e โดยมีหนึ่งข้อยกเว้น; คำสั่งย่อย E ไม่ตรวจสอบการเปลี่ยนแปลงที่ทำกับ บัฟเฟอร์หลังจากคำสั่งย่อย w สุดท้าย การเปลี่ยนแปลงที่คุณได้ทำไปแล้วก่อนทำการแก้ไขซ้ำจะสูญหาย

คุณสามารถใช้คำสั่งย่อย e หรือ E เพื่อดำเนินงานต่อไปนี้:

- การแก้ไขไฟล์ปัจจุบันซ้ำ โดยไม่ทำการบันทึก
- การแก้ไขไฟล์ปัจจุบันซ้ำ หลังการบันทึก
- การแก้ไขไฟล์หลังจาก ไฟล์ปัจจุบันถูกบันทึก
- การแก้ไขไฟล์โดยไม่บันทึก ไฟล์ปัจจุบัน

**เมื่อต้องการแก้ไขไฟล์ ปัจจุบัน อีกครั้งโดยไม่บันทึก ไฟล์**

พิมพ์คำสั่งย่อยต่อไปนี้:

E

เอดิเตอร์ ed แสดงจำนวนของอักขระใน ไฟล์ การเปลี่ยนแปลงที่คุณได้ทำไปแล้วก่อนทำการแก้ไขซ้ำจะสูญหาย

**เมื่อต้องการ แก้ไขไฟล์ ปัจจุบัน อีกครั้ง หลังจากบันทึกไฟล์**

พิมพ์คำสั่งย่อยต่อไปนี้:

e

เอดิเตอร์ ed แสดงจำนวนของอักขระใน ไฟล์

**เมื่อต้องการแก้ไขไฟล์หลังจากไฟล์ ปัจจุบัน ถูกบันทึก**

พิมพ์คำสั่งย่อยต่อไปนี้:

e File

โดยที่พารามิเตอร์ File คือ ชื่อของไฟล์ที่มีอยู่หรือไฟล์ใหม่ที่คุณต้องการแก้ไข

สำหรับไฟล์ที่มีอยู่ เอดิเตอร์ ed แสดงจำนวน อักขระในไฟล์ สำหรับไฟล์ใหม่เอดิเตอร์ ed แสดง ? (เครื่องหมายคำถาม) และชื่อของไฟล์

**เมื่อต้องการแก้ไขไฟล์โดยไม่ บันทึกไฟล์ ปัจจุบัน**

พิมพ์คำสั่งย่อยต่อไปนี้:

E File

โดยที่พารามิเตอร์ *File* คือ ชื่อของไฟล์ที่มีอยู่หรือไฟล์ใหม่ที่คุณต้องการแก้ไข

สำหรับไฟล์ที่มีอยู่ เอดิเตอร์แสดงจำนวน อักขระในไฟล์ สำหรับไฟล์ใหม่เอดิเตอร์ ed แสดง ? (เครื่องหมายคำถาม) และชื่อของไฟล์

## ฟังก์ชันเบ็ดเตล็ดของคำสั่งย่อยเอดิเตอร์ ed

คุณสามารถใช้คำสั่งย่อยเอดิเตอร์ ed ในการดำเนินงานดังต่อไปนี้:

- เปลี่ยนพร้อมต์สตริง
- ป้อนคำสั่งระบบ
- ออกจากเอดิเตอร์ ed
- ร้องขอวิธีใช้

### การเปลี่ยนพร้อมต์สตริง

ไอเท็ม คำอธิบาย

P คำสั่งย่อย P (Prompt) เปิดหรือปิดสตริงพร้อมต์ เอดิเตอร์ ed ซึ่งถูกแสดงโดย \* (เครื่องหมายดอกจัน) ค่าโดยเริ่มต้น คำสั่งย่อย P ถูกปิด

### เมื่อต้องการสตาาร์ทหรือหยุดการแสดงสตริงพร้อมต์

พิมพ์คำสั่งย่อยต่อไปนี้:

P

เอดิเตอร์ ed พร้อมต์, an \* (เครื่องหมายดอกจัน) ถูกแสดง หรือไม่แสดง ขึ้นกับการตั้งค่าก่อนหน้านี้

### การเข้าสู่คำสั่งระบบ

ไอเท็ม  
! คำสั่ง

คำอธิบาย

! คำสั่งย่อยอนุญาตให้คุณรันคำสั่งระบบปฏิบัติการ โดยไม่ต้องออกจากเอดิเตอร์ ed ข้อมูลที่ต่อจากคำสั่งย่อย เอดิเตอร์ ed ถูกแปล เป็นคำสั่งระบบปฏิบัติการ ภายในข้อความของสตริงคำสั่งนั้น เอดิเตอร์ ed แทนที่ unescaped % (เครื่องหมายเปอร์เซ็นต์) ด้วยชื่อไฟล์ปัจจุบัน ถ้ามีอยู่

คุณสามารถทำซ้ำคำสั่งระบบปฏิบัติการก่อนหน้านี้ โดยป้อน ! (เครื่องหมายตกใจ) หลัง ! ซึ่งเป็นคำสั่งย่อยเอดิเตอร์ ed ถ้าตัวแปลคำสั่ง ระบบปฏิบัติการ (คำสั่ง sh) ขยายสตริงคำสั่ง เอดิเตอร์ ed แสดงบรรทัดที่ขยาย ! ไม่เปลี่ยนบรรทัดปัจจุบัน

คุณสามารถใช้คำสั่งย่อย ! เพื่อดำเนิน การต่อไปนี้:

- รันหนึ่งคำสั่งของระบบ ปฏิบัติการ
- ทำซ้ำคำสั่ง ระบบปฏิบัติการ
- รันคำสั่งระบบปฏิบัติการ หลายคำสั่ง

### เมื่อต้องการรันหนึ่งคำสั่งของระบบปฏิบัติการ

พิมพ์คำสั่งย่อยต่อไปนี้:

!Command

โดยที่พารามิเตอร์ *Command* ระบุ คำสั่งระบบปฏิบัติการโดยทั่วไปถูกป้อนที่พร้อมต์

คำสั่งรันและแสดงเอาต์พุต หลังจากคำสั่งสมบูรณ์ เอดิเตอร์จะแสดง! (เครื่องหมาย ตกใจ)

### เมื่อต้องการทำซ้ำคำสั่งระบบปฏิบัติการ

พิมพ์คำสั่งย่อยต่อไปนี้:

!

คำสั่งระบบปฏิบัติการที่รันก่อนหน้านั้นรันและ แสดงเอาต์พุต หลังจากคำสั่งสมบูรณ์ เอดิเตอร์จะแสดง! (เครื่องหมาย ตกใจ)

### เมื่อต้องการรันคำสั่งระบบปฏิบัติการหลายคำสั่ง

1. พิมพ์คำสั่งย่อยต่อไปนี้เพื่อแสดงพร้อมๆ ระบบปฏิบัติการ:

!sh

2. พิมพ์คำสั่งระบบปฏิบัติการ

3. กด Enter เพื่อรันคำสั่งและแสดงเอาต์พุต

4. ทำซ้ำขั้นตอนที่ 2 และ 3 เพื่อรันคำสั่งระบบปฏิบัติการ เพิ่มเติม

5. กด Ctrl+D เพื่อกลับไปโหมดคำสั่ง เอดิเตอร์แสดง! (เครื่องหมาย ตกใจ)

### การออกจากเอดิเตอร์ ed

ไอเท็ม คำอธิบาย

q คำสั่งย่อย q (quit) ออกจากเอดิเตอร์ ed หลังจากตรวจสอบว่าบัฟเฟอร์ได้ถูกบันทึกไปที่ไฟล์หลังจากการเปลี่ยนแปลง ข้อมูลล่าสุดแล้วหรือไม่ ถ้าบัฟเฟอร์ยังไม่ได้ถูกบันทึกไปที่ไฟล์ คำสั่งย่อย q แสดงข้อความ? ((เครื่องหมายคำถาม) ป้อนคำสั่งย่อย q อีกครั้งเพื่อออกจากเอดิเตอร์ ed เท่านั้น การเปลี่ยนแปลงกับไฟล์ปัจจุบันจะสูญหาย

Q คำสั่งย่อย Q (Quit) ออกจากเอดิเตอร์ ed โดยไม่ตรวจสอบว่ามีเปลี่ยนแปลงหรือไม่ ตั้งแต่ที่บัฟเฟอร์ถูกบันทึก ไปที่ไฟล์ การเปลี่ยนแปลงที่ทำกับบัฟเฟอร์ ตั้งแต่การบันทึกล่าสุดจะสูญหาย

### เมื่อต้องการออกหลังจากการตรวจสอบการแก้ไข

1. พิมพ์คำสั่งย่อยต่อไปนี้:

q

2. ถ้าเอดิเตอร์ ed แสดง?, ให้พิมพ์หนึ่งในคำสั่งย่อยต่อไปนี้:

เมื่อต้องการ บันทึกการเปลี่ยนแปลงก่อนการออก ให้พิมพ์:

w

แล้วกด Enter

เมื่อต้องการออกโดยไม่บันทึกการเปลี่ยนแปลง ให้พิมพ์:

q

3. กด Enter

### เมื่อต้องการออกและละเว้นการแก้ไข

1. พิมพ์คำสั่งย่อยต่อไปนี้:

Q

2. กด Enter การเปลี่ยนแปลงที่ทำกับบัฟเฟอร์ ตั้งแต่การบันทึกล่าสุดจะสูญหาย

## การร้องขอวิธีใช้

ไอเท็ม คำอธิบาย

**h** คำสั่งย่อย **h** (help) มีข้อความวิธีใช้ย่อสำหรับ? การวินิจฉัยหรือข้อความแสดงความผิดพลาด ที่แสดงล่าสุด

**H** คำสั่งย่อย **H** (Help) ทำให้เอดิเตอร์ **ed** แสดงข้อความวิธีใช้สำหรับข้อความวินิจฉัย? ที่ตามมาทั้งหมด คำสั่งย่อย **H** ย่ออธิบาย? ก่อนหน้านี้ ถ้ามี คำสั่งย่อย **H** เป็นอีกทางเลือกในการเปิดและปิดโหมดนี้ คำเริ่มต้นคือปิด

### เมื่อต้องการสตา์ทหรือหยุดการแสดงข้อความวิธีใช้

พิมพ์คำสั่งย่อยต่อไปนี้:

**H**

ข้อความวิธีใช้ถูกแสดงหรือไม่ถูกแสดงสำหรับการตอบสนอง? จากเอดิเตอร์ **ed** ขึ้นกับการตั้งค่าก่อนหน้านี้

### เมื่อต้องการแสดงข้อความวิธีใช้ล่าสุด

พิมพ์คำสั่งย่อยต่อไปนี้:

**h**

ข้อความวิธีใช้ถูกแสดงสำหรับการตอบกลับ? จาก เอดิเตอร์ **ed**

## การสนับสนุนคลาสอักขระในเอดิเตอร์ **ed**

ในนิพจน์ *Patterns* มาตรฐาน นิพจน์ขอบเขตจับคู่ชุดของอักขระทั้งหมดที่อยู่ระหว่างสองอักขระในลำดับการเรียงของโลแคล ปัจจุบัน ไวยากรณ์ของ นิพจน์ขอบเขตเป็นดังนี้:

[*character-character*]

อักขระแรกต้องน้อยกว่าหรือเท่ากับอักขระ ที่สองในลำดับการเรียง สำหรับตัวอย่าง [a-c] ตรง กับอักขระ a, b หรือ c ในโลแคล En\_US

นิพจน์ขอบเขตโดยทั่วไปถูกใช้เพื่อจับคู่คลาสอักขระ ตัวอย่าง [0-9] ถูกใช้เพื่อเฉลี่ยตัวเลขทั้งหมด และ [a-zA-Z] ถูกใช้เพื่อเฉลี่ยตัวอักษรทั้งหมด ซึ่งฟอร์มอาจสร้างผลที่ไม่คาดคิด เมื่อขอบเขต ถูกแปลตามลำดับการเรียงในโลแคลปัจจุบัน

แทนการใช้ฟอร์มนำหน้าให้ใช้นิพจน์คลาสอักขระ ภายใน [] (วงเล็บเหลี่ยม) เพื่อจับคู่อักขระ ระบบแปล นิพจน์ชนิดนี้ตาม นิยามคลาสอักขระใน โลแคลปัจจุบัน อย่างไรก็ตาม คุณไม่สามารถใช้นิพจน์คลาสอักขระในนิพจน์ ขอบเขต

ไวยากรณ์ของนิพจน์คลาสอักขระเป็นดังนี้:

[*:CharacterClass*:]

นั่นคือ วงเล็บเหลี่ยมซ้าย โคลอน ชื่อของคลาสอักขระ โคลอน และวงเล็บเหลี่ยมขวา

คลาสอักขระต่อไปนี้สนับสนุนใน โลแคลทั้งหมด:

ไอเท็ม	คำอธิบาย
[ :upper: ]	ตัวอักษรตัวพิมพ์ใหญ่
[ :lower: ]	ตัวอักษรตัวพิมพ์เล็ก
[ :alpha: ]	ตัวอักษรตัวพิมพ์ใหญ่และตัวพิมพ์เล็ก
[ :digit: ]	ตัวเลข
[ :alnum: ]	อักขระแบบตัวอักษรผสมตัวเลข
[ :xdigit: ]	ตัวเลขฐานสิบหก
[ :punct: ]	อักขระเครื่องหมายวรรคตอน (ไม่ใช่อักขระควบคุมหรือตัวอักษรผสมตัวเลข)
[ :space: ]	Space, แท็บ, ปัดแคร่, ขึ้นบรรทัดใหม่, แท็บแนวตั้ง หรืออักขระ ป้อนกระดาษ
[ :print: ]	อักขระที่พิมพ์ได้รวมทั้งช่องว่าง
[ :graph: ]	อักขระที่พิมพ์ได้ไม่รวมช่องว่าง
[ :cntrl: ]	อักขระควบคุม
[ :blank: ]	อักขระช่องว่างและแท็บ

วงเล็บเหลี่ยมเป็นส่วนหนึ่งของนิยามคลาสอักขระ เมื่อต้องการจับคู่ตัวอักษร ASCII ตัวพิมพ์ใหญ่หรือตัวเลข ASCII ให้ใช้ นิพจน์ทั่วไปต่อไปนี้:

```
[[:upper:]] [[:digit:]]
```

อย่าใช้นิพจน์ [A-Z0-9]

โลแคลอาจสนับสนุนคลาสอักขระเพิ่มเติม

อักขระ newline เป็นส่วนหนึ่งของคลาสอักขระ [ :space: ] แต่จะไม่ถูกจับคู่โดยคลาสอักขระนี้ อักขระ newline อาจถูกจับคู่ เฉพาะโดยอักขระการค้นหาค่าพิเศษ \$ (เครื่องหมายดอลลาร์) และ ^ (แคโร็ต)

## สถานะออก

คำสั่ง ed และ red ส่งคืนค่า exit ต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง edit”

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง rsh

คำสั่ง sed

คำสั่ง view

## คำสั่ง edit

### วัตถุประสงค์

จัดเตรียมเอดิเตอร์รายบรรทัดอย่างง่ายสำหรับผู้ใช้นิพจน์

## ไวยากรณ์

`edit [-r] [File ...]`

### คำอธิบาย

คำสั่ง `edit` สตาร์ทเอดิเตอร์รายการบรรทัดที่ออกแบบสำหรับ ผู้ใช้เริ่มต้น เวอร์ชันแบบธรรมดาของเอดิเตอร์ `ex` เอดิเตอร์ `edit` เป็นของ กลุ่มเอดิเตอร์ที่อยู่ในเอดิเตอร์ `ed`, เอดิเตอร์ `ex`, และเอดิเตอร์ `vi` การทราบข้อมูลเกี่ยวกับเอดิเตอร์ `edit` สามารถช่วยให้คุณเรียนรู้คุณลักษณะระดับสูงยิ่งขึ้น ของเอดิเตอร์อื่น เมื่อต้องการแก้ไขเนื้อหาของไฟล์ ให้ป้อน:

```
edit File
```

เมื่อไฟล์ที่ระบุโดยพารามิเตอร์ `File` กำหนดชื่อ ไฟล์ที่มีอยู่ คำสั่ง `edit` คัดลอกไฟล์ไปที่บัฟเฟอร์ และแสดงจำนวนบรรทัดและอักขระภายใน จากนั้นจะแสดง : (โคลอน) พร้อมต์เพื่อแสดงว่าพร้อมที่จะอ่านคำสั่งย่อย จากอินพุตมาตรฐาน

ถ้าไฟล์ที่ระบุในพารามิเตอร์ `File` ไม่มีอยู่ คำสั่ง `edit` จะรับรู้ข้อมูลนี้ และสร้างไฟล์ใหม่ คุณสามารถระบุได้มากกว่าหนึ่งชื่อไฟล์ สำหรับพารามิเตอร์ `File` ในกรณีที่คำสั่ง `edit` คัดลอกไฟล์แรกลงในบัฟเฟอร์และเก็บชื่อไฟล์ที่เหลือ ในรายการอาร์กิวเมนต์ เพื่อใช้ในภายหลัง เอดิเตอร์ `edit` ไม่ได้ทำการเปลี่ยนแปลงกับไฟล์ที่แก้ไขจนกว่าคุณจะใช้คำสั่งย่อย `w` ในการเขียนการเปลี่ยนแปลง

เอดิเตอร์ `edit` ทำงานในหนึ่งในสองโหมดดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
โหมดคำสั่ง	รู้จักและรันคำสั่งย่อยเอดิเตอร์ <code>edit</code> เมื่อคุณสตาร์ทเอดิเตอร์ <code>edit</code> จะทำงานในโหมดคำสั่ง เมื่อต้องการเข้าสู่โหมดคำสั่งในภายหลัง ให้ป้อนเฉพาะ . (จุด) ที่จุดเริ่มต้นบรรทัด
โหมดอินพุตข้อความ	อนุญาตให้คุณป้อนข้อความในบัฟเฟอร์เอดิเตอร์ <code>edit</code> เข้าสู่โหมด อินพุตข้อความโดยใช้คำสั่งย่อย <code>append (a)</code> , คำสั่งย่อย <code>change (c)</code> หรือคำสั่งย่อย <code>insert (i)</code> เมื่อต้องการจบการทำงานโหมดอินพุตข้อความ ให้ป้อนเพียง . (จุด) ที่จุดเริ่มต้นบรรทัด

### แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-r</code>	กู้คืนไฟล์จากการถูกแก้ไขหลังจากเอดิเตอร์หรือระบบทำงานผิดพลาด

### การค้นหาบรรทัดในไฟล์

เอดิเตอร์ `edit` ใช้ชนิดของแอดเดรสสามชนิดดังต่อไปนี้:

- แอดเดรสหมายเลขบรรทัด
- แอดเดรสตำแหน่งสัมพัทธ์
- แอดเดรสรูปแบบ

#### แอดเดรสหมายเลขบรรทัด

แอดเดรสหมายเลขบรรทัดระบุบรรทัดภายในไฟล์ตามหมายเลขบรรทัดหรือ ชื่อสัญลักษณ์ เมธอดนี้เป็นวิธีที่ง่ายที่สุดในการแอดเดรสบรรทัด

เมื่อต้องการแอดเดรสบรรทัดแรกตามชื่อสัญลักษณ์ ให้ป้อน:

เมื่อต้องการแอดเดรสบรรทัดสุดท้ายตามชื่อสัญลักษณ์ให้ป้อน:

\$

คุณยังสามารถระบุช่วงของบรรทัดโดยการแยกหมายเลขบรรทัดหรือ การกำหนดแอดเดรสเชิงสัญลักษณ์ด้วยคอมมาหรือเซมิโคลอน แอดเดรสที่สองต้องอ้างถึง บรรทัดที่ต่อจากบรรทัดแรกในช่วง

ตัวอย่างเช่น:

1,5

แอดเดรสบรรทัด 1 ถึง 5

.,\$

แอดเดรสบรรทัดแรกจนถึงบรรทัดสุดท้าย

**แอดเดรสตำแหน่งสัมพันธ์**

เอดิเตอร์ edit สามารถแอดเดรสบรรทัดตามตำแหน่งสัมพันธ์กับบรรทัด ปัจจุบัน แอดเดรสที่เริ่มต้นด้วยพารามิเตอร์ -Number หรือ +Number แอดเดรสบรรทัดจำนวนบรรทัดที่ระบุ ก่อนหรือหลังจากบรรทัดปัจจุบัน ตามลำดับ

ตัวอย่างเช่น:

+8

แอดเดรส 8 บรรทัดหลังจากบรรทัดปัจจุบัน

คุณยังสามารถแอดเดรสความสัมพันธ์ของบรรทัดกับบรรทัดแรกหรือบรรทัดสุดท้ายโดยใช้ชื่อเชิงสัญลักษณ์โดยการรวมกันของแอดเดรส -Number หรือ +Number

ตัวอย่างเช่น:

+.3

แอดเดรส 3 บรรทัดหลังจากบรรทัดแรก และ:

\$-10

แอดเดรส 10 บรรทัดก่อนบรรทัดสุดท้าย

**แอดเดรสรูปแบบ**

คุณสามารถระบุบรรทัดแอดเดรสโดยการค้นหาบัพเฟอรรูปแบบ จำเพาะ เอดิเตอร์ edit ค้นหาไปข้างหน้าหรือย้อนกลับและหยุดที่บรรทัดแรก ที่มีข้อมูลตรงกับพารามิเตอร์ Pattern ถ้าจำเป็น การค้นหาจะวนผ่านจุดสิ้นสุดหรือจุดเริ่มต้นของบัพเฟอรรจนกว่า จะพบข้อมูลที่ตรงกันหรือกลับไปบรรทัดปัจจุบัน

เมื่อต้องการค้นหาไปข้างหน้าให้ป้อน:

/Pattern/

เมื่อต้องการค้นย้อนกลับให้ป้อน:

?Pattern?

คุณยังสามารถระบุช่วงของบรรทัดโดยการแยกพารามิเตอร์ *Pattern* ด้วยคอมมาหรือเซมิโคลอน แอดเดรสที่สองต้องอ้างถึงบรรทัดที่ต่อจากบรรทัดแรกในช่วง

ตัวอย่างเช่น:

Pattern,Pattern

อักขระต่อไปนี้มีคําหมายพิเศษเมื่อใช้เป็นส่วหนึ่งของพารามิเตอร์ *Pattern*:

ไอเท็ม คําอธิบาย

^ จับจุดเริ่มต้นของบรรทัดเมื่อใช้เป็นอักขระแรกของพารามิเตอร์ *Pattern*

\$ จับจุดสิ้นสุดของบรรทัดเมื่อใช้เป็นอักขระสุดท้ายของพารามิเตอร์ *Pattern*

## การใช้คําสั่งย่อเอดิเตอร์ edit

คําสั่งย่อเอดิเตอร์ edit มีผลกับบรรทัดปัจจุบัน ซึ่งแสดงโดย . (จุด) เมื่อคุณเริ่มเอดิเตอร์ edit บรรทัด ปัจจุบันจะเป็นบรรทัดสุดท้ายในบัฟเฟอร์ เมื่อบัฟเฟอร์ถูกแก้ไข บรรทัดปัจจุบัน จะเปลี่ยนเป็นบรรทัดสุดท้ายที่ได้รับผลโดยคําสั่งย่อ เมื่อต้องการทำงานกับ ส่วนต่างๆ ของไฟล์ คุณต้องทราบวิธีในการ ค้นหา บรรทัดปัจจุบัน และ วิธีในการแอดเดรสบรรทัดที่ต่างกัน ในไฟล์

คุณสามารถใช้คําสั่งย่อเอดิเตอร์ edit ในการดำเนินงานดังต่อไปนี้:

- เพิ่มข้อความ
- เปลี่ยนชื่อของไฟล์ปัจจุบัน
- เปลี่ยนข้อความ
- ลบข้อความ
- แสดงชื่อและสถานะของชื่อไฟล์ปัจจุบัน
- แสดงข้อความและค้นหาบรรทัดปัจจุบัน
- แก้ไขไฟล์เพิ่มเติม
- จบการทำงานและออกจากเอดิเตอร์ edit
- ทำการเปลี่ยนแปลงระดับโกลบอล
- ย้ายหรือคัดลอกข้อความ
- บันทึกไฟล์หลังจากระบบระบบขัดข้อง
- บันทึกข้อความ
- แทนที่ข้อความ
- ยกเลิกการเปลี่ยนแปลง

## การเพิ่มข้อความ

ในคําสั่งย่อต่อไปนี้พารามิเตอร์ *Address* เป็นทางเลือก ถ้าคุณระบุแอดเดรสไม่ต้องพิมพ์วงเล็บ คุณสามารถใช้คําสั่งย่อแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

## ไอเท็ม

[Address]append (a) Text

### คำอธิบาย

ต่อท้ายข้อความที่คุณพิมพ์หลังบรรทัดปัจจุบัน ถ้าคุณไม่ได้ระบุ พารามิเตอร์ Address คุณอาจจำเป็นต้อง ค้นหาบรรทัดปัจจุบัน หรือ ระบุแอดเดรส ถ้า คุณไม่ได้อยู่ในตำแหน่งที่ต้องการในบัฟเฟอร์

ถ้าคุณระบุแอดเดรสคำสั่งย่อย a ผนวก ข้อความหลังจากบรรทัดที่ระบุ ถ้าคุณระบุ 0 แอดเดรส, คำสั่งย่อย a วางข้อความที่จุดเริ่มต้นของบัฟเฟอร์

พิมพ์ข้อความ กดคีย์ Enter ที่จุดสิ้นสุดของแต่ละ บรรทัด เมื่อคุณได้ป้อนข้อความทั้งหมดแล้ว ให้พิมพ์ . (จุด) จุดเดียวที่ตำแหน่งเริ่มต้นของบรรทัดเพื่อสิ้นสุดโหมดอินพุตและกลับไปโหมด คำสั่ง คุณสามารถใช้คำสั่งย่อย 1, \$p เพื่อแสดงเนื้อหา ทั้งหมดของบัฟเฟอร์

**หมายเหตุ:** คำสั่งย่อย a ต่างจากคำสั่งย่อย i ที่ตำแหน่งการวางข้อความ

[Address]insert (i) Text.

แทรกข้อความก่อนบรรทัดปัจจุบัน ถ้าคุณไม่ได้ระบุ พารามิเตอร์ Address คุณอาจจำเป็นต้อง ค้นหาบรรทัดปัจจุบัน หรือ ระบุแอดเดรส ถ้า คุณไม่ได้อยู่ในตำแหน่งที่ต้องการในบัฟเฟอร์

ถ้าคุณระบุแอดเดรสคำสั่งย่อย i แทรก ข้อความก่อนบรรทัดที่ระบุ คุณไม่สามารถระบุ 0 แอดเดรส

พิมพ์ข้อความของคุณ กดคีย์ Enter ที่จุดสิ้นสุดของแต่ละบรรทัด เมื่อคุณได้ป้อน ข้อความของคุณทั้งหมดแล้ว ให้พิมพ์ . (จุด) จุดเดียวที่ตำแหน่งเริ่มต้นของบรรทัดเพื่อสิ้นสุดโหมดอินพุตและกลับไปโหมด คำสั่ง คุณสามารถใช้คำสั่งย่อย 1, \$p เพื่อแสดงเนื้อหา ทั้งหมดของบัฟเฟอร์

**หมายเหตุ:** คำสั่งย่อย i ต่างจากคำสั่งย่อย a ที่ตำแหน่งการวางข้อความ

## การเปลี่ยนแปลง Name ของไฟล์ Current

### ไอเท็ม

file File

### คำอธิบาย

เปลี่ยนชื่อไฟล์ปัจจุบันเป็นชื่อที่ระบุโดยพารามิเตอร์ File เอดิเตอร์ edit ไม่พิจารณาว่าไฟล์นี้จะถูกแก้ไข

## การเปลี่ยนข้อความ

ในคำสั่งย่อยต่อไปนี้พารามิเตอร์ Address เป็นทางเลือก ถ้าคุณระบุแอดเดรส ไม่ต้องพิมพ์วงเล็บ คุณสามารถ ใช้คำสั่งย่อยแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

### ไอเท็ม

[Address1,Address2]change (c).

### คำอธิบาย

แทนที่บรรทัดปัจจุบันด้วยข้อความที่คุณพิมพ์ ถ้าคุณไม่ได้ระบุพารามิเตอร์ Address คุณอาจจำเป็นต้อง ค้นหาบรรทัดปัจจุบัน หรือ ระบุแอดเดรส ถ้า คุณไม่ได้อยู่ในตำแหน่งที่ต้องการในบัฟเฟอร์

Text

ถ้าคุณระบุแอดเดรส คำสั่งย่อย c แทนที่ บรรทัดแอดเดรส คุณยังสามารถระบุช่วงของบรรทัดโดยแยก แอดเดรสด้วยคอมมา

พิมพ์ข้อความของคุณ กดคีย์ Enter ที่จุดสิ้นสุดของแต่ละบรรทัด เมื่อคุณได้ป้อน ข้อความของคุณทั้งหมดแล้ว ให้พิมพ์ . (จุด) จุดเดียวที่ตำแหน่งเริ่มต้นของบรรทัดเพื่อสิ้นสุดโหมดอินพุตและกลับไปโหมด คำสั่ง คุณสามารถใช้คำสั่งย่อย 1, \$p เพื่อแสดงเนื้อหา ทั้งหมดของบัฟเฟอร์ บรรทัดอินพุต สุดท้ายกลายเป็นบรรทัดปัจจุบัน

## การลบข้อความ

ในคำสั่งย่อยต่อไปนี้พารามิเตอร์ Address และ Buffer เป็นทางเลือก ถ้าคุณระบุแอดเดรส หรือบัฟเฟอร์ ไม่ต้องพิมพ์วงเล็บ คุณสามารถ ใช้คำสั่งย่อยแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

## ไอเท็ม

[Address1,Address2] delete [Buffer] (d)

### คำอธิบาย

ลบบรรทัดปัจจุบัน ถ้าคุณไม่ได้รับพารามิเตอร์ Address คุณอาจจำเป็นต้อง ค้นหาบรรทัดปัจจุบัน หรือ ระบุแอดเดรส ถ้า คุณไม่ได้อยู่ในตำแหน่งที่ต้องการในบัฟเฟอร์

ถ้าคุณระบุแอดเดรส คำสั่งย่อย d ลบ บรรทัดแอดเดรส คุณยังสามารถระบุช่วงของบรรทัดโดยแยก แอดเดรสด้วยคอมมา บรรทัดต่อบรรทัดที่ลบล่าสุดกลายเป็น บรรทัดปัจจุบัน

ถ้าคุณระบุบัฟเฟอร์โดยกำหนด อักษรตัวพิมพ์เล็กจาก a ถึง z, เอดิเตอร์ edit บันทึกบรรทัดที่แอดเดรสในบัฟเฟอร์นั้น ถ้าคุณระบุอักษรตัวพิมพ์ใหญ่ เอดิเตอร์ ed ผนวก บรรทัดเข้ากับบัฟเฟอร์นั้น คุณสามารถใช้คำสั่งย่อย pu เพื่อนำบรรทัดที่ลบกลับมาสู่บัฟเฟอร์

## การแสดงชื่อไฟล์ และสถานะ Current

ในคำสั่งย่อยต่อไปนี้ คุณสามารถ ใช้คำสั่งย่อยแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

### ไอเท็ม

file (f)

### คำอธิบาย

แสดงชื่อไฟล์ปัจจุบันตามด้วยข้อมูลที่เกี่ยวข้องต่อไปนี้:

- ไฟล์ถูกแก้ไขตั้งแต่มีการใช้คำสั่งย่อย w ล่าสุดหรือไม่
- หมายเลขบรรทัด Current
- จำนวนบรรทัดในบัฟเฟอร์
- เปอร์เซ็นต์ของบัฟเฟอร์ที่ระบุตำแหน่งบรรทัดปัจจุบัน

## การแสดงความละเอียดและการค้นหาบรรทัด ปัจจุบัน

ในคำสั่งย่อยต่อไปนี้พารามิเตอร์ Address เป็นทางเลือก ถ้าคุณระบุแอดเดรส ไม่ต้องพิมพ์วงเล็บ คุณสามารถ ใช้คำสั่งย่อยแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

### ไอเท็ม

[Address1,Address2]number (nu)

### คำอธิบาย

แสดงบรรทัดที่แอดเดรสหรือบรรทัดที่นำหน้าด้วย หมายเลขบรรทัดบัฟเฟอร์ ถ้าคุณไม่ระบุพารามิเตอร์ Address , คำสั่งย่อย nu แสดงบรรทัดและหมายเลขปัจจุบัน

ถ้าคุณระบุแอดเดรส คำสั่งย่อย nu แสดง บรรทัดแอดเดรส คุณยังสามารถระบุช่วงของบรรทัดโดยแยกแอดเดรสด้วยคอมมา บรรทัดสุดท้ายที่แสดง กลายเป็นบรรทัดปัจจุบัน

[Address1,Address2]print (p)

แสดงบรรทัดที่แอดเดรส ถ้าคุณไม่ระบุ พารามิเตอร์ Address , คำสั่งย่อย p แสดงบรรทัดปัจจุบัน

[Address]=

ถ้าคุณ ระบุแอดเดรส คำสั่งย่อย p แสดง บรรทัดแอดเดรส คุณยังสามารถระบุช่วงของบรรทัดโดยแยกแอดเดรสด้วยคอมมา บรรทัดสุดท้ายที่แสดงกลายเป็นบรรทัดปัจจุบัน

[Address]z

แสดงหมายเลขบรรทัดของบรรทัดที่แอดเดรส ถ้าคุณไม่ระบุพารามิเตอร์ Address คำสั่งย่อย = แสดงหมายเลขบรรทัดของบรรทัดปัจจุบัน

[Address]z-

แสดงจอภาพของข้อความเริ่มต้นด้วยบรรทัดที่แอดเดรส ถ้าพารามิเตอร์ Address ไม่ถูกระบุ คำสั่งย่อย z แสดงจอภาพของข้อความเริ่มต้นด้วยบรรทัดปัจจุบัน

[Address]z.

แสดงจอภาพของข้อความโดยมีบรรทัดที่แอดเดรสอยู่ทางด้านล่าง ถ้าพารามิเตอร์ Address ไม่ถูกระบุ คำสั่งย่อย z- แสดงจอภาพของข้อความโดยมีบรรทัดปัจจุบัน อยู่ทางด้านล่าง

แสดงจอภาพของข้อความโดยมีบรรทัดที่แอดเดรสตรงกลาง ถ้าพารามิเตอร์ Address ไม่ถูกระบุ คำสั่งย่อย z. แสดงจอภาพของข้อความโดยมีบรรทัดปัจจุบัน อยู่ตรงกลาง

## การแก้ไขไฟล์เพิ่มเติม

ในคำสั่งย่อยต่อไปนี้ คุณสามารถ ใช้คำสั่งย่อยแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

ไอเท็ม edit File (e)	<p><b>คำอธิบาย</b> เริ่มเซสชันการแก้ไขบนไฟล์ใหม่ที่ระบุโดยพารามิเตอร์ <i>File</i> ชั้นแรกเอดิเตอร์จะตรวจสอบเพื่อดูว่า บัฟเฟอร์ถูกแก้ไข เมื่อมีการใช้คำสั่งย่อย write (w) ครั้งล่าสุดหรือไม่</p> <p>ถ้าไฟล์ถูกแก้ไขในคำสั่งย่อย w ล่าสุด เอดิเตอร์ edit ส่งค่าเตือนและยกเลิกคำสั่งย่อย e มิฉะนั้น เอดิเตอร์ edit จะลบเนื้อหาของเอดิเตอร์บัฟเฟอร์ ทำไฟล์ที่กำหนดชื่อให้เป็นไฟล์ปัจจุบัน และแสดงชื่อไฟล์ใหม่</p> <p>หลังจากประกันว่าไฟล์นี้สามารถถูกแก้ไข เอดิเตอร์ edit จะอ่านไฟล์ลงในบัฟเฟอร์ ถ้าเอดิเตอร์ edit อ่านไฟล์โดยไม่มีข้อผิดพลาด จะแสดงจำนวนบรรทัดและอักขระที่อ่าน บรรทัดสุดท้ายที่อ่านกลายเป็นบรรทัดปัจจุบัน คัดลอกไฟล์ต่อไปที่มีชื่อในรายการอาร์กิวเมนต์บรรทัดคำสั่งไปที่ บัฟเฟอร์เพื่อการแก้ไข</p>
next (n)	

## สิ้นสุดและออกจากเอดิเตอร์ edit

ในคำสั่งย่อยต่อไปนี้ คุณสามารถใช้คำสั่งย่อยแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

ไอเท็ม quit (q)	<p><b>คำอธิบาย</b> สิ้นสุดเซสชันการแก้ไขหลังจากใช้คำสั่งย่อย write (w) ถ้าคุณได้แก้ไขบัฟเฟอร์ และไม่ได้เขียนการเปลี่ยนแปลง เอดิเตอร์ edit แสดงข้อความแจ้งเตือน และไม่จบเซสชันการแก้ไข</p>
quit! (q!)	<p>สิ้นสุดเซสชันการแก้ไข ละเว้นการเปลี่ยนแปลงที่ทำกับบัฟเฟอร์ ตั้งแต่คำสั่งย่อย w สุดท้าย</p>

## การทำารเปลี่ยนแปลงระดับโกลบอล

ในคำสั่งย่อยต่อไปนี้พารามิเตอร์ *Address* เป็นทางเลือก ถ้าคุณระบุแอดเดรส ไม่ต้องพิมพ์วงเล็บ คุณสามารถใช้คำสั่งย่อยแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

ไอเท็ม [ <i>Address1</i> , <i>Address2</i> ]global/ <i>Pattern</i> / <i>SubcommandList</i> (g)	<p><b>คำอธิบาย</b> ทำเครื่องหมายแต่ละบรรทัดที่แอดเดรสซึ่งตรงกับพารามิเตอร์ <i>Pattern</i> จากนั้นเอดิเตอร์ edit ดำเนินการรายการคำสั่งย่อย ที่ระบุในพารามิเตอร์ <i>SubcommandList</i> ในแต่ละบรรทัดที่ทำเครื่องหมาย</p> <p>ถ้าคุณไม่ระบุพารามิเตอร์ <i>Address</i> คำสั่งย่อย g ทำงานกับบรรทัดปัจจุบัน คุณอาจจำเป็นต้อง ค้นหาบรรทัดปัจจุบัน หรือ ระบุแอดเดรส ถ้า คุณไม่ได้อยู่ในตำแหน่งที่ต้องการในบัฟเฟอร์</p> <p>ถ้าคุณระบุแอดเดรส คำสั่งย่อย g ทำงานกับ บรรทัดที่แอดเดรส คุณยังสามารถระบุช่วงของบรรทัดโดยแยก แอดเดรสด้วยคอมมา</p> <p>คำสั่งย่อยเดี่ยวหรือ คำสั่งย่อยแรกในรายการคำสั่งย่อยแสดงในบรรทัดเดียวกับคำสั่งย่อย g คำสั่งย่อยที่เหลือต้องแสดงในบรรทัดแยก โดยที่แต่ละบรรทัด (ยกเว้นบรรทัดสุดท้าย) สิ้นสุดด้วย \ (backslash) คำสั่งย่อยดีฟอลต์คือคำสั่งย่อย print (p)</p> <p>รายการคำสั่งย่อยสามารถ มีคำสั่งย่อย append (a), คำสั่งย่อย insert (i), คำสั่งย่อยและคำสั่งย่อย change (c) และอินพุตที่เกี่ยวข้อง ในกรณีนี้ ถ้าช่วงสิ้นสุดอยู่ในบรรทัดสุดท้าย ของรายการคำสั่ง คุณสามารถข้ามได้</p> <p><b>หมายเหตุ:</b> คำสั่งย่อย undo (u) และคำสั่งย่อย g ไม่สามารถแสดงในรายการคำสั่งย่อย</p>
---	---

## การทำย้ายหรือคัดลอกข้อความ

ในคำสั่งย่อยต่อไปนี้พารามิเตอร์ *Address1* และ *Address2* เป็นทางเลือก ถ้าคุณระบุแอดเดรส ไม่ต้องพิมพ์วงเล็บ คุณต้องระบุพารามิเตอร์ *Address3* คุณสามารถใช้คำสั่งย่อยแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

ไอเท็ม [Address1,Address2]move Address3 (m)	คำอธิบาย ย้ายบรรทัดปัจจุบันหลังจากบรรทัดที่ระบุโดย พารามิเตอร์ Address3 ถ้าคุณไม่ได้ระบุแอดเดรส หรือช่วงแอดเดรส คุณอาจจำเป็นต้อง ค้นหาบรรทัดปัจจุบัน หรือ ระบุแอดเดรส ถ้า คุณไม่ได้อยู่ในตำแหน่งที่ถูกต้องในบัพเฟอร์
[Address1,Address2]yank [Buffer] (ya)	ถ้าคุณ ระบุแอดเดรส คำสั่งย่อย m ย้าย บรรทัดที่แอดเดรส คุณยังสามารถระบุแอดเดรสของบรรทัดโดยแยก แอดเดรสด้วยคอมมา บรรทัดที่ย้ายบรรทัดแรกกลายเป็นบรรทัดปัจจุบัน คัดลอกบรรทัดที่ระบุไปยัง Buffer พารามิเตอร์เป็นทางเลือกที่ระบุโดยอักขระตัวอักษรเดี่ยว a ถึง z คุณสามารถใช้ คำสั่งย่อย pu เพื่อวางบรรทัดเหล่านี้ลงในไฟล์อื่น
[Address]put [Buffer] (pu)	เรียกเนื้อหาของพารามิเตอร์ Buffer ที่ระบุและวางตำแหน่งหลังจากบรรทัดปัจจุบัน ถ้าคุณไม่ได้ ระบุแอดเดรส คุณอาจจำเป็นต้อง ค้นหาบรรทัดปัจจุบัน หรือ ระบุแอดเดรส ถ้า คุณไม่ได้อยู่ในตำแหน่งที่ถูกต้องในบัพเฟอร์
	ถ้าคุณระบุแอดเดรส คำสั่งย่อย pu เรียกคืน เนื้อหาของบัพเฟอร์ที่ระบุและวางตำแหน่งหลังจากบรรทัดที่แอดเดรส ถ้าคุณไม่ระบุพารามิเตอร์ Buffer คำสั่งย่อย pu เรียกคืนข้อความที่ลบหรือคัดลอกล่าสุด
	คุณสามารถใช้คำสั่งย่อย pu กับคำสั่งย่อย delete (d) เพื่อย้ายบรรทัดภายในไฟล์หรือ ด้วยคำสั่งย่อย yank (ya) เพื่อทำซ้ำบรรทัดระหว่างไฟล์
	คุณไม่สามารถใช้คำสั่งย่อย pu และ ya ภายในแมโคร

### การบันทึกไฟล์หลังจากระบบขัดข้อง

ไอเท็ม preserve	คำอธิบาย บันทึกบัพเฟอร์เอดิเตอร์ปัจจุบันเหมือนกับระบบเฟิงทำงานผิดพลาด ใช้คำสั่งย่อยนี้เมื่อคำสั่งย่อย write (w) มีข้อผิดพลาด และคุณไม่ทราบวิธี บันทึกงานของคุณ ใช้คำสั่งย่อย recover เพื่อกู้คืน ไฟล์
recover File	กู้คืนไฟล์ที่ระบุโดยพารามิเตอร์ File จากพื้นที่บันทึกระบบ ใช้คำสั่งย่อยนี้หลังจากระบบขัดข้องหรือหลังจากคำสั่งย่อย preserve

### การบันทึกข้อความ

ในคำสั่งย่อยต่อไปนี้พารามิเตอร์ Address เป็นทางเลือก ถ้าคุณระบุแอดเดรส ไม่ต้องพิมพ์วงเล็บ คุณสามารถ ใช้คำสั่งย่อยแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

ไอเท็ม [Address1,Address2]write [File] (w)	คำอธิบาย เขียนเนื้อหาทั้งหมดของบัพเฟอร์ไปที่ไฟล์ ที่ระบุโดยพารามิเตอร์ File ถ้าคุณไม่ได้ระบุ แอดเดรส
	ถ้าคุณระบุแอดเดรสคำสั่งย่อย w เขียนบรรทัดที่แอดเดรสไปที่ไฟล์ที่ระบุ คุณยังสามารถระบุช่วงของบรรทัดโดยแยก แอดเดรสด้วยคอมมา เอดิเตอร์ edit แสดงจำนวนบรรทัดและอักขระที่เขียนข้อมูล
	ถ้าคุณไม่ระบุไฟล์ เอดิเตอร์ edit ใช้ชื่อไฟล์ปัจจุบัน ถ้าพารามิเตอร์ File ไม่มีอยู่ เอดิเตอร์จะสร้างขึ้นมา

### การแทนที่ข้อความ

ในคำสั่งย่อยต่อไปนี้พารามิเตอร์ Address เป็นทางเลือก ถ้าคุณระบุแอดเดรส ไม่ต้องพิมพ์วงเล็บ คุณสามารถ ใช้คำสั่งย่อยแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

## ไอเท็ม

[Address1,Address2] substitute/  
Pattern/Replacement/ (s)[Address1,  
Address2] substitute/Pattern/  
Replacement/g

## คำอธิบาย

แทนที่อินสแตนซ์แรกของพารามิเตอร์ *Pattern* ที่ระบุบนแต่ละบรรทัดที่แอดเดรส คุณสามารถแทนที่ทุกอินสแตนซ์ของพารามิเตอร์ *Pattern* โดยการเพิ่มคำสั่งย่อย **global (g)** ให้กับจุดสิ้นสุดของคำสั่งย่อย **s**

ถ้าคุณไม่ระบุแอดเดรส คำสั่งย่อย **s** ทำงานกับไฟล์ปัจจุบัน คุณอาจจำเป็นต้อง ค้นหาบรรทัดปัจจุบัน หรือ ระบุแอดเดรส ถ้า คุณไม่ได้อยู่ในตำแหน่งที่ถูกต้องในบัพเฟอร์ ถ้าคุณระบุแอดเดรส คำสั่งย่อย **s** ทำงานกับ บรรทัดที่แอดเดรส คุณยังสามารถระบุช่วงของบรรทัดโดยแยก แอดเดรสด้วยคอมมา

## การยกเลิกการเปลี่ยนแปลง

ในคำสั่งย่อยต่อไปนี้ คุณสามารถใช้คำสั่งย่อยแบบเต็มหรือตัวย่อ ซึ่งถูกแสดงในวงเล็บ

## ไอเท็ม

undo (u)

## คำอธิบาย

ย้อนกลับการเปลี่ยนแปลงที่ทำในบัพเฟอร์โดยคำสั่งย่อยการแก้ไข บัพเฟอร์ล่าสุด คุณไม่สามารถยกเลิกคำสั่งย่อย **write (w)** หรือคำสั่งย่อย **edit (e)**

หมายเหตุ: คำสั่งย่อย **global** ถือเป็นคำสั่งย่อยเดี่ยว ต่อคำสั่งย่อย **u**

## สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง ed หรือ red” ในหน้า 297

“คำสั่ง ex” ในหน้า 456

## ข้อมูลที่เกี่ยวข้อง:

vi หรือ vedit

---

## คำสั่ง edquota

### วัตถุประสงค์

แก้ไขโควตาผู้ใช้และกลุ่ม

### ไวยากรณ์

เมื่อต้องการแก้ไขโควตาผู้ใช้

```
edquota [-u] [-p Proto-UserName] UserName ...
```

เมื่อต้องการแก้ไขโควตากลุ่ม

```
edquota [-g] [-p Proto-GroupName] GroupName ... ]
```

### To Edit Change User or Group Grace Period

```
edquota -t [-u | -g]
```

## คำอธิบาย

คำสั่ง `edquota` สร้างและแก้ไขโควตาสำหรับระบบไฟล์ JFS สำหรับข้อมูลเกี่ยวกับวิธีการจัดการโควตาบนระบบไฟล์ JFS2 ดูที่คำสั่ง `j2edlimit` ใน *Commands Reference, Volume 3*

คำสั่ง `edquota` สร้างไฟล์ชั่วคราวที่มีโควตาดิסקปัจจุบันของแต่ละผู้ใช้และกลุ่ม ซึ่งจะกำหนดรายการของระบบไฟล์ด้วยโควตาที่กำหนดไว้แล้วจากไฟล์ `/etc/filesystems` คำสั่ง `edquota` ยังเรียกเอดิเตอร์ `vi` (หรือเอดิเตอร์ที่ระบุโดยตัวแปรสภาวะแวดล้อม `EDITOR`) บนไฟล์ชั่วคราวเพื่อที่โควตาสามารถถูกเพิ่มและแก้ไขได้

**หมายเหตุ:** ถ้าคุณระบุเอดิเตอร์ในตัวแปรสภาวะแวดล้อม `EDITOR` คุณต้องระบุชื่อพาธแบบเต็มของเอดิเตอร์

โควตาถูกรักษาแยกกันสำหรับแต่ละระบบไฟล์ เมื่อคุณสร้างหรือแก้ไขโควตาสำหรับผู้ใช้หรือกลุ่มโควตาใช้กับระบบไฟล์จำเพาะ โควตาดังกล่าวจะถูกเซตในแต่ละระบบไฟล์ที่คุณต้องการใช้โควตา

โดยดีฟอลต์ หรือเมื่อใช้กับแฟล็ก `-u`, คำสั่ง `edquota` แก้ไขโควตาของผู้ใช้ที่ระบุโดยพารามิเตอร์ `UserName` บนบรรทัดคำสั่ง เมื่อใช้กับแฟล็ก `-g`, คำสั่ง `edquota` แก้ไขของโควตาของกลุ่มที่ระบุโดยพารามิเตอร์ `GroupName` แฟล็ก `-p` ระบุผู้ใช้ prototypical (`UserName`) หรือกลุ่ม prototypical (`Proto-GroupName`) และทำซ้ำโควตาเหล่านี้สำหรับผู้ใช้หรือกลุ่มที่ระบุ

ผู้ใช้มีค่าเกินข้อจำกัดขั้นต้นที่กำหนดไว้สำหรับระยะ `grace` ดีฟอลต์คือ 1 สัปดาห์ เมื่อหมดเวลาของระยะ `grace` ข้อจำกัดขั้นต้นจะถูกบังคับเป็นข้อจำกัดเข้มงวด ระยะ `grace` สามารถถูกระบุเป็น วัน ชั่วโมง นาที หรือวินาที ค่า 0 หมายถึงว่าระยะ `grace` ดีฟอลต์ ถูกกำหนดไว้; ค่า 1 วินาทีหมายถึงไม่มีการให้ระยะ `grace` แฟล็ก `-t` เปลี่ยน ระยะ `grace`

ฟิลด์ที่แสดงในไฟล์ชั่วคราวคือ:

ไอเท็ม	คำอธิบาย
บล็อกที่ใช้อยู่	จำนวนปัจจุบันของบล็อกระบบไฟล์ 1KB ที่ใช้โดยผู้ใช้หรือกลุ่มนี้
Inodes ที่ใช้อยู่	จำนวนปัจจุบันของไฟล์ที่ใช้โดยผู้ใช้หรือกลุ่มนี้
ข้อจำกัดบล็อกขั้นต้น	จำนวนของบล็อก 1KB ที่ผู้ใช้หรือกลุ่มจะได้รับอนุญาตให้ใช้ ระหว่างการดำเนินการปกติ
ข้อจำกัดบล็อกเข้มงวด	จำนวนรวมของบล็อก 1KB ที่ผู้ใช้หรือกลุ่มจะได้รับอนุญาตให้ใช้ รวมถึงหน่วยเก็บชั่วคราว ระหว่างการดำเนินการปกติ
ข้อจำกัดขั้นต้น Inode	จำนวนของไฟล์ที่ผู้ใช้หรือกลุ่มจะได้รับอนุญาตให้สร้าง ระหว่างการดำเนินการปกติ
ข้อจำกัด Inode เข้มงวด	จำนวนรวมของไฟล์ที่ผู้ใช้หรือกลุ่มจะได้รับอนุญาตให้สร้าง รวมถึงไฟล์ชั่วคราวที่สร้าง ระหว่างการดำเนินการปกติ

**หมายเหตุ:** ข้อจำกัดเข้มงวดที่มีค่า 1 หมายถึงว่าไม่อนุญาตให้ทำการจัดสรร ข้อจำกัดขั้นต้นที่มีค่า 1, รวมกับ ข้อจำกัดเข้มงวดที่มีค่า 0 หมายถึงการจัดสรรได้รับอนุญาตเฉพาะ เป็นการชั่วคราว

เมื่อเอดิเตอร์จบการทำงาน คำสั่ง `edquota` อ่านไฟล์ชั่วคราวและแก้ไขไฟล์โควตาไบนารี เพื่อสะท้อนถึงการเปลี่ยนแปลง

ข้อจำกัดเข้มงวดหรือขั้นต้นสามารถถูกระบุได้ในจำนวน 1 KB บล็อกทั้งหมดเท่านั้น

## แฟล็ก

## ไอเท็ม คำอธิบาย

- g แกะไขโควตาของกลุ่มที่ระบุหนึ่งกลุ่มหรือมากกว่านั้น
  - p เมื่อเรียกด้วยแฟล็ก -u, ทำซ้ำ โควตาที่กำหนดสำหรับผู้ใช้ prototypical สำหรับแต่ละผู้ใช้ที่ระบุ เมื่อเรียก ด้วยแฟล็ก -g แฟล็ก -p ทำซ้ำโควตาที่กำหนดสำหรับกลุ่ม prototypical สำหรับแต่ละกลุ่ม ที่อยู่ในรายการ
  - t เปลี่ยนระยะ grace ระหว่างโควตาซึ่งมีค่าเกินไต่ก่อน ขอจำกัดขั้นต้นจะถูกกำหนดเป็นข้อจำกัดเข้มงวด ค่าตีพอลล์ของระยะ grace คือ 1 สัปดาห์ เมื่อเรียกด้วยแฟล็ก -u ระยะ grace ถูกเซตสำหรับระบบไฟล์ทั้งหมด พร้อมโควตาผู้ใช้ที่ระบุในไฟล์ /etc/filesystems เมื่อเรียกด้วยแฟล็ก -g ระยะ grace ถูกเซตสำหรับระบบไฟล์ทั้งหมด พร้อมโควตากลุ่มที่ระบุในไฟล์ /etc/filesystems
- หมายเหตุ: หลังจากเปลี่ยน ระยะ grace โดยใช้คำสั่ง edquota ค่าระยะ grace ใหม่จะยังไม่แสดงผล จนกว่าไฟล์ quota.user และ quota.group จะถูก รีเฟรชโดยการรัน คำสั่ง quotaoff ตามด้วยคำสั่ง quotaon ผู้ใช้ซึ่งมีค่าถึงระยะ grace เก่า ต้องลดการใช้งานระบบไฟล์ลงเป็นระดับที่ต่ำกว่าข้อจำกัด ขั้นต้นเพื่อให้สามารถใช้ระยะ grace ใหม่ได้ในอนาคต เมื่อผู้ใช้เหล่านี้มีค่าระยะเกินข้อจำกัดขั้นต้น grace period ใหม่จะแสดงผล
- u แกะไขโควตาของผู้ใช้

หมายเหตุ: ถ้าชื่อผู้ใช้หรือชื่อกลุ่มมีตัวเลขทั้งหมด ดังนั้นจะถูก ปฏิบัติเป็น user หรือ group ID จากนั้นโควตาจะถูกแก้ไข สำหรับ ID ไม่ใช่สำหรับชื่อ

## ความปลอดภัย

ไอเท็ม	คำอธิบาย
ค่าควบคุมการเข้าถึง:	เฉพาะผู้ใช้ root ที่สามารถเรียกใช้งานคำสั่งนี้ได้

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

เมื่อต้องการสร้างโควตาสำหรับผู้ใช้ shar1 ให้ใช้โควตาที่กำหนดสำหรับผู้ใช้ davec เป็นต้นแบบ ให้ป้อน:

```
edquota -u -p davec shar1
```

## ไฟล์

ไอเท็ม	คำอธิบาย
quota.user	ระบุโควตาผู้ใช้
quota.group	ระบุโควตากลุ่ม
/etc/filesystems	มีชื่อระบบไฟล์และตำแหน่ง

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง quota

คำสั่ง quotacheck

quotaon และ quotaoff

ภาพรวมระบบโควตาดีสก์

---

## คำสั่ง **efsenable**

### วัตถุประสงค์

เรียกทำงานความสามารถ Encrypted File System (EFS) บนระบบ

### ไวยากรณ์

```
efsenable -a [-v] [-k <algo>] [-f <cipher>] [-m <mode>] [-u <yes/no>] [-e <algo>] [-d Basedn]
```

```
efsenable -q
```

### คำอธิบาย

คำสั่ง **efsenable** เรียกทำงานความสามารถ EFS บนระบบ โดยจะสร้าง EFS administration keystore, keystore ผู้ใช้ และ keystore กลุ่มความปลอดภัย Keystore คือที่เก็บคีย์ ที่มีข้อมูลความปลอดภัย EFS แอ็กเชสคีย์เพื่อเข้าถึง EFS administration keystore ถูกเก็บใน user's keystore แอ็คทีฟที่สร้างขึ้นใหม่ และใน keystore กลุ่มความปลอดภัย คำสั่ง **efsenable** สร้างไดเรกทอรี `/var/efs` ไฟล์ `/etc/security/user` และ `/etc/security/group` ถูกอัปเดตด้วยแอ็คทีฟอีทีริบิต EFS ใหม่ คำสั่ง **efsenable** ยังอัปเดตฐานข้อมูล `Config_Rules ODM`

**หมายเหตุ:** แพ็กเกจ Crypto Library (CLiC) `clirc.rte` ต้องถูก ติดตั้งบนระบบ เพื่อให้คำสั่งนี้ทำงานสำเร็จ คำสั่ง EFS ยังต้องการให้ Role Based Access Control (RBAC) ถูกเปิดใช้บนระบบ ซึ่งเป็นค่ากำหนดดีฟอลต์

**หมายเหตุ:** ชุดไฟล์ Crypto Library (CLiC) `clirc.rte.lib` ขั้นต่ำต้องเป็น 4.6 สำหรับรีลีส AIX ของ **efsenable** 6.1 TL3 และใหม่กว่า

### แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-a</code>	เรียกทำงานความสามารถ EFS บนระบบ
<code>-d Basedn</code>	ใช้ชื่อ distinguished names (DN) ฐาน <code>ou=UsrKeystore, ou=GrpKeystore, ou=EfsCookies</code> และ <code>ou=AdmKeystore</code> บนเซิร์ฟเวอร์ LDAP เพื่อทำให้ง่ายขึ้นสำหรับรายการ keystore ที่จะถูกสร้างพร้อมกับ โครงสร้างไดเรกทอรีโลคัลสำหรับ keystore <code>Basedn</code> ที่ส่ง เป็นอาร์กิวเมนต์ตามด้วยแฟล็กนี้จะถูกใช้เป็น <code>Basedn</code> สำหรับ keystore base distinguished names
<code>-v</code>	โหมดรายละเอียด
<code>-k algo</code>	ดีฟอลต์อัลกอริทึมสำหรับคีย์ แฟล็ก <code>algo</code> มีค่าหนึ่งในค่าต่อไปนี้: <ul style="list-style-type: none"><li>• RSA_1024 (ดีฟอลต์)</li><li>• RSA_2048</li><li>• RSA_4096</li></ul>
<code>-f cipher</code>	การเข้ารหัสดีฟอลต์สำหรับไฟล์ แฟล็ก <code>cipher</code> มีค่าใดค่าหนึ่งต่อไปนี้: <ul style="list-style-type: none"><li>• AES_128_CBC (ดีฟอลต์)</li><li>• AES_192_CBC</li><li>• AES_256_CBC</li><li>• AES_128_ECB</li><li>• AES_192_ECB</li><li>• AES_256_ECB</li></ul>

ไอเท็ม	คำอธิบาย
-m mode	ดีฟอลต์โหมดสำหรับ keystores แฟล็ก mode มีค่าหนึ่งในค่าต่อไปนี้: <ul style="list-style-type: none"> <li>admin (ดีฟอลต์)</li> <li>guard</li> </ul>
-u [yesno]	ระบุว่าผู้ใช้สามารถเปลี่ยนโหมดได้หรือไม่ ค่าดีฟอลต์คือ "yes"
-e algo	อัลกอริทึมสำหรับคีย์การดูแลระบบ EFS ค่า algo ที่เป็นไปได้เหมือนกับค่าของแฟล็ก -k
-q	แสดงรายการของอัลกอริทึมที่มีอยู่

## สถานะออก

ไอเท็ม	คำอธิบาย
0	คำสั่งทำงานเสร็จสมบูรณ์
1	มีข้อผิดพลาดเกิดขึ้นระหว่างการดำเนินการคำสั่ง
2	มีข้อผิดพลาดทางไวยากรณ์เกิดขึ้นบนบรรทัดคำสั่ง

## ความปลอดภัย

ไอเท็ม	คำอธิบาย
ค่าควบคุมการเข้าถึง:	เฉพาะผู้ใช้ root หรือผู้ใช้ที่มีสิทธิ์ aix.security.efs และเป็นสมาชิกของกลุ่มความปลอดภัย เท่านั้นที่สามารถรันคำสั่งนี้ได้

## ตัวอย่าง

- เมื่อต้องการแสดงอัลกอริทึมที่มีให้พร้อม:

```
efsenable -q
```
- เมื่อต้องการเรียกทำงาน EFS ด้วยดีฟอลต์พารามิเตอร์ให้พร้อม:

```
efsenable -a
```
- เมื่อต้องการเรียกทำงาน EFS ด้วย non-default อัลกอริทึมสำหรับคีย์ และการเข้ารหัสสำหรับไฟล์ให้พร้อม:

```
efsenable -a -k RSA_4096 -f AES_256_CBC -e RSA_4096
```
- เมื่อต้องการเรียกทำงาน EFS ด้วย DN ฐานที่สร้างบนเซิร์ฟเวอร์ LDAP พร้อมกับ โครงสร้างไดเรกทอรีโลคัล ให้พิมพ์คำสั่งต่อไปนี้:

```
efsenable -a -d cn=aixdata
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/etc/security/user	มีอ็อปเตตของแอ็ททริบิวต์ EFS
/etc/security/group	มีอ็อปเตตของแอ็ททริบิวต์ EFS
/var/efs/users/	มีไดเรกทอรีสำหรับ keystores ผู้ใช้
/var/efs/groups/	มีไดเรกทอรีสำหรับกลุ่ม keystores
/var/efs/efs_admin/	มีไดเรกทอรีสำหรับ EFS administration keystore
/var/efs/efsenabled	สั่งให้ EFS ถูกเปิดใช้งานบนระบบ

## ข้อมูลที่เกี่ยวข้อง:

การรักษาความปลอดภัยระบบปฏิบัติการพื้นฐาน

---

## คำสั่ง **efskeymgr**

### วัตถุประสงค์

จัดการที่เก็บผู้ใช้และกลุ่ม สำหรับคีย์ Encrypted File System (EFS) (หรือที่เก็บคีย์)

### ไวยากรณ์

**efskeymgr -?**

**efskeymgr -q**

**efskeymgr -V**

**efskeymgr [-L *load\_module*]-C <group>**

**efskeymgr -P <Open-SSH Public Key file >**

หมายเหตุ: ไฟล์พับลิคคีย์อยู่ในไดเรกทอรี `~/.ssh/` directory

**efskeymgr [-L *load\_module*] [-d] [-k <ks>] [-g] [-p <pw>] -v**

**efskeymgr [-L *load\_module*] [-d] [-k <ks>] -m**

**efskeymgr [-L *load\_module*] [-d] [-k <ks>] [-g] [-p <pw>] -o <cmd>**

**efskeymgr [-L *load\_module*] [-d] [-c <cmd>]**

**efskeymgr [-L *load\_module*] [-d] [-k <ks>] [-g] [-p <pw>] -n**

**efskeymgr [-L *load\_module*] [-d] [-k <ks>] [-g] [-p <pw>] -r <mode>**

**efskeymgr [-L *load\_module*] [-d] [-k <ks>] [-g] [-p <pw>] -s <ks2>**

**efskeymgr [-L *load\_module*] [-d] [-k <ks>] [-g] [-p <pw>] -S <ks2>**

**efskeymgr [-L *load\_module*] [-d] [-k <ks>] [-g] [-p <pw>] -R <algo>**

**efskeymgr [-L *load\_module*] [-d] [-k <ks>] [-g] [-p <pw>] -D <fp>**

**efskeymgr [-L *load\_module*] [-d] [-k <ks>] [-g] [-p <pw>] -e <file>**

### คำอธิบาย

คำสั่ง **efskeymgr** ใช้กับการดำเนินการจัดการคีย์ทั้งหมดที่ EFS ต้องใช้ เมื่อ เปิดใช้งาน EFS บนระบบที่มีคำสั่ง **efsenable** ที่เก็บคีย์ (ที่เก็บคีย์พับลิคและไพรเวต) ถูกสร้างขึ้นใน ไดเรกทอรี `/var/efs`

รหัสผ่านเริ่มต้นของ keystore ผู้ใช้คือรหัสผ่านล็อกอินของผู้ใช้ keystores กลุ่มและ keystores ผู้ดูแลระบบไม่ถูกป้องกันด้วยรหัสผ่าน แต่โดยแอ็กเซสคีย์ แอ็กเซสคีย์ถูกเก็บไว้ภายใน keystores ผู้ใช้ทั้งหมด ที่เป็นของกลุ่มนี้

เมื่อคุณเปิด keystore (ขณะล็อกอินหรือด้วยคำสั่ง `efskeymgr`) ไพรเวตคีย์ที่มีใน keystore นี้ถูกส่งไปที่เคอร์เนลและเชื่อมโยงกับกระบวนการ ถ้าพบแอ็กเซสคีย์ใน keystore, keystores ที่ตรงกัน จะถูกเปิดเช่นกันและคีย์จะถูกส่งไปที่เคอร์เนลของตัวเองโดยอัตโนมัติ

Keystores สนับสนุนโหมดการดูแลระบบสองโหมด: โหมด admin และโหมด guard

### โหมด admin

เมื่อที่เก็บคีย์ถูกตั้งค่าเป็นโหมดนี้ ผู้ดูแลระบบ EFS ที่มี การอนุญาต `aix.security.efs` RBAC และคีย์เข้าถึง เพื่อดูแลที่เก็บคีย์สามารถเปิดที่เก็บคีย์ขึ้นเพื่อดำเนินการจัดการ รวมถึงรีเซ็ต รหัสผ่าน การสร้างคีย์ใหม่ การเพิ่มหรือการลบคีย์เข้าถึง และอื่นๆ

### โหมด guard

เมื่อ keystore ถูกเซตเป็นโหมดนี้ ผู้ดูแลระบบ EFS ไม่สามารถ เข้าถึง keystore ได้ ในโหมดนี้ ถ้ารหัสผ่าน keystore สูญหาย เป็นไปไม่ได้ที่จะเรียกคืนไพรเวตคีย์

เมื่อรหัสผ่าน keystore เหมือนกับรหัสผ่านล็อกอิน keystore จะถูกเปิดโดยอัตโนมัติเมื่อเวลาล็อกอินและคีย์พร้อมใช้งานในเซสชัน รหัสผ่าน keystore ถูกเก็บโดย sync กับ รหัสผ่านล็อกอินเมื่อคำสั่ง `passwd` ถูกใช้ และรหัสผ่านเก่าถูกระบุ ถ้าในบางจุดที่รหัสผ่าน keystore ไม่ sync กับรหัสผ่านล็อกอิน คุณสามารถเปลี่ยนรหัสผ่าน keystore โดยใช้คำสั่ง `efskeymgr` เมื่อรหัสผ่านไม่ได้ถูก synchronize, คีย์จะไม่เชื่อมโยงกับ เซสชันโดยอัตโนมัติเมื่อคุณล็อกอินอีกต่อไป

คำสั่งต่อไปนี้ให้หรือลบ EFS credentials เฉพาะ การดำเนินการคำสั่ง `cmd` เมื่อ คำสั่ง `cmd` ส่งคืน, credentials ของกระบวนการก่อนหน้านั้นถูกเรียกคืน

```
efskeymgr -o <cmd> and efskeymgr -c <cmd>
```

เมื่อ ไพรเวตคีย์ถูกสร้างขึ้นใหม่ในที่เก็บคีย์ ไพรเวตคีย์ใหม่จะถูกสร้าง และคีย์เก่าถูกทำเครื่องหมาย "deprecated"

**หมายเหตุ:** คีย์ใหม่ไม่ถูกส่งไปที่เคอร์เนล คุณต้องเปิด keystore ของคุณอีกครั้ง ด้วยคำสั่ง `efskeymgr` หรือโดยการปิดและเปิดเซสชันของคุณ สำหรับคีย์ใหม่เพื่อให้พร้อมใช้ สำหรับการดำเนินการกับไฟล์ คีย์ที่ถูกปฏิเสธยังสามารถ ถูกใช้เพื่อถอดรหัสไฟล์ แต่ไม่สามารถใช้เพื่อเข้ารหัสไฟล์ได้อีก คีย์ที่ถูกปฏิเสธสามารถถูกเอาออกจาก keystore, แต่ในกรณีนี้ ไฟล์ทั้งหมดที่ถูกเข้ารหัสด้วยคีย์เก่าจะไม่สามารถถูกเข้าถึงได้อีกต่อไป

**หมายเหตุ:** คำสั่ง EFS ต้องการให้ Role Based Access Control (RBAC) ถูกเปิดใช้บนระบบ ซึ่งเป็นค่ากำหนดดีฟอลต์

### การดำเนินการที่หน่วง

ในบางกรณี keystore ไม่สามารถถูกแก้ไขได้โดยตรงด้วยคำสั่ง หรือการดำเนินการ เมื่อเป็นเช่นนั้น ไฟล์พิเศษจะถูกสร้างในไดเรกทอรี keystore และจะถูกวิเคราะห์ค่าในครั้งต่อไปที่ keystore ถูกเปิด ไฟล์ พิเศษนี้เรียกว่าคูกี้ สำหรับ keystores ในโหมด admin, คูกี้จะถูกวิเคราะห์ค่าโดยอัตโนมัติเมื่อ keystore ถูกเปิด (เมื่อล็อกอิน หรือเมื่อคำสั่ง `efskeymgr` ถูกรัน) สำหรับ keystores ในโหมด guard, จะไม่มีการวิเคราะห์ค่าคูกี้โดยอัตโนมัติ ผู้ใช้ต้องให้ การอนุมัติสำหรับการแก้ไขแต่ละครั้งของที่เก็บคีย์ เมื่อ คุณเปิดเซสชัน ข้อความจะถูกแสดง ถ้ามีการดำเนินการ คางอยู่ใน EFS keystore ของคุณ:

- ไพรเวตคีย์ของคุณต้องถูกสร้างใหม่
- คุณได้รับสิทธิให้เข้าถึง `group/group1` keystore

คุณต้องรันคำสั่ง `efskeymgr -v` เพื่อประมวลผลการดำเนินการที่ค้างอยู่

การดำเนินการต่อไปนี้เป็นไปได้:

- การสร้างโพรเวตคีย์ใหม่ ซึ่งมีผลให้โพรเวตคีย์ใหม่ถูกสร้าง และคีย์เก่าถูกทำเครื่องหมายเป็น "deprecated"
- แอ็กเซสคีย์ใหม่ เมื่อคุณยอมรับคูกี้นี้ คุณจะได้รับการเข้าถึง ที่เก็บคีย์ใหม่ (ตัวอย่างเช่น ที่เก็บคีย์ของกลุ่มซึ่งคุณถูกเพิ่มเข้าไป)
- ลบแอ็กเซสคีย์ เมื่อคุณยอมรับคูกี้นี้ (ตัวอย่าง เมื่อ แอ็กเซสคีย์ถูกลบออกจากกลุ่ม) คุณจะเสียการเข้าถึง keystore

**หมายเหตุ:** เมื่อคุณรันคำสั่ง `efskeymgr` ด้วยแฟล็กที่เปิด keystore ของคุณ ตัวอย่าง, แฟล็ก `-v` คุณจะได้รับพร้อมท์ว่าคุณต้องการทำอะไรกับแต่ละคูกี้ อีอพชัน มีดังนี้:

- ยอมรับคูกี้: keystore ของคุณถูกแก้ไขตาม คูกี้ จากนั้นคูกี้ก็ถูกทำลาย
- เลื่อนเวลาคูกี้: keystore ของคุณไม่ถูกแก้ไข และคูกี้ไม่ถูกลบ คุณจะได้รับพร้อมท์ในครั้งต่อไปสำหรับการดำเนินการ
- ลบคูกี้: keystore ของคุณไม่ถูกแก้ไข และคูกี้ถูกลบ คุณต้องใช้คำสั่ง `efskeymgr` เพื่อทำแอ็กชัน อีกครั้ง

## แฟล็ก

ไอเท็ม	คำอธิบาย
แฟล็กทั่วไป:	
-d	โหมดยุติการ
-g	ไม่ประมวลผลการดำเนินการที่ค้างอยู่เมื่อเปิด keystore
-k ks	การดำเนินการถูกกำหนดไว้กับ ks keystore แทน user's keystore ที่แอ็กทีฟ ค่า ks มีค่าได้ดังนี้:
	<b>user/ &lt;login&gt;</b> ที่เก็บคีย์ <login> ผู้ใช้
	<b>group/ &lt;grpname&gt;</b> ที่เก็บคีย์ <grpname> กลุ่ม
	<b>admin/</b> EFS administration keystore
-L load_module	ระบุโมดูลที่โหลดได้ที่จะใช้สำหรับการดำเนินการ keystore
-ppw	รหัสผ่านที่ใช้เพื่อเปิด keystore ไม่แนะนำให้ใช้แฟล็กนี้เนื่องจากผู้ใช้อื่นสามารถเห็นได้โดยใช้คำสั่ง ps เป็นต้น
-P filename	ส่งพับลิกคีย์คูกี้สำหรับคีย์ทั้งหมดที่อยู่ไปไว้ในไฟล์ OpenSSH ที่อยู่ในไดเรกทอรี ~/.ssh/authorized_keys
แฟล็กสำหรับ คำสั่ง (ไม่มีการเข้าถึงไฟล์ keystore):	
-?	แสดงวิธีใช้และการจบการทำงานของคำสั่ง
-q	แสดงรายการของอัลกอริทึมที่สนับสนุนสำหรับการสร้างคีย์ใหม่
-V	แสดงคีย์ที่เชื่อมโยงกับ credentials กระบวนการที่แอ็กทีฟในเคอร์เนล
แฟล็กสำหรับคำสั่ง (เข้าถึงแบบ อ่านอย่างเดียวกับ keystore):	
-c <cmd>	ลบคีย์ทั้งหมดออกจากเคอร์เนล จากนั้นรันคำสั่ง cmd คีย์ถูกเรียกคืนเมื่อคำสั่ง cmd สิ้นสุด
-m	แสดงการดำเนินการที่ค้างอยู่ทั้งหมดใน keystore
-o <cmd>	เปิด keystore และส่งคีย์ จากนั้นรันคำสั่ง cmd คีย์ถูกละเว้นเมื่อคำสั่ง cmd สิ้นสุด
-v	แสดงเนื้อหาของไฟล์ keystore
แฟล็กสำหรับคำสั่ง (เข้าถึงแบบ อ่าน/เขียน กับ keystore)	
:	
-C <group>	สร้าง keystore ของกลุ่ม group
-D <fp>	ลบโพรเวตคีย์ที่ปฏิเสธจาก keystore ค่า fp เป็นคีย์ fingerprint

ไอเท็ม คำอธิบาย

-e <file> เอ็กซ์พอร์ต keystore ไปที่ไฟล์ ไฟล์คือ PKCS#12 ที่เข้ารหัสและมีคีย์พับลิกและไพรเวตจาก keystore ไฟล์นี้สามารถถูกใช้ใน openssh, เป็นต้น

-n สำหรับ keystores ผู้ใช้พร้อมดอร์ทส์ผ่านใหม่สำหรับ keystore สำหรับ keystores กลุ่ม สร้างเอ็กซ์เซสคีย์ใหม่และส่งไปที่สมาชิก กลุ่ม สำหรับ admin keystores, สร้างเอ็กซ์เซสคีย์ใหม่จากนั้นคีย์ ต้องถูกส่งไปที่ผู้ดูแลระบบ EFS ด้วยคำสั่ง efskeymgr

-R <algo> สร้างไพรเวตคีย์ของ keystore ใหม่ ดูที่แฟล็ก -q สำหรับค่าที่ใช้ได้กับพารามิเตอร์ algo

-r <mode> เปลี่ยนโหมดการดูแลระบบ keystore ค่า mode มีค่าได้ดังนี้:

admin ผู้ดูแลระบบ EFS สามารถดูแลที่เก็บคีย์ การดำเนินการที่ค้างอยู่ ถูกนำมาใช้โดยอัตโนมัติ

guard ผู้ดูแลระบบ EFS ไม่สามารถจัดการ keystore ผู้ใช้ได้รับ พร้อมดอร์ทส์สำหรับการดำเนินการที่ค้างอยู่

-S <ks2> ลบเอ็กซ์เซสคีย์ ks2 ออกจาก keystore ในการเปิดที่เก็บคีย์ ครั้งหลัง ไพรเวตคีย์ ks2 จะไม่ถูกกกดโดยอัตโนมัติ อีกต่อไป

-s <ks2> ส่งเอ็กซ์เซสคีย์ keystore ไปที่ ks2 keystore ในการเปิดคีย์ ks2 ครั้งต่อๆ ไป ไพรเวตคีย์ของที่เก็บคีย์ ถูกโหลดโดยอัตโนมัติ

## สถานะออก

ไอเท็ม	คำอธิบาย
0	คำสั่งที่รันเป็นผลสำเร็จ
1	มีข้อผิดพลาดเกิดขึ้นระหว่างการดำเนินการคำสั่ง
2	มีข้อผิดพลาดทางไวยากรณ์เกิดขึ้นบนบรรทัดคำสั่ง

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

- เมื่อต้องการดูเนื้อหา keystore ของคุณ ให้ป้อน:
 

```
efskeymgr -v
```
- เมื่อต้องการดูคีย์ที่เกี่ยวข้องกับแอ็คทีฟเซลล์ ให้ป้อน:
 

```
efskeymgr -V
```
- เมื่อต้องการสร้างไพรเวตคีย์ใหม่จาก keystore ของคุณ ให้ป้อน:
 

```
efskeymgr -R RSA_1024
```
- เมื่อต้องการลบคีย์ที่ปฏิเสธให้ป้อน:
 

```
efskeymgr -D dbb62547:d6925088:45357fd3:54cddbba:27b255a9
```
- เมื่อต้องการส่งคีย์เข้าถึงของกลุ่ม "students" ไปยังผู้ใช้ "joe" ให้ป้อน:
 

```
efskeymgr -k group/students -s user/joe
```
- เมื่อต้องการเก็บคีย์ Open-SSH Client users Open-SSH Public key ในที่เก็บคีย์ปลายทาง โดยที่ไฟล์ ~/.ssh/authorized\_keys มีพับลิกคีย์ที่ติดตั้งไว้ให้ป้อน:
 

```
efskeymgr -P ~/.ssh/authorized_keys
```
- เมื่อต้องการสร้าง Group keystore โดยตรงบน LDAP, ถ้ามีการตั้งค่าไว้:
 

```
efskeymgr -L LDAP -C staff
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/var/efs	มี keystores ทั้งหมด
/etc/security/user	มีแอ็ททริบิวต์ EFS สำหรับการสร้างและการจัดการ users keystore
/etc/security/group	มีแอ็ททริบิวต์ EFS สำหรับการสร้าง groups keystore

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง efsenable” ในหน้า 347

“คำสั่ง efsmgr” ในหน้า 355

### ข้อมูลที่เกี่ยวข้อง:

การรักษาความปลอดภัยระบบปฏิบัติการพื้นฐาน

---

## คำสั่ง efskstoldif

### วัตถุประสงค์

พิมพ์ users หรือ groups keystore ของ EFS ที่ถูกกำหนดแบบโลคัล กับ **stdout** ในรูปแบบ Idif

### ไวยากรณ์

```
efskstoldif -d baseDN [-u | -g] { ALL | Name [Name] ... }
```

### คำอธิบาย

คำสั่ง **efskstoldif** อ่านข้อมูลจากไฟล์ users หรือ groups keystore ของ EFS ที่กำหนดแบบโลคัลและพิมพ์ผลไปที่ **stdout** ในรูปแบบ Idif ถ้ามีการเปลี่ยนทิศทางไปที่ไฟล์ ผลลัพธ์สามารถถูกเพิ่มให้กับเซิร์ฟเวอร์ LDAP ด้วยคำสั่ง **ldapadd** พร้อมกับแฟล็ก **-b** หรือคำสั่ง **ldif2db**

คำสั่ง **efskstoldif** อ่านไฟล์ `/etc/security/ldap/sectoldif.cfg` เพื่อกำหนด ข้อมูลที่จะกำหนดชื่อ ผู้ใช้ กลุ่ม และคูกี้ sub-trees ที่จะเอ็ททริบิวต์ ข้อมูลส่งไป คำสั่ง **efskstoldif** เอ็ททริบิวต์ข้อมูล ไปที่ชนิด USERKEYSTORE, GROUPKEYSTORE, EFSCOOKIES และ ADMINKEYSTORE ที่กำหนดในไฟล์เท่านั้น ชื่อที่ระบุในไฟล์จะถูกใช้ เพื่อสร้าง sub-trees ได้ distinguished name (DN) ฐานที่ระบุ ด้วยแฟล็ก **-d** สำหรับข้อมูลเพิ่มเติม ดูที่ไฟล์ `/etc/security/ldap/sectoldif.cfg` ใน AIX เวอร์ชัน 6.1 TL 4 สำหรับการอ้างอิง

การสร้างเอาต์พุต LDIF ไม่ได้ดูที่แอ็ททริบิวต์ `efs_keystore_access` หรือ `efs_adminks_access` ของ users/groups ค่าที่เป็นไปได้คือ “file” หรือ “ldap” ที่รูปแบบ LDIF จะถูกสร้าง สำหรับ users หรือ groups keystore ที่มีการสร้างรูปแบบ Idif, ถ้ามีคูกี้ที่อยู่สำหรับ keystore เหล่านั้น จะมีการสร้าง Idif แทนที่ในจำนวนเท่ากัน

**หมายเหตุ:** ถ้ามีคูกี้ที่อยู่ในไฟล์, จะมีการสร้าง LDIF ที่เท่ากันขึ้นด้วย ผู้ดูแลระบบ จำเป็นต้องดูและความสอดคล้องกันของรายการ keystore บน LDAP และไฟล์ ถ้าจำเป็น

### แฟล็ก

ไอเท็ม	คำอธิบาย
-d baseDN	Specifies the base distinguished names (DN) under which to place the EFS Keystore data.
-g ALLNames ...	กำหนดให้คำสั่งสร้างเอาต์พุตสำหรับ กลุ่มที่ระบุในอาร์กิวเมนต์ที่สืบทอด
	ALL ระบุว่ากลุ่มทั้งหมดต้องถูกนำมาพิจารณา
	ชื่อ ระบุว่าชื่อกลุ่มเดี่ยวหรือรายการชื่อกลุ่มที่แยก ด้วยช่องว่าง
-u ALLNames ...	กำหนดให้คำสั่งสร้างเอาต์พุตสำหรับ ผู้ใช้ที่ระบุในอาร์กิวเมนต์ที่สืบทอด
	ALL ระบุว่าผู้ใช้ทั้งหมดต้องถูกนำมาพิจารณา
	ชื่อ ระบุว่าชื่อผู้ใช้เดี่ยวหรือรายการชื่อผู้ใช้ที่แยก ด้วยช่องว่าง

## สถานะออก

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: คำสั่งนี้ควรให้สิทธิ์การเข้าถึง execute (x) เฉพาะกับผู้ใช้ root

**ข้อควรทราบสำหรับ ผู้ใช้ RBAC และผู้ใช้ Trusted AIX:** คำสั่งนี้สามารถดำเนินการที่ต้องใช้สิทธิ์ เฉพาะผู้ใช้ที่ได้รับสิทธิ์ใช้งานเท่านั้น สามารถรับการดำเนินงานพิเศษได้ สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการอนุญาต และสิทธิ์ใช้งาน, ดูที่ [ฐานข้อมูลคำสั่ง Privileged](#) ใน *ความปลอดภัย* สำหรับรายการ ของสิทธิ์ใช้งานและการอนุญาตที่เชื่อมโยงกับคำสั่งนี้ โปรดดูที่ คำสั่ง `Issecattr` หรือคำสั่งย่อย `getcmdattr`

## ไฟล์

ไอเท็ม	คำอธิบาย
/etc/security/user	มีแอตทริบิวต์ EFS สำหรับการสร้างและการจัดการ users keystore
/etc/security/group	มีแอตทริบิวต์ EFS สำหรับการสร้างและการจัดการ users keystore
/var/efs	มี keystores ทั้งหมด

## ตัวอย่าง

- เมื่อต้องการเอ็กซ์พอร์ตเนื้อหา users และ groups keystore ทั้งหมดไปที่รูปแบบ Idif ด้วย DN ฐานของ cn=aixdata, ให้พิมพ์คำสั่งต่อไปนี้:

```
efskstoldif -d cn=aixdata
```
- เมื่อต้องการเอ็กซ์พอร์ตเนื้อหา users keystore ทั้งหมดไปที่รูปแบบ Idif ด้วย DN ฐานของ cn=aixdata, ให้พิมพ์คำสั่งต่อไปนี้:

```
efskstoldif -d cn=aixdata -u ALL
```
- เมื่อต้องการเอ็กซ์พอร์ตเนื้อหา groups keystore ทั้งหมดไปที่รูปแบบ Idif ด้วย DN ฐานของ cn=aixdata, ให้พิมพ์คำสั่งต่อไปนี้:

```
efskstoldif -d cn=aixdata -g ALL
```
- เมื่อต้องการเอ็กซ์พอร์ตเฉพาะเนื้อหา users keystore ที่เลือกไปที่รูปแบบ Idif ด้วย DN ฐานของ cn=aixdata, ให้พิมพ์คำสั่งต่อไปนี้:

```
efskstoldif -d cn=aixdata -u davis smith
```

5. เมื่อต้องการเอ็กซ์พอร์ตเฉพาะเนื้อหา groups keystore ที่เลือกไปที่รูปแบบ ldif ด้วย DN ฐานของ cn=aixdata, ให้พิมพ์คำสั่งต่อไปนี้:

```
efskstoldif -d cn=aixdata -g finance managers
```

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mksecldap

คำสั่ง sectoldif

คำสั่ง /etc/security/ldap/sectoldif.cfg

การรักษาความปลอดภัยระบบปฏิบัติการพื้นฐาน

---

## คำสั่ง efsmgr

### วัตถุประสงค์

จัดการการเข้ารหัสไฟล์และการถอดรหัสไฟล์สำหรับ Encrypted File System (EFS)

### ไวยากรณ์

**efsmgr -?**

**efsmgr -q [-v]**

**efsmgr -C <cipher> [-v]**

**efsmgr [-c <file>] -e <file> [-v]**

**efsmgr [-c <cipher>] [-s] -E <dir> [-v]**

**efsmgr [-c <cipher>] -t <file> [-v]**

**efsmgr [-c <cipher>] [-s] -T <dir> [-v]**

**efsmgr -d <file> [-v]**

**efsmgr [-s] -D <dir> [-v]**

**efsmgr -l <file> [-v]**

**efsmgr [-s] -L <dir> [-v]**

**efsmgr -a <file> [-u <user> | -g <group>] [-v]**

**efsmgr -r <file> [-u <user> | -g <group>] [-v]**

## คำอธิบาย

คำสั่ง `efsmgr` ถูกกำหนดไว้ให้กับการจัดการการเข้ารหัสไฟล์ภายใน EFS ไฟล์ที่เข้ารหัสสามารถถูกสร้างเฉพาะในระบบไฟล์ EFS-enabled JFS2 สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการเปิดใช้ EFS บนระบบของคุณ ดูที่คำสั่ง `mkfs`, `chfs`, `crfs` และ `efsenable`

มีสองวิธีในการสร้างไฟล์ที่เข้ารหัส: ทางตรงโดยการใช้ คำสั่งต่อไปนี้, หรือทางอ้อมเมื่อการสืบทอดถูกเซตบนระบบไฟล์ หรือไดเรกทอรี ซึ่งไฟล์ถูกสร้าง

```
efsmgr -e <file>
```

เมื่อการสืบทอดถูกเซตบนไดเรกทอรีไฟล์ใหม่ทั้งหมดที่สร้างในไดเรกทอรีนี้ ถูกเข้ารหัสโดยดีฟอลต์รหัสที่ใช้ในการเข้ารหัสไฟล์เป็นรหัส ที่สืบทอด ไดเรกทอรีใหม่สืบทอดรหัสเดียวกันนี้เช่นกัน ถ้าการสืบทอดถูกปิดใช้งาน ในไดเรกทอรีย่อย ไฟล์ใหม่ที่สร้างในไดเรกทอรีย่อยนี้จะไม่ถูก เข้ารหัส

เมื่อการสืบทอดถูกเซตบนระบบไฟล์ไฟล์ใหม่ทั้งหมดที่สร้างในระบบไฟล์ นี้ถูกเข้ารหัสโดยรหัสที่สืบทอด ถ้าการสืบทอดถูกเซต ทั้งไดเรกทอรีและระบบไฟล์ด้วยรหัสที่ต่างกัน, ไฟล์ใหม่ที่สร้าง ในไดเรกทอรีนี้จะถูกเข้ารหัสโดยรหัสที่สืบทอดจากไดเรกทอรี

การตั้งค่าหรือการลบการสืบทอดบนไดเรกทอรีหรือระบบไฟล์ไม่มีผล กับไฟล์ที่มีอยู่ คำสั่ง `efsmgr` ต้องถูก ใช้ทางตรงเพื่อเข้ารหัสหรือถอดรหัสไฟล์

ไพรเวจซี่ของเจ้าของไฟล์ต้องถูกโหลดลงในกระบวนการ ก่อนที่ ไฟล์ที่เข้ารหัสจะสามารถถูกสร้างได้ การเข้าถึงไฟล์ที่เข้ารหัสสามารถให้สิทธิ์ กับผู้ใช้หรือกลุ่มที่มี keystore, ซึ่งเป็นที่เก็บคีย์ที่มี ข้อมูลความปลอดภัย EFS สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการจัดการที่เก็บผู้ใช้ และกลุ่ม ดูที่คำสั่ง `efskeymgr`

เมื่อไฟล์ที่เข้ารหัสถูกเปิด, Discretionary Access Control (DAC) และ Access Control List (ACL) ถูกตรวจสอบสำหรับสิทธิ์ในการเข้าถึงไฟล์ ถ้าการเข้าถึงได้รับอนุญาต คีย์ที่โหลดลงในเคอร์เนลสำหรับกระบวนการ ถูกค้นหาไพรเวจซี่ ที่ตรงกับหนึ่งในคีย์การป้องกันของไฟล์ ถ้าพบคีย์ที่ตรงกัน เนื้อหาไฟล์สามารถอ่านได้ มิฉะนั้นการเข้าถึงจะถูก ปฏิเสธ

**หมายเหตุ:** คำสั่ง EFS ต้องการให้ Role Based Access Control (RBAC) ถูกเปิดใช้บนระบบ ซึ่งเป็นค่ากำหนดดีฟอลต์

## แฟล็ก

### ไวยากรณ์

คำอธิบาย	คำอธิบาย
-c <cipher>	ใช้รหัสนี้แทนรหัสที่สืบทอดหรือรหัสดีฟอลต์ ดูที่ คำสั่ง -q สำหรับค่า cipher ที่ใช้ได้
-g <group>	กลุ่มนี้ต้องถูกเพิ่มหรือลบออกจากรายการการเข้าถึง EFS คำ group เป็นได้ทั้ง gid หรือชื่อกลุ่ม
-s	การดำเนินการมีเป้าหมายที่ระบบไฟล์ไม่ใช่ไดเรกทอรี ในกรณีนี้พารามิเตอร์ dir ต้องเป็นจุดประกอบเข้า ของระบบไฟล์ที่สนับสนุน EFS
-u <user>	ผู้ใช้นี้ต้องถูกเพิ่มหรือลบออกจากรายการการเข้าถึง EFS คำ user เป็นได้ทั้ง uid หรือชื่อล๊อคอิน
-v	โหมตรายละเอียด
-?	แสดงวิธีใช้และการจบการทำงานของคำสั่ง
-a <file>	เพิ่มการเข้าถึงกับไฟล์ที่ระบุไว้กับรายการผู้ใช้และกลุ่มที่ระบุ ด้วยแฟล็ก -u และ -g
-C <cipher>	เปลี่ยนรหัสดีฟอลต์สำหรับผู้ใช้ของคุณเป็นค่า cipher
-D <dir>	ลบการสืบทอดบนไดเรกทอรี เมื่อต้องการใช้คำสั่งกับระบบไฟล์ทั้งหมด คุณต้องเพิ่มแฟล็ก -s
-d <file>	ถอดรหัสไฟล์ที่ระบุ
-E <dir>	เซตการสืบทอดบนไดเรกทอรี dir เมื่อต้องการใช้คำสั่งกับระบบไฟล์ทั้งหมด คุณต้องเพิ่มแฟล็ก -s
-e <file>	เข้ารหัสไฟล์ที่ระบุ
-L <dir>	แสดงรหัสที่สืบทอดบนไดเรกทอรีที่ระบุ
-I <file>	แสดงข้อมูลการเข้ารหัสของไฟล์ที่ระบุ: รหัสและคีย์ ที่สามารถถอดรหัสไฟล์
-q	แสดงรายการของรหัสที่สนับสนุน

ไอเท็ม	คำอธิบาย
-r <file>	เรียกคืนสิทธิ์การเข้าถึงกับไฟล์ที่ระบุให้กับรายการผู้ใช้และกลุ่มที่ระบุด้วยแฟล็ก -u และ -g
-T <dir>	เปลี่ยนรหัสที่สืบทอดบนไดเรกทอรีที่ระบุ เมื่อต้องการใช้คำสั่งกับระบบไฟล์สมบูร์น คุณต้องเพิ่มแฟล็ก -s
-t <file>	รีเฟรชสิทธิ์การเข้าถึงของไฟล์ที่ระบุ ซึ่งสามารถถูกใช้เพื่อเปลี่ยนรหัสไฟล์

## สถานะออก

ไอเท็ม	คำอธิบาย
0	คำสั่งทำงานเสร็จสมบูรณ์
1	มีข้อผิดพลาดเกิดขึ้นระหว่างการดำเนินการคำสั่ง
2	มีข้อผิดพลาดทางไวยากรณ์เกิดขึ้นบนบรรทัดคำสั่ง

## ตัวอย่าง

- เมื่อต้องการเข้ารหัสไฟล์ `database.txt` โดยใช้ strong cipher, ให้ป้อน:

```
efsmgr -e database.txt -c AES_256_CBC
```
- เมื่อต้องการแสดงรายการของคีย์ที่สามารถเปิดไฟล์ให้ป้อน:

```
efsmgr -l database.txt
```
- เมื่อต้องการเพิ่มการเข้าถึงให้กับผู้ใช้ joe และกับกลุ่ม maintainers กับไฟล์ให้ป้อน:

```
efsmgr -a database.txt -u joe -g maintainers
```
- เมื่อต้องการเซตการสืบทอดบนระบบไฟล์ของโฮมไดเรกทอรีให้ป้อน:

```
efsmgr -c AES_128_CBC -s -E /home
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/etc/security/user	มีแอตทริบิวต์รหัสดีพอลต์สำหรับผู้ใช้

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `efsenable`” ในหน้า 347

“คำสั่ง `efskeymgr`” ในหน้า 349

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `mkfs`

การรักษาความปลอดภัยระบบปฏิบัติการพื้นฐาน

---

## คำสั่ง `egrep`

### วัตถุประสงค์

ค้นหาไฟล์ตามรูปแบบ

### ไวยากรณ์

```
egrep [-h] [-i] [-p [Separator]] [-s] [-u] [-v] [-w] [-x] [-y] [[-b] [-n] | [-c | -l | -q]] { { -ePattern | -fStringFile } ... | Pattern } [ File ... ]
```

## คำอธิบาย

คำสั่ง `egrep` ค้นหา ไฟล์อินพุต (อินพุตมาตรฐานโดยดีฟอลต์) สำหรับบรรทัดที่ตรงกับรูปแบบ ที่ระบุโดยพารามิเตอร์ *Pattern* รูปแบบเหล่านี้ เป็น นิพจน์ทั่วไป แบบเต็มเหมือน ในคำสั่ง `ed` (ยกเว้น `\` (backslash) และ `\` (backslash คู่)) กฎต่อไปนี้ ยังใช้ กับคำสั่ง `egrep`:

- นิพจน์ทั่วไปที่ตามด้วย `+` (เครื่องหมายบวก) จับคู่ข้อมูลที่ตรงกับนิพจน์ทั่วไป
- นิพจน์ทั่วไปที่ตามด้วย `?` (เครื่องหมายคำถาม) จับคู่จำนวนการมีอยู่ของนิพจน์ทั่วไปที่มีค่าเป็นศูนย์หรือหนึ่ง
- นิพจน์ทั่วไปหลายนิพจน์ที่แยกด้วย `|` (แถบแนวตั้ง) หรือโดยอักขระขึ้นบรรทัดใหม่ จับคู่สตริงที่ ตรงกับ นิพจน์ทั่วไปที่มีอยู่
- นิพจน์ทั่วไปอาจถูกปิดใน `()` (วงเล็บ) เพื่อการจัดกลุ่ม

อักขระ new-line จะไม่ถูกจับคู่โดยนิพจน์ทั่วไป

ลำดับความสำคัญสำหรับโอเปอเรเตอร์คือ `[, ], *, ?, +, การต่อข้อมูล, |` และอักขระ new-line

**หมายเหตุ:** คำสั่ง `egrep` เหมือนกับคำสั่ง `grep` ที่มีแฟล็ก `-E` ยกเว้นว่าข้อความแสดงข้อผิดพลาดและการใช้งาน ต่างกัน และแฟล็ก `-s` ทำงานต่างไป

คำสั่ง `egrep` แสดงไฟล์ ที่มีบรรทัดที่ตรงกัน ถ้าคุณระบุพารามิเตอร์ *File* มากกว่าหนึ่งพารามิเตอร์ อักขระที่มีความหมาย พิเศษ สำหรับเซลล์ (`$, *, [, \, ^, (, ), \`) ต้องอยู่ในเครื่องหมายคำพูด เมื่อปรากฏในพารามิเตอร์ *Pattern* เมื่อพารามิเตอร์ *Pattern* ไม่ใช่สตริงปกติ โดยปกติคุณต้องปิดรูปแบบทั้งหมดในเครื่องหมายคำพูด เดียว ในนิพจน์เช่น `[a-z]`, เครื่องหมายลบหมายถึง "ถึง" ตามลำดับการเรียงปัจจุบัน ลำดับการเรียงอาจกำหนดคลาสเทียบเท่า สำหรับใช้ในช่องอักขระ โดยใช้ deterministic algorithm ที่รวดเร็วซึ่งบางครั้งจำเป็น ต้องใช้พื้นที่ที่วิเศษ

### Notes:

1. บรรทัดถูกจำกัดที่ 2048 ไบต์
2. ย่อหน้า (ภายใต้แฟล็ก `-p`) ขณะนี้ถูกจำกัดที่ความยาว 5000 อักขระ
3. อัยารันคำสั่ง `grep` กับไฟล์พิเศษเนื่องจากจะสร้างผลที่คาดการณไม่ได้
4. บรรทัดอินพุตไม่ควรมีอักขระ NULL
5. อินพุตไฟล์ควรจบด้วยอักขระขึ้นบรรทัดใหม่
6. แม้ว่าสามารถระบุ บางแฟล็กได้แบบพร้อมกัน บางแฟล็กจะแทนที่แฟล็กอื่นๆ ตัวอย่าง ถ้าคุณระบุ `-l` และ `-n` พร้อมกัน เฉพาะชื่อไฟล์ ถูกเขียนไปที่เอาต์พุตมาตรฐาน

## แฟล็ก

ไอเท็ม	คำอธิบาย
-b	นำหน้าแต่ละบรรทัดด้วยหมายเลขบล็อกที่พบ ใช้แฟล็กนี้เพื่อช่วยค้นหาหมายเลขบล็อกของคั่นดิสก์ตามบริบท แฟล็ก -b ไม่สามารถใช้ร่วมกับอินพุตจาก stdin หรือ ไฟล์
-c	แสดงเฉพาะจำนวนบรรทัดที่ตรงกันเท่านั้น
-e Pattern	ระบุ Pattern นี้ทำงานเหมือน Pattern ปกติ แต่มีประโยชน์เมื่อ Pattern เริ่มต้นด้วย - (เครื่องหมายลบ)
-f StringFile	ระบุไฟล์ที่มีสตริง
-h	ยังยั้งชื่อไฟล์เมื่อไฟล์หลายไฟล์กำลังถูกประมวลผล
-i	ละเว้นตัวพิมพ์ของตัวอักษร เมื่อทำการเปรียบเทียบ
-l	แสดงเพียงชื่อไฟล์ (ครั้งเดียว) กับบรรทัดที่พบ แต่ละชื่อไฟล์ถูกแยกด้วยอักขระขึ้นบรรทัดใหม่ ถ้าอินพุตมาตรฐาน ถูกค้นหา ชื่อพาร " (StandardInput)" ถูก ส่งคืน
-n	นำหน้าแต่ละบรรทัดด้วยหมายเลขบรรทัดสัมพัทธ์ในไฟล์
-p[Separator]	แสดงย่อหน้าทั้งหมดที่มีบรรทัดที่ตรงกัน ย่อหน้า ถูกค้นด้วยตัวค้นย่อหน้า ตามที่ระบุโดยพารามิเตอร์ Separator ซึ่งรูปแบบอยู่ในฟอร์ม เดียวกับรูปแบบการค้นหา บรรทัดที่มีตัวค้นย่อหน้า ถูกใช้เฉพาะเป็นตัวค้น; จะไม่มีการรวมไว้ในเอาต์พุต ตัวค้นย่อหน้าดีฟอลต์คือบรรทัดว่าง
-q	ยับยั้งเอาต์พุตทั้งหมดที่เอาต์พุตมาตรฐาน ไม่ว่าบรรทัด จะตรงกันหรือไม่จบ ด้วยสถานะ 0 ถ้าบรรทัดอินพุตถูกเลือก
-s	แสดงเฉพาะข้อความแสดงความผิดพลาด นี้มีประโยชน์สำหรับการตรวจสอบ สถานะ
-u	ทำให้เอาต์พุตถูกยกเลิกบัฟเฟอร์
-v	แสดงบรรทัดทั้งหมดยกเว้นบรรทัดที่ตรงกับรูปแบบ ที่ระบุ
-w	ทำการคนหาคำ
-x	แสดงบรรทัดที่ตรงกับรูปแบบที่ระบุอย่างชัดเจน โดยไม่มีอักขระเพิ่มเติม
-y	ละเว้นตัวพิมพ์ของตัวอักษร เมื่อทำการเปรียบเทียบ

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	พบข้อมูลตรงกัน
1	ไม่พบข้อมูลตรงกัน
>1	พบข้อผิดพลาดทางไวยากรณ์หรือไม่เข้าถึงไม่ได้ (แม้ว่า จะพบข้อมูลที่ตรงกัน)

## ตัวอย่าง

เมื่อต้องการใช้รูปแบบส่วนขยายที่มีอักขระ การจับคู่รูปแบบ บางส่วน +, ?, |, (, and), ให้ป้อน:

```
egrep "\([([A-z]+|[0-9]+)\)" my.txt
```

นี้จะแสดงบรรทัดที่มีตัวอักษรใน วงเล็บหรือตัวเลขในวงเล็บ แต่ไม่ใส่วงเล็บกับการรวมกันของ ตัวอักษรและตัวเลข โดยจะจับคู่ (y) และ (783902), แต่ไม่ใช่ (alpha19c)

**หมายเหตุ:** เมื่อใช้คำสั่ง `egrep`, \ (backslash ตามด้วย วงเล็บเปิด) หรือ \ (backslash ตามด้วยวงเล็บปิด) จับคู่วงเล็บในข้อความ แต่ ( (วงเล็บเปิด) และ ) (วงเล็บ ปิด) เป็นอักขระพิเศษที่จับกลุ่มส่วนของรูปแบบ การย้อนกลับเป็น true เมื่อใช้คำสั่ง `grep`

## ไฟล์

ไอเท็ม  
/usr/bin/egrep  
/bin/egrep

คำอธิบาย  
มีอาร์ดลิงก์ไปที่คำสั่ง egrep  
ระบุลิงก์สัญลักษณ์ไปที่คำสั่ง egrep

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง grep” ในหน้า 736

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง awk

คำสั่ง Shells

ภาพรวม National Language Support

---

## คำสั่ง eimadmin

### วัตถุประสงค์

จัดการโดเมน Enterprise Identity Mapping (EIM)

### ไวยากรณ์

```
eimadmin -a | -p | -l | -m | -e -D | -R | -I | -A | -C [-s switch] [-v verboseLevel] [-c accessType] [-f accessUserType] [-g registryParent] [-i identifier] [-j otherIdentifier] [-k URI] [-n description] [-o information] [-q accessUser] [-r registryName] [-t associationType] [-u registryUser] [-x registryAlias] [-y registryType] [-z registryAliasType] [-d domainDN] [-h ldapHost] [-b bindDN] [-w bindPassword] [-K keyFile [-P keyFilePassword] [-N certificateLabel]] [-S connectType]
```

### คำอธิบาย

คำสั่ง **eimadmin** เป็นเครื่องมือ AIX System Services Shell ผู้ดูแลระบบสามารถใช้เพื่อกำหนดโดเมน EIM และไพร้มโดเมนด้วย รีจิสตรี, identifiers และความสัมพันธ์ระหว่างผู้ใช้ identifiers และ รีจิสตรี ผู้ดูแลระบบยังสามารถใช้ **eimadmin** เพื่อให้ผู้ใช้ (และผู้ดูแลระบบคนอื่น) เข้าถึงโดเมน EIM หรือแสดงหรือลบ EIM entities

ผู้ดูแลระบบสามารถใช้คำสั่ง **eimadmin** ในสองวิธี:

- โดยการรวมข้อมูลที่มีอ็อปชันบรรทัดคำสั่งในคำสั่ง **eimadmin**
- โดยการรวมข้อมูลในอินพุตไฟล์ที่คำสั่ง **eimadmin** อ้างอิง

คุณสามารถสร้างไฟล์ด้วยตัวเองหรือโดยการเอ็กซ์พอร์ตเรกคอร์ดจากฐานข้อมูล ผู้ดูแลระบบกำหนดการประมวลผลยูทิลิตี้ โดยการระบุ อ็อปชัน บรรทัดคำสั่งรวมกัน

คำสั่ง **eimadmin** สามารถดำเนินการต่อไปนี้:

- เพิ่มอ็อบเจกต์ (-a)
- ลบอ็อบเจกต์ (-d)
- แสดงอ็อบเจกต์ (-l)
- แก้ไขแอตทริบิวต์ที่เชื่อมโยงกับอ็อบเจกต์ (-m)

- ลบแอ็ททริบิวต์ (-e)

กับอ็อบเจ็กต์ต่อไปนี้:

- Domains (-D)
- Registries (-R)
- Identifiers (-I)
- Associations (-A)
- Access authorities (-C)

หมายเหตุ:

1. แต่ละคำสั่ง **eimadmin** ต้องมีหนึ่งการดำเนินการและหนึ่งชนิด อ็อบเจ็กต์ ขึ้นกับอ็อบเจ็กต์และการดำเนินการที่คุณกำลังทำอยู่, EIM อาจต้องการพารามิเตอร์เพิ่มเติม
2. บางอ็อบชันมีไว้สำหรับแอ็ททริบิวต์ที่มีหลายค่า ซึ่งคุณสามารถระบุได้มากกว่า หนึ่งครั้ง อ็อบชันอื่นมีไว้สำหรับแอ็ททริบิวต์ค่าเดียว ซึ่งคุณสามารถระบุได้เพียง หนึ่งเดียว (ถ้าคุณกำหนดค่าอ็อบชันซ้ำสำหรับแอ็ททริบิวต์ค่าเดียว, **eimadmin** ประมวลผลเฉพาะค่าแรกที่พบในคำสั่ง) นอกจากข้อกำหนดเหล่านี้ ลำดับที่คุณระบุพารามิเตอร์ ไม่มีความสำคัญ
3. คุณสามารถโค๊ดพารามิเตอร์ของคำสั่ง **eimadmin** ได้หลายวิธี:
  - เชื่อมต่อการดำเนินการและอ็อบเจ็กต์, ละเว้นเครื่องหมายขีดคั่นที่ฝังตัว: -aD
  - รวมสองเครื่องหมายขีดคั่น, และแยกสองอ็อบชันด้วยช่องว่าง: -a -D

อีกนัยหนึ่ง ตัวอย่างต่อไปนี้ ไม่ถูกต้อง เนื่องจากมีทั้งเครื่องหมายขีดคั่นและไม่มีช่องว่างก่อน -D: -a -D

## แฟล็ก

คำสั่ง **eimadmin** รับแฟล็กดำเนินการต่อไปนี้

ไอเท็ม	คำอธิบาย
-a	เพิ่มอ็อบเจ็กต์ (สร้างนิยามอ็อบเจ็กต์และ แอ็ททริบิวต์)
-e	ลบแอ็ททริบิวต์ (ลบแอ็ททริบิวต์ค่าเดียว หรือลบแอ็ททริบิวต์หลายค่า)
-l	แสดงอ็อบเจ็กต์ (เรียกนิยามอ็อบเจ็กต์และ แอ็ททริบิวต์)
-m	แก้ไขแอ็ททริบิวต์ (เปลี่ยนแอ็ททริบิวต์ของอ็อบเจ็กต์ ที่มีอยู่ โดยการเปลี่ยนแอ็ททริบิวต์ค่าเดียวหรือเพิ่มแอ็ททริบิวต์ หลายค่า)
-p	ลบอ็อบเจ็กต์ (ลบนิยามอ็อบเจ็กต์และ แอ็ททริบิวต์)

คำสั่ง **eimadmin** รับอ็อบเจ็กต์แฟล็กต่อไปนี้

ไอเท็ม	คำอธิบาย
-A	ความสัมพันธ์ นี้เป็นความสัมพันธ์ระหว่าง identifier ในโดเมน EIM และ ID ผู้ใช้
-C	สิทธิการเข้าถึง นี้เป็นกลุ่มควบคุมการเข้าถึง EIM-defined LDAP
-D	โดเมน นี้เป็นคอลเล็กชันของ identifiers, รีจิสตรี ผู้ใช้ และความสัมพันธ์ระหว่าง identifiers และ ID ผู้ใช้, ที่เก็บภายใน โดเร็กทอรี LDAP
-I	identifier นี้เป็นชื่อของบุคคลหรือ entity ที่รวมในโดเมน EIM
-R	รีจิสตรี นี้เป็นชื่อของรีจิสตรีผู้ใช้ ความสัมพันธ์ ถูกกำหนดระหว่าง identifiers และ ID ผู้ใช้ ในรีจิสตรีผู้ใช้

คำสั่ง **eimadmin** รับแฟล็กการควบคุมกระบวนการต่อไปนี้

ไอเท็ม  
-s switch

คำอธิบาย  
switch ระบุค่าที่มีผลต่อการทำงาน ฟังก์ชันคำสั่ง eimadmin คุณสามารถระบุค่าต่อไปนี้:

#### RMDEPS

ลบความเชื่อมโยงเมื่อลบโดเมนหรือรีจิสตรีระบบ ซึ่งทำให้ง่ายต่อการลบโดเมนโดยชั้นแรกลบ identifiers และรีจิสตรีทั้งหมดที่กำหนดสำหรับโดเมน และยังทำให้ง่ายต่อการลบรีจิสตรีระบบ โดยชั้นแรกลบแอ็พพลิเคชันรีจิสตรีทั้งหมดที่กำหนดสำหรับรีจิสตรี

ข้อควรสนใจ: ข้อควรใส่ใจ: คำสั่ง eimadmin ไม่เตือนคุณ ว่ามีความเชื่อมโยงอยู่ ก่อนทำการลบ ดังนั้นโปรดใช้สวิตช์นี้ด้วยความระมัดระวัง

-v verboseLevel

พารามิเตอร์ verboseLevel เป็นจำนวนเต็มจาก 1 ถึง 10 ที่ควบคุมจำนวนของการติดตามรายละเอียดที่คำสั่ง eimadmin แสดง (มีไว้สำหรับการวินิจฉัยปัญหาในยูทิลิตี้ eimadmin) ค่าดีฟอลต์ 0 หมายถึงไม่มีข้อมูลการติดตาม คุณสามารถระบุ ค่าจำนวนเต็มจาก 1 ถึง 10, จากจำนวนน้อยที่สุดไปถึงจำนวนมากที่สุดของข้อมูลการติดตาม ยูทิลิตี้ ตรวจสอบค่าและแสดงข้อมูลการติดตามที่กำหนดสำหรับ ระดับและระดับที่ต่ำกว่าทั้งหมด ระดับต่อไปนี้ทริกเกอร์ข้อมูลจำเพาะ:

- 3- หมายถึง EIM API เรียกพารามิเตอร์และส่งคืนค่า
- 6- หมายถึงค่าอ็อปชันและไฟล์เลเบลอินพุต
- 9- หมายถึงรายการรูนยูทิลิตี้และคำสั่ง exit

คำสั่ง eimadmin รับแฟล็ก แอ็ททริบิวต์ที่จำเป็นและที่เป็นทางเลือก ที่แสดงในตารางต่อไปนี้ อ็อปชันแฟล็กเป็นค่าเดี่ยวยกเว้นมีการระบุเป็นอย่างอื่น ถ้าคุณระบุอ็อปชันมากกว่าหนึ่งครั้ง ยูทิลิตี้ ประมวลผลเฉพาะอ็อปชันแรก

#### หมายเหตุ:

1. คุณสามารถระบุแอ็ททริบิวต์เหล่านี้เป็นอ็อปชันคำสั่งหรือเป็นฟิลด์ในอินพุต ไฟล์ ถ้าคุณระบุอ็อปชันคำสั่ง คุณต้องค่าที่มีช่องว่างให้อยู่ภายในเครื่องหมายคำพูด ("") หรือ (') เครื่องหมายคำพูดเป็นทางเลือก สำหรับค่าคำเดียว การระบุค่า multiword โดยไม่มีเครื่องหมายคำพูด จะมีการตัดอ็อปชันบรรทัดคำสั่ง; ค่าหลังจากคำแรก จะถูกตัด
2. อักขระพิเศษต่อไปนี้ใช้ไม่ได้ใน registryName, registryParent หรือ identifier:

, = + < > # ; \ \*

ไอเท็ม  
-c accessType

คำอธิบาย  
ระบุขอบเขตของสิทธิ์การเข้าถึงที่ผู้ใช้มีในโดเมน EIM accessType ต้องมีค่าหนึ่งในค่าต่อไปนี้:

ADMIN ระบุการเข้าถึงเพื่อดูและระบบ

#### REGISTRY

ระบุการเข้าถึงรีจิสตรี ถ้าคุณระบุ REGISTRY, คุณต้อง ระบุค่ารีจิสตรีเช่นกัน (-r) ค่ารีจิสตรีสามารถเป็นชื่อรีจิสตรีเฉพาะ หรือเป็นเครื่องหมายดอกจัน (\*) เพื่อระบุการเข้าถึงรีจิสตรีทั้งหมด

#### IDENTIFIER

ระบุการเข้าถึง identifier

#### MAPPING

ระบุการแม็พการเข้าถึงการดำเนินการ  
ระบุชนิดของชื่อผู้ใช้งานการเข้าถึง accessUserType ต้องมีค่าหนึ่งในค่าต่อไปนี้:

-f accessUserType

DN accessUser เป็น distinguished name

#### KERBEROS

accessUser เป็น Kerberos identity

## ไอเท็ม

**-g registryParent**  
**-i identifier**  
**-j otherIdentifier**  
**-k URI**  
**-n คำอธิบาย**  
**-o information**  
**-q accessUser**  
**-r registryName**  
**-t associationType**

## คำอธิบาย

ระบุชื่อของรีจิสตรีระบบ แอ็พพลิเคชัน รีจิสตรีเป็นเซตย่อยของรีจิสตรีระบบ ถ้าคุณเพิ่มแอ็พพลิเคชันรีจิสตรี คุณต้องใช้อ็อปชัน **-r** และอ็อปชัน **-g** ค่า **-r** เป็นแอ็พพลิเคชันรีจิสตรีที่คุณกำหนด อ็อปชัน **-g** เป็นรีจิสตรีระบบที่มีอยู่ก่อน

ระบุชื่อ identifier เฉพาะ ตัวอย่าง: John Day

ระบุชื่อ identifier ที่เข้าได้ ตัวอย่าง: John

**หมายเหตุ:** คุณสามารถระบุอ็อปชันนี้ได้หลายครั้งเพื่อกำหนด identifiers ที่เข้าได้หลายค่า

ระบุ Universal Resource Identifier (URI) สำหรับ รีจิสตรี (ถ้ามี)

ระบุข้อความ (ที่คุณกำหนด) เพื่อเชื่อมโยงกับ โดเมน รีจิสตรี identifier หรือความสัมพันธ์

**หมายเหตุ:** คุณสามารถกำหนดรายละเอียดผู้ใช้เฉพาะสำหรับความสัมพันธ์เป้าหมาย

ระบุข้อมูลเพิ่มเติมเพื่อเชื่อมโยงกับ identifier หรือความสัมพันธ์

**หมายเหตุ:** คุณสามารถกำหนดข้อมูลผู้ใช้เฉพาะกับความสัมพันธ์เป้าหมาย คุณสามารถระบุอ็อปชันนี้ได้หลายครั้งเพื่อกำหนดส่วนของข้อมูลหลายส่วน

ระบุ distinguished name (DN) หรือ Kerberos identity ที่มีการเข้าถึง EIM, ขึ้นกับ *accessUserType* ที่ระบุ

ระบุชื่อของรีจิสตรี เมื่อคุณเพิ่มรีจิสตรีใหม่, **eimadmin** ปฏิบัติต่อรีจิสตรีเป็นรีจิสตรีระบบ นอกจากนี้คุณได้ระบุอ็อปชัน

**-g** ด้วย ถ้าคุณระบุอ็อปชัน **-g**, **eimadmin** ปฏิบัติต่อรีจิสตรีเป็นแอ็พพลิเคชันรีจิสตรี

ระบุความสัมพันธ์ระหว่าง identifier และ รีจิสตรี *associationType* ต้องเป็นหนึ่งในค่าต่อไปนี้:

**ADMIN** ระบุความสัมพันธ์ของ ID ผู้ใช้ กับ identifier สำหรับจุดประสงค์ทางการดูแลระบบ

## SOURCE

ระบุว่า ID ผู้ใช้ เป็นซอร์ส (หรือจาก) ของการดำเนินการค้นหา

## TARGET

ระบุว่า ID ผู้ใช้ เป็นปลายทาง (หรือถึง) ของการดำเนินการค้นหา

**หมายเหตุ:** คุณสามารถระบุอ็อปชันนี้ได้หลายครั้งเพื่อกำหนด หลายความสัมพันธ์

ระบุ ID ผู้ใช้ ของผู้ใช้ที่กำหนดในรีจิสตรี

ระบุชื่ออื่นสำหรับรีจิสตรี คุณต้องระบุ อ็อปชันนี้หลายครั้งเพื่อกำหนดหลาย aliases

ระบุชนิดของรีจิสตรี ชนิดกำหนดลงหน้าที่ **eimadmin** รู้จักรวมทั้งค่าต่อไปนี้:

- RACF<sup>®</sup>
- OS/400<sup>®</sup>
- KERBEROS (สำหรับการละเว้นขนาดตัวพิมพ์)
- KERBEROSX (ตรงตามตัวพิมพ์)
- AIX
- NDS
- LDAP
- PD (Policy Director)
- WIN2K

คุณยังสามารถสร้างชนิดของคุณเอง โดยการเชื่อม unique OID ด้วยหนึ่งในสองเมธอดการทำให้เป็นกรณีปกติต่อไปนี้:

- caseIgnore
- caseExact

**-u registryUser**  
**-x registryAlias**  
**-y registryType**

## ไอเอ็ม

`-z registryAliasType`

### คำอธิบาย

ระบุชนิดสำหรับรีจิสตรี `alias` คุณสามารถสร้าง ค่าของคุณเองหรือใช้หนึ่งในค่าที่แนะนำต่อไปนี้:

- `DNSHostName`
- `KerberosRealm`
- `IssuerDN`
- `RootDN`
- `TCPIPAddress`
- `LdapDnsHostName`

**หมายเหตุ:** สำหรับชุดอ็อปชันบรรทัดคำสั่งหรือเร็กคอร์ดข้อมูลอินพุต เดียว คำสั่ง `eimadmin` รู้จักเฉพาะค่ากำหนดแรก ของ `registryAliasType` อย่างไรก็ตามคำสั่ง `eimadmin` รู้จักหลายรีจิสตรี `aliases` และเชื่อมโยงทั้งหมดด้วย `registryAliasType` ชนิดเดียว

## คำสั่ง `eimadmin` รับแฟล็กชนิดการเชื่อมต่อต่อไปนี้

### ไอเอ็ม

`-b bindDN`

`-d domainDN`

### คำอธิบาย

ระบุ distinguished name ที่จะใช้สำหรับการเชื่อมต่อ ธรรมดา กับ LDAP

ระบุ distinguished name (DN) แบบเต็มของโดเมน EIM `domainDN` เริ่มต้นด้วย ' `ibm-eimDomainName='` และประกอบด้วยส่วนประกอบต่อไปนี้:

`domainName`

ชื่อของโดเมน EIM ที่คุณกำลังสร้าง ตัวอย่าง `MyDomain`

### พารามิเตอร์ distinguished name

distinguished name สำหรับรายการที่อยู่เหนือรายการที่กำหนด ในลำดับชั้นผังข้อมูลไดเรกทอรี เช่น `o=ibm,c=us` ตัวอย่างเช่น:

`ibm-eimDomainName=MyDomain,o=ibm,c=us`

`-h ldapHost`

ระบุ URL และพอร์ตสำหรับเซิร์ฟเวอร์ LDAP เพื่อควบคุม ข้อมูล EIM รูปแบบ คือ:

`ldap://some.ldap.host:389`

`ldaps://secure.ldap.host:636`

`-K keyFile`

ระบุชื่อของไฟล์ฐานข้อมูลคีย์ SSL รวมทั้งชื่อพารามิเตอร์ ถ้าไม่พบไฟล์ จะถือว่าเป็นชื่อของ คีย์ริง RACF ที่มีใบรับรองการพิสูจน์ตัวตน คำนี้อาจเป็นการสื่อสาร SSL กับโฮสต์ secure LDAP (นำหน้าด้วย `ldaps://`) ตัวอย่างเช่น:

`/u/eimuser/ldap.kdb`

`-N certificateLabel`

ระบุใบรับรองที่จะใช้จากไฟล์ฐานข้อมูลคีย์ หรือคีย์ริง RACF ถ้าอ็อปชันนี้ไม่ถูกระบุ ใบรับรอง ทำเครื่องหมายเป็น ดีฟอลต์ในไฟล์หรือริงถูกใช้

`-P keyFilePassword`

ระบุรหัสผ่านที่จำเป็นในการเข้าถึงข้อมูลที่เข้ารหัส ในไฟล์ฐานข้อมูลคีย์ อีกวิธีหนึ่ง คุณสามารถระบุไฟล์ stash ของรหัสผ่าน SSL สำหรับอ็อปชันนี้ โดยนำหน้าชื่อไฟล์ stash ด้วย `file://` ตัวอย่างเช่น:

`secret or file:///u/eimuser/ldapclient.sth`

**หมายเหตุ:** คำสั่ง `eimadmin` พร้อมต์ของรหัสผ่านไฟล์คีย์ ถ้าคุณระบุชื่อของไฟล์ฐานข้อมูลคีย์สำหรับอ็อปชัน `-K` แต่ไม่ใช่อ็อปชัน `-P` ในบรรทัดคำสั่ง

ไอเอ็ม  
-S connectType

คำอธิบาย  
ระบุเมธอดของการพิสูจน์ตัวตนกับเซิร์ฟเวอร์LDAP connectType ต้องเป็นหนึ่งใน ค่าต่อไปนี้:

- SIMPLE (เชื่อมโยง DN และรหัสผ่าน)
- CRAM-MD5 (เชื่อมโยง DN และรหัสผ่านที่ป้องกัน)
- EXTERNAL (ใบรับรองดิจิทัล)
- GSSAPI (Kerberos)

ถ้าไม่ระบุ, connectType ดีฟอลต์เป็น SIMPLE สำหรับชนิดการเชื่อมต่อ GSSAPI, ดีฟอลต์ Kerberos credential ถูกนำมาใช้ credential ต้องถูกสร้างโดยใช้เซอริสเช่น kinit ก่อน ที่จะรัน eimadmin สำหรับ KINIT และข้อมูลที่เกี่ยวข้องโปรดอ้างอิง AIX Authentication Service Administration ระบุรหัสผ่านที่เชื่อมโยงกับ bind DN

-w bindPassword

ข้อมูลการเชื่อมต่อที่จำเป็นต่อยูทิลิตี้ รวมถึงโดเมน EIM (-d) และเซิร์ฟเวอร์การควบคุม (-h), identity (-b, -w; or -K, -P, -N) การการพิสูจน์ตัวตน (bind) กับเซิร์ฟเวอร์และเมธอดการพิสูจน์ตัวตน (-S)

สำหรับชนิดอ็อบเจกต์ที่ไม่ใช่โดเมน (-D), การระบุโดเมน, เซิร์ฟเวอร์และ bind identity เป็นทางเลือก ถ้าไม่มีการระบุข้อมูลเหล่านี้ ข้อมูลจะถูกเรียกมาจากโปรไฟล์ RACF

หมายเหตุ: ถ้ามีการระบุข้อมูลเชื่อมต่อ ชุดของค่าที่สมบูรณ์สำหรับชนิดการเชื่อมต่อ ต้องถูกระบุเช่นกัน การละเว้นค่าหนึ่งค่าหรือมากกว่านั้น (แต่ไม่ใช่ทั้งหมด) จะทำให้ มีข้อผิดพลาด ตารางต่อไปนี้แสดงค่าที่จำเป็นและที่เป็นทางเลือกสำหรับแต่ละการเชื่อมต่อและชนิดโฮสต์ เมื่อระบุด้วยคำสั่ง eimadmin

ชนิดการเชื่อมต่อ/ชนิดโฮสต์	ค่าที่ต้องการ	ค่าทางเลือก
SIMPLE หรือ CRAM-MD5/secure (ldaps://)	-d, -h, -b, -w, -K, -P	-N
SIMPLE หรือ CRAM-MD5/nonsecure (ldap://)	-d, -h, -b, -w	
EXTERNAL/secure (ldaps://)	-d, -h, -K, -P, -S	-N
EXTERNAL/nonsecure (ldap://)	ไม่สนับสนุน	ไม่สนับสนุน
GSSAPI/secure (ldaps://)	-d, -h, -K, -P, -S	-N
GSSAPI/nonsecure (ldap://)	-d, -h, -S	

หมายเหตุ:

1. มีสองข้อยกเว้นกับตารางเบื้องต้น:

- โดเมนอ็อบเจกต์ (-d) ไม่จำเป็นสำหรับโดเมนฟังก์ชัน ถ้า ค่าถูกระบุผ่านอินพุตไฟล์
- รหัสผ่านไฟล์ฐานข้อมูลคือ SSL หรือไฟล์ stash (-P) ไม่จำเป็นเมื่อ -K ระบุคีย์ริง RACF

2. คำสั่ง eimadmin พร้อมสำหรับรหัสผ่านเชื่อมโยงธรรมดา ถ้าจำเป็น และไม่มีการระบุ -w ในบรรทัดคำสั่ง และพร้อมสำหรับรหัสผ่านไฟล์ฐานข้อมูล SSL ถ้ามีความจำเป็นและ -P ไม่ถูกระบุในบรรทัดคำสั่ง

ตารางต่อไปนี้สรุปแฟล็กที่จำเป็นและที่เป็นทางเลือก สำหรับแต่ละชนิดอ็อบเจกต์ และคู่มือการดำเนินการ คุณสามารถระบุค่าสำหรับอ็อบเจกต์ส่วนใหญ่ในอินพุตไฟล์ แทนการระบุในบรรทัดคำสั่ง

ชนิดอ็อบเจกต์ (การดำเนินการ)	แฟล็ก	หมายเหตุ
D (a)	<ul style="list-style-type: none"> <li>ต้องการ: d, h</li> <li>เป็นทางเลือก: n</li> </ul>	เพิ่มโดเมน
D (p)	<ul style="list-style-type: none"> <li>ต้องการ: d, h</li> <li>เป็นทางเลือก: s</li> </ul>	ลบโดเมน ถ้าโดเมนไม่ว่าง ให้เพิ่ม -s RMDEPS
D (l)	<ul style="list-style-type: none"> <li>ต้องการ: d, h</li> <li>เป็นทางเลือก:</li> </ul>	แสดงโดเมน ระบุ -d* เพื่อแสดงโดเมนทั้งหมด
D (m)	<ul style="list-style-type: none"> <li>ต้องการ: d, h</li> <li>เป็นทางเลือก: n</li> </ul>	แก้ไขหรือเพิ่มโดเมนแอตทริบิวต์
D (e)	<ul style="list-style-type: none"> <li>ต้องการ: d, h</li> <li>เป็นทางเลือก: n</li> </ul>	ลบหรือล้างโดเมนแอตทริบิวต์
R (a)	<ul style="list-style-type: none"> <li>ต้องการ: r, y</li> <li>เป็นทางเลือก: g, k, n, x, z</li> </ul>	เพิ่มรีจิสตรี ค่าที่ระบุสำหรับ -r ถือว่า เป็นรีจิสตรีระบบใหม่ นอกจากมีการระบุ -g ด้วย ในกรณีนี้ค่า -r ระบุแอ็พพลิเคชันรีจิสตรีใหม่
R (p)	<ul style="list-style-type: none"> <li>ต้องการ: r</li> <li>เป็นทางเลือก: s</li> </ul>	ลบรีจิสตรี
R (l)	<ul style="list-style-type: none"> <li>ต้องการ: r</li> <li>เป็นทางเลือก: y</li> </ul>	แสดงรีจิสตรี ส่งคืนรายการรีจิสตรีทั้งหมดใน โดเมนที่ตรงกับตัวกรองการค้นหาค่า -r ที่ระบุ ซึ่งอาจ มี wild card *
R (m)	<ul style="list-style-type: none"> <li>ต้องการ: r</li> <li>เป็นทางเลือก: k, n, x, z</li> </ul>	แก้ไขหรือเพิ่มแอตทริบิวต์รีจิสตรี, รวมถึงรีจิสตรี alias
R (e)	<ul style="list-style-type: none"> <li>ต้องการ: r</li> <li>เป็นทางเลือก: k, n, x, z</li> </ul>	ลบหรือล้างแอตทริบิวต์รีจิสตรี, รวมถึงรีจิสตรี alias
I (a)	<ul style="list-style-type: none"> <li>ต้องการ: i</li> <li>เป็นทางเลือก: j, n, o</li> </ul>	เพิ่ม identifier
I (p)	<ul style="list-style-type: none"> <li>ต้องการ: i</li> <li>เป็นทางเลือก:</li> </ul>	ลบ identifier
I (l)	<ul style="list-style-type: none"> <li>ต้องการ: i</li> <li>เป็นทางเลือก:</li> </ul>	แสดง identifier ตามชื่อ identifier เฉพาะ ส่งคืน รายการ identifier ทั้งหมดในโดเมนที่ตรงกับตัวกรองการค้นหาค่า -i ที่ระบุ ซึ่งอาจ มี wild card *
I (l)	<ul style="list-style-type: none"> <li>ต้องการ: j</li> <li>เป็นทางเลือก:</li> </ul>	แสดง identifier ตามชื่อ identifier ที่เข้าได้ ส่งคืน รายการ identifier ทั้งหมดในโดเมนที่มีชื่อ identifier ที่เข้าได้ตรงกับตัวกรองการค้นหาค่า -j ที่ระบุ ซึ่งอาจ มี wild card *
I (m)	<ul style="list-style-type: none"> <li>ต้องการ: i</li> <li>เป็นทางเลือก: j, n, o</li> </ul>	แก้ไขหรือเพิ่มแอตทริบิวต์ identifier

ชนิดอ็อบเจกต์ (การดำเนินการ)	แฟล็ก	หมายเหตุ
I (e)	<ul style="list-style-type: none"> <li>ต้องการ: i</li> <li>เป็นทางเลือก: j, n, o</li> </ul>	ลบหรือล้างแอตทริบิวต์ identifier
A (a)	<ul style="list-style-type: none"> <li>ต้องการ: i, r, u, t</li> <li>เป็นทางเลือก: n, o</li> </ul>	เพิ่มความสัมพันธ์ คุณสามารถใช้อ็อบชัน -t เพื่อเพิ่มชนิดความสัมพันธ์หลายรายการ แฟล็ก -n และ -o สัมพันธ์เฉพาะกับความสัมพันธ์ TARGET
A (p)	<ul style="list-style-type: none"> <li>ต้องการ: i, r, u, t</li> <li>เป็นทางเลือก:</li> </ul>	ลบความสัมพันธ์ คุณสามารถใช้อ็อบชัน -t เพื่อลบชนิดความสัมพันธ์หลายรายการ
A (l)	<ul style="list-style-type: none"> <li>ต้องการ: i</li> <li>เป็นทางเลือก: t</li> </ul>	แสดงความสัมพันธ์ ส่งคืนความสัมพันธ์ทั้งหมดในโดเมน สำหรับ identifier เฉพาะ -i ที่ระบุ บุคคล -t เพื่อจำกัดรายการที่ส่งกลับเป็นชนิดความสัมพันธ์ที่กำหนด
A (m)	<ul style="list-style-type: none"> <li>ต้องการ: r, u</li> <li>เป็นทางเลือก: n, o</li> </ul>	แก้ไขหรือเพิ่มแอตทริบิวต์ความสัมพันธ์ แฟล็ก -n และ -o สัมพันธ์เฉพาะกับความสัมพันธ์ TARGET
A (e)	<ul style="list-style-type: none"> <li>ต้องการ: r, u</li> <li>เป็นทางเลือก: n, o</li> </ul>	ลบหรือล้างแอตทริบิวต์ความสัมพันธ์ แฟล็ก -n และ -o สัมพันธ์เฉพาะกับความสัมพันธ์ TARGET
C (a)	<ul style="list-style-type: none"> <li>ต้องการ: c, q, f</li> <li>เป็นทางเลือก: r</li> </ul>	เพิ่มการเข้าถึง สำหรับชนิดการเข้าถึง REGISTRY, จัดเตรียมคำรีจิสตรี -r หรือ wild card * ระบุการเข้าถึงกับ รีจิสตรีทั้งหมดในโดเมน
C (p)	<ul style="list-style-type: none"> <li>ต้องการ: c, q, f</li> <li>เป็นทางเลือก: r</li> </ul>	ลบการเข้าถึง สำหรับชนิดการเข้าถึง REGISTRY, จัดเตรียมคำรีจิสตรี -r หรือ wild card * ระบุการเข้าถึงกับ รีจิสตรีทั้งหมดในโดเมน
C (l)	<ul style="list-style-type: none"> <li>ต้องการ: c</li> <li>เป็นทางเลือก: r</li> </ul>	แสดงการเข้าถึงตามชนิด สำหรับชนิดการเข้าถึง REGISTRY, จัดเตรียมคำรีจิสตรี -r หรือ wild card * ระบุการเข้าถึงกับ รีจิสตรีทั้งหมดในโดเมน
C (l)	<ul style="list-style-type: none"> <li>ต้องการ: q, f</li> <li>เป็นทางเลือก:</li> </ul>	แสดงการเข้าถึงตามผู้ใช้

## สถานะออก

คำสั่ง eimadmin ส่งคืนโค้ดจบการทำงานต่อไปนี้ เมื่อทำงานเสร็จสมบูรณ์:

0	ไอเท็ม คำอธิบาย
0	สำเร็จ
4	พบข้อผิดพลาด แต่ถ้าคุณระบุ อินพุตไฟล์ เร็กคอร์ดทั้งหมดจะถูกประมวลผล
8	มีข้อผิดพลาดร้ายแรงเกิดขึ้น ที่ทำให้การประมวลผลหยุดลง ก่อนถึงจุดสิ้นสุดของสุดท้ายของอินพุตไฟล์, ถ้ามีการระบุ

## ตัวอย่าง

1. เมื่อต้องการแสดงโดเมนเดียว พิมพ์:

```
eimadmin -lD -h ldap://my.server -b "cn=EIM admin,o=MyCompany,c=US" -d "ibm-eimDomainName=My Employees,o=My Company,c=US"
```

นี้จะส่งคืนข้อมูลเหมือนเอาต์พุตต่อไปนี้:

domain name: My Employees  
domain DN: ibm-eimDomainName=My Employees,o=My Company,c=US  
คำอธิบาย: employees in my company

2. เมื่อต้องการแสดงรีจิสตรีเดียว พิมพ์:

```
eimadmin -lR -r MyRegistry
```

นี้จะส่งคืนข้อมูลเหมือนเอาต์พุตต่อไปนี้:

```
registry: MyRegistry  
registry kind: APPLICATION  
registry parent: MySystemRegistry  
registry type: RACF  
คำอธิบาย: my racf registry  
URI: ldap://some.big.host:389/profileType=User,cn=RACFA,o=My Company,c=US  
registry alias: TCPGROUP  
registry alias type: DNSHostName
```

3. เมื่อต้องการแสดง identifiers พิมพ์:

```
eimadmin -lI -i "J.C.Smith"
```

นี้จะส่งคืนข้อมูลเหมือนเอาต์พุตต่อไปนี้:

```
unique identifier: J.C.Smith  
other identifier: J.C.Smith  
other identifier: Joseph  
other identifier: Joe  
คำอธิบาย: 004321  
information: D01  
information: 1990-04-11
```

4. เมื่อต้องการแสดงความสัมพันธ์เป้าหมาย พิมพ์:

```
eimadmin -lA -i "J.C.Smith" -t target
```

นี้จะส่งคืนข้อมูลเหมือนเอาต์พุตต่อไปนี้:

```
unique identifier: J.C.Smith  
registry: MyRegistry  
registry type: RACF  
association: target  
registry user: SMITH  
คำอธิบาย: TSO  
information: 1989-08-01  
information: ADMIN1
```

5. เมื่อต้องการแสดงการเข้าถึง พิมพ์:

```
eimadmin -lC -c admin
```

นี้จะส่งคืนข้อมูลเหมือนเอาต์พุตต่อไปนี้:

```
access user: cn=JoeUser,o=My Company,c=us  
access user: cn=admin1,o=My Company,c=us  
access user: cn=admin2,o=My Company,c=us
```

## Location

/usr/bin/eimadmin

## ความปลอดภัย

ผู้ดูแลระบบ LDAP มีสิทธิ์ในการใช้คำสั่ง `eimadmin` และเข้าถึงฟังก์ชันทั้งหมดที่มี ผู้ดูแลระบบ EIM สามารถใช้คำสั่ง `trab` ที่สภาวะต่อไปนี้เป็น `true`:

- มี `bind distinguished name` และรหัสผ่านที่กำหนดที่เซิร์ฟเวอร์ LDAP ที่มีโดเมน EIM
- `bind distinguished name` ของผู้ดูแลระบบมีหนึ่งในสิทธิ์ EIM:
  - EIM administrator
  - EIM registries administrator
  - EIM registry X administrator
  - EIM identifiers administrator

## ข้อผิดพลาดมาตรฐาน

คำสั่ง `eimadmin` ส่งข้อความเพื่อพร้อมตัวซอร์สผ่าน หรือแสดงข้อผิดพลาด โปรดอย่าคาดหวังว่าจะได้รับข้อความการทำงานสำเร็จ นอกจากคุณใช้อินพุตไฟล์ เมื่อประมวลผลเรียกคอร์ตในอินพุตไฟล์, `eimadmin` ส่งข้อความแจ้งข้อมูลเมื่อกระบวนการเริ่มต้น และหยุดทำงาน นอกเหนือจากข้อความการประมวลผลทุก 50 เรียกคอร์ต

หมายเหตุ: คำสั่ง `eimadmin` ส่งคืนบรรทัดข้อมูลสำหรับการร้องขอ `list (-1)` นอกจากจะไม่พบรายการ EIM ที่ตรงกัน, หรือ `bind identity` ไม่ได้รับอนุญาตให้เข้าถึงข้อมูลนั้น

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `eimadmin.conf`

---

## คำสั่ง `elogevent`

### วัตถุประสงค์

ข้อมูลเหตุการณ์บันทึกการทำงานที่สร้างโดย event response resource manager (ERRM) กับล็อกไฟล์ที่ระบุ

### ไวยากรณ์

`elogevent [-h] log_file`

### คำอธิบาย

`elogevent` จับข้อมูลเหตุการณ์ที่ถูกโพสต์โดย event response resource manager (ERRM) ในตัวแปรสภาวะแวดล้อม ERRM สร้างเมื่อเหตุการณ์เกิดขึ้น สคริปต์นี้สามารถถูกใช้เป็นการดำเนินการที่ถูกรัน โดยรีซอร์สการตอบกลับเหตุการณ์ และยังสามารถนำมาใช้เป็นเทมเพลตเพื่อสร้าง การดำเนินการที่ผู้ใช้กำหนดเอง อื่น สคริปต์นี้ส่งกลับข้อความเป็นภาษาอังกฤษเสมอ

ข้อมูลเหตุการณ์ที่ถูกส่งกลับเกี่ยวกับตัวแปรสภาวะแวดล้อม ERRM รวมถึงข้อมูลต่อไปนี้:

## Local Time

เวลาเมื่อเหตุการณ์หรือเวลาที่ผู้ใช้ใหม่ถูกพบ ตัวแปรสถานะแวดล้อมจริง ที่ระบุโดย ERRM คือ ERRM\_TIME ค่านี้ถูกแปลและแปลง เป็นรูปแบบที่อ่านได้ ก่อนถูกแสดง

สคริปต์นี้ใช้คำสั่ง `alog` เพื่อเขียนข้อมูลเหตุการณ์ และอ่านข้อมูลเหตุการณ์จากไฟล์ `log_file` ที่ระบุ

## แฟล็ก

`-h` เขียนคำสั่งการใช้สคริปต์ไปที่เอาต์พุตมาตรฐาน

## พารามิเตอร์

### `log_file`

ระบุชื่อไฟล์ที่ข้อมูลเหตุการณ์ถูกบันทึก พารามิเตอร์สำหรับพารามิเตอร์ `log_file` ควรถูกระบุ

`log_file` ถูกใช้เป็นบันทึกไว้เป็นวงรอบ และมีขนาดที่กำหนดไว้ นั่นคือ 64KB เมื่อ `log_file` เต็ม รายการใหม่ถูกเขียนทับรายการเก่าที่สุดที่มีอยู่

หาก `log_file` มีอยู่ก่อนแล้ว รายละเอียดของเหตุการณ์จะถูกต่อท้าย บันทึกการทำงานนั้น ถ้าไม่มี `log_file`, ไฟล์จะถูกสร้างเพื่อที่ข้อมูลเหตุการณ์ สามารถถูกเขียนลงไปได้

## สถานะออก

- 0 สคริปต์รันสำเร็จ
- 1 ไม่ได้ระบุ `log_file` ที่ต้องการ
- 2 พาร `log_file` ไม่ถูกต้อง

## ข้อจำกัด

- สคริปต์นี้ต้องรันบนโหนดที่ ERRM กำลังรัน
- ผู้ใช้ที่รันสคริปต์นี้ต้องมีสิทธิ์ในการเขียน `log_file` ซึ่งข้อมูลเหตุการณ์จะถูกบันทึก

## เอาต์พุตมาตรฐาน

เมื่อมีการระบุแฟล็ก `-h` คำสั่งการใช้สคริปต์จะถูก เขียนไปที่เอาต์พุตมาตรฐาน

## ตัวอย่าง

1. เมื่อต้องการบันทึกข้อมูลให้ระบุชื่อไฟล์เป็น `/tmp/event.log` ERRM รันคำสั่งนี้:  
`/usr/sbin/rsct/bin/logevent/tmp/event.log`

ไฟล์ `/tmp/event.log` ไม่จำเป็นต้องมีอยู่เมื่อคำสั่ง ถูกรัน

2. เมื่อต้องการดูเนื้อหาของไฟล์ `/tmp/event.log` ให้รันคำสั่งนี้:  
`alog -f /tmp/event.log -o`

เอาต์พุตตัวอย่างต่อไปนี้แสดงเหตุการณ์ค่าเตือน สำหรับระบบไฟล์ `/var` (รีซอร์สระบบไฟล์):

=====  
Event reported at Mon Mar 27 16:38:03 2007

Condition Name: /var space used  
Severity: Warning  
Event Type: Event  
Expression: PercentTotUsed>90

Resource Name: /var  
Resource Class Name: IBM.FileSystem  
Data Type: CT\_UINT32  
Data Value: 91

## Location

/usr/sbin/rsct/bin/elogevent

---

## คำสั่ง emgr

### วัตถุประสงค์

เริ่มต้น interim fix manager ซึ่งติดตั้ง ลบ แสดงรายการ และตรวจสอบโปรแกรมฟิซเฉพาะกิจของระบบ

### ไวยากรณ์

**emgr -l** [-L *Label*] [-n *interimfixNumber*] [-u *VUID*] [-v{ 1|2|3 }] [-X] [-a *path*]

**emgr -e** *interimfixPackage* | -f *ListFile* [-w *Directory*] [-b] [-k] [-I] [-p] [-q] [-m] [-o] [-X] [-a *path*]

**emgr -i** *interimfixPackage* | -f *ListFile* [-w *Directory*] [-C] [-p] [-q] [-X] [-a *path*]

**emgr -C -i** *interimfixPackage* | -f *ListFile* [-w *Directory*] [-p] [-q] [-X] [-a *path*]

**emgr -C -L** *Label* [-p] [-q] [-X]

**emgr -r** [-L *Label*] [-n *interimfixNumber*] [-u *VUID*] [-f *ListFile*] [-w *Directory*] [-a *path*] [-b] [-k] [-I] [-p] [-q] [-X]

**emgr -c** [-L *Label*] [-n *interimfixNumber*] [-u *VUID*] [-f *ListFile*] [-w *Directory*] [-a *path*] [-v{ 1|2|3 }] [-X]

**emgr -M** [-U] [-L *Label*] [-n *interimfixNumber*] [-u *VUID*] [-f *ListFile*] [-w *Directory*] [-a *path*] [-X]

**emgr -R** *interimfixLabel* [-w *Directory*] [-a *path*] [-X]

**emgr -P** [*Package*] [-a *path*] [-X]

**emgr -d -e** *interimfixPackage* | -f *ListFile* [-w *Directory*] [-v{ 1|2|3 }]

## คำอธิบาย

คำสั่ง **emgr** (interim fix manager) สามารถใช้เพื่อติดตั้งและจัดการโปรแกรมฟิกซ์เฉพาะกิจของระบบ ตัวจัดการ interim fix ติดตั้งแพ็คเกจที่สร้างด้วยคำสั่ง **epkg** และดูแลฐานข้อมูลที่มีข้อมูล interim fix คำสั่ง **emgr** ดำเนินปฏิบัติการต่อไปนี้:

- การติดตั้งแพ็คเกจ interim fix
- การลบ interim fix
- interim fix listing
- interim fix checking
- interim fix mounting
- การยกเลิกการเมาท์ interim fix
- แพ็คเกจล็อกการแสดงผล
- interim fix ที่ติดตั้งบังคับการถอนออก

### หมายเหตุ:

- ถ้ามีความพยายามในการอัปเดตชุดไฟล์ (โดยใช้คำสั่ง **installp**, **install\_all\_updates** หรือ **smit update\_all**) ที่ได้ถูกล็อกโดยตัวจัดการ interim fix การแจ้งเตือนจะถูกแสดงเพื่อระบุชุดไฟล์ที่ถูกล็อกในบางกรณี ไม่มีการแจ้งเตือนเพื่อระบุเหตุผลที่ชุดไฟล์ถูกป้องกันจากการถูกติดตั้ง คำสั่ง **lspp** แสดงว่าชุดไฟล์ที่ล็อกใดๆ อยู่ในสถานะ IFIXLOCKED
- โลบารีหรือโปรแกรมเรียกทำงานที่อัปเดตโดย interim fix หรือ เซอร์วิสอัปเดต ซึ่งถูกใช้โดยกระบวนการที่แอคทีฟจะยังไม่แสดงผล ในกระบวนการนั้น นอกจากกระบวนการจะถูกรีสตาร์ท ตัวอย่างเช่น การอัปเดตที่เปลี่ยนแปลง ksh จะไม่มีการเปลี่ยนแปลงแสดงในกระบวนการ ksh ใดๆ ที่กำลังรันอยู่ในทำนองเดียวกัน การอัปเดตกับโลบารี **libc.a** จะไม่แสดงในกระบวนการใดๆ ที่กำลังรันอยู่นอกจากนั้น กระบวนการใดๆ ที่กำลังใช้โลบารีและมีการดำเนินการ **dlopen** ของโลบารีเดียวกันนั้น หลังจากโลบารีได้ถูกอัปเดตอาจพบความไม่สอดคล้องกันหากไม่ถูกรีสตาร์ท

## การอ้างอิง Ifix

วิธีในการอ้างอิง interim fix มีดังนี้:

### อ้างอิงด้วยเลเบล

แต่ละ interim fix ที่ถูกติดตั้งบนระบบที่กำหนด จะมีเลเบล interim fix เฉพาะ นี่เป็นคีย์เฉพาะที่เชื่อมโยงอ็อบเจกต์ฐานข้อมูลที่แตกต่างกันทั้งหมด เมื่อต้องการอ้างอิง interim fix ตามเลเบล ให้ส่งเลเบลเป็นพารามิเตอร์กับแฟล็ก **-L** ตัวอย่าง เมื่อต้องการรันการดำเนินการบน interim fix ที่มีเลเบล **ABC123**, ให้ป้อน:

```
emgr -cL ABC123
```

### การอ้างอิงโดย Ifix ID

แต่ละ interim fix ที่ถูกติดตั้งบนระบบที่กำหนด มี interim fix ID interim fix ID เป็นเพียง หมายเลขลำดับ ซึ่ง interim fix ถูกแสดงรายการในฐานข้อมูล interim fix การใช้ชื่อขั้นนี้อาจสะดวก ถ้าคุณดำเนินการบนโปรแกรมฟิกซ์เฉพาะกิจตามรายการ interim fix คำสั่ง **emgr** จะแปลง interim fix ID ลงในเลเบล interim fix ก่อนดำเนินการที่กำหนด เมื่อต้องการอ้างอิง interim fix ตาม ID ให้ส่ง ID เป็นพารามิเตอร์ กับแฟล็ก **-n**

**หมายเหตุ:** Ifix IDs สามารถเปลี่ยนเมื่อลบ และเพิ่มโปรแกรมฟิกซ์เฉพาะกิจ ตรวจสอบ หมายเลข interim fix ID ปัจจุบันเสมอโดยใช้แฟล็ก **-I** เพื่อแสดงรายการ interim fix เฉพาะ หรือ โปรแกรมฟิกซ์เฉพาะกิจทั้งหมด ตัวอย่าง เมื่อต้องการรันการดำเนินการตรวจสอบบน interim fix แรก ที่มี ID เท่ากับ 1 ให้ป้อน:

emgr -cn1

## อ้างอิงด้วย VUID

เนื่องจากแพ็คเกจ interim fix จะถูกติดตามอย่างเป็นรูปแบบ โดย entity, จึงเป็นไปได้ที่เลเบล interim fix เดียวกัน สามารถถูกใช้กับแพ็คเกจ interim fix มากกว่าหนึ่งแพ็คเกจ อย่างไรก็ตาม คำสั่ง **emgr** ไม่ยอมรับการติดตั้ง มากกว่าหนึ่ง interim fix ที่มีเลเบล interim fix เหมือนกันในเวลาเดียวกัน VUID (Virtually Unique ID) สามารถถูกใช้กับแพ็คเกจต่างกัน ที่มีเลเบล interim fix เหมือนกัน คำสั่ง **emgr** แปลง VUID ลงในเลเบล interim fix ก่อนดำเนินการปฏิบัติการที่กำหนด ตัวอย่าง เมื่อต้องการแสดง interim fix ที่ติดตั้งด้วย VUID เท่ากับ

**000775364C00020316020703**, ให้ป้อน:

```
emgr -l -u 000775364C00020316020703
```

**หมายเหตุ:** VUID ถูกแสดงในเฟสแสดงตัวอย่างของการติดตั้งและการลบ interim fix VUID ยังถูกแสดงเมื่อแสดงรายการด้วยระดับ verbosity ที่ตั้งค่าเป็น 3 ให้กับแฟล็ก **-v**

## การล๊อค Ifix

การดำเนินการต่อไปนี้จะถูกล๊อคในล๊อคไฟล์คำสั่ง **emgr /var/adm/ras/emgr.log**:

- การติดตั้ง
- การลบ
- การตรวจสอบ
- การเมาท์
- การยกเลิกการเมาท์
- บังคับการลบ

## การเปิดใช้การลบ Interim Fix โดยอัตโนมัติโดย installp

interim fix สามารถถูกแพ็คเกจโดยคำสั่ง **epkg** ให้มีไฟล์การอ้างอิง APAR ที่มีหมายเลขการอ้างอิง APAR หมายเลขการอ้างอิง APAR จะอนุญาตให้ **installp** แม็พ interim fix กลับไปที่ APARs สำหรับ Technology Levels ทั้งหมดที่การแก้ไขถูกจัดส่ง ถ้า **installp** กำหนดว่า interim fix อยู่ใน Technology Level, Service Pack, หรือ PTF ถูกนำมาใช้, **installp** จะลบ interim fix โดยอัตโนมัติก่อนนำอัพเดทมาใช้

ถ้า interim fix ถูกเปิดใช้สำหรับการลบอัตโนมัติ คำสั่ง **emgr** จะแสดงข้อความต่อไปนี้ ระหว่างการติดตั้ง interim fix:

ข้อความใส่ใจ: Interim fix ถูกเปิดใช้การลบอัตโนมัติโดย installp

## การอัปเดตพร้อมกัน

คำสั่ง **emgr** สนับสนุนการติดตั้ง interim fix ชนิดใหม่เรียกว่าการอัปเดตพร้อมกัน รูปแบบ interim fix นี้มีการแก้ไขกับ เคอร์เนล AIX หรือหนึ่งในส่วนขยายเคอร์เนล ที่สามารถนำมาใช้โดยตรง กับหน่วยความจำระบบและไม่จำเป็นต้องรีบูตระบบ การใช้โปรแกรมแก้ไขโดยตรงกับหน่วยความจำระบบ ทำให้คุณสามารถประเมินค่า และทดสอบการแก้ไขเคอร์เนลได้อย่างปลอดภัยโดยไม่ต้องแก้ไขไฟล์ที่มี เคอร์เนลปัจจุบันของระบบบนดิสก์ การอัปเดตพร้อมกันใดๆ ที่ใช้กับ หน่วยความจำระบบ จะไม่คงอยู่หลังจากการรีบูต นอกจาก คุณเลือกที่จะส่งการเปลี่ยนแปลงที่ทำการอัปเดตพร้อมกัน ไปที่ดิสก์โดยใช้แฟล็ก **-C** คุณสามารถใช้ การอัปเดตพร้อมกันได้โดยตรงกับแพตช์อื่น ที่เป็นโมดูลเดียวกัน คุณไม่จำเป็นต้องลบแพตช์ก่อนหน้านี้

อย่างไรก็ตาม ต้องมีเพียง โมดูลที่โหลดหนึ่งเวอร์ชันเท่านั้น นอกจากนี้ คุณไม่สามารถรัน การดำเนินการอัปเดตพร้อมกัน (ในหน่วยความจำ หรือบนดิสก์) สำหรับโปรแกรมฟิกซ์เฉพาะกิจในสภาวะ REBOOT\_REQUIRED จนกว่าจะรีบูตระบบ

คำสั่ง **emgr** สนับสนุนการใช้หน่วยความจำภายในพร้อมกับอัปเดต บนเซิร์ฟเวอร์ NIM แบบ thin (โคลเอ็นต์แบบไม่มีดิสก์ หรือไม่มีข้อมูล) เนื่องจากเซิร์ฟเวอร์แบบ thin แบ่งใช้ไฟล์ระบบปฏิบัติการกับโคลเอ็นต์อื่นๆ (ไดรเ็กทอรี /usr เป็นแบบอ่านอย่างเดียว) อ็อพชัน **emgr** ที่ต้องการ commit อัปเดตแบบพร้อมเพียงกันไปยังดิสก์ (แฟล็ก -C) จึงไม่ถูกต้องบนเซิร์ฟเวอร์ thin

**หมายเหตุ:** หากไฟล์ระบบปฏิบัติการของเซิร์ฟเวอร์ thin ต้องการแพตช์ไปยังดิสก์ โปรแกรมฟิกซ์ชั่วคราวอาจถูกใช้กับรีซอร์ส SPOT บน NIM หลักที่ให้บริการเซิร์ฟเวอร์แบบ thin โปรดอ้างอิงที่ การติดตั้ง Interim Fix ในส่วนรีซอร์ส SPOT ของ Installation Guide หรือ /usr/lpp/bos.sysmgmt/nim/README file (NIM IFIX/EMGR section) บน NIM master ของคุณ สำหรับรายละเอียดเกี่ยวกับการติดตั้ง interim fix ลงใน SPOT

ฐานข้อมูล **emgr** จะตั้งอยู่ในไดรเ็กทอรี /var/emgrdata บนเซิร์ฟเวอร์แบบ thin เนื่องจากระบบไฟล์ /usr เป็นแบบอ่านอย่างเดียวบนเซิร์ฟเวอร์ thin

การดำเนินการ **emgr** อาจไม่ได้รับการสนับสนุน ในสภาวะแวดล้อมของเซิร์ฟเวอร์แบบ thin เช่น bosboot การขยายระบบไฟล์ ตามผลลัพธ์ที่ได้ แฟล็ก **emgr** ต่อไปนี้ไม่ได้รับการสนับสนุน ในสภาวะแวดล้อมของเซิร์ฟเวอร์แบบ thin: -C, -e, -I, -k และ -X และ แฟล็ก **-b** ซึ่งข้ามกระบวนการ bosboot สำหรับโปรแกรมฟิกซ์ชั่วคราวที่ต้องการรีบูต จะถูกใช้เสมอเมื่อสามารถเรียกใช้งานได้เนื่องจากการดำเนินการ bosboot อาจไม่ได้รับการสนับสนุนสำหรับเซิร์ฟเวอร์แบบ thin

## แฟล็ก

ไอเท็ม  
-a path

คำอธิบาย

ระบุไดรเ็กทอรีพาททางเลือกสำหรับการติดตั้ง

**หมายเหตุ:** แฟล็ก **-a** ใช้ระหว่างการลบ interim fix ออกต่อเมื่อ แฟล็ก **-e** และ **-a** ของคำสั่ง **emgr** ถูกใช้ระหว่างการติดตั้งของ interim fix ถ้า interim fix ไม่ได้ติดตั้ง โดยใช้แฟล็ก **-e** และ **-a** คำสั่ง **emgr** จะไม่ลบโปรแกรมฟิกซ์เฉพาะกิจออกจากไดรเ็กทอรีพาททางเลือก อย่างสมบูรณ์

สำหรับวิธีแก้ปัญหาให้ใช้คำสั่งต่อไปนี้เพื่อลบ interim fix ที่ถูกติดตั้ง ในไดรเ็กทอรีพาทเลือกออก:

```
chroot /alt_inst /usr/sbin/emgr -r -L efix_label
```

-b  
-c  
-C

ทำให้คำสั่ง **emgr** ข้ามกระบวนการ AIX bosboot ปกติสำหรับโปรแกรมฟิกซ์เฉพาะกิจที่ต้องรีบูต

ระบุการดำเนินการตรวจสอบ สิ่งให้คำสั่ง **emgr** รัน การดำเนินการตรวจสอบบน interim fix หรือโปรแกรมฟิกซ์เฉพาะกิจที่ระบุ ส่งข้อมูล interim fix ที่มีการอัปเดตพร้อมกัน ไปที่ดิสก์ อ็อพชันนี้ต้องถูกใช้พร้อมกับอ็อพชัน **-i** หรือสามารถถูกใช้หลังจาก interim fix ได้นำมาใช้กับอ็อพชัน **-i** นี้ทำให้การอัปเดตพร้อมกัน คงอยู่หลังการรีบูตระบบ

-d

หลังจากการอัปเดตพร้อมกันถูกส่ง ข้อมูล การลบจะมีผลให้โมดูลถูกเรียกคืนกลับสู่ สภาวะ un-patched เดิม, ไม่ว่าแพตช์อื่น สำหรับโมดูลมีอยู่หรือไม่ แพตช์ก่อนหน้านี้ทั้งหมดสำหรับโมดูล ถูกลบเมื่อทำการส่งข้อมูล

-e interimfixPackage

แสดงเนื้อหาของการจัดเรียง อ็อพชันนี้ มีประโยชน์กับแฟล็ก **-v** ในการแสดงเอาต์พุต verbosity

ระบุพารามิเตอร์ของไฟล์แพ็กเกจ interim fix และติดตั้งแพ็กเกจ interim fix ไฟล์แพ็กเกจ interim fix ต้องถูกสร้าง ด้วยคำสั่ง **epkg** และ ต้องจบด้วย ส่วนขยายการบีบอัดข้อมูล 16-บิต, **.Z**

-f ListFile

ระบุไฟล์ที่มีข้อมูลหนึ่งในข้อมูลต่อไปนี้:

- รายการของตำแหน่งแพ็กเกจสำหรับการดำเนินการติดตั้ง (หนึ่ง รายการต่อบรรทัด)
- รายการของเลเบล interim fix สำหรับการดำเนินการ ลบ เมาทัยกเลิกการเมาท์ และการตรวจสอบ (หนึ่งรายการต่อบรรทัด)

คำสั่ง **emgr** ละเว้นบรรทัดว่าง หรือบรรทัด ที่อ็อกซ์ระ non-white-space แรกคืออ็อกซ์ระ #

## ไอเท็ม

-i *interimfixPackage*

### คำอธิบาย

ระบุพาทของไฟล์แพ็คเกจ interim fix ที่มี การอัปเดตพร้อมกัน และใช้การอัปเดตพร้อมกันกับหน่วยความจำ ระบบ การอัปเดตจะไม่คงอยู่เมื่อระบบรีบูต นอกจากมีการใช้แฟล็ก -C

คุณยังสามารถใช้แฟล็ก -i เพื่อใช้หนึ่งการอัปเดตพร้อมกัน เพื่ออัปเดต ส่วนอื่น สำหรับโมดูลเดียวกัน การอัปเดตแบบคอนเคอร์เร้นท์นั้นถูกเรียกว่า "follow-on"

-I

รันดีบักเกอร์ระดับต่ำสำหรับ AIX bosboot โดยใช้คำสั่ง bosboot ที่ระบุแฟล็ก -I

-k

โหลดดีบักเกอร์ระดับต่ำระหว่าง AIX bosboot โดยใช้คำสั่ง bosboot ที่ระบุแฟล็ก -D

-l

สั่งให้คำสั่ง emgr รันการดำเนินการแสดงรายการบน interim fix หรือโปรแกรมฟิสิกซ์เฉพาะกิจที่ระบุ

-L *Label*

เลือก interim fix สำหรับการดำเนินการนี้ ตามเลเบล interim fix

-m

สั่งให้คำสั่ง emgr ให้ดำเนินการติดตั้งเมท เมื่อ interim fix ถูก mount-installed, ไฟล์ interim fix จะถูกเมทกับไฟล์เป้าหมาย

-M

สั่งให้คำสั่ง emgr ติดตั้ง interim fix หรือโปรแกรมฟิสิกซ์เฉพาะกิจที่ติดตั้งแล้ว โดยใช้แฟล็ก -m แฟล็ก -M สามารถถูกใช้เพื่อเมท

interim fix ที่ถูกติดตั้งโดยใช้แฟล็ก -m และ ได้ถูกยกเลิกการเมทโดยแฟล็ก -U หรือโดยวิธีอื่น เช่นการรีบูตระบบ

-n *interimfixID*

เลือก interim fix สำหรับการดำเนินการนี้ โดยระบุ interim fix ID

-o

ระบุว่าการติดตั้ง interim fix สามารถเขียนทับแพ็คเกจที่มี

-p

สั่งให้คำสั่ง emgr ทำการแสดงตัวอย่างสำหรับ การติดตั้งหรือการลบออก การแสดงตัวอย่าง รันการดำเนินการตรวจสอบทั้งหมด แต่ไม่ทำการเปลี่ยนแปลงใด

-P [*Package*]

ระบุการดำเนินการ package-view ซึ่ง แสดงแพ็คเกจทั้งหมดที่ถูกล็อกโดยตัวจัดการ interim fix, โปรแกรมติดตั้ง และเลเบลการล็อก

-q

ยับยั้งเอาต์พุตทั้งหมดที่ไม่ใช่ข้อผิดพลาดและ คำเตือนร้ายแรง

-r

สั่งให้คำสั่ง emgr รันการดำเนินการลบบน interim fix หรือโปรแกรมฟิสิกซ์เฉพาะกิจที่ระบุ

การลบ แอ็คทีฟแพ็คเกจ นำแพ็คเกจก่อนหน้าสำหรับโมดูลกลับมาใช้, มีการจัดเตรียม แพ็คเกจที่มีอยู่ ถ้าไม่มีแพ็คเกจก่อนหน้า โมดูลจะถูกคืนค่าเป็น สถานะ un-patched เดิม

-R *Label*

สั่งให้คำสั่ง emgr รันการดำเนินการ force-remove อ็อพชันนี้ลบ interim fix การล็อกข้อมูลและการล็อกแพ็คเกจที่เชื่อมโยงกับเลเบล interim fix โดยไม่มีการลบ interim fix ไฟล์ออกจริง, รันสคริปต์การลบ หรือกระบวนการบูต อ็อพชันนี้สามารถถูกใช้สำหรับหนึ่ง interim fix ต่อหนึ่งครั้งเท่านั้น เลเบล interim fix จำเป็นในการระบุ interim fix เป้าหมาย

### ข้อควรสนใจ:

- เมธอดนี้ของการลบ interim fix ควรถูกพิจารณาเป็นกระบวนการฉุกเฉิน เนื่องจากเมธอดนี้ สามารถสร้างความไม่สอดคล้องกันบนระบบเป้าหมาย, เมธอดการบังคับการลบ ควรถูกนำมาใช้เฉพาะถ้าเมธอดการลบ interim fix อื่นทั้งหมดทำงานไม่สำเร็จ
- คุณต้องใช้กระบวนการลบมาตรฐาน (แฟล็ก -r) เพื่อลบโปรแกรมฟิสิกซ์เฉพาะกิจที่ติดตั้งไว้ในโพธิ์เตอร์ฉุกเฉิน คุณสามารถใช้แฟล็ก -R เพื่อบังคับให้ลบเลเบล แฟล็ก -R ต้องการแฟล็ก -F เพื่อลบเลเบล เมื่อคุณระบุแฟล็ก -F พร้อมกับแฟล็ก -R ตัวอย่างเช่น emgr -R *ifix\_label* อ็อพชันบังคับให้ลบไม่ได้ลบไฟล์โปรแกรมฟิสิกซ์เฉพาะกิจ ข้อมูลที่บันทึกไว้ หรือดำเนินการสคริปต์ การลบใดๆ อ็อพชันนี้ต้องใช้เฉพาะถ้าไม่สามารถทำการลบมาตรฐานให้สำเร็จ

-u *VUID*

เลือก interim fix สำหรับการดำเนินการนี้ โดยการระบุ VUID

-U

สั่งให้คำสั่ง emgr ถอนการติดตั้ง interim fix หรือโปรแกรมฟิสิกซ์เฉพาะกิจที่ติดตั้งแล้ว โดยใช้แฟล็ก -m

-v {1|2|3}

ระบุระดับ verbosity สำหรับการแสดงรายการ หรือระดับการตรวจสอบสำหรับการดำเนินการตรวจสอบ ระดับ ที่ใช้ได้คือ 1, 2 และ 3

-w *Directory*

สั่งให้คำสั่ง emgr เพื่อใช้ไดเรกทอรีทำงานที่ระบุแทนค่าดีโฟลต์ไดเรกทอรี /tmp

-X

พยายามขยายระบบไฟล์ ซึ่งมีพื้นที่ ไม่เพียงพอในการดำเนินการ emgr ที่ร้องขอ อ็อพชันอ็อพชันขยายระบบไฟล์จากการประมาณพื้นที่และขนาด ที่มีที่ถูกต้องเตรียมโดยแพ็คเกจ interim fix และ คำสั่ง emgr

### หมายเหตุ:

1. เป็นไปได้ที่จะใช้พื้นที่เกินพื้นที่ดิสก์ที่มีอยู่ระหว่างการติดตั้ง แม้ว่าจะมีการใช้แฟล็ก -X ซึ่งเป็นไปได้ว่าไฟล์อื่น ถูกสร้างหรือขยายในระบบไฟล์เดียวกัน ระหว่างการติดตั้ง
2. ระบบไฟล์รีโมตไม่สามารถถูกขยายโดยคำสั่ง emgr

## สถานะออก

0 การดำเนินการคำสั่ง emgr ทั้งหมด เสร็จสมบูรณ์

>0 มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

ผู้ดูแลระบบหรือผู้ใช้ที่มีสิทธิ์ `aix.system.install` สามารถรันคำสั่ง `emgr` บนระบบ multi-level secure (MLS) ข้อมูล Ifix, ไฟล์ที่บันทึก และไฟล์ชั่วคราวสามารถเข้าถึงได้โดยผู้ใช้ `root` เท่านั้น

คำสั่ง `emgr` ค้นหาคำสั่งการสร้าง MD5 ที่สนับสนุนบนระบบ ถ้าพบ คำสั่ง `emgr` จะ แสดงเช็คซัม MD5 แก่ผู้ใช้นั้นผู้ใช้สามารถตรวจสอบ MD5 sum นี้กับซอร์สที่ปลอดภัย ถ้าไม่พบคำสั่งการสร้าง MD5 คำสั่ง `emgr` จะไม่มีการดำเนินการต่อ

ผู้ใช้สามารถบังคับเช็คพาสเป็นคำสั่ง MD5 โดยการเอ็กซ์พอร์ต ตัวแปรเซลล์ `EMGR_MD5_CMD` ตัวแปรนี้ควรมีพาสลับบุรณ์ ต่อคำสั่งการสร้าง MD5

**หมายเหตุ:** คุณลักษณะนี้ไม่ได้รับการสนับสนุนในวิธีสแตนด์แบบของการจัดการกับ interim fix ขอแนะนำให้ผู้ใช้อัพเดทเป็นระดับล่าสุดของการจัดการ interim fix โดยอัพเดท `bos.rte.install` เป็นระดับล่าสุด

## ตัวอย่าง

- เมื่อต้องการแสดงตัวอย่างการติดตั้งของแพ็คเกจ interim fix ชื่อ `games.020303.epkg.Z`, ให้ป้อน:  
`emgr -p -e games.020303.epkg.Z`
- เมื่อต้องการติดตั้งแพ็คเกจ interim fix ชื่อ `games.020303.epkg.Z` และ ขยายระบบไฟล์โดยอัตโนมัติ ถ้าต้องการพื้นที่เพิ่มเติม ให้ป้อน:  
`emgr -X -e games.020303.epkg.Z`
- เมื่อต้องการแสดงรายการโปรแกรมฟิซเฉพาะกิจทั้งหมดบนระบบ ให้ป้อน:  
`emgr -l`
- เมื่อต้องการทำการแสดงรายการระดับ 3 ของเลเบล interim fix `games` ให้ป้อน:  
`emgr -lv3 -L games`
- เมื่อต้องการลบ interim fix ที่มีเลเบล `games`, ให้ป้อน:  
`emgr -r -L games`
- เมื่อต้องการแสดงตัวอย่างการลบเลเบล interim fix ในไฟล์ `/tmp/myfixes` ให้ป้อน:  
`emgr -rp -f /tmp/myfixes`
- เมื่อต้องการตรวจสอบโปรแกรมฟิซเฉพาะกิจทั้งหมดที่มีระดับการตรวจสอบ 2 ให้ป้อน:  
`emgr -cv2`
- เมื่อต้องการตรวจสอบ interim fix ID number 3 ที่มีระดับการตรวจสอบ 1 (ระดับการตรวจสอบดีฟอลต์) ให้ป้อน:  
`emgr -c -n3`
- เมื่อต้องการตรวจสอบ interim fix ที่มี VUID `000775364C00020316020703` และ ระดับการตรวจสอบ 3 ให้ป้อน:  
`emgr -u 000775364C00020316020703 -c -v3`
- เมื่อต้องการแสดงแพ็คเกจที่ล็อกทั้งหมดและเลเบล interim fix ให้ป้อน:  
`emgr -P`
- เมื่อต้องการแสดงเลเบล interim fix ทั้งหมดที่ล็อกแพ็คเกจ `installp bos.rte.lvm` ให้ป้อน:  
`emgr -P bos.rte.lvm`
- เมื่อต้องการ mount-install แพ็คเกจ interim fix ชื่อ `games.020303.epkg.Z` และยับยั้ง AIX `bosboot`, ให้ป้อน

- emgr -e games.020303.epkg.Z -mb
13. เมื่อต้องการเมาท์ไฟล์ interim fix ทั้งหมดที่ถูก mount-installed บนระบบโดยใช้อ็อปชัน **-m** ให้ป้อน:  
emgr -M
  14. เมื่อต้องการยกเลิกการเมาท์ไฟล์ interim fix ทั้งหมดที่สัมพันธ์กับเลเบล interim fix **games** ให้ป้อน:  
emgr -U -L games
  15. เมื่อต้องการใช้แพ็คเกจ interim fix ชื่อ **kernelmod.031007.epkg.Z** พร้อมกับการอัปเดตไปที่หน่วยความจำระบบ ให้ป้อน:  
emgr -i kernelmod.031007.epkg.Z
  16. เมื่อต้องการคอมมิตการอัปเดตแบบคอนเคอร์เรนต์ที่สัมพันธ์กับเลเบล interim fix **kernelmod** กับดิสก์ ให้ป้อน:  
emgr -C -L kernelmod
  17. เมื่อต้องการใช้แพ็คเกจ interim fix ชื่อ **kernelmod2.031007.epkg.Z** พร้อมกับการอัปเดตไปที่หน่วยความจำระบบ และส่งการอัปเดตพร้อมกันไปที่ดิสก์ ให้ป้อน:  
emgr -i kernelmod2.031007.epkg.Z -C
  18. เมื่อต้องการแสดงเอาต์พุตรายละเอียดระดับ 3 บนแพ็คเกจ interim fix **test.102403.epkg.Z** ให้ป้อน:  
emgr -v3 -d test.102403.epkg.Z

## ไฟล์

### ไอเท็ม

/usr/sbin/emgr  
 /usr/emgrdata/DBS/ifix.db  
 /usr/emgrdata/DBS/files.db  
 /usr/emgrdata/DBS/pkglck.db  
 /usr/emgrdata/DBS/prereq.db  
 /usr/emgrdata/DBS/e2prereq.db  
 /usr/emgrdata/DBS/aparref.db

### คำอธิบาย

มีคำสั่ง **emgr**  
 มีฐานข้อมูลส่วนหัว interim fix  
 มีฐานข้อมูลไฟล์ interim fix  
 มีฐานข้อมูลล็อกแพ็คเกจ  
 มีฐานข้อมูลสิ่งที่จำเป็นต้องมี  
 มีฐานข้อมูลสิ่งที่จำเป็นต้องมี interim fix  
 มีฐานข้อมูลไฟล์อ้างอิง APAR

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง epkg” ในหน้า 410

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **bosboot**

การติดตั้งผลิตภัณฑ์ซอฟต์แวร์ทางเลือกหรืออัปเดตเซอวิส

---

## คำสั่ง **emstat**

### วัตถุประสงค์

แสดงสถิติ exception อีมีเลชัน

### ไวยากรณ์

**emstat** [ **-a** | **-v** ] [ *Interval* ] [ *Count* ]

## คำอธิบาย

คำสั่ง `emstat` แสดงสถิติ exception อีมีเลชัน ข้อยกเว้นอีมีเลชันสามารถเกิดขึ้นเมื่อบางแอปพลิเคชันหรือไลบรารีที่มีอยู่ ซึ่งมีคำสั่งที่ถูกลบออกจากสถาปัตยกรรมตัวประมวลผลเก่า ถูกเรียกใช้บนตัวประมวลผลใหม่ คำสั่งเครื่องเหล่านี้อาจทำให้เกิดข้อยกเว้น โปรแกรมคำสั่งเครื่องผิดกฎหมาย ระบบปฏิบัติการจับข้อยกเว้น เหล่านี้ และเลียนแบบคำสั่งเก่ากว่าเพื่อคงฟังก์ชันการทำงาน ของโปรแกรม ที่เป็นไปได้คือค่าใช้จ่ายประสิทธิภาพการทำงานของโปรแกรม

exception อีมีเลชันนับตั้งแต่ครั้งสุดท้ายที่เครื่องถูกรีบูต และจำนวนนับในช่วงเวลาปัจจุบันถูกแสดง ผู้ใช้มีทางเลือก ที่แสดงแสดงสถิติ exception การจัดตำแหน่งหรือสถิติอีมีเลชันโพรเซสเซอร์แต่ละรายการ

ดีฟอลต์เอาต์พุตแสดงสถิติทุกวินาที ช่วงเวลาการสุ่มตัวอย่าง และจำนวน การทำซ้ำสามารถระบุได้

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
Interval	ช่วงเวลาระหว่างการสุ่มตัวอย่าง
Count	จำนวนการวนซ้ำ

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a	แสดงสถิติ exception การจัดตำแหน่ง แฟล็กนี้ใช้ไม่ได้กับ แฟล็ก -v
-v	แสดงแต่ละสถิติโพรเซสเซอร์ แฟล็กนี้ไม่สามารถใช้กับแฟล็ก -a

## ตัวอย่าง

- เมื่อต้องการแสดงสถิติอีมีเลชันทุกวินาที พิมพ์:

```
emstat
```

ซึ่ง จะสร้างเอาต์พุตต่อไปนี้:

Emulation	Emulation
SinceBoot	Delta
8845591	0
8845591	0
8845591	0
8845591	0
8845591	0
8845591	0
8845591	0

...

- เมื่อต้องการแสดงสถิติ exception อีมีเลชันและการจัดตำแหน่ง ทุกสองวินาที รวม 5 ครั้งให้พิมพ์:

```
emstat -a 2 5
```

ซึ่งจะได้ผลลัพธ์ตามที่แสดงดังต่อไปนี้:

Alignment	Alignment	Emulation	Emulation
SinceBoot	Delta	SinceBoot	Delta
21260604	0	70091846	0
23423104	2162500	72193861	2102015
25609796	2186692	74292759	2098898
27772897	2163101	76392234	2099475
29958509	2185612	78490284	2098050

3. เมื่อต้องการแสดงสถิติอีมีลชัน ทุก 5 วินาทีสำหรับแต่ละโพรเซสเซอร์ ให้พิมพ์:

```
emstat -v 5
```

ซึ่งจะสร้างเอาต์พุตต่อไปนี้:

Emulation	Emulation	Emulation	Emulation
SinceBoot	Delta	Delta00	Delta01
88406295	0	0	0
93697825	5291530	0	5291530
98930330	5232505	5232505	0
102595591	3665261	232697	3432564
102595591	0	0	0

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง alstat

---

## คำสั่ง emsvcsctrl

### วัตถุประสงค์

สตาร์ทระบบย่อยการจัดการเหตุการณ์

### ไวยากรณ์

```
emsvcsctrl [-a | -s | -k | -d | -c | -t | -o | -h]
```

### คำอธิบาย

emsvcsctrl เป็นสคริปต์ควบคุมที่เริ่มระบบย่อยการจัดการเหตุการณ์ การจัดการเหตุการณ์เป็นระบบย่อยแบบกระจายของ RSCT ที่มีชุดของเซอร์วิสที่พร้อมใช้สูงสำหรับเซิร์ฟเวอร์ IBM RS/6000® โดยการจับคู่ข้อมูลเกี่ยวกับสถานะของรีซอร์สระบบที่มีข้อมูลเกี่ยวกับสภาวะรีซอร์สที่มีประโยชน์ต่อโคลเอ็นต์โปรแกรม, ซึ่งสร้างเหตุการณ์ โคลเอ็นต์โปรแกรมสามารถใช้เหตุการณ์เพื่อตรวจจับและกู้คืนจากระบบล้มเหลว, ซึ่งช่วยเพิ่มสภาพพร้อมใช้งานของระบบ สคริปต์ควบคุม emsvcsctrl ควบคุมการดำเนินการของระบบย่อย Event Management ระบบย่อยอยู่ ภายใต้การควบคุมของ System Resource Controller (SRC) และเป็นของ กลุ่มระบบย่อยที่เรียกว่า emsvcs daemon ถูกเชื่อมโยงกันด้วยแต่ละระบบย่อย สคริปต์ emsvcsctrl ยังควบคุมการดำเนินการของระบบย่อย AIX Resource Monitor ระบบย่อยอยู่ภายใต้ตัวควบคุม SRC และยังเป็น ของกลุ่มระบบย่อย emsvcs daemon ถูกเชื่อมโยงกันด้วยแต่ละระบบย่อย

อินสแตนซ์ของ Event Management และระบบย่อย AIX Resource Monitor ทำงานบนแต่ละโหนดในคลัสเตอร์ HACMP/ES จากมุมมองของการปฏิบัติการ กลุ่มระบบย่อย Event Management ถูกจัดระบบดังต่อไปนี้:

#### ระบบย่อย

การจัดการเหตุการณ์

#### กลุ่มระบบย่อย

emsvcs

#### ระบบย่อย SRC

ระบบย่อย emsvcs ถูกเชื่อมโยงกับ haemd daemon

## emaixos

emaixos ถูกเชื่อมโยงกับ harmad daemon

## Daemons

haemd daemon จัดเตรียมเซอวิวิส Event Management harmad daemon เป็นรีซอร์สมอนิเตอร์สำหรับรีซอร์สระบบปฏิบัติการ AIX

สคริปต์ emsvcsctrl โดยปกติจะไม่ถูกเรียกใช้งานจากบรรทัดคำสั่ง โดยปกติ จะถูกเรียกโดยคำสั่งสคริปต์เริ่มทำงาน HACMP/ES ระหว่างการติดตั้ง ของระบบ

สคริปต์ emsvcsctrl มีการควบคุมที่หลากหลายสำหรับการดำเนินการระบบย่อย Event Management:

- การเพิ่ม สตาร์ท หยุด และลบระบบย่อย
- การล้างระบบย่อย
- การเปิดและปิดทำงานการติดตาม

การเพิ่มระบบย่อย: เมื่อมีการระบุแฟล็ก -a เหตุการณ์ควบคุม จะใช้คำสั่ง mkssys เพื่อเพิ่มระบบย่อย Event Management และ AIX Resource Monitor ให้กับ SRC สคริปต์ควบคุมจะทำงาน ดังนี้:

1. ตรวจสอบว่าระบบย่อย emsvcs และ emaixos ถูกหยุดการทำงาน
2. ลบระบบย่อย emsvcs และ emaixos จาก SRC (เฉพาะในกรณี ที่ระบบย่อยดังกล่าวยังมีอยู่)
3. เพิ่มระบบย่อย emsvcs ให้กับ SRC
4. เพิ่มระบบย่อย emaixos ให้กับ SRC
5. เพิ่มกลุ่ม haemrm โดยใช้คำสั่ง mkgroup ถ้ายังไม่มีอยู่ ข้อผิดพลาดที่เกิดขึ้นจะถูกเขียนลงในล็อกไฟล์ชื่อ /var/ha/log/em.mkgroup
6. ซึ่งจะสร้างไดเรกทอรี /var/ha/lck/haem and /var/ha/soc/haem ถ้า ยังไม่มีอยู่ ข้อผิดพลาดที่เกิดขึ้นจะถูกเขียนลงในล็อกไฟล์ชื่อ /var/ha/log/em.mkdir
7. คัดลอก Event Management Configuration Database, (EMCDB) จาก ตำแหน่งที่ติดตั้ง, /usr/sbin/rsct/install/config/em.HACMP.cdb ไปที่ตำแหน่ง รันไทม์, /etc/ha/cfg/em.HACMP.cdb ข้อผิดพลาดที่เกิดขึ้นจากการคัดลอกจะถูกเขียนลงในล็อกไฟล์ชื่อ /var/ha/log/em.cp

สตาร์ทระบบย่อย: เมื่อมีการระบุแฟล็ก -s, สคริปต์ควบคุม ใช้คำสั่ง startsrc เพื่อสตาร์ทระบบย่อย Event Management, emsvcs, และระบบย่อย AIX Resource Monitor, emaixos

หยุดระบบย่อย: เมื่อมีการระบุแฟล็ก -k, สคริปต์ควบคุม ใช้คำสั่ง stopsrc เพื่อหยุดระบบย่อย Event Management, emsvcs, และระบบย่อย AIX Resource Monitor, emaixos

ลบระบบย่อย: เมื่อมีการระบุแฟล็ก -d เหตุการณ์ควบคุม จะใช้คำสั่ง rmssys เพื่อลบระบบย่อย Event Management และ AIX Resource Monitor จาก SRC สคริปต์ควบคุมจะทำงาน ดังนี้:

1. ตรวจสอบว่าระบบย่อย emsvcs และ emaixos ถูกหยุดการทำงาน
2. ลบระบบย่อย emsvcs และ emaixos จาก SRC โดยใช้คำสั่ง rmssys

ล้างระบบย่อย: เมื่อมีการระบุ -c, สคริปต์ ควบคุมหยุดและลบระบบย่อย Event Management สำหรับพาร์ติชัน ระบบทั้งหมดออกจาก SRC สคริปต์ควบคุมจะทำงาน ดังนี้:

1. ทำการหยุดอินสแตนซ์ของระบบย่อยทั้งหมดในกลุ่มระบบย่อยโดยใช้ คำสั่ง `stopsrc -g emsvcs`
2. ลบอินสแตนซ์ของระบบย่อยทั้งหมดในกลุ่มระบบย่อยออกจาก SRC โดยใช้คำสั่ง `rmssys`
3. ลบ Event Management Configuration Database (EMCDB) ออกจาก ตำแหน่งรันไทม์, `/etc/ha/cfg/em.HACMP.cdb`

**เปิดการติดตาม:** เมื่อมีการระบุแฟล็ก `-t` สคริปต์ควบคุม เปิดการติดตามสำหรับ `haemd daemon`, โดยใช้คำสั่ง `haemtrcon` การติดตาม `harmad daemon` ถูกเปิดใช้งานเช่นกัน โดยใช้คำสั่ง `traceson`

**ปิดการติดตาม:** เมื่อมีการระบุแฟล็ก `-o` สคริปต์ควบคุม ปิดการติดตามสำหรับ `haemd daemon`, โดยใช้คำสั่ง `haemtrcoff` การติดตาม `harmad daemon` ถูกปิดใช้งานเช่นกัน โดยใช้คำสั่ง `tracesoff`

**บันทึก:** ขณะที่มีการทำงาน, โดยปกติ Event Management daemon จัดเตรียมข้อมูลเกี่ยวกับการดำเนินการและข้อผิดพลาดโดยการเขียนรายการไปที่บันทึกข้อผิดพลาด AIX ถ้าไม่สามารถทำได้ ข้อผิดพลาดจะถูกเขียนไปที่ล็อกไฟล์ ชื่อ `/var/ha/log/em.default.cluster_name`

## แฟล็ก

- a เพิ่มระบบย่อย
- s สตาร์ทระบบย่อย
- k หยุดทำงานระบบย่อย
- d ลบระบบย่อย
- c ล้างระบบย่อย
- t เปิดการติดตามสำหรับระบบย่อย
- o ปิดการติดตามสำหรับระบบย่อย
- h แสดงข้อมูลการใช้งาน

## ความปลอดภัย

คุณต้องรันด้วย ID ผู้ใช้ `root`

## สถานะออก

- 0 บ่งชี้ความสำเร็จของคำสั่ง
- 1 หมายถึงมีข้อผิดพลาดเกิดขึ้น

## ข้อจำกัด

คำสั่งนี้ใช้ได้เฉพาะในสภาวะแวดล้อม HACMP™ เท่านั้น

## ข้อผิดพลาดมาตรฐาน

คำสั่งนี้เขียนข้อความแสดงความผิดพลาด (ตามความจำเป็น) ไปที่ข้อผิดพลาดมาตรฐาน

## ตัวอย่าง

1. เมื่อต้องการเพิ่มระบบย่อย Event Management ให้กับ SRC ให้ป้อน:

```
emsvcsctrl -a
```

2. เมื่อต้องการสแตร์ระบบย่อย Event Management ให้ป้อน:

```
emsvcsctrl -s
```

3. เมื่อต้องการหยุดระบบย่อย Event Management ให้ป้อน:

```
emsvcsctrl -k
```

4. เมื่อต้องการลบระบบย่อย Event Management จาก SRC ให้ป้อน:

```
emsvcsctrl -d
```

5. เมื่อต้องการล้างระบบย่อย Event Management ให้ป้อน:

```
emsvcsctrl -c
```

6. เมื่อต้องการเปิดการติดตามสำหรับ Event Management daemon ให้ป้อน:

```
emsvcsctrl -t
```

7. เมื่อต้องการปิดการติดตามสำหรับ Event Management daemon ให้ป้อน:

```
emsvcsctrl -o
```

## Location

`/usr/sbin/rsct/bin/emsvcsctrl`

มีสคริปต์ `emsvcsctrl`

## ไฟล์

`/var/ha/log/em.default.cluster_name`

มีไฟล์บันทึกดีฟอลต์ของ haemd daemon บนคลัสเตอร์ที่ชื่อ `cluster_name`

`/var/ha/log/em.cp`

มีไฟล์บันทึกข้อผิดพลาดที่เกิดขึ้นขณะทำการคัดลอก Event Management Configuration Database

`/var/ha/log/em.trace.cluster_name`

มีไฟล์บันทึกการติดตามของ haemd daemon บนคลัสเตอร์ที่ชื่อ `cluster_name`

`/var/ha/log/em.mkgroup`

มีไฟล์บันทึกข้อผิดพลาดที่เกิดขึ้นขณะทำการกลุ่ม haemrm

`/var/ha/log/em.mkdir`

มีไฟล์บันทึกข้อผิดพลาดที่เกิดขึ้นระหว่างการสร้างไดเร็กทอรี `/var/ha/lck/haem` และ `/var/ha/soc/haem`

---

## คำสั่ง enable

คำสั่ง `enable` ประกอบด้วยข้อมูลสำหรับ `AIX Print Subsystem enable` และ `System V Print Subsystem enable`

## AIX คำสั่ง Print Subsystem enable

## วัตถุประสงค์

เปิดใช้อุปกรณ์พริ้นเตอร์คิว

## ไวยากรณ์

`enable PrinterName ...`

## คำอธิบาย

คำสั่ง `enable` นำอุปกรณ์พริ้นเตอร์คิว ที่ระบุโดยพารามิเตอร์ `PrinterName` ออนไลน์, หรือเปิดใช้อุปกรณ์พริ้นเตอร์คิวที่จะถูกใช้กับระบบ

### Notes:

1. คุณต้องมีสิทธิ์ผู้ใช้ `root` หรือเป็นสมาชิกกลุ่ม `printq` เพื่อ รันคำสั่งนี้
2. ถ้าคุณป้อน `enable -?`, ระบบจะแสดง ข้อความแสดงความผิดพลาดต่อไปนี้:  
enq: (FATAL ERROR): 0781-048: Bad queue or device name: -?

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ตัวอย่าง

เมื่อต้องการเปิดใช้อุปกรณ์คิวงานพิมพ์ `lp0:lpd0`, ให้ป้อน:

```
enable lp0:lpd0
```

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/etc/qconfig</code>	มีไฟล์คอนฟิกูเรชันคิว
<code>/etc/qconfig.bin</code>	มีเวอร์ชันไบนารีแบบย่อของไฟล์ <code>/etc/qconfig</code>
<code>/usr/sbin/qdaemon</code>	มีการคิว daemon
<code>/var/spool/lpd/qdir/*</code>	มีการรองขอคิว
<code>/var/spool/lpd/stat/*</code>	มีข้อมูลเกี่ยวกับสถานะของอุปกรณ์
<code>/var/spool/qdaemon/*</code>	มีสำเนาชั่วคราวของไฟล์ที่จัดเข้าคิว

## คำสั่ง System V Print Subsystem enable

## วัตถุประสงค์

เปิดใช้งานพริ้นเตอร์ LP

## ไวยากรณ์

`enable printers`

### คำอธิบาย

คำสั่ง `enable` เรียกทำงาน *พริ้นเตอร์* ที่ได้กำหนดชื่อไว้, ทำให้สามารถพิมพ์การร้องขอที่ส่งโดยคำสั่ง `lp` ถ้าพริ้นเตอร์เป็นรีโมต, คำสั่ง จะเปิดใช้เพียงการถ่ายโอนของการร้องขอไปที่ระบบรีโมต; คำสั่ง `enable` ต้องถูกรันอีกครั้ง บนระบบรีโมตเพื่อเรียกทำงาน พริ้นเตอร์ (รัน `lpstat -p` เพื่อรับสถานะของพริ้นเตอร์)

เมื่อการเปลี่ยนแปลงถูกดำเนินการกับแอตทริบิวต์ของอุปกรณ์การพิมพ์ จะมีการจำไว้โดย `enable` ดังนั้นเมื่อต้องการเปลี่ยน นิยามหรือการจัดสรร สำหรับอุปกรณ์ คุณต้องปิดใช้งานพริ้นเตอร์บนอุปกรณ์นั้น เปลี่ยนอุปกรณ์ แล้วรัน `enable` แอตทริบิวต์ อุปกรณ์ใหม่จะมีผล เมื่อ `enable` ถูกเรียกใช้งาน

ชื่อพริ้นเตอร์เป็น *คำที่กำหนดโดยระบบ* ดังนั้น ควรถูกจำกัดเป็นอักขระ ASCII ตัวพิมพ์ใหญ่และตัวพิมพ์เล็ก

ถ้าคุณป้อน `enable -?`, ระบบจะแสดงข้อความการใช้ คำสั่งและส่งกลับ 0

### ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

### ไฟล์

`/var/spool/lp/*`

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `disable`” ในหน้า 167

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `cancel`

คำสั่ง `lp`

การเริ่มทำงานและหยุดทำงานคิวงานพิมพ์

---

## คำสั่ง `enotifyevent` Command, `notifyevent`

### วัตถุประสงค์

ข้อมูลเหตุการณ์เมลที่สร้างโดย event response resource manager (ERRM) กับล็อก ID ผู้ใช้ที่ระบุ

## ไวยากรณ์

`enotifyevent [-h] [user-ID]`

`notifyevent [-h] [user-ID]`

## คำอธิบาย

สคริปต์ `enotifyevent` จะส่งคืนข้อความในรูปของภาษาอังกฤษ ภาษาที่ข้อความสคริปต์ `notifyevent` ถูกส่งคืนขึ้นอยู่กับ คำติดตั้งโลแคล

สคริปต์เหล่านี้ดึงข้อมูลเหตุการณ์ที่ติดประกาศโดย event response resource manager (ERRM) ในตัวแปรสถานะแวดล้อมที่ถูกสร้างขึ้นโดย ERRM เมื่อเหตุการณ์เกิดขึ้น สคริปต์เหล่านี้สามารถใช้ในการดำเนินการที่รันโดย รีซอร์สการตอบกลับเหตุการณ์ ซึ่งยังสามารถใช้เป็นเท็มเพลตเพื่อสร้างการดำเนินการ ที่กำหนดโดยผู้ใช้

ข้อมูลเหตุการณ์ที่ถูกส่งกลับเกี่ยวกับตัวแปรสถานะแวดล้อม ERRM และ รวมถึงข้อมูลต่อไปนี้:

### Local Time

เวลาเมื่อเหตุการณ์หรือเวลาที่ผู้ใช้ใหม่ถูกพบ ตัวแปรสถานะแวดล้อมจริง ที่ระบุโดย ERRM คือ ERRM\_TIME ค่านี้ถูกแปลและแปลง เป็นรูปแบบที่อ่านได้ ก่อนถูกแสดง

ใน AIX สคริปต์เหล่านี้ ใช้คำสั่ง `mail` เพื่อส่งข้อมูลเหตุการณ์ ไปยัง ID ผู้ใช้ที่ระบุ เมื่อระบุ ID ผู้ใช้ไว้ ID นั้นจะถูกสมมุติว่าถูกต้อง และถูกใช้โดยไม่มี การตรวจสอบความถูกต้อง หาก ID ผู้ใช้ไม่ได้ระบุไว้ ผู้ใช้ที่กำลังรันคำสั่ง ถูกใช้เป็นค่าดีฟอลต์

`user-ID` คือ ID เพื่อเลือกของผู้ใช้ที่รายละเอียดของเหตุการณ์ จะถูกเมล หากไม่ได้ระบุ `user-ID` ไว้ ผู้ใช้ที่กำลังรันคำสั่งนี้ ถูกใช้ตามค่าดีฟอลต์

## แฟล็ก

`-h` เขียนคำสั่งการใช้สคริปต์ไปที่เอาต์พุตมาตรฐาน

## พารามิเตอร์

### `log_file`

ระบุชื่อไฟล์ที่ข้อมูลเหตุการณ์ถูกบันทึก พารามิเตอร์สำหรับพารามิเตอร์ `log_file` ควรถูกระบุ

สำหรับ AIX `log_file` จะถือว่าเป็นการบันทึกแบบวนซ้ำและมีขนาดคงที่ 64KB เมื่อ `log_file` เต็ม รายการใหม่จะถูกเขียน ทับรายการเก่าที่สุดที่มีอยู่

สำหรับ แพลตฟอร์มอื่นๆ ขนาดของ `log_file` นั้น ไม่จำกัด และจะไม่มีการเขียนทับตัวเอง ขนาดไฟล์ จะเพิ่มขึ้นแบบไม่จำกัด ยกเว้นว่า ผู้ดูแลระบบลบรายการเป็นระยะๆ

หาก `log_file` มีอยู่ก่อนแล้ว รายละเอียดของเหตุการณ์จะถูกต่อท้าย บันทึกการทำงานนั้น ถ้าไม่มี `log_file`, ไฟล์จะถูกสร้างเพื่อที่ข้อมูลเหตุการณ์ สามารถถูกเขียนลงไปได้

## สถานะออก

0 คำสั่งถูกรันได้เป็นผลสำเร็จ

## ข้อจำกัด

1. สคริปต์เหล่านี้ต้องถูกรันบนโนหนดโดยที่ERRM กำลังรันอยู่
2. คำสั่ง mail ถูกใช้เพื่ออ่านไฟล์

## เอาต์พุตมาตรฐาน

เมื่อมีการระบุแฟล็ก **-h** คำสั่งการใช้สคริปต์จะถูก เขียนไปที่เอาต์พุตมาตรฐาน

## ตัวอย่าง

1. คุณสามารถใช้คำสั่ง mail เพื่ออ่านเนื้อหาของ รายละเอียดเหตุการณ์ ตัวอย่างต่อไปนี้แสดงวิธีที่เหตุการณ์เตือนสำหรับระบบไฟล์ /var (รีซอร์สระบบไฟล์) ถูกจัดรูปแลลและ บันทึกการทำงานไว้:

```
=====  
Event reported at Sun Mar 26 16:38:03 2002
```

```
Condition Name:      /var space used  
Severity:           Warning  
Event Type:         Event  
Expression:         PercentTotUsed>90
```

```
Resource Name:      /var  
Resource Class Name: IBM.FileSystem  
Data Type:          CT_UINT32  
Data Value:         91
```

## Location

```
/usr/sbin/rsct/bin/enotifyevent  
มีสคริปต์ notifyevent
```

```
/usr/sbin/rsct/bin/notifyevent  
มีสคริปต์ notifyevent
```

---

## คำสั่ง enq

### วัตถุประสงค์

จัดคิวไฟล์

### ไวยากรณ์

เมื่อต้องการ ประมวลผลไฟล์

```
enq [-] [-B CharacterPair] [-c] [-C] [-G] [-j] [-m Text] [-M File] [-n] [-N Number] [-o Option] [-P Queue] [-r] [-R Number] [-t "User"] [-T Title] [-Y] [-Z Name] File
```

เมื่อต้องการเปลี่ยนลำดับความสำคัญของงานพิมพ์

```
enq -a Number -# JobNumber
```

เมื่อต้องการแสดงสถานะ

```
enq [-q|-A][ -L][ -W][ -e][ -# JobNumber][ -u Name][ -w Seconds][ -s]
```

เมื่อต้องการเปลี่ยนคิวและสถานะ daemon ของคิว

```
enq [-d][ -D][ -G][ -K][ -L][ -q|-A][ -U]
```

เมื่อต้องการ ยกเลิกอ็อปชัน

```
enq [-X][ -xNumber][ -PPrinter]
```

เมื่อต้องการพัก รีลีสหรือย้ายงานพิมพ์ไปที่คิวอื่น

```
enq { -h|-p|-Q NewQueue } { -# JobNumber[ -P Queue ]|-u User|-P Queue }
```

เมื่อต้องการเข้าคิวและพักงานพิมพ์

```
enq -H File ...
```

## คำอธิบาย

คำสั่ง `enq` เป็นยูทิลิตี้ที่เนกประสงค์สำหรับการจัดคิวการร้องขอของรีจิสเตอร์ที่แบ่งใช้ ปกติคืออุปกรณ์การพิมพ์ ใช้คำสั่ง `enq` เพื่อจัดคิวการร้องขอ ยกเลิกการร้องขอ เปลี่ยนระดับความสำคัญของเมื่อต้องการ และแสดงสถานะของคิวและอุปกรณ์

คำสั่ง `enq` มีห้าแผนภาพไวยากรณ์ ต่างกัน เนื่องจากแฟล็กทั้งหมดไม่ได้ถูกกำหนดให้ทำงานร่วมกัน บางส่วนของ แฟล็กเหล่านี้ ถูกกำหนดไว้สำหรับการประมวลผลไฟล์และรับ `FileName` เป็นอ็อปชัน แฟล็กอื่นถูกใช้สำหรับการเปลี่ยนระดับความสำคัญของงานพิมพ์ การแสดงสถานะ การเปลี่ยนสถานะของคิวหรือ คิว daemon และการยกเลิกงานพิมพ์

เมื่อต้องการจัดคิวไฟล์บนคิวเจาะจง ให้ใช้แฟล็ก `-P (-P Queue)` ถ้ามากกว่าหนึ่งเซอรั่วอุปกรณ์ต่อคิว คุณสามารถร้องขออุปกรณ์ได้โดยการระบุอุปกรณ์นั้น (`:device`) หลังชื่อของคิว ถ้าคุณไม่ระบุอุปกรณ์ งานจะถูกส่ง ไปที่อุปกรณ์ที่ใช้ได้อุปกรณ์แรก ถ้าคุณไม่ระบุไฟล์ คำสั่ง `enq` คัดลอกอินพุตมาตรฐานลงในไฟล์และจัดคิว เพื่อการพิมพ์

คำร้องขอคำสั่ง `enq` สามารถมีข้อความตัวดำเนินการ ที่เชื่อมโยง คุณลักษณะนี้มีประโยชน์ในสภาวะแวดล้อม แบบกระจายหรือบนระบบที่มีผู้ใช้หลายคน ข้อความถูกใช้เพื่อแจ้งกับ พรินเตอร์ถึงข้อมูลเช่น การร้องขอเพื่อโหลดฟอร์มพิเศษ หรือกระดาษสีอื่น ลงในพรินเตอร์ก่อนจะอนุญาตให้พิมพ์งาน ข้อความเหล่านี้ถูกระบุด้วยแฟล็ก `-m` และ `-M` คำสั่ง `qdaemon` ประมวลผล การร้องขอ `enq` เมื่อ `qdaemon` พร้อมทั้งจะเริ่มการร้องขอที่มีข้อความที่เชื่อมโยง, ระบบ แสดงข้อความบนคอนโซลของเครื่องที่กระบวนการ `qdaemon` รันอยู่ ข้อความของเมสเสจมาพร้อมกับพร้อมท์ ที่แจ้งแก่พรินเตอร์โอเปอเรเตอร์ ถึงวิธีส่งสัญญาณให้กับการร้องขอเพื่อ ดำเนินต่อ หรือวิธียกเลิกการร้องขอ

การแสดงผลที่ สร้างโดยคำสั่ง `enq -A` มีสองรายการสำหรับ รีโมตคิว รายการแรกมีคิวบนโลคัลของไคลเอ็นต์ และชื่ออุปกรณ์โลคัลและข้อมูลสถานะ รายการที่สองจะตามหลังมาในทันที ซึ่งมีชื่อคิวโลคัลของไคลเอ็นต์ (อีกครั้ง) แล้วตามด้วยชื่อคิวแบบรีโมต งานใดๆ ที่ส่งไปยังรีโมตคิว จะถูกแสดงก่อนบนฝั่งโลคัลและถูกย้าย ไปยังอุปกรณ์รีโมตเมื่องานถูกประมวลผลบนเครื่องรีโมต

เนื่องจากคำสั่งสถานะสื่อสารกับเครื่องรีโมต การแสดงสถานะอาจดูเหมือนจะหยุดเป็นบางครั้ง เมื่อรอการตอบสนอง จากเครื่องรีโมต คำสั่งจะหมดเวลาในที่สุด หากไม่สามารถสร้างการเชื่อมต่อระหว่างสองเครื่อง

**Notes:**

1. ก่อนที่คุณจะสามารถจัดคิวไฟล์ คุณต้องมีสิทธิ์ เข้าถึงเพื่ออ่านไฟล์ เมื่อต้องการลบไฟล์, (ดูที่แฟล็ก -r) คุณยังต้องมีสิทธิ์ เข้าถึงเพื่อเขียน กับไดเรกทอรีที่มี ไฟล์
2. ถ้าคุณต้องการเปลี่ยนแปลงไฟล์ต่อหลังจากเรียกใช้คำสั่ง **enq** แต่ก่อนที่จะถูกพิมพ์ คุณต้องใช้แฟล็ก **-c**
3. เมื่อทำการจัดคิวไฟล์บนพรีนเตอร์ แฟล็กสามารถถูก กระจายไปในทุกลำดับ
4. แฟล็ก **-d** และ **-G** มีผลในทันที ข้อผิดพลาดทางไวยากรณ์ปรากฏขึ้นก่อนที่แฟล็ก บนบรรทัดคำสั่งเหล่านี้ถูกรายงาน ข้อผิดพลาดทางไวยากรณ์ปรากฏขึ้นหลังจากแฟล็ก บนบรรทัดคำสั่งเหล่านี้ถูกข้าม

**แฟล็ก**

**อ็อปชัน การประมวลผลไฟล์**

ถ้าคุณให้รายชื่อไฟล์กับคำสั่ง **enq** คำสั่งจะจัดคิวทั้งหมดเพื่อการประมวลผลไฟล์บนอุปกรณ์ดีฟอลต์ หรือบนอุปกรณ์ที่ระบุ

**ไอเท็ม**

- คำอธิบาย  
ทำให้คำสั่ง **enq** ทำงานเหมือนเป็นตัวกรอง คำสั่ง **enq** อ่านอินพุตมาตรฐานโดยอัตโนมัติถ้าคุณ ไม่ได้ระบุไฟล์ แต่ ถ้าคุณระบุไฟล์ คุณยังสามารถใช้ เส้นประ (-) เพื่อบังคับให้คำสั่ง **enq** อ่านอินพุตมาตรฐาน เส้นประ (-) จริงๆ แล้วไม่ใช่แฟล็ก แต่เป็นชนิดชื่อไฟล์พิเศษ ดังนั้น จึงต้องระบุหลังจากระบุแฟล็กอื่นทั้งหมด บนบรรทัดรับคำสั่ง ควบคุมการพิมพ์ของเบิร์สเพจตามค่าของ *CharacterPair* ดังนี้ (n = never, a = always, g = group อักษรแรก สำหรับส่วนหัว อักษรที่สองสำหรับส่วนท้าย)

**-B CharacterPair**

HT	คำอธิบาย
nn	ไม่มีส่วนหัว ไม่ส่วนท้าย
na	ไม่มีส่วนหัว มีส่วนท้ายทุกไฟล์
ng	ไม่มีส่วนหัว มีส่วนท้ายที่ตำแหน่งสิ้นสุดงาน
an	มีส่วนหัวทุกไฟล์ ไม่มีส่วนท้าย
aa	มีส่วนหัวและส่วนท้ายทุกไฟล์ในงาน
ag	มีส่วนหัวทุกไฟล์ มีส่วนท้ายหลังจากงาน
gn	มีส่วนหัวที่จุดเริ่มต้นของงาน ไม่มีส่วนท้าย
ga	มีส่วนหัวที่จุดเริ่มต้นของงาน มีส่วนท้ายทุกไฟล์
gg	มีส่วนหัวที่จุดเริ่มต้นของงาน มีส่วนท้ายที่จุดสิ้นสุดของงาน

stanza ส่วนหัวและส่วนท้ายในไฟล์ /etc/qconfig กำหนดการดูแลดีฟอลต์ของเบิร์สเพจ

**-c**

หมายเหตุ: ในสภาวะแวดล้อมการพิมพ์แบบรีโมต ดีฟอลต์คือการพิมพ์เพจส่วนหัวและไม่มี เพจส่วนท้าย คัดลอกไฟล์ เมื่อต้องการประหยัดพื้นที่ดิสก์ คำสั่ง **enq** จัดจำชื่อไฟล์ แต่ไม่ทำการคัดลอกไฟล์จริงๆ ใช้แฟล็ก **-c** ถ้าคุณต้องการทำการเปลี่ยนแปลง ไฟล์ต่อ ขณะที่คุณรอการพิมพ์สำเนาปัจจุบัน

## ไอเท็ม

-C

### คำอธิบาย

ระบุว่าคำสั่ง `mail` ถูกใช้แทนคำสั่ง `write` สำหรับข้อความแสดงข้อผิดพลาด และการแจ้งเตือนงานเสร็จสมบูรณ์ (การใช้แฟล็กนี้มีประโยชน์สำหรับการเขียนแอสกีเพลคชัน PostScript เนื่องจากให้ผลสะท้อนที่ดีกว่าจากพรินเตอร์) ข้อความแสดงข้อผิดพลาดและข้อความความสำเร็จ (ทั้งสองถูกสร้างโดยคำสั่ง `piobe`) และข้อมูลที่อ่านจากพรินเตอร์ ถูกส่งกลับโดยเมลเช่นกัน

แฟล็ก -C ใช้กับงานพิมพ์โลคัลเท่านั้น หากคุณต้องการได้รับแจ้งเมื่องานที่ส่งไปยัง พรินเตอร์แบบรีโมตเสร็จสิ้น ให้ใช้แฟล็ก -n เพื่อรับข้อความเมล

หมายเหตุ: มีบางข้อความที่ไม่สามารถเปลี่ยนทิศทางจาก `qdaemon` และ เครื่องพิมพ์ back-end ได้เลย ข้อความเหล่านี้เป็นข้อผิดพลาดระบบ และถูกส่งไปยังไฟล์ `/dev/console` โดยตรง

-j

ระบุว่าข้อความ หมายถึงงานคือ: `gnk`, โดยที่ `gnk` คือหมายเลขงานที่กำหนด ที่จะถูกแสดงกับเอาต์พุต มาตรฐาน ซึ่งเกิดขึ้นเมื่อ งานถูกส่งไปยังคิวการพิมพ์โลคัลเท่านั้น

-m Text

ส่งข้อความโอเปอเรเตอร์ด้วยการร้องขอคำสั่ง `enq` ข้อความที่ระบุมีเมสเสจ

-M File

ส่งข้อความโอเปอเรเตอร์ด้วยการร้องขอคำสั่ง `enq` ไฟล์ที่ระบุมีข้อความของเมสเสจ

-n

แจ้งเตือนคุณเมื่องานพิมพ์ของคุณเสร็จสิ้น ถ้าแฟล็ก -t ถูกใช้ด้วย คำสั่ง `enq` ยังแจ้งเตือนผู้ใช้ว่าการร้องขอเป็นของผู้ใช้ใด (ดูที่แฟล็ก -t)

-N Number

พิมพ์ `Number` สำเนาของไฟล์ โดยปกติ ไฟล์ถูกพิมพ์เพียงครั้งเดียว

-o Option

ระบุว่าแฟล็กเฉพาะสำหรับ backend ถูกผ่าน ไปยัง backend ตัวนั้น สำหรับแต่ละคิวจะมีแฟล็กที่ไม่ได้อธิบายไว้

ในส่วนนี้ที่สามารถรวมอยู่ในบรรทัดคำสั่ง `enq` ดูที่คำสั่ง `piobe` สำหรับรายชื่อของแฟล็กเหล่านี้

-P Queue

ระบุคิวที่งานถูกส่งไป อุปกรณ์เฉพาะบนคิว สามารถระบุถึงได้โดยพิมพ์ -P Queue:Device

-r

ลบไฟล์หลังจากพิมพ์เสร็จแล้ว

-R Number

เซ็ระดับความสำคัญของงานปัจจุบันเป็น `Number` แฟล็กนี้ถูกใช้ขณะทำการส่งงาน ใช้แฟล็ก -a เพื่อเปลี่ยนระดับความสำคัญหลังจากงานถูกส่ง ตัวเลขที่สูงกว่ากำหนดระดับความสำคัญที่สูงกว่า ระดับความสำคัญที่พอลต์คือ 15

ระดับความสำคัญ สูงสุดคือ 20 สำหรับผู้ใช้ส่วนใหญ่และ 30 สำหรับผู้ใช้ที่มีสิทธิผู้ใช้ `root`

เลเบลเอาต์พุตสำหรับส่งไปยัง `User` โดยปกติ เอาต์พุตถูกเลเบลสำหรับจัดส่งให้กับชื่อผู้ใช้ของบุคคล ที่ส่งการร้องขอคำสั่ง `enq` ค่าของ `User` ต้องเป็นค่าเดียวกับข้อกำหนดเดียวกันของ ID ผู้ใช้ปกติ

-t "User"

ใส่หัวเรื่องบนเพจส่วนหัวและแสดงเมื่อแฟล็ก -q ถูกระบุ โดยปกติหัวเรื่องงานคือ ชื่อไฟล์ ถ้าคำสั่ง `enq` อ่านจาก

-T Title

อินพุตมาตรฐาน ชื่องานคือ `STDIN.#` โดยที่ # คือ process ID ของคำสั่ง `enq`

-Y

แจ้งกับคำสั่ง `enq` ให้ละเว้นส่วนที่เหลือของ บรรทัดคำสั่งหลังจากแฟล็กนี้ มีประโยชน์สำหรับการพิจารณาว่า

คิวใช้ได้หรือไม่ (ถ้าอยู่ในไฟล์ `/etc/qconfig`) ตัวอย่าง พิมพ์ `enq -P lp4 -Y` ส่งคืนค่า `exit` เป็น 0 ถ้าพรินเตอร์ราย

บรรทัด `lp4` คือคิวที่ใช่ได้; หรือมีค่านั้นส่งคืนค่าที่ไม่ใช่ศูนย์ การใช้ แฟล็กนี้ยังดีสำหรับการบังคับคำสั่ง `qdaemon`

ให้วิเคราะห์ไฟล์ `/etc/qconfig` อีกครั้ง

-Z Name

ระบุผู้เริ่มงานพิมพ์รีโมต

## อ็อปชันลำดับความสำคัญงานพิมพ์

### ไอเท็ม

-a Number

### คำอธิบาย

เปลี่ยนระดับความสำคัญของงานที่กำหนดชื่อเป็น `Number` งาน ต้องถูกส่ง สำหรับการพิมพ์ก่อนเข้าสู่คำสั่ง `enq` ที่มีแฟล็กนี้

ดูที่แฟล็ก -R สำหรับรายละเอียดของระดับความสำคัญ ใช้แฟล็ก -# เพื่อระบุหมายเลขงาน แฟล็กนี้จะถูกต้อง สำหรับงาน

พิมพ์บนโลคัล

-#JobNumber

ระบุหมายเลขงานที่ใช้โดยคำสั่ง `enq -q` หรือคำสั่ง `enq -a` และแสดงเฉพาะงานที่ระบุ ในเอาต์พุตสถานะ

### หมายเหตุ:

1. ระบุ -P Queue เพื่อลบล้างค่าพรินเตอร์ปลายทางที่เป็นดีพอลต์
2. หากงานที่ 1, 2 และ 3 อยู่ในคิวของเครื่องพิมพ์ และคุณ ระบุว่าต้องการสถานะของงานที่ 3 ขณะทำงานที่ 1 กำลังรันอยู่ ข้อมูลสถานะ จะแสดงงานที่ 1 และงานที่ 3 ไม่ใช่แสดงเฉพาะงานที่ 3
3. หากคุณระบุหมายเลขงานที่ไม่มีอยู่ ระบบจะแสดงหมายเลขงานปัจจุบันบนคิวแทนข้อความแสดงข้อผิดพลาด

## แสดงอ็อปชันสถานะ

ไอเท็ม	คำอธิบาย
-A	จัดเตรียมสถานะของคิวทั้งหมด ซึ่งเหมือนการรันคำสั่ง <code>enq -q</code> ครั้งเดียวสำหรับแต่ละคิวในไฟล์ <code>qconfig</code>
-e	แยกข้อมูลสถานะออกจากคิวที่ไม่ได้อยู่ภายใต้การควบคุมของคำสั่ง <code>qdaemon</code> สถานะจากคิวดังกล่าวอาจอยู่ในรูปแบบที่ต่างกัน แฟล็ก <code>-e</code> สามารถใช้กับการรวมกันของแฟล็ก
-L	ระบุสถานะ <code>long</code> แฟล็กนี้สามารถใช้กับแฟล็ก <code>-A</code> , <code>-q</code> หรือ <code>-W</code> หากใช้แฟล็ก <code>-L</code> และแฟล็ก <code>-W</code> พร้อมกัน ผลลัพธ์จะแสดงสถานะแบบยาวของงานพิมพ์ ในรูปแบบที่คั่นด้วยเครื่องหมายเซมิโคลอน- ใช้แฟล็ก <code>-L</code> เพื่อแสดงไฟล์หลายไฟล์ ที่จะถูกพิมพ์ในงานพิมพ์เดียว
-q	แสดงสถานะของดีฟอลต์คิว ตัวแปรสภาวะแวดล้อม <code>LPDEST</code> และ <code>PRINTER</code> ควบคุมชื่อของพริเตอร์ดีฟอลต์ ถ้าตัวแปรสภาวะแวดล้อม <code>LPDEST</code> มีค่า ค่านั้นจะถูกใช้ก่อนเสมอ ถ้าตัวแปร <code>LPDEST</code> ไม่มีค่าคำสั่ง <code>enq</code> ใช้ตัวแปรสภาวะแวดล้อม <code>PRINTER</code> ถ้าตัวแปรสภาวะแวดล้อม <code>PRINTER</code> ไม่มีค่า, คำสั่ง <code>enq</code> จะใช้ระบบดีฟอลต์
	<b>Notes:</b>
	1. ใช้แฟล็ก <code>-P Queue</code> กับแฟล็ก <code>-q</code> เพื่อแสดงสถานะของคิวเฉพาะ
	2. อีพซันบรรทัดรับคำสั่งปลายทางจะแทนที่ทั้งตัวแปรสภาวะแวดล้อม <code>LPDEST</code> และ <code>PRINTER</code>
-s	มีสถานะของคิวงานพิมพ์โดยไม่แสดงไฟล์
-u Name	ระบุชื่อผู้ใช้ที่จะพิมพ์สถานะงานพิมพ์
-w Seconds	ระบุเอาต์พุตต่อเนื่องของสถานะคิว อัปเดตจอภาพ ทุก <code>Seconds</code> ที่ระบุจนคิวว่าง (ดูที่ คำสั่ง <code>lpq</code> ) เมื่อ คิวว่าง กระบวนการหยุดทำงาน แฟล็กนี้ถูกใช้เฉพาะกับแฟล็ก <code>-q</code> หรือแฟล็ก <code>-A</code> หรือแฟล็ก <code>-L</code>
-W	ระบุรูปแบบสถานะ <code>wide</code> กับชื่อคิว, ชื่อ อุปกรณ์ และหมายเลขงาน ที่ยาวกว่า แฟล็กนี้สามารถใช้กับแฟล็ก <code>-A</code> , <code>-q</code> หรือ <code>-L</code> หากใช้แฟล็ก <code>-L</code> และ <code>-W</code> พร้อมกัน ผลลัพธ์จะแสดงสถานะแบบยาวของงานพิมพ์ ในรูปแบบที่คั่นด้วยเครื่องหมายเซมิโคลอน-

## เปลี่ยนอ็อปชันสถานะคิวและสถานะ daemon ของคิว

ไอเท็ม	คำอธิบาย
-d	รันคำสั่ง <code>digest</code> บนไฟล์ <code>/etc/qconfig</code> เมื่อการวิเคราะห์เสร็จสมบูรณ์ การเปลี่ยนแปลงกับไฟล์ <code>/etc/qconfig</code> มีผลใน ไฟล์ <code>/etc/qconfig.bin</code> ผู้ใช้ต้องมีสิทธิ์ผู้ใช้ <code>root</code> เพื่อรันอ็อปชันนี้

นอกเหนือจากแฟล็กที่ใช้ได้ก่อนหน้านี้สำหรับ ผู้ใช้ทั้งหมด คำสั่ง `enq` รับแฟล็กต่อไปนี้เมื่อ ถูกป้อนโดยผู้ใช้ที่มีสิทธิ์ผู้ใช้ `root` สิทธิผู้ใช้ `Root` หมายถึงคุณเป็น `root` หรือคุณเป็นสมาชิกกลุ่ม `printq`

หมายเหตุ: แฟล็ก ต่อไปนี้สามารถใช้บนงานพิมพ์โลคัลเท่านั้น

ไอเท็ม	คำอธิบาย
-D	Device DOWN. ปิดอุปกรณ์ที่เชื่อมโยงกับคิว กระบวนการ <code>qdaemon</code> ไม่ส่งงานไปที่อุปกรณ์อีกต่อไป และการป้อนคำสั่ง <code>enq -q</code> แสดงสถานะเป็น DOWN งานใดที่รันอยู่ในปัจจุบันบน อุปกรณ์ได้รับอนุญาตให้ทำงานจนเสร็จสิ้น
-G	Die GRACEFULLY ลีนสุดกระบวนการ <code>qdaemon</code> หลังจาก งานที่รันอยู่ในปัจจุบันทั้งหมดเสร็จสิ้น การใช้แฟล็กนี้เพียงวิธีเดียว ที่จะปิดกระบวนการ <code>qdaemon</code> ได้อย่างเรียบร้อย การใช้คำสั่ง <code>kill</code> อาจทำให้เกิดปัญหา เช่นงานค้างอยู่ในคิว

ถ้ากระบวนการ `qdaemon` รันอยู่ภายใต้ `srcmstr` (ดีฟอลต์ default configuration), `enq -G` ไม่ป้องกัน `qdaemon` จากการถูกรีสตาร์ทโดยอัตโนมัติ คุณต้องใช้คำสั่ง `chssys` ซึ่งเปลี่ยนดีฟอลต์ configuration และป้องกัน การรีสตาร์ทอัตโนมัติของกระบวนการ `qdaemon` คำสั่ง ต่อไปนี้:

```
chssys -s qdaemon -0
```

ถูกเรียกใช้ก่อนคำสั่ง `enq -G` ป้องกันการรีสตาร์ทโดยอัตโนมัติของคำสั่ง `qdaemon`

คำสั่ง ต่อไปนี้:

```
startsrc -s qdaemon
```

รีสตาร์ทกระบวนการ `qdaemon` ด้วยตัวเอง

-K	ทำงานเหมือนกับแฟล็ก <code>-D</code> ยกเว้นงานปัจจุบันทั้งหมดจะถูก kill งานยังคงอยู่ในคิว และถูกรันอีกครั้งเมื่ออุปกรณ์ถูกเปิด
-L	ระบุสถานะ <code>long</code> แฟล็กนี้สามารถใช้กับแฟล็ก <code>-A</code> , <code>-q</code> หรือ <code>-W</code> ใช้แฟล็ก <code>-L</code> เพื่อแสดงไฟล์หลายไฟล์ ที่จะถูกพิมพ์ในงานพิมพ์เดียว
-U	เปิดอุปกรณ์ที่เชื่อมโยงกับคิว กระบวนการ <code>qdaemon</code> ส่งงานไปยังคิวอีกครั้ง และการป้อน คำสั่ง <code>enq -q</code> แสดง สถานะเป็น ready

หมายเหตุ: หากมีมากกว่าหนึ่งอุปกรณ์ที่เชื่อมโยงกับคิว คุณต้องระบุอุปกรณ์และคิวเมื่อคุณใช้แฟล็ก -D, แฟล็ก -K และแฟล็ก -U ตัวอย่าง การป้อน -P 1p:1pd กำหนดอุปกรณ์เดิม เฉพาะถ้าไม่มี อุปกรณ์อื่นในคิวนั้น

### ตัวเลือกของการยกเลิก

ไอเท็ม	คำอธิบาย
-X	ยกเลิกการพิมพ์งานของคุณ ถ้าคุณมีสิทธิ์ผู้ใช้ root งานทั้งหมดบนคิวที่ระบุจะถูกลบ แฟล็กนี้ใช้ได้เฉพาะกับงานพิมพ์ ไลคัล
-x Number	ยกเลิกการพิมพ์ของงานที่ระบุ Number
-P Printer	ระบุ Printer ที่งานทั้งหมด หรือหมายเลขงานที่เลือกที่จะถูกยกเลิก

ข้อควรระวัง: หากคุณมีสิทธิ์ผู้ใช้ root และไม่ได้รับคิว งานทั้งหมดบนทุกคิว จะถูกลบออก

### การพักและการรีลีสอ็อปชัน งานพิมพ์

ไอเท็ม	คำอธิบาย
-#JobNumber	กำหนดหมายเลขของงานพิมพ์ที่จะถูกพักหรือรีลีส
-h	พักงานพิมพ์ที่ระบุ
-H	คิวและพักไฟล์ที่ระบุด้วยพารามิเตอร์ File
-p	รีลีสงานพิมพ์ที่ระบุ
-P Queue	กำหนดคิวงานพิมพ์ที่จะถูกพักหรือรีลีส
-u User	กำหนดผู้ใช้ซึ่งงานพิมพ์กำลังจะถูกพักหรือรีลีส

### การย้ายอ็อปชันงานพิมพ์

ไอเท็ม	คำอธิบาย
-#JobNumber	กำหนดหมายเลขของงานพิมพ์ที่จะถูกย้าย
-P Queue	กำหนดคิวงานพิมพ์ที่จะถูกย้าย ค่าของตัวแปร Queue เป็นได้ทั้งชื่อคิวหรืออยู่ในรูปแบบ คิว:ชื่ออุปกรณ์
-Q NewQueue	กำหนดคิวเป้าหมายที่งานพิมพ์จะถูกย้าย ไป ค่าของตัวแปร NewQueue เป็นได้ทั้งในรูปแบบชื่อคิวหรืออยู่ในรูปแบบ คิว:ชื่ออุปกรณ์
-u User	กำหนดผู้ใช้ซึ่งงานพิมพ์จะถูกย้าย

### ความปลอดภัย

การตรวจสอบเหตุการณ์:

เหตุการณ์	ข้อมูล
ENQUE_admin	ชื่อคิว, ชื่ออุปกรณ์, ชื่องาน, ชื่อผู้ใช้

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

### ตัวอย่าง

- เมื่อต้องการพิมพ์ไฟล์ memo บน พรินเตอร์ดีฟอลต์ให้ป้อน:  
enq memo
- เมื่อต้องการพิมพ์ไฟล์ prog.c พร้อมกับหมายเลขหน้าให้ป้อน:  
pr prog.c | enq

คำสั่ง **pr** วางส่วนหัวไว้ที่ด้านบนสุดของแต่ละหน้า ซึ่งรวมถึง วันที่ไฟล์ถูกแก้ไขล่าสุด ชื่อของไฟล์และหมายเลขหน้า จากนั้นคำสั่ง **enq** พิมพ์ไฟล์

- เมื่อต้องการพิมพ์ไฟล์พร้อมกับหมายเลขหน้า อ่านข้อมูลจาก อินพุตมาตรฐาน ให้ป้อน:

```
pr x | enq -P bill -n -r fn1 - fn3
```

เส้นประ (-) ชื่อไฟล์ พิเศษบอกให้คำสั่ง **enq** อ่านจากอินพุตมาตรฐาน โดยปกติคำสั่ง **enq** จะไม่อ่านจากอินพุตมาตรฐาน หากมีชื่อไฟล์บนบรรทัดคำสั่ง และยังมีระบุดำดับ ในการพิมพ์ คำสั่ง **pr** สร้างไฟล์ **x** ในเวอร์ชันที่มีหมายเลขหน้าและผ่านไปยัง คำสั่ง **enq** ซึ่งสร้างไฟล์ชั่วคราวที่มี เอาต์พุตนั้นในไฟล์ **/var/spool/qdaemon**

คำสั่ง **enq** สร้างงานที่มีสี่ไฟล์และส่ง ไปยังคิวที่ชื่อ **bill** ซึ่งจะพิมพ์ไฟล์ **fn1** สองครั้ง จากนั้น จะพิมพ์ข้อมูลที่เป็นเอาต์พุต ของคำสั่ง **pr** สุดท้าย จะพิมพ์ไฟล์ **fn3** สี่ไฟล์จะถูกปฏิบัติเป็นหนึ่งงานเพื่อ จุดประสงค์ของเบิร์สเพจ การแจ้งเตือน ถูกส่ง (แฟล็ก -n) เมื่องานเสร็จสมบูรณ์ เนื่องจากแฟล็ก -r ถูกระบุ ไฟล์ **fn1** และ **fn3** ถูกลบเมื่องานเสร็จสมบูรณ์ ไฟล์ชั่วคราวที่สร้างโดย ไฟล์เส้นประ (-) ถูกลบเสมอ

คำสั่ง **pr** จะวาง ส่วนหัวไว้ที่ด้านบนสุดของแต่ละหน้า ซึ่งรวมถึง วันที่ไฟล์ถูกแก้ไขล่าสุด ชื่อของไฟล์และหมายเลขหน้า จากนั้นคำสั่ง **enq** พิมพ์ไฟล์

- เมื่อต้องการพิมพ์ไฟล์ **report** บนเครื่องพิมพ์ตัวถัดไปที่วาง ที่กำหนดค่าไว้สำหรับคิว **fred** ให้ป้อน:

```
enq -P fred report
```

- เมื่อต้องการพิมพ์ไฟล์หลายไฟล์ที่เริ่มต้นด้วย **sam** บน เครื่องพิมพ์ตัวถัดไปที่วางที่กำหนดค่าไว้สำหรับคิว **fred** ให้ป้อน:

```
enq -P fred sam*
```

ไฟล์ทั้งหมด ที่ขึ้นต้นด้วย **sam** จะถูกรวมไว้ในหนึ่งงานพิมพ์ คำสั่งสถานะปกติแสดงเฉพาะ ชื่อของงานพิมพ์ ซึ่งในกรณีนี้คือชื่อของไฟล์แรกในคิว นอกจากนี้ถูกระบุด้วยแฟล็ก -T เมื่อต้องการแสดงชื่อของไฟล์ทั้งหมดในงานพิมพ์ ให้ใช้ คำสั่ง สถานะ **long enq -A -L**

- เมื่อต้องการตรวจสอบคิวการพิมพ์เพื่อดูว่าไฟล์ยังคง รอการพิมพ์อยู่หรือไม่ ให้ป้อน:

```
enq -q
```

คำสั่งนี้แสดงสถานะของดีฟอลต์คิวของผู้ใช้ ถ้าไฟล์ยังไม่ถูกพิมพ์ ไฟล์ จะแสดงในรายการสถานะคิว ดีฟอลต์คิวระบบ ถูกกำหนด เป็นคิวแรกในไฟล์ **/etc/qconfig[.bin]** ผู้ใช้ สามารถเขียนทับค่าดีฟอลต์ของตัวเอง โดยการตั้งค่าและเอ็กซ์พอร์ตตัวแปรสถานะแวดล้อม **PRINTER**

- เมื่อต้องการแสดงสถานะของคิวที่ไม่ใช่คิวดีฟอลต์ **lp0** ให้ป้อน:

```
enq -q -P lp0
```

- เมื่อต้องการรับสถานะคิวแบบยาวให้ป้อน:

```
enq -L
```

- เมื่อต้องการรับสถานะของคิวทั้งหมดให้ป้อน:

```
enq -A
```

- เมื่อต้องการรับสถานะของคิวแบบยาวทั้งหมด ให้ป้อน:

```
enq -A -L
```

11. เมื่อต้องการรับสถานะของดีฟอลต์คิวในรูปแบบ wide ให้ป้อน:

```
enq -W
```

12. เมื่อต้องการรับสถานะ wide ของคิวทั้งหมด ให้ป้อน:

```
enq -W -A
```

13. เมื่อต้องการหยุดงานพิมพ์ (งานเป็นไฟล์หนึ่งไฟล์หรือมากกว่านั้น), ให้ป้อน:

```
enq -x 413
```

คำสั่งนี้ยกเลิกการร้องขอที่คุณสร้างขึ้นก่อน การพิมพ์งาน ตัวเลขได้มาจากรายการที่ได้รับโดยการป้อนคำสั่ง `enq -q` หากงานถูกพิมพ์อยู่ในขณะนี้ พรินเตอร์จะหยุดทันที ถ้างานยังไม่ถูกพิมพ์ งานจะถูก ลบออกจากคิว เพื่อที่จะไม่ถูกพิมพ์ ถ้างานไม่อยู่ใน คิว คำสั่ง `enq` แสดงข้อความเหมือนดัง ต่อไปนี้:

```
no such request from you -- perhaps it's done?
```

14. เมื่อต้องการยกเลิกการเชื่อมต่อพรินเตอร์จากระบบการคิว ให้ป้อน:

```
enq -P lp0:d1p0 -D
```

การป้อนคำสั่งนี้ หยุดการร้องขอคำสั่ง `enq` จากการถูกส่ง ไปที่พรินเตอร์ที่บริการคิว `lp0` ถ้าไฟล์ ถูกพิมพ์อยู่ในขณะนี้ ไฟล์ได้รับอนุญาตให้พิมพ์จนเสร็จสิ้น คุณต้องสามารถรันคำสั่ง `qadm` เพื่อรัน คำสั่ง `enq`

หมายเหตุ: เครื่องพิมพ์ที่ให้บริการคิวที่ระบุ ถูกตั้งชื่อโดยชื่อ stanza ของอุปกรณ์ที่ปรากฏในไฟล์ `/etc/qconfig[.bin]`

15. เมื่อต้องการพิมพ์หมายเลขหน้าโดยใช้ backend คำสั่ง `piobe` บนเครื่องพิมพ์ดีฟอลต์ให้ป้อน:

```
enq -o -p filename
```

แฟล็ก `-p` ไม่ถูกตรวจสอบโดยคำสั่ง `enq` แฟล็ก `-o` แจกแก่คำสั่ง `enq` ให้ส่งรายการถัดไป ซึ่งสามารถอยู่ในเครื่องหมายคำพูดได้ ไปที่แบ็คเอนด์โดยไม่มีการเปลี่ยนแปลง ดังนั้น คำสั่ง `enq` ผ่านแฟล็ก `-p` ไปที่กระบวนการ `qdaemon` ซึ่งส่งผ่านต่อไปที่แบ็คเอนด์ `piobe` แฟล็ก `-p` ทำให้ `piobe` รันตัวกรอง `/usr/bin/pr` เพื่อใช้หมายเลขหน้ากับเอกสาร ก่อนส่งข้อมูลให้กับอุปกรณ์ สามารถระบุหลายอ็อปชันในเครื่องหมายคำพูดที่นำหน้าโดยแฟล็ก `-o` หนึ่งแฟล็ก คุณยังสามารถระบุหลายอ็อปชันโดยไม่มีเครื่องหมายคำพูดโดยแต่ละอ็อปชัน นำหน้าด้วยแฟล็ก `-o`

16. สมมติว่าไฟล์ `qconfig` มีข้อมูลต่อไปนี้:

```
qname:
        device = fred
fred:
        file = /tmp/hello
        backend = /usr/bin/sh /usr/bin/diff
```

และกำหนดคำสั่งต่อไปนี้:

```
rm /tmp/hello
touch /tmp/hello
pr /etc/hosts|enq -P qname:fred - /etc/hosts
```

กระบวนการ `qdaemon` จะรันโปรแกรม `/usr/bin/diff` ด้วยสองอาร์กิวเมนต์ อาร์กิวเมนต์หนึ่งเป็นชื่อไฟล์ชั่วคราวและอีกอาร์กิวเมนต์เป็นไฟล์ `/etc/hosts` ความแตกต่าง ประการเดียวระหว่างทั้งสองไฟล์คือ ไฟล์หนึ่งถูกรันผ่านคำสั่ง `pr` ไฟล์ `/tmp/hello` จะมี ความแตกต่างระหว่างสองไฟล์ กระบวนการ `qdaemon` ไม่สร้างไฟล์ `/tmp/hello` ถ้าไฟล์ ไม่มีอยู่

17. คำสั่ง ต่อไปนี้:

```
enq -m'i want pink paper for this job' /etc/passwd
```

ส่งข้อความโอเปอเรเตอร์ที่ระบุไปที่คอนโซลของโอเปอเรเตอร์ ก่อนที่งานพิมพ์จะถูกพิมพ์ โอเปอเรเตอร์ต้องตอบสนองข้อความนี้ เพื่อดำเนินการต่อหรือยกเลิกงาน

```
enq -M pink /etc/passwd
```

คำสั่งนี้ ดำเนินการให้ผลเหมือนกัน มีเพียงข้อความถูกเก็บไว้ในไฟล์ชื่อ `pink`

18. เมื่อต้องการยกเลิกงานทั้งหมดในคิว `fred` ให้ป้อน:

```
enq -X -P fred
```

ถ้า ผู้ใช้ที่ป้อนคำสั่งนี้มีสิทธิ์ผู้ใช้ `root` งานทั้งหมดจาก คิว `fred` จะถูกลบ ถ้าผู้ใช้ไม่มี สิทธิ์ผู้ใช้ `root` เฉพาะงานของผู้ใช้ถูกลบออกจากคิว

19. เมื่อต้องการเข้าคิวไฟล์ชื่อ `MyFile` และส่งคั่นหมายเลขงาน `MyFile` ไปยังไฟล์ `fdf` ให้ป้อน:

```
enq -j MyFile
```

20. เมื่อต้องการพักหมายเลขงานพิมพ์ 310 ให้ป้อน:

```
enq -h -#310
```

เมื่อต้องการ รีเซ็ตงานพิมพ์ที่พักไว้หมายเลข 310 ให้ป้อน:

```
enq -p -#310
```

21. เมื่อต้องการพักงานพิมพ์ทั้งหมดบนคิว `lp0` ให้ป้อน:

```
enq -h -P lp0
```

เมื่อต้องการ รีเซ็ตคิว `lp0` ให้ป้อน:

```
enq -p -P lp0
```

22. เมื่อต้องการพักงานพิมพ์ทั้งหมดที่สร้างโดย `fred` ให้ป้อน:

```
enq -h -u fred
```

เมื่อต้องการ รีเซ็ตงานพิมพ์ ที่สร้างโดย `fred` ให้ป้อน:

```
enq -p -u fred
```

23. เมื่อต้องการย้ายหมายเลขงาน 318 ไปยังคิว `lp0` ให้ป้อน:

enq -Q lp0 -#318

แฟล็กที่ควบคุมการย้ายงานพิมพ์ทำงาน ในแบบเดียวกับแฟล็กที่ปักไฟล์พิมพ์ แฟล็กพักและตัวแปร แสดงอยู่ในตัวอย่างก่อนหน้า

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/sbin/qdaemon	การคิว daemon
/etc/qconfig	ไฟล์คอนฟิกูเรชันคิว
/var/spool/lpd/qdir/*	การรองขอคิว
/var/spool/lpd/stat/*	ข้อมูลเกี่ยวกับสถานะของอุปกรณ์
/var/spool/qdaemon/*	สำเนาชั่วคราวของไฟล์ที่จัดคิว
/etc/qconfig.bin	ไบนารีเวอร์ชันของไฟล์ /etc/qconfig ที่วิเคราะห์แล้ว

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mkque  
การเปลี่ยนแปลง หรือการแสดงคุณสมบัติคิว  
ข้อมูลของเครื่องพิมพ์เฉพาะ  
ระเบียบไฟล์ colon เครื่องพิมพ์

---

## คำสั่ง enroll

### วัตถุประสงค์

เซิร์ฟเวอร์ที่ ใช้เพื่อสร้างช่องสื่อสาร ที่ปลอดภัย

### ไวยากรณ์

enroll

### คำอธิบาย

คำสั่ง enroll สร้าง รหัสผ่านและรักษาความปลอดภัยช่องสื่อสาร ซึ่งข้อความสามารถถูกอ่าน ได้เพียงผู้รับที่กำหนดไว้ รหัสผ่าน ถูกใช้เพื่อรับเมลลับ

คำสั่ง enroll ถูกใช้กับ คำสั่ง xsend และ xget เพื่อส่งและรับเมลลับ คำสั่ง xsend ส่งเมลลับ คำสั่ง xget ถามรหัสผ่านของคุณและ ส่งเมลลับให้คุณ

### ตัวอย่าง

เมื่อต้องการเซิร์ฟเวอร์ให้ป้อน:

```
enroll
```

เมื่อมีพร้อมท์ให้ป้อนรหัสผ่านของคุณ อนุญาตให้ผู้อื่น บนระบบส่งเมลลับของคุณ ใช้คำสั่ง xget เพื่ออ่านเมลลับ

# ไฟล์

## ไอเท็ม

/var/spool/secretmail/User.key  
/usr/bin/enroll

คำอธิบาย  
มีคีย์ที่เข้ารหัสสำหรับผู้ใช้  
มีคำสั่ง enroll

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mail

คำสั่ง xget

คำสั่ง xsend

การส่งและการรับเมลความลับ

---

## คำสั่ง **enscript**

### วัตถุประสงค์

แปลงไฟล์ข้อความในรูปแบบ PostScript เพื่อการพิมพ์

### ไวยากรณ์

```
enscript [-1 -2 -c -g -k -l -m -o -q -r -B -G -K -R] [-b Header] [-f Font] [-fO CodeSet:Font] [-f1 CodeSet:Font] [-p Out] [-F Hfont] [-FO CodeSet:Font] [-F1 CodeSet:Font] [-L Lines] [-M MediaName] [-X CodesetName] [SpoolerOptions] [File ...]
```

### คำอธิบาย

คำสั่ง **enscript** อ่านไฟล์ข้อความ, แปลงเป็นรูปแบบ PostScript และสพูลไฟล์ สำหรับการพิมพ์บนพรินเตอร์ PostScript คุณสามารถรันคำสั่งนี้ เพื่อระบุฟอนต์, ส่วนหัว, อ็อพชันการจัดรูปแบบที่จำกัด, และอ็อพชันการสพูล

ตัวอย่างเช่น :

```
enscript -daleph bubble.txt
```

พิมพ์สำเนาของไฟล์ **bubble.txt** บนพรินเตอร์ชื่อ **aleph**, และ

```
enscript -2r finder.c
```

พิมพ์รายการแนวนอน two-up ของไฟล์ **finder.c** บนพรินเตอร์ดีฟอลต์

ตัวแปรสถานะแวดล้อม **ENSCRIPT** สามารถถูกใช้เพื่อระบุดีฟอลต์ค่า **ENSCRIPT** ถูกวิเคราะห์เป็นสตริงของอาร์กิวเมนต์ก่อนอาร์กิวเมนต์ที่ถูกแสดงบน บรรทัดคำสั่ง ตัวอย่างเช่น:

```
ENSCRIPT=' -fTimes-Roman8'
```

เซตขนาดชนิดรูปทรงและฟอนต์ดีฟอลต์ เป็น 8-point Times Roman

ข้อมูลที่มีขนาดของสื่อบันทึกต่างๆ สำหรับคำสั่ง **psdit** และคำสั่ง **enscript** จะมีอยู่ในไฟล์ **/usr/lib/ps/MediaSizes**

ข้อมูลต้องการรายการแต่ละรายการในไฟล์ **MediaSizes** ที่สามารถขอรับได้จาก **PostScript Printer** คำอธิบาย หรือ **PPD** ไฟล์ ที่ตรงกับพรินเตอร์ PostScript ที่ใช้กับ **TranScript** ไฟล์ **PPD** หาได้จาก Adobe Systems, Incorporated หน่วยวัดที่แยกจากไฟล์ **PPD** ถูกแสดงในหน่วยวัดของพรินเตอร์เรียกว่า point พ้อยต์ของพรินเตอร์คือ 1/72 ของนิ้ว

บรรทัดในไฟล์ **MediaSizes** เริ่มต้นด้วย ASCII \* (เครื่องหมายดอกจัน) ถูกละเว้นเมื่อ ชื่อขนาดสื่อบันทึกที่ตรงกัน มีอยู่ใน บรรทัดคำสั่งให้กับคำสั่ง **encrypt** และคำสั่ง **psdit**

แต่ละรายการในไฟล์ **MediaSizes** มี 8 หรือ 9 ฟیلด์ 8 ฟیلด์แรกจำเป็นสำหรับรายการทั้งหมด ฟیلด์ที่ 9 เป็นทางเลือก ฟิลด์จะถูก คั่นโดย ช่องว่าง ฟิลด์สำหรับแต่ละรายการมีดังต่อไปนี้:

ชื่อฟیلด์	คำอธิบาย
EntryName	มีสตริงอักขระที่จะจับคู่กับชื่อสื่อที่จัดเตรียม โดยแฟล็ก -M กับคำสั่ง <b>encrypt</b> หรือคำสั่ง <b>psdit</b>
MediaWidth	ระบุความกว้างของสื่อเป็นพอยท์
MediaDepth	ระบุความลึกของสื่อเป็นพอยท์
ImageableLLX	ระบุพิกัด x มุมซ้ายทางด้านล่างของ imageable เป็นพอยท์
ImageableLLY	ระบุพิกัด y มุมซ้ายทางด้านล่างของ imageable เป็นพอยท์
ImageableURX	ระบุพิกัด x มุมขวาทางด้านบนของ imageable เป็นพอยท์
ImageableURY	ระบุพิกัด y มุมขวาทางด้านบนของ imageable เป็นพอยท์
PageRegionName	ระบุลำดับ PostScript สำหรับพรินเตอร์ เพื่อระบุขนาดของพื้นที่ imageable
PaperTrayName	ระบุลำดับ PostScript สำหรับพรินเตอร์ เพื่อเลือกถาด กระดาษ/สื่อบันทึก ฟิลด์นี้เป็นทางเลือก

หมายเหตุ: ลำดับสามารถเป็นตัวดำเนินการ PostScript จำนวนมากและคำสำหรับทั้งฟیلด์ PageRegionName และฟیلด์ PaperTrayName เมื่อต้องการระบุลำดับดังกล่าว ให้ใช้ ASCII " (อักขระ เครื่องหมายคำพูด) เพื่อคั่นลำดับ ทั้งหมด

ตารางต่อไปนี้แสดงตัวอย่างของรายการฟیلด์ ในไฟล์ **MediaSizes** :

ชื่อ	ค่าฟิลด์
จดหมาย	<b>Width</b> 612 <b>ความลึก</b> 792 <b>llx</b> 18 <b>lly</b> 17 <b>urx</b> 597 <b>ury</b> 776 <b>เพจ-ขอบเขต- ชื่อ</b> จดหมาย <b>กระดาษ- ถาด- ชื่อ</b> จดหมาย

ชื่อ	ค่าฟิลด์
Legal	<b>Width</b> 612 <b>ความลึก</b> 1008 <b>llx</b> 18 <b>lly</b> 17 <b>urx</b> 597 <b>ury</b> 992 <b>เพจ-ขอบเขต-ชื่อ</b> Legal <b>กระดาษ-ขนาด-ชื่อ</b> Legal

### PostScript Font Information

ตาราง PostScript Fonts for Transcript แสดงฟอนต์ที่ใช้ได้สำหรับคำสั่ง encript ชื่อฟอนต์ถูกระบุด้วย แฟล็กคำสั่ง **-F** และ **-f** ของ **encript** อีกชระแบบตัวอักษร สนใจขนาดตัวพิมพ์:

PostScript Fonts for Transcript

ชื่อฟอนต์	ตระกูลฟอนต์
AvantGarde-Book	AvantGarde
AvantGarde-Demi	AvantGarde
AvantGarde-DemiOblique	AvantGarde
AvantGarde-BookOblique	AvantGarde
Bookman-Demi	Bookman
Bookman-DemiItalic	Bookman
Bookman-Light	Bookman
Bookman-LightItalic	Bookman
Courier	Courier
Courier-Bold	Courier
Courier-BoldOblique	Courier
Courier-Oblique	Courier
Garamond-Bold	Garamond
Garamond-BoldItalic	Garamond
Garamond-Light	Garamond
Garamond-LightItalic	Garamond

PostScript Fonts for Transcript

ชื่อฟอนต์	ตระกูลฟอนต์
Helvetica	Helvetica
Helvetica-Bold	Helvetica
Helvetica-Oblique	Helvetica
Helvetica-BoldOblique	Helvetica
Helvetica-Narrow	Helvetica
Helvetica-Narrow-Bold	Helvetica
Helvetica-Narrow-BoldOblique	Helvetica
Helvetica-Narrow-Oblique	Helvetica
LubalinGraph-Book	Lubalin
LubalinGraph-BookOblique	Lubalin
LubalinGraph-Demi	Lubalin
LubalinGraph-DemiOblique	Lubalin

ชื่อฟอนต์	ตระกูลฟอนต์
Miryam-Iso	Miryam Iso
Miryam-IsoBold	Miryam Iso
Miryam-IsoBoldItalic	Miryam Iso
Miryam-IsoItalic	Miryam Iso
NarkissimIso	Narkissim Iso
NarkissimIso-Bold	Narkissim Iso
NarkissimIso-BoldItalic	Narkissim Iso
NarkissimIso-Italic	Narkissim Iso
NarkissTamIso	Narkiss Tam Iso
NarkissTamIso-Bold	Narkiss Tam Iso
NarkissTamIso-BoldItalic	Narkiss Tam Iso
NarkissTamIso-Italic	Narkiss Tam Iso
NewCenturySchlbk	NewCentury
NewCenturySchlbk-Bold	NewCentury
NewCenturySchlbk-Italic	NewCentury
NewCenturySchlbk-Roman	NewCentury
Optima	Optima

ชื่อฟอนต์	ตระกูลฟอนต์
Optima-Bold	Optima
Optima-BoldOblique	Optima
Optima-Oblique	Optima
Palatino-Bold	Palatino
Palatino-BoldItalic	Palatino
Palatino-Italic	Palatino
Palatino-Roman	Palatino
Rokaa	Rokaa
Rokaa-Bold	Rokaa
Rokaa-BoldItalic	Rokaa
Rokaa-Italic	Rokaa

ชื่อฟอนต์	ตระกูลฟอนต์
คำติดตั้ง	คำติดตั้ง
Setting-Bold	คำติดตั้ง
Setting-BoldItalic	คำติดตั้ง
Setting-Italic	คำติดตั้ง
ShalomIso	ShalomIso Iso
ShalomIso-Bold	ShalomIso Iso
ShalomIso-BoldItalic	ShalomIso Iso
ShalomIso-Italic	ShalomIso Iso
Souvenir-Demi	Souvenir
Souvenir-DemiItalic	Souvenir
Souvenir-Light	Souvenir
Souvenir-LightItalic	Souvenir
Times-Bold	Times
Times-BoldItalic	Times
Times-Italic	Times
Times-Roman	Times
Typing	Typing
Typing-Bold	Typing
Typing-BoldItalic	Typing

ชื่อฟอนต์	ตระกูลฟอนต์
Typing-Italic	Typing
สัญลักษณ์	(none)
ZapfChancery-MediumItalic	Zapf
ZapfDingbats	(none)

## พารามิเตอร์

### ไอเท็ม

*SpoolerOptions*

### คำอธิบาย

มีอ็อปชันสำหรับการสพูลไฟล์ที่พิมพ์ต่อไปนี้เป็นแฟล็ก *SpoolerOptions* :

`{-d|-P}Queue`

คิวเอาต์พุตไปที่คิวที่ระบุชื่อ

`-n Number`

สร้างจำนวนสำเนาที่ระบุตีพอลด์คือ 1

`-tTitle`

เซตชื่องานสำหรับใช้บนเพจแถบป้ายแรก

**ไฟล์**

ระบุไฟล์ข้อความที่จะถูกแปลงเป็นรูปแบบ PostScript ถ้าคุณปล่อยให้พารามิเตอร์นี้ว่างไว้คำสั่ง **encrypt** จะอ่านจากอินพุตมาตรฐาน

## แฟล็ก

### ไอเท็ม

`-1`

### คำอธิบาย

เซตใน 1 คอลัมน์ (ตีพอลด์)

`-2`

เซตใน 2 คอลัมน์

`-c`

ตัดปลายบรรทัดที่ยาวเกินความกว้างของหน้า โดยปกติ บรรทัดที่ยาวจะถูกตัดลงมาอยู่ในบรรทัดต่อมาบนหน้า

`-g`

ไม่มีการทำงานใด แต่แฟล็ก `-g` ยังคงมีไว้เพื่อความเข้ากันโดยย้อนหลัง

`-k`

เปิดใช้การป้อนกระดาษลงหน้า (ถ้าพรินเตอร์สนับสนุน) นี้ทำให้เอกสาร ธรรมดา (เช่นโปรแกรมแสดงในฟอนต์เดียว) พิมพ์เร็วขึ้น โดยคงการทำงานของพรินเตอร์ระหว่างหน้า

`-l`

จำลองหน้าการพิมพ์พรินเตอร์รายบรรทัด ความยาว 66 บรรทัดและละเว้น ส่วนหัว

`-m`

ส่งเมลหลังจากไฟล์ถูกพิมพ์

`-o`

แสดงอักขระที่หายไปถ้าคำสั่ง **encrypt** ไม่พบอักขระในฟอนต์

`-q`

ทำให้คำสั่ง **encrypt** ไม่รายงานเกี่ยวกับสิ่งที่กำลังทำ คำสั่ง **encrypt** ไม่สามารถรายงาน บนหน้า, ปลายทาง,

`-r`

อักขระที่ละเว้น และอื่นๆ ข้อผิดพลาดรุนแรงยังคงถูก รายงานไปที่เอาต์พุตข้อผิดพลาดมาตรฐาน

หมุนเอาต์พุต 90 องศา (โหมดแนวนอน) ใช้แฟล็กนี้สำหรับ เอาต์พุตที่ต้องการหน้ากว้าง หรือสำหรับการแสดงโปรแกรมเมื่อใช้รวม กับแฟล็ก `-2` ตัวอย่างต่อไปนี้จะแสดงวิธีหนึ่ง ในการรับการแสดงผลโปรแกรม:

```
encrypt -2r File . . .
```

`-B`

ละเว้นส่วนหัวของหน้า

`-G`

พิมพ์ในโหมด *gaudy*, ทำให้ ส่วนหัวของหน้า วันที่ และหมายเลขหน้า ถูกพิมพ์ในลักษณะจุดจาด โดยมีผลการทำงานลดลงเล็กน้อย

`-K`

ปิดใช้งานการป้อนหน้าลงหน้า (ตีพอลด์)

`-R`

พิมพ์ในโหมดแนวตั้ง (ไม่หมุน) ซึ่งเป็นตีพอลด์

`-bHeader`

เซตสตริงที่จะถูกใช้สำหรับส่วนหัวของหน้าเป็นคําของตัวแปร *Header* ส่วนหัวตีพอลด์ถูกสร้างจาก ชื่อไฟล์ วันที่แก้ไขล่าสุด และหมายเลขหน้า

## ไอเท็ม

-fFont

### คำอธิบาย

เซตฟอนต์ที่จะถูกใช้สำหรับส่วนเนื้อความของแต่ละหน้า ดีฟอลต์คือ Courier10, นอกจากมีการใช้โหมดที่หมุนสองคอลัมน์ ซึ่งดีฟอลต์ คือ Courier7

### หมายเหตุ:

1. ชื่อฟอนต์ PostScript (เช่น Times-Roman, Times-BoldItalic, Helvetica, Courier)
2. ขนาดพอยท์ (1 พอยท์ = 1/72 นิ้ว) ฟอนต์ถูกระบุในรูปแบบนี้: Courier-Bold8 is 8-point Courier Bold;

Helvetica12 คือ 12-พอยท์ Helvetica

-fO Codeset:Font

เซตชื่อโค๊ดเซตของอักขระ ซึ่งถูกเขียนลงในไฟล์ PostScript และฟอนต์ SBCS เพื่อใช้สำหรับเนื้อความของแต่ละหน้า ดีฟอลต์ ถูกกำหนดโดยไฟล์คอนฟิกูเรชัน /usr/lib/ps/transcript.conf สำหรับแต่ละโลแคล

-f1 Codeset:Font

เซตชื่อโค๊ดชุดอักขระ ซึ่งถูกเขียนลงในไฟล์ PostScript และฟอนต์ MBCS เพื่อใช้สำหรับเนื้อความของแต่ละหน้า ดีฟอลต์ ถูกกำหนดโดยไฟล์คอนฟิกูเรชัน /usr/lib/ps/transcript.conf สำหรับแต่ละโลแคล

-pOut

ทำให้ไฟล์ PostScript ถูกเขียนลงในไฟล์ที่กำหนดชื่อ แทนการทำสพูลเพื่อการพิมพ์ เนื่องจากเป็นกรณีพิเศษ การป้อนข้อมูลต่อไปนี้จะส่งไฟล์ PostScript ไปที่เอาต์พุตมาตรฐาน:

-FHfont

-D -

ตั้งค่าฟอนต์ที่จะถูกใช้สำหรับส่วนหัวของหน้า ดีฟอลต์คือ Courier Bold10

### หมายเหตุ: ค่ากำหนดฟอนต์มีสองส่วน:

- ชื่อฟอนต์ PostScript (เช่น Times-Roman, Times-BoldItalic, Helvetica, Courier)
- ขนาดพอยท์ (1 พอยท์ = 1/72 นิ้ว) ฟอนต์ถูกระบุในรูปแบบนี้: Courier-Bold8 is 8-point Courier Bold;

Helvetica12 คือ 12-พอยท์ Helvetica

-FO Codeset:Font

ตั้งค่าชื่อโค๊ดชุดอักขระ ซึ่งถูกเขียนลงในไฟล์ PostScript และฟอนต์ SBCS เพื่อใช้สำหรับส่วนหัวของแต่ละหน้า ดีฟอลต์ ถูกกำหนดโดยไฟล์คอนฟิกูเรชัน /usr/lib/ps/transcript.conf สำหรับแต่ละโลแคล

-F1 Codeset:Font

ตั้งค่าชื่อโค๊ดชุดอักขระ ซึ่งถูกเขียนลงในไฟล์ PostScript และฟอนต์ MBCS เพื่อใช้สำหรับส่วนหัวของแต่ละหน้า ดีฟอลต์ ถูกกำหนดโดยไฟล์คอนฟิกูเรชัน /usr/lib/ps/transcript.conf สำหรับแต่ละโลแคล

-LLines

ตั้งจำนวนบรรทัดสูงสุดที่จะพิมพ์บนหน้า คำสั่ง `enscript` โดยปกติคำนวณจำนวนบรรทัดที่จะใส่ไว้บนหน้าจากขนาดพอยท์ (ซึ่งอาจใส่จำนวนบรรทัดต่อหน้า น้อยกว่าที่ร้องขอโดยแฟล็ก `-L`).

-MMediaName

ระบุชื่อของสื่อบันทึกที่ใช้เพื่อกำหนดจำนวนของพื้นที่ที่สามารถทำเป็นอิมเมจได้ บนกระดาษ ชื่อที่จัดเตรียมไว้ตรงกับรายการในไฟล์ `MediaSizes` ตัวอย่างเช่น `-M legal` จะร้องขอขนาดของกระดาษที่ต้องการซึ่งเป็นพื้นที่ที่สามารถทำเป็นอิมเมจได้ ถ้าแฟล็ก `n` นี้ไม่ถูกใช้ ขนาดดีฟอลต์จะเป็นขนาดตัวอักษร ซึ่งคือกว้าง 8.5 นิ้ว คูณลึก 11.0 นิ้ว (กว้าง 21.6 เซนติเมตร คูณด้วยลึก 27.9 เซนติเมตร)

-XCodesetName

ระบุชุดโค๊ดสำหรับข้อมูลอินพุต โดยดีฟอลต์ อินพุตชุดโค๊ด ถูกกำหนดโดยรูทีนย่อย `nl_langinfo` ถ้าแฟล็กนี้ถูกใช้ ชุดโค๊ดถูกกำหนดโดย `CodesetName`

## ส่วนสับสหนุอักขระสากล

อักขระทั้งหมดที่ไม่พบในฟอนต์จะถูกแทนที่ด้วยอักขระ? (เครื่องหมายคำถาม) สำหรับรายการสมบูรณ์ของอักขระที่ไม่พบให้ใช้แฟล็ก `-o` ไฟล์ `NLSvec` มีข้อมูลเกี่ยวกับการแปลงอักขระ

## ตัวแปรสภาวะแวดล้อม

### ไอเท็ม

ENSCRIPT

### คำอธิบาย

ระบุสตริงของอ็อพชันที่จะถูกใช้โดยคำสั่ง `enscript`

LPDEST

ระบุปลายทางพรินเตอร์ อ็อพชันสพูลเลอร์ `-d` เขียนทับตัวแปรสภาวะแวดล้อมนี้

PSLIBDIR

มีชื่อพารของไดเรกทอรีที่จะใช้แทนไดเรกทอรี `/usr/lib/ps` สำหรับคำสั่ง `enscript` ส่วนนำและไฟล์สัดส่วนฟอนต์

PSTEMPDIR

มีชื่อพารของไดเรกทอรีชั่วคราวที่จะใช้แทนไดเรกทอรี `/var/tmp` ของไฟล์ชั่วคราวที่สพูล

TRANSCRIPT

มีชื่อพารสมบูรณ์ของไฟล์ที่จะใช้ แทนไฟล์คอนฟิกูเรชัน `/usr/lib/ps/transcript.conf` สำหรับการจัดการ MBCS

## ไฟล์

ไอเท็ม  
/usr/lib/ps/\*.afm  
/usr/lib/ps/font.map  
/usr/lib/ps/enscript.pro  
/usr/lib/ps/MediaSizes

คำอธิบาย  
มีไฟล์ Adobe Font Metrics (AFM)  
มีรายชื่อฟอนต์ที่มีตัวอักษรย่อ  
มีส่วนนำสำหรับไฟล์คำสั่ง **enscript**  
มีไฟล์ดีฟอลต์ที่ใช้สำหรับขนาดของสื่อบันทึก

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **managefonts**  
คำสั่ง **pic**  
คำสั่ง **refer**  
คำสั่ง **troff**

---

## คำสั่ง **entstat**

### Purpose

แสดงไดรเวอร์อุปกรณ์อีเทอร์เน็ตและสถิติอุปกรณ์

### Syntax

```
entstat [-d -r -t] Device_Name
```

### รายละเอียด

คำสั่ง **entstat** แสดงสถิติที่รวบรวม โดยไดรเวอร์อุปกรณ์อีเทอร์เน็ตที่ระบุ ผู้ใช้สามารถระบุเป็นทางเลือกว่า สถิติเฉพาะอุปกรณ์ถูกแสดงเพิ่มเติมกับสถิติทั่วไปของอุปกรณ์ ถ้าไม่มีการระบุแฟล็ก, เฉพาะสถิติทั่วไปของอุปกรณ์ เท่านั้นที่ถูกแสดง

คำสั่งนี้ยังถูกใช้เมื่อคำสั่ง **netstat** ถูกรันพร้อมกับแฟล็ก **-v** คำสั่ง **netstat** ไม่เรียกแฟล็กคำสั่ง **entstat**

ถ้ามีการระบุ *Device\_Name* ที่ไม่ถูกต้อง คำสั่ง **entstat** สร้างข้อความแสดงความผิดพลาด เริ่มต้นว่าไม่สามารถเชื่อมต่อไปที่อุปกรณ์

### แฟล็ก

ไอเท็ม คำอธิบาย  
**-d** แสดงสถิติทั้งหมด รวมถึงสถิติเฉพาะอุปกรณ์  
**-r** รีเซ็ตสถิติกลับเป็นค่าตั้งต้น แฟล็กนี้ สามารถถูกเรียกโดยผู้ใช้ที่ได้รับ privilege  
**-t** สลับการติดตามการตีบในบางไดรเวอร์อุปกรณ์

### พารามิเตอร์

ไอเท็ม  
Device\_Name

คำอธิบาย  
ชื่อของอุปกรณ์อีเทอร์เน็ต ตัวอย่าง ent0

## ฟิลด์สถิติ

หมายเหตุ: บางอะแด็ปเตอร์อาจไม่สนับสนุน สถิติ ค่าฟิลด์สถิติที่ไม่สนับสนุนเป็น 0 เสมอ

ฟิลด์สถิติที่แสดงในเอาต์พุตของคำสั่ง `entstat` และรายละเอียดคือ:

### ฟิลด์หัวเรื่อง

#### ไอเท็ม

ชนิดของอุปกรณ์  
ฮาร์ดแวร์แอดเดรส  
เวลาที่ผ่านไป

#### คำอธิบาย

แสดงรายละเอียดของชนิดอะแด็ปเตอร์  
แสดงอีเทอร์เน็ตเน็ตเวิร์กแอดเดรสที่ใช้โดยอุปกรณ์ในปัจจุบัน  
แสดงระยะเวลาจริง ที่ผ่านไปตั้งแต่ครั้งสุดท้ายที่ สถิติถูกรีเซ็ต ส่วนของสถิติอาจถูกรีเซ็ตโดยไดรเวอร์อุปกรณ์  
ระหว่างการแก้ไขข้อผิดพลาดเมื่อข้อผิดพลาดของฮาร์ดแวร์ถูกตรวจพบ จะมี Elapsed Time อื่นถูกแสดงในส่วน  
กลางของเอาต์พุตเมื่อสถานการณ์นี้เกิดขึ้น เพื่อที่จะแสดงเวลาที่ต่างกันระหว่างสถิติ

### ฟิลด์สถิติการส่ง

#### ไอเท็ม

Packets  
Bytes  
อินเตอร์รัปต์  
ข้อผิดพลาดการส่งข้อมูล

แพ็กเก็ตที่ถูกครีอป

แพ็กเก็ตสูงสุดสำหรับคิวการส่งผ่านซอฟต์แวร์  
คิวโอเวอร์โฟลว์การส่งผ่านซอฟต์แวร์  
Current S/W+H/W Transmit Queue Length  
แพ็กเก็ตการรอคอยคาสต์  
มัลติคาสต์แพ็กเก็ต  
ไม่มี Carrier Sense  
DMA Underrun  
ข้อผิดพลาดการสูญเสีย CTS

ข้อผิดพลาดการชนกันของข้อมูลสูงสุด

ข้อผิดพลาดการชนกันข้อมูลแบบ Late  
Deferred

SQE Test

ข้อผิดพลาดหมดเวลาใช้งาน  
จำนวนการชนกันของข้อมูลหนึ่งครั้ง

จำนวนการชนกันของข้อมูลหลายครั้ง

Current HW Transmit Queue Length  
ข้อผิดพลาด CRC

#### คำอธิบาย

จำนวนของแพ็กเก็ตที่ส่งสำเร็จโดยอุปกรณ์  
จำนวนไบต์ที่ส่งสำเร็จโดยอุปกรณ์  
จำนวนของอินเตอร์รัปต์การส่งที่ได้รับโดยไดรเวอร์จาก อะแด็ปเตอร์  
จำนวนของข้อผิดพลาดเอาต์พุตที่พบบนอุปกรณ์นี้ เป็น ตัวนับการส่งข้อมูลไม่สำเร็จ เนื่องจาก  
จากข้อผิดพลาดทาง ฮาร์ดแวร์/เน็ตเวิร์ก  
จำนวนของแพ็กเก็ตที่ยอมรับโดยไดรเวอร์อุปกรณ์สำหรับการส่งข้อมูลซึ่งไม่ได้ให้กับ  
อุปกรณ์ (สำหรับเหตุผลใดๆ)  
จำนวนสูงสุดของแพ็กเก็ตขาออกที่คิวกับคิวการส่ง ซอฟต์แวร์  
จำนวนสูงสุดของแพ็กเก็ตขาออกที่เคย overflow คิวการส่งของซอฟต์แวร์  
จำนวนของแพ็กเก็ตขาออกขาออก บนคิวการส่งข้อมูล หรือคิวการส่งข้อมูลซอฟต์แวร์  
จำนวนของแพ็กเก็ตกระจายที่ส่งโดยไม่มีข้อผิดพลาด  
จำนวนของมัลติคาสต์แพ็กเก็ตที่ส่งโดยไม่มีข้อผิดพลาด  
จำนวนของการส่งข้อมูลที่ไม่สำเร็จเนื่องจากข้อผิดพลาด ไม่มี carrier sense  
จำนวนของการส่งข้อมูลไม่สำเร็จเนื่องจากข้อผิดพลาด DMA underrun  
จำนวนของการส่งข้อมูลที่ไม่สำเร็จเนื่องจากข้อผิดพลาดการสูญเสียสัญญาณ Clear-to-Send  
จำนวนของการส่งข้อมูลที่ไม่สำเร็จเนื่องจาก มีการชนกันของข้อมูลมากเกินไป จำนวนของ  
การชนกันของข้อมูล ที่พบมีมากเกินไปจำนวนการพยายามข้าม อะแด็ปเตอร์  
จำนวนของการส่งข้อมูลไม่สำเร็จเนื่องจากข้อผิดพลาดการชนของข้อมูลแบบ late  
จำนวนของแพ็กเก็ตขาออกที่เลื่อนเวลาไประหว่างการส่งข้อมูล การเลื่อนเวลา หมายถึง  
ถึงอะแด็ปเตอร์จำเป็นต้องรอเวลาขณะพยายามส่งเฟรม สภาวะ นี้เกิดขึ้นถ้าเน็ตเวิร์กไม่ว่าง  
เมื่ออะแด็ปเตอร์พร้อมที่จะส่งข้อมูล อะแด็ปเตอร์จะเลื่อนเวลาเฉพาะในครั้งแรกที่พยายาม  
ส่งแพ็กเก็ต หลังจากนั้น อะแด็ปเตอร์จะส่งแพ็กเก็ตโดยไม่มีกรตรวจสอบ ถ้าเน็ตเวิร์กยังคง  
ไม่ว่าง การชนกันของข้อมูลจะถูกบันทึกไว้  
มีจำนวนของการทดสอบ "Signal Quality Error" (หรือ Heartbeat) ที่ดำเนินการสำเร็จ  
ระหว่างการส่งข้อมูล  
จำนวนของการส่งข้อมูลที่ไม่สำเร็จเนื่องจากอะแด็ปเตอร์รายงานข้อผิดพลาดการหมดเวลา  
จำนวนของแพ็กเก็ตขาออกที่พบการชนกันของข้อมูลหนึ่งครั้ง (เพียงครั้งเดียว) ระหว่างการ  
ส่งข้อมูล  
จำนวนของแพ็กเก็ตขาออกที่พบการชนกันของข้อมูลหลายครั้ง (2-15) ระหว่างการส่งข้อมูล  
จำนวนของแพ็กเก็ตขาออกซึ่งขณะนี้อยู่บนคิว การส่งข้อมูลฮาร์ดแวร์  
จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาด Checksum (FCS)

## ไอเอ็ม

DMA Overrun

ข้อผิดพลาดการจัดตำแหน่ง

ข้อผิดพลาดโมริซอร์ส

ข้อผิดพลาดการชนกัน เมื่อรับข้อมูล

ข้อผิดพลาดแพ็กเก็ตสั้นเกินไป

ข้อผิดพลาดแพ็กเก็ตยาวเกินไป

แพ็กเก็ตที่ถูกทิ้งโดยอะแดปเตอร์

จำนวนการเริ่มต้นตัวรับ

## คำอธิบาย

จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาด DMA overrun

จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาดการจัดตำแหน่ง

จำนวนของแพ็กเก็ตขาเข้าที่ลดลงโดยฮาร์ดแวร์เนื่องจากข้อผิดพลาดโมริซอร์ส ข้อผิดพลาด

นี้โดยปกติเกิดขึ้นเนื่องจากบัฟเฟอร์รับข้อมูลบนถูกใช้หมด บางอะแดปเตอร์จะมีขนาด

ของบัฟเฟอร์รับข้อมูล เป็นพารามิเตอร์ที่กำหนดค่าได้ ตรวจสอบแอดดริบิตคอนฟิกรูชัน

ของอุปกรณ์ (หรือวิธีใช้ smit) สำหรับข้อมูลการปรับแต่งที่เป็นไปได้

จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาดการชนกันของข้อมูลระหว่าง การรับข้อมูล

จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาดของความยาวระบุ ขนาดแพ็กเก็ตน้อยกว่า

ขนาดแพ็กเก็ตน้อยที่สุดของอีเทอร์เน็ต

จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาดของความยาวระบุ ขนาดแพ็กเก็ตใหญ่กว่า

ขนาดแพ็กเก็ตสูงสุดของอีเทอร์เน็ต

จำนวนของแพ็กเก็ตขาเข้าที่ลดลงโดยฮาร์ดแวร์ไม่ว่าด้วยเหตุใดก็ตาม

จำนวนครั้งที่ตัวรับ (หน่วยรับข้อมูล) บนอะแดปเตอร์ ได้ถูกสตาท

## ฟิลด์สถิติการรับ

### ไอเอ็ม

Packets

Bytes

อินเตอร์รัปต์

ข้อผิดพลาดของการรับ

แพ็กเก็ตที่ถูกครอบ

แพ็กเก็ตที่ไม่ดี

แพ็กเก็ตการบรอดคาสต์

มัลติคาสต์แพ็กเก็ต

ข้อผิดพลาด CRC

DMA Overrun

ข้อผิดพลาดการจัดตำแหน่ง

ข้อผิดพลาดโมริซอร์ส

ข้อผิดพลาดการชนกัน เมื่อรับข้อมูล

ข้อผิดพลาดแพ็กเก็ตสั้นเกินไป

ข้อผิดพลาดแพ็กเก็ตยาวเกินไป

แพ็กเก็ตที่ถูกทิ้งโดยอะแดปเตอร์

จำนวนการเริ่มต้นตัวรับ

### คำอธิบาย

จำนวนแพ็กเก็ตที่รับเสร็จโดยอุปกรณ์

จำนวนไบต์ที่รับเสร็จโดยอุปกรณ์

จำนวนของอินเตอร์รัปต์การรับที่ได้รับโดยไดรเวอร์จาก อะแดปเตอร์

จำนวนของข้อผิดพลาดอินพุตที่พบบนอุปกรณ์นี้ เป็น ตัวนับการรับข้อมูลไม่สำเร็จ

เนื่องจากข้อผิดพลาดทาง ฮาร์ดแวร์/เน็ตเวิร์ก

จำนวนแพ็กเก็ตที่ได้รับโดยไดรเวอร์อุปกรณ์จากอุปกรณ์นี้ ซึ่งไม่ถูกกำหนด (ไม่ว่า

ด้วยเหตุใด) ให้กับ demuxer

จำนวนของแพ็กเก็ตไม่ถูกต้องที่ได้รับ (หรือบันทึก) โดยไดรเวอร์อุปกรณ์

จำนวนของแพ็กเก็ตกระจายที่รับโดยไม่มีข้อผิดพลาด

จำนวนของมัลติคาสต์แพ็กเก็ตที่รับโดยไม่มีข้อผิดพลาด

จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาด Checksum (FCS)

จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาด DMA overrun

จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาดการจัดตำแหน่ง

จำนวนของแพ็กเก็ตขาเข้าที่ลดลงโดยฮาร์ดแวร์เนื่องจากข้อผิดพลาดโมริซอร์ส

จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาดการชนกันของข้อมูลระหว่าง การรับข้อมูล

จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาดของความยาวระบุ ขนาดแพ็กเก็ตน้อย

กว่าขนาดแพ็กเก็ตน้อยที่สุดของอีเทอร์เน็ต

จำนวนของแพ็กเก็ตขาเข้าที่มีข้อผิดพลาดของความยาวระบุ ขนาดแพ็กเก็ตใหญ่

กว่าขนาดแพ็กเก็ตสูงสุดของอีเทอร์เน็ต

จำนวนของแพ็กเก็ตขาเข้าที่ลดลงโดยฮาร์ดแวร์ไม่ว่าด้วยเหตุใดก็ตาม

จำนวนครั้งที่ตัวรับ (หน่วยรับข้อมูล) บนอะแดปเตอร์ ได้ถูกสตาท

## ฟิลด์สถิติทั่วไป

### ไอเอ็ม

No mbuf Errors

จำนวนรีเซตของอะแดปเตอร์

Adapter Data Rate

Driver Flags

### คำอธิบาย

จำนวนครั้งที่ mbufs ไม่พร้อมใช้งานกับไดรเวอร์อุปกรณ์ ซึ่งปกติเกิดขึ้นระหว่างการรับข้อมูล เมื่อไดรเวอร์

ต้องรับ บัฟเฟอร์ mbuf เพื่อประมวลผลแพ็กเก็ตขาเข้า ถ้า mbuf สำหรับขนาดที่ร้องขอ วางเปล่า แพ็กเก็ต

จะถูกทิ้งทั้ง คำสั่ง netstat -m สามารถถูกใช้เพื่อยืนยันสิ่งนี้

จำนวนครั้งที่อะแดปเตอร์ถูกรีเซ็ต (กำหนดค่าเริ่มต้นใหม่)

อัตราข้อมูลสูงสุดของอะแดปเตอร์ในหน่วย Mbps (เมกะบิตต่อ วินาที)

แฟล็กสถานะภายในของไดรเวอร์อุปกรณ์ที่เปิดอยู่ขณะนี้

## ฟิลด์สถิติจำเพาะอุปกรณ์

ส่วนนี้ของการแสดงผลอาจต่างไปสำหรับแต่ละ ชนิดของอะแดปเตอร์ โดยอาจมีข้อมูลจำเพาะอะแดปเตอร์และสถิติเพิ่มเติม

บางส่วนของไม่รวมอยู่ในสถิติทั่วไป บาง อะแดปเตอร์อาจไม่มีสถิติจำเพาะอุปกรณ์

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

### ตัวอย่าง

1. เมื่อต้องการแสดงสถิติทั่วไปของอุปกรณ์สำหรับ **ent0**, ให้ป้อน:

```
entstat ent0
```

ซึ่งจะสร้างเอาต์พุตต่อไปนี้:

```
ETHERNET STATISTICS (ent0) :
Device Type: Ethernet High Performance LAN Adapter
Hardware Address: 02:60:8c:2e:d0:1d
Elapsed Time: 0 days 0 hours 8 minutes 41 seconds

Transmit Statistics:      Receive Statistics:
-----
Packets: 3                Packets: 2
Bytes: 272                Bytes: 146
Interrupts: 3            Interrupts: 2
Transmit Errors: 0        Receive Errors: 0
Packets Dropped: 0        Packets Dropped: 0
Max Packets on S/W        Bad Packets: 0
Transmit Queue:0
S/W Transmit Queue
Overflow: 0
Current S/W+H/W Transmit
Queue Length: 0

Broadcast Packets: 2      CRC Errors: 0
Multicast Packets: 0      Broadcast Packets: 1
No Carrier Sense: 0      Multicast Packets: 0
DMA Underrun: 0          DMA Overrun: 0
Lost CTS Errors: 0        Alignment Errors: 0
Max Collision Errors: 0   No Resource Errors: 0
Late Collision Errors: 0  Receive Collision Errors: 0
Deferred: 0               Packet Too Short Errors: 0
SQE Test: 0               Packet Too Long Errors: 0
Timeout Errors: 0         Packets Discarded by Adapter: 0
Single Collision          Receiver Start Count: 1
Count: 0
Multiple Collision Count: 0
Current HW Transmit Queue
Length: 0

General Statistics:
-----
```

No mbuf Errors: 0  
Adapter Reset Count: 0  
Adapter Data Rate: 2000  
Driver Flags: Up Broadcast Running Simplex

2. เมื่อต้องการแสดงสถิติทั่วไปของอุปกรณ์ Ethernet และสถิติอุปกรณ์อีเทอร์เน็ตเฉพาะสำหรับ ent0 ให้ป้อน:

entstat -d ent0

ซึ่งจะสร้างเอาต์พุตต่อไปนี้:

ETHERNET STATISTICS (ent0) :  
Device Type: Ethernet High Performance LAN Adapter  
Hardware Address: 02:60:8c:2e:d0:1d  
Elapsed Time: 0 days 2 hours 6 minutes 30 seconds

Transmit Statistics:	Receive Statistics:
-----	-----
Packets: 3	Packets: 2
Bytes: 272	Bytes: 146
Interrupts: 3	Interrupts: 2
Transmit Errors: 0	Receive Errors: 0
Packets Dropped: 0	Packets Dropped: 0
Max Packets on S/W	Receiver Start Count: 1
Transmit Queue:0	
Bad Packets: 0	
S/W Transmit Queue Overflow: 0	
Current S/W+H/W Transmit Queue Length: 0	
Broadcast Packets: 0	Broadcast Packets: 0
Multicast Packets: 0	Multicast Packets: 0
No Carrier Sense: 0	CRC Errors: 0
DMA Underrun: 0	DMA Overrun: 0
Lost CTS Errors: 0	Alignment Errors: 0
Max Collision Errors: 0	No Resource Errors: 0
Late Collision Errors: 0	Receive Collision Errors: 0
Deferred: 0	Packet Too Short Errors: 0
SQE Test: 0	Packet Too Long Errors: 0
Timeout Errors: 0	Packets Discarded by Adapter: 0
Single Collision Count: 0	Receiver Start Count: 1
Multiple Collision Count: 0	
Current HW Transmit Queue Length: 0	

General Statistics:

-----  
No mbuf Errors: 0  
Adapter Reset Count: 0  
Adapter Data Rate: 2000  
Driver Flags: Up Broadcast Running Simplex

Ethernet High Performance LAN Adapter Specific Statistics:

-----  
Receive Buffer Pool Size: 37  
Transmit Buffer Pool Size: 39  
In Promiscuous Mode for IP Multicast: No  
Packets Uploaded from Adapter: 0

Host End-of-List Encountered: 0  
82586 End-of-List Encountered: 0  
Receive DMA Timeouts: 0  
Adapter Internal Data: 0x0 0x0 0x0 0x0 0x0

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fddistat” ในหน้า 531

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง atmstat

คำสั่ง netstat

คำสั่ง tokstat

---

## คำสั่ง env

### วัตถุประสงค์

การแสดงผลภาวะแวดล้อมปัจจุบันหรือเซตสถานะแวดล้อม สำหรับการดำเนินการคำสั่ง

### ไวยากรณ์

เมื่อต้องการแสดงตัวแปรสถานะแวดล้อมหลายตัวแปร

```
env [ -i|- ] [Name=Value ]... [Command [ Argument... ]]
```

เมื่อต้องการแสดงตัวแปรสถานะแวดล้อมเดียว

```
env [Name]
```

### รายละเอียด

คำสั่ง env ช่วยให้ท่าน แสดงสถานะแวดล้อมปัจจุบันของคุณหรือรันคำสั่งที่ระบุในสถานะแวดล้อมที่เปลี่ยนแปลง

ถ้าไม่มีการระบุแฟล็กหรือพารามิเตอร์, คำสั่ง env แสดงสถานะแวดล้อมปัจจุบันของคุณ แสดงหนึ่งคู่ *Name=Value* ต่อบรรทัด

### แฟล็ก

ไอเท็ม คำอธิบาย

-i      ละเว้นสถานะแวดล้อมที่สืบทอดและเรียกคำสั่งที่ระบุโดยพารามิเตอร์ *Command* ที่มีสถานะแวดล้อมที่ระบุโดยพารามิเตอร์ *Name=Value*

### พารามิเตอร์

ไอเท็ม	คำอธิบาย
<code>Name=Value</code>	คุณสามารถรันคำสั่งในเวอร์ชันดัดแปลงของสถานะแวดล้อมปัจจุบัน โดยการระบุพารามิเตอร์ <code>Name=Value</code> หนึ่งพารามิเตอร์ หรือมากกว่านั้น ใช้แฟล็ก <code>-i</code> หากคุณต้องการแทนที่สถานะแวดล้อมปัจจุบันทั้งหมด ด้วยพารามิเตอร์ <code>Name=Value</code> ที่ระบุในทั้งสองกรณี การเปลี่ยนแปลงสถานะแวดล้อมมีเฉพาะในขณะที่ยังรันอยู่
คำสั่ง	พารามิเตอร์ <code>Command</code> มีตัวแปร <code>Argument</code> ทางเลือก ถ้าคำสั่งที่ระบุเป็นหนึ่งใน คำสั่งในตัวพิเศษของเชลล์ Korn, ผลลัพธ์จะระบุไม่ได้ คำสั่งในตัวเชลล์ Korn ถูกอธิบายในคำสั่ง <code>ksh</code>

## สถานะออก

ถ้าพารามิเตอร์ `Command` ถูกระบุ สถานะ `exit` ของคำสั่ง `env` คือสถานะ `exit` ของคำสั่งที่ระบุในพารามิเตอร์ `Command` มิฉะนั้น คำสั่ง `env` จบการทำงานด้วยหนึ่งในค่า ต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	คำสั่ง <code>env</code> สำเร็จสมบูรณ์
1-125	มีข้อผิดพลาดเกิดขึ้นในคำสั่ง <code>env</code>
126	พบคำสั่งที่ระบุโดยพารามิเตอร์ <code>Command</code> แต่ไม่สามารถเรียกใช้ได้
127	ไม่พบคำสั่งที่ระบุโดยพารามิเตอร์ <code>Command</code>

## ตัวอย่าง

- เมื่อต้องการเปลี่ยนตัวแปรสถานะแวดล้อม TZ ขณะรันคำสั่ง `date` ให้พิมพ์:

```
TZ=MST7MDT date
```

OR

```
env TZ=MST7MDT date
```

แต่ละคำสั่งแสดงเวลาใน mountain time และวันที่ ปัจจุบัน สองคำสั่งที่แสดงเหมือนกัน เมื่อคำสั่ง `date` เสร็จสิ้น ค่าก่อนหน้าของตัวแปรสถานะแวดล้อม TZ มีผลอีกครั้ง

- เมื่อต้องการรันคำสั่ง `make` ในสถานะแวดล้อมที่มีเฉพาะนิยามสำหรับตัวแปรสถานะแวดล้อม `PATH`, `IDIR`, และ `LIBDIR` ให้พิมพ์:

```
env -i PATH=$PATH IDIR=/HOME/include LIBDIR=/HOME/lib make
```

คุณต้องระบุตัวแปรสถานะแวดล้อม `PATH` เพื่อให้เชลล์สามารถค้นพบคำสั่ง `make` เมื่อคำสั่ง `make` เสร็จสิ้น สถานะแวดล้อมก่อนหน้านี้ มีผล

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/env</code>	มีคำสั่ง <code>env</code>

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `printenv`

คำสั่งด้านสถานะแวดล้อม

รูปแบบไฟล์โปรไฟล์

ภาพรวมโปรไฟล์

---

## คำสั่ง epkg

### วัตถุประสงค์

สร้างแพ็คเกจ interim fix ที่สามารถถูกติดตั้งโดยตัวจัดการ interim fix, **emgr**

### ไวยากรณ์

```
epkg [ -w WorkDirectory ] [ -a APARrefFile ] [ -p PrerequisiteFile ] [ -d DescriptionFile ] [ -e interimfixControlFile ] [ -g PrerequisiteFile ] [ -l LockFile ] [ -S SupersedeFile ] [ -u {yln} ] [ -r {ylnlo} ] [ -s ] [ -T {yln} ] [ -X ] [ -v ] interimfixLabel
```

### คำอธิบาย

ทูล **epkg** สามารถถูกรันในสองโหมดต่างกัน: *interactive* และ *template-based* โหมด *interactive* พร้อมท์คุณด้วยคำถามหลายคำถามและสร้าง แพ็คเกจ interim fix จากคำตอบเหล่านั้น โหมด *template-based* ใช้ไฟล์ควบคุม interim fix ที่ถูกจัดเตรียมด้วย คำตอบของคำถามที่ถูกถามในโหมด *interactive* แพ็คเกจ interim fix ถูกติดตั้งโดยตัวจัดการ interim fix, ซึ่งถูกเริ่มต้นด้วย คำสั่ง **emgr**

### โหมดการโต้ตอบ

คำสั่ง **epkg** รันให้โหมดการโต้ตอบ โดยดีฟอลต์ เฉพาะพารามิเตอร์ที่ต้องการเท่านั้นคือเลเบล interim fix ถ้าคุณอินเตอร์รัปต์เซสชัน **epkg** ไฟล์ควบคุม interim fix จะถูกบันทึก ถ้าคุณสตาร์ทเซสชันใหม่ด้วยเลเบล interim fix เดียวกัน คุณจะถูกถามว่าคุณต้องการทำงานต่อกับ ไฟล์ควบคุม interim fix ก่อนหน้านี้หรือไม่ เมื่อต้องการจัดเตรียมข้อมูลนี้ ก่อนที่คุณจะเริ่มเซสชัน **epkg** แบบโต้ตอบ, รัน **epkg** ด้วยแฟล็ก **-u**

คำสั่ง **epkg** ดูแลเร็กคอร์ดของ ลำดับคำถามและอนุญาตให้คุณนำทางระหว่างคำถามโดยใช้ คำสั่งย่อ นอกจากนี้ คำสั่ง **epkg** จำคำตอบ ก่อนหน้านี้ ที่คุณตอบและเซตคำตอบนั้นเป็น คำตอบดีฟอลต์ คำสั่งย่อ **epkg** ถูกอธิบายไว้ใน ส่วน คำสั่งย่อ

หลังจากคุณตอบคำถามทั้งหมด, คำสั่ง **epkg** ตรวจสอบไฟล์ควบคุม interim fix และสร้างแพ็คเกจ tar ที่บีบอัดที่สามารถถูกติดตั้งด้วยคำสั่ง **emgr**

### การใช้ เพิ่มเพลตไฟล์ควบคุม

คุณสามารถสร้างแพ็คเกจ interim fix แบบไม่โต้ตอบโดย การใช้ไฟล์ควบคุม interim fix เป็นเพิ่มเพลต ต่อไปนี้เป็น ตัวอย่างของไฟล์ควบคุม interim fix ที่สมบูรณ์:

```
# interim fix control file complete example
ABSTRACT=This is a test of epkg.
PRE_INSTALL=/tmp/pre_install
POST_INSTALL=.
PRE_REMOVE=/tmp/pre_remove
POST_REMOVE=.
REBOOT=yes
PREREQ=.
DESCRIPTION=/tmp/description
EFIX_FILES=2
APARREF=/tmp/aparref
LKU_CAPABLE=no
```

**EFIX\_FILE:**

```

EFIX_FILE_NUM=1
SHIP_FILE=/home/test/ls
TARGET_FILE=/usr/bin/ls
TYPE= 1
INSTALLER= 1
ACL= DEFAULT
AR_MEM=.

```

**EFIX\_FILE:**

```

EFIX_FILE_NUM=2
SHIP_FILE=/home/test/mystrcat.o
TARGET_FILE=/usr/ccs/lib/libc.a
TYPE= 2
INSTALLER= 1
ACL= root:system:555
AR_MEM=strcat.o

```

ค่าไฟล์ควบคุม interim fix มีดังนี้:

**ABSTRACT**

อธิบายแพ็คเกจ interim fix อย่างย่อ บทความย่อ ถูกจำกัดที่ 38 ไบต์

**PRE\_INSTALL**

ระบุตำแหน่งของสคริปต์ที่ถูกรันหลังจากการแสดงผลตัวอย่าง การติดตั้งและก่อนที่ไฟล์ interim fix ถูกติดตั้ง ความล้มเหลว ในสคริปต์ PRE\_INSTALL จะทำให้การติดตั้งแพ็คเกจ interim fix ถูกยกเลิก คอมโพเนนต์นี้เป็นทางเลือก

**POST\_INSTALL**

ระบุตำแหน่งของสคริปต์ที่ถูกรันหลังจากไฟล์ interim fix ทั้งหมด ถูกติดตั้งสำเร็จ คอมโพเนนต์นี้เป็นทางเลือก

**PRE\_REMOVE**

ระบุตำแหน่งของสคริปต์ที่ถูกรันหลังจากการแสดงผลตัวอย่าง การลบและก่อนที่ไฟล์ interim fix จะถูกลบออกระหว่างการดำเนินการลบ คอมโพเนนต์นี้เป็นทางเลือก

**POST\_REMOVE**

ระบุตำแหน่งของสคริปต์ที่ถูกรันหลังจากไฟล์ interim fix ถูกลบออกระหว่างการดำเนินการลบ คอมโพเนนต์นี้เป็นทางเลือก

**REBOOT**

ระบุการรีบูตจำเป็นสำหรับ interim fix นี้หรือไม่ ค่าที่ใช้ได้คือ yes หรือ no ถ้าค่านี้ถูกเซตเป็น yes คำสั่ง the **emgr** จะทำการเปลี่ยนแปลงตามความจำเป็นกับ บูตอิมเมจ และส่งข้อความแนะนำแก่ผู้ใช้ให้รีบูตหลังจาก การติดตั้ง

**PREREQ**

ระบุตำแหน่งของไฟล์ที่มีสิ่งจำเป็นของ **installp** คอมโพเนนต์นี้เป็นทางเลือก

**APARREF**

ระบุตำแหน่งของไฟล์ที่มีหมายเลขการอ้างอิง APAR และ บทความย่อซึ่งสัมพันธ์กับโปรแกรมฟิกซ์เฉพาะกิจนี้ แต่ละบรรทัดของไฟล์มีหมายเลขการอ้างอิง APAR, หมายเลข APAR และบทความย่อ APAR รูปแบบของไฟล์เป็นดังนี้:

```
APAR reference|:|APAR number|:|APAR abstract
```

ไม่ใช่ว่าทุกฟิลด์จำเป็นในการทำให้ไฟล์ APARREF ถูกต้อง ถ้าฟิลด์เฉพาะไม่รู้จัก หรือไม่จำเป็น ให้ระบุ NONE หรือปล่อยฟิลด์นั้นว่างไว้ บางตัวอย่างของไฟล์ APARREF ที่ถูกต้องมีดังนี้:

#### ตัวอย่าง 1

```
123456|:|IV12345|:|This is the APAR abstract  
789012|:|IV67890|:|This is another APAR abstract
```

#### ตัวอย่าง 2

```
123456|:|NONE|:|NONE  
789012|:|NONE  
345678
```

#### ตัวอย่าง 3

```
NONE|:|IV12345|:|This is the APAR abstract
```

#### ตัวอย่าง 4

```
NONE
```

ถ้าคุณระบุไฟล์การอ้างอิง APAR พร้อมกับหมายเลขอ้างอิง APAR คุณลักษณะการลบ อัตโนมัตโดยคำสั่ง **installp** จะเปิดใช้งานสำหรับโปรแกรมฟิกซ์เฉพาะกิจ การลบ อัตโนมัตโดยคำสั่ง **installp** หมายถึงความสามารถในการลบ โปรแกรมฟิกซ์เฉพาะกิจ โดยอัตโนมัติ ถ้าโปรแกรมฟิกซ์มีอยู่ในระดับเทคโนโลยี เซอร์วิสแพ็ก หรือ PTF ที่ใช้คำสั่ง **installp** ถ้า NONE แสดงอยู่ในฟิลด์การอ้างอิง APAR คุณลักษณะการลบอัตโนมัติจะไม่เปิดใช้งานสำหรับโปรแกรมฟิกซ์เฉพาะกิจ

#### คำอธิบาย

ระบุตำแหน่งของไฟล์ที่มีคำอธิบายละเอียด ของแพ็คเกจ interim fix ที่ถูกติดตั้ง

#### EFIX\_FILES

ระบุจำนวนไฟล์รวมใน interim fix

#### EFIX\_FILE\_NUM

ระบุจำนวนไฟล์ใน interim fix แต่ละไฟล์ใน interim fix ต้องมีหมายเลขเฉพาะ จาก 1 ถึง 400 คำสั่ง **epkg** สามารถสนับสนุนไฟล์สูงสุด 400 ไฟล์ต่อ interim fix

#### SHIP\_FILE

ระบุตำแหน่งของไฟล์ที่ **epkg** จะ เก็บถาวรลงในแพ็คเกจ interim fix คุณสามารถระบุพารามิเตอร์หรือพารามิเตอร์กับไฟล์นี้

#### TARGET\_FILE

ระบุตำแหน่งซึ่ง **SHIP\_FILE** จะ ถูกติดตั้ง ตำแหน่งนี้อยู่บนระบบซึ่งแพ็คเกจ interim fix จะถูกติดตั้ง คุณต้องระบุพารามิเตอร์กับไฟล์นี้ ถ้าไฟล์นี้เป็นส่วนหนึ่งของแพ็คเกจที่รีจิสเตอร์ เช่นแพ็คเกจ RPM Package Manager (RPM) หรือ **installp** คุณต้องระบุตำแหน่ง ที่ติดตาม

**TYPE** ระบุชนิดของไฟล์ที่ถูกติดตั้ง ตัวเลือกที่ถูกต้องมีดังต่อไปนี้ :

- 1 ไฟล์ (มาตรฐานหรือที่รันได้)
- 2 ไลบรารีหรือสมาชิกไฟล์เก็บถาวร

#### INSTALLER

ระบุชนิดของโปรแกรมติดตั้ง ถ้ามี ซึ่งจะติดตามแพ็คเกจ interim fix ตัวเลือกที่ถูกต้องมีดังต่อไปนี้ :

- 1 ที่ติดตามในขณะนี้โดย **installp**
- 2 ขณะนี้ถูกติดตามโดย **RPM**
- 3 ที่ติดตามในขณะนี้โดย **ISMP**
- 4 ที่ติดตามในขณะนี้โดยโปรแกรมติดตั้งอื่น
- 5 นี่เป็นไฟล์ใหม่ที่จะถูกติดตามโดย **installp**
- 6 ไฟล์ใหม่ที่จะถูกติดตามโดย **RPM**
- 7 ไฟล์ใหม่ที่จะถูกติดตามโดย **ISMP**
- 8 ไฟล์ใหม่ที่จะถูกติดตามโดยโปรแกรมติดตั้งอื่น
- 9 ไม่ถูกติดตามโดยโปรแกรมติดตั้ง

**ACL** ระบุแอตทริบิวต์การเข้าถึง (โหมดและความเป็นเจ้าของ) สำหรับ ไฟล์ ถ้าแอตทริบิวต์นี้ถูกเซตเป็น **DEFAULT**, คำสั่ง **emgr** ดูแลสิทธิปัจจุบันของ ไฟล์ที่จะถูกแทนที่ อย่างไรก็ตาม ถ้าไฟล์ปลายทางเป็นไฟล์ใหม่ หรือถ้าผู้ใช้ต้องการที่จะระบุสิทธิด้วยแฟล็ก **-v**, แอตทริบิวต์ **ACL** สามารถถูกป้อนด้วยไวยากรณ์ **Owner:Group:OctalModes**, เหมือนดังนี้:

ACL= root:system:555

#### **AR\_MEM**

ระบุชื่อของสมาชิกไฟล์เก็บถาวร อีพซันนี้ใช้ได้เฉพาะ ถ้า **TYPE=2** ในกรณีนี้ **SHIP\_FILE** คือ ตำแหน่งโลคัลของสมาชิกไฟล์เก็บถาวรที่กำลังถูกจัดส่ง **TARGET\_FILE** คือไฟล์เก็บถาวรเป้าหมาย และ **ACL** ใช้กับสมาชิกไฟล์เก็บถาวร สำหรับตัวอย่าง การตั้งค่าต่อไปนี้จะทำให้โลคัลไฟล์ **myshr.o** เป็นสมาชิก **shr.o** ในไฟล์เก็บถาวรปลายทาง **/usr/ccs/lib/libc.a**:

```
TYPE=2
SHIP_FILE=/home/myshr.o
TARGET_FILE=/usr/ccs/lib/libc.a
AR_MEM=shr.o
```

#### **BUILD\_BOOT\_IMAGE**

ระบุว่าบูตอิมเมจจำเป็นต้องถูกสร้างใหม่หรือไม่ ค่าที่ใช้ได้คือ **yes** หรือ **no** การรีบูตจำเป็นถ้าฟิลด์นี้ถูกเซตเป็น **yes** ถ้าฟิลด์นี้ถูกเซตเป็น **yes** และฟิลด์ **REBOOT** ถูก เซตเป็น **no**, **epkg** ส่งคืนข้อผิดพลาด

#### **E2E\_PREREQ**

ระบุตำแหน่งของไฟล์สิ่งที่จำเป็นต้องมี **interim fix** ในไฟล์ควบคุม **interim fix**

#### **PKGLOCKS**

ระบุตำแหน่งโลคัลไฟล์ของแพ็คเกจล็อกไฟล์ใน ล็อกไฟล์ **interim fix**

#### **SUPERSEDE**

ระบุตำแหน่งโลคัลไฟล์ของไฟล์แทนที่ใน ล็อกไฟล์ **interim fix**

#### **FIXTESTED**

ระบุว่า **interim fix** นี้ได้ถูกทดสอบหรือไม่ ค่าที่ใช้ได้คือ **yes** หรือ **no**

#### **LKU\_CAPABLE**

ระบุว่าโปรแกรมฟิกซ์เฉพาะกิจนี้เข้ากันได้กับการดำเนินการ **Live Update** หรือไม่ แอตทริบิวต์นี้สามารถมีค่าเป็น **yes** หรือ **no** ในทางทฤษฎีแล้ว โปรแกรมฟิกซ์เฉพาะกิจทั้งหมดต้องถูกทำเครื่องหมายเป็นมีความสามารถ **Live**

Update ความเข้ากันได้จำเป็นสำหรับการติดตั้งโปรแกรมพิเศษเฉพาะกิจเป็นกลุ่ม หากโปรแกรมพิเศษเฉพาะกิจไม่เหมาะสำหรับการดำเนินการ Live Update แอ็ตทริบิวต์ LKU\_CAPABLE จะถูกตั้งค่าเป็นค่า no อย่างไรก็ตาม โปรแกรมพิเศษเฉพาะกิจส่วนใหญ่มีการตั้งค่าแอ็ตทริบิวต์นี้เป็น yes

### การสนับสนุนสำหรับ Superseding

สามารถสามารถระบุไฟล์ที่มีชื่อเลเบล interim fix ที่จะถูกแทนที่เมื่อ epkg ถูกติดตั้ง ซึ่งจะทำให้คำสั่ง emgr ลบเลเบล interim fix ทั้งหมด ที่ถูกระบบในไฟล์นี้ (ถ้าถูกติดตั้ง) ก่อนการติดตั้ง แพ็กเกจ interim fix ความล้มเหลวในการลบ interim fix การแทนที่ที่ติดตั้ง จะยกเลิกการติดตั้งแพ็กเกจ interim fix จำนวนที่สนับสนุน สูงสุดของเลเบลแทนที่คือ 32 โปรแกรมแพจเกจสามารถระบุไฟล์แทนที่ด้วยคำสั่ง epkg ในวิธีต่อไปนี้:

- ระบุตำแหน่งไฟล์ด้วยแฟล็ก **-S supersede\_file** ตัวอย่างเช่น:  
epkg -S /tmp/superseded.epkg myefix
- คำสั่ง **epkg** จะพร้อมต์ขอไฟล์แทนที่ ถ้าแฟล็กอ็อปชันเพิ่มเติม (**-v**) ถูกใช้ในโหมดการโต้ตอบ ตัวอย่างเช่น:  
Enter the location for the supersede file or "." to skip.  
-> /tmp/superseded.epkg
- เช็ทแอ็ตทริบิวต์ **SUPERSEDE** เป็นตำแหน่งไฟล์โลคัลของ ไฟล์แทนที่ในไฟล์ควบคุม interim fix ตัวอย่างเช่น:  
SUPERSEDE=/tmp/superseded.epkg

รูปแบบของไฟล์แทนที่คือหนึ่งเลเบล interim fix ที่จะถูกแทนที่ต่อบรรทัด หมายเหตุเริ่มต้นด้วยเครื่องหมาย # และช่องว่างนำหน้าถูกละเว้น ตัวอย่างเช่น:

```
# Requisites for efix myefix3  
myefix1  
myefix2
```

### การสนับสนุนสำหรับ prereqs และ xreqs

ผู้จัดทำแพ็กเกจสามารถระบุไฟล์ที่มีชื่อเลเบล interim fix ของโปรแกรมพิเศษเฉพาะกิจ ซึ่งเป็นสิ่งที่จำเป็นต้องมีก่อน จะติดตั้งแพ็กเกจ interim fix ซึ่งจะทำให้คำสั่ง emgr ตรวจสอบว่าเลเบล interim fix ถูกติดตั้งหรือไม่ (PREREQ) ถ้าข้อมูลที่จำเป็นไม่ได้ถูกติดตั้ง คำสั่ง emgr จะยกเลิกการติดตั้งของแพ็กเกจ interim fix ผู้ใช้ยังสามารถระบุเลเบล XREQ interim fix คำนี้จะทำให้คำสั่ง emgr ไม่ติดตั้ง interim fix ถ้า xreq interim fix ที่ระบุชื่อถูกติดตั้ง

โปรแกรมแพจเกจสามารถระบุไฟล์ที่จำเป็นต้องมีของ interim fix ด้วยคำสั่ง epkg ในวิธีต่อไปนี้:

- ระบุตำแหน่งไฟล์ด้วยแฟล็ก **-g efix\_prereq\_file** ตัวอย่างเช่น:  
epkg -g /tmp/efixprereq.epkg myefix
- คำสั่ง **epkg** จะพร้อมต์ขอไฟล์ interim fix prereq ถ้าแฟล็กอ็อปชันเพิ่มเติม (**-v**) ถูกใช้ในโหมดการโต้ตอบ ตัวอย่างเช่น:  
Enter the location for the efix prerequisite file or "." to skip.  
-> /tmp/efixprereq.epkg
- เช็ทแอ็ตทริบิวต์ **E2E\_PREREQ** เป็นตำแหน่งไฟล์โลคัลของข้อมูลไฟล์ที่จำเป็นต้องมี interim fix ในไฟล์ควบคุม interim fix ตัวอย่างเช่น:  
E2E\_PREREQ=/tmp/efixprereq.epkg

รูปแบบของรายการไฟล์ที่จำเป็นต้องมี interim fix เป็นดังนี้:

```
EfixLabel RequisiteType: PREREQ/XREQ
```

ตัวอย่างเช่น:

```
oldefix1 PREREQ # Make sure oldefix1 is already installed
```

```
oldefix4 XREQ # Make sure oldefix4 is NOT installed
```

จำนวนสูงสุดของสิ่งที่จำเป็นต้องมีของ interim fix ที่สนับสนุน คือ 32

### การสนับสนุนการเปิดใช้การลบการแก้ไขระหว่างช่วงเวลาอัตโนมัติโดย **installp**

โปรแกรมแพ็คเกจสามารถระบุไฟล์อ้างอิง APAR ที่มีหมายเลขอ้างอิง APAR หมายเลขการอ้างอิง APAR จะอนุญาตให้ **installp** แม้พ interim fix กลับไปที่ APARs สำหรับ Technology Levels ทั้งหมดที่การแก้ไขถูกจัดส่ง ถ้า **installp** กำหนดว่า interim fix อยู่ใน Technology Level, Service Pack, หรือ PTF ถูกนำมาใช้, **installp** จะลบ interim fix โดยอัตโนมัติก่อนนำอ็อปเดตมาใช้

### เอาต์พุตและการจัดเรียง

แฟล็ก **emgr -d** แสดงเนื้อหาและการจัดเรียงของแพ็คเกจ interim fix อ็อปชัน **-d** จะทำงานกับอ็อปชัน **-v verbose** ระดับที่ใช้ได้ของ verbosity คือ 1-3

Verbosity ระดับ 1 (ดีฟอลต์) จะแสดง:

- LABEL
- EFIX FILES
- TARGET LOCATION

Verbosity ระดับ 2 จะแสดง:

- เอาต์พุตระดับ 1 ทั้งหมด
- ABSTRACT
- REBOOT
- PRE-REQUISITES
- PRE\_INSTALL
- POST\_INSTALL
- PRE\_REMOVE
- POST\_REMOVE
- FILE TYPE

Verbosity ระดับ 3 จะแสดง:

- เอาต์พุตระดับ 2 ทั้งหมด
- PACKAGING DATE
- VUID
- SIZE
- ACL

- CKSUM
- PACKAGE
- EFIX คำอธิบาย
- CONTENTS OF INSTALL SCRIPTS (if text files)

ตัวอย่างเช่น:

- To get level 1 verbosity output on interim fix package test.102403.epkg.Z, type:  
emgr -d test.102403.epkg.Z
- To get level 3 verbosity output on interim fix package test.102403.epkg.Z, type:  
emgr -v3 -d test.102403.epkg.Z

### การสนับสนุนการล็อกแพ็คเกจเพิ่มเติม

โปรแกรมแพ็คเกจสามารถระบุไฟล์ที่มีชื่อแพ็คเกจที่ควรถูกล็อกโดยคำสั่ง **emgr** เพิ่มเติมจากที่ถูกล็อกโดยอัตโนมัติจากความ เป็นเจ้าของไฟล์โปรแกรมแพ็คเกจ ต้องระบุชื่อของแพ็คเกจ โปรแกรมติดตั้ง และชนิดของ การดำเนินการล็อกแพ็คเกจ (ALWAYS/IFINST) โปรแกรมแพ็คเกจสามารถระบุแพ็คเกจล็อกไฟล์โดยใช้คำสั่ง **epkg** ใน วิธีต่อไปนี้:

- ระบุตำแหน่งไฟล์ด้วยแฟล็ก **-l *pkg\_locks\_file*** ตัวอย่างเช่น:  
epkg -l /tmp/pkglock.epkg myefix
- คำสั่ง **epkg** จะพร้อมต่อขอแพ็คเกจล็อกไฟล์ ถ้าแฟล็กอ็อปชันเพิ่มเติม (**-v**) ถูกใช้ ตัวอย่างเช่น:  
Enter the location for the package locks file or "." to skip.  
-> /tmp/pkglock.epkg
- เช็ทแอดทริบิวต์ **PKGLOCKS** เป็นตำแหน่งแพ็คเกจล็อกไฟล์ในไฟล์ควบคุม interim fix ตัวอย่างเช่น:  
PKGLOCKS=/tmp/pkglock.epkg

รูปแบบของแพ็คเกจล็อกไฟล์มีดังนี้:

*PackageName PackageAction PackageType*

โดยที่ *PackageName* คือชื่อของแพ็คเกจ ที่จะถูกล็อกและ *PackageAction* คือหนึ่งใน ค่าต่อไปนี้:

ไอเท็ม	คำอธิบาย
ALWAYS	พยายามล็อกแพ็คเกจนี้เสมอ ความล้มเหลวในการล็อกแพ็คเกจมีผลให้การติดตั้งล้มเหลว
IFINST	ความพยายามในการล็อกแพ็คเกจนี้เฉพาะถ้าแพ็คเกจ ถูกติดตั้ง ความล้มเหลวในการล็อกแพ็คเกจ <i>installed</i> มีผลให้การติดตั้งล้มเหลว

*PackageType* คือ installp (ดีฟอลต์), rpm, ISMP, other

**หมายเหตุ:** สนับสนุนเฉพาะการล็อก installp

จำนวนสูงสุดของรายการล็อกแพ็คเกจที่สนับสนุนคือ 32

ตัวอย่าง:

```
bos.rte.lvm ALWAYS installp
bos.games IFINST installp
```

ในตัวอย่างด้านบนคำสั่ง `emgr` จะพยายามล็อก `bos.rte.lvm` เสมอระหว่างการติดตั้ง และจะปลดล็อกเมื่อทำการลบ คำสั่ง `emgr` จะล็อก `bos.games` เฉพาะถ้ามีการติดตั้ง และจะปลดล็อกเมื่อทำการลบ (ถ้าล็อกอยู่)

### การสนับสนุนอ็อปชัน `bosboot`

อ็อปชันรีบูตของคำสั่ง `epkg` รวมการรีบูต โดยไม่มีการสร้างบูตอิมเมจใหม่

ผู้ใช้งานสามารถระบุรีบูตโดยไม่มี `bosboot` ในวิธี ต่อไปนี้:

- อาร์กิวเมนต์ 0 สำหรับแฟล็ก `epkg -r` ระบุว่า จำเป็นต้องรีบูต ("only") แต่คำสั่ง `emgr` ไม่ควรเรียก `bosboot` (นั่นคือ สร้างบูตอิมเมจใหม่)
- รีบูตพร้อมต์ในโหมดการโต้ตอบแสดงอ็อปชัน ต่อไปนี้:
  - เลือกนโยบายรีบูตสำหรับแพ็กเกจ `efix` นี้:
    - 1) ไม่จำเป็นต้องรีบูต
    - 2) Reboot is required. The boot image will be rebuilt.
    - 3) Reboot is required. The boot image will NOT be rebuilt.
- ตั้งค่าแอตทริบิวต์ `BUILD_BOOT_IMAGE` และ `REBOOT` เป็น "yes" หรือ "no" ในไฟล์ควบคุม `interim fix` อ็อปชัน `REBOOT` และ `BUILD_BOOT_IMAGE` ต่อไปนี้ได้รับการสนับสนุน:

#### ไอเท็ม

`REBOOT=no & BUILD_BOOT_IMAGE=no`  
`REBOOT=yes & BUILD_BOOT_IMAGE=yes`  
`REBOOT=yes & BUILD_BOOT_IMAGE=no`

#### คำอธิบาย

ไม่จำเป็นต้องบูตใหม่  
จำเป็นต้องบูตใหม่ บูตอิมเมจจะถูกสร้างใหม่  
จำเป็นต้องบูตใหม่ บูตอิมเมจจะ *ไม่* ถูกสร้างใหม่

หมายเหตุ: `REBOOT=no & BUILD_BOOT_IMAGE=yes` จะให้ผลลัพธ์ข้อผิดพลาดจากคำสั่ง `epkg`

## แฟล็ก

#### ไอเท็ม

`-a APARrefFile`  
`-d คำอธิบายFile`  
`-e interimfixControlFile`  
`-g PrerequisiteFile`  
  
`-l LockFile`  
  
`-p PrerequisiteFile`  
`-r {ynlo}`  
  
`-S SupersedeFile`  
`-s`  
`-T`  
`-u {yesno}`  
`-v`  
  
`-w WorkDirectory`  
`-X`

#### คำอธิบาย

ระบุไฟล์ที่มีหมายเลขอ้างอิง APAR  
ระบุไฟล์ที่มีรายละเอียด `interim fix`  
ระบุไฟล์ควบคุม `interim fix` ที่ควบคุมวิธีที่ `interim fix` ถูกสร้าง  
ระบุตำแหน่งของไฟล์ที่จำเป็นต้องมี `interim fix` ที่มีชื่อเลเบล `interim fix` เลเบลเหล่านี้ จำเป็นก่อนที่จะติดตั้งแพ็กเกจ `interim fix`  
ระบุตำแหน่งของไฟล์ที่ล็อกที่มีชื่อแพ็กเกจ แพ็กเกจเหล่านี้ถูกล็อกโดยคำสั่ง `emgr` หรือจากความเป็นเจ้าของไฟล์โดยอัตโนมัติ  
ระบุไฟล์ที่มีสิ่งที่จำเป็นต้องมีของ `installp`  
เซตแอตทริบิวต์ `epkg REBOOT` ซึ่งทำให้คำสั่ง `emgr` ทำการเปลี่ยนแปลงตามความจำเป็นกับบูตอิมเมจ และส่งข้อความแนะนำแก่ผู้ใช้ให้รีบูตหลังจาก การติดตั้ง อาร์กิวเมนต์ `y` ระบุว่า `bosboot` จำเป็น อาร์กิวเมนต์ `n` ระบุว่าไม่จำเป็นต้องรีบูต อาร์กิวเมนต์ `o` ระบุว่าจำเป็นต้องรีบูต แต่ `emgr` ไม่ควรเรียก `bosboot`  
ระบุตำแหน่งของไฟล์แทนที่ `interim fix` ที่มีชื่อเลเบล `interim fix` เลเบลเหล่านี้ จะถูกแทนที่เมื่อ `epkg` ถูกติดตั้ง  
ทำให้คำสั่ง `epkg` ข้ามคำถามเกี่ยวกับสคริปต์และไฟล์ที่จำเป็นต้องมี  
ระบุว่า `interim fix` นี้ ถูกทดสอบแล้วหรือไม่ ค่าที่ใช้ได้คือ `yes` หรือ `no` ดีฟอลต์คือ `no`  
ระบุว่าคุณจะใช้ไฟล์ควบคุม `interim fix` ที่มีอยู่หรือไม่  
ทำให้คำสั่ง `epkg` ถามคำถามเพิ่มเติมสำหรับอ็อปชันส่วนขยาย ซึ่งรวมถึงถามคุณ ให้ระบุสิทธิ์กับไฟล์ `interim fix` ทั้งหมด  
ระบุไดเรกทอรีทำงานสำรองที่ คำสั่ง `epkg` จะใช้ ไดเรกทอรีทำงานดีฟอลต์คือ `$HOME/epkgwork`  
ทำให้คำสั่ง `emgr` ขยายระบบไฟล์โดยอัตโนมัติ เมื่อ `interim fix` ถูก ติดตั้ง ถ้าพื้นที่ที่จำเป็นและส่วนขยายเป็นไปได้

## พารามิเตอร์

### *interim fixLabel*

ระบุสตริงที่ระบุเป็นค่าเฉพาะแพ็คเกจ interim fix นี้ ความยาวสูงสุดของเลเบล interim fix คือ 10 ไบต์

หมายเหตุ: ตัวจัดการ interim fix ต้องการให้แต่ละเลเบล interim fix บนระบบเป็นค่าเฉพาะ

## คำสั่งย่อย

- b! ส่งกลับไปที่คำถามก่อนหน้านี้
- s! แสดงสถานะของไฟล์ควบคุม interim fix ปัจจุบัน
- q! ออกโดยไม่บันทึกไฟล์ควบคุม interim fix (ใช้ลำดับคีย์ Ctrl+C ทำให้คำสั่ง epkg ถามคุณว่าคุณต้องการบันทึกไฟล์ควบคุม interim fix หรือไม่)
- h! แสดงข้อมูลวิธีใช้สำหรับคำถามปัจจุบัน

## สถานะออก

- 0 การดำเนินการคำสั่ง epkg สำเร็จ สมบูรณ์
- >0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการรันคำสั่ง epkg ในโหมดการโต้ตอบ และสร้างแพ็คเกจ interim fix ด้วยเลเบล interim fix ของ myfix ให้พิมพ์:  
epkg myfix
2. เมื่อต้องการสร้างแพ็คเกจ interim fix ด้วยเลเบล interim fix ของ myfix โดยใช้ไฟล์ควบคุม interim fix ที่ชื่อ /tmp/ecfile ให้พิมพ์:  
epkg -e /tmp/ecfile myfix
3. เมื่อต้องการสร้างแพ็คเกจ interim fix ด้วยเลเบล interim fix ของ myfix และระบบไฟล์สิ่งที่จะต้อง มี /tmp/prereq, รายละเอียด /tmp/คำอธิบาย, และอ็อปชันเพิ่มเติม ให้พิมพ์:  
epkg -v -p /tmp/prereq -d /tmp/description myfix

## ไฟล์

ไอเท็ม  
/usr/sbin/epkg

คำอธิบาย  
มีคำสั่ง epkg

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง emgr” ในหน้า 371

### ข้อมูลที่เกี่ยวข้อง:

การติดตั้งผลิตภัณฑ์ซอฟต์แวร์ทางเลือกหรืออัปเดตเซอวิส

---

## คำสั่ง eqn

### Purpose

จัดรูปแบบข้อความทางคณิตศาสตร์สำหรับคำสั่ง troff

### ไวยากรณ์

```
eqn [-d Delimiter1Delimiter2] [-f Font] [-p Number] [-s Size] [-T Name] [-] [File ...|-]
```

### รายละเอียด

คำสั่ง eqn คือตัวประมวลผลก่อนของ troff สำหรับข้อความทางคณิตศาสตร์ typesetting บน phototypesetter หรืออุปกรณ์ที่เทียบได้ เอาต์พุตของคำสั่ง eqn โดยทั่วไปไปพ่วงลงในคำสั่ง troff ดังนี้:

```
eqn [Flag...] File... | troff [Flag...]| [Typesetter]
```

คำสั่ง eqn อ่านไฟล์ที่ระบุโดย พารามิเตอร์ File โดยอ่านอินพุตมาตรฐานเมื่อ - (เครื่องหมายลบ) ถูกระบุเป็นพารามิเตอร์สุดท้าย บรรทัดจะขึ้นต้นด้วยแม่โคร .EQ ที่ทำเครื่องหมายเริ่มต้น ของข้อความสมการ ส่วนท้ายของข้อความสมการถูกทำเครื่องหมายโดยบรรทัดที่ขึ้นต้นด้วยแม่โคร .EN บรรทัดเหล่านี้ไม่ถูกเปลี่ยนแปลงโดยคำสั่ง troff ดังนั้นสามารถถูกกำหนดในแม่โครแพ็คเกจเพื่อจัดเตรียมฟังก์ชันการจัดรูปแบบเพิ่มเติม เช่นการจัดให้อยู่ตรงกลาง และการกำหนดหมายเลข

### คีย์เวิร์ด

ต่อไปนี้เป็นคีย์เวิร์ดที่ทั้งสองคำสั่ง eqn และ neqn รู้จัก

above	dot	gsize	over	tdefine
back	dotdot	hat	pile	tilde
bar	down	italic	rcol	to
bold	dyad	lcol	right	under
ceiling	fat	left	roman	up
ccol	floor	lineup	rpile	vec
col	font	lpile	size	
cpile	from	mark	sqrt	
define	fwd	matrix	sub	
delim	gfont	ndefine	sup	

คีย์เวิร์ดที่รู้จักโดยคำสั่ง eqn สามารถถูกเซตแยกด้วย ช่องว่าง แท็บ บรรทัดใหม่ เครื่องหมายปีกกา เครื่องหมายคำพูดคู่ tildes และเครื่องหมายที่อยู่บนสระ ใช้ { } (เครื่องหมายปีกกา) สำหรับการจัดกลุ่ม คุณสามารถใช้อักขระเดี่ยวได้ทุกที่ เช่น X คุณสามารถแทนที่โครงสร้างที่ซับซ้อนที่อยู่ในเครื่องหมายปีกกา ~ (tilde) หมายถึงที่ว่างเต็มในเอาต์พุต และ ^ (circumflex) หมายถึงครึ่งที่ว่าง

สร้างตัวห้อยและตัวยกโดยใช้คีย์เวิร์ด sub และ sup สร้างเศษส่วนด้วยคีย์เวิร์ด over สร้างสแควร์รูทด้วยคีย์เวิร์ด sqrt

เพิ่มการจำกัดค่าล่างและค่าบนโดยใช้คีย์เวิร์ด from และ to สร้างตัวคั่น (เช่น วงเล็บเหลี่ยมและเครื่องหมายปีกกาซ้ายและขวา) ของความสูงที่ต้องการโดยใช้คีย์เวิร์ด left และ right อักขระ Legal หลังคีย์เวิร์ด left และ right คือเครื่องหมายปีกกา วงเล็บเหลี่ยม แท่ง, c และ f สำหรับ ceiling และ floor และ “ ” (อัญประกาศ) สำหรับค่าว่างเปล่า (ซึ่งมีประโยชน์สำหรับวงเล็บเหลี่ยมด้านขวาเท่านั้น) อักขระ left ไม่จำเป็นต้องตรงกับอักขระ right แต่อักขระ right ต้องตรงกับ อักขระ left

รายการแนวตั้ง (ซ้อน) ของข้อมูลถูกสร้างด้วยคีย์เวิร์ด **pile**, **lpile**, **cpile** และ **rpile** ชั้นข้อมูลมีจำนวนส่วนประกอบ ตามที่กำหนด คีย์เวิร์ด **lpile** จัดชิดซ้าย คีย์เวิร์ด **pile** และ **cpile** จัดกลาง (แต่มีระยะห่างแนวตั้งต่างไป) และคีย์เวิร์ด **rpile** จัดขวา เมทริกถูกสร้างด้วยคีย์เวิร์ด **matrix** นอกจากนี้ มีคีย์เวิร์ด **rcol** สำหรับคอลัมน์ ที่จัดขอบขวา

เครื่องหมายการออกเสียงถูกสร้างด้วยคีย์เวิร์ด **dot**, **dotdot**, **hat**, **tilde**, **bar**, **vec**, **dyad** และ **under**

คุณสามารถเปลี่ยนขนาดพอยท์และฟอนต์ด้วยคีย์เวิร์ด **size Number** (หรือ **size +/-Number**), **roman**, **italic**, **bold** และ **font Number** คุณสามารถเปลี่ยนขนาดพอยท์และฟอนต์ ทั้งทั้งเอกสารด้วยคีย์เวิร์ด **gsize Number** และ **gfont Number** หรือด้วยแฟล็กบรรทัดคำสั่ง **-sNumber** และ **-fNumber**

โดยปกติ ตัวห้อยและตัวยกถูกลดขนาดลง สามพอยท์จากขนาดก่อนหน้านี้ คุณสามารถเปลี่ยนได้ด้วยแฟล็กบรรทัดคำสั่ง **-pNumber**

คุณสามารถเรียงพารามิเตอร์แสดงผลที่ต่อเนื่องกัน วาง คีย์เวิร์ด **mark** หน้าจุดที่ต้องการเรียง ใน สมการแรก วางคีย์เวิร์ด **lineup** ที่จุดที่ต้องการเรียงในแนวตั้งในสมการต่อมา

คุณสามารถกำหนดตัวเลขหรือกำหนดคีย์เวิร์ดใหม่ ด้วยคีย์เวิร์ด **define** ตัวอย่าง:

```
define Thing%Replacement%
```

ตัวอย่างก่อนหน้านี้กำหนดโทเคนใหม่ชื่อ *Thing* ซึ่งถูกแทนที่โดย *Replacement* เมื่อใดก็ตามที่ค่าแสดงในภายหลัง % (เครื่องหมายเปอร์เซ็นต์) สามารถเป็นอักขระใดๆ ที่ไม่มีอยู่ใน *Replacement*

คีย์เวิร์ดเช่น **sum**, **int**, **inf** และตัวเลขเช่น **>=**, **!=** และ **->** เป็น ที่รู้จัก ตัวอักษรกรีกถูกสะกดในตัวพิมพ์ที่ต้องการ ดังนั้น **alpha** หรือ **GAMMA** คำทางคณิตศาสตร์เช่น **sin**, **cos** และ **log** ถูกทำเป็น Roman โดยอัตโนมัติ คำสั่ง **troff command 4**-อักขระ escapes, เช่น **\(dd**, ซึ่งสร้าง dagger คุณสามารถใช้ได้ทุกที่ สตรีงใน “ ” (อัญประกาศ) ถูกส่งผ่านโดยไม่มีการแก้ไข เพื่ออนุญาตให้คีย์เวิร์ดถูกป้อนได้เป็นข้อความ และสามารถถูกใช้เพื่อสื่อสารกับ คำสั่ง **troff** ได้เสมอ

## แฟล็ก

### ไอเท็ม

**-dDelimiter1Delimiter2**

### คำอธิบาย

เซตสองอักขระ ASCII *Delimiter1* และ *Delimiter2*, เป็นตัวคั่นของข้อความที่จะถูกประมวลผล โดยคำสั่ง **eqn** นอกเหนือจากอินพุตที่ถูกปิดล้อม โดยแม่โคร **.EQ** และ **.EN** ข้อความ ระหว่างตัวคั่นเหล่านี้ถูกปฏิบัติเป็นอินพุตของคำสั่ง **eqn**

หมายเหตุ: ภายในไฟล์ คุณยังสามารถ เซตตัวคั่นสำหรับข้อความ **eqn** โดยใช้คำสั่ง **delim Delimiter1Delimiter2** ซึ่งถูกปิด โดยคำสั่ง **delim off** ข้อความทั้งหมดที่ไม่อยู่ระหว่างแม่โคร **.EQ** และ **.EN** ถูกส่งผ่านโดยไม่มีการประมวลผล

**-fFont**

เปลี่ยนฟอนต์ในคำสั่ง **eqn** ทั้งหมดที่ประมวลผล ข้อความเป็นค่าที่ระบุโดยตัวแปร *Font* คำ *font* (ชื่อฟอนต์หรือตำแหน่ง) ต้องเป็นอักขระ ASCII หนึ่งหรือสองอักขระ ลดตัวห้อยและตัวยกตามจำนวนพอยต์ขนาด ที่ระบุ (ดีฟอลต์คือ 3)

**-pNumber**

เปลี่ยนขนาดฟอนต์ในคำสั่ง **eqn** ทั้งหมดที่ประมวลผล ข้อความเป็นค่าที่ระบุโดยตัวแปร *Size* เตรียมเอาต์พุตสำหรับอุปกรณ์การพิมพ์ที่ระบุ Terminal Names for Phototypesetter หรือ Comparable Devices จัดเตรียมตัวแปร *Name* คำดีฟอลต์คือ **ibm3816**

**-s Size**

**-T Name**

-

-

บังคับให้อ่านอินพุตจากอินพุตมาตรฐาน (เส้นประคู่) หมายถึงจุดสิ้นสุดแฟล็ก

## ไฟล์

ไอเท็ม  
/usr/share/lib/pub/eqnchar

คำอธิบาย  
มีนิยามของอักขระพิเศษ

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mvt

คำสั่ง neqn

คำสั่ง troff

รูปแบบไฟล์ eqnchar

---

## คำสั่ง errclear

### วัตถุประสงค์

ลบรายการจากบันทึกข้อผิดพลาด

### ไวยากรณ์

```
errclear [-d ErrorClassList] [-i File] [-J ErrorLabel[,Errorlabel]] [-K ErrorLabel[,Errorlabel]] [-I SequenceNumber] [-m Machine] [-n Node] [-N ResourceNameList] [-R ResourceTypeList] [-S ResourceClassList] [-T ErrorTypeList] [-y FileName] [-j ErrorID[,ErrorID]] [-k ErrorID[,ErrorID]] Days
```

### คำอธิบาย

คำสั่ง `errclear` ลบ รายการบันทึกข้อผิดพลาดที่เก่ากว่าจำนวนวันที่ระบุโดยพารามิเตอร์ `Days` เมื่อต้องการลบรายการบันทึกข้อผิดพลาด ระบุค่า `0` สำหรับพารามิเตอร์ `Days`

ถ้าแฟล็ก `-i` ไม่ถูกใช้ กับคำสั่ง `errclear` ไฟล์บันทึกข้อผิดพลาดที่ลบโดย `errclear` คือไฟล์ที่ระบุในฐานข้อมูลคอนฟิกูเรชันบันทึกข้อผิดพลาด (เมื่อต้องการดูข้อมูลในฐานข้อมูลคอนฟิกูเรชันบันทึกข้อผิดพลาด ใช้คำสั่ง `errdemon`)

หมายเหตุ: คำสั่ง `errclear` ลบรายการที่ระบุแต่ไม่ลบขนาดไฟล์บันทึกข้อผิดพลาด

คุณสามารถใช้พาดว่น `smit errclear` ของ System Management Interface Tool (SMIT) เพื่อรันคำสั่งนี้

### แฟล็ก

ไอเท็ม

`-d List`

`-i File`

`-j ErrorID[,ErrorID]`

`-J ErrorLabel`

คำอธิบาย

ลบรายการบันทึกข้อผิดพลาดในคลาสข้อผิดพลาดที่ระบุโดยตัวแปร `List` ค่าตัวแปร `List` สามารถถูกแยกโดย, (คอมมา) หรืออยู่ใน " " (เครื่องหมายคำพูด) และแยกโดย, (คอมมา) หรืออักขระเว้นวรรค ค่าตัวแปร `List` ที่ใช้ได้คือ `H` (hardware), `S` (software), `O` (ข้อความ `errlogger`) และ `U` (undetermined) ใช้ไฟล์บันทึกข้อผิดพลาดที่ระบุโดยตัวแปร `File` ถ้าแฟล็กนี้ไม่ถูกระบุ คำสั่ง `errclear` จะใช้ค่าจากฐานข้อมูลคอนฟิกูเรชันบันทึกข้อผิดพลาด

ลบรายการบันทึกข้อผิดพลาดที่ระบุโดยตัวแปร `ErrorID` (error identifier) ค่าตัวแปร `ErrorID` สามารถถูกแยกโดย, (คอมมา) หรืออยู่ใน " " (เครื่องหมายคำพูด) และแยกโดย, (คอมมา) หรืออักขระเว้นวรรค

ลบรายการบันทึกข้อผิดพลาดที่ระบุโดยตัวแปร `ErrorLabel` ค่าตัวแปร `ErrorLabel` สามารถถูกแยกโดย, (คอมมา) หรืออยู่ใน " " (เครื่องหมายคำพูด) และแยกโดย, (คอมมา) หรืออักขระเว้นวรรค

ไอเท็ม	
-k <i>ErrorID</i> {, <i>ErrorID</i> }	คำอธิบาย ลบรายการบันทึกข้อผิดพลาดทั้งหมดยกเว้นที่ระบุโดยตัวแปร <i>ErrorID</i> (error identifier) ค่าตัวแปร <i>ErrorID</i> สามารถถูกแยกโดย, (คอมมา) หรืออยู่ใน " " (เครื่องหมายคำพูด) และแยกโดย, (คอมมา) หรืออักขระเว้นวรรค
-K <i>ErrorLabel</i>	ลบรายการบันทึกข้อผิดพลาดทั้งหมดยกเว้นที่ระบุโดยตัวแปร <i>ErrorLabel</i> ค่าตัวแปร <i>ErrorLabel</i> สามารถถูกแยกโดย, (คอมมา) หรืออยู่ใน " " (เครื่องหมายคำพูด) และแยกโดย, (คอมมา) หรืออักขระเว้นวรรค
-l <i>SequenceNumber</i>	ลบรายการบันทึกข้อผิดพลาดที่มีหมายเลขลำดับที่ระบุ ค่าตัวแปร <i>SequenceNumber</i> สามารถแยกโดย, (จุลภาค) หรืออยู่ใน " " (เครื่องหมายคำพูด) และแยกโดย, (จุลภาค) หรืออักขระเว้นวรรค
-m <i>Machine</i>	ลบรายการบันทึกข้อผิดพลาดสำหรับเครื่องที่ระบุโดยตัวแปร <i>Machine</i> คำสั่ง <code>uname -m</code> ส่งกลับ ค่าของตัวแปร <i>Machine</i>
-n <i>Node</i>	ลบรายการบันทึกข้อผิดพลาดสำหรับโหนดที่ระบุโดยตัวแปร <i>Node</i> คำสั่ง <code>uname -n</code> ส่งกลับ ค่าของตัวแปร <i>Node</i>
-N <i>List</i>	ลบรายการบันทึกข้อผิดพลาดสำหรับชื่อรีซอร์สที่ระบุโดยตัวแปร <i>List</i> ตัวแปร <i>List</i> คือรายชื่อของรีซอร์สที่ตรวจพบข้อผิดพลาด สำหรับข้อผิดพลาดซอฟต์แวร์ ข้อมูลเหล่านี้เป็นชื่อของรีซอร์สที่ตรวจพบข้อผิดพลาด สำหรับข้อผิดพลาด ฮาร์ดแวร์ ข้อมูลเหล่านี้เป็นชื่อของคอมโพเนนต์อุปกรณ์หรือระบบ ซึ่งไม่ได้ระบุว่าคอมโพเนนต์ล้มเหลวหรือจำเป็นต้องเปลี่ยน แต่ถูกใช้เพื่อกำหนดโมดูลการวินิจฉัยที่เหมาะสมเพื่อนำมาใช้เพื่อวิเคราะห์ข้อผิดพลาด ค่าตัวแปร <i>List</i> สามารถถูกแยกโดย, (คอมมา) หรืออยู่ใน " " (เครื่องหมายคำพูด) และแยกโดย, (คอมมา) หรืออักขระเว้นวรรค
-R <i>List</i>	ลบรายการบันทึกข้อผิดพลาดสำหรับชนิดรีซอร์สที่ระบุโดยตัวแปร <i>List</i> สำหรับข้อผิดพลาดฮาร์ดแวร์ ตัวแปร <i>List</i> คือชนิดอุปกรณ์ สำหรับข้อผิดพลาดซอฟต์แวร์ ค่าของตัวแปร <i>List</i> คือ LPP ค่าตัวแปร <i>List</i> สามารถถูกแยกโดย, (คอมมา) หรืออยู่ใน " " (เครื่องหมายคำพูด) และแยกโดย, (คอมมา) หรืออักขระเว้นวรรค
-S <i>List</i>	ลบรายการบันทึกข้อผิดพลาดสำหรับคลาสรีซอร์สที่ระบุโดยตัวแปร <i>List</i> สำหรับข้อผิดพลาดฮาร์ดแวร์ ตัวแปร <i>List</i> คือคลาสอุปกรณ์ ค่าตัวแปร <i>List</i> สามารถถูกแยกโดย, (คอมมา) หรืออยู่ใน " " (เครื่องหมายคำพูด) และแยกโดย, (คอมมา) หรืออักขระเว้นวรรค
-T <i>List</i>	ลบรายการบันทึกข้อผิดพลาดสำหรับชนิดข้อผิดพลาดที่ระบุโดยตัวแปร <i>List</i> ค่าตัวแปร <i>List</i> ที่ใช้ได้คือ: <b>PERM, TEMP, PERF, PEND, INFO</b> และ <b>UNKN</b> ค่าตัวแปร <i>List</i> สามารถถูกแยกโดย, (คอมมา) หรืออยู่ใน " " (เครื่องหมายคำพูด) และแยกโดย, (คอมมา) หรืออักขระเว้นวรรค
-y <i>FileName</i>	ใช้ไฟล์เพิ่มเติมบันทึกข้อผิดพลาดที่ระบุโดยตัวแปร <i>FileName</i>

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: เฉพาะผู้ใช้ root เท่านั้นที่สามารถรันคำสั่งนี้ได้

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

- เมื่อต้องการลบรายการทั้งหมดจากบันทึกข้อผิดพลาด ให้ป้อน:

```
errclear 0
```
- เมื่อต้องการลบรายการทั้งหมดในบันทึกข้อผิดพลาดที่จำแนก เป็นข้อผิดพลาดซอฟต์แวร์ ให้ป้อน:

```
errclear -d S 0
```
- เมื่อต้องการล้างรายการทั้งหมดจากไฟล์บันทึกข้อผิดพลาด สำหรับ `/var/adm/ras/errlog.alternate`, ให้ป้อน:

```
errclear -i /var/adm/ras/errlog.alternate 0
```
- To clear all hardware entries from the alternate error-log file `/var/adm/ras/errlog.alternate`, enter:

```
errclear -i /var/adm/ras/errlog.alternate -d H 0
```

# ไฟล์

ไอเท็ม  
/etc/objrepos/SWservAt

คำอธิบาย  
มีคลาสอ็อบเจ็กต์ Software Service Aids Attributes ซึ่ง คือฐานข้อมูลคอนฟิกูเรชันบันทึกข้อผิดพลาด

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง errlogger” ในหน้า 435

ข้อมูลที่เกี่ยวข้อง:

รูทีนย่อย uname

คำสั่ง errlog

ภาพรวมล็อกข้อผิดพลาด

---

## คำสั่ง errctrl

### วัตถุประสงค์

แก้ไขหรือแสดงแอตทริบิวต์การตรวจสอบข้อผิดพลาดของคอมโพเนนต์ระบบ ค่าแอตทริบิวต์คองที่สามารถระบุให้กับคอมโพเนนต์ที่ยังไม่ถูกสร้างได้เช่นกัน

### ไวยากรณ์

```
errctrl [ -nru ] ComponentSelector ... subcommand ...
```

```
errctrl -p [ -ru ] ComponentSelector ... subcommand ...
```

```
errctrl -P [ -ru ] ComponentSelector ... subcommand ...
```

```
errctrl -x { -P | -p } [ -ru ] ComponentSelector ...
```

```
errctrl -q [ -rupP ] { ComponentSelector ... }
```

```
errctrl { -h | -? }
```

```
errctrl -P { errcheckon | errcheckoff }
```

### คำอธิบาย

คำสั่ง `errctrl` แก้ไขหรือแสดงค่าแอตทริบิวต์การตรวจสอบข้อผิดพลาดของคอมโพเนนต์บางส่วนหรือทั้งหมด คอมโพเนนต์ถูกเลือกตามชื่อ ตาม `alias` หรือตามชนิดหรือชนิดย่อย

ค่าที่สนับสนุนของ `ComponentSelector` มีดังนี้:

`-c` componentPatternList

`-l` aliasPatternList

-t typePatternList

แต่ละรายการประกอบด้วยรูปแบบหนึ่งรูปแบบหรือมากกว่านั้นค้นด้วย เครื่องหมายคำพูดว่าง หรือคอมมาแพ็คเกจ์ที่รัน สามารถมีอักขระพิเศษตามที่อธิบายโดยรูทีนย่อย `fnmatch` อักขระแพ็คเกจ์ที่รัน เครื่องหมายคำถาม (?), เครื่องหมายดอกจัน (\*) และ ([ ]) มีการสนับสนุน แต่คลาสอักขระและลำดับ การเรียงไข่มไม่ได้ภายในวงเล็บเหลี่ยม ([ ]) การระบุ `-call` เลือกคอมโพเนนต์ทั้งหมด ถ้าไม่มีการใช้ `ComponentSelector` อื่น

คำสั่ง `errctrl` ยังสามารถถูกใช้กับแฟล็ก `-p` หรือ `-P` เพื่อระบุ การกำหนดเองแอดทริบิวต์คังที่ สำหรับข้อมูลเพิ่มเติม เกี่ยวกับการกำหนดค่าแอดทริบิวต์ถาวรเอง ดูที่ การกำหนดค่าถาวรเอง

เมื่อต้องการเปิดหรือปิดใช้งานการตรวจสอบข้อผิดพลาดสำหรับคอมโพเนนต์ทั้งหมดทันที และตลอดไปให้ระบุคำสั่งย่อย `errcheckon` หรือ `errcheckoff` กับแฟล็ก `-P` พอร์มของคำสั่ง นี้ไม่อนุญาตแฟล็กหรือคำสั่งย่อยอื่น คำสั่ง `bosboot` จำเป็นในการทำให้การตั้งค่าคงอยู่หลังรีสตาร์ท

แอดทริบิวต์ที่ดัดแปลงขึ้นอยู่กับคำสั่งย่อยที่ระบุในบรรทัด คำสั่ง สามารถระบุได้หลายคำสั่งย่อยในการเรียกครั้งเดียว คำสั่งย่อยต่อไปนี้ มีอยู่:

ไอเท็ม	คำอธิบาย
<code>errcheckon</code>	เปิดการตรวจสอบข้อผิดพลาด
<code>errcheckoff</code>	ปิดการตรวจสอบข้อผิดพลาด
<code>errcheckminimal</code>	เซ็ระดับการตรวจสอบข้อผิดพลาดเป็น 1
<code>errchecknormal</code>	เซ็ระดับการตรวจสอบข้อผิดพลาดเป็น 3
<code>errcheckdetail</code>	เซ็ระดับการตรวจสอบข้อผิดพลาดเป็น 7
<code>errchecklevel={0-9}</code>	เซ็ระดับการตรวจสอบข้อผิดพลาดเป็นค่าที่ระบุ
<code>lowsevdisposition={disp}</code>	เซ็การจัดการสำหรับข้อผิดพลาดความสำคัญต่ำเป็น ค่าที่ระบุ
<code>medsevdisposition={disp}</code>	เซ็การจัดการสำหรับข้อผิดพลาดความสำคัญปานกลางเป็น ค่าที่ระบุ

การจัดการข้อผิดพลาด `disp` มีค่าหนึ่งในค่า ต่อไปนี้:

- `ignore` (หรือ 48)
- `log` (หรือ 64)
- `livedump` (หรือ 80)
- `isolate` (หรือ 96)
- `sysdump` (หรือ 112)

คำสั่งย่อยอื่นสามารถถูกจดจำได้ตามแต่ละคอมโพเนนต์ คำสั่งย่อย ที่คอมโพเนนต์ไม่รู้จักจะถูกข้ามไป

ค่าแอดทริบิวต์ `Current` สามารถแสดงด้วยแฟล็ก `-q` ถ้าไม่มีการใช้ `ComponentSelector` ค่าแอดทริบิวต์ ถูกแสดงสำหรับคอมโพเนนต์ทั้งหมด ซึ่งมีการสนับสนุนการตรวจสอบข้อผิดพลาด

ระบบการตรวจจับการซ้อนกันของหน่วยความจำสำหรับหน่วยความจำเน็ตเวิร์กสามารถถูกเปิดใช้ได้โดยการ เซ็ระดับข้อผิดพลาดโดยละเอียดสำหรับคอมโพเนนต์ `netmalloc componen` เพิ่ม `errlevel` สำหรับ คอมโพเนนต์ `netmalloc` เป็นห้าหรือมากกว่า (ดีฟอลต์: 3(ปกติ)) เพื่อรวบรวมข้อมูลบัฟเฟอร์ ควบคุมหน่วยความจำเน็ตเวิร์กสำหรับการจัดสรรเน็ตเวิร์กและเหตุการณ์ว่างทั้งหมด หมายเหตุ การเพิ่มระดับข้อผิดพลาดเป็นเจ็ด (รายละเอียด) หรือมากกว่า นั้น ยังเป็นการเปิดใช้ระบบตรวจจับการซ้อนกันของหน่วยความจำเน็ตเวิร์ก เมื่อต้องการเปิดใช้งานเฉพาะอ็อพชัน `net_malloc_police` และการบันทึก `outstanding memory allocation (OSTD)` สำหรับการจัดสรรหน่วยความจำเน็ตเวิร์กและเหตุการณ์ว่างทั้งหมด, เพิ่ม ระดับข้อผิดพลาดเป็นห้า

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการแก้ไข errlevel ดูที่ “ตัวอย่าง” ในหน้า 427 สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการเพิ่มระดับการติดตามเพื่อรวบรวมข้อมูลการติดตามใน คอมโพเนนต์ netmalloc ดูที่คำสั่ง ctctrl

คำสั่งนี้สามารถถูกใช้เพื่อเซตความเป็นไปได้ (ความถี่) และค่า กับฟังก์ชัน netmalloc ต่อไปนี้

- police\_frequency
- frag\_mask

ความน่าจะเป็นเป็นเป็นตัวตั้งจาก 1024 (ตัวอย่าง 10%: 102, 5%: 51, 1%: 10, 0.1%: 1)

### การกำหนด Persistent เอง

แฟล็ก -p และ -P อนุญาตให้ ระบุค่าแอตทริบิวต์สำหรับคอมโพเนนต์ระบบที่ยังไม่ได้สร้าง ดังนั้น แอตทริบิวต์สำหรับคอมโพเนนต์ที่สร้างใหม่ สามารถถูกกำหนดก่อน กลายเป็นแอคทีฟ แฟล็ก -p ถูกใช้เพื่อระบุ การกำหนดเองสำหรับคอมโพเนนต์ที่จะถูกสร้างในอนาคต แต่ก่อนที่ คุณรีสตาร์ท AIX แฟล็ก -P ถูกใช้เพื่อระบุ การกำหนดเองที่จะมีผลหลังจากการรีสตาร์ทครั้งต่อไป การกำหนดเองเหล่านี้ ถูกเพิ่มให้กับไฟล์ /var/adm/ras/rasptune คุณต้อง รันคำสั่ง bosboot เพื่อบันทึกการกำหนดเองเหล่านี้ ในบูตอิมเมจและรีสตาร์ท AIX เพื่อให้การกำหนดเองมีผล

*ComponentSelectors* มีอีกวิธีการจับคู่รูปแบบ ได้ ดังนั้น การกำหนดค่าตัวเอง สามารถใช้กับคอมโพเนนต์มากกว่าหนึ่งคอมโพเนนต์ นอกจากนี้ การกำหนดเองหลายครั้งสามารถใช้กับคอมโพเนนต์เดียวกัน ถ้ามีการใช้ *ComponentSelectors* ที่ต่างกัน ถ้ามีการระบุค่าแอตทริบิวต์ขัดแย้งกัน ในหลายการกำหนดเอง การกำหนดเองล่าสุดจะมีลำดับความสำคัญ มากที่สุด ถ้าการกำหนดเองมีอยู่แล้วใน *ComponentSelector* ที่ระบุ การกำหนดเองใหม่จะแทนที่ค่าเก่า

สามารถมีหลาย *ComponentSelectors* ได้เมื่อการกำหนดเองที่มีค่า คงอยู่ถูกระบุ แต่ในทุกกรณี การใช้หลายอ็อปชันเหมือนกับการระบุหลายคำสั่ง โดยแต่ละคำสั่งมีหนึ่งอ็อปชันคอมโพเนนต์ ตัวอย่าง การกำหนดเอง "errctrl -p -l hdisk0 -l hdisk1 errchecknormal" เหมือนกับสองการกำหนดเองต่อไปนี้:

```
errctrl -p -l hdisk0 errchecknormal
errctrl -p -l hdisk1 errchecknormal
```

การกำหนดเองที่ระบุด้วยแฟล็ก -p or -P ไม่ถูกลบหลังจากที่ถูกใช้ ดังนั้น การกำหนดเองหนึ่งครั้ง สามารถมีผลกับหลายคอมโพเนนต์ใหม่ การกำหนดเองที่มีค่าคงอยู่สามารถถูกลบ ได้ด้วยแฟล็ก -x *ComponentSelector* ต้องถูกระบุเหมือนกับที่ถูกระบุเมื่อการกำหนดเอง ถูกสร้างขึ้น ตัวอย่าง ถ้าการกำหนดเองถูกสร้างด้วย *ComponentSelector -l hdisk0* การกำหนดเองไม่สามารถถูกลบ ด้วย *ComponentSelector -l hdisk[0]*, แมว่าทั้งสอง *ComponentSelectors* ตรงกับ คอมโพเนนต์ alias เดียวกัน เมื่อการกำหนดเองที่มีค่าคงอยู่ถูกลบ จะไม่มีการเปลี่ยนแปลงกับแอตทริบิวต์ของคอมโพเนนต์ที่ถูกสร้างเมื่อการกำหนดเอง แอคทีฟ

การกำหนดเองที่มีค่าคงอยู่ที่ลบด้วยแฟล็ก -x และแฟล็ก -P จะยังมีผลอยู่นอกจากคุณรันคำสั่ง bosboot และรีสตาร์ท AIX การกำหนดเองที่มีค่าคงอยู่ที่ถูกสร้าง ด้วยแฟล็ก -P สามารถถูกลบหลังจาก รีสตาร์ทโดยใช้แฟล็ก -x และ -p ในกรณีนี้ การกำหนดเองจะถูกแอคทีฟอีกครั้งถ้าคุณรีสตาร์ท AIX

ถ้าคุณไม่รู้จักการกำหนดเองที่ได้สร้างขึ้น แต่ต้องการ เรียกคืนค่าติดตั้งระบบดีฟอลต์ คุณสามารถเลือกทำได้หนึ่งในหัวข้อต่อไปนี้:

- ในไฟล์ /var/adm/ras/rasptune ลบ บรรทัดที่สัมพันธ์กับการกำหนดเองและรันคำสั่ง bosboot เพื่อเริ่มต้น AIX
- อ่านไฟล์ /var/adm/ras/rasptune เพื่อ ค้นหาแฟล็กและพารามิเตอร์ที่เหมาะสมที่ได้ถูกระบุ จากนั้นใช้ แฟล็ก -x เพื่อลบ การกำหนดเองตามที่แสดงใน ตัวอย่าง 6 ในหน้า 427 รันคำสั่ง bosboot และรีสตาร์ท AIX

แฟล็ก `-r` และ `-u` สามารถถูกใช้ เมื่อระบบการกำหนดเองที่มีค่าคงอยู่ การใช้หนึ่งแฟล็กระบุ name space อื่นสำหรับอ็อปชันคอมโพเนนต์ที่ระบุ การใช้สองแฟล็ก ในเวลาเดียวกัน เหมือนกับการเรียกสองคำสั่งแยกกัน แต่ละคำสั่ง มีแฟล็กคนละแฟล็ก ตัวอย่าง การกำหนดเองที่มีค่าคงอยู่ "errctrl -p -l hdisk0 -u -r errcheckdetail" เหมือนกับสองการกำหนดเอง ที่แยกกัน:

```
errctrl -p -l hdisk0 -u errcheckdetail
errctrl -p -l hdisk0 -r errcheckdetail
```

การกำหนดค่าถาวรเองต่อไปนี้ทั้งหมดแตกต่างกัน และสามารถ ถูกแก้ไขหรือลบออกโดยอิสระ

```
errctrl -p -l hdisk0 errcheckdetail
errctrl -p -l hdisk0 -r errcheckdetail
errctrl -p -l hdisk0 -u errcheckdetail
```

การกำหนดเอง Recursive-down (ที่ระบุโดยแฟล็ก `-r`) มีลำดับความสำคัญเหนือการกำหนดเองอื่นทั้งหมด ไม่ว่าลำดับซึ่งมีการระบุจะสัมพันธ์กับการกำหนดเองแบบ non-recursive-down อื่น

การกำหนดเองที่มีค่าคงอยู่สามารถถูกเคียวรีโดยใช้แฟล็ก `-q` พร้อมกับแฟล็ก `-P` หรือ `-p` การระบุแฟล็ก `-q` พร้อมกับแฟล็ก `-P` แสดงบรรทัดจากไฟล์ `/var/adm/ras/rasptune` การระบุแฟล็ก `-q` พร้อมกับแฟล็ก `-p` และ `-r` แสดงการกำหนดเองที่มีค่าคงอยู่ต้นฉบับ ที่ระบุด้วยแฟล็ก `-r` โดยไม่มีแฟล็ก `-r` แฟล็ก `-q` และ `-p` แสดงการกำหนดเองที่มีค่าคงอยู่ที่ระบุโดยมีหรือไม่มีแฟล็ก `-u`

การกำหนดเองที่มีค่าคงอยู่อนุญาตให้ระบุหลายคำสั่งย่อยได้ ถ้ามีการใช้คำสั่งย่อยที่ขัดแย้งกัน คำสั่งย่อยสุดท้ายจะถูกใช้ ตัวอย่าง คำสั่งย่อย `errchecknormal` และ `errcheckdetail` ระบุค่าต่างกันสำหรับแอตทริบิวต์การตรวจสอบข้อผิดพลาด ดังนั้นคำสั่งย่อยที่ระบุสุดท้ายจะถูกใช้

## แฟล็ก

### ไอเท็ม

<code>-c ComponentList</code>	<b>คำอธิบาย</b> ระบุรายการชื่อคอมโพเนนต์ที่แยกด้วยคอมมาหรือ แยกด้วยช่องว่าง แฟล็ก <code>-c all</code> เลือกคอมโพเนนต์ทั้งหมด ถ้าคอมโพเนนต์คือ <code>ComponentSelector</code> เท่านั้น
<code>-h or -?</code>	แสดงข้อความการใช้
<code>-l aliasList</code>	ระบุรายการคอมโพเนนต์ alias ที่แยกด้วยคอมมาหรือ แยกด้วยช่องว่าง
<code>-n</code>	ใช้คำสั่งย่อยทันที แฟล็กนี้เป็นค่าดีฟอลต์หากไม่ใช่แฟล็ก <code>-p</code> หรือ <code>-P</code>
<code>-P</code>	ระบุคำสั่งย่อยที่จะคงอยู่หลังจากรีสตาร์ท คุณต้องรันคำสั่ง <code>bosboot</code> และรีสตาร์ท AIX เพื่อให้คำสั่งย่อยเหล่านี้แอคทีฟ
<code>-x</code>	ลบการกำหนดค่าถาวรเองสำหรับ คอมโพเนนต์ที่ระบุ <code>ComponentSelector</code> ต้องถูกป้อนเหมือนกับ ที่ถูกป้อน เมื่อการกำหนดเองถูกระบุในครั้งแรก
<code>-p</code>	ระบุคำสั่งย่อย persistent คำสั่งย่อยที่ระบุ จะถูกใช้กับคอมโพเนนต์ที่สร้างใหม่
<code>-q</code>	เคียวรีการตั้งค่าแอตทริบิวต์ของคอมโพเนนต์ที่เลือก แฟล็กนี้ยังสามารถใช้กับแฟล็ก <code>-p</code> หรือ <code>-P</code> เพื่อแสดงการกำหนดเองที่มีค่าคงอยู่
<code>-r</code>	ใช้คำสั่งย่อยซ้ำกับคำสั่งย่อยทั้งหมด ของคอมโพเนนต์ที่เลือก
<code>-t type_subtypeList</code>	ระบุรายการที่แยกด้วยช่องว่างหรือแยกด้วยคอมมา ของชื่อ <code>type</code> หรือ <code>type_subtype</code> ชื่อ <code>type</code> ที่ใช้ได้รวมถึง <code>device</code> , <code>filesystem</code> , <code>network</code> , <code>services</code> , <code>storage</code> , และ <code>ui</code> รายการสมบูรณ์ของชื่อ <code>type</code> และ <code>type_subtype</code> อยู่ในเฮดเดอร์ไฟล์ <code>/usr/include/sys/ras_base.h</code>
<code>-u</code>	ใช้คำสั่งย่อยซ้ำกับต้นกำเนิด ของคอมโพเนนต์ที่ระบุ

หมายเหตุ: แฟล็ก `-u` และ `-r` สามารถใช้ร่วมกันได้ แฟล็ก `-c`, `-l` และ `-t` หลายแฟล็กสามารถใช้บนบรรทัดคำสั่งได้

## สถานะออก

ไอเท็ม  
0  
>0

คำอธิบาย  
ดำเนินการเป็นผลสำเร็จ  
มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

1. เมื่อต้องการเปิดการตรวจสอบข้อผิดพลาดโดยละเอียดสำหรับคอมโพเนนต์ JFS2 userdata ทั้งหมดให้ป้อน:

```
errctrl -c 'jfs2.filesystem.*.userdata' errcheckdetail
```

2. เมื่อต้องการระบุการกำหนดเองที่มีค่าคงอยู่สำหรับคอมโพเนนต์ userdata ของระบบไฟล์ JFS2 ใหม่ให้ป้อน:

```
errctrl -p -c 'jfs2.filesystem.*.userdata' errcheckminimal
```

จะไม่มีผลกับคอมโพเนนต์ userdata ที่มีอยู่แล้ว

3. เมื่อต้องการระบุการกำหนดเองที่จะคงอยู่หลังการรีสตาร์ทให้ป้อน:

```
errctrl -P -c 'jfs2.filesystem.*.userdata' errcheckminimal
```

ถ้าคุณรัน คำสั่ง **bosboot** และรีสตาร์ท AIX, การตรวจสอบข้อผิดพลาดขั้นต่ำ จะมีผลกับคอมโพเนนต์ JFS2 userdata ทั้งหมด

4. เมื่อต้องการขีดการตรวจสอบข้อผิดพลาดขั้นต่ำสำหรับคอมโพเนนต์ JFS2 userdata ปัจจุบันและในอนาคตให้ป้อน:

```
errctrl -npP -c 'jfs2.filesystem.*.userdata' errcheckminimal
```

5. เมื่อต้องการระบุค่าแอตทริบิวต์ที่คงอยู่หลายค่าให้กับคอมโพเนนต์อีเทอร์เน็ตให้ป้อน:

```
errctrl -P -c ethernet errcheckminimal medsevdposition=80
```

6. เมื่อต้องการลบการกำหนดเองที่ระบุในตัวอย่าง 2 ให้ป้อน:

```
errctrl -p -x -c 'jfs2.filesystem.*.userdata'
```

7. เมื่อต้องการแสดงการเองแอตทริบิวต์ ที่มีค่าคงอยู่, recursive-down ทั้งหมดให้ป้อน:

```
errctrl -q -p -r
```

8. เมื่อต้องการแสดงค่าแอตทริบิวต์การตรวจสอบข้อผิดพลาดปัจจุบันสำหรับคอมโพเนนต์ JFS2 และคอมโพเนนต์ที่สืบทอดให้ป้อน:

```
errctrl -q -c jfs2 -r
```

9. เมื่อต้องการเปิดใช้งาน memory overlay detection system (MODS) สำหรับหน่วยความจำเน็ตเวิร์ก (เพิ่มระดับข้อผิดพลาดเป็นระดับรายละเอียดสำหรับคอมโพเนนต์ให้ netmalloc) ให้ป้อน:

```
errctrl errcheckdetail -c netmalloc
```

หรือ

```
errctrl errchecklevel=7 -c netmalloc
```

หมายเหตุ: ซึ่งเปิดใช้อ็อปชัน `net_malloc_police` สำหรับการจัดสรรหน่วยความจำเน็ตเวิร์กและเหตุการณ์ว่าง เช่นกัน

- เมื่อต้องการเปิดใช้อ็อปชัน `net_malloc_police` สำหรับการจัดสรร หน่วยความจำและเหตุการณ์ว่างทั้งหมด ให้เพิ่มระดับข้อผิดพลาดคอมโพเนนต์ `netmalloc` เป็นห้าหรือมากกว่าให้ป้อน:

```
errctrl errchecklevel=5 -c netmalloc
```

คำสั่ง นี้ยังเปิดใช้การบันทึก outstanding memory allocation (OSTD) สำหรับหน่วยความจำเน็ตเวิร์กด้วย

- เมื่อต้องการเปลี่ยนความถี่ของการจัดสรรการควบคุม `netmalloc` และเหตุการณ์ว่าง เป็น 25%, เปลี่ยนความน่าจะเป็นของฟังก์ชัน `police_frequency` เป็น 256 ให้ป้อน:

```
errctrl police_frequency=256 -c netmalloc.police
```

## Location

`/usr/sbin/errctrl`

## ไฟล์

ไอเท็ม

`/var/adm/ras/rasptune`

คำอธิบาย

ไฟล์ที่มีการกำหนดค่าแอตทริบิวต์ตัวเอง ที่จะนำใช้หลังการรีสตาร์ท ถ้าคุณรันคำสั่ง `bosboot` ก่อน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dumpctrl`” ในหน้า 287

ข้อมูลที่เกี่ยวข้อง:

`ras_register` และ `ras_unregister`

คำสั่ง `ras_control`

คำสั่ง `/var/adm/ras/rasptune`

---

## คำสั่ง `errdead`

### วัตถุประสงค์

แยกเร็กคอร์ดข้อผิดพลาดจากดัมพ์ระบบหรือดัมพ์ live

### ไวยากรณ์

```
/usr/lib/errdead [ -i FileName ] DumpFile [UnixFile]
```

### คำอธิบาย

คำสั่ง `errdead` แยกเร็กคอร์ดข้อผิดพลาดจาก ดัมพ์ระบบหรือดัมพ์ live ที่มีบัพเฟอร์ภายในที่ดูแลโดยไฟล์ `/dev/error` คำสั่ง `errdead` แยกเร็กคอร์ดข้อผิดพลาดจากดัมพ์ไฟล์และเพิ่มข้อผิดพลาดเหล่านั้น ลงในบันทึกข้อผิดพลาดโดยตรง

`error log daemon` ไม่จำเป็นต้องรันอยู่เมื่อคำสั่ง `errdead` ถูกรัน

## แฟล็ก

ไอเท็ม -iFileName	คำอธิบาย เพิ่มเร็กคอร์ดข้อผิดพลาดที่แยกมา ไปที่ไฟล์บันทึกข้อผิดพลาดที่ระบุ โดยตัวแปร <i>FileName</i> ถ้าไฟล์ไม่มีอยู่ คำสั่ง <code>errdead</code> จะสร้างขึ้นมา ถ้าไม่ระบุแฟล็กนี้ ค่าจากฐานข้อมูลคอนฟิกเรชันบันทึกข้อผิดพลาด จะถูกนำมาใช้
----------------------	---

## พารามิเตอร์

ไอเท็ม <i>DumpFile</i> <i>UnixFile</i>	คำอธิบาย ระบุตั้มพ้อมเมจให้เปิดทำงาน ระบุไฟล์ UNIX ที่ถูกใช้เมื่อ รับข้อมูล ตั้มพ้อมระบบหรือตั้มพ้อม live ไปนี้ไม่จำเป็นถ้าใช้คำสั่ง <code>errdead</code> บนระบบเดียวกันที่ตั้มพ้อมถูกสร้างขึ้น
--	---

## ความปลอดภัย

การควบคุมสิทธิ์เข้าถึง: เฉพาะผู้ใช้ `root` เท่านั้นที่สามารถรัน คำสั่งนี้ได้

## ตัวอย่าง

เมื่อต้องการรับข้อมูลบันทึกข้อผิดพลาดจากตั้มพ้อมเมจที่อยู่ในไฟล์ `/var/adm/ras/vmcore.0` ให้ป้อน:

```
/usr/lib/errdead /var/adm/ras/vmcore.0
```

ข้อมูลการบันทึกข้อผิดพลาดอยู่ในตั้มพ้อมเมจ ถ้า `errdemon` daemon ไม่ได้รันเมื่อมีการตั้มพ้อมเกิดขึ้น

## File

ไอเท็ม <code>/etc/objrepos/SWservAt</code>	คำอธิบาย มีคลาสอ็อบเจกต์แอ็ททริบิวต์การช่วยเหลือเซอร์วิสซอฟต์แวร์; ซึ่ง คือ ฐานข้อมูลคอนฟิกเรชันบันทึกข้อผิดพลาด
---	---

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `errclear`” ในหน้า 421

“คำสั่ง `errinstall`” ในหน้า 432

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `errlog`

ภาพรวมล็อกข้อผิดพลาด

---

## errdemon Daemon

### วัตถุประสงค์

เริ่ม daemon (`errdemon`) การบันทึกข้อผิดพลาดและเขียนรายการไปที่บันทึกข้อผิดพลาด

## ไวยากรณ์

```
errdemon [ [ -B BufferSize ] [ -d | -D ][ -i File ] [ -s LogSize ] [ -t Time ][ -m MaxDups ]  
| -l ]
```

## คำอธิบาย

daemon การบันทึกข้อผิดพลาดอ่านเร็กคอร์ดจากไฟล์ /dev/error และสร้างรายการบันทึกข้อผิดพลาดในบันทึกข้อผิดพลาดระบบ นอกจากการเขียนรายการไปที่บันทึกข้อผิดพลาดระบบในแต่ละครั้งที่ข้อผิดพลาดถูกบันทึก daemon การบันทึกข้อผิดพลาดทำการแจ้งเตือนข้อผิดพลาดตามที่ระบุในฐานข้อมูลการแจ้งเตือนข้อผิดพลาด ไฟล์ /etc/objrepos/errnotify คือฐานข้อมูลการแจ้งเตือนข้อผิดพลาด บันทึกข้อผิดพลาดระบบดีฟอลต์ ถูกดูแลในไฟล์ /var/adm/ras/errlog รายการข้อผิดพลาดล่าสุดถูกกำหนดไว้ใน nonvolatile random access memory (NVRAM) ระหว่างที่ระบบเริ่มทำงาน รายการข้อผิดพลาดล่าสุดนี้ถูกอ่านจาก NVRAM และเพิ่มลงใน บันทึกข้อผิดพลาด เมื่อ daemon การบันทึกข้อผิดพลาดถูกเริ่ม

Daemon การลือกข้อผิดพลาดไม่สร้างรายการ ลือกข้อผิดพลาดสำหรับข้อผิดพลาดที่ลือกถ้าเพิ่มเพลตเร็กคอร์ดข้อผิดพลาด ระบบ Log=FALSE

ถ้าคุณใช้ daemon การบันทึกข้อผิดพลาดโดยไม่มีแฟล็ก ระบบจะเริ่มงาน daemon การบันทึกข้อผิดพลาดโดยใช้ค่าคอนฟิกูเรชัน ที่เก็บใน ฐานข้อมูลคอนฟิกูเรชันบันทึกข้อผิดพลาด โดยดีฟอลต์ errdemon daemon ลบรายการบันทึกข้อผิดพลาด ที่ซ้ำออก เมื่อรายการถูกบันทึกเร็วมากๆ เพื่อป้องกัน การบันทึกข้อผิดพลาดไม่ให้มีผลกระทบร้ายแรงกับผลการทำงานของระบบ จำนวน ของรายการซ้ำ สามารถดูได้ด้วยรายงานความผิดพลาดโดยละเอียด

ถ้าเปิดใช้งานบันทึกข้อผิดพลาด PowerHA® pureScale® รายการบันทึกข้อผิดพลาดจะถูกส่งไปยัง PowerHA pureScale logstream นอกเหนือจากบันทึกข้อผิดพลาดระบบโลคัล สถานะการลือกข้อผิดพลาด PowerHA pureScale และชื่อ logstream ถูกระบุด้วยค่า errlg\_pscale\_enabled และ errlg\_pscale\_logstream ของ ฐานข้อมูลคอนฟิกูเรชันลือกข้อผิดพลาด ชุดไฟล์โคลเอ็นต์ PowerHA pureScale ต้องติดตั้งไว้บนระบบ และการเชื่อมข้อมูลสำหรับเซอร์วิสที่ชื่อ CentralizedLogService ต้องถูกติดตั้งไว้ พื้นที่บันทึกและอ็อบเจ็กต์สตรีมบันทึกที่ระบุเป็น PowerHA pureScale logstream ต้องมีอยู่

ใช้คำสั่ง errclear เพื่อลบ รายการจากบันทึกข้อผิดพลาดระบบ

**ข้อควรสนใจ:** Daemon การลือกข้อผิดพลาดโดยปกติเริ่มทำงาน ระหว่างการกำหนดค่าเริ่มต้นระบบ การหยุด daemon การบันทึกข้อผิดพลาดทำให้ข้อมูลข้อผิดพลาดชั่วคราวที่เก็บในบัฟเฟอร์ ภายในถูกเขียนทับ ก่อนที่จะถูกบันทึกลงในไฟล์บันทึกข้อผิดพลาด

# แฟล็ก

## ไอเท็ม

-B BufferSize

### คำอธิบาย

ใช้จำนวนไบต์ที่ระบุโดยพารามิเตอร์ *BufferSize* สำหรับบัฟเฟอร์ในหน่วยความจำของไดเรกทอรีอุปกรณ์บันทึกข้อมูลผิดพลาด ขนาดบัฟเฟอร์ที่ระบุถูกบันทึกในฐานข้อมูลคอนฟิกูเรชันบันทึกข้อมูลผิดพลาด ถ้าพารามิเตอร์ *BufferSize* มีขนาดใหญ่กว่าขนาดบัฟเฟอร์ที่ใช้อยู่ปัจจุบัน บัฟเฟอร์ในหน่วยความจำจะถูกเพิ่มขนาดทันที ถ้าพารามิเตอร์ *BufferSize* มีขนาดเล็กกว่าขนาดบัฟเฟอร์ที่ใช้อยู่ปัจจุบัน ขนาดใหม่จะมีผลในครั้งถัดไปที่ daemon การบันทึกข้อมูลผิดพลาด ถูกเริ่มทำงานหลังจากรีบูตระบบ บัฟเฟอร์ไม่สามารถมีขนาดเล็กกว่าดีฟอลต์ที่ฮาร์ดโค้ดไว้คือ 8 KB

ถ้าพารามิเตอร์นี้ไม่ถูก ระบุ daemon การบันทึกข้อมูลผิดพลาดจะใช้ขนาดบัฟเฟอร์จาก ฐานข้อมูลคอนฟิกูเรชันบันทึกข้อมูลผิดพลาด

ขนาดที่ระบุถูกปรับเพิ่มขึ้นเป็นจำนวนเต็มผลคูณของขนาดเพจหน่วยความจำ (4 KB) หน่วยความจำที่ใช้สำหรับบัฟเฟอร์ในหน่วยความจำของไดเรกทอรีอุปกรณ์บันทึกข้อมูลผิดพลาด ไม่พร้อมใช้งานโดยการประมวลผลอื่นๆ (บัฟเฟอร์ถูกตรึง) ขอให้ระมัดระวังมิให้ส่งผลต่อผล การทำงานของระบบของคุณด้วยการกำหนดบัฟเฟอร์มีขนาดใหญ่เกินไป แต่คุณกำหนดให้บัฟเฟอร์มีขนาดเล็กเกินไป บัฟเฟอร์สามารถเต็มได้ รายการขอผิดพลาดมีเข้ามาเร็วกว่าที่ถูกอ่านจากบัฟเฟอร์ และนำไปไว้ในไฟล์บันทึก เมื่อบัฟเฟอร์ เต็ม รายการใหม่จะถูกละเว้นจนกว่าจะมีพื้นที่เหลือในบัฟเฟอร์ เมื่อสถานการณ์นี้เกิดขึ้น daemon การบันทึกข้อมูลผิดพลาด จะสร้างรายการบันทึกข้อมูลผิดพลาดเพื่อแจ้งคุณถึงปัญหา คุณสามารถแก้ไขปัญหานี้โดยการขยายขนาดบัฟเฟอร์

-d

-D

-i File

-l

-m MaxDups

-s LogSize

ระบุรายการบันทึกข้อมูลผิดพลาดที่เข้าไม่สามารถถูกลบได้ การทำงาน ดีฟอลต์คือลบข้อมูลซ้ำ ซึ่งถูกระบุด้วยแฟล็ก -D

ระบุรายการซ้ำจะถูกลบ นี่คือน คัดฟอลต์

ใช้ไฟล์บันทึกข้อมูลผิดพลาดที่ระบุโดยตัวแปร *File* ชื่อไฟล์ที่ระบุถูกบันทึกในฐานข้อมูลการตั้งค่า บันทึกข้อมูลผิดพลาด และถูกนำไปใช้ทันที

แสดงค่าสำหรับชื่อไฟล์บันทึกข้อมูลผิดพลาด ขนาดไฟล์ ขนาดบัฟเฟอร์ และค่าการจัดการข้อมูลซ้ำ จากฐานข้อมูลคอนฟิกูเรชันบันทึกข้อมูลผิดพลาด

ระบุจำนวนรายการซ้ำสูงสุดที่มีได้ ก่อนที่ ข้อมูลผิดพลาดที่ซ้ำกันจะถูกเอาออกไป ค่าดีฟอลต์คือ 1000 เมื่อข้อมูลผิดพลาดซ้ำกันเท่ากับจำนวนครั้งที่ถูกระบุใน *MaxDups*, ข้อมูลผิดพลาดที่ซ้ำจะถูกเขียนเหมือนเป็นข้อมูลผิดพลาดรายการเดียว ที่ถูกบันทึก ค่าที่ใช้ได้สำหรับ *MaxDups* คือ 1 ถึง 2147483647

ใช้ขนาดที่ระบุโดยตัวแปร *LogSize* สำหรับขนาดสูงสุดของไฟล์บันทึกข้อมูลผิดพลาด ข้อจำกัดขนาดล็อกไฟล์ ที่ระบุถูกบันทึกในฐานข้อมูลคอนฟิกูเรชันบันทึกข้อมูลผิดพลาด และถูกนำไปใช้ทันที ถ้าข้อจำกัดขนาดล็อกไฟล์เล็กกว่าขนาดของล็อกไฟล์ที่ใช้อยู่ daemon การบันทึกข้อมูลผิดพลาดจะเปลี่ยนชื่อล็อกไฟล์บันทึกข้อมูลผิดพลาด โดยเชื่อม .old กับชื่อไฟล์ daemon การบันทึกข้อมูลผิดพลาด สร้างล็อกไฟล์ใหม่ด้วยข้อจำกัดขนาดที่ระบุ สร้าง รายงานจากล็อกไฟล์เก่าโดยใช้แฟล็ก -i ของคำสั่ง `errpt`

ถ้าพารามิเตอร์นี้ไม่ถูก ระบุ daemon การบันทึกข้อมูลผิดพลาดจะใช้ขนาดล็อกไฟล์จาก ฐานข้อมูลคอนฟิกูเรชันบันทึกข้อมูลผิดพลาด

-t Time

ระบุช่วงเวลาโดยประมาณ (เป็นมิลลิวินาที) ภายในช่วงเวลา ที่คิดว่าข้อมูลผิดพลาดซ้ำเกิดขึ้น ถ้าเหมือนกับข้อมูลผิดพลาดก่อนหน้านี้ ข้อมูลผิดพลาดที่เกิดขึ้นหลังจากช่วงเวลานี้ไม่ถือว่าเป็นข้อมูลซ้ำ แม้ว่าจะเหมือนกับข้อมูลผิดพลาดก่อนหน้านี้ ช่วงเวลาดีฟอลต์คือ 10000 หรือ 10 วินาที ค่าที่ใช้ได้สำหรับ *Time* คือ 1 ถึง 2147483647

หมายเหตุ: แฟล็กนี้จัดรายการซ้ำในกรณี ของตัวบันทึกข้อมูลผิดพลาดบันทึกข้อมูลผิดพลาดเหมือนกันอย่างรวดเร็ว ซึ่งโดยปกติแสดง ถึงสภาวะการวนซ้ำ โดยไม่ได้ถูกกำหนดให้เก็บข้อมูลซ้ำทั้งหมด ซึ่งอาจเป็นอับเจกต์การแจ้งเตือนข้อมูลผิดพลาด การกำหนดค่านี้ใหม่เกินไป อาจสร้างความเสียหายต่อการแจ้งเตือนข้อมูลผิดพลาด โดยการลบข้อมูลผิดพลาดที่มากเกินไป ดูที่คำสั่ง `errpt` สำหรับรายละเอียด ของการลบข้อมูลผิดพลาดซ้ำในรายงานความผิดพลาด

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: เฉพาะผู้ใช้ root เท่านั้นที่สามารถรัน daemon นี้ได้

## ตัวอย่าง

1. เมื่อต้องการสตาร์ท daemon การบันทึกข้อมูลผิดพลาด ให้ป้อน:

```
/usr/lib/errdemom
```

2. เมื่อต้องการดูขนาดบันทึกข้อมูลผิดพลาดสูงสุดปัจจุบัน ให้ป้อน:

```
/usr/lib/errdemon -l
```

3. เมื่อต้องการเปลี่ยนขนาดบันทึกข้อผิดพลาดสูงสุดจาก 1MB เป็น 64KB ให้ป้อน:

```
/usr/lib/errdemon -s 65536
```

4. เมื่อต้องการพิจารณาข้อผิดพลาดที่ถูกบันทึกภายใน 10 มิลลิวินาทีล่าสุดว่าซ้ำซ้อนหรือไม่ให้ป้อน

```
/usr/lib/errdemon -t 10
```

## ไฟล์

### ไอเท็ม

```
/dev/error  
/var/adm/ras/errtmplt  
/usr/lib/errdemon  
/etc/objrepos/SWservAt
```

### คำอธิบาย

ซอร์สของเร็กคอร์ดข้อผิดพลาด  
มีที่เก็บเพิ่มเติมเพื่อบันทึกข้อผิดพลาด  
มี `errdemon` daemon  
มีคลาสอ็อบเจกต์แอ็ดทริบิวต์การช่วยเหลือเซอร์วิสซอฟต์แวร์; ซึ่ง คือ ฐานข้อมูลคอนฟิกูเรชัน  
บันทึกข้อผิดพลาด

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `errpt`” ในหน้า 438

### ข้อมูลที่เกี่ยวข้อง:

การบันทึกข้อผิดพลาด

รูทีนย่อย `errlog`

ภาพรวมล็อกข้อผิดพลาด

---

## คำสั่ง `errinstall`

### วัตถุประสงค์

ติดตั้งข้อความในเซตข้อความการบันทึกข้อผิดพลาด

### ไวยากรณ์

```
errinstall [-c] [-f] [-q] [-z FileName] File
```

### คำอธิบาย

คำสั่ง `errinstall` เป็นส่วนช่วยเหลือ การติดตั้งที่เพิ่มหรือแทนที่ข้อความในเซต id ข้อความ Error คำอธิบาย, Probable Cause, User Cause, Install Cause, Failure Cause, Recommended Action, และ Detailed Data ของแค็ตตาล็อกข้อความบันทึกข้อผิดพลาด

พารามิเตอร์ *File* ระบุ อินพุตไฟล์ที่มีข้อความที่จะถูกเพิ่มหรือแทนที่ ถ้าคุณไม่ระบุพารามิเตอร์ *File* หรือถ้าคุณระบุเป็น - (เครื่องหมายลบ) คำสั่ง `errinstall` อ่านข้อมูลจากอินพุตมาตรฐาน

หมายเหตุ: โปรแกรมและแอปพลิเคชัน in-house ที่ได้รับไลเซนส์ ต้องใช้ข้อความที่กำหนดไว้ล่วงหน้าจากชุดข้อความการบันทึกข้อผิดพลาด แสดงข้อความที่กำหนดไว้ก่อนโดยใช้คำสั่ง `errmsg -w` เมื่อต้องการเพิ่มข้อความใหม่ คู่ค้าซอฟต์แวร์

third-party ควร ติดต่อ IBM Developer Solutions เพื่อรีจิสเตอร์ข้อความใหม่ ระหว่าง การพัฒนาแอปพลิเคชัน in-house คำสั่ง `errmsg` สามารถใช้เพื่อเพิ่มข้อความ แต่ข้อความใหม่ต้องไม่ขัดแย้งกับ ข้อความที่เพิ่มสำหรับแอปพลิเคชัน in-house อื่น

### คุณลักษณะการเลิกทำ

คำสั่ง `errinstall` สร้างไฟล์ เลิกทำในไดเรกทอรีปัจจุบันไฟล์ชื่อ `File.undo` (ถ้าคำสั่ง `errinstall` อ่านจากอินพุตมาตรฐาน ข้อมูลไฟล์เลิกทำจะถูกเขียนไปที่เอาต์พุตมาตรฐาน) ไฟล์ `File.undo` สามารถใช้เป็นอินพุตกับคำสั่ง `errinstall` เพื่อยกเลิก การเปลี่ยนแปลงที่คำสั่ง `errinstall` เพิ่งทำไป เมื่อต้องการ ยกเลิกการเปลี่ยนแปลง ให้รันคำสั่ง `errinstall` ด้วยแฟล็ก `-f` และระบุไฟล์ `File.undo` สำหรับพารามิเตอร์ `File`

### รูปแบบไฟล์อินพุตไฟล์ (หรือ Standard In)

สองบรรทัดข้อมูลที่แยกกัน จำเป็นในการเพิ่ม หรือแทนที่ ข้อความเดียวในแค็ตตาล็อกข้อความบันทึกข้อผิดพลาด คุณ สามารถรวม หลาย การเพิ่มเติมหรือการแทนที่ในไฟล์เดียว บรรทัดแรกจำเป็น ในการระบุเซตข้อความ ซึ่งข้อความถูกเพิ่ม หรือแทนที่ ใช้รูปแบบต่อไปนี้:

```
SET MessageSetID
```

โดยที่พารามิเตอร์ `MessageSetID` คือหนึ่งในอักขระเดี่ยวต่อไปนี้:

ไอเท็ม	คำอธิบาย
E	ระบุ Error คำอธิบาย
P	ระบุ Probable Cause
U	ระบุ User Cause
I	ระบุ Install Cause
F	ระบุ Failure Cause
R	ระบุ Recommended Action
D	ระบุ Detailed Data

บรรทัดที่สองแสดง message ID พร้อมกับข้อความ ที่จะถูกเพิ่มหรือแทนที่ ต้องการอย่างน้อยหนึ่งบรรทัด และบรรทัดหลาย บรรทัด สามารถถูกรวมได้ ต่อจากบรรทัดเดียวที่ระบุเซตข้อความ ตาม ที่อธิบายเบื้องต้น ผู้ใช้ควรติดต่อเจ้าหน้าที่เซอร์วิสเพื่อ รับ message ID นอกจากนี้จะเป็นสำหรับแอปพลิเคชัน in-house เท่านั้น (ใน กรณีดังกล่าว ใช้คำสั่ง `errmsg` เพื่อติดตั้ง ข้อความ แสดงความผิดพลาดโดยไม่มี error message ID ที่กำหนดไว้ก่อน)

คุณต้องใส่ช่องว่างไว้ระหว่าง message ID และ ข้อความเมสเสจ และปิดล้อมข้อความของเมสเสจในอัญประกาศดังนี้:

```
message ID "message text"
```

นอกจากสองบรรทัดข้อมูลที่จำเป็น คุณยังสามารถรวมบรรทัดความคิดเห็นได้ บรรทัดหมายเหตุต้องมีโอเปอเรเตอร์ \$ (เครื่องหมาย ดอลลาร์) หรือ \* (เครื่องหมายดอกจัน) ในคอลัมน์แรก เครื่องหมายดอกจันเป็น อีพซันตามต้องการ

**หมายเหตุ:** ข้อความที่เพิ่ม ให้กับเซตข้อความ Error คำอธิบาย, Probable Cause และ Detailed Data ID ต้องมีความยาว ไม่เกิน 40 อักขระ ข้อความที่เพิ่ม ให้กับเซตข้อความ User Cause, Install Cause, Failure Cause, และ Recommended Action ต้องมีความยาวไม่ เกิน 128 อักขระ ถ้าข้อความเกินความยาวเหล่านี้ คำสั่ง `errinstall` จะแสดงข้อความเตือน แต่ เพิ่ม ข้อความให้กับแค็ตตาล็อก codepoint ข้อความเหล่านี้จะถูกตัด เมื่อแสดงโดยคำสั่ง `summary errpt`

## แฟล็ก

ไอเท็ม	คำอธิบาย
-c	ตรวจสอบพารามิเตอร์ <i>File</i> อินพุตเพื่อหาข้อผิดพลาดทางไวยากรณ์
-f	แทนที่ข้อความที่มี ID ซ้ำ เมื่อมีความพยายามในการเพิ่มข้อความโดยใช้ message ID ที่ถูกใช้อยู่แล้ว แฟล็ก -f บังคับให้คำสั่ง <b>errinstall</b> แทนที่ข้อความเมสเสจเก่าด้วยข้อความเมสเสจใหม่ ถ้าไม่ระบุแฟล็ก -f นี้ ข้อความเมสเสจเก่าจะไม่ถูกแทนที่และข้อความเตือนจะถูกเขียนไปที่ข้อผิดพลาดมาตรฐาน แฟล็ก -f จำเป็นเช่นกันในการยกเลิกการติดตั้งข้อความที่ยังการสร้างไฟล์ <b>undo</b>
-q	
-z <i>FileName</i>	ใช้แค่ดวล็อกข้อความการบันทึกข้อผิดพลาดที่ระบุโดยพารามิเตอร์ <i>FileName</i>

## ความปลอดภัย

การควบคุมสิทธิ์เข้าถึง: เฉพาะผู้ใช้ root เท่านั้นที่สามารถรัน คำสั่งนี้ได้

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

- เมื่อต้องการติดตั้งข้อความบันทึกข้อผิดพลาดสำหรับ ผลิตภัณฑ์ lpp ที่ได้รับไลเซนส์ให้ป้อน:  

```
errinstall -f /tmp/lpp.desc
```
- เมื่อต้องการยกเลิกการเปลี่ยนแปลงที่ทำกับแค็ตตาล็อกข้อความบันทึกข้อผิดพลาดโดยตัวอย่างด้านบนของคำสั่ง **errinstall** ให้ป้อน:  

```
errinstall -f /tmp/lpp.desc.undo
```
- เมื่อต้องการติดตั้งข้อความแสดงความผิดพลาดในเซตข้อความ Probable Cause ให้ป้อน:  

```
errinstall
```

\* Add a probable cause for widget failure:  
SET P  
E100 "widget adapter"
- เมื่อต้องการแทนที่ข้อความที่มี ID ซ้ำในข้อความ Probable Cause ให้ป้อน:  

```
errinstall -f
```

\* Replace the message associated with ID E100 in the  
\* Recommended Action message set  
SET R  
E100 "Replace disk drive"
- ถ้าคุณตั้งชื่ออินพุตไฟล์ **in\_file** แล้วต้องการให้ใช้เพื่อติดตั้งข้อความแสดงความผิดพลาดใหม่ให้ป้อน:  

```
errinstall in_file
```
- เมื่อต้องการเขียนทับข้อความแสดงความผิดพลาดในเซตข้อความ ใช้หมายเลข ID ที่กำหนดไว้ก่อนหน้านี้ใน **in\_file** ของคุณ และระบุแฟล็ก -f ด้วยคำสั่ง **errinstall** ดังต่อไปนี้:  

```
errinstall -f in_file
```
- ตัวอย่างต่อไปนี้แสดงเนื้อหาตัวอย่าง ของอินพุตไฟล์ที่จะถูกติดตั้ง

```

*
* Add these error messages to the Detailed Data message set:
*
SET D
8105 "Logical channel number"
8106 "Timer reference stamp"
*
* Add these error messages to the Probable Cause message set:
*
SET P
E861 "Bad memory card"
E865 "Unexpected System Halt"
E876 "Fiber Optic Cable"
*
* Add this message to the Recommended Action message set:
*
SET R
E850 "Install updated driver code"

```

## ไฟล์

### ไอเท็ม

/usr/lib/nls/msg/\$LANG/codepoint.cat

### คำอธิบาย

มีแค่ด้าล็อกข้อความบันทึกข้อผิดพลาด ในสหรัฐอเมริกาของตัวแปรสถานะแวดล้อม \$LANG คือ En\_US

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“errdemon Daemon” ในหน้า 429

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง errsav

คำสั่ง errlog

การบันทึกข้อผิดพลาด

---

## คำสั่ง **errlogger**

### วัตถุประสงค์

บันทึกข้อความโอเปอเรเตอร์

### ไวยากรณ์

**errlogger** Message

### คำอธิบาย

คำสั่ง **errlogger** สร้าง รายการบันทึกข้อผิดพลาดของโอเปอเรเตอร์ที่มีข้อความโอเปอเรเตอร์ที่ยาวได้ถึง 1024 ไบต์

## ความปลอดภัย

การควบคุมสิทธิ์เข้าถึง: เฉพาะผู้ใช้ root เท่านั้นที่สามารถรัน คำสั่งนี้ได้

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

เมื่อต้องการสร้างข้อความโอเปอเรเตอร์สำหรับการตั้งค่าไตรฟ์ระบบใหม่ให้ป้อน:

ไตรฟ์ระบบ errlogger ที่ตั้งค่าใหม่

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง errpt” ในหน้า 438

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง errsav

คำสั่ง errlog

ภาพรวมล็อกข้อผิดพลาด

---

## คำสั่ง errmsg

### วัตถุประสงค์

เพิ่มข้อความไปที่แค็ตตาล็อกข้อความบันทึกข้อผิดพลาด

### ไวยากรณ์

```
errmsg[ -c ][ -z FileName ][ -w Set_List | File ]
```

### คำอธิบาย

คำสั่ง **errmsg** อัปเดตและ แสดงแค็ตตาล็อกข้อความบันทึกข้อผิดพลาดที่มีในเซตข้อความ Error คำอธิบาย, Probable Cause, User Cause, Install Cause, Failure Cause, Recommended Action และ Detailed Data ID

เซตข้อความซึ่งจะเพิ่มหรือลบข้อความ ถูกแสดงอยู่ในพารามิเตอร์ *File* อินพุต ดังนี้:

ไอเท็ม	คำอธิบาย
* or \$	บรรทัดหมายเหตุต้องมี * (เครื่องหมายดอกจัน) หรือโอเปอเรเตอร์หมายเหตุ \$ (เครื่องหมายดอลลาร์) ใน คอลัมน์แรก * เป็นอ็อปชันตามต้องการ
+	ข้อความที่จะถูกเพิ่มต้องถูกนำหน้าด้วย + (เครื่องหมายบวก)
-	ข้อความที่จะถูกลบต้องถูกนำหน้าด้วย - (เครื่องหมายลบ)
SET	ID เชื่อมข้อความ
"Message Text"	ข้อความเมสเสจต้องถูกปิดในเครื่องหมายคำพูด
Message ID	Message ID ของข้อความที่จะถูกลบ

ข้อความที่เพิ่มให้กับเซตข้อความ Error คำอธิบาย, Probable Cause และ Detailed Data ID ต้องมีความยาวไม่เกิน 40 อักขระ  
 ข้อความที่เพิ่มให้กับเซตข้อความ User Cause, Install Cause, Failure Cause, และ Recommended Action ต้องมีความยาวไม่เกิน 128 อักขระ ข้อความผู้ใช้กำหนดเองสูงสุด 2047 ข้อความสามารถถูกเพิ่มให้กับแต่ละเซตข้อความ

คำสั่ง `errmsg` ถูกใช้โดย ผู้พัฒนาแอปพลิเคชันเพื่อสร้างข้อความใหม่ที่ใช้ใน Error Record Templates Repository ข้อความที่มีอยู่ควรถูกใช้เสมอ ถ้าเป็นไปได้

ถ้าไม่มีการระบุแฟล็กบนบรรทัดคำสั่ง การดำเนินการดีฟอลต์จะเป็นอัปเดต อัปเดตถูกระบุในพารามิเตอร์ `File` อินพุต ถ้าพารามิเตอร์ `File` อินพุต ไม่ถูกระบุหรือถ้ามีการระบุ - (เครื่องหมายลบ) แทนพารามิเตอร์ `File` คำสั่ง `errmsg` อ่านจากอินพุตมาตรฐาน สำหรับแต่ละข้อความที่ถูกเพิ่ม คำสั่ง `errmsg` จะกำหนดเป็น identifier นอกจากนี้ การเพิ่มข้อความให้กับแค็ตตาล็อกข้อความ คำสั่ง `errmsg` เขียน identifier และข้อความเมสเสจไปที่ไฟล์ `File.out` ไฟล์ `File.out` ถูกสร้างเช่นกัน เมื่อมีการลบจากแค็ตตาล็อกข้อความ ถ้าคำสั่ง `errmsg` ถูกอ่านจาก อินพุตมาตรฐาน identifier และข้อความเมสเสจจะถูกเขียนไปที่เอาต์พุตมาตรฐาน

## แฟล็ก

ไอเท็ม	คำอธิบาย
-c	ตรวจสอบอินพุตไฟล์เพื่อหาข้อผิดพลาดทางไวยากรณ์
-w <code>Set_List</code>	แสดงเซตบันทึกข้อผิดพลาดที่ระบุโดยตัวแปร <code>Set_List</code> อ็อปชันนี้แสดงข้อความที่มีในเซตข้อความ Error Log และ identifiers เอาต์พุตถูกเขียนไปที่เอาต์พุต มาตรฐาน ตัวแปร <code>Set_List</code> สามารถถูกแยกโดย คอมมาหรือปิดด้วยเครื่องหมายคำพูดและแยกด้วยคอมมาหรือช่องว่าง ตัวแปร <code>Set_List</code> คือ ID ชุดข้อความ หรือถ้าค่า ของตัวแปร <code>Set_List</code> คือ all ถูกระบุ เนื้อหาของชุดข้อความ ข้อผิดพลาดทั้งหมดจะถูกแสดง ค่าที่ใช้ได้ของ <code>Set_List</code> ตัวแปร คือ:
ทั้งหมด	แสดงเซตข้อความทั้งหมด
D	แสดงเซตข้อความ Detailed Data ID
E	เซตข้อความ Error คำอธิบาย
F	เซตข้อความ Failure Cause
I	แสดงเซตข้อความ Install Cause
P	แสดงเซตข้อความ Probable Cause
R	แสดงเซตข้อความ Recommended Action
U	แสดงแสดงเซตข้อความ User Cause
-z <code>Filename</code>	ใช้แค็ตตาล็อกข้อความการบันทึกข้อผิดพลาดที่ระบุโดยตัวแปร <code>Filename</code>

## ความปลอดภัย

การควบคุมสิทธิ์เข้าถึง: เฉพาะผู้ใช้ root เท่านั้นที่สามารถรัน คำสั่งนี้ได้

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

1. เมื่อต้องการลบข้อความจากเซตข้อความ Probable Cause ให้ป้อน:

```
errmsg
* Delete messages FF1A, FF1B, and FF1C from the Probable Cause
* message set
SET P
- FF1A
- FF1B
- FF1C
```

2. เมื่อต้องการเพิ่มข้อความให้กับเซตข้อความ Probable Cause message สำหรับข้อผิดพลาด Widget Failure ให้ป้อน:

```
errmsg
* Add a Probable Cause for Widget Failure
SET P
+ "WIDGET ADAPTER"
```

## File

### ไอเท็ม

/usr/lib/nls/msg/\$LANG/codepoint.cat

### คำอธิบาย

มีแค็ตตาล็อกข้อความบันทึกข้อผิดพลาด ในสหรัฐอเมริกา  
ค่าของ \$LANG คือ en\_US

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง errlogger” ในหน้า 435

### ข้อมูลที่เกี่ยวข้อง:

รูทีนย่อย errsav

รูทีนย่อย errlog

ไฟล์พิเศษสำหรับการล็อกข้อผิดพลาด

---

## คำสั่ง errpt

### วัตถุประสงค์

สร้างรายงานของข้อผิดพลาดที่บันทึกไว้

### ไวยากรณ์

เมื่อต้องการประมวลผลรายงานจากบันทึกข้อผิดพลาด

```
errpt [-@wpar_name] [-a] [-A] [-c] [-dErrorClassList] [-D] [-eEndDate] [-g] [-iFile]
[-IFile] [-jErrorID[,ErrorID]] [-kErrorID[,ErrorID]] [-JErrorLabel[,ErrorLabel]] [-K
```

```
ErrorLabel[ ,ErrorLabel ] ][ -ISequenceNumber ] [ -mMachine ] [ -nNode ] [ -sStartDate ] [ -F
FlagList ] [ -NResourceNameList ] [ -P ] [ -RResourceTypeList ] [ -SResourceClassList ] [ -T
ErrorTypeList ] [ -yFile ] [ -zFile ]
```

เมื่อต้องการประมวลผลรายงานจากที่เก็บเพิ่มเพลตเร็กคอร์ดข้อผิดพลาด

```
errpt[ -a ] [ -A ] [ -IFile ] [ -t ] [ -dErrorClassList ] [ -jErrorID[ ,ErrorID ] ] [ -kErrorID[ ,
ErrorID ] ] [ -JErrorLabel[ ,ErrorLabel ] ] [ -KErrorLabel[ ,ErrorLabel ] ] [ -FFlagList ] [ -P ] [
-TErrorTypeList ] [ -yFile ] [ -zFile ]
```

## คำอธิบาย

คำสั่ง **errpt** สร้าง รายงานความผิดพลาดจากรายการในบันทึกข้อผิดพลาด ซึ่งรวมแฟล็กสำหรับการเลือกข้อผิดพลาดที่ตรงกับเงื่อนไขที่เจาะจง โดยการใช้สภาวะดีฟอลต์ คุณสามารถแสดงรายการบันทึกข้อผิดพลาดในลำดับย้อนกลับที่ข้อผิดพลาดเกิดขึ้นและถูกบันทึกไว้โดยการใช้แฟล็ก **-c** (concurrent) คุณสามารถแสดงข้อผิดพลาด ตามที่เกิดขึ้น ถ้าแฟล็ก **-i** ไม่ถูกใช้กับคำสั่ง **errpt** ไฟล์บันทึกข้อผิดพลาดที่ประมวลผลโดย **errpt** คือไฟล์ที่ระบุในฐานข้อมูลคอนฟิกูเรชัน บันทึกข้อผิดพลาด (เมื่อต้องการดูข้อมูลในฐานข้อมูลคอนฟิกูเรชันบันทึกข้อผิดพลาด ใช้คำสั่ง **errdemon**)

รายงานสรุปดีฟอลต์มีข้อมูลหนึ่งบรรทัด สำหรับแต่ละข้อผิดพลาด คุณสามารถใช้แฟล็กเพื่อสร้างรายงานที่มีรูปแบบต่างกัน

**หมายเหตุ:** คำสั่ง **errpt** ไม่ทำการวิเคราะห์บันทึกข้อผิดพลาด; สำหรับการวิเคราะห์ให้ใช้ คำสั่ง **diag** เมื่อมีการทำ การวิเคราะห์บันทึกข้อผิดพลาด การวินิจฉัยอาจเพิ่มข้อมูลวินิจฉัย กลับไปในบันทึกข้อผิดพลาด ข้อมูลดังกล่าวถูกแสดง ข้อมูลรายละเอียด ของรายการบันทึกข้อผิดพลาดที่ตรงกัน

คุณสามารถใช้พารามิเตอร์ **smit errpt** ของ System Management Interface Tool (SMIT) เพื่อรันคำสั่งนี้

## แฟล็ก

ไอเท็ม	คำอธิบาย
<b>-@ wpar_name</b>	เลือกรายการข้อผิดพลาดสำหรับชื่อ WPAR ที่ระบุ
<b>-a</b>	แสดงข้อมูลเกี่ยวกับข้อผิดพลาดในไฟล์บันทึกข้อผิดพลาดในรูปแบบ ที่มีรายละเอียด ถ้าใช้ร่วมกับแฟล็ก <b>-t</b> ข้อมูลทั้งหมดจากเพิ่มเพลตไฟล์จะถูกแสดง
<b>-A</b>	แสดงแบบย่อของรายงานละเอียดที่สร้างโดยแฟล็ก <b>-a</b> แฟล็ก <b>-A</b> ใช้ไม่ได้กับ <b>-a</b> , <b>-g</b> หรือแฟล็ก <b>-t</b> รายการที่รายงานคือ เลเบล, วันที่และเวลา, ชนิด, ชื่อรีซอร์ส, คำอธิบาย และข้อมูลละเอียด เอาต์พุตตัวอย่างของแฟล็กนี้อยู่ในรูปแบบต่อไปนี้:  <pre>LABEL:          STOK_RCVRY_EXIT Date/Time:      Tue Dec 14 15:25:33 Type:          TEMP Resource Name:  tok0 Description PROBLEM RESOLVED Detail Data FILE NAME line: 273 file: stok_wdt.c SENSE DATA 0000 0000 0000 0000 0000 0000 DEVICE ADDRESS 0004 AC62 25F1</pre>
<b>-c</b>	จัดรูปแบบและแสดงแต่ละรายการข้อผิดพลาดพร้อมกัน คือเวลาที่ข้อมูลดังกล่าวถูกบันทึก รายการที่มีอยู่ในล็อกไฟล์ถูกแสดงในลำดับซึ่งถูกบันทึก

## ไอเท็ม

`-d ErrorClassList`

### คำอธิบาย

จำกัดรายงานความผิดพลาดให้เป็นชนิดที่แน่นอนของเรกคอร์ดข้อผิดพลาดที่ระบุโดยตัวแปร `ErrorClassList` ที่ใช้ได้: **H** (hardware), **S** (software), **O** (ข้อความคำสั่ง `errlogger`) และ **U** (undetermined) เรกคอร์ดข้อผิดพลาดในตัวแปร `ErrorClassList` สามารถถูกแยกโดย , (คอมมา) หรือปิดด้วย " " (เครื่องหมายคำพูด) และที่แยกโดย , (คอมมา) หรืออักขระเว้นวรรค

`-D`

รวมข้อผิดพลาดที่ซ้ำกัน รายงานความผิดพลาดแบบละเอียดที่ได้รับด้วยแฟล็ก `-a`, รายงานตัวเลข และเวลาการซ้ำครั้งแรก และครั้งล่าสุด ดูที่ ภาพรวมการบันทึกข้อผิดพลาดใน *General Programming Concepts: Writing and Debugging Programs*

หมายเหตุ: แฟล็ก `-D` ไม่สามารถใช้ได้กับแฟล็ก `-c`, `-g`, `-l`, `-t` และ `-P`

`-e EndDate`

ระบุเรกคอร์ดทั้งหมดที่โพสต์ก่อน และรวมตัวแปร `EndDate` โดยที่ตัวแปร `EndDate` มีรูปแบบ `mmddhhmmyy` (เดือน วัน ชั่วโมง นาที และปี)

ไอเอ็ม  
-g

### คำอธิบาย

แสดงข้อมูล ASCII ของรายการการบันทึกข้อผิดพลาดที่ไม่ได้จัดรูปแบบ เอาต์พุตของแฟล็กนี้อยู่ในรูปแบบต่อไปนี้:

#### el\_sequence

หมายเลขประทับบันทึกข้อผิดพลาด

el\_label เลเบลข้อผิดพลาด

#### el\_timestamp

การประทับเวลาบันทึกข้อผิดพลาด

el\_crcid ตัวระบุข้อผิดพลาด cyclic-redundancy-check (CRC) เฉพาะ

#### el\_machineid

ตัวแปร Machine ID

#### el\_nodeid

ตัวแปร Node ID

el\_class คลาสข้อผิดพลาด

el\_type ชนิดข้อผิดพลาด

#### el\_resource

ชื่อรีซอร์ส

el\_rclass คลาสรีซอร์ส

el\_rtype ชนิดรีซอร์ส

#### el\_vpd\_ibm

IBM vital product data (VPD)

#### el\_vpd\_user

User VPD

el\_in ตำแหน่งโค้ดของอุปกรณ์

#### el\_connwhere

Hardware-connection ID (ตำแหน่งบนอุปกรณ์ เช่นหมายเลข สล็อต)

et\_label เลเบลข้อผิดพลาด

et\_class คลาสข้อผิดพลาด

et\_type ชนิดข้อผิดพลาด

et\_desc รายละเอียดข้อผิดพลาด

#### et\_probcauses

สาเหตุที่เป็นไปได้

#### et\_usercauses

สาเหตุจากผู้ใช้

#### et\_useraction

การผู้ใช้ของผู้ใช้

## ไอเท็ม

### คำอธิบาย

#### et\_instcauses

สาเหตุของการติดตั้ง

#### et\_instaction

การดำเนินการการติดตั้ง

#### et\_failcauses

สาเหตุความล้มเหลว

#### et\_failaction

การความล้มเหลวของความล้มเหลว

#### et\_detail\_length

ความยาวฟิลด์ Detail-data

#### et\_detail\_descid

Detail-data identifiers

#### et\_detail\_encode

รายละเอียดของรูปแบบอินพุต detail-data

#### et\_logflg

แฟล็กไฟล์บันทึก

#### et\_alertflg

แฟล็กข้อผิดพลาดที่แจ้งเตือนได้

#### et\_reportflg

แฟล็กรายงานความผิดพลาด

#### el\_detail\_length

ความยาวอินพุต Detail-data

#### el\_detail\_data

อินพุต Detail-data

### -F *FlagList*

เลือกเพิ่มเพลต error-record ตามค่าของฟิลด์ Alert, Log, หรือ Report ของเท็มเพลต ตัวแปร *FlagList* สามารถถูกแยกโดย, (คอมมา) หรือปิดด้วย " " (เครื่องหมายคำพูด) และที่แยกโดย, (คอมมา) หรืออักขระเว้นวรรค แฟล็ก -F ถูกใช้กับแฟล็ก -t เท่านั้น

ค่าที่ใช้ได้ของตัวแปร *FlagList* ประกอบด้วย:

**alert=0** เลือกเพิ่มเพลต error-record โดยมีฟิลด์ Alert เซ็ตเป็น False

**alert=1** เลือกเพิ่มเพลต error-record โดยมีฟิลด์ Alert เซ็ตเป็น True

**log=0** เลือกเพิ่มเพลต error-record โดยมีฟิลด์ Log เซ็ตเป็น False

**log=1** เลือกเพิ่มเพลต error-record โดยมีฟิลด์ Log เซ็ตเป็น True

**report=0** เลือกเพิ่มเพลต error-record โดยมีฟิลด์ Report เซ็ตเป็น False

**report=1** เลือกเพิ่มเพลต error-record โดยมีฟิลด์ Report เซ็ตเป็น True

### -i *File*

ใช้ไฟล์บันทึกข้อผิดพลาดที่ระบุโดยตัวแปร *File* ถ้าไม่ระบุแฟล็กนี้ ค่าจากฐานข้อมูลคอนฟิกูเรชันบันทึกข้อผิดพลาดจะถูกนำมาใช้

### -I *File*

ใช้ชื่อไฟล์การวินิจฉัยที่ระบุโดย *File* ถ้าไม่ถูกระบุ ชื่อพาร์ติชัน `/var/adm/ras/diag_log` จะถูกใช้

### -j *ErrorID[,ErrorID]*

รวมเฉพาะรายการบันทึกข้อผิดพลาดที่ระบุโดยตัวแปร *ErrorID* (error identifier) ตัวแปร *ErrorID* สามารถถูกแยกโดย, (คอมมา) หรือปิดด้วย " " (เครื่องหมายคำพูด) และที่แยกโดย, (คอมมา) หรืออักขระเว้นวรรค เมื่อรวมกับแฟล็ก -t รายการจะถูกประมวลผลจากที่เก็บ error-template (มีฉะนั้นรายการจะถูกประมวลผลจากที่เก็บบันทึกข้อผิดพลาด)

## ไอเท็ม

-J ErrorLabel	คำอธิบาย รวมรายการบันทึกข้อผิดพลาดที่ระบุโดยตัวแปร <i>ErrorLabel</i> ตัวแปร <i>ErrorLabel_List</i> สามารถถูกแยกโดยคอมมาหรือปิดด้วย เครื่องหมายคำพูดและแยกด้วยคอมมาหรือช่องว่าง เมื่อรวมกับแฟล็ก -t รายการจะถูกประมวลผลจากที่เก็บเพิ่มเพลดข้อผิดพลาด (มีฉะนั้นรายการ จะถูกประมวลผลจากที่เก็บบันทึกข้อผิดพลาด)
-k ErrorID[,ErrorID]	แยกรายการบันทึกข้อผิดพลาดที่ระบุโดยตัวแปร <i>ErrorID</i> ตัวแปร <i>ErrorID</i> สามารถถูกแยกโดย , (คอมมา) หรือปิดด้วย " " (เครื่องหมายคำพูด) และที่แยกโดย , (คอมมา) หรืออักขระเว้นวรรค เมื่อรวมกับแฟล็ก -t รายการจะถูกประมวลผลจากที่เก็บ error-template (มีฉะนั้นรายการ จะถูกประมวลผลจากที่เก็บบันทึกข้อผิดพลาด)
-K ErrorLabel	รวมรายการบันทึกข้อผิดพลาดสำหรับตัวแปร <i>ErrorLabel</i> ตัวแปร <i>ErrorLabel_List</i> สามารถถูกแยกโดยคอมมาหรือปิดด้วย เครื่องหมายคำพูดและแยกด้วยคอมมาหรือช่องว่าง เมื่อรวมกับแฟล็ก -t รายการจะถูกประมวลผลจากที่เก็บเพิ่มเพลดข้อผิดพลาด (มีฉะนั้นรายการ จะถูกประมวลผลจากที่เก็บบันทึกข้อผิดพลาด)
-l SequenceNumber	เลือกรายการบันทึกข้อผิดพลาดเฉพาะ ที่ระบุโดยตัวแปร <i>SequenceNumber</i> แฟล็กนี้ถูกใช้โดยเมธอดในคลาสอ็อบเจกต์ <i>error-notification</i> ตัวแปร <i>SequenceNumber</i> สามารถถูกแยกโดย , (คอมมา) หรือปิดด้วย " " (เครื่องหมายคำพูด) และที่แยกโดย , (คอมมา) หรืออักขระเว้นวรรค
-m Machine	รวมรายการบันทึกข้อผิดพลาดสำหรับตัวแปร <i>Machine</i> ที่ระบุ คำสั่ง <i>uname -m</i> ส่งกลับ ค่าตัวแปร <i>Machine</i>
-n Node	รวมรายการล็อกข้อผิดพลาดสำหรับตัวแปร <i>Node</i> ที่ระบุ คำสั่ง <i>uname -n</i> ส่งกลับ ค่าตัวแปร <i>Node</i>
-N ResourceNameList	สร้างรายงานของชื่อรีซอร์สที่ระบุโดยตัวแปร <i>ResourceNameList</i> ตัวแปร <i>ResourceNameList</i> คือรายชื่อของรีซอร์สที่ตรวจพบข้อผิดพลาด สำหรับข้อผิดพลาดซอฟต์แวร์ ตัวแปร <i>ResourceNameList</i> แสดง ชื่อของรีซอร์สที่ตรวจพบข้อผิดพลาด สำหรับข้อผิดพลาด ฮาร์ดแวร์ ข้อมูลเหล่านี้จะเป็นชื่อของคอมโพเนนต์อุปกรณ์หรือระบบ ซึ่งไม่ได้รับ ว่าคอมโพเนนต์ล้มเหลวหรือจำเป็นต้องเปลี่ยน แต่ ถูกใช้เพื่อ กำหนดโมดูลการวินิจฉัยที่เหมาะสม เพื่อนำมาใช้เพื่อวิเคราะห์ข้อผิดพลาด
-P	ชื่อของตัวแปร <i>ResourceNameList</i> สามารถถูกแยกโดย , (คอมมา) หรือปิดด้วย " " (เครื่องหมายคำพูด) และที่แยกโดย , (คอมมา) หรืออักขระเว้นวรรค แสดงเฉพาะข้อผิดพลาดซึ่งซ้ำกับข้อผิดพลาดก่อนหน้านี้ แฟล็ก -P ใช้เฉพาะกับข้อผิดพลาดซ้ำที่สร้างโดย ไดรเวอร์อุปกรณ์บันทึกข้อผิดพลาด ข้อผิดพลาดเหล่านี้เป็นข้อมูลซ้ำ ที่เกิดขึ้นภายใน ช่วงเวลาโดยประมาณที่ระบุโดยแอตทริบิวต์การบันทึกข้อผิดพลาด <i>errlg_duptime</i> ที่ควบคุมโดยแฟล็ก <i>errdemon daemon -t</i> แฟล็ก -P ใช้ไม่ได้ กับแฟล็ก -D
-R ResourceTypeList	สร้างรายงานของชนิดรีซอร์สที่ระบุโดยตัวแปร <i>ResourceTypeList</i> สำหรับข้อผิดพลาดฮาร์ดแวร์ ตัวแปร <i>ResourceTypeList</i> คือชนิดอุปกรณ์ สำหรับข้อผิดพลาดซอฟต์แวร์ จะเป็นค่า <i>LPP</i> รายการในตัวแปร <i>ResourceTypeList</i> สามารถถูกแยกโดย , (คอมมา) หรือปิดด้วย " " (เครื่องหมายคำพูด) และที่แยกโดย , (คอมมา) หรืออักขระเว้นวรรค
-s StartDate	ระบุเรีกครดทั้งหมดที่โพสเมื่อเวลา และหลังจากเวลาของ ตัวแปร <i>StartDate</i> โดยที่ตัวแปร <i>StartDate</i> มีรูปแบบ <i>mmddhhmmyy</i> (เดือน วัน ชั่วโมง นาที และปี)
-S ResourceClassList	สร้างรายงานของคลาสรีซอร์สที่ระบุโดยตัวแปร <i>ResourceClassList</i> สำหรับข้อผิดพลาดฮาร์ดแวร์ ตัวแปร <i>ResourceClassList</i> คือคลาสอุปกรณ์ คลาสรีซอร์สต้อง ถูกแยกแต่ละรายการโดย , (คอมมา) หรือปิดด้วย " " (เครื่องหมายคำพูด) และที่แยกโดย , (คอมมา) หรืออักขระเว้นวรรค
-t	ประมวลผลที่เก็บเพิ่มเพลดบันทึกข้อผิดพลาด แทนบันทึก ข้อผิดพลาด แฟล็ก -t สามารถถูกใช้เพื่อดูเพิ่มเพลดเรีกครดข้อผิดพลาดในรูปแบบรายงาน
-T ErrorTypeList	จำกัดรายงานความผิดพลาดเป็นชนิดข้อผิดพลาดที่ระบุโดยตัวแปร <i>ErrorTypeList</i> ที่ใช้ได้: <i>INFO</i> , <i>PEND</i> , <i>PERF</i> , <i>PERM</i> , <i>TEMP</i> และ <i>UNKN</i> ชนิดข้อผิดพลาด สามารถถูกแยกแต่ละรายการโดย , (คอมมา) หรือปิดด้วย " " (เครื่องหมายคำพูด) และแยกโดย , หรืออักขระเว้นวรรค
-y File	ใช้ไฟล์เพิ่มเพลดเรีกครดข้อผิดพลาดที่ระบุโดยตัวแปร <i>File</i> เมื่อรวมกับแฟล็ก -t รายการจะถูกประมวลผลจากที่เก็บเพิ่มเพลดข้อผิดพลาดที่ระบุ (มีฉะนั้นรายการ จะถูกประมวลผลจากที่เก็บบันทึกข้อผิดพลาด โดยใช้ที่เก็บเพิ่มเพลด ข้อผิดพลาดที่ระบุ)
-z File	ใช้แค็ตตาล็อกข้อความการบันทึกข้อผิดพลาดที่ระบุโดยตัวแปร <i>File</i> เมื่อรวมกับแฟล็ก -t รายการจะถูกประมวลผลจากที่เก็บเพิ่มเพลดข้อผิดพลาด (มีฉะนั้นรายการ จะถูกประมวลผลจากที่เก็บบันทึกข้อผิดพลาด)

## ตัวอย่าง

1. เมื่อต้องการแสดงรายงานสรุปแบบสมบูรณ์ให้ป้อน:

```
errpt
```

2. เมื่อต้องการแสดงรายงานรายละเอียดแบบสมบูรณ์ให้ป้อน:

```
errpt -a
```

3. เมื่อต้องการแสดงรายงานแบบละเอียดของข้อผิดพลาดทั้งหมดที่บันทึกสำหรับ identifier ข้อผิดพลาด E19E094F ให้ป้อน:

```
errpt -a -j E19E094F
```

4. เมื่อต้องการแสดงรายงานแบบละเอียดของข้อผิดพลาดทั้งหมดที่บันทึกใน 24 ชั่วโมงที่ผ่านมาให้ป้อน:

```
errpt -a -s mmdhmmmy
```

โดยที่สตริง mmdhmmmy เท่ากับ เดือน วัน ชั่วโมง นาที และปี ปัจจุบัน ลบ 24 ชั่วโมง

5. เมื่อต้องการแสดงเพิ่มเพลตเร็คคอร์ดข้อผิดพลาด ซึ่งการบันทึกถูกปิด สำหรับรายการบันทึกข้อผิดพลาดให้ป้อน:

```
errpt -t -F log=0
```

6. เมื่อต้องการดูรายการทั้งหมดจากไฟล์บันทึกข้อผิดพลาด สำหรับ /var/adm/ras/errlog.alternate ให้ป้อน:

```
errpt -i /var/adm/ras/errlog.alternate
```

7. เมื่อต้องการดูรายการฮาร์ดแวร์ทั้งหมดจากไฟล์บันทึกข้อผิดพลาด สำหรับ /var/adm/ras/errlog.alternate ให้ป้อน:

```
errpt -i /var/adm/ras/errlog.alternate -d H
```

8. เมื่อต้องการแสดงรายงานแบบละเอียดของข้อผิดพลาดทั้งหมดที่บันทึกสำหรับ เลเบลข้อผิดพลาด ERRLOG ให้ป้อน:

```
errpt -a -J ERRLOG_ON
```

9. เมื่อต้องการแสดงรายงานแบบละเอียดของข้อผิดพลาดทั้งหมดและข้อผิดพลาดซ้ำซ้อนกลุ่ม ให้ป้อน:

```
errpt -aD
```

10. เมื่อต้องการแสดงรายงานแบบละเอียดของข้อผิดพลาดทั้งหมดที่บันทึกสำหรับเลเบลข้อผิดพลาด DISK\_ERR1 and DISK\_ERR2 ระหว่างเดือนสิงหาคมให้ป้อน:

```
errpt -a -J DISK_ERR1,DISK_ERR2 -s 0801000004 -e 0831235904"
```

## ไฟล์

### ไอเท็ม

/etc/objrepos/SWservat

### คำอธิบาย

มีคลาสอ็อบเจกต์แอ็ดทริบิวต์การช่วยเหลือเซอร์วิสซอฟต์แวร์; ซึ่ง คือ ฐานข้อมูลคอนฟิกูเรชัน บันทึกข้อผิดพลาด

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง diag” ในหน้า 142

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง uname

รูทีนย่อย errlog

ตัวอย่างของรายงานความผิดพลาดละเอียด

---

## คำสั่ง errstop

## วัตถุประสงค์

ยกเลิก error logging daemon

## ไวยากรณ์

errstop

### คำอธิบาย

**ข้อควรสนใจ:** การรันคำสั่ง `errstop` ปิดใช้งาน ฟังก์ชันการวินิจฉัยและการกู้คืน โดยปกติคำสั่ง `errdemon` ถูกเริ่มต้นโดยอัตโนมัติ ระหว่างการกำหนดค่าเริ่มต้นระบบและหยุดระหว่างระบบถูกปิด บันทึกข้อผิดพลาดไม่ควรถูกหยุดการทำงานระหว่างการดำเนินการปกติ คำสั่ง `errstop` ควรถูกใช้เฉพาะระหว่างสถานการณ์พิเศษ เมื่อจำเป็นจริงๆ และผลที่ตามมาจะมีเหตุผลอย่างชัดเจน

คำสั่ง `errstop` หยุด error logging daemon ที่เริ่มต้นโดยคำสั่ง `errdemon`

### ความปลอดภัย

ค่าควบคุมการเข้าถึง: เฉพาะผู้ใช้ root เท่านั้นที่สามารถรันคำสั่งนี้ได้

### ตัวอย่าง

เมื่อต้องการหยุด `errdemon` daemon, ให้ป้อน:

```
/usr/lib/errstop
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `errdead`” ในหน้า 428

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `errsave`

คำสั่ง `errlog`

ภาพรวมสื่อข้อผิดพลาด

---

## คำสั่ง `errupdate`

### Purpose

อัปเดตที่เก็บเพิ่มเพลตเร็กคอร์ดข้อผิดพลาด

## ไวยากรณ์

```
errupdate [-c] [-f] [-h] [-n] [-p] [-q] [-y FileName] [File]
```

### รายละเอียด

คำสั่ง `errupdate` เพิ่มหรือลบ รายการในที่เก็บเพิ่มเพลตเร็กคอร์ดข้อผิดพลาด หรือแก้ไข ล็อก รายงาน หรือคุณสมบัติแจ้งเตือนของรายการที่มีอยู่ คำสั่ง `errupdate` อ่านข้อมูลจากพารามิเตอร์ `File` ที่ระบุ ถ้าไม่ระบุพารามิเตอร์ `File` คำสั่ง `errupdate` อ่านข้อมูลจากอินพุตมาตรฐานและเขียนไปที่เอาต์พุตมาตรฐาน

แต่ละรายการที่ถูกเพิ่ม ลบ หรือแก้ไข ต้อง นำหน้าด้วยโอเปอเรเตอร์โอเปอเรเตอร์ที่ใช้ได้คือ:

ไอเท็ม	คำอธิบาย
+	เพิ่มรายการ (โอเปอเรเตอร์เพิ่ม)
-	ลบรายการ (โอเปอเรเตอร์ลบ)
=	แก้ไข ล็อก รายงาน หรือคุณสมบัติแจ้งเตือนของรายการ

รายการในอินพุตไฟล์ต้องถูกแยกด้วยบรรทัดว่าง

หมายเหตุในไฟล์อินพุตสามารถถูกนำไปไว้ระหว่างแท็กเพิ่มเพลต และถูกระบุโดย \* (เครื่องหมายดอกจัน) ในคอลัมน์แรก

ถ้าข้อความ X/Open Portability Guide Issue 4 ถูกใช้ในแท็กเพิ่มเพลตข้อผิดพลาด ต้องมีการระบุ แค็ตตาล็อกข้อความ ซึ่งทำได้ด้วยบรรทัดของฟอร์ม:

```
<!*catalog-name>
```

ตัวอย่าง

```
*!mycat.cat
```

แค็ตตาล็อกที่ระบุใช้กับข้อความ XPG4 ที่พบในแท็กเพิ่มเพลตต่อมา จนกว่าจะพบ "\*" ของตัวระบุแค็ตตาล็อก นอกจากนี้ "\*" ของตัวระบุอาจถูกเขียนทับบนพื้นฐานของแท็กเพิ่มเพลตแต่ละแท็กเพิ่มเพลตที่มีคีย์เวิร์ด "catname"

นอกจากชื่อพาธเต็มไปที่แค็ตตาล็อกถูกระบุ กฎทั่วไปสำหรับการรับแค็ตตาล็อกข้อความเป็นดังนี้ ตัวอย่าง, ในตัวอย่างด้านบน mycat.cat ถือว่าอยู่ใน /usr/lib/nls/msg/%L

รายการที่จะถูกเพิ่มต้องถูกกำหนดในรูปแบบจำเพาะ ฟอร์มทั่วไปของแท็กเพิ่มเพลตเรกคอร์ดข้อผิดพลาดคือ:

Error Record Template

```
+ LABEL:
        Comment=
        Class=
        Log=
        Report=
        Alert=
        Err_Type=
        Err_Desc=
        Prob_Causes=
        User_Causes=
        User_Actions=
        Inst_Causes=
        Inst_Actions=
        Fail_Causes=
        Fail_Actions=
        Detail_Data= <data_len>, <data_id>,
        <data_encoding>
```

นอกจากนี้ชื่อแค็ตตาล็อกสำหรับข้อความ XPG4 สามารถถูกระบุด้วย:

```
catname = <catalog>
```

เพิ่มฟิลด์ที่มีข้อความ XPG4 , คีย์เวิร์ด catname, มากกว่าแปดรายการข้อมูลละเอียดจะถูกอ้างอิงเป็นเพิ่มฟิลด์ XPG4 เพิ่มฟิลด์ XPG4 ไม่สามารถแจ้งเตือนได้ และใช้ต่างจากการคำนวณ error id เล็กน้อย

ฟิลด์เพิ่มฟิลด์เรียกคอร์ตข้อผิดพลาดถูกอธิบายดังนี้:

ไอเท็ม	คำอธิบาย
Alert	ระบุรายการบันทึกข้อผิดพลาดสามารถถูกประมวลผลโดยผลิตภัณฑ์ ที่ทำงานตามข้อกำหนด SNA Generic Alert Architecture ฟิลด์ Alert สามารถถูกเซตเป็น True หรือ False ถ้าฟิลด์นี้ถูกละเว้นจาก เพิ่มฟิลด์ ค่าจะเป็นค่าดีฟอลต์คือ False ถ้าฟิลด์ Alert ถูกเซตเป็น True คำสั่ง <b>errupdate</b> จะไม่เพิ่มเพิ่มฟิลด์ นอกจากเนื้อหาของฟิลด์ Err_Desc , Inst_Actions , Fail_Cause, Fail_Actions และ Detail_Data data_id เป็นค่าที่รู้จักโดย SNA Generic Alert Architecture (ตามเอกสาร GA27-3136) ถ้ามีค่าที่ใช้ซึ่งไม่เป็นที่รู้จักของ SNA Generic Alert Architecture หรือเพิ่มฟิลด์ เป็นเพิ่มฟิลด์ XPG4 , และฟิลด์ Alert ถูกเซตเป็น True, แฟล็ก -p ต้องถูกระบุเพื่อเพิ่มหรืออัปเดตเพิ่มฟิลด์
Class	อธิบายว่าข้อผิดพลาดเกิดขึ้นในฮาร์ดแวร์หรือซอฟต์แวร์ เป็นข้อความโอเปอเรเตอร์ หรือกำหนดไม่ได้ หนึ่งในคลาส descriptors เหล่านี้ต้องถูกระบุ:  <b>H</b> แสดงว่าข้อผิดพลาดเป็นความขัดข้องของฮาร์ดแวร์ <b>O</b> แสดงว่าข้อผิดพลาดเป็นข้อความโอเปอเรเตอร์ <b>S</b> แสดงว่าข้อผิดพลาดเป็นความล้มเหลวของซอฟต์แวร์ <b>U</b> แสดงว่าข้อผิดพลาดกำหนดไม่ได้
Comment	ระบุหมายเหตุที่จะถูกรวมกับประโยคคำสั่ง <b>#define</b> ที่ถูกสร้างสำหรับเซตข้อความ Error ID หมายเหตุ ต้องไม่เกิน 40 อักขระและต้องถูกปิดอยู่ในเครื่องหมายคำพูด หมายเหตุที่ยาวกว่า 40 อักขระจะถูกตัดท้ายโดยอัตโนมัติ คำสั่ง <b>errupdate</b> ปิดหมายเหตุในตัวค้นหมายเหตุของภาษา C /* (slash, เครื่องหมายดอกจัน) และ */ (เครื่องหมายดอกจัน, slash)
Detail_Data	อธิบายข้อมูลรายละเอียด เช่น ชื่อโมดูลการตรวจจับ, ข้อมูลการรับรู้ หรือโคตส่งคืน ที่ถูกบันทึกกับข้อผิดพลาดเมื่อข้อผิดพลาดเกิดขึ้น ถ้า ไม่มีการบันทึกข้อมูลรายละเอียดพร้อมกับข้อผิดพลาด ฟิลด์นี้ปล่อยให้ว่างไว้ได้ หรือสามารถแสดงข้อความจากเซตข้อความ Detailed Data ID โดยการระบุ <b>data_len</b> ให้มีค่าเป็นศูนย์ ค่าสามค่าต่อไปนี้ จำเป็นสำหรับแต่ละฟิลด์ <b>Detail_Data</b> และต้อง ถูกแยกด้วยคอมมา:  <b>data_len</b> จำนวนไบต์ของข้อมูลที่ถูกเชื่อมโยงกับค่า <b>data_id</b> ค่า <b>data_len</b> ถูกแปล เป็นค่าฐานสิบ เมื่อต้องการระบุขนาดขึ้นกับสถานะแวดล้อมให้ใช้ "W" "W" จะถูก ปฏิบัติเป็น 8 ไบต์ ถ้าข้อผิดพลาดถูกบันทึกจากสถานะแวดล้อม 64-บิต มิฉะนั้น จะเป็น 4 ไบต์ <p style="margin-left: 40px;">หมายเหตุ: ระหว่างการคำนวณความยาวข้อมูลรายละเอียด แต่ละ "W" ถูกปฏิบัติเป็นมีความยาว 8 ไบต์ และไม่สนใจขนาดตัวพิมพ์</p> <b>data_id</b> ระบุเมสเสจข้อความจากเซตข้อความ Detailed Data ID "D" ที่จะถูกพิมพ์ในรายงานความผิดพลาด ด้านหน้าของข้อมูลรายละเอียด ค่า ถูกแปลเป็นเลขฐานสิบหกไม่มีเครื่องหมาย มีความยาวได้ถึง 4 หลัก  <b>data_encoding</b> อธิบายวิธีที่ข้อมูลรายละเอียดถูกพิมพ์ในรายงานความผิดพลาด ค่าที่ใช้ได้คือ: <b>ALPHA</b> ข้อมูลรายละเอียดเป็นสตริงอักขระ ASCII ที่พิมพ์ได้ <b>DEC</b> ข้อมูลรายละเอียดเป็นการแทนแบบฐานสองของค่าจำนวนเต็ม และค่าเทียบเท่าฐานสิบจะถูกพิมพ์ <b>LDEC</b> ข้อมูลรายละเอียดเป็นการแทนแบบฐานสองของค่า 64-บิต และค่าเทียบเท่าฐานสิบจะถูกพิมพ์ <b>HEX</b> ข้อมูลรายละเอียดจะถูกพิมพ์แบบเลขฐานสิบหก
	รายการ Detail_Data มากถึง 16 รายการที่สามารถระบุได้ต่อเพิ่มฟิลด์ จำนวนของข้อมูลที่บันทึกที่มี ข้อผิดพลาดต้องไม่เกิน ERR_REC_MAX ที่กำหนดในไฟล์ <b>/usr/include/sys/err_rec.h</b> ข้อมูลข้อผิดพลาดที่ไม่สามารถอยู่ในรายการบันทึกข้อผิดพลาดควร ถูกบันทึกที่อื่น ข้อมูลรายละเอียดในรายการบันทึกข้อผิดพลาดควรมีข้อมูลที่ ควรถูกใช้เพื่อเชื่อมโยงข้อมูลข้อผิดพลาดและรายการบันทึกข้อผิดพลาด

ไอเท็ม  
Err\_Desc

**คำอธิบาย**

อธิบายข้อผิดพลาดที่เกิดขึ้น ตัวระบุข้อความ Error Description ต้องถูกระบุในฟิลด์นี้ ค่านี้ระบุเมสเสจข้อความ จากเซตข้อความ Error Description “E” ที่จะถูกแสดงสำหรับการเกิดขึ้นของข้อผิดพลาด ค่าถูกแปลเป็นเลขฐานสิบหกไม่มีเครื่องหมาย มีความยาวได้ถึง 4 หลัก ฟิลด์นี้อาจระบุข้อความลักษณะ XPG4 ซึ่งจะกล่าวถึงในภายหลัง

Err\_Type

- PERF** สภาวะที่ผลการทำงานของอุปกรณ์หรือคอมพิวเตอร์ถูกลดลง ต่ำกว่าระดับที่ยอมรับได้ (performance)
- PERM** สภาวะที่ไม่สามารถทำการกู้คืน (permanent)
- PEND** สภาวะที่ชัดเจนว่าการสูญเสียสภาพพร้อมใช้งานของอุปกรณ์หรือคอมพิวเตอร์กำลังเกิดขึ้นแล้ว (impending)
- TEMP** สภาวะที่ถูกกู้คืนจากจำนวนการพยายามที่ไม่สำเร็จ (temporary)
- UNKN** สภาวะที่เป็นไปไม่ได้ที่จะกำหนดความรุนแรงของ ข้อผิดพลาด (unknown)
- INFO** สภาวะสำหรับรายการบันทึกข้อผิดพลาดเชิงข้อมูล

Fail\_Actions

อธิบายการดำเนินการที่แนะนำสำหรับการแก้ไขข้อผิดพลาดที่เป็นผลมาจาก สาเหตุความล้มเหลว สามารถระบุรายการของ identifier ข้อความ Recommended Action แยกกันด้วยคอมมาได้ถึง 4 รายการ ค่านี้ระบุเมสเสจข้อความจากเซตข้อความ Recommended Action “R” ที่จะถูกแสดงสำหรับการเกิดขึ้นของข้อผิดพลาด ค่า ถูกแปลเป็นเลขฐานสิบหกไม่มีเครื่องหมาย มีความยาวได้ถึงสี่หลัก ฟิลด์นี้ต่อว่าง ถ้าฟิลด์ Fail\_Causes ว่าง

Fail\_Causes

ลำดับซึ่งการดำเนินการ ที่แนะนำที่ถูกแสดง ควรถูกพิจารณาตามสูญเสียการดำเนินการ และความน่าจะเป็นที่การดำเนินการ จะแก้ไขข้อผิดพลาดได้ แสดงรายการการดำเนินการที่ไม่มีข้อเสียหรือมีไม่มาก (หรือมีน้อยหรือไม่มีผลกระทบ) บนระบบ ก่อนเสมอ แสดงการดำเนินการซึ่งความน่าจะเป็นในการแก้ไขข้อผิดพลาด เท่ากันหรือเกือบจะเท่ากันเป็นรายการถัดไป, แสดงรายการที่การดำเนินการมีผลเสีย น้อยที่สุดก่อน แสดงการดำเนินการที่เลือตามลำดับความน่าจะเป็นที่ลดลง ฟิลด์นี้อาจระบุข้อความลักษณะ XPG4 ซึ่งจะกล่าวถึงในภายหลัง

Inst\_Actions

อธิบายการดำเนินการที่แนะนำสำหรับการแก้ไขข้อผิดพลาดที่เกิดจากการติดตั้ง ฟิลด์นี้สามารถแสดงรายการได้ถึงสี่ identifier ข้อความ Recommended Action แยก ด้วยคอมมา ค่านี้ระบุเมสเสจข้อความจากเซตข้อความ Recommended Action “R” ที่จะถูกแสดงสำหรับการเกิดขึ้นของข้อผิดพลาด ค่าถูกแปลเป็นเลขฐานสิบหกไม่มีเครื่องหมาย มีความยาวได้ถึงสี่หลัก ฟิลด์นี้ต่อว่าง ถ้าฟิลด์ Inst\_Causes ถูกปล่อยไว้ว่าง ลำดับซึ่งการดำเนินการที่แนะนำที่ถูกแสดง ถูกพิจารณาตามสูญเสียการดำเนินการและความน่าจะเป็นที่การดำเนินการ จะแก้ไขข้อผิดพลาดได้ การดำเนินการที่ไม่มีข้อเสียหรือมีไม่มาก หรือมีน้อยหรือไม่มีผลกระทบ กับระบบควรถูกแสดงรายการก่อนเสมอ การดำเนินการซึ่งความน่าจะเป็นในการแก้ไขข้อผิดพลาดเท่ากันหรือเกือบจะเท่ากันควรถูกแสดงถัดไป โดยแสดงการดำเนินการที่มีผลเสีย น้อยที่สุดก่อน การดำเนินการ การที่เลือก ควรถูกแสดงตามลำดับความน่าจะเป็นที่ลดลง ฟิลด์นี้อาจระบุข้อความลักษณะ XPG4 ซึ่งจะกล่าวถึงในภายหลัง

Inst\_Causes

อธิบายสาเหตุการติดตั้งสำหรับข้อผิดพลาดที่เกิดขึ้น สาเหตุการติดตั้ง ถูกกำหนดให้เป็นสภาวะที่เป็นผลมาจากการติดตั้ง เริ่มต้น หรือการเชื่อมต่อของรีซอร์ส สามารถระบุรายการของ identifier ข้อความ Install Cause ที่แยกกันด้วยคอมมาได้ถึง 4 รายการ ค่านี้ระบุเมสเสจข้อความจากเซตข้อความ Install Cause “I” ที่จะถูกแสดงสำหรับการเกิดขึ้นของ ข้อผิดพลาด ค่า ถูกแปลเป็นเลขฐานสิบหกไม่มีเครื่องหมายมีความยาวได้ถึงสี่หลัก สาเหตุการติดตั้งควรถูกแสดงตามลำดับความน่าจะเป็นที่ลดลง ฟิลด์นี้สามารถปล่อยไว้ว่างได้ถ้าไม่สามารถนำไปใช้กับ ข้อผิดพลาดที่เกิดขึ้น ถ้าฟิลด์นี้ถูกปล่อยว่าง ฟิลด์ User\_Causes หรือ Fail\_Causes ต้องไม่ว่าง ฟิลด์นี้อาจระบุข้อความลักษณะ XPG4 ซึ่งจะกล่าวถึงในภายหลัง

LABEL

ระบุเลเบลเฉพาะได้มากถึง 19 อักขระที่ต้องถูกจัดเตรียม สำหรับแต่ละเพิ่มเพลตการบันทึกข้อผิดพลาด สตรีมมี “#define #ERRID\_label Error\_ID”, โดยที่ค่า Error\_ID คือ ID เฉพาะที่กำหนดให้กับ Error Record Template ถูกเขียน ไปที่เอาต์พุตมาตรฐาน ถ้าแฟล็ก -h ถูกระบุที่ บรรทัดคำสั่ง

Log

หมายเหตุ: หากฟิลด์ LABEL เกิน 19 อักขระ ดังนั้นจะยอมรับเฉพาะ 19 อักขระแรกเท่านั้น

ระบุรายการบันทึกข้อผิดพลาดควรถูกสร้างสำหรับข้อผิดพลาดนี้หรือไม่ เมื่อเกิดขึ้น ฟิลด์ log สามารถถูกเซตเป็น True หรือ False ถ้าฟิลด์นี้ถูกละเว้นจาก เพิ่มเพลต ค่าจะเป็นค่าดีฟอลต์คือ True เมื่อฟิลด์นี้ ถูกเซตเป็น False ฟิลด์ Report และ Alert จะถูกละเว้น

ไอเท็ม Prob_Causes	คำอธิบาย อธิบายสาเหตุที่เป็นไปได้ 1 สาเหตุหรือมากกว่านั้นสำหรับข้อผิดพลาดที่เกิดขึ้น สามารถระบุรายการของ identifier ข้อความ Probable Cause ที่แยกกันด้วยคอมมาได้ถึง 4 รายการ ค่านี้ระบุเมสเสจข้อความจากเซตข้อความ Probable Cause "P" ที่จะถูกแสดงสำหรับการเกิดขึ้นของข้อผิดพลาด ค่าถูกแปลงเป็นเลขฐานสิบหกไม่มีเครื่องหมาย มีความยาวได้ถึง 4 หลัก สาเหตุการติดตั้งควรแสดงตามลำดับความน่าจะเป็นที่ลดลง อย่างน้อย ต้องมีสาเหตุที่เป็นไปได้หนึ่งสาเหตุ ฟิลด์ ยังอาจระบุข้อความลักษณะ XPG4 ซึ่งจะกล่าวถึงในภายหลัง
Report	ระบุนวการเกิดขึ้นของข้อผิดพลาดนี้ที่บันทึกไว้ควรถูกรายงาน เมื่อรายงานความผิดพลาดถูกพิมพ์หรือไม่ ฟิลด์ Report สามารถถูกเซตเป็น True หรือ False ถ้าฟิลด์นี้ถูกละเว้นจาก เพิ่มเฟลต ค่าจะเป็นค่าดีฟอลต์คือ True
User_Actions	อธิบายการดำเนินการที่แนะนำสำหรับการแก้ไขข้อผิดพลาดที่เกิดจากการผู้ใช้ สามารถระบุรายการของ identifier ข้อความ Recommended Action แยกกันด้วยคอมมาได้ถึง 4 รายการ ค่านี้ระบุเมสเสจข้อความจากเซตข้อความ Recommended Action "R" ที่จะถูกแสดงสำหรับการเกิดขึ้นของข้อผิดพลาด ค่าถูกแปลงเป็นเลขฐานสิบหกไม่มีเครื่องหมาย มีความยาวได้ถึง 4 หลัก ฟิลด์นี้ต้องถูกปล่อยว่าง ถ้าฟิลด์ User_Causes ถูกปล่อยไว้ว่าง ลำดับซึ่งการดำเนินการที่แนะนำที่ถูกแสดง ถูกพิจารณา ตามความเสียหายจากข้อผิดพลาดและความน่าจะเป็นที่การดำเนินการ จะแก้ไขข้อผิดพลาดได้ การดำเนินการที่มีความน่าจะมีข้อเสียหรือมีไม่มาก หรือมี น้อยหรือไม่มีผลกระทบต่อระบบควรแสดงรายการก่อนเสมอ การดำเนินการซึ่งความน่าจะเป็นในการ แก้ไขข้อผิดพลาดเท่ากันหรือเกือบจะเท่ากันควรแสดงถัดไป โดยแสดงการดำเนินการ ที่มีผลเสียน้อยที่สุดก่อน การดำเนินการที่เหลือ ควรแสดงตามลำดับความน่าจะเป็นที่ลดลง ฟิลด์ ยังอาจระบุข้อความลักษณะ XPG4 ซึ่งจะกล่าวถึงในภายหลัง
User_Causes	อธิบายสาเหตุจากผู้ใช้สำหรับข้อผิดพลาดที่เกิดขึ้น สาเหตุจากผู้ ใช้ ถูกกำหนดเป็นสภาวะที่สามารถถูกแก้ไขได้โดยไม่ต้องติดต่อองค์กร ให้บริการ สามารถระบุรายการของ identifier ข้อความ User Cause ที่แยกกันด้วยคอมมาได้ถึงสี่รายการ ค่านี้ระบุเมสเสจข้อความจากเซตข้อความ User Cause "U" ที่จะถูกแสดงสำหรับการเกิดขึ้นของ ข้อผิดพลาด ค่าถูกแปลงเป็นเลขฐานสิบหกไม่มีเครื่องหมาย มีความยาวได้ถึงสี่หลัก สาเหตุจากผู้ ใช้ควรแสดงตามลำดับความน่าจะเป็นที่ลดลง ฟิลด์นี้สามารถปล่อยไว้ว่างไว้ได้ถ้าไม่สามารถนำไปใช้กับ ข้อผิดพลาดที่เกิดขึ้น ถ้าฟิลด์นี้ถูกปล่อยว่าง ฟิลด์ Inst_Causes หรือ Fail_Causes ต้องไม่ว่าง ฟิลด์ ยังอาจระบุข้อความลักษณะ XPG4 ซึ่งจะกล่าวถึงในภายหลัง

catname ถูกใช้เพื่อระบุแค็ตตาล็อกข้อความที่จะถูกใช้เพื่อเรียกข้อความ XPG4 สำหรับเพิ่มเฟลตปัจจุบัน ซึ่งจะเขียนทับแค็ตตาล็อกที่ระบุ ด้วยตัวระบุแค็ตตาล็อก "\*"! ก่อนหน้านั้น เพิ่มเฟลตที่มีข้อความ XPG4 ต้องมีที่ระบุด้วย catname หรือ "\*"!. ชื่อแค็ตตาล็อก ต้องอยู่ในอัญประกาศ นอกจากชื่อพารามิเตอร์ที่แค็ตตาล็อกถูกระบุ กฎทั่วไปสำหรับการรับแค็ตตาล็อกข้อความ เป็นดังนี้

ตัวอย่าง ถ้า

```
catname = "mycat.cat"
```

ถูกระบุ mycat.cat จะถือว่าอยู่ใน /usr/lib/nls/msg/%L

ข้อความ Error คำอธิบาย, Probable Cause, User Cause, Install Cause, Failure Cause, Recommended Actions และ Detailed Data ID ต้องเป็น identifier ข้อความแสดงความผิดพลาดที่ดูแลในแค็ตตาล็อกข้อความข้อความบันทึกความผิดพลาด หรือข้อความ XPG4

identifier ข้อความแสดงความผิดพลาดประกอบด้วยตัวเลขฐานสิบหกได้ถึง 4 หลักโดยไม่มี "0x" นำหน้า ตัวอย่าง 1234 หรือ ABCD คำสั่ง `errmsg -w` สามารถถูกใช้เพื่อพิมพ์ข้อความเหล่านี้ตามด้วย identifiers คำสั่ง `errmsg` สามารถ ถูกใช้เพื่อเพิ่มข้อความใหม่

ข้อความ XPG4 ถูกระบุโดยใช้ฟอร์ม

```
{<set>, <number>, <"default text">}
```

เซต หมายเลข และข้อความดีฟอลต์จำเป็นทั้งหมด การอ้างอิงข้อความทางสัญลักษณ์ ไม่สนับสนุน นอกจากนี้ เพิ่มเฟลตที่มีข้อความ XPG4 จะ ไม่สามารถแจ้งเตือนได้

แค็ตตาล็อกข้อความต้องถูกระบุสำหรับข้อความ XPG4 ซึ่งทำได้โดยตัวระบุแค็ตตาล็อก "\*"! หรือคีย์เวิร์ด catname

การบันทึกข้อผิดพลาดไม่สนับสนุนคุณลักษณะทั้งหมดของการสร้างข้อความแสดงข้อผิดพลาดปกติ สตริงที่ใช้ในเพิ่มเพลตบันทึกข้อผิดพลาด ต้องเป็นไปตามข้อจำกัด

- ไม่สนับสนุนการแทนที่ตัวแปร ตัวอย่าง สตริงอาจ ไม่ถูกใช้ตามรูปแบบที่ระบุกับค่าการพิมพ์ สตริงอาจมีเพียง อักขระการจัดรูปแบบ "\t" และ "\n"
- สตริงข้อความดีฟอลต์ต้องยาวไม่เกิน 1 kb, 1024 ไบต์
- ซึ่งต้องหมายเหตุไว้ว่ารายละเอียดข้อผิดพลาดถูกพิมพ์ใน 40 พื้นที่อักขระ บนรายงานที่ไม่มีรายละเอียด ไม่มีการจัดรูปแบบสตริงสำหรับรายงานเหล่านี้ และเฉพาะ 40 อักขระแรกเท่านั้นที่จะถูกพิมพ์
- สตริงไม่ควรมีการขึ้นบรรทัดใหม่ในส่วนท้าย ซึ่งถูกกำหนดโดย errpt

สำหรับแต่ละรายการที่เพิ่ม คำสั่ง errupdate กำหนด Error ID เฉพาะที่ถูกเขียนให้กับไฟล์ส่วนหัวที่ระบุโดย File.h (โดยที่พารามิเตอร์ File คือชื่อของ อินพุตไฟล์คำสั่ง errupdate) ถ้าคำสั่ง errupdate กำลังอ่านข้อมูล จากอินพุตมาตรฐาน ประโยค #define ถูกเขียนไปที่เอาต์พุตมาตรฐาน ค่าที่กำหนดสำหรับฟิลด์ Class, Err\_Desc, Err\_Type, Fail\_Actions, Fail\_Causes, Inst\_Actions, Inst\_Causes, Prob\_Causes, User\_Actions, User\_Causes และ Detail\_Data ค่า data\_id ถูกใช้เพื่อคำนวณ Error ID เฉพาะสำหรับข้อผิดพลาดนั้น สำหรับเพิ่มเพลต XPG4 Label ถูกรวมไว้ในการคำนวณเช่นกัน

เนื้อหาของฟิลด์ Log, Report และ Alert ไม่ถูกรวมอยู่ในการคำนวณของ Error ID เฉพาะ; ดังนั้นคุณสมบัติ การบันทึก รายงาน และการแจ้งเตือน ของข้อผิดพลาด สามารถถูกแก้ไขได้ตลอดเวลาในนิยามรายการข้อผิดพลาดที่เก็บใน Error Record Template Repository โดยใช้คำสั่ง errupdate และหมายเหตุไว้ว่าส่วน data\_len และ data\_encode ของฟิลด์ข้อมูลรายละเอียด ไม่ถูกใช้

คำสั่ง errupdate สร้างไฟล์ เลิกทำในไดเรกทอรีปัจจุบันไฟล์ชื่อ File.undo ถ้าคำสั่ง errupdate กำลังอ่านข้อมูล จากอินพุตมาตรฐาน ไฟล์ undo ถูกเขียน ไปที่ไฟล์ errids.undo ไฟล์ undo มีอินพุตไปที่คำสั่ง errupdate เพื่อเลิกทำการเปลี่ยนแปลงที่ คำสั่ง errupdate สร้างขึ้น

คำสั่ง errpt -t สามารถถูก ใช้เพื่อดูเนื้อหาของ Error Record Template Repository เพิ่มเพลต ถูกประมวลผลและพิมพ์ตามที่ จะแสดงในรายงานความผิดพลาด จริง

**ข้อควรใส่ใจ:** ถ้าคุณเปลี่ยนเพิ่มเพลตข้อผิดพลาด โปรดระวังว่าเพิ่มเพลตเหล่านี้อาจถูกเปลี่ยนโดยการอัปเดตในภายหลัง คุณ ควรเก็บบันทึกการเปลี่ยนแปลงทั้งหมดที่ทำและนำการเปลี่ยนแปลงมาใช้อีกครั้งเมื่อ ระบบของคุณถูกอัปเดต นี่จำเป็นเฉพาะหลังจากการอัปเดตระบบหลัก เช่นการย้ายระบบปฏิบัติการไปสู่ระดับใหม่ และบันทึกดังกล่าวอนุญาตให้คุณเปลี่ยนแปลงเพิ่มเพลตของคุณ ถ้าคุณติดตั้งใหม่ วิธีที่ง่ายที่สุด ในการเก็บเรียกคอร์ดดังกล่าวคือ ทำการเปลี่ยนแปลงเพิ่มเพลตของคุณจากหนึ่ง ไฟล์ต้นฉบับ errupdate

## แฟล็ก

ไอเท็ม	คำอธิบาย
-c	ตรวจสอบอินพุตไฟล์เพื่อหาข้อผิดพลาดทางไวยากรณ์
-f	บังคับให้เพิ่มเพลตทั้งหมดถูกอัปเดต รวมถึงเพิ่มเพลตที่มี error ids เหมือนกับที่อยู่ในอินพุตเพิ่มเพลต
-h	สร้างประโยค #define สำหรับแต่ละ Error ID ที่กำหนดให้กับเพิ่มเพลตข้อผิดพลาด ถ้าชื่อไฟล์ถูกระบุบน บรรทัดคำสั่งชื่อไฟล์ส่วนหัวจะเป็นชื่อไฟล์ที่กำหนดต่อด้วย .h มิฉะนั้นประโยค #define ถูกเขียนไปที่เอาต์พุตมาตรฐาน
-n	ยับยั้งข้อมูลเพิ่มเติมของเพิ่มเพลตเรียกคอร์ดข้อผิดพลาดให้กับ Error Record Template Repository
-p	เพิ่มหรืออัปเดตเพิ่มเพลตโดยมีฟิลด์ Alert เซ็ตเป็น True ซึ่งมีค่า id ข้อมูล Error คำอธิบาย, Probable Cause, User Cause, User Action, Install Cause, Install Action, Failure Cause, Fail Action, หรือ Detailed Data ซึ่งไม่เป็นที่รู้จักโดย SNA Generic Alert Architecture (เอกสาร GA27-3136) คำสั่ง errupdate จะไม่อนุญาตให้คุณเพิ่มเพิ่มเพลตที่มีคุณสมบัติเหล่านี้ นอกจากนี้คุณระบุแฟล็กนี้
-q	ยับยั้งการสร้างไฟล์เล็กทำ
-y <i>FileName</i>	ใช้ไฟล์เพิ่มเพลตเรียกคอร์ดข้อผิดพลาดที่ระบุโดยพารามิเตอร์ <i>FileName</i>

## ความปลอดภัย

คำควบคุมการเข้าถึง: ไม่มี แต่คุณต้องมีสิทธิ์ในการเขียน ไฟล์เพิ่มเพลตที่คุณเปลี่ยนแปลงข้อมูล, /var/adm/ras/errtmpl โดยดีฟอลต์

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

- เมื่อต้องการเพิ่มรายการ กำหนดรายการในอินพุตไฟล์ ในรูปแบบต่อไปนี้:

```
+ CDRom_ERR22:
  Comment=      "Temporary CDRom read error"
  Class= H
  Log=          True
  Report= True
  Alert=        False
  Err_Type=     TEMP
  Err_Desc=     E801
  Prob_Causes=  5004
  Fail_Causes=  E800, 6312
  Fail_Actions= 1601, 0000
  Detail_Data=  120, 11, HEX
  Detail_Data=  4, 8058, DEC
  Detail_Data=  4, 8059, DEC
```

เมื่อต้องการป้อนข้อมูล,

```
errupdate <input file>
```

- เมื่อต้องการแก้ไขคุณสมบัติบันทึก รายงานและการแจ้งเตือน ของรายการ 99999999, ระบุโอเปอเรเตอร์แก้ไข = (เครื่องหมายเท่ากับ) ตามด้วย Error ID เฉพาะ และคุณสมบัติใหม่ สำหรับรายการที่จะถูกแก้ไข:

```
errupdate
=99999999:
  Report = False
  Log = True
```

3. เมื่อต้องการลบรายการ 99999999 จาก Error Record Template Repository, ระบุโอเปอเรเตอร์ลบ - (เครื่องหมายลบ) ตามด้วย Error ID เฉพาะของรายการที่จะถูกลบ:

```
errupdate
-99999999:
```

4. เมื่อต้องการเขียนทับแค็ตตาล็อกข้อความ XPG4 ที่ระบุสำหรับอินพุต stream ด้วย "\*", ใช้คีย์เวิร์ด "catname"

```
*!mycat.cat
```

\*mycat.cat ถูกใช้สำหรับข้อความ XPG4 ทั้งหมด ต่อจากนี้

\* ยกเว้นข้อมูลนี้:

```
+ CDROM_ERR23:
    Comment=      "Temporary CDROM read error"
    catname= "othercat.cat"
    Class= H
    Log=          True
    Report= True
    Alert=        False
    Err_Type=     TEMP
    Err_Desc=     {1, 1, "CD ROM is broken"}
    Prob_Causes= {2, 1, "cause 1"},\
                {2, 2, "Cause 2"}
    Fail_Causes=  E800, 6312
    Fail_Actions= 1601, 0000
    Detail_Data=  120, 11, HEX
    Detail_Data=  4, 8058, DEC
    Detail_Data=  4, 8059, DEC
```

แค็ตตาล็อก othercat.cat จะถูกใช้สำหรับเพิ่มเพลต CDROM\_ERR23 template เท่านั้น

หมายเหตุ: เพิ่มเพลตอาจมีทั้งข้อความ XPG4 และ id ข้อผิดพลาดทั่วไป หรือ codepoints

## ไฟล์

ไอเท็ม

```
/usr/include/sys/errids.h
/usr/include/sys/err_rec.h
```

คำอธิบาย

มีไฟล์ส่วนหัวที่มี Error ID  
มีไฟล์ส่วนหัวที่มีโครงสร้างสำหรับการบันทึกข้อผิดพลาด

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง errpt” ในหน้า 438

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง errlog

ภาพรวมลือกข้อผิดพลาด

---

## คำสั่ง ethchan\_config

### วัตถุประสงค์

เพิ่มอะแดปเตอร์ให้แก่ EtherChannel หรือลบอะแดปเตอร์ออกจาก EtherChannel

## ไวยากรณ์

```
ethchan_config { -a [ -b ] | -d } [ -p ParentName ] EtherChannel Adapter
```

```
ethchan_config -c [ -p ParentName ] EtherChannel Attribute New Value
```

```
ethchan_config -f [ -p ParentName ] EtherChannel
```

## คำอธิบาย

คำสั่งนี้เพิ่มอะแดปเตอร์ให้แก่ EtherChannel หรือลบอะแดปเตอร์ออกจาก EtherChannel คำสั่งนี้ยังสามารถใช้เพื่อแก้ไขแอตทริบิวต์ *EtherChannel* การเพิ่ม การลบ หรือการแก้ไขเหล่านี้จะเกิดขึ้นแม้ว่า จะมีการกำหนดค่าอินเตอร์เฟซของ EtherChannel นั่นคือ ไม่จำเป็นต้องปลด อินเตอร์เฟซของ EtherChannel เพื่อเพิ่มหรือลบอะแดปเตอร์ หรือแก้ไขแอตทริบิวต์ EtherChannel ส่วนใหญ่

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a	เพิ่ม Adapter ที่ระบุให้แก่ EtherChannel ที่ระบุ หากอะแดปเตอร์ต้องถูกเพิ่มเป็นอะแดปเตอร์สำรอง ต้องระบุแฟล็ก -b
-b	ระบุว่า Adapter ถูกเพิ่มเป็นอะแดปเตอร์สำรอง แฟล็กนี้ใช้ได้ต่อเมื่อใช้กับแฟล็ก -a
-c	เปลี่ยน Attribute ที่ระบุของ แอตทริบิวต์ EtherChannel ที่ระบุ ให้แก่ New Value ที่ระบุ
-d	ลบ Adapter ที่ระบุออกจาก EtherChannel ที่ระบุ แฟล็ก -b ต้องไม่ใช้กับ แฟล็ก -d
-f	กำหนดความล้มเหลวของ EtherChannel ที่ระบุ failover จะเกิดขึ้นเฉพาะหากอะแดปเตอร์ในสถานะที่ไม่ทำงานเป็น up หากอะแดปเตอร์ในสถานะที่ไม่ทำงานเป็น down EtherChannel จะยังคงทำงาน บนสถานะที่แอ็คทีฟและไม่มีการ failover
-p	ระบุพารามิเตอร์ของ EtherChannel หาก Shared Ethernet Adapter (SEA) ถูกกำหนดค่านับ EtherChannel แฟล็กนี้ต้องถูกใช้พร้อมกับแฟล็กอื่นเพื่อเปลี่ยนแอตทริบิวต์ใดๆ ของ EtherChannel (เช่น การเพิ่ม หรือการลบอะแดปเตอร์)

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
Adapter	ระบุอะแดปเตอร์เพื่อเพิ่มหรือลบ
แอตทริบิวต์	ระบุแอตทริบิวต์ของ EtherChannel ที่ระบุ
EtherChannel	ระบุ EtherChannel
New Value	ระบุค่าใหม่สำหรับแอตทริบิวต์ที่ระบุ ของ EtherChannel ที่ระบุ
ParentName	ระบุพารามิเตอร์ของ EtherChannel

## สถานะออก

ไอเท็ม	คำอธิบาย
0	คำสั่งดำเนินการเสร็จด้วยผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการเพิ่มอะแดปเตอร์ ent0 เป็นอะแดปเตอร์สำรอง ใน EtherChannel ที่ชื่อ ent7 ให้ป้อนคำสั่งต่อไปนี้:

```
/usr/lib/methods/ethchan_config -a -b ent7 ent0
```

2. เมื่อต้องการเปลี่ยนแอตเดรสเพื่อ ping แอตทริบิวต์ของ EtherChannel ที่ชื่อ ent7 เป็น 10.10.10.10 ให้ป้อนคำสั่งต่อไปนี้:

```
/usr/lib/methods/ethchan_config -c ent7 netaddr 10.10.10.10
```

- เมื่อต้องการบังคับการ failover ของ EtherChannel ที่ชื่อ ent7 จาก แชนแนลที่แอ็คทีฟในปัจจุบันเป็นแชนแนลที่ไม่ทำงานให้ป้อน คำสั่งต่อไปนี้:

```
/usr/lib/methods/ethchan_config -f ent7
```

- เมื่อต้องการลอบะแต่ปเตอร์ ent13 จาก EtherChannel ที่ชื่อ ent18 ซึ่งเป็นของ SEA ที่ชื่อ ent32 ให้ป้อนคำสั่งต่อไปนี้:

```
/usr/lib/methods/ethchan_config -d -p ent32 ent18 ent13
```

## ข้อจำกัด

การใช้แอ็คทีวิตี `use_jumbo_frame` ไม่สามารถ แก้ไขโดยคำสั่งนี้ หากคุณพยายามแก้ไขแอ็คทีวิตีนี้ คำสั่งนี้จะแสดงข้อความแสดงข้อผิดพลาด

## Location

```
/usr/lib/methods
```

---

## คำสั่ง ewallevent

### วัตถุประสงค์

กระจายเหตุการณ์หรือเหตุการณ์ใหม่ไปยังผู้ใช้ทั้งหมดที่ล็อกอิน

### ไวยากรณ์

```
ewallevent [-c] [-h]
```

### คำอธิบาย

สคริปต์ `ewallevent` กระจายข้อความของเหตุการณ์หรือเหตุการณ์ `rearm` ไปยังผู้ใช้ทั้งหมด ที่ขณะนี้ล็อกอินอยู่ในโฮสต์เมื่อเหตุการณ์หรือเหตุการณ์ `rearm` เกิดขึ้น ข้อมูลเหตุการณ์ หรือเหตุการณ์ที่ใช้ใหม่ถูกบันทึกและ ตัดประกาศโดย event response resource manager ในตัวแปรสถานะแวดล้อมที่ถูกสร้างโดย event response resource manager เมื่อเหตุการณ์หรือเหตุการณ์ที่ใช้ใหม่ เกิดขึ้น สคริปต์นี้สามารถถูกใช้เป็นการดำเนินการที่ถูกรัน โดยรีซอร์สการตอบกลับเหตุการณ์ และยังสามารถนำมาใช้เป็นเทมเพลตเพื่อสร้าง การดำเนินการที่ผู้ใช้กำหนดเอง อื่น สคริปต์นี้ส่งกลับข้อความเป็นภาษาอังกฤษเสมอ

ข้อความถูกแสดงในรูปแบบนี้ที่คอนโซลของผู้ใช้ทั้งหมดที่ ล็อกอิน เมื่อเหตุการณ์หรือเหตุการณ์ `rearm` เกิดขึ้นซึ่งสคริปต์นี้คือการดำเนินการตอบกลับ :

```
Broadcast message from user@host (tty) at hh:mm:ss...
```

```
severity event_type occurred for Condition condition_name  
on the resource resource_name of resource_class_name at hh:mm:ss mm/dd/yy  
The resource was monitored on node_name and resided on {node_names}.
```

ข้อมูลเหตุการณ์ที่ถูกส่งกลับเกี่ยวกับตัวแปรสถานะแวดล้อม `ERRM` และ รวมถึงข้อมูลต่อไปนี้:

### Local Time

เวลาเมื่อเหตุการณ์หรือเวลาที่ผู้ใช้ใหม่ถูกพบ ตัวแปรสถานะแวดล้อมจริง ที่ระบุโดย `ERRM` คือ `ERRM_TIME` ค่านี้ถูกแปลและแปลง เป็นรูปแบบที่อ่านได้ ก่อนถูกแสดง

สคริปต์นี้บันทึกค่าตัวแปรสถานะแวดล้อม และใช้คำสั่ง `wall` เพื่อเขียนข้อความไปยังคอนโซลผู้ใช้ที่ล็อกอินขณะนี้

## แฟล็ก

- c สั่งให้ `ewallevent` กระจาย `ERRM_VALUE` ของเหตุการณ์ `ERRM` เมื่อแฟล็ก `-c` ถูกระบุ `ewallevent` กระจายข้อความการดักข้อมูล SNMP
- h เขียนคำสั่งการใช้สคริปต์ไปที่เอาต์พุตมาตรฐาน

## พารามิเตอร์

### *log\_file*

ระบุชื่อไฟล์ที่ข้อมูลเหตุการณ์ถูกบันทึก พาส์สมบูรณ์สำหรับพารามิเตอร์ *log\_file* ควรถูกระบุ

*log\_file* ถูกใช้เป็นบันทึกไว้เป็นวงรอบ และมีขนาดที่กำหนดไว้ นั่นคือ 64KB เมื่อ *log\_file* เต็ม รายการใหม่ถูกเขียนทับรายการเก่าที่สุดที่มีอยู่

หาก *log\_file* มีอยู่ก่อนแล้ว รายละเอียดของเหตุการณ์จะถูกต่อท้าย บันทึกการทำงานนั้น ถ้าไม่มี *log\_file*, ไฟล์จะถูกสร้างเพื่อที่ข้อมูลเหตุการณ์ สามารถถูกเขียนลงไปได้

## สถานะออก

- 0 สคริปต์รันเสร็จสมบูรณ์
- 1 เกิดข้อผิดพลาดเมื่อรันสคริปต์

## ข้อจำกัด

1. สคริปต์นี้ต้องรันบนโหนดที่ `ERRM` กำลังรัน
2. คำสั่ง `wall` ถูกใช้เพื่อเขียนข้อความไปยังคอนโซล ผู้ใช้ที่ล็อกอินขณะนี้ อ้างถึงหน้า `wall man` เพื่อดูข้อมูลเพิ่มเติมเกี่ยวกับคำสั่ง `wall`

## เอาต์พุตมาตรฐาน

เมื่อมีการระบุแฟล็ก `-h` คำสั่งการใช้สคริปต์จะถูก เขียนไปที่เอาต์พุตมาตรฐาน

## ตัวอย่าง

1. สมมติสคริปต์ `ewallevent` เป็นการดำเนินการที่กำหนดไว้แล้วในการตอบกลับ การแจ้งเตือนวิกฤต ซึ่งเชื่อมโยงกับเงื่อนไข `/var space used` บนรีซอร์ส `/var` ถึงค่าขีดจำกัดของนิพจน์เหตุการณ์ที่กำหนด สำหรับเงื่อนไข และมีเหตุการณ์เกิดขึ้น การตอบกลับการแจ้งเตือน สำคัญเกิดขึ้น และ `ewallevent` รัน ข้อความต่อไปนี้ จะแสดงบนคอนโซลของผู้ใช้ทั้งหมดที่ล็อกอิน:

```
Broadcast message from joe@neverland.com (pts/6) at 18:42:03...
```

```
Critical event occurred for Condition /var space used  
on the resource /var of filesystem of IBM.FileSystem at 18:41:50 03/28/02  
The resource was monitored on c174n05 and resided on {c174n05}.
```

2. เมื่อเหตุการณ์ `rearm` เกิดขึ้นสำหรับ `/เงื่อนไขที่ใช้พื้นที่ บนรีซอร์ส /var`, ข้อความดังต่อไปนี้ถูกแสดงบนคอนโซลของผู้ใช้ทั้งหมดซึ่งได้ล็อกอินเรียบร้อยแล้ว

Broadcast message from joe@neverland.com (pts/6) at 18:42:03...

Critical rearm event occurred for Condition /var space used  
on the resource /var of filesystem of IBM.FileSystem at 18:41:50 03/28/02  
The resource was monitored on c174n05 and resided on {c174n05}.

## Location

/usr/sbin/rsct/bin/ewallevent

---

## คำสั่ง ex

### วัตถุประสงค์

เอดิเตอร์สำหรับไฟล์ข้อความ

### ไวยากรณ์

```
ex[ -c Subcommand] [ -l] [ -R] [ -s] [ -tTag] [ -V] [ -wNumber] [ -v| -] [ +[Subcommand]] [ -r[File]] [File...]
```

### คำอธิบาย

คำสั่ง ex เริ่มทำงานเอดิเตอร์ ex เอดิเตอร์ ex เป็นส่วนหนึ่งของตระกูลเอดิเตอร์ ที่รวมเอดิเตอร์คำสั่ง edit ซึ่งเป็น เวอร์ชันที่ใช้งานง่ายกว่าของเอดิเตอร์ ex สำหรับผู้ใช้งานเริ่มต้น หรือการใช้งานโดยทั่วไป และเอดิเตอร์คำสั่ง vi ซึ่งเป็นเอดิเตอร์ที่แสดงเต็มหน้าจอ การเรียกใช้เอดิเตอร์ vi โดยตรงจะตั้งค่า ตัวแปรสภาวะแวดล้อมสำหรับการแก้ไขหน้าจอ เอดิเตอร์ ex มีประสิทธิภาพสูงกว่า เอดิเตอร์รายบรรทัดอย่างง่ายเนื่องจากเป็นเซ็ทย่อยของเอดิเตอร์ vi และสามารถ เข้าถึงความสามารถในการแก้ไขหน้าจอของเอดิเตอร์ vi

พารามิเตอร์ *File* ระบุ ไฟล์เดี่ยวหรือหลายไฟล์ที่จะแก้ไข หากคุณกำหนดชื่อไฟล์มากกว่าหนึ่งชื่อ เอดิเตอร์ ex จะแก้ไขแต่ละไฟล์ตามลำดับที่ระบุ

### Notes:

1. ในการกำหนดว่าเวิร์กสเตชันของคุณจะสามารถดำเนินการ ได้อย่างมีประสิทธิภาพมากยิ่งขึ้นได้อย่างไร เอดิเตอร์ ex จะใช้ฐานข้อมูลความสามารถเวิร์กสเตชัน **terminfo** และชนิดของเวิร์กสเตชันที่คุณกำลังใช้งาน จากตัวแปรสภาวะแวดล้อม **TERM**
2. คำสั่ง ex มีผลกับบรรทัดปัจจุบัน ยกเว้นคุณระบุเป็นอย่างอื่น เพื่อทำงานกับ ส่วนอื่นของไฟล์ คุณต้องทราบวิธี กำหนดแอตเตรสบรรทัดในไฟล์
3. หากอินพุตมาตรฐานไม่ใช่อุปกรณ์เทอร์มินัล จะเป็นตามที่คุณ ได้ระบุแฟล็ก -s

### แฟล็ก

## ไอเท็ม

-c Subcommand

### คำอธิบาย

ดำเนินการคำสั่งย่อยเอดิเตอร์ ex ให้เสร็จก่อนการแก้ไขเริ่ม เมื่อ มีการพิมพ์ตัวถูกดำเนินการ null เช่นใน -c '' เอดิเตอร์จะวางบรรทัดปัจจุบันที่ท้ายไฟล์ (โดยปกติ เอดิเตอร์ ex ตั้งค่าบรรทัดปัจจุบันที่จุดเริ่มต้นไฟล์ หรือที่แท็กหรือรูปแบบที่ระบุบาง อย่าง)

-I

เยื้องอย่างเหมาะสมสำหรับโค้ด LISP และยอมรับอักขระ ( ) (วงเล็บ เปิดหรือปิด), { } (วงเล็บปีกกาซ้ายหรือขวา) และ [[ ]] (วงเล็บเหลี่ยมคู่ ซ้าย หรือคู่ขวา) เป็นข้อความแทนการแปลความหมายเป็น คำสั่งย่อย vi แฟล็กนี้แอ็คทีฟในโหมดเห็นภาพและโหมดเปิด

-R

ตั้งค่าออปชัน **readonly** เพื่อป้องกันคุณจากการเปลี่ยนแปลงไฟล์

-s

ไม่แสดงผลป้อนกลับที่โต้ตอบกับผู้ใช้ทั้งหมด หากคุณใช้แฟล็กนี้ ไฟล์ อินพุตและข้อผิดพลาดเอาต์พุตจะไม่สร้างข้อความแสดงความผิดพลาดที่เป็นประโยชน์ การใช้ แฟล็กนี้จะเหมือนกับการใช้แฟล็ก -ข้ามค่า ของTERM และชนิดดีฟอลต์

-t Tag

เทอร์มินัลการนำไปปฏิบัติใดๆ และถือว่าเทอร์มินัล มีชนิดที่ไม่สามารถสนับสนุนโหมดเปิดและโหมดเห็นภาพ โหมดไฟล์ที่มีแท็กที่ระบุโดยพารามิเตอร์ *Tag* และกำหนดตำแหน่งเอดิเตอร์ที่แท็กนั้น ในการใช้แฟล็กนี้ อันดับแรกคุณต้องสร้างฐานข้อมูลของชื่อฟังก์ชันและตำแหน่ง โดยใช้คำสั่ง **ctags**

-w Number

ตั้งค่าขนาดหน้าต่างดีฟอลต์เป็น *Number*

-v

เรียกใช้เอดิเตอร์ vi

หมายเหตุ: เมื่อเลือกแฟล็ก -v ชุดของคำสั่งย่อยที่ขยายใหญ่ขึ้นจะ พร้อมใช้งาน รวมถึงคุณลักษณะการแก้ไขหน้า

จอและการย้ายเคอร์เซอร์ โปรดดูที่คำสั่ง vi

-V

เรียกใช้เอดิเตอร์ในโหมดตรายละเอียด

-

ไม่แสดงผลป้อนกลับที่โต้ตอบกับผู้ใช้ทั้งหมด หากคุณใช้แฟล็กนี้ ข้อผิดพลาด อินพุต/เอาต์พุตไฟล์จะไม่สร้างข้อความแสดงความผิดพลาดที่เป็นประโยชน์ การใช้ แฟล็กนี้จะเหมือนกับการใช้แฟล็ก -s

+ [ Subcommand ]

เริ่มการแก้ไขที่การค้นหาหรือคำสั่งย่อยเอดิเตอร์ที่ระบุ เมื่อ ไม่มีการพิมพ์พารามิเตอร์ +คำสั่งย่อยจะวางบรรทัดปัจจุบันที่ท้ายไฟล์ โดยปกติ เอดิเตอร์ ex ตั้งค่าบรรทัดปัจจุบันที่จุดเริ่มต้นไฟล์ หรือที่แท็กหรือรูปแบบที่ระบุบาง อย่าง

-r [File]

กู้คืนไฟล์หลังจากเอดิเตอร์หรือระบบขัดข้อง หากคุณไม่ระบุ พารามิเตอร์ *File* รายการของไฟล์ที่บันทึกทั้งหมดจะถูกแสดง

## สถานะออก

ค่าออกต่อไปนี้ถูกส่งคืน:

ไอเท็ม คำอธิบาย

0 ดำเนินการเป็นผลสำเร็จ

>0 มีข้อผิดพลาดเกิดขึ้น

## ไฟล์

ไอเท็ม

/usr/sbin/exrecover

/usr/sbin/expreserve

\$HOME/.exrc

./exrc

/var/tmp/Exnnnnn

/var/tmp/Rxnnnnn

/var/preserve

คำอธิบาย

กู้คืนคำสั่งย่อย

สงวนคำสั่งย่อย

ไฟล์เริ่มทำงานเอดิเตอร์

ไฟล์เริ่มทำงานเอดิเตอร์

ค่าชั่วคราวของเอดิเตอร์

ชื่อบัพเฟอร์ชั่วคราว

ไดเรกทอรีการสงวน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง ed หรือ red” ในหน้า 297

“คำสั่ง edit” ในหน้า 336

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง ctags

คำสั่ง vi

ตัวแปรสถานะแวดล้อม TERM

---

## คำสั่ง `execerror`

### วัตถุประสงค์

เขียนข้อความแสดงความผิดพลาดไปยังข้อผิดพลาดมาตรฐาน

### ไวยากรณ์

`execerror`

### คำอธิบาย

คำสั่ง `execerror` ถูกเรียกใช้งาน โดยรูทีนย่อย `exec` เมื่อการโหลดโปรแกรมจริง ไม่สำเร็จ คำสั่งได้รับชื่อของไฟล์ที่จะถูกเรียกใช้งานและสตริงข้อความแสดงความผิดพลาดของตัวโหลดที่ไม่มีหรือมีหลายสตริงที่ถูกส่งมา แต่ละสตริงข้อความแสดงความผิดพลาดของตัวโหลด จะมีหมายเลขข้อผิดพลาดตามด้วยข้อมูลข้อผิดพลาด

### ตัวอย่าง

คำสั่ง `execerror` ถูกใช้ ดังนี้:

```
char *buffer[1024];
buffer[0] = "execerror" ;
buffer[1] = "name of program that failed to load";
loadquery(L_GETMESSAGES, &buffer[2], sizeof buffer -8);
execvp("/usr/sbin/execerror",buffer);
```

โค้ดตัวอย่างนี้ทำให้แอปพลิเคชันจบการทำงาน หลังจากเขียนข้อความไปยังข้อผิดพลาดมาตรฐาน

### ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/sbin/execerror</code>	มีคำสั่ง <code>execerror</code>

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `exec`

คำสั่ง `loadquery`

---

## คำสั่ง `execrset`

### วัตถุประสงค์

รันโปรแกรมหรือคำสั่งที่แนบกับ `rset`

## ไวยากรณ์

```
execrset [ -P ] [ -F ] -c CPUlist [ -m MEMlist ] -e command [ parameters ]
```

or

```
execrset [ -P ] [ -F ] [ -S ] rsetname [ -e ] command [ parameters ]
```

## คำอธิบาย

คำสั่ง `execrset` เรียกใช้งานคำสั่งที่มีการแนบ กับ `rset` ซึ่งทำให้คำสั่งที่ระบุจะถูกจำกัดให้รันบนตัวประมวลผลและ/หรือส่วนหน่วยความจำที่อยู่ใน `rset` เท่านั้น ชื่อ `rset` ในเรจิสตรีระบบสามารถใช้เพื่อระบุ ตัวประมวลผลและ/หรือส่วนหน่วยความจำที่คำสั่งได้รับอนุญาตให้ใช้ หรือ `rset` ที่มีตัวประมวลผลและส่วนหน่วยความจำที่ระบุ สามารถถูกแนบกับกระบวนการได้

## แฟล็ก

ไอเท็ม	คำอธิบาย
-F	กำหนดให้คำสั่ง <code>execrset</code> ทำงาน แฟล็กนี้ลบการโยง <code>bindprocessor</code> และ <code>rset</code> ของเรดทั้งหมดใน กระบวนการออกก่อนเรียกใช้คำสั่ง หากระบุแฟล็ก -P เช่นกัน คำสั่งจะแยก <code>rset</code> ที่ใช้อยู่และ <code>rset</code> ของเรดทั้งหมดออกจากกระบวนการก่อนเรียกใช้คำสั่ง
-P	แนบ <code>rset</code> เป็น <code>rset</code> พาร์ติชัน
-c <i>CPUlist</i>	รายการ CPUs ที่จะอยู่ใน <code>rset</code> เพื่อ ใช้แนบกับกระบวนการซึ่งเรียกใช้งานโปรแกรมหรือคำสั่ง ซึ่งสามารถเป็นหนึ่ง CPU หรือมากกว่านั้นหรือช่วงของ CPU
-m <i>MEMlist</i>	รายการส่วนหน่วยความจำที่จะอยู่ใน <code>rset</code> โดยคำนี้สามารถเป็นส่วนหน่วยความจำหนึ่งหรือหลายส่วนหรือช่วง.
-e <i>command</i> [ <i>parameters</i> ]	ระบุคำสั่งเพื่อรันตามด้วยพารามิเตอร์ใดๆ แฟล็ก -e ต้องเป็นแฟล็กสุดท้ายที่ใช้ในคำสั่ง
-S	คำแนะนำที่ระบุว่ากระบวนการต้องถูกกำหนดเวลา เพื่อรันในโหมดเรดเดียว มีเพียงหนึ่งในฮาร์ดแวร์เรดของแต่ละตัวประมวลผลฟิลิคัลเท่านั้นที่ถูกรวมอยู่ใน <code>rset</code> ที่ระบุที่จะถูกใช้เพื่อกำหนดเวลา งาน หากฮาร์ดแวร์เรดทั้งหมดของตัวประมวลผลฟิลิคัลไม่ถูกรวมใน <code>rset</code> ที่ระบุ ตัวประมวลผลนั้นจะถูกละเว้น <code>rset</code> ที่ระบุ ต้องเป็น <code>rset</code> เฉพาะมีฉะนั้นคำสั่งจะล้มเหลว การระบุแฟล็กนี้อนุญาต งานรันโดยมีลักษณะการทำงานแบบเรดเดียว

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
<i>rsetname</i>	ชื่อของ <code>rset</code> ในเรจิสตรี ระบบที่จะแนบกับกระบวนการการเรียกใช้งานโปรแกรมหรือคำสั่ง

## ความปลอดภัย

ผู้ใช้งานต้องมีสิทธิ์ `root` หรือมีความสามารถ `CAP_NUMA_ATTACH` ผู้ใช้ต้องมีสิทธิ์ `root` เพื่อแนบพาร์ติชัน `rset` กับกระบวนการของคำสั่ง (แฟล็ก -P)

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ตัวอย่าง

1. ในการรันโปรแกรม test1 บน CPUs 0-7 ให้พิมพ์:  
`execrset -c 0-7 -e test1`
2. ในการรันโปรแกรม 'test2 parm1 parm2' ที่มีการแนบกับ rset ชื่อ test/cpus0to15 ให้พิมพ์:  
`execrset test/cpus0to15 test parm1 parm2`
3. ในการรันคำสั่ง ls -l บน CPU 0 ให้พิมพ์:  
`execrset -c 0 -e ls -l`

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/execrset	มีคำสั่ง execrset

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง detachrset” ในหน้า 106

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง attachrset

คำสั่ง lsrset

คำสั่ง mkrset

---

## คำสั่ง expand

### วัตถุประสงค์

เขียนไปยังเอาต์พุตมาตรฐานเอาต์พุตโดยที่แท็บถูกเปลี่ยนเป็นเว้นวรรค

### ไวยากรณ์

```
expand [-t TabList] [File ...]
```

```
expand [-tabstop][-tab1,tab2,...,tabn] [File ...]
```

### คำอธิบาย

คำสั่ง `expand` เขียน ลงไฟล์ที่ระบุชื่อ หรืออินพุตมาตรฐานไปเอาต์พุตมาตรฐาน และแทนที่อักขระแท็บ ด้วยอักขระเว้นวรรค อย่างน้อยหนึ่งอักขระ อักขระถอยกลับใดๆ จะถูกคัดลอกไปยัง เอาต์พุตและทำให้ตำแหน่งคอลัมน์มีค่าสำหรับการคำนวณแท็บหยุด เพื่อลดค่า การนับตำแหน่งคอลัมน์จะไม่ถูกลดค่าต่ำกว่าศูนย์

หมายเหตุ: พารามิเตอร์ *File* ต้องเป็นไฟล์ข้อความ

### แฟล็ก

ไอเท็ม  
-t TabList

คำอธิบาย  
ระบุตำแหน่งของระยะแท็บ ค่าดีฟอลต์ของระยะแท็บ คือ 8 ตำแหน่งคอลัมน์

ตัวแปร *TabList* ต้องประกอบด้วยเลขจำนวนเต็มบวกฐานสิบค่าเดียวหรือ เลขจำนวนเต็มบวกฐานสิบหลายค่า เลขจำนวนเต็มหลายค่าต้องอยู่ในลำดับ จากน้อยไปหามากและต้องคั่นด้วยเครื่องหมายจุลภาคหรือด้วยอักขระช่องว่างที่มีเครื่องหมายคำพูดรอบเลขจำนวนเต็ม ตัวแปร *TabList* เดี่ยว ตั้งค่าระยะแท็บด้วยจำนวนตำแหน่งคอลัมน์เท่าต่างหาก ตัวแปร *TabList* หลายตัวตั้งค่าแท็บหยุดที่ตำแหน่งคอลัมน์ที่สอดคล้องกับจำนวนเต็มในตัวแปร *TabList*

หากคำสั่ง `expand` ประมวลผล แท็บหยุดเลยแท็บหยุดสุดท้ายที่ระบุในตัวแปร *TabList* แท็บหยุดจะถูกแทนที่โดยอักขระเว้นวรรคเดียวในเอาต์พุต

## พารามิเตอร์

ไอเท็ม  
*tabstop*  
*tab1, tab2, ..., tabn*

คำอธิบาย  
ระบุเป็นอาร์กิวเมนต์เดียว โดยตั้งค่า *tabstop* เป็นอักขระ SPACE แยกแทนค่าดีฟอลต์ 8 ตั้งค่าอักขระ TAB ที่คอลัมน์ที่ระบุโดย *-tab1, tab2, ..., tabn*

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย  
0 ดำเนินการเป็นผลสำเร็จ  
>0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. ในการปรับเปลี่ยนแท็บหยุดด้วยจำนวน ระยะห่างที่เท่ากันใน `text.fil` ให้ป้อน:

```
expand -t 3 text.fil
```

หาก `text.fil` ประกอบด้วย:

```
1 2      3456789
```

ดังนั้นคำสั่ง `expand` จะแสดง:

```
1 2      3456789
```

2. ในการปรับเปลี่ยนแท็บหยุดด้วยจำนวนที่แตกต่างกันใน `text.fil` ให้ป้อน:

```
expand -t 3,15,22 text.fil
```

OR

```
expand -t "3 15 22" text.fil
```

หาก `text.fil` ประกอบด้วย:

```
1 2      3      456789
```

ดังนั้นคำสั่ง `expand` จะแสดง:

```
1 2      3      456789
```

# ไฟล์

ไอเท็ม  
/usr/bin/expand

คำอธิบาย  
มีคำสั่ง expand

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง newform

คำสั่ง tab

คำสั่ง unexpand

การเปลี่ยนทิศทางอินพุตและเอาต์พุต

---

## คำสั่ง **exfilt**

### วัตถุประสงค์

เอ็กซ์พอร์ตกฎตัวกรองไปยังเอ็กซ์พอร์ตไฟล์

### ไวยากรณ์

```
exfilt [ -p ] [ -q ] [ -r ] [ -v 4 | 6 ] -f directory [ -l filt_id_list ]
```

### คำอธิบาย

ใช้คำสั่ง **exfilt** เพื่อเอ็กซ์พอร์ตกฎตัวกรองไปไว้ใน ไฟล์ข้อความที่เอ็กซ์พอร์ต ซึ่งสามารถใช้โดยคำสั่ง **impfilt** คำสั่งนี้มีประโยชน์หากคุณต้องการกำหนดกฎที่คล้ายคลึง บนหลายๆ เครื่อง

**หมายเหตุ:** รายละเอียดตัวกรองบนเครื่องหนึ่งอาจไม่มีความหมาย หรือทำให้เกิดความตีความหมายที่ผิดในเครื่องอื่น ไฟล์นี้ไม่ถูกเอ็กซ์พอร์ต

กฎตัวกรอง IPsec สำหรับคำสั่งนี้สามารถกำหนดค่าได้โดยใช้คำสั่ง **genfilt** หรือ IPsec **smit** (IP เวอร์ชัน 4 หรือ IP เวอร์ชัน 6)

### แฟล็ก

ไอเท็ม  
-f directory  
-l filt\_id\_list

#### คำอธิบาย

ระบุไดเรกทอรีที่จะสร้างไฟล์ข้อความที่เอ็กซ์พอร์ต ไดเรกทอรี จะถูกสร้างขึ้นหากยังไม่มี แสดง IDs ของกฎตัวกรองที่คุณต้องการเอ็กซ์พอร์ต ID กฎตัวกรอง สามารถค้นด้วย "," หรือ "-" หากไม่ใช้แฟล็กนี้ กฎตัวกรองทั้งหมดที่กำหนดในตารางกฎ ตัวกรองสำหรับ IP versions ที่ใช้จะถูกเอ็กซ์พอร์ต

-p

อนุญาตสำหรับกฎที่กำหนดไว้แล้ว

-q

ระบุมืด quiet ไม่แสดงเอาต์พุตไปยัง stdout

-r

ระบุมืดที่ไม่แก้ไข เอ็กซ์พอร์ตกฎตัวกรองตามที่ เป็น และไม่มีการย้อน ทิศทางในกฎ ใช้แฟล็กนี้เมื่อกฎตัวกรองถูกเอ็กซ์พอร์ตและอิมพอร์ต ตามที่เป็น ตัวอย่างเช่น เพื่อบันทึกการกำหนดค่าหรือทำสำเนาการกำหนดค่าไปยัง เครื่องอื่น

ด้วยแฟล็ก -r ทิศทาง ของการรับส่งข้อมูลถูกสลับไว้ เช่น หากมีกฎอยู่บนโฮสต์ 10.0.0.1 เพื่อบันทึกการรับข้อมูลขาเข้า จาก 10.0.0.2 ซึ่ง exfilt ที่มีแฟล็ก -r จะเขียนกฎตัวกรองเดียวกัน

-v

การไม่ระบุแฟล็ก -r จะทำให้ทิศทางถูกสลับจากขาเข้าเป็นขาออกในเอ็กซ์พอร์ต ไฟล์

IP version ของกฎตัวกรองที่คุณต้องการเอ็กซ์พอร์ต ค่า 4 ระบุ IP version 4 และค่า 6 ระบุ IP version 6 เมื่อไม่ใช้แฟล็กนี้ กฎ IP version 4 และ IP version 6 ทั้งสองจะถูกเอ็กซ์พอร์ต

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `impfilt`

---

## คำสั่ง `explain`

### วัตถุประสงค์

จัดให้มีพจนานุกรมคำพ้องแบบโต้ตอบ

### ไวยากรณ์

`explain`

### คำอธิบาย

คำสั่ง `explain` มีพจนานุกรมคำพ้องแบบโต้ตอบสำหรับวลีภาษาอังกฤษที่พบโดยคำสั่ง `diction` ก่อนการใช้ คำสั่ง `explain` ให้ใช้คำสั่ง `diction` เพื่อจัดหารายการวลีที่เป็นคำไม่ถูกต้อง เมื่อคุณใช้คำสั่ง `explain` ระบบพร้อมดีให้คุณป้อนวลีและตอบกลับด้วยทางเลือกอื่นที่ยอมรับได้ในด้านไวยากรณ์ คุณสามารถพิมพ์วลีต่อ หรือคุณสามารถออกจากการทำงานโดยป้อนลำดับปุ่ม `Ctrl-D`

ไม่มีพารามิเตอร์บรรทัดคำสั่งอื่นที่ใช้ได้

### ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/lib/explain.d</code>	มีพจนานุกรมคำพ้อง

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `diction`” ในหน้า 149

---

## คำสั่ง `explore`

### วัตถุประสงค์

เริ่มทำงานเบราว์เซอร์ WebExplorer World Wide Web

### ไวยากรณ์

`explore` [ `-iFileName` ] [ `-tNumber` ] [ `-q` ] [ [ `-url` ] `URL` ]

## คำอธิบาย

คำสั่ง `explore` เปิดหน้าต่างหลัก WebExplorer และเชื่อมต่อไปยัง Uniform Resource Locator (URL) เพื่อดูเอกสารโฮม

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-iFileName</code>	ระบุไฟล์การกำหนดค่าเริ่มต้นอื่น โดยที่ <i>FileName</i> คือชื่อพารามิเตอร์เต็มของไฟล์เพื่อใช้แทน <code>\$HOME/.explore-preferences</code> ค่าดีฟอลต์ คำสั่งนี้อนุญาตให้คุณเริ่มทำงาน WebExplorer ด้วยชุดของ preferences ผู้ใช้ทางเลือก
<code>-tNumber</code>	ระบุจำนวนเรตต์เพื่อใช้สำหรับการโหลดรูปภาพ โดยที่ <i>Number</i> คือจำนวนเรตต์ตัวโหลดรูปภาพ แต่ละเรตต์ ถูกแสดงในพื้นที่สถานะของหน้าต่างหลัก สามารถระบุได้สูงสุด แปด และค่าดีฟอลต์คือสี่
<code>-q</code>	ระบุโหมด quiet คำนี้อยู่ยังหน้าต่างหัวเรื่อง WebExplorer ไม่ให้แสดงเมื่อคุณเริ่มทำงานแอปพลิเคชันและข้ามหน้าต่างการยืนยันเมื่อคุณออกจากโปรแกรม
<code>-url URL</code>	ระบุเอกสารเฉพาะเพื่อโหลดเมื่อเริ่มทำงาน WebExplorer โดยที่ <i>URL</i> คือ URL ของเอกสารที่จะโหลด หาก WebExplorer มีเอกสารโฮมถูกกำหนด URL นี้จะแทนที่เอกสารนั้น คุณไม่จำเป็นต้องนำหน้า URL ด้วยแฟล็ก <code>-url</code> หากคุณระบุ URL โดยใช้ตัวเอง WebExplorer จะยอมรับค่านั้น

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

การควบคุมสิทธิ์เข้าถึง: ผู้ใช้ใดๆ

เหตุการณ์ของระบบตรวจสอบ: N/A

## ตัวอย่าง

ในการเริ่มทำงานเบราว์เซอร์โดยไม่ให้หน้าต่างหัวเรื่องปรากฏ และตรงไปที่ Dilbert Zone URL ให้ป้อน:

```
explore -q http://www.unitedmedia.com/comics/dilbert/
```

or

```
explore -q -url http://www.unitedmedia.com/comics/dilbert/
```

## ไฟล์

ไอเท็ม  
/usr/lpp/explorer/bin/explore  
\$HOME/.explore-preferences

\$HOME/.mailcap  
\$HOME/.mimetypes

คำอธิบาย  
มีคำสั่ง explore  
มีไฟล์การกำหนดค่าเริ่มต้นที่ระบุ preferences ผู้ใช้สำหรับการตั้งค่าเช่นจำนวน  
สีที่ใช้  
มีไฟล์คอนฟิกูเรชันที่แมพ mimetype กับวิวเวอร์ภายนอก  
มีไฟล์คอนฟิกูเรชันที่ผู้ใช้กำหนดเองที่แมพ mimetype กับ วิวเวอร์ภายนอก ซึ่ง  
ถูกตั้งค่าจากไดอะล็อก Configure Viewers ไฟล์นี้แทนที่การตั้งค่า .mailcap

---

## คำสั่ง `exportfs`

### Purpose

เอ็กซ์พอร์ตและยกเลิกเอ็กซ์พอร์ต ไดเรกทอรีกับไคลเอ็นต์ NFS

### ไวยากรณ์

```
/usr/sbin/exportfs [-a] [-v] [-u] [-i] [-fFile] [-F] [-oOption [,Option ...]] [-V Exported Version] [Directory]
```

### รายละเอียด

คำสั่ง `exportfs` ทำให้ไลคัลไดเรกทอรี พร้อมใช้สำหรับประกอบเข้าของไคลเอ็นต์ Network File System (NFS) คำสั่ง นี้โดย  
ปกติถูกเรียกกระหว่างระบบเริ่มทำงานโดยไฟล์ `/etc/rc.nfs` และใช้ข้อมูลในไฟล์ `/etc/exports` เพื่อเอ็กซ์พอร์ตไดเรกทอรี ซึ่ง  
ต้องถูก ระบุตัวชื่อพาธแบบเต็ม

ไฟล์ `/etc/xtab` แสดงไดเรกทอรี ที่ถูกเอ็กซ์พอร์ตขณะนี้ เมื่อต้องการแสดงไฟล์นี้ ให้ป้อนคำสั่ง `exportfs` โดยไม่ต้องมีแฟล็ก  
หรืออาร์กิวเมนต์ เมื่อต้องการแก้ไขไฟล์หรือ แก้ไขคุณสมบัติของหนึ่งในไดเรกทอรี ผู้ใช้ `root` สามารถแก้ไขไฟล์ `/etc/exports`  
และรันคำสั่ง `exportfs` การแก้ไขดังกล่าวทำได้ตลอดเวลาอย่าแก้ไขไฟล์ `/etc/xtab` โดยตรง

### หมายเหตุ:

1. คุณไม่สามารถเอ็กซ์พอร์ตไดเรกทอรีที่เป็นไดเรกทอรีหลักหรือไดเรกทอรีย่อย ของไดเรกทอรีที่ถูกเอ็กซ์พอร์ตอยู่และอยู่  
ในระบบไฟล์เดียวกัน
2. NFS เวอร์ชัน 2 และ 3 อนุญาตให้ทั้งไดเรกทอรีและไฟล์ถูกเอ็กซ์พอร์ต เฉพาะไดเรกทอรีที่สามารถถูกเอ็กซ์พอร์ตสำหรับ  
การเข้าถึง NFS เวอร์ชัน 4
3. ถ้าสองรายการสำหรับไดเรกทอรีเดียวกันมีเวอร์ชันต่างกัน 2 (หรือ 3) และ 4 มีอยู่ในไฟล์ `/etc/exports` คำสั่ง `exportfs`  
จะเอ็กซ์พอร์ตทั้งสองรายการ
4. ถ้าอ็อปชันสำหรับ NFS เวอร์ชัน 2 (หรือ 3) และ 4 เหมือนกันสำหรับไดเรกทอรีหนึ่ง จะมีหนึ่งรายการในไฟล์ `/etc/  
exports` ที่ระบุ `-vers=3:4`.

### แฟล็ก

## ไอเท็ม

- คำอธิบาย**
- a เอ็กซ์พอร์ตไดเรกทอรีทั้งหมดที่แสดงรายการในเอ็กซ์พอร์ตไฟล์
  - v พิมพ์ชื่อของแต่ละไดเรกทอรีตามที่ถูกเอ็กซ์พอร์ตหรือยกเลิกเอ็กซ์พอร์ต
  - u ยกเลิกเอ็กซ์พอร์ตไดเรกทอรีที่คุณระบุ เมื่อใช้กับแฟล็ก -a ยกเลิกเอ็กซ์พอร์ตไดเรกทอรีที่เอ็กซ์พอร์ตทั้งหมด เมื่อใช้กับทั้งแฟล็ก -a และ -f ยกเลิกเอ็กซ์พอร์ตไดเรกทอรีทั้งหมด ในไฟล์เอ็กซ์พอร์ตที่ระบุ
  - i อนุญาตการเอ็กซ์พอร์ตไดเรกทอรีที่ไม่ได้ระบุในเอ็กซ์พอร์ตไฟล์ หรือละเว้นอ็อปชันในเอ็กซ์พอร์ตไฟล์ นอกจากแฟล็ก -f ถูกใช้เพื่อระบุไฟล์ทางเลือก คำสั่ง `exportfs` ปกติจะค้นหาไฟล์ `/etc/exports` สำหรับอ็อปชันที่เชื่อมโยงกับไดเรกทอรีที่เอ็กซ์พอร์ต
  - f File ระบุเอ็กซ์พอร์ตไฟล์ แทนไฟล์ `/etc/exports` ที่มีรายการของไดเรกทอรีที่คุณสามารถเอ็กซ์พอร์ต ไฟล์นี้ควรเป็นไปตามรูปแบบเดียวกับไฟล์ `/etc/exports` หมายเหตุ: ไฟล์สำรองนี้จะไม่ถูกใช้เพื่อเอ็กซ์พอร์ตไดเรกทอรีโดยอัตโนมัติเมื่อระบบและ NFS ถูกเริ่มงาน ไฟล์ `/etc/exports` เป็นเพียงไฟล์เดียวที่สนับสนุนการระบุไดเรกทอรีเพื่อเอ็กซ์พอร์ตขณะที่ระบบสตาร์ท
  - F ระบุว่าควรมีการบังคับการยกเลิกเอ็กซ์พอร์ต ใช้แฟล็กนี้เฉพาะกับแฟล็ก -u แฟล็กนี้ไม่มีผลเมื่อยกเลิกการเอ็กซ์พอร์ต V2/V3 ยกเลิกเอ็กซ์พอร์ต V4 อาจล้มเหลวได้เนื่องจากสภาวะที่เกี่ยวข้อง แฟล็กนี้บังคับวิธีของสภาวะที่เกี่ยวข้องกับการเอ็กซ์พอร์ต V4

### คำอธิบาย

ระบุคุณสมบัติเป็นทางเลือกสำหรับไดรฟ์ทอรัสที่กำลังถูกเอ็กซ์พอร์ต คุณสามารถป้อนได้มากกว่าหนึ่งตัวแปรโดยแยกด้วยคอมมา สำหรับ อ็อพชันที่รับพารามิเตอร์ *Client*, *Client* สามารถระบุชื่อโฮสต์, IP แอดเดรสแบบจุด, ชื่อเน็ตเวิร์ก หรือ ตัวกำหนด subnet ตัวกำหนด subnet อยู่ในรูปแบบ "@host/mask", โดยที่ *host* คือชื่อโฮสต์หรือ IP แอดเดรสแบบจุดและ *mask* ระบุจำนวนบิตที่ใช้เมื่อทำการตรวจสอบการเข้าถึง ถ้า *mask* ไม่ถูกระบุ *mask* แบบเต็มจะถูกใช้ ตัวอย่างตัวกำหนด @client.group.company.com/16 จะจับคู่ Clients ทั้งหมดบน company.com subnet ตัวกำหนด @client.group.company.com/24 จะจับคู่เฉพาะ Clients บน group.company.com subnet เลือก จากอ็อพชันต่อไปนี้:

**ro**      เอ็กซ์พอร์ตไดรฟ์ทอรัสที่มีสิทธิการใช้งานอ่านอย่างเดียว ถ้าไม่ระบุ ไดรฟ์ทอรัสจะถูกเอ็กซ์พอร์ตด้วยสิทธิการใช้งาน อ่าน/เขียน

**ro=Client[:Client]**

เอ็กซ์พอร์ตไดรฟ์ทอรัสที่มีสิทธิการใช้งานอ่านอย่างเดียวไปที่ Clients ที่ระบุ เอ็กซ์พอร์ตไดรฟ์ทอรัสที่มีสิทธิอ่าน/เขียนไปที่ Clients ที่ไม่ได้ระบุ ในรายการ รายการอ่านอย่างเดียวไม่สามารถถูกระบุได้ ถ้ามีการระบุรายการ อ่าน/เขียน

**rw**      เอ็กซ์พอร์ตไดรฟ์ทอรัสที่มีสิทธิอ่าน/เขียนไปที่ Clients ทั้งหมด

**rw=Client[:Client]**

เอ็กซ์พอร์ตไดรฟ์ทอรัสที่มีสิทธิการใช้งานอ่าน/เขียนไปที่ Clients ที่ระบุ เอ็กซ์พอร์ตไดรฟ์ทอรัสอ่านอย่างเดียวไปที่ Clients ที่ไม่ได้ระบุ ในรายการ รายการ อ่าน/เขียนไม่สามารถถูกระบุได้ ถ้ามีการระบุรายการอ่านอย่างเดียว

**anon =UID**

ใช้ค่า *UID* เป็น user ID ที่มีผล ถ้าการร้องขอมาจากผู้ใช้ root

ค่าดีฟอลต์สำหรับอ็อพชันนี้คือ -2 ใน NFS เวอร์ชัน 2 และ NFS เวอร์ชัน 3 การตั้งค่าของอ็อพชัน *anon* เป็น -1 ปิดใช้งานการเข้าถึง anonymous ดังนั้นโดยดีฟอลต์ secure NFS รับการร้องขอที่ไม่ปลอดภัยเป็น anonymous และผู้ใช้ที่ต้องการความปลอดภัยมากกว่า สามารถปิดคุณลักษณะนี้ได้โดยตั้งค่า *anon* เป็นค่า -1

**root=Client[:Client]**

อนุญาตให้มีการเข้าถึงแบบ root จากไคลเอ็นต์ที่ระบุในรายการ การนำโฮสต์ ไปใส่ในรายการ root ไม่เขียนทับซีแมนทิกส์ของอ็อพชันอื่น ตัวอย่าง อ็อพชันนี้ปฏิเสธการเข้าถึง mount จากโฮสต์ที่มีอยู่ใน รายการ root แต่ไม่มีอยู่ในรายการการเข้าถึง

**access=Client[:Client,...]**

ให้การเข้าถึง mount กับแต่ละไคลเอ็นต์ที่อยู่ในรายการ ไคลเอ็นต์สามารถเป็นได้ทั้งชื่อโฮสต์ หรือชื่อกลุ่ม net แต่ละไคลเอ็นต์ในรายการถูกตรวจสอบอันดับแรกในฐานข้อมูล /etc/netgroup จากนั้นในฐานข้อมูล /etc/hosts ค่าดีฟอลต์อนุญาตให้ทุกเครื่อง mount ไดรฟ์ทอรัสที่กำหนด

**secure**      ต้องการให้ไคลเอ็นต์ใช้โปรโตคอลที่มีความปลอดภัยมากขึ้น เมื่อเข้าถึงไดรฟ์ทอรัส

## คำอธิบาย

`sec=flavor[:flavor...]`

อ็อปชันนี้ถูกใช้เพื่อระบุรายการของเมธอดความปลอดภัยที่อาจถูกใช้เพื่อเข้าถึงไฟล์ภายใต้ไตรีกทอรีที่เอ็กซ์พอร์ต อ็อปชัน `exports` ส่วนใหญ่สามารถถูกคลัสเตอร์โดยใช้อ็อปชัน `sec` อ็อปชันที่ตามอ็อปชัน `sec` ถือว่าเป็นของอ็อปชัน `sec` ก่อนหน้าจำนวนของ `sec` stanzas สามารถระบุได้ตามต้องการ แต่แต่ละเมธอดความปลอดภัยสามารถระบุได้เพียงหนึ่งเดียว ภายในแต่ละ `sec` stanza อ็อปชัน `ro`, `rw`, `root` และ `access` ระบุได้เพียงครั้งเดียว เฉพาะอ็อปชัน `public`, `anon` และ `vers` ที่จะถือว่าเป็นโกลบอลสำหรับเอ็กซ์พอร์ต ถ้าอ็อปชัน `sec` ถูกใช้เพื่อระบุเมธอดความปลอดภัยใด ต้องถูกใช้เพื่อระบุเมธอดความปลอดภัยทั้งหมด ถ้าไม่มีอ็อปชัน `sec` รูปแบบการพิสูจน์ตัวตนทั้งหมดสามารถใช้ได้

ค่า `flavor` ที่อนุญาตได้ ได้แก่:

<code>sys</code>	การพิสูจน์ตัวตน UNIX นี้เป็นดีฟอลต์เมธอด
<code>dh</code>	การพิสูจน์ตัวตน DES
<code>none</code>	อนุญาตการร้องขอ <code>mount</code> ให้ดำเนินการด้วย <code>anonymous credentials</code> ถ้าการร้องขอ <code>mount</code> ใช้รูปแบบการพิสูจน์ตัวตนที่ไม่ระบุในการเอ็กซ์พอร์ต
<code>krb5</code>	Kerberos. การพิสูจน์ตัวตนเท่านั้น
<code>krb5i</code>	Kerberos. การพิสูจน์ตัวตนและ integrity
<code>krb5p</code>	Kerberos. การพิสูจน์ตัวตน integrity และความเป็นส่วนตัว

อ็อปชัน `secure` สามารถระบุได้ แต่ต้องไม่รวมกับอ็อปชัน `sec` ไม่ยอมรับอ็อปชัน `secure` และอาจถูก ลบ ใช้ `sec=dh` แทน

`vers=version_number[:version_number...]`

ระบุเวอร์ชัน NFS ที่สามารถเข้าถึงไตรีกทอรีที่เอ็กซ์พอร์ต เวอร์ชันที่ใช้ได้คือ 2, 3 และ 4 เวอร์ชัน 2 และ 3 ไม่สามารถถูกเลือกเป็นแยกต่างหากได้ การระบุเวอร์ชัน 2 หรือเวอร์ชัน 3 จะอนุญาตการเข้าถึงโดย NFS เวอร์ชัน 2 และ NFS เวอร์ชัน 3 เวอร์ชัน 4 สามารถถูกเลือกต่างหากได้ ดีฟอลต์คืออนุญาตให้เข้าถึงโดยใช้ NFS โพรโตคอลเวอร์ชัน 2 และ 3

`exname=external-name`

เอ็กซ์พอร์ตไตรีกทอรีตามชื่อยานนอกที่ระบุชื่อยานนอก ต้องเริ่มต้นด้วยชื่อ `nfsroot` ดูที่รายละเอียดของไฟล์ `/etc/exports` สำหรับรายละเอียดของ ชื่อ `nfsroot` อ็อปชันนี้ใช้เฉพาะกับไตรีกทอรีที่เอ็กซ์พอร์ต สำหรับการเข้าถึงโดยโปรโตคอล NFS เวอร์ชัน 4

`deleg={yes|no}`

เปิดใช้งานหรือปิดใช้งานการมอบหมายไฟล์สำหรับการเอ็กซ์พอร์ตที่ระบุ อ็อปชันนี้เขียนทับการเปิดใช้การกระจายข้อมูล `system-wide` สำหรับการเอ็กซ์พอร์ตนี้ การเปิด ใช้ `system-wide` ดำเนินการผ่าน `nfsd`

## คำอธิบาย

**refer=rootpath@host[+host][:rootpath@host[+host]]**

การอ้างอิงเนมสเปซจะถูกสร้างที่พาธที่ระบุ การอ้างอิง กำหนดทิศทางโคลเอ็นต์ไปที่ตำแหน่งสำรองที่ระบุ ซึ่งโคลเอ็นต์สามารถดำเนิน การดำเนินการต่อได้ การอ้างอิงเป็นอ็อบเจ็กต์พิเศษ ถ้ามีอ็อบเจ็กต์ที่ไม่มีการอ้างอิงอยู่ที่พาธที่ระบุ การเอ็กซ์พอร์ตจะทำได้ และข้อความแสดงความผิดพลาดจะถูกพิมพ์ออกมา ถ้าไม่มีข้อมูลใดที่พาธที่ระบุ อ็อบเจ็กต์การอ้างอิงจะถูกสร้างขึ้น ซึ่งรวมถึงไดเรกทอรีชื่อพาธที่นำไปสู่อ็อบเจ็กต์ การอ้างอิงหลายการอ้างอิง สามารถถูกสร้างได้ภายในระบบไฟล์ ไม่สามารถระบุการอ้างอิงสำหรับ **nfsroot** ชื่อ **localhost** ไม่สามารถใช้เป็น **hostname** อ็อพชัน **refer** นี้ใช้ได้เฉพาะสำหรับการเอ็กซ์พอร์ตเวอร์ชัน 4 ถ้าค่ากำหนด เอ็กซ์พอร์ตอนุญาตการเข้าถึงเวอร์ชัน 2 หรือเวอร์ชัน 3 ข้อความแสดงความผิดพลาดจะถูก พิมพ์และการเอ็กซ์พอร์ตจะไม่สามารถทำได้ การยกเลิกการเอ็กซ์พอร์ตอ็อบเจ็กต์ การอ้างอิงมีผลในการลบข้อมูลตำแหน่งการอ้างอิงจาก อ็อบเจ็กต์การอ้างอิง ตัวอ็อบเจ็กต์เองจะไม่ถูกลบโดยการยกเลิกการเอ็กซ์พอร์ต ใช้ **rm** ถ้าคุณต้องการลบอ็อบเจ็กต์ ผู้ดูแลระบบต้องตรวจสอบว่า ข้อมูลที่เหมาะสมมีอยู่ที่เซิร์ฟเวอร์การอ้างอิง อ็อพชันนี้ใช้ได้เฉพาะ บน AIX 5L Version 5.3 with the 5300-03

Recommended Maintenance package หรือสูงกว่า

**หมายเหตุ:** การเอ็กซ์พอร์ตการอ้างอิงทำได้ เฉพาะ ถ้าเรพลิเคชันถูกเปิดใช้บนเซิร์ฟเวอร์ ใช้ **chnfs -R on** เพื่อ เปิดใช้เรพลิเคชัน

**replicas=rootpath@host[+host][:rootpath@host[+host]]**

ข้อมูลตำแหน่งเรพลิคาจะถูกเชื่อมโยงกับพาธเอ็กซ์พอร์ต ข้อมูลเรพลิคาสามารถถูกใช้โดยโคลเอ็นต์ NFS เวอร์ชัน 4 เพื่อเปลี่ยนทิศทางดำเนินการ ไปที่ตำแหน่งสำรองที่ระบุ ถ้าเซิร์ฟเวอร์ปัจจุบันไม่พร้อมใช้งาน ผู้ดูแลระบบควรตรวจสอบว่าข้อมูลที่เหมาะสมมีอยู่ที่เซิร์ฟเวอร์ เรพลิคา เนื่องจากข้อมูลเรพลิคาใช้กับระบบไฟล์ทั้งหมด พาธที่ระบุต้องเป็น **root** ของระบบไฟล์ ถ้าพาธไม่ใช่ **root** ของระบบไฟล์ การเอ็กซ์พอร์ตจะทำได้ และข้อความแสดงความผิดพลาดจะถูกพิมพ์ออกมา ชื่อ **localhost** ไม่สามารถใช้เป็น **hostname** อ็อพชัน **replicas** นี้มีความหมายเฉพาะสำหรับการเอ็กซ์พอร์ตเวอร์ชัน 4 ถ้าอ็อพชัน ถูกใช้ในการเอ็กซ์พอร์ตที่อนุญาตการเข้าถึงเวอร์ชัน 2 หรือเวอร์ชัน 3 การดำเนินการ สามารถทำได้ แต่ข้อมูลเรพลิคาจะถูกละเว้นโดยเซิร์ฟเวอร์เวอร์ชัน 2 และเวอร์ชัน 3 ถ้าไดเรกทอรีที่กำลังถูกเอ็กซ์พอร์ตไม่ได้อยู่ในรายการเรพลิคา รายการ **exported directory@current host** จะถูกเพิ่มเป็น ตำแหน่งเรพลิคาแรก อ็อพชันนี้ใช้ได้เฉพาะ บน AIX 5.3 with 5300-03 หรือสูงกว่า การเอ็กซ์พอร์ตเรพลิคาทำได้ เฉพาะ ถ้าเรพลิเคชันถูกเปิดใช้บนเซิร์ฟเวอร์ โดยดีฟอลต์ เรพลิเคชันไม่ถูกเปิดใช้ ถ้าการเอ็กซ์พอร์ต เรพลิคาจะถูกกระทำเมื่อระบบบูต เรพลิเคชันควรถูกเปิดใช้โดยใช้คำสั่ง **chnfs -R on** ไม่สามารถระบุตำแหน่งเรพลิคาสำหรับ **nfsroot** ซึ่งทำได้เพียงการใช้ **chnfs -R host[+host]** ถ้าโฮสต์ปัจจุบันไม่ถูกระบุในรายการ โฮสต์จะถูกเพิ่มเป็นโฮสต์เรพลิคาแรก **rootpath** ไม่จำเป็น หรือใช้ไม่ได้ในกรณีนี้ เนื่องจาก **nfsroot** ถูกทำเรพลิคาเฉพาะกับ **nfsroots** ของโฮสต์ที่ระบุ โปรแกรม **chnfs** สามารถถูกใช้เพื่อเปิดหรือปิดใช้งานเรพลิเคชัน การเปลี่ยนโหมดเรพลิเคชันสามารถทำได้เฉพาะถ้าไม่มีเอ็กซ์พอร์ต NFS เวอร์ชัน 4 แอ็คทีฟอยู่ ถ้าโหมดเรพลิเคชันของเซิร์ฟเวอร์ถูกเปลี่ยน การจัดการไฟล์ที่เรียกโดยเซิร์ฟเวอร์ ระหว่างโหมดเรพลิเคชันก่อนหน้านี้จะไม่ถูกรับรองโดยเซิร์ฟเวอร์ ซึ่ง สามารถทำให้แอฟพลิเคชันมีข้อผิดพลาดบนโคลเอ็นต์ที่เก็บการจัดการไฟล์เก่าไว้ โปรดระวัง เมื่อทำการเปลี่ยนโหมดเรพลิเคชันของเซิร์ฟเวอร์ ถ้าเป็นไปได้อื่นทั้งหมด ที่ประกอบเข้ากับเซิร์ฟเวอร์ควรถูกยกเลิก ก่อนเปลี่ยนโหมดเรพลิเคชัน ของเซิร์ฟเวอร์ ข้อมูลตำแหน่งเรพลิคาที่เชื่อมโยงกับไดเรกทอรี สามารถถูกเปลี่ยนแปลงโดยแก้ไขรายการเรพลิคาและเอ็กซ์พอร์ตไดเรกทอรีอีกครั้ง ข้อมูลเรพลิคาใหม่แทนที่ข้อมูลเรพลิคาเก่า โคลเอ็นต์ NFS ถูกคาดหวังจะรีเฟรชข้อมูลเรพลิคาเป็นประจำ ถ้า เซิร์ฟเวอร์เปลี่ยนข้อมูลเรพลิคาสำหรับการเอ็กซ์พอร์ต อาจใช้เวลาชักระยะ ที่โคลเอ็นต์จะรับรู้ ซึ่งไม่เป็นปัญหากมากนัก ถ้าตำแหน่งเรพลิคาใหม่ถูกเพิ่ม เนื่องจากโคลเอ็นต์ที่เก็บข้อมูลเก่ายังมีข้อมูลเรพลิคาที่ถูกต้อง ถ้ายังไม่สมบูรณ์ การลบข้อมูลเรพลิคาอาจมีปัญหากเกิดขึ้นได้ เนื่องจากมีอาจผลให้โคลเอ็นต์เก็บค่าข้อมูลเรพลิคาที่ไม่ถูกต้อง ในช่วงเวลาหนึ่ง เพื่อช่วยโคลเอ็นต์ในการตรวจหาข้อมูลใหม่ **exportfs** จะพยายามตรวจสอบไดเรกทอรีที่เรพลิคา ซึ่งจะเปลี่ยนการประทับเวลา บนไดเรกทอรี ซึ่งเป็นผลให้โคลเอ็นต์ดึงข้อมูลแอ็คทีฟของไดเรกทอรี อีกครั้ง การดำเนินการนี้อาจเป็นไปได้ ถ้าระบบไฟล์ที่ทำเรพลิคาเป็นแบบอ่านอย่างเดียว เมื่อทำการเปลี่ยนข้อมูลเรพลิคาสำหรับไดเรกทอรี ให้ระวังว่าอาจมีเวลาแฝง ระหว่างการเปลี่ยนข้อมูล และการที่โคลเอ็นต์รับรู้ถึงข้อมูลใหม่

## ไอเท็ม

-o Options (continued)

## คำอธิบาย

<b>noauto</b>	รับค่ากำหนดเรพลิคาตามที่เป็น ไม่ทำการแทรกชื่อโฮสต์ หลักเป็นหนึ่งในตำแหน่งเรพลิคาโดยอัตโนมัติ ถ้าไม่มีการระบุ
<b>scatter</b>	กำหนดวิธีที่รายการตำแหน่งสำรองถูกสร้างจากเซิร์ฟเวอร์ ที่ระบุบนอ็อปชัน refer หรือ replicas ถ้าอ็อปชัน <b>noauto</b> ไม่ถูกใช้ รายการตำแหน่งสำรองจะรวมชื่อโฮสต์หลักเป็นหนึ่งใน ตำแหน่งเรพลิคา อ็อปชัน <b>scatter</b> นี้ใช้เฉพาะกับไดเรกทอรีที่เอ็กซ์พอร์ต สำหรับการเข้าถึงโดยโปรโตคอล NFS เวอร์ชัน 4 อ็อปชัน <b>scatter</b> มีค่าที่ใช้ได้สามค่า:  <b>full</b> เซิร์ฟเวอร์ทั้งหมดถูกกระจายเพื่อฟอร์มการรวมกันของตำแหน่ง สำรอง  <b>partial</b> ตำแหน่งแรกของการรวมทั้งหมดกำหนดตายตัวให้กับเซิร์ฟเวอร์แรก ที่ระบุบนอ็อปชัน refer หรือ replicas ส่วนที่เหลือของตำแหน่งและตำแหน่งแรกถูกกระจาย เหมือนกับถูกกระจายโดย ใช้เมธอด scatter=full  <b>none</b> ไม่มีการใช้การกระจาย ค่ายังสามารถถูกใช้เพื่อปิดใช้งานการกระจาย ถ้ามีการเปิดใช้ก่อนหน้า นี้

เมื่อใดก็ตามที่แอตทริบิวต์ของไคลเอ็นต์เปลี่ยนแปลง รายการเอ็กซ์พอร์ตทั้งหมดที่มี ไคลเอ็นต์เป็นพารามิเตอร์ควร ถูกเอ็กซ์พอร์ตอีกครั้ง เหตุการณ์ที่สามารถเปลี่ยน แอตทริบิวต์ของไคลเอ็นต์รวมถึงการแก้ไข netgroup หรือการ เปลี่ยน IP แอดเดรสของไคลเอ็นต์ ความล้มเหลวในการกระทำดังกล่าวสามารถมีผลให้เซิร์ฟเวอร์ ใช้ข้อมูลไคลเอ็นต์ เก่า ระบุหมายเลขเวอร์ชัน หมายเลขเวอร์ชัน ที่ใช้ได้คือ 2, 3 และ 4

-V Exported Version

## ความเข้ากันได้กับ Solaris

คำสั่ง **exportfs** อาจถูกเรียกเป็น **share**, **shareall**, **unshare**, หรือ **unshareall** เมื่อคำสั่ง **exportfs** ถูกเรียกเป็น **share** หรือ **shareall**, การทำงานจะเหมือนกับ **exportfs** และ **exportfs -a** ตามลำดับ ยกเว้นว่าอ็อปชัน **sec** ต้องถูกใช้เพื่อระบุเมธอดความปลอดภัย เมื่อคำสั่ง **exportfs** ถูกเรียกเป็น **unshare** หรือ **unshareall**, การทำงานเหมือนกับ **exportfs -u** และ **exportfs -u -a**, ตามลำดับ

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

1. เมื่อต้องการเอ็กซ์พอร์ตไดเรกทอรีทั้งหมดในไฟล์ **/etc/exports** ให้ป้อน:

```
exportfs -a
```

2. เมื่อต้องการเอ็กซ์พอร์ตหนึ่งไดเรกทอรีในไฟล์ **/etc/exports** ให้ป้อน:

```
exportfs /home/notes
```

ในตัวอย่างนี้ไดเรกทอรี **/home/notes** ถูกเอ็กซ์พอร์ต

หมายเหตุ: เพื่อให้คำสั่งนี้ทำงานไดเรกทอรี **/home/notes** directory ต้องถูกระบุในไฟล์ **/etc/exports**

3. เมื่อต้องการยกเลิกการเอ็กซ์พอร์ตไดเรกทอรีให้ป้อน:

```
exportfs -u /home/notes
```

ในตัวอย่างนี้ไดเรกทอรี /home/notes ถูกยกเลิกเอ็กซ์พอร์ต

4. เมื่อต้องการแสดงชื่อของไดเรกทอรีที่กำลังถูกเอ็กซ์พอร์ตขณะนี้ให้ป้อน:

```
exportfs -v
```

5. เมื่อต้องการเอ็กซ์พอร์ตไดเรกทอรีที่ไม่ถูกระบุในไฟล์ /etc/exports ให้ป้อน:

```
exportfs -i /home/zeus
```

ในตัวอย่างนี้ไดเรกทอรี /home/zeus ถูกเอ็กซ์พอร์ตโดยไม่มีข้อจำกัด

6. เมื่อต้องการเอ็กซ์พอร์ตไดเรกทอรีและให้สิทธิ์สมาชิก netgroup เพื่อเข้าถึงไดเรกทอรีนี้ให้ป้อน:

```
exportfs access=cowboys:oilers /home/notes -o
```

ในตัวอย่างนี้ไดเรกทอรี /home/notes ถูกเอ็กซ์พอร์ตและอนุญาตให้ผู้ใช้ของเครื่องโฮสต์ cowboys และ oilers มีสิทธิ์เข้าถึง

7. เมื่อต้องการเอ็กซ์พอร์ตไดเรกทอรีที่มีอ็อปชันต่างจากไฟล์ /etc/exports ให้ป้อน:

```
exportfs -i -o root=zorro:silver /directory
```

ในตัวอย่างนี้ไดเรกทอรี /directory ถูกเอ็กซ์พอร์ต และให้สิทธิ์เข้าถึงของผู้ใช้ root กับเครื่องโฮสต์ zorro และ silver ไม่ว่า สิทธิการเข้าถึงที่ระบุในไฟล์ /etc/exports จะเป็นอย่างไร

8. เมื่อต้องการเอ็กซ์พอร์ตไดเรกทอรี /common/docs ที่มีสิทธิ์เขียน ไปที่ไคลเอ็นต์โดยใช้การพิสูจน์ตัวตน Kerberos แต่ให้สิทธิ์อ่านอย่างเดียว กับไคลเอ็นต์โดยใช้การพิสูจน์ตัวตน UNIX ให้เพิ่มข้อความต่อไปนี้กับไฟล์ /etc/exports :

```
/common/docs -sec=krb5,rw,sec=sys,ro
```

แล้วป้อน exportfs /common/docs เพื่อทำการเอ็กซ์พอร์ต

9. เมื่อต้องการสร้างการอ้างอิงที่ไดเรกทอรี /usr/info ไปที่ /usr/info บนโฮสต์ infoserver ให้เพิ่มบรรทัดต่อไปนี้กับ /etc/exports แล้วเอ็กซ์พอร์ต /usr/info:

```
/usr/info -vers=4,refer=/usr/info@infoserver
```

10. เมื่อต้องการระบุเรพลิคาสำหรับไดเรกทอรี /common/info ที่โฮสต์ backup1 และ backup2, เพิ่มบรรทัดต่อไปนี้ที่ /etc/exports แล้วเอ็กซ์พอร์ต /common/info:

```
/common/info -vers=4,replicas=/common/info@backup1:/common/info@backup2,<other options>
```

11. เมื่อต้องการเอ็กซ์พอร์ตไดเรกทอรี /common/docs ที่มีทั้งเวอร์ชัน 3 และเวอร์ชัน 4 ให้ป้อนคำสั่งต่อไปนี้:

```
exportfs -V 3:4 /common/docs
```

12. เมื่อต้องการเอ็กซ์พอร์ตรายการเวอร์ชัน 4 ในไฟล์ /etc/exports ให้ป้อนคำสั่งต่อไปนี้:

```
exportfs -a -V 4
```

13. เมื่อต้องการยกเลิกเอ็กซ์พอร์ตไดเรกทอรี /common/docs เฉพาะ เวอร์ชัน 3 ให้ป้อนคำสั่งต่อไปนี้:

```
exportfs -u -V 3 /common/docs
```

14. เมื่อต้องการยกเลิกเอ็กซ์พอร์ตรายการเวอร์ชัน 3 ในไฟล์ /etc/xtab ให้ป้อนคำสั่งต่อไปนี้:

```
exportfs -ua -V 3
```

15. เมื่อต้องการระบุการอ้างอิงสำหรับไดเรกทอรี /common/docs ที่โฮสต์ชื่อ s1, s2 และ s3 และกระจายการอ้างอิงแบบเต็ม เพิ่ม บรรทัดต่อไปนี้ที่ไฟล์ /etc/exports จากนั้นเอ็กซ์พอร์ต ไดเรกทอรี /common/docs:

```
/common/docs -vers=4,refer=/common/docs@s1:/common/docs@s2:/common/docs@s3,scatter=full
```

16. เมื่อต้องการระบุสำเนาสำหรับไดเรกทอรี `/common/docs` ที่โฮสต์ชื่อ `s1`, `s2`, `s3` และ `s4` และกระจายการอ้างอิงแบบบางส่วน (เซิร์ฟเวอร์ fail over แรกคือ `s1` สำหรับการรวมทั้งหมด) เพิ่มบรรทัดต่อไปนี้ในไฟล์ `/etc/exports` จากนั้นเอ็กซ์พอร์ตไดเรกทอรี `/common/docs`:

```
/common/docs -vers=4,noauto,replicas=/common/docs@s1:/common/docs@s2:/common/docs@s3:/common/docs@s4,scatter=partial
```

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/etc/exports</code>	แสดงไดเรกทอรีที่เซิร์ฟเวอร์สามารถเอ็กซ์พอร์ต
<code>/etc/xtab</code>	แสดงไดเรกทอรีที่เอ็กซ์พอร์ตในปัจจุบัน
<code>/etc/hosts</code>	มีรายการสำหรับแต่ละโฮสต์บนเน็ตเวิร์ก
<code>/etc/netgroup</code>	มีข้อมูลเกี่ยวกับแต่ละกลุ่มผู้ใช้บนเน็ตเวิร์ก
<code>/etc/rc.nfs</code>	มีสคริปต์เริ่มทำงานสำหรับ NFS และ NIS daemons

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `chmfsexp`

คำสั่ง `mknfsexp`

วิธีเอ็กซ์พอร์ตระบบไฟล์โดยใช้ Secure NFS

รายการของคำสั่ง NFS

## คำสั่ง `exportvg`

### วัตถุประสงค์

เอ็กซ์พอร์ตนิยามของกลุ่มวอลุ่มจากชุดของฟิสิคัลวอลุ่ม

### ไวยากรณ์

```
exportvg VolumeGroup
```

### คำอธิบาย

คำสั่ง `exportvg` ลบ นิยามของกลุ่มวอลุ่มที่ระบุโดยพารามิเตอร์ `VolumeGroup` ออกจากระบบ เนื่องจากระบบจะทราบวากลุ่มวอลุ่ม และเนื้อหาของกลุ่มวอลุ่มถูกลบออก ทำให้กลุ่มวอลุ่มที่เอ็กซ์พอร์ตไม่สามารถเข้าถึงได้อีกต่อไป คำสั่ง `exportvg` ไม่ได้แก้ไขข้อมูลผู้ใช้ใดๆ ในกลุ่มวอลุ่ม

กลุ่มวอลุ่มเป็นรีซอร์สที่ไม่มีการแบ่งใช้ภายในระบบ จึงไม่ควรถูกเข้าถึงโดยตัวประมวลผลอื่นจนกว่าจะถูกเอ็กซ์พอร์ตอย่างชัดเจน จากตัวประมวลผลปัจจุบัน และอิมพอร์ตบนตัวประมวลผล การใช้งานหลัก ของคำสั่ง `exportvg` คู่กับคำสั่ง `importvg` คืออนุญาตให้วอลุ่มที่เคลื่อนย้ายได้สามารถ แลกเปลี่ยนระหว่างตัวประมวลผล กลุ่มวอลุ่มทั้งกลุ่มเท่านั้นที่สามารถเอ็กซ์พอร์ต ไม่ใช่แต่ละฟิสิคัลวอลุ่ม

การใช้คำสั่ง `exportvg` และ คำสั่ง `importvg` คุณยังสามารถ สลับความเป็นเจ้าของข้อมูลบนฟิสิคัลวอลุ่มที่แบ่งใช้ระหว่างตัวประมวลผลสองตัว

**หมายเหตุ:** ในการใช้คำสั่งนี้ คุณ ต้องมีสิทธิ `root` หรือเป็นสมาชิกของกลุ่ม `system`

คุณสามารถใช้พาดวง `smit exportvg` ของ System Management Interface Tool (SMIT) เพื่อรันคำสั่งนี้

#### Notes:

1. กลุ่มวอลุ่มที่มีวอลุ่มพื้นที่การสลับหน้าอยู่จะไม่สามารถถูกเอ็กซ์พอร์ต ขณะที่พื้นที่การสลับหน้าแฉีกที่พอยู่ ก่อนการเอ็กซ์พอร์ตกลุ่มวอลุ่มที่มี วอลุ่มพื้นที่การสลับหน้าแฉีกที่พ ตรวจสอบให้แน่ใจว่าพื้นที่การสลับหน้าไม่ถูกเรียกทำงาน โดยอัตโนมัติในตอนทำการกำหนดค่าเริ่มต้น จากนั้นบูตระบบใหม่
2. ข้อมูลจุดที่เม้าท์ของโลจิคัลวอลุ่มจะหายไปจาก LVCB (logical volume control block) หากมีความยาวมากกว่า 128 อักขระ โปรดจัดทำข้อมูลบันทึกจุดที่เม้าท์ที่มีความยาวมากกว่า 128 อักขระ เนื่องจากคุณจะต้องแก้ไขไฟล์ `/etc/filesystems` ด้วยตนเอง เมื่อทำการเรียกใช้คำสั่ง `importvg` เพื่ออิมพอร์ตกลุ่มวอลุ่ม นี้เสร็จเรียบร้อยแล้ว

#### สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

#### ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

#### ตัวอย่าง

ในการลบกลุ่มวอลุ่ม `vg02` ออกจากระบบให้ป้อน:

```
exportvg vg02
```

**หมายเหตุ:** กลุ่มวอลุ่มต้องถูกทำให้ขึ้นต่อกัน ก่อนการเอ็กซ์พอร์ต

นิยามของ `vg02` ถูกลบออกจากระบบและกลุ่มวอลุ่มไม่สามารถเข้าถึงได้

#### ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/sbin</code>	ไดเรกทอรีที่มีคำสั่ง <code>exportvg</code> อยู่

#### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `importvg`

คำสั่ง `varyonvg`

หน่วยเก็บโลจิคัลวอลุ่ม

เครื่องมืออินเทอร์เฟซการจัดการระบบ

---

## คำสั่ง expr

### วัตถุประสงค์

หาค่าอาร์กิวเมนต์ของนิพจน์

### ไวยากรณ์

`expr Expression`

### คำอธิบาย

คำสั่ง `expr` อ่านพารามิเตอร์ `Expression` หาค่าพารามิเตอร์ และเขียนผลลัพธ์ไปยังเอาต์พุตมาตรฐาน

คุณต้องใช้กฎต่อไปนี้กับพารามิเตอร์ `Expression`:

- คั่นแต่ละเทอมด้วยช่องว่าง
- นำหน้าอักขระที่พิเศษสำหรับเชลล์ด้วย \ (แบ็กสแลช)
- ใส่เครื่องหมายคำพูดสตริงที่มีช่องว่างหรืออักขระ พิเศษอื่นๆ

เลขจำนวนเต็มอาจถูกนำหน้าด้วยเครื่องหมายขีดคั่น สำหรับภายใน จำนวนเต็มจะถือเป็นตัวเลขส่วนเติมเต็มสองตัว 32 บิต

หมายเหตุ: คำสั่ง `expr` ส่งคืนค่า 0 เพื่อระบุค่าศูนย์ แทนสตริง null

รายการต่อไปนี้อธิบายตัวดำเนินการและคีย์เวิร์ด พารามิเตอร์ `Expression` อักขระที่จำเป็นต้องยกเว้น จะถูกนำหน้าด้วย \ (แบ็กสแลช) รายการถูกแสดงตามลำดับการมาก่อน ของการเพิ่ม โดยที่ตัวดำเนินการมาก่อนเท่ากับจัดกลุ่มอยู่ใน { } (วงเล็บปีกกา):

#### ไอเท็ม

`Expression1 \ Expression2`

`Expression1 \& Expression2`

`Expression1 { =, \>, \>=, \<, \<=, != } Expression2`

`Expression1 { +, - } Expression2`

`Expression1 { \*, /, % } Expression2`

#### คำอธิบาย

ส่งคืน `Expression1` หากไม่เป็นค่า null หรือค่า 0 มิฉะนั้น จะส่งคืน

`Expression2`

ส่งคืน `Expression1` หาก นิพจน์ทั้งสองไม่เป็นค่า null หรือค่า 0 มิฉะนั้นจะส่งคืน ค่า 0

ส่งคืนผลลัพธ์ของการเปรียบเทียบเลขจำนวนเต็มหาก นิพจน์ทั้งสองเป็นเลขจำนวนเต็ม มิฉะนั้นจะส่งคืนผลลัพธ์การเปรียบเทียบสตริง

เพิ่มหรือลบอาร์กิวเมนต์ที่เป็นค่าเลขจำนวนเต็ม

คูณหาร หรือแสดงเศษเหลือจาก การหารอาร์กิวเมนต์ที่เป็นค่าเลขจำนวนเต็ม

## ไอเท็ม

*Expression1* : *Expression2*

## คำอธิบาย

เปรียบเทียบสตริงที่เป็นผลลัพธ์จากการหาค่าของ *Expression1* กับรูปแบบนิพจน์ปกติที่เป็นผลลัพธ์จากการหาค่า *Expression2* ไวยากรณ์นิพจน์ทั่วไปเหมือนกับของคำสั่ง `ed` ยกเว้นว่า รูปแบบทั้งหมดจะถูกยึดกับการเริ่มต้นสตริง (นั่นคือ เฉพาะ ลำดับที่ขึ้นต้นด้วยอักขระตัวแรกของสตริงที่ถูกจับคู่โดยนิพจน์ปกติเท่านั้น) ดังนั้น `^` (caret) ไม่ใช่อักขระพิเศษในบริบทนี้

โดยปกติตัวดำเนินการจับคู่จะส่งคืนจำนวนอักขระ ที่ตรง (0 เมื่อล้มเหลว) หากรูปแบบมีนิพจน์ย่อย นั่นคือ:

```
\( Expression \)
```

ดังนั้นสตริงที่มีอักขระที่ตรงกันจริงจะถูกส่งคืน

ลำดับการเรียงสามารถกำหนดคลาส equivalence สำหรับ ใช้ในช่วงอักขระ ดูที่ "การทำความเข้าใจตัวแปรสถานะแวดล้อม Locale" ใน *National Language Support Guide and Reference* เพื่อข้อมูลเพิ่มเติม เกี่ยวกับลำดับการเรียง และคลาส equivalence

**หมายเหตุ:** สตริง อาร์กิวเมนต์ต่อไปนี้ถูกขยายเกินค่ามาตรฐาน และลักษณะการทำงาน อาจแตกต่างกันตามระบบปฏิบัติการ สตริงอาร์กิวเมนต์เหล่านี้ ไม่สามารถเคลื่อนย้ายได้

## ไอเท็ม

`match String1 String2`

`length String1`

`index String1 String2`

`substr String1 StartPosition Length`

## คำอธิบาย

เหมือนกับ *Expression1* : *Expression2*.

ส่งกลับค่าความยาวของ *String1*

ส่งกลับตำแหน่งแรกใน *String1* ที่มีอักขระใดๆ ใน *String2* อยู่

ส่งคืนสตริงที่เริ่มต้นด้วยอักขระที่ *StartPosition* ใน *String1*

และต่อเนื่องไปเป็นจำนวน *Length* อักขระ

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย

0 พารามิเตอร์ *Expression* ได้ค่าที่ไม่ใช่ null หรือ 0

1 พารามิเตอร์ *Expression* ได้ค่าที่เป็น null หรือ 0

2 พารามิเตอร์ *Expression* ไม่ถูกต้อง

>2 มีข้อผิดพลาดเกิดขึ้น

**หมายเหตุ:** หลังจากการประมวลผล พารามิเตอร์โดยเซลล์ คำสั่ง `expr` ไม่สามารถแยกความแตกต่าง ระหว่างตัวดำเนินการ และตัวถูกดำเนินการยกเว้นโดยค่า ดังนั้น หากค่าของ `$a` เป็น `j` คำสั่ง:

```
expr $a = j
```

จะเหมือน:

```
expr j = j
```

หลังจากเซลล์ส่งอาร์กิวเมนต์ไปยังคำสั่ง `expr` ต่อไปนี้จะเป็น true เช่นกัน:

```
expr X$a = Xj
```

## ตัวอย่าง

1. ในการแก้ไขตัวแปรเชลล์ให้ป้อน:

```
COUNT=`expr $COUNT + 1`
```

ค่านี้เพิ่ม 1 ให้กับ ตัวแปรเชลล์ \$COUNT คำสั่ง `expr` อยู่ในเครื่องหมาย grave accents ซึ่งทำให้เชลล์แทนค่าเอาต์พุตมาตรฐานจากคำสั่ง `expr` ลงในคำสั่ง `COUNT=` ตัวแปร \$COUNT ต้องถูกเตรียมข้อมูลเบื้องต้นก่อนการใช้

2. ในการหาค่าความยาวของตัวแปรเชลล์ \$STR ให้ป้อน:

```
LENGTH=`expr $STR : ".*"`
```

ค่านี้จะตั้งค่าตัวแปร LENGTH เป็นค่าที่กำหนดโดยตัวดำเนินการ: (โคลอน) รูปแบบ .\* (จุด, เครื่องหมายดอกจัน) จะจับคู่สตริงใดๆ ตั้งแต่เริ่มต้นถึงสิ้นสุด ดังนั้นตัวดำเนินการโคลอนจะให้ค่าความยาวของตัวแปร \$STR เป็นจำนวนอักขระที่มีค่าตรงไปตรงมาว่า .\* ต้องอยู่ในเครื่องหมายคำพูดเพื่อป้องกันมิให้เชลล์ถือว่า \* (เครื่องหมายดอกจัน) เป็นอักขระการจับคู่รูปแบบ เครื่องหมายคำพูดมิได้เป็นส่วนหนึ่งของรูปแบบ

หากตัวแปร \$STR ถูกตั้งค่าเป็นสตริง null หรือมี white space (ช่องว่าง หรือแท็บ) ใดๆ คำสั่งจะแสดงข้อความแสดงข้อผิดพลาด `expr: syntax error` ซึ่งเกิดขึ้นเนื่องจากเชลล์ไม่ส่งค่าสตริง null ปกติไปยังคำสั่ง ในกรณีนี้ คำสั่ง `expr` เห็นเฉพาะ:

```
:.*
```

ดังนั้น เชลล์จะลบเครื่องหมายคำพูดเดี่ยวออก ซึ่งจะไม่สามารถทำงานได้เนื่องจากตัวดำเนินการ โคลอนต้องมีสองค่า ปัญหาได้รับการแก้ไขโดยการใส่เครื่องหมายคำพูดคู่ให้ตัวแปรเชลล์:

```
LENGTH=`expr "$STR" : ".*"`
```

ถึงตอนนี้หากค่าของตัวแปร \$STR เป็น null ตัวแปร LENGTH จะถูกตั้งค่าเป็นค่า 0 โดยทั่วไปแล้ว ขอแนะนำให้ใส่เครื่องหมายคำพูดคู่แก่ตัวแปรเชลล์ อย่าใส่เครื่องหมายคำพูดเดี่ยวแก่ตัวแปรเชลล์

3. ในการใช้ส่วนของสตริงให้ป้อน:

```
FLAG=`expr "$FLAG" : "-*\(.*\)"`
```

คำสั่งนี้จะลบเครื่องหมายขีดคั่นที่นำหน้า หากมี ออกจากตัวแปร \$FLAG ตัวดำเนินการโคลอน ให้ส่วนของตัวแปร FLAG ที่จับคู่โดย นิพจน์ย่อยอยู่ในระหว่างอักขระ \ ( และ \) (แบ็กสแลช, วงเล็บ เปิด และแบ็กสแลช, วงเล็บ ปิด) หากคุณไม่ใส่อักขระนิพจน์ย่อย \ ( และ \) ตัวดำเนินการโคลอนจะแสดงจำนวนอักขระที่ตรง

หากตัวแปร \$FLAG ถูกตั้งค่าเป็น - (เครื่องหมายขีดคั่น) คำสั่งจะแสดงข้อความข้อผิดพลาดทางไวยากรณ์ ซึ่งเกิดขึ้นเนื่องจากเชลล์แทนค่าของตัวแปร \$FLAG ก่อนการรันคำสั่ง `expr` คำสั่ง `expr` ไม่ทราบว่าเครื่องหมายขีดคั่นเป็นค่า ของตัวแปร โดยคำสั่งสามารถเห็นเฉพาะ:

```
- : -*\(.*\)
```

และ แปลความหมายเครื่องหมายขีดคั่นแรกเป็นตัวดำเนินการลบ ในการกำจัดปัญหานี้ให้ใช้:

```
FLAG=`expr "x$FLAG" : "x-*\(.*)"`
```

4. ในการใช้คำสั่ง `expr` ใน ประโยคคำสั่ง `if` ให้ป้อน:

```
if expr "$ANSWER" : "[yY]" >/dev/null
    ดังนั้น
    echo ANSWER begins with "y" or "Y"
fi
```

หากตัวแปร \$ANSWER ขึ้นต้นด้วย y หรือ Y ส่วน then ของประโยคคำสั่ง if จะถูกดำเนินการ หาก มีการจับคู่ตรง ผลลัพธ์ของ นิพจน์คือ 1 และคำสั่ง expr ส่งคืนค่าการออกเป็น 0 ซึ่งจะถูกละเป็น ค่าลอจิคัล True โดยประโยคคำสั่ง if หากการจับคู่ ไม่ตรง ผลลัพธ์เป็น 0 และค่าการออกเป็น 1 (False)

การเปลี่ยนทิศทางเอาต์พุตมาตรฐานของคำสั่ง expr ไปยังไฟล์พิเศษ /dev/null จะทิ้งผลลัพธ์ของนิพจน์ หากคุณไม่ เปลี่ยนทิศทาง ผลลัพธ์จะถูกเขียนไปยังเอาต์พุต มาตรฐาน ซึ่งโดยปกติคือหน้าจอเวิร์กสเตชัน

5. พิจารณานิพจน์ต่อไปนี้:

```
expr "$STR" = "="
```

หากตัวแปร \$STR มีค่า = (เครื่องหมายเท่ากับ) หลังจากเซลล์ประมวลผลคำสั่งนี้ คำสั่ง expr จะเป็นนิพจน์:

```
= = =
```

คำสั่ง expr แปลความหมาย คำนี้เป็นตัวดำเนินการ = สามตัวในแถวและแสดงข้อความข้อผิดพลาดทางไวยากรณ์ สิ่งนี้ เกิดขึ้นเมื่อค่าของตัวแปรเซลล์เหมือนกับหนึ่งใน ตัวดำเนินการ expr คุณสามารถหลีกเลี่ยงปัญหานี้ได้โดย การแยกส่วน นิพจน์เป็น:

```
expr "x$STR" = "x="
```

6. ในการส่งคืนค่าความยาวของตัวแปรสภาวะแวดล้อม \$SHELL /usr/bin/ksh ให้ป้อน:

```
expr length $SHELL
```

ค่าต่อไปนี้จะแสดง:

```
12
```

7. ในการส่งคืนตำแหน่งแรกของอักขระใดๆ ในสตริง "de" ที่พบใน "abcdef" ให้ป้อน:

```
expr index abcdef de
```

ค่าต่อไปนี้จะแสดง:

```
4
```

8. ในการส่งคืนตำแหน่งแรกของอักขระใดๆ ในสตริง "fd" ที่พบใน "abcdef" ให้ป้อน:

```
expr index abcdef fd
```

ค่าต่อไปนี้จะแสดง:

```
4
```

9. ในการส่งคืนสตริงเริ่มต้นที่ตำแหน่ง 11 โดยความยาวเป็น 6 ของสตริง "Goodnight Ladies" ให้ป้อน:

```
expr substr "Goodnight Ladies" 11 6
```

ค่าต่อไปนี้จะแสดง:

```
Ladies
```

ไฟล์

ไอเท็ม  
/usr/bin/expr

คำอธิบาย  
มีคำสั่ง expr

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง ed หรือ red” ในหน้า 297

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง bsh

คำสั่ง csh

ภาพรวม National Language Support

---

## คำสั่ง `exptun`

### วัตถุประสงค์

เอ็กซ์พอร์ตนิยามช่องสัญญาณและอาจเลือกกฎตัวกรองที่ผู้ใช้กำหนดทั้งหมด ที่สัมพันธ์กับช่องสัญญาณ สร้างเอ็กซ์พอร์ตไฟล์ช่องสัญญาณและเอ็กซ์พอร์ตไฟล์ กฎตัวกรองเพื่อเลือกที่สามารถใช้สำหรับพาร์ทเนอร์ช่องสัญญาณ

### ไวยากรณ์

```
exptun [-v 4|6] -f directory [-t tid_list] [-r] [-l manual]
```

### คำอธิบาย

ใช้คำสั่ง `exptun` เพื่อสร้างคอนเท็กซ์เอ็กซ์พอร์ตไฟล์ ช่องสัญญาณ และอาจเลือกไฟล์เสริมกฎตัวกรองสำหรับพาร์ทเนอร์ช่องสัญญาณ เพื่ออิมพอร์ต คำสั่งนี้ไม่เรียกทำงานช่องสัญญาณ จะเพียงแค่สร้าง ไฟล์ที่จำเป็นสำหรับพาร์ทเนอร์ช่องสัญญาณ

หมายเหตุ: ไฟล์เอ็กซ์พอร์ตที่สร้างมีคีย์ที่ใช้โดย ช่องสัญญาณ ป้องกันไฟล์เหล่านี้ด้วยคุณลักษณะการปกป้องระบบไฟล์ ของระบบปฏิบัติการ

### แฟล็ก

ไอเท็ม	คำอธิบาย
-f	กำหนดไดเรกทอรีที่จะเขียนเอ็กซ์พอร์ตไฟล์ไปไว้ ไดเรกทอรี จะถูกสร้างขึ้นหากยังไม่มี จากนั้นเอ็กซ์พอร์ตไฟล์อาจ ถูกส่งไปยังพาร์ทเนอร์ช่องสัญญาณที่ถูกอิมพอร์ต ขอแนะนำให้อีเอ็กซ์พอร์ตไฟล์ สำหรับแต่ละพาร์ทเนอร์ช่องสัญญาณมีข้อกำหนดคุณลักษณะไดเรกทอรีต่างกัน
-l	ชนิดของช่องสัญญาณที่คุณต้องการเอ็กซ์พอร์ต หากระบุ เป็นคู่มือ ช่องสัญญาณ ibm ด้วยตนเองเท่านั้นที่ถูกเอ็กซ์พอร์ต
-r	เอ็กซ์พอร์ตกฎตัวกรองที่ผู้ใช้กำหนดทั้งหมดที่สัมพันธ์ กับช่องสัญญาณ หากไม่ใช่แฟล็กนี้ นิยามช่องสัญญาณเท่านั้น ที่จะถูกเอ็กซ์พอร์ต
-t	ระบุรายการ IDs ช่องสัญญาณที่จะใช้สำหรับเอ็กซ์พอร์ตไฟล์ รายการ อาจระบุเป็นลำดับของ IDs ช่องสัญญาณที่คั่นด้วย “,” หรือ “-” (1, 3, 10, 50-55) หากไม่ใช่แฟล็กนี้ นิยามช่องสัญญาณทั้งหมดจากฐานข้อมูลช่องสัญญาณจะถูกเอ็กซ์พอร์ต
-v	IP เวอร์ชันของช่องสัญญาณที่จะถูกเอ็กซ์พอร์ต ค่า 4 ระบุช่องสัญญาณ IP version 4 ค่า 6 ระบุช่องสัญญาณ IP version 6 หากไม่ใช่แฟล็กนี้ นิยามช่องสัญญาณ IP version 4 และ IP version 6 ทั้งสองจะถูกเอ็กซ์พอร์ต

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง gentun” ในหน้า 688

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง chtun

คำสั่ง lstun

คำสั่ง mktun

---

## คำสั่ง extendlv

### Purpose

เพิ่มขนาดของโลจิคัลวอลุ่มโดยการเพิ่มพาร์ติชันที่ยังไม่ได้จัดสรรจากภายในกลุ่มวอลุ่ม

### ไวยากรณ์

ในการเพิ่มพาร์ติชันที่มีอยู่

```
extendlv [-a Position] [-e Range] [-u Upperbound] [-s Strict] LogicalVolumePartitions [PhysicalVolume ...]
```

ในการเพิ่มพาร์ติชันที่เจาะจง

```
extendlv [-m MapFile] LogicalVolumePartitions
```

### รายละเอียด

คำสั่ง **extendlv** เพิ่มจำนวนโลจิคัลพาร์ติชันที่จัดสรรให้แก่ *LogicalVolume* โดยการจัดสรรโลจิคัลพาร์ติชันเพิ่มจำนวนหนึ่งที่แสดงโดยพารามิเตอร์ *Partitions* พารามิเตอร์ *LogicalVolume* สามารถเป็นชื่อโลจิคัลวอลุ่ม หรือ ID โลจิคัล วอลุ่ม ในการจำกัดการจัดสรรให้แก่พาร์ติชันที่เจาะจงให้ใช้ชื่อของพาร์ติชันอย่างน้อยหนึ่งชื่อในพารามิเตอร์ *PhysicalVolume* มิฉะนั้น พาร์ติชันทั้งหมดในกลุ่มวอลุ่มจะพร้อมใช้สำหรับการจัดสรร พาร์ติชันใหม่

โดยค่าดีฟอลต์ โลจิคัลวอลุ่มถูกขยายโดยใช้คุณสมบัติที่มีอยู่ที่แสดงเมื่อคุณใช้คำสั่ง **lslv** ในการแทนที่คุณสมบัติที่มีอยู่เหล่านี้สำหรับพาร์ติชันใหม่เท่านั้น ให้เลือกเฉพาะพาร์ติชันใหม่ เลือกค่าที่แตกต่างสำหรับคุณสมบัติเหล่านี้โดยใช้แฟล็ก

จำนวนพาร์ติชันสูงสุดค่าดีฟอลต์สำหรับโลจิคัล วอลุ่มคือ 512 ก่อนที่จะขยายโลจิคัลวอลุ่มมากกว่า 512 โลจิคัลวอลุ่ม ให้ใช้คำสั่ง **chlv** เพื่อเพิ่มค่าดีฟอลต์

นโยบายการจัดสรรดีฟอลต์คือใช้จำนวน ฟิสิคัลวอลุ่มน้อยที่สุดต่อสำเนาโลจิคัลวอลุ่ม, เพื่อวางฟิสิคัลพาร์ติชัน ที่เป็นของ สำเนาให้ต่อเนื่องกันเท่าที่จะเป็นไปได้ แล้ว วางฟิสิคัลพาร์ติชัน ในขอบเขตที่ต้องการที่ระบุโดย แฟล็ก **-a** และโดยดีฟอลต์แต่ ละสำเนาของ โลจิคัลพาร์ติชันถูกกำหนดไว้ในฟิสิคัลวอลุ่มแยก

คุณสามารถระบุขนาดโลจิคัลวอลุ่มเป็น 512 บล็อก/KB/MB/GB เมื่อใช้ คำสั่ง **extendlv** (ดูที่ “ตัวอย่าง” ในหน้า 481)

**หมายเหตุ:**

1. เมื่อขยายสโตร์พดโลจิคัลวอลุ่ม จำนวนพาร์ติชันต้อง เป็นผลคูณจำนวนคู่ของความกว้างการสโตร์พด
2. ขอแนะนำให้โลจิคัลวอลุ่มใช้จำนวนพาร์ติชันขนาดใหญ่ (มากกว่า 800MB) เพื่อถูกขยายส่วนเพิ่มทีละน้อย
3. การเปลี่ยนแปลงกับโลจิคัลวอลุ่มไม่มีผลใน ระบบไฟล์ ในการเปลี่ยนคุณสมบัติระบบไฟล์ ให้ใช้คำสั่ง **chfs**
4. คุณต้องมีสิทธิผู้ใช้ **root** หรือเป็นสมาชิกของกลุ่ม **system** เพื่อใช้คำสั่งนี้
5. คำสั่ง **extendlv** ไม่อนุญาตให้ใช้บนกลุ่มสแน็ปช็อต วอลุ่ม

คุณสามารถใช้พาร์ตวัน (SMIT) **smit extendlv** ของ System Management Interface Tool (SMIT) เพื่อรันคำสั่งนี้

**แฟล็ก**

**หมายเหตุ:** แฟล็ก **-e** และ **-s** ใช้ไม่ได้กับโลจิคัลวอลุ่มที่ **stripe**

**ไอเท็ม**

**-a Position**

**คำอธิบาย**

ตั้งค่านโยบายการจัดสรรภายในฟิสิคัลวอลุ่ม (ตำแหน่ง ของโลจิคัลพาร์ติชันบนฟิสิคัลวอลุ่ม) ตัวแปร *Position* สามารถ เป็นค่าใดค่าหนึ่งต่อไปนี้:

- m** จัดสรรโลจิคัลพาร์ติชันในส่วนกลางด้านนอกของ แต่ละฟิสิคัลวอลุ่ม นี่เป็นตำแหน่งดีฟอลต์
- c** จัดสรรโลจิคัลพาร์ติชันในส่วนกลางของ แต่ละฟิสิคัลวอลุ่ม
- e** จัดสรรโลจิคัลพาร์ติชันในส่วนวงรอบด้านนอกของ แต่ละฟิสิคัลวอลุ่ม
- ie** จัดสรรโลจิคัลพาร์ติชันในส่วนขอบที่อยู่ด้านในของ แต่ละฟิสิคัลวอลุ่ม

**-e Range**

**im** จัดสรรโลจิคัลพาร์ติชันในส่วนกลางด้านในของ แต่ละฟิสิคัลวอลุ่ม  
ตั้งค่านโยบายการจัดสรรภายในฟิสิคัลวอลุ่ม (จำนวน ฟิสิคัลวอลุ่มเพื่อขยายออก โดยใช้อวาลุ่มที่จัดให้มี การจัดสรรที่ดีที่สุด) ค่าของ *Range* ถูกจำกัดโดยตัวแปร *Upperbound* (ที่ตั้งค่าด้วยแฟล็ก **-u**) และสามารถเป็นค่าใดค่าหนึ่งต่อไปนี้:

- x** จัดสรรโลจิคัลพาร์ติชันตามจำนวนฟิสิคัลวอลุ่มสูงสุด

**-m MapFile**

**m** จัดสรรโลจิคัลพาร์ติชันโดยตลอดของจำนวนฟิสิคัลวอลุ่ม ต่ำสุด  
ระบุฟิสิคัลพาร์ติชันที่จะจัดสรร พาร์ติชันถูก ใช้ตามลำดับที่กำหนดโดยไฟล์ที่กำหนดโดยพารามิเตอร์ *MapFile* ฟิสิคัลพาร์ติชันทั้งหมดเป็นของสำเนาชุดที่ถูกจัดสรร ก่อนการจัดสรรสำหรับสำเนาถัดไป รูปแบบ *MapFile* คือ:

**PVname: PPnum1[-PPnum2]**

โดยที่ *PVname* คือชื่อฟิสิคัลวอลุ่ม ( ตัวอย่างเช่น *hdisk0*) โดยเป็นหนึ่งในเรกคอร์ดต่อหนึ่ง ฟิสิคัลพาร์ติชันหรือ ช่วงของฟิสิคัลพาร์ติชันที่ต่อเนื่องกัน

**PVname** ชื่อของฟิสิคัลวอลุ่มตามที่ระบุโดยระบบ

**PPnum** หมายเลขฟิสิคัลพาร์ติชัน

**Important:** เมื่อคุณใช้แม่พไฟล์ คุณต้องเข้าใจและยึดตามพารามิเตอร์ LV-allocation ทั้งหมด เช่น *strictness*, *upperbound* และ *stripe width* การใช้แม่พไฟล์ ข้ามการตรวจสอบที่กระทำในรูทีน LVM-allocation นี้เป็นสิ่งสำคัญสำหรับ LVs แบบสโตร์พด ซึ่ง จะถูกมารูรูปแบบการจัดสรรแบบสโตร์พดปกติที่เป็นไปตาม ความกว้างสโตร์พด

ไอเท็ม  
-s *Strict*

#### คำอธิบาย

กำหนดนโยบายการจัดสรรที่เข้มงวด สำเนาของโลจิคัลพาร์ติชัน สามารถถูกจัดสรรเพื่อ แบ่งใช้หรือไม่แบ่งใช้ฟิลิคัลวอลุ่มเดียวกัน ตัวแปร *Strict* ถูกแทนโดยค่าใดค่าหนึ่งต่อไปนี้:

- y      เขียนนโยบายการจัดสรรเข้มงวด เพื่อที่สำเนาสำหรับโลจิคัลพาร์ติชัน ไม่สามารถแบ่งใช้ฟิลิคัลวอลุ่มเดียวกันได้
- n      ไม่เขียนนโยบายการจัดสรรเข้มงวด เพื่อที่สำเนาสำหรับโลจิคัลพาร์ติชัน สามารถแบ่งใช้ฟิลิคัลวอลุ่มเดียวกันได้
- s      ตั้งค่านโยบายการจัดสรรการจำกัดขั้นสูง เพื่อให้พาร์ติชันที่จัดสรร สำหรับมिरเรอร์หนึ่งไม่สามารถแบ่งใช้ฟิลิคัลวอลุ่มกับพาร์ติชันจากมिरเรอร์อื่น

หมายเหตุ: เมื่อเปลี่ยนโลจิคัลวอลุ่มที่ไม่ได้จำกัดขั้นสูงเป็นโลจิคัลวอลุ่ม ที่จำกัดขั้นสูง คุณต้องระบุฟิลิคัลวอลุ่มหรือใช้แฟล็ก -u

-u *Upperbound*

ขีดจำกัดสูงสุดของฟิลิคัลวอลุ่มสำหรับการจัดสรรใหม่ ค่า ของตัวแปร *Upperbound* ควรอยู่ระหว่าง หนึ่งถึงจำนวนฟิลิคัลวอลุ่มรวม เมื่อใช้ การจำกัดสูงสุด ขอบเขตบนแสดงจำนวนสูงสุดของ ฟิลิคัลวอลุ่มที่ใช้ได้สำหรับแต่ละสำเนามิเรอร์ เมื่อใช้สไทรพต์โลจิคัลวอลุ่ม ขอบเขตบนต้อง เป็นผลคูณของ *Stripe\_width*

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

1. ในการเพิ่มขนาดของโลจิคัลวอลุ่มที่แทน โดยไดเร็กทอรี lv05 โดยสามโลจิคัลพาร์ติชัน ให้พิมพ์:

```
extendlv lv05 3
```

2. ในการร้องขอโลจิคัลวอลุ่มชื่อ lv05 ที่มี ขนาดสูงสุด 10MB ให้พิมพ์:

```
extendlv lv05 10M #
```

คำสั่ง **extendlv** จะพิจารณาจำนวนพาร์ติชันที่จำเป็นในการสร้าง โลจิคัลวอลุ่มที่มีค่าน้อยเท่าขนาดนั้น

คุณสามารถใช้ตัวอักษรตัวพิมพ์ใหญ่และ ตัวพิมพ์เล็กดังนี้:

B/b	512 byte blocks
K/k	KB
M/m	MB
G/g	GB

## ไฟล์

ไอเท็ม  
/usr/sbin/

คำอธิบาย  
ไดเร็กทอรีที่มีคำสั่ง **extendlv** อยู่

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **chfs**

คำสั่ง **lslv**

หน่วยเก็บโลจิคัลวอลุ่ม

การควบคุมดูแล PowerHA SystemMirror

---

## คำสั่ง `extendvg`

### วัตถุประสงค์

เพิ่มฟิสิคัลวอลุ่มในกลุ่มวอลุ่ม

### ไวยากรณ์

```
extendvg [-f] [-p mirrorpool] volumegroup physicalvolume ...
```

### คำอธิบาย

คำสั่ง `extendvg` เพิ่มขนาดของ `volumegroup` โดยการเพิ่ม `physicalvolumes` อย่างน้อยหนึ่ง

ฟิสิคัลวอลุ่มถูกตรวจสอบเพื่อยืนยันว่า ยังไม่มีอยู่ในกลุ่มวอลุ่มอื่น หากระบบเชื่อว่า ฟิสิคัลวอลุ่มเป็นของกลุ่มวอลุ่มที่ถูกครอบครองแล้ว ระบบ จะออกจากการทำงาน แต่หากระบบตรวจพบพื้นที่รายละเอียดจากกลุ่ม วอลุ่มที่ยังไม่ถูกครอบครอง ระบบจะพร้อมให้ผู้ยืนยัน ที่จะทำงานคำสั่งต่อ เนื้อหาก่อนหน้าของฟิสิคัล วอลุ่มจะสูญหาย ดังนั้นผู้ต้องใช้ความระมัดระวังเมื่อใช้ฟังก์ชัน การแทนที่

**หมายเหตุ:** ในการใช้คำสั่งนี้ คุณ ต้องมีสิทธิ `root` หรือเป็นสมาชิกของกลุ่ม `system`

สำหรับกลุ่มวอลุ่มที่สร้างก่อน AIX 5.3 หรือสำหรับกลุ่มวอลุ่ม ที่สร้างบน AIX 5.3 แต่ แปรตามแฟล็ก `varyonvg -M` นั้น `extendvg` จะ ลมเหลวถ้าฟิสิคัลวอลุ่มมีขนาดถ่ายโอนสูงสุดที่เล็ก กว่าขนาดกลุ่มการติดตามโลจิคัลของกลุ่มวอลุ่ม สำหรับกลุ่มวอลุ่มที่สร้างบน AIX 5.3 และแปรตามโดยไม่มีแฟล็ก `varyonvg -M` นั้น `extendvg` จะลดขนาดกลุ่มการติดตามไดนามิกของกลุ่มวอลุ่ม โดยอัตโนมัติถ้าฟิสิคัลวอลุ่มมีขนาดถ่ายโอนสูงสุดที่เล็กกว่า ขนาดกลุ่มการติดตามโลจิคัลของกลุ่มวอลุ่ม

**หมายเหตุ:** คำสั่ง `extendvg` ไม่ อนุญาตให้ใช้บนกลุ่มสแน็ปช็อตวอลุ่ม

คุณสามารถใช้พารามิเตอร์ `smit extendvg` ของ System Management Interface Tool (SMIT) เพื่อรันคำสั่งนี้

**หมายเหตุ:** คำสั่งนี้จะล้มเหลวในการเพิ่มดิสก์ให้กับกลุ่มวอลุ่ม ถ้า ดิสก์ระบุว่า ถูกจัดการโดยตัวจัดการวอลุ่มของบริษัทอื่น เมื่อต้องการเขียนทับและเคลียร์ดิสก์ของตัวจัดการวอลุ่มของบริษัทอื่น ให้ใช้ `chpv -C HDiskName`

**หมายเหตุ:** เมื่อขยาย Volume Group (VG) พร้อมกัน อันดับแรกคุณต้องตรวจให้แน่ใจว่าดิสก์ใหม่แต่ละดิสก์ที่จะเพิ่มใน VG มี Physical Volume Identifier (PVID) กำหนดให้ และตรวจว่า PVID ที่เก็บ ใน Object Data Manager (ODM) เหมือนกับค่าที่อยู่บนทุกโหนด เมื่อ ใช้ยูทิลิตี้ Cluster Single Point of Control (C-SPOC) เพื่อขยาย VG การตรวจสอบนี้จะถูกดำเนินการโดยอัตโนมัติ

**หมายเหตุ:** VG จะตรวจสอบเพื่อดูว่าข้อจำกัดของประเภท PV ที่มีอยู่ในตำแหน่ง หากมีข้อจำกัดแล้ว รายการฟิสิคัลวอลุ่มบนบรรทัดรับคำสั่ง `extendvg` จะถูกตรวจสอบเพื่อให้แน่ใจว่า ฟิสิคัลวอลุ่มเหล่านี้ตรงตามข้อจำกัด หากพบดิสก์ หนึ่งตัวหรือมากกว่าหนึ่งที่ไม่ตรงกับข้อจำกัดของประเภท PV คำสั่งจะล้มเหลว

**หมายเหตุ:** คุณไม่สามารถผสมฟิสิคัลวอลุ่ม (PV) ที่มีบล็อกขนาด 4 KB กับ PV ที่มีบล็อกขนาดอื่น ขนาดบล็อกของ PV ทั้งหมดในกลุ่มวอลุ่ม ต้องเป็นขนาดเดียวกัน

## แฟล็ก

ไอเท็ม	คำอธิบาย
-f	กำหนดให้ฟิลิคัลวอลุ่มเพิ่มในกลุ่มวอลุ่ม ที่ระบุยกเว้นว่าเป็นสมาชิกของกลุ่มวอลุ่มอื่นใน Device Configuration Database หรือของกลุ่มวอลุ่มที่แอ็คทีฟ
-p <i>mirrorpool</i>	กำหนดแต่ละฟิลิคัลวอลุ่มที่ถูกเพิ่มให้กับมิเรอร์พูล ที่ระบุ มิเรอร์พูลถูกเปิดใช้งานในกลุ่มวอลุ่ม, กลุ่มวอลุ่ม ไม่สามารถถูกอิมพอร์ตเข้าสู่เวอร์ชันของ AIX ที่ไม่สนับสนุนมิเรอร์พูล ได้อีกต่อไป

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

ในการเพิ่มฟิลิคัลวอลุ่ม `hdisk3` และ `hdisk8` ไปยังกลุ่มวอลุ่ม `vg3` ให้ป้อน:

```
extendvg vg3 hdisk3 hdisk8
```

หมายเหตุ: กลุ่มวอลุ่มต้องถูกทำให้ขึ้นต่อกันก่อนการขยาย

## ข้อจำกัด

คำสั่ง `extendvg` ไม่สามารถรันบนกลุ่มสแน็ปช็อตวอลุ่ม

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/sbin/extendvg</code>	มีคำสั่ง <code>extendvg</code>

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `reducevg`

หน่วยเก็บโลจิคัลวอลุ่ม

คำสั่ง `chpv`

การควบคุมดูแล PowerHA SystemMirror



---

## f

คำสั่ง AIX ต่อไปนี้เริ่มต้นด้วยตัวอักษร f

---

### คำสั่ง f

#### วัตถุประสงค์

แสดงข้อมูลผู้ใช้ คำสั่งนี้เหมือนกับ คำสั่ง finger

#### ไวยากรณ์

```
{ f | finger } [ [-b] [-h] [-l] [-p] ] | [-i] [-q] [-s] [-w] ]
```

```
[-f] [-m] [ User | User@Host | @Host ]
```

#### คำอธิบาย

คำสั่ง /usr/bin/f แสดง ข้อมูลเกี่ยวกับผู้ใช้ที่ล็อกอินในโฮสต์ขณะนี้ รูปแบบของ เอาต์พุตแตกต่างกันตามอ็อปชันของข้อมูลที่แสดง

#### รูปแบบดีฟอลต์

รูปแบบดีฟอลต์ประกอบด้วยรายการต่อไปนี้:

- ชื่อล็อกอิน
- ชื่อนามสกุลผู้ใช้
- ชื่อเทอร์มินัล
- สถานะการเขียน (\* (เครื่องหมายดอกจัน) หน้าชื่อ เทอร์มินัลระบุว่าสิทธิในการเขียนถูกปฏิเสธ)

สำหรับผู้ใช้แต่ละคนบนโฮสต์ รายการข้อมูลดีฟอลต์ ยังประกอบด้วยรายการต่อไปนี้ ถ้าทราบ:

- เวลาเดินเครื่องเปล่า (เวลาเดินเครื่องเปล่าเป็นนาทีหากเป็นเลขจำนวนเต็ม เดียว เป็นชั่วโมงและนาที หาก : (โคลอน) แสดงอยู่ หรือวันและชั่วโมงหาก "d" แสดง)
- เวลาล็อกอิน
- ข้อมูลเฉพาะไซต์

ข้อมูลเฉพาะไซต์ถูกเรียกข้อมูลจาก ไฟล์ gecos ในไฟล์ /etc/passwd ไฟล์ gecos อาจมีชื่อนามสกุลผู้ใช้ตามด้วยเครื่องหมายจุลภาค ข้อมูลทั้งหมด ที่ตามหลังเครื่องหมายจุลภาคจะถูกแสดงโดยคำสั่ง finger ที่มีข้อมูล เฉพาะไซต์

#### รูปแบบยาว

รูปแบบยาวใช้โดยคำสั่ง f เมื่อใดที่กำหนดรายการชื่อของผู้ใช้ (ชื่อแอดเดสส์รวมถึง ชื่อและนามสกุลของผู้ใช้ก็ยอมรับได้) รูปแบบนี้เป็นแบบหลายบรรทัด และประกอบด้วยข้อมูลทั้งหมดที่อธิบายข้างต้นรวมถึงต่อไปนี้:

- ไตเร็กทอรี \$HOME ของผู้ใช้
- ล็อกอินเชลล์ของผู้ใช้
- เนื้อหาของไฟล์ .plan ในไตเร็กทอรี \$HOME ของผู้ใช้
- เนื้อหาของไฟล์ .project ในไตเร็กทอรี \$HOME ของผู้ใช้

คำสั่ง f ยังอาจ ถูกใช้เพื่อค้นหาผู้ใช้บนระบบรีโมต รูปแบบคือเพื่อระบุผู้ใช้ เป็น User@Host หากคุณไม่ระบุชื่อผู้ใช้ คำสั่ง f จะ จัดให้มีการแสดงรายการรูปแบบมาตรฐานบนระบบรีโมต

สร้างไฟล์ .plan และ .project โดยใช้เท็กซ์เอดิเตอร์ที่ต้องการของคุณและวางไฟล์ในไตเร็กทอรี \$HOME ของคุณ คำสั่ง f ใช้รู ทินย่อของ toascii เพื่อแปลงอักขระภายนอก ขอบเขตอักขระ ASCII ปกติ เมื่อแสดงเนื้อหาของไฟล์ .plan และ .project คำสั่ง f และ M- หน้าอักขระ แต่ละตัวที่ถูกแปลง

เมื่อคุณระบุผู้ใช้ด้วยพารามิเตอร์ User คุณสามารถระบุด้วยชื่อ นามสกุล หรือแอดเคาต์ของผู้ใช้ เมื่อคุณระบุผู้ใช้ คำสั่ง f ที่ โสสต์ที่ระบุ จะส่งกลับข้อมูลเกี่ยวกับ ผู้ใช้เหล่านี้ในรูปแบบยาวเท่านั้น

สำหรับข้อมูลเพิ่มเติมๆ เกี่ยวกับคำสั่ง f โปรดดูที่ "การติดตั้ง TCP/IP" ใน Networks and communication management

## แฟล็ก

ไอเท็ม คำอธิบาย

- b แสดงรายการรูปแบบยาวอย่างย่อ
- f ระวังการพิมพ์บรรทัดส่วนหัวบนเอาต์พุต (บรรทัดแรกที่กำหนดฟิลด์ที่จะถูกแสดง)
- h ระวังการพิมพ์ไฟล์ .project ด้วยรูปแบบยาว และรูปแบบยาวอย่างย่อ
- i แสดงรายการที่มีเวลาเดินเครื่องเปล่าอย่างรวดเร็ว
- l แสดงรายการรูปแบบยาว
- m ถือว่าพารามิเตอร์ User ระบุ ID ผู้ใช้ (ใช้สำหรับค่าควบคุมการเข้าถึงอย่างรอบคอบ) ไม่ใช่ชื่อล็อกอินผู้ใช้
- p ระวังการพิมพ์ไฟล์ .plan ในรูปแบบยาว และรูปแบบยาวอย่างย่อ
- q แสดงรายการแบบรวดเร็ว
- s แสดงรายการรูปแบบสั้น
- w แสดงรายการรูปแบบสั้นแบบจำกัด

## พารามิเตอร์

ไอเท็ม

คำอธิบาย

@Host

ระบุผู้ใช้ที่ล็อกอินทั้งหมดบนรีโมตโฮสต์

User

ระบุ ID ผู้ใช้โลคัล (ใช้สำหรับค่าควบคุมการเข้าถึงอย่างรอบคอบ) หรือชื่อล็อกอินผู้ใช้โลคัล ดังที่ระบุในไฟล์ /etc/passwd

User@Host

ระบุ ID ผู้ใช้บนรีโมตโฮสต์ โดยแสดงในรูปแบบยาว

## ตัวอย่าง

1. ในการรับข้อมูลเกี่ยวกับผู้ใช้ทั้งหมดที่ล็อกอิน บนโฮสต์ alcatraz ให้ป้อน:

```
f @alcatraz
```

ข้อมูล จะคล้ายกับที่แสดงต่อไปนี้:

```
[alcatraz.austin.ibm.com]
Login   Name      TTY Idle      When      Site Info
brown   Bob Brown console  2d   Mar 15 13:19
smith   Susan Smith pts0   11:   Mar 15 13:01
jones   Joe Jones  tty0   3     Mar 15 13:01
```

ผู้ใช้ brown ล็อกอินที่ console ผู้ใช้ smith ล็อกอินจาก pseudo teletype line pts0 และผู้ใช้ jones ล็อกอินจาก tty0

2. ในการรับข้อมูลเกี่ยวกับผู้ใช้ brown ที่ alcatraz ให้ป้อน:

```
f brown@alcatraz
```

ข้อมูล จะคล้ายกับที่แสดงต่อไปนี้:

```
Login name: brown
Directory: /home/brown   Shell: /home/bin/xinit -L -n Startup
On since May 8 07:13:49 on console
No Plan.
```

3. ในการรับข้อมูลเกี่ยวกับผู้ใช้ brown ที่โลคัลโฮสต์ในรูปแบบสั้น ให้ป้อน:

```
f -q brown
```

ข้อมูล จะคล้ายกับที่แสดงต่อไปนี้:

Login	TTY	When
brown	pts/6	Mon Dec 17 10:58

## ไฟล์

ไอเท็ม

/usr/bin/f

/etc/utmp

/etc/passwd

/etc/security/passwd

/var/adm/lastlog

\$HOME/.plan

\$HOME/.project

คำอธิบาย

มีคำสั่ง f

มีรายการของผู้ใช้ที่ล็อกอินขณะนี้

กำหนดแอดเดสส์ ชื่อ และโฮมไดเรกทอรี

กำหนดรหัสผ่านผู้ใช้

มีเวลาล็อกอินล่าสุด

ไฟล์ทางเลือกที่มีรายละเอียดแบบหนึ่งบรรทัดของแผนงานของผู้ใช้

ไฟล์ทางเลือกที่มีการมอบหมายโปรเจกต์ของผู้ใช้

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง hostname” ในหน้า 802

“คำสั่ง finger” ในหน้า 577

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง rwho

คำสั่งสำหรับการแสดงข้อมูลเกี่ยวกับผู้ใช้ที่ล็อกอิน

---

## คำสั่ง factor

### วัตถุประสงค์

แยกตัวประกอบตัวเลข

### ไวยากรณ์

**factor** [ *Number* ]

## คำอธิบาย

เมื่อเรียกใช้โดยไม่มีการระบุค่าสำหรับพารามิเตอร์ *Number* คำสั่ง **factor** จะรอให้คุณป้อนค่าจำนวนบวกที่น้อยกว่า 1E14 (100,000,000,000,000) จากนั้นเขียนตัวประกอบจำนวนเฉพาะของ เลขจำนวนนั้นไปยังเอาต์พุตมาตรฐาน คำสั่งแสดงตัวประกอบแต่ละค่าตามจำนวนครั้งที่เหมาะสม ในการออกให้ป้อน 0 หรืออักขระใดๆ ที่ไม่ใช่ตัวเลข

เมื่อเรียกใช้พร้อมอาร์กิวเมนต์ คำสั่ง **factor** จะพิจารณาตัวประกอบจำนวนเฉพาะของพารามิเตอร์ *Number* เขียนผลลัพธ์ไปยังเอาต์พุตมาตรฐานและออกจากการทำงาน

## ตัวอย่าง

ในการคำนวณตัวประกอบจำนวนเฉพาะของ 123 ให้ป้อน:

```
factor 123
```

ข้อมูลนี้ถูกแสดง:

```
123
  3
 41
```

## ไฟล์

ไอเท็ม  
/usr/bin/factor

คำอธิบาย  
มีคำสั่ง **factor**

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **bc**

---

## คำสั่ง true หรือ false

### วัตถุประสงค์

ส่งคืนค่าออกของคุณ (true) หรือค่าออกที่ไม่ใช่ศูนย์ (false)

### ไวยากรณ์

**true**

**false**

## คำอธิบาย

คำสั่ง **true** ส่งคืนค่าการตอบกลับ คำสั่ง **false** ส่งคืนค่าออกที่ไม่ใช่ศูนย์ ซึ่งโดยส่วนใหญ่แล้ว คำสั่งเหล่านี้ถูกใช้เป็นส่วนหนึ่งของ สคริปต์เชลล์

## ตัวอย่าง

หากต้องการสร้างรูปที่แสดงวันที่และเวลา หนึ่งครั้งในแต่ละนาที ให้ใช้โค้ดต่อไปนี้ในสคริปต์เชลล์:

**488** AIX เวอร์ชัน 7.2: ข้อมูลอ้างอิงคำสั่ง วอลุ่ม 2, d - h

```
while true
do
    date
    sleep 60
done
```

---

## คำสั่ง **reboot** หรือ **fastboot**

### วัตถุประสงค์

รีสตาร์ทระบบ

### ไวยากรณ์

```
{ reboot | fastboot } [-l] [-n] [-q] [-t mmddHHMM[yy]]
```

### คำอธิบาย

คำสั่ง **reboot** สามารถใช้เพื่อดำเนินการรีบูตหากไม่มีผู้ใช้อื่นล็อกอินเข้าสู่ระบบ คำสั่ง **lsattr** และป้อน **lsattr -D -l sys0** ค่าดีฟอลต์คือ **true** เมื่อต้องการรีเซ็ตค่าแอตทริบิวต์รีสตาร์ทอัตโนมัติ เป็น **false** ใช้ **/var/adm/wtmp** ไฟล์แอคเคาต์ล็อกอิน การดำเนินการเหล่านี้ถูกยับยั้ง หากแฟล็ก **-l**, **-n** หรือ **-q** แสดงอยู่

คำสั่ง **fastboot** รีสตาร์ทระบบ โดยเรียกคำสั่ง **reboot** คำสั่ง **fsck** จะรันในระหว่างที่ระบบเริ่มทำงานเพื่อตรวจสอบระบบไฟล์ คำสั่งนี้จัดเตรียม ความเข้ากันได้ของ BSD

### แฟล็ก

ไอเท็ม คำอธิบาย

- l ไม่บันทึกการรีบูตหรือวางเรีกคอร์ดการปิดระบบในไฟล์ แอคเคาต์ แฟล็ก **-l** ไม่ได้หยุดทำงานการอัปเดตไฟล์ แอคเคาต์ แฟล็ก **-n** และ **-q** หมายถึง **-l**
- n ห้ามดำเนินการกับคำสั่ง **sync** การใช้แฟล็กนี้ สามารถทำให้ระบบไฟล์เสียหายได้
- q รีสตาร์ทโดยไม่ปิดระบบที่กำลังรันกระบวนการต่างๆ  
หมายเหตุ: การชิงโครโนระบบไฟล์จะไม่เกิดขึ้นหากใช้แฟล็ก **-q** หากคุณต้องการให้ระบบไฟล์ถูกชิงโครโนซ์ให้รันคำสั่ง **sync** แบบแมนนวลหรือใช้คำสั่ง **shutdown -r**
- t ปิดระบบโดยทันที จากนั้นรีสตาร์ทระบบ ตามวันที่ระบุ วันที่ที่ถูกต้องมีรูปแบบต่อไปนี้:

*mmddHHMM*[*yy*]

โดยที่:

*mm* ระบุเดือน

*dd* ระบุวัน

*HH* ระบุชั่วโมง

*MM* ระบุนาที

*yy* ระบุปี (เมื่อเลือก) ค่าตัวเลขสองตัวแสดงถึง ค่าของปีในศตวรรษปัจจุบัน (ขึ้นอยู่กับเวลาของระบบ) ตัวอย่างเช่น ถ้าปีปัจจุบันตามเวลาของระบบ คือ 1985, 99 หมายถึง 1999 และถ้าปีปัจจุบันคือ 2005 ผลคือ 99 หมายถึง 2099 และ 04 หมายถึง 2004

## ความปลอดภัย

ขอควรทราบสำหรับผู้ใช้ RBAC และผู้ใช้ Trusted AIX : คำสั่งนี้สามารถดำเนินการกับการดำเนินการที่มีสิทธิ์ใช้งาน เฉพาะผู้ใช้ที่ได้รับสิทธิ์ใช้งานเท่านั้น สามารถรันการดำเนินงานพิเศษได้ สำหรับข้อมูลเพิ่มเติม เกี่ยวกับการอนุญาต และสิทธิ์ ดูที่ฐานข้อมูลคำสั่งที่ใช้สิทธิ์ใน การรักษาความปลอดภัย สำหรับรายการ ของสิทธิ์ใช้งานและการอนุญาตที่เชื่อมโยงกับคำสั่งนี้ โปรดดูที่ คำสั่ง Issecattr หรือคำสั่งย่อย getcmdattr

## ตัวอย่าง

หากต้องการปิดระบบโดยไม่มีกระบวนการรีบูต ให้ป้อน:

```
reboot -l
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/etc/rc	ระบุสคริปต์การเริ่มต้นทำงานกับระบบ
/var/adm/wtmp	ระบุไฟล์แอคเคาต์

---

## คำสั่ง fc

### วัตถุประสงค์

ประมวลผลรายการประวัติคำสั่ง

### Syntax

ในการเปิดเอดิเตอร์เพื่อแก้ไขและเรียกใช้งานคำสั่งที่ป้อนก่อนหน้าใหม่

```
fc [-r] [-e Editor] [First [Last]]
```

ในการสร้างรายการของคำสั่งที่ป้อนก่อนหน้า

```
fc -l [-n] [-r] [First [Last]]
```

ในการสร้างรายการของคำสั่งที่ป้อนก่อนหน้าที่มีเวลาของการดำเนินการ

```
fc -t [-n] [-r] [First [Last]]
```

ในการเรียกใช้งานคำสั่งที่ป้อนก่อนหน้าใหม่

```
fc -s [Old=New] [First]
```

### คำอธิบาย

คำสั่ง fc แสดงเนื้อหา ของไฟล์ประวัติคำสั่งของคุณ หรือเรียกใช้เอดิเตอร์เพื่อแก้ไขและเรียกใช้งาน คำสั่งที่ป้อนก่อนหน้าใหม่ในเซลล์

ไฟล์ประวัติคำสั่งแสดงรายการคำสั่งตามหมายเลข หมายเลขแรกในรายการถูกเลือกโดยไม่มีกฎเกณฑ์ ความสัมพันธ์ของ หมายเลขกับคำสั่งจะไม่เปลี่ยนแปลงยกเว้นเมื่อผู้ใช้ล็อกอินและ ไม่มีกระบวนการอื่นกำลังเข้าถึงรายการ ในกรณีเช่นนั้น ระบบ จะรีเซ็ตการกำหนดหมายเลข เพื่อเริ่มต้นให้คำสั่งที่เก็บไว้นานที่สุดมีค่า 1

หากหมายเลขในไฟล์ประวัติคำสั่งมีค่าถึงขีดจำกัดที่มากกว่าค่าของตัวแปรสถานะแวดล้อม HISTSIZE หรือ 32767 แล้ว แม้ว่าค่าใดมากกว่า ค่าจะวนกลับไปเป็นค่า 1 ถึงอย่างไรก็ตามการวนกลับหมายเลขนี้เป็นทางเลือก คำสั่ง fc จะดูแลรักษาลำดับ การจัดเรียงตามเวลาของคำสั่งไว้ ตัวอย่างเช่น หากมีสามคำสั่งในลำดับที่กำหนดให้เป็นหมายเลข 32766, 32767 และ 1 (ที่ วนกลับ) คำสั่ง 32767 จะยังคงถูกพิจารณาว่ามาก่อนคำสั่ง 1

คำสั่งในไฟล์ประวัติสามารถแสดงโดยใช้แฟล็ก -l (L ตัวพิมพ์เล็ก) เมื่อไม่ระบุแฟล็ก -l และคำสั่งถูกแก้ไขโดยใช้แฟล็ก -e Editor บรรทัดผลลัพธ์จะถูกใส่ที่ท้ายของไฟล์ประวัติ จากนั้นถูกเรียกใช้งานโดยเซลล์อีกครั้ง (คำสั่ง fc -e Editor ไม่ถูกป้อนลงใน รายการประวัติคำสั่ง) หากเอดิเตอร์ส่งคืนค่าสถานะการออกไม่เป็นศูนย์ คำนี้จะไม่แสดงรายการในไฟล์ประวัติและไม่ทำการเรียกใช้คำสั่งอีกครั้ง

การกำหนดค่าตัวแปรบรรทัดคำสั่งใดๆ หรือตัวดำเนินการ การเปลี่ยนทิศทางที่เข้ากับคำสั่ง fc จะเรียกใช้คำสั่งก่อนหน้านี้ อีก ครั้ง โดยไม่แสดงข้อผิดพลาดมาตรฐานสำหรับคำสั่ง fc และคำสั่งก่อนหน้านี้ทั้งสอง ตัวอย่างเช่น :

```
fc -s -- -1 2>/dev/null
```

## แฟล็ก

ไอเท็ม	คำอธิบาย
-e Editor	แก้ไขคำสั่งโดยใช้เอดิเตอร์ที่ระบุ พารามิเตอร์ Editor ควรเป็นชื่อคำสั่ง คำสั่งถูกกำหนดโดยใช้ตัวแปรสถานะแวดล้อม PATH ค่าในตัวแปรสถานะแวดล้อม FCEDIT ถูกใช้เป็นค่าดีฟอลต์เมื่อไม่ระบุแฟล็ก -e หากตัวแปรสถานะแวดล้อม FCEDIT เป็น null หรือไม่ถูกตั้งค่า จะใช้เอดิเตอร์ cd
-l	(L ตัวพิมพ์เล็ก) แสดงรายการคำสั่งในไฟล์ประวัติของคุณ ไม่มีเอดิเตอร์ ถูกเรียกใช้เพื่อแก้ไขไฟล์ คำสั่งถูกเขียนตามลำดับที่ระบุ โดยพารามิเตอร์ First และ Last ถูกเปิดผลจากแฟล็ก -r ที่แต่ละครั้งถูกนำหน้า ด้วยหมายเลขคำสั่ง
-n	ไม่แสดงหมายเลขคำสั่งเมื่อใช้กับแฟล็ก -l
-r	ย้อนลำดับของคำสั่งที่แสดงรายการ (เมื่อใช้กับแฟล็ก -l) หรือย้อนลำดับของคำสั่งที่ถูกแก้ไข (เมื่อ ไม่ระบุแฟล็ก -l)
-s	เรียกใช้งานคำสั่งใหม่โดยไม่เรียกใช้เอดิเตอร์ หากไม่ระบุพารามิเตอร์ First เช่นกัน แฟล็ก -s จะเรียกใช้งานคำสั่งก่อนหน้านี้ใหม่
-t	แสดงรายการคำสั่งในไฟล์ประวัติของคุณที่มี เวลาในการเรียกใช้งาน การทำงานเหมือนกับแฟล็ก -l แต่เวลาในการเรียกใช้งานของ คำสั่งจะถูกแสดง หมายเหตุ: หาก ฟิลต์เวลาถูกบันทึกไว้ก่อนหน้านี้โดยการตั้งค่า EXTENDED_HISTORY=ON ดังนั้นฟิลต์เวลาที่จัดรูปแบบจะแสดง มิฉะนั้นจะแสดง "?"

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
First or Last	เลือกคำสั่งเพื่อแสดงรายการหรือแก้ไข จำนวนของคำสั่งก่อนหน้านี้ ที่สามารถเข้าถึงได้จะถูกกำหนดโดยค่าของตัวแปรสถานะแวดล้อม HISTSIZE พารามิเตอร์ First และ Last ต้องมีค่าใดค่าหนึ่งต่อไปนี้: [+]Number แสดงหมายเลขคำสั่งที่เจาะจง หมายเลขคำสั่งสามารถแสดง ด้วยแฟล็ก -l+ (เครื่องหมายบวก) เป็นค่า ดีฟอลต์ -Number แสดงคำสั่งที่ถูกเรียกใช้งานก่อนหน้านี้ ที่ระบุโดยจำนวนของคำสั่งเพื่อสำรองข้อมูลในไฟล์ประวัติ ตัวอย่าง เช่น -1 ระบุ คำสั่งก่อนหน้านี้ สตริง ระบุคำสั่งที่ป้อนล่าสุดที่ขึ้นต้นด้วยสตริง ที่ระบุ หากระบุพารามิเตอร์ Old=New โดยไม่มีแฟล็ก -s สตริงจาก พารามิเตอร์ First จะไม่สามารถมี = (เครื่องหมายเท่ากับ) ผังอยู่ เมื่อใช้แฟล็ก -s การไม่ระบุพารามิเตอร์ First จะทำให้ใช้คำสั่ง ก่อนหน้านี้

เมื่อไม่ระบุแฟล็ก -s กฎต่อไปนี้จะใช้:

- เมื่อใช้แฟล็ก -l การไม่ระบุพารามิเตอร์ *Last* จะทำให้คำสั่ง ก่อนหน้านี้เป็นค่าดีฟอลต์
- เมื่อใช้แฟล็ก -r, -n และ -e การไม่ระบุพารามิเตอร์ *Last* ทำให้ค่าดีฟอลต์เป็นพารามิเตอร์ *First*
- หากทั้งพารามิเตอร์ *First* และ *Last* ไม่ถูกระบุ 16 คำสั่งก่อนหน้านี้ จะถูกแสดงรายการ หรือคำสั่งเดียวก่อนหน้านี้ถูกแก้ไข (ขึ้นกับว่า ใช้แฟล็ก -l หรือไม่)
- หากพารามิเตอร์ทั้ง *First* และ *Last* มีแสดงอยู่ คำสั่งทั้งหมดจะถูกแสดงรายการ (เมื่อ ระบุแฟล็ก -l) หรือแก้ไข (เมื่อไม่ ระบุแฟล็ก -l) การแก้ไขหลายๆ แก้ไขจะถูกทำสำเร็จ โดยการส่งคำสั่งทั้งหมดไปยังเอดิเตอร์ในครั้งเดียว โดยแต่ละคำสั่ง จะขึ้นต้น บรรทัดใหม่ หากพารามิเตอร์ *First* แสดง คำสั่งที่ใหม่กว่าในพารามิเตอร์ *Last* คำสั่ง ถูกแสดงรายการและแก้ไข ตามลำดับย้อนกลับ นี้เท่ากับการใช้แฟล็ก -r ตัวอย่างเช่น คำสั่งต่อไปนี้บนบรรทัดแรก จะเท่ากับคำสั่งที่สอดคล้องกันบน บรรทัดที่สอง:

```
fc -r 10 20          fc      30 40
fc      20 10        fc -r 40 30
```

- เมื่อใช้ช่วงคำสั่ง จะไม่ถือ เป็นข้อผิดพลาดหากระบุค่า *First* หรือ *Last* ที่ไม่มีอยู่ในรายการประวัติ คำสั่ง fc แทนค่าที่แสดง คำสั่งที่เก่าที่สุดหรือที่ใหม่ที่สุดในรายการ ตามความเหมาะสม ตัวอย่างเช่น หากมีสิบคำสั่งเท่านั้นในรายการประวัติ เป็นหมายเลข 1 ถึง 10 คำสั่ง:

```
fc -l
fc 1 99
```

แสดงรายการและแก้ไขทั้งสิบคำสั่งตามลำดับ

ไอเท็ม            คำอธิบาย  
Old=New           ในคำสั่งที่จะถูกเรียกใช้งานใหม่ ให้แทนที่ค่าแรก ของสตริงเก่าด้วยสตริงใหม่

## ตัวแปรสถานะแวดล้อม

ตัวแปรสถานะแวดล้อมต่อไปนี้มีผลต่อการเรียกใช้งาน ของคำสั่ง fc:

ไอเท็ม	คำอธิบาย
EXTENDED_HISTORY	ใช้เพื่อควบคุมการบันทึกเวลาของการเรียกใช้งานคำสั่ง ในไฟล์ประวัติ หากตัวแปรถูกตั้งค่าเป็น ON ดังนั้นเวลา จะถูก บันทึก มิฉะนั้นจะไม่ถูกบันทึก
FCEDIT	เมื่อขยายโดยเชลล์ จะพิจารณาค่าดีฟอลต์สำหรับตัวแปร -e editor หากตัวแปรสถานะแวดล้อม FCEDIT เป็น null หรือไม่ถูกตั้งค่า เอดิเตอร์ ed จะเป็นค่าดีฟอลต์
HISTDATEFMT	นี้ใช้เพื่อควบคุมรูปแบบของเวลาที่แสดง โดยคำสั่ง fc -t ตัวอย่างเช่น หาก HISTDATEFMT=%Y ดังนั้น fc -t จะแสดงปี เมื่อเรียกใช้งาน คำสั่ง การจัดรูปแบบคล้ายกับที่ทำโดยคำสั่ง date
HISTFILE	พิจารณาชื่อพารของไฟล์ประวัติคำสั่ง หากตัวแปรสถานะแวดล้อม HISTFILE ไม่ถูกตั้งค่า เชลล์อาจ พยายามเข้าถึง หรือสร้างไฟล์ .sh_history ใน โสมโตเร็กทอรีของผู้ใช้
HISTSIZE	พิจารณาเลขทศนิยมในการแสดงขีดจำกัดของจำนวน คำสั่งก่อนหน้านี้ที่สามารถเข้าถึงได้ หากตัวแปรนี้ไม่ถูกตั้ง ค่า จะใช้ ค่าดีฟอลต์คือ 128

## สถานะออก

ค่าออกต่อไปนี้ถูกส่งคืน:

ไอเท็ม คำอธิบาย  
0 การรายการสำเร็จเรียบร้อย  
>0 มีข้อผิดพลาดเกิดขึ้น

มีฉะนั้น ค่าสถานะการออกคือค่าของคำสั่ง ที่ถูกเรียกใช้งานโดยคำสั่ง fc

## ตัวอย่าง

1. ในการเรียกใช้เডিเตอร์ที่กำหนดโดยตัวแปรสถานะแวดล้อม FCEDIT ของคำสั่งล่าสุด (ดีฟอลต์เডিเตอร์คือ /usr/bin/ed) ให้ป้อน:

```
fc
```

คำสั่ง ถูกเรียกใช้เมื่อคุณแก้ไขเสร็จสิ้น

2. ในการแสดงรายการสองคำสั่งก่อนหน้านี้ที่ถูกเรียกใช้งาน ให้ป้อน:

```
fc -l -2
```

3. ในการค้นหาคำสั่งที่ขึ้นต้นด้วย cc ให้เป็น foo เป็น bar และแสดงและเรียกใช้งานคำสั่ง ให้ป้อน:

```
fc -s foo=bar cc
```

4. ในการแสดงคำสั่งที่เรียกใช้งานก่อนหน้านี้พร้อมกับเวลาในการเรียกใช้งาน ให้พิมพ์:

```
fc -t
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/ksh	มีคำสั่งในตัว fc ของคอร์นเชลล์
/usr/bin/fc	มีคำสั่ง fc

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง ksh

---

## คำสั่ง fccheck

### วัตถุประสงค์

ดำเนินการพิจารณาปัญหาเบื้องต้นบนยูทิลิตี้ First Failure Data Capture (FFDC)

### ไวยากรณ์

```
/usr/sbin/rsct/bin/fccheck [-q] | [-h]
```

### คำอธิบาย

fccheck ดำเนินการพิจารณาปัญหาเบื้องต้น สำหรับยูทิลิตี้ First Failure Data Capture คำสั่งจะตรวจหาเงื่อนไข และข้อมูลต่อไปนี้บนโลคัลโหนด:

- ตรวจสอบว่าการใช้งาน FFDC Error Stack ถูกปิดใช้งานในสถานะแวดล้อมของกระบวนการ ปัจจุบันหรือไม่
- จัดการ IP แอดเดรสที่ขณะนี้ FFDC ใช้เพื่อระบุ โลคัลโหนด

- ตรวจสอบว่า `/var/adm/ffdc/stacks` มีอยู่ และ หากมีอยู่ มีพื้นที่ว่างจำนวนเท่าใดในระบบไฟล์ที่มีไดเรกทอรีอยู่ ตรวจสอบเพื่อดูว่าจะมีพื้นที่ไม่เพียงพอในการสร้าง FFDC Error Stacks หรือไม่
- ตรวจสอบว่า `/var/adm/ffdc/dumps` มีอยู่ และ หากมีอยู่ มีพื้นที่ว่างจำนวนเท่าใดในระบบไฟล์ที่มีไดเรกทอรีอยู่

ผลลัพธ์ของการทดสอบเหล่านี้จะถูกแสดงไปยังเอาต์พุตมาตรฐาน ยกเว้นว่าได้ระบุ อีอ็อปชัน "ไม่โต้ตอบ" ไว้ `fccheck` ตั้งค่าสถานะ ที่มีเพื่อระบุเงื่อนไขสำคัญสูงสุดที่จะตรวจหาระหว่างการเรียกใช้งาน การทดสอบ

## แฟล็ก

- h แสดงข้อมูลวิธีใช้และการใช้งานไปยังเอาต์พุตมาตรฐาน ไม่มีการประมวลผลอื่น
- q ระบุโหมด "ไม่โต้ตอบ" คำสั่งจะไม่แสดงผลของ แต่ละการทดสอบเอาต์พุตมาตรฐาน สถานะการออกของคำสั่ง ต้องใช้ เพื่อพิจารณาผลลัพธ์ของการตรวจสอบ หากตรวจพบเงื่อนไขมากกว่าหนึ่งเงื่อนไข สถานะการออกจะแสดงเงื่อนไขที่สำคัญสูงสุดที่ตรวจพบโดย `fccheck`

## สถานะออก

โค้ดสถานะการออกเลขจำนวนเต็มต่อไปนี้อาจสร้างขึ้นโดยคำสั่งนี้:

- 0 เงื่อนไขทั้งหมดที่ทดสอบโดย `fccheck` ตรวจพบว่า เป็นพารามิเตอร์การดำเนินการปกติ
- 2 ข้อมูลวิธีใช้ถูกแสดงเรียบร้อยแล้ว ไม่มีการประมวลผลใดๆ อีก
- 12 ไม่มีการดำเนินการตรวจสอบ มีการระบุอีอ็อปชันไม่ถูกต้องให้แก่คำสั่งนี้
- 19 ไดเรกทอรี `/var/adm/ffdc/stacks` ไม่ถูกเมาท์ หรือไม่มีอยู่
- 20 ไม่สามารถเข้าถึง หรือตรวจสอบไดเรกทอรีอย่างน้อยหนึ่งไดเรกทอรีในพาท `/var/adm/ffdc/stacks` เปลี่ยนอาจถูกเปลี่ยนบนไดเรกทอรีอย่างน้อย หนึ่งไดเรกทอรีในพาทนี้เพื่อป้องกันการเข้าถึง
- 24 ไม่สามารถเข้าถึง หรือตรวจสอบไดเรกทอรีอย่างน้อยหนึ่งไดเรกทอรีในพาท `/var/adm/ffdc/dumps` เปลี่ยนอาจถูกเปลี่ยนบนไดเรกทอรีอย่างน้อย หนึ่งไดเรกทอรีในพาทนี้เพื่อป้องกันการเข้าถึง
- 32 ไดเรกทอรี `/var/adm/ffdc/dumps` ไม่ถูกเมาท์ หรือไม่มีอยู่
- 40 มีพื้นที่ว่างไม่เพียงพอในไดเรกทอรี `/var/adm/ffdc/stacks` ในการสร้าง FFDC Error Stacks บนโลคัลโหนด
- 41 ไม่สามารถจัดหาข้อมูลระบบไฟล์จากระบบปฏิบัติการ คำนี้ระบุว่าอาจมีปัญหากเกิดขึ้นในระบบปฏิบัติการเอง
- 42 การสร้างและการใช้งาน FFDC Error Stack ถูกปิดใช้งานในสภาพแวดล้อม กระบวนการนี้

## ตัวอย่าง

ในการตรวจหาปัญหาที่อาจเกิดขึ้นกับยูทิลิตี้ FFDC บนโลคัลโหนด:

```
fccheck
สถานะ fccheck: การทดสอบทั้งหมดเสร็จสมบูรณ์
```

หากโลคัลโหนดถูกปิดใช้งานมิให้สร้าง FFDC Error Stacks `fccheck` จะระบุว่าเป็นปัญหา:

fccheck

สถานะ fccheck: การสร้างและใช้ FFDC Error Stacks ได้ถูกปิดใช้งาน  
ปิดใช้งานในสถานะแวดล้อมการเรียกใช้งานปัจจุบัน กระบวนการใดๆ ที่สร้างขึ้นใน  
สถานะแวดล้อมการเรียกใช้งานปัจจุบันจะไม่สามารถสร้าง FFDC Error Stacks ของตนเอง  
หรือสืบทอดการใช้ FFDC Error Stacks ที่มีอยู่แล้วได้

fccheck Status: การตรวจสอบทั้งหมดเสร็จสมบูรณ์ ตรวจสอบเอาต์พุตสถานะก่อนหน้าสำหรับ  
เงื่อนไขปัญหา FFDC ที่อาจเกิดขึ้น และทำตามการดำเนินการที่แนะนำที่แสดงรายการใน  
ข้อความเหล่านี้

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fcclear”

“คำสั่ง fcinit” ในหน้า 503

---

## คำสั่ง fcclear

### วัตถุประสงค์

ลบ FFDC Error Stacks และไฟล์ข้อมูลรายละเอียดออกจากโลคัลโหนด

### ไวยากรณ์

```
/usr/sbin/rsct/bin/fcclear -h | [-d filename [,filename,...]] [-D filename [,filename,...]] [-f FFDC_Failure_ID [,  
FFDC_Failure_ID,...]] [-F FFDC_Failure_ID [,FFDC_Failure_ID,...]] [-s file_name [,filename,...]] [-S file_name  
[,filename,...]] [-t days ]]
```

### คำอธิบาย

fcclear ถูกใช้เพื่อลบไฟล์ FFDC Error Stack ที่ไม่จำเป็นสำหรับการพิจารณาปัญหาจากโลคัลโหนด ไฟล์ FFDC Error Stack  
ที่ระบุสามารถลบได้ รวมถึงไฟล์ FFDC Error Stack ที่มีเรีกคอร์ดของ FFDC Failure Identifiers ที่ระบุรายการ แต่ละรายการ  
ภายใน FFDC Error Stack ไม่สามารถลบได้

การใช้ตัวเลือก -t สามารถใช้ fcclear เพื่อลบไฟล์ FFDC Error Stack ที่มีอายุเก่ากว่าจำนวนวันที่ระบุในการใช้ fcclear ใน  
ลักษณะที่เป็นการทำงานอัตโนมัติเพื่อล้างค่า FFDC Error Stacks ที่ไม่จำเป็น ดูที่คำสั่ง cron สำหรับ การทำให้การเรียกใช้คำสั่ง  
เป็นอัตโนมัติ

ในการลบ FFDC Error Stacks ทั้งหมดออกจากโลคัลโหนด ให้ระบุค่า ศูนย์ (0) สำหรับอาร์กิวเมนต์ตัวเลือกจำนวนวัน

### แฟล็ก

-d      ลบไฟล์ข้อมูลรายละเอียดโดยการระบุรายการของชื่อไฟล์ข้อมูลรายละเอียด อย่างน้อยหนึ่งชื่อ ชื่อไฟล์เหล่านี้ อาจ

เป็นชื่อพาสเวิร์ดหรือสัปดาห์ไปยังไดเรกทอรี `/var/adm/ffdc/dumps` ไฟล์เหล่านี้จะถูกลบหากมีอยู่บนโลคัลโหนด ไฟล์บนรีโมตโหนดไม่สามารถลบได้โดยใช้คำสั่งนี้ หากมีการกำหนดมากกว่าหนึ่งชื่อไฟล์ ต้องคั่นด้วยเครื่องหมายจุลภาค (,) โดยไม่มี white space ใดๆ แทรก

- D สแกนไฟล์ข้อมูลรายละเอียดโดยการระบุรายการของชื่อไฟล์ข้อมูลรายละเอียด อย่างน้อยหนึ่งชื่อชื่อไฟล์เหล่านี้จะเป็นชื่อพาสเวิร์ดหรือสัปดาห์ไปยังไดเรกทอรี `/var/adm/ffdc/dumps` ไฟล์เหล่านี้ จะถูกเก็บรักษาไว้หากมีอยู่บนโลคัลโหนด ไฟล์บนรีโมตโหนดไม่สามารถเก็บรักษาไว้ได้ โดยใช้คำสั่งนี้ หากมีการกำหนดมากกว่าหนึ่งชื่อไฟล์ ต้องคั่น ด้วยเครื่องหมายจุลภาค (,) โดยไม่มี white space ใดๆ แทรก
- f ลบไฟล์ FFDC Error Stack โดยการระบุรายการของ FFDC Failure Identifiers อย่างน้อยหนึ่งค่า FFDC Error Stacks ที่เชื่อมโยงกับ FFDC Error Identifiers จะถูกกำหนดตำแหน่งและลบออกหากมีแสดงอยู่บนโลคัลโหนด FFDC Error Stacks บนรีโมตโหนดจะไม่ถูกลบ หากมีการระบุมากกว่าหนึ่ง FFDC Failure Identifier ต้องคั่นด้วยเครื่องหมายจุลภาค (,) ที่ไม่มี white space แทรก
- F สแกนไฟล์ FFDC Error Stack โดยการระบุรายการของ FFDC Failure Identifiers อย่างน้อยหนึ่งค่า FFDC Error Stacks ที่เชื่อมโยงกับ FFDC Error Identifiers จะถูกกำหนดตำแหน่งและเก็บรักษาหากมีแสดงอยู่บนโลคัลโหนด FFDC Error Stacks บนรีโมตโหนดจะไม่ถูกเก็บรักษา หากมีการระบุมากกว่าหนึ่ง FFDC Failure Identifier ต้องคั่นด้วยเครื่องหมายจุลภาค (,) ที่ไม่มี white space แทรก
- h แสดงข้อมูลวิธีใช้และการใช้งานไปยังอุปกรณ์เอาต์พุตมาตรฐาน ไม่มี การประมวลผลอื่น
- s ลบไฟล์ FFDC Error Stack โดยการระบุกรของชื่อไฟล์ FFDC Error Stack อย่างน้อยหนึ่งชื่อชื่อไฟล์เหล่านี้สามารถเป็นชื่อไฟล์พาสเวิร์ดหรือชื่อไฟล์สัปดาห์ไปยังไดเรกทอรี `/var/adm/ffdc/stacks` ไฟล์เหล่านี้ จะถูกลบหากมีอยู่บนโลคัลโหนด FFDC Error Stacks บนรีโมตโหนดไม่สามารถลบได้โดยใช้คำสั่งนี้ หากมีการกำหนดมากกว่าหนึ่งชื่อไฟล์ แต่ละชื่อ ต้องคั่นด้วยเครื่องหมายจุลภาค (,) โดยไม่มี white space ใดๆ แทรก
- S ลบไฟล์ FFDC Error Stack โดยการระบุกรของชื่อไฟล์ FFDC Error Stack อย่างน้อยหนึ่งชื่อชื่อไฟล์เหล่านี้สามารถเป็นชื่อไฟล์พาสเวิร์ดหรือชื่อไฟล์สัปดาห์ไปยังไดเรกทอรี `/var/adm/ffdc/stacks` ไฟล์เหล่านี้ จะถูกลบหากมีอยู่บนโลคัลโหนด FFDC Error Stacks บนรีโมตโหนดไม่สามารถลบได้โดยใช้คำสั่งนี้ หากมีการกำหนดมากกว่าหนึ่งชื่อไฟล์ แต่ละชื่อ ต้องคั่นด้วยเครื่องหมายจุลภาค (,) โดยไม่มี white space ใดๆ แทรก
- t บ่งชี้ว่า FFDC Error Stacks และไฟล์ข้อมูลรายละเอียดที่เก่ากว่า จำนวนวันที่ระบุควรถูกลบออกจากโลคัลโหนดเกณฑ์ การเลือกนี้ไม่ขึ้นกับเกณฑ์การเลือกอื่นๆ

## สถานะออก

**fcclear** จะสร้างค่าสถานะการออกต่อไปนี้เมื่อ ดำเนินการเสร็จสิ้น:

- 0 คำสั่งดำเนินการสำเร็จ คำสั่งอาจดำเนินการสำเร็จเรียบร้อย หากไม่มีไฟล์ FFDC Error Stack หรือไฟล์ข้อมูลรายละเอียดที่ตรงกับเกณฑ์การเลือก
- 2 ข้อมูลวิธีใช้ถูกแสดงเรียบร้อย ไม่มีการประมวลผลใดๆ อีก
- 10 ไม่มีไฟล์ถูกลบออกจากโลคัลท้องถิ่น อีพซันที่จำเป็นไม่ถูก ระบุในคำสั่งนี้
- 11 ไม่มีไฟล์ถูกลบออกจากโลคัลท้องถิ่น อาร์กิวเมนต์ของอีพซัน -t ไม่เป็นตัวเลข
- 12 ไม่มีไฟล์ถูกลบออกจากโลคัลท้องถิ่น มีการระบุอีพซันที่ไม่รู้จัก โดยผู้เรียกใช้
- 19 ไดเรกทอรี `/var/adm/ffdc/stacks` ไม่มีอยู่หรือไม่ถูกเมาท์

- 26 ไม่มีไฟล์ถูกลบออกจากโลคัลท็องถิ่น ระบุชื่อพจนานุกรมเหมือนกัน มากกว่าหนึ่งครั้ง
- 28 ไม่มีไฟล์ถูกลบออกจากระบบ ผู้เรียกใช้ได้กำหนดชื่อพจนานุกรมที่สั่งให้คำสั่งทั้งลบออกและเก็บรักษาไฟล์เดียวกันไว้ สถานะเช่นนี้สามารถเกิดขึ้นได้เมื่อผู้ใช้คำสั่งระบุ FFDC Failure Identifier ที่ถูกบันทึกในไฟล์ FFDC Error Stack ที่ระบุด้วยชื่อไปยังคำสั่งนี้

## ตัวอย่าง

ในการลบ FFDC Error Stack และไฟล์ข้อมูลรายละเอียดใดๆ ที่เก่ากว่าเจ็ดวัน ออกจากโลคัลโหนด

```
fcclear -t 7
```

ในการลบ FFDC Error Stack และไฟล์ข้อมูลรายละเอียดทั้งหมดที่เก่ากว่าเจ็ดวันออก แต่เก็บรักษา FFDC Error Stack ที่มีข้อมูลสำหรับ FFDC Failure Identifier /3Iv04ZVVfvp.wtY0xRXQ7..... ไว้ให้เรียกใช้คำสั่ง:

```
fcclear -t 7 -F /3Iv04ZVVfvp.wtY0xRXQ7.....
```

เมื่อต้องการลบไฟล์ FFDC Error Stack ที่มี เร็กคอร์ดสำหรับ FFDC Failure Identifier /3Iv04ZVVfvp.wtY0xRXQ7..... ให้เรียกใช้คำสั่ง:

```
fcclear -f /3Iv04ZVVfvp.wtY0xRXQ7.....
```

ในการลบไฟล์ FFDC Error Stack **myprog.14528.19990204134809** และ **a.out.5134.19990130093256** ออกจากระบบ รวมถึงไฟล์ข้อมูลรายละเอียด **myprog.14528.19990204135227**:

```
fcclear -s myprog.14528.19990204134809,a.out.5134.19990130093256
-d myprog.14528.19990204135227
```

ในการขยายคำสั่งก่อนหน้านี้ให้ลบไฟล์ที่ระบุชื่อรวมถึง FFDC Error Stack และไฟล์ข้อมูลรายละเอียดใดๆ ที่เก่ากว่า 14 วัน:

```
fcclear -s myprog.14528.19990204134809,a.out.5134.19990130093256
-d myprog.14528.19990204135227 -t 14
```

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fccheck” ในหน้า 493

“คำสั่ง fcreport” ในหน้า 521

“คำสั่ง fcstkrpt” ในหน้า 526

---

## คำสั่ง `fcdecode`

### วัตถุประสงค์

แปล First Failure Data Capture (FFDC) Failure Identifier จาก รูปแบบมาตรฐานให้เป็นส่วนคอมพิวเตอร์ โดยแสดงข้อมูลนี้ไปยัง อุปกรณ์เอาต์พุตมาตรฐานในรูปแบบที่สามารถอ่านเข้าใจได้

### ไวยากรณ์

```
/usr/sbin/rsct/bin/fcdecode FFDC_Failure_ID[,FFDC_Failure_ID,...] -h
```

### คำอธิบาย

`fcdecode` ถอดรหัส FFDC Failure Identifier 42 อักขระให้เป็นส่วนคอมพิวเตอร์ และแสดงสามส่วนในรูปแบบที่สามารถอ่านเข้าใจได้ เอาต์พุตของคำสั่งนี้แสดงข้อมูลต่อไปนี้ ซึ่งแยกมาจาก FFDC Failure Identifier:

- เนื้อเวิร์กแอตเดรส (ในรูปแบบ ASCII) ของโหนดที่มีรายงานอยู่
- เวลาที่ทำการบันทึกนี้แสดงโดยใช้การตั้งค่าเขตเวลาที่แอดคีย์ฟออยู่ขณะนี้
- ค่าใดค่าหนึ่งต่อไปนี้ ซึ่งขึ้นอยู่กับตำแหน่งที่ข้อมูลถูกบันทึก:
  - ID เพิ่มเพลต AIX Error Log ที่ใช้ทำการบันทึกนี้ หาก เร็กคอร์ดถูกบันทึกใน AIX Error Log บนโหนดนั้น หรือ
  - ชื่อของไฟล์ FFDC Error Stack ที่มีการบันทึกนี้ หาก เร็กคอร์ดเป็นไฟล์ใน FFDC Error Stack และ FFDC Error Stack อยู่บนโหนดนี้
- คำสั่งที่แนะนำที่สามารถใช้เพื่อจัดทำรายงานที่ระบุที่สัมพันธ์ กับ FFDC Failure Identifier นี้

### แฟล็ก

`-h` แสดงข้อความวิธีใช้ไปที่เอาต์พุตมาตรฐานและออก ไม่มีการประมวลผลอื่น ไม่ว่าจะมีการระบุอ็อปชันใด

### พารามิเตอร์

`FFDC_Failure_ID`

FFDC Failure Identifier ที่ส่งกลับจากการเรียกใช้ไปยังคำสั่ง `fcpushstk` และ `fclogerr` ก่อนหน้านี้ หรือ ส่งกลับจากการเรียกใช้ไปยังรูทีนย่อย `fc_push_stack` หรือ `fc_log_error` ก่อนหน้านี้ identifier นี้ระบุรายการใช้ทำรายงานความล้มเหลว หรือเหตุการณ์บันทึกอื่นๆ โดยสามารถระบุ FFDC Failure Identifier มากกว่าหนึ่งค่าเพื่อใช้เป็นอาร์กิวเมนต์ในคำสั่งนี้ได้ แต่อย่างไรก็ตาม แต่ละ identifier ต้องคั่นด้วยเครื่องหมายจุลภาค (,) โดยไม่มีการแทรกของ white space ระหว่าง identifiers

### สถานะออก

`fcdecode` ส่งคืนโค้ดสถานะเลขจำนวนเต็มค่าใดค่าหนึ่ง ต่อไปนี้เมื่อดำเนินการเสร็จเรียบร้อย:

- 0 FFDC Failure Identifier ถอดรหัสเสร็จเรียบร้อยแล้ว
- 2 มีการข้อมูลวิธีใช้แสดงและการประมวลผลสิ้นสุด

- 10 FFDC Failure Identifier ไม่ถูกจัดให้เป็นอาร์กิวเมนต์แก่คำสั่งนี้
- 12 อีพซันไม่ถูกต้องหรือไม่สนับสนุนถูกกำหนดให้แก่คำสั่งนี้
- 27 ไม่มีข้อมูลถูกเขียนไปยังอุปกรณ์เอาต์พุตมาตรฐาน อาร์กิวเมนต์ FFDC Failure Identifier ไม่ถูกต้อง

## ตัวอย่าง

FFDC Failure Identifier ถูกแสดงโดยค่าฐาน 64 อ่านจาก ขวาไปซ้าย แต่ละจุดแสดงศูนย์นำหน้า ในการถอดรหัส FFDC Failure Identifier **.3Iv04ZVVfvp.wtY0xRXQ7.....** ให้เป็นส่วนคอมโพเนนต์:

```
fcdecode .3Iv04ZVVfvp.wtY0xRXQ7.....
```

```
Information for First Failure Data Capture identifier
.3Iv04ZVVfvp.wtY0xRXQ7.....
Generated by the local system
Generated Thu Sep 3 11:40:17 1998 EDT
Recorded to the AIX Error Log using template 460bb505
To obtain the AIX Error Log information for this entry, issue
the following command on the local system:
TZ=EST5EDT errpt -a -j 460bb505 -s 0903114098 | more
Search this output for an AIX Error Log entry that contains
the following ERROR ID code:
.3Iv04ZVVfvp.wtY0xRXQ7.....
```

คำสั่งเดียวกันที่รันบนโหนดอื่นจะมีผลลัพธ์ต่อไปนี้:

```
fcdecode .3Iv04ZVVfvp.wtY0xRXQ7.....
```

```
Information for First Failure Data Capture identifier
.3Iv04ZVVfvp.wtY0xRXQ7.....
Generated on a remote system with the following Internet address:
9.114.55.125
Generated Thu Sep 3 11:40:17 1998 EDT
Recorded to the AIX Error Log using template 460bb505
TZ=EST5EDT errpt -a -j 460bb505 -s 0903114098 | more
Search this output for an AIX Error Log entry that contains
the following ERROR ID code:
.3Iv04ZVVfvp.wtY0xRXQ7.....
```

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fcdispfid” ในหน้า 500

“คำสั่ง fcreport” ในหน้า 521

“คำสั่ง fcstkprt” ในหน้า 526

---

## คำสั่ง `fcdispfid`

### วัตถุประสงค์

แสดง First Failure Data Capture Failure Identifier (FFDC Failure Identifier) ไปยังอุปกรณ์ข้อผิดพลาดมาตรฐาน

### ไวยากรณ์

```
/usr/sbin/rsct/bin/fcdispfid [-q]FFDC_Failure_ID|-h
```

### คำอธิบาย

คำสั่งนี้ถูกใช้งานโดยสคริปต์เพื่อแสดงค่า FFDC Failure Identifier ไปยังอุปกรณ์ข้อผิดพลาดมาตรฐาน อินเทอร์เน็ตเฟสนี้มีขึ้นเนื่องจากสคริปต์โปรแกรม ไม่มีสำหรับสำหรับการส่งข้อมูลกลับไปโคลเอ็นต์ของตุนยกเว้นจะส่ง ทางโค้ดสถานะการออกเอาต์พุตมาตรฐาน และข้อผิดพลาดมาตรฐาน ในการดำเนินงาน "การส่งกลับ" ค่า FFDC Failure Identifier ไปยังโคลเอ็นต์ในสถานะแวดล้อมเช่นนั้น ได้สำเร็จ `fcdispfid` จะใช้หมายเลขข้อความที่จัดทำแค็ตตาล็อก XPG/4 2615-000 ในการแสดงข้อมูลนี้ไปยังอุปกรณ์ข้อผิดพลาดมาตรฐาน โคลเอ็นต์ของสคริปต์สามารถดักจับข้อมูลข้อผิดพลาดมาตรฐาน ค้นหาหมายเลขข้อความที่ระบุ และจัดการ FFDC Failure Identifier จากสคริปต์

สคริปต์ต้องระบุว่า FFDC Failure Identifiers ใดๆ ที่สร้างโดย สคริปต์จะถูกนำไปที่อุปกรณ์ข้อผิดพลาดมาตรฐานในเอกสารคู่มือผู้ใช้ของอุปกรณ์ โดยค่าดีฟอลต์ โคลเอ็นต์ไม่สามารถทราบถึงลักษณะการทำงานนี้

### แฟล็ก

- h แสดงข้อความวิธีใช้ไปที่เอาต์พุตมาตรฐานและออก ไม่มีการประมวลผลอื่น ไม่ว่าจะมีการระบุอ็อปชันใด
- q ไม่แสดงข้อความเตือนจากคำสั่งนี้ หากไม่ได้กำหนดอ็อปชันนี้ คำสั่งนี้จะแสดงข้อความเมื่อตรวจพบ FFDC Failure Identifier ที่ไม่ถูกต้อง

### พารามิเตอร์

*FFDC\_Failure\_ID*

ระบุ FFDC Failure Identifier นี้เป็น identifier ที่ส่งกลับ จากการเรียกใช้ไปยัง `fcpushstk` หรือ `fclogerr` ก่อนหน้านี้ และระบุรายการที่ใช้สร้างรายงานความล้มเหลวที่พบ โดยสคริปต์ identifier นี้ถูกเขียนไปยังอุปกรณ์ข้อผิดพลาดมาตรฐานโดยใช้ข้อความ FFDC 2615-000

### สถานะออก

- 0 FFDC Failure Identifier ถูกแสดงไปยังข้อผิดพลาดมาตรฐาน
- 2 มีการข้อมูลวิธีใช้แสดงและการประมวลผลสิ้นสุด
- 12 ไม่มีข้อมูลถูกเขียนไปยังอุปกรณ์ข้อผิดพลาดมาตรฐาน มีการระบุอ็อปชันไม่ถูกต้อง
- 27 ไม่มีข้อมูลถูกเขียนไปยังอุปกรณ์ข้อผิดพลาดมาตรฐาน อาร์กิวเมนต์ *FFDC\_Failure\_ID* ไม่แสดงว่ามีรูปแบบที่ไม่ถูกต้อง

## ตัวอย่าง

ในการแสดง FFDC Failure Identifier ไปยังไคลเอ็นต์โดยอุปกรณ์เอาต์พุต มาตรฐาน:

```
FID=$(fclogerr -e FFDC_ERROR -t ERRID_SP_FFDCXEMPL_ER -i /usr/lpp/ssp/inc/
myprog.h -r myprog -s myprog.ksh -p $LINEPOS -v "1.1" -l PSSP -d $MINUSDOPTS -x
$MINUSXOPTS -y $MINUSYOPTS -b "myprog Configuration Failure - Exiting")
RC=$?
if ((RC == 0))
    ดังนั้น
        fcdispfid $FID
        return 1
    else
        :
    fi
```

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fcdecode” ในหน้า 498

“คำสั่ง fcfilter”

“คำสั่ง fcstkrpt” ในหน้า 526

---

## คำสั่ง fcfilter

### วัตถุประสงค์

ค้นหาและแสดง First Failure Data Capture (FFDC) Failure Identifiers ใดๆ ในไฟล์หรือในอินพุตมาตรฐาน อาจระบุได้มากกว่าหนึ่งไฟล์

### ไวยากรณ์

```
/usr/sbin/rsct/bin/fcfilter [ file_name ] [ ... ]
```

### คำอธิบาย

คำสั่งนี้สแกนไฟล์ต่างๆ ที่แสดงรายการเป็นอาร์กิวเมนต์สำหรับ First Failure Data Capture (FFDC) Failure Identifiers หากไม่กำหนดชื่อไฟล์เป็นอาร์กิวเมนต์ คำสั่งนี้จะตรวจสอบอินพุตมาตรฐานเพื่อหา FFDC Failure Identifiers หากตรวจพบ FFDC Failure Identifier **fcfilter** จะแสดง identifier ไปยังเอาต์พุตมาตรฐานบนบรรทัดของตนเอง

**fcfilter** สามารถใช้โดยสคริปต์เพื่อแยก FFDC Failure Identifiers ที่ส่งกลับโดยกระบวนการชดเชยจากอุปกรณ์ข้อผิดพลาดมาตรฐาน

หาก **fcfilter** ตรวจพบ FFDC Failure Identifier มากกว่าหนึ่ง ในอินพุต คำสั่งจะแสดง FFDC Failure Identifiers ที่พบทั้งหมด โดยแต่ละค่าอยู่บนบรรทัดเอาต์พุตแยก

## พารามิเตอร์

*file\_name*

ชื่อไฟล์ที่จะถูกค้นหา FFDC Failure Identifier สามารถระบุได้มากกว่าหนึ่งไฟล์ ถ้าไม่มีการระบุชื่อไฟล์ *fcfilter* จะอ่านจากอินพุตมาตรฐาน

## สถานะออก

*fcfilter* ส่งคืนโค้ดสถานะเลขจำนวนเต็มต่อไปนี้เมื่อดำเนินการเสร็จสิ้น:

- 0 *fcfilter* ดำเนินการเรียกใช้งานเสร็จสมบูรณ์ สถานะการออกนี้ไม่จำเป็นต้องหมายความว่า จะตรวจพบ FFDC Failure Identifiers ใดๆ
- >0 *fcfilter* ถูกขัดจังหวะหรือหยุดทำงานโดยสัญญาณ สถานะการออกเป็นค่าเลขจำนวนเต็มแสดงสัญญาณที่หยุดทำงาน คำสั่ง

## ตัวอย่าง

FFDC Failure Identifier ถูกแสดงโดยค่าฐาน 64 อ่านจากขวาไปซ้าย แต่ละจุดแสดงศูนย์นำหน้า ในการจัดหารายการของ FFDC Failure Identifiers ที่สร้างโดยการรันคำสั่ง *mycmd*:

```
mycmd 2> /tmp/errout
fcfilter /tmp/errout
/.00...JMr4r.p9E.xRXQ7.....
/.00...JMr4r.pMx.xRXQ7.....
```

ในการจัดการ FFDC Failure Identifier จากกระบวนการชายต์ในพาเรนต์สคริปต์ สคริปต์สามารถใช้คำสั่ง *fcfilter* ดังนี้:

```
RESULTS=$(mychild 2> /tmp/errout)
if (($? != 0)) # mychild ended in failure, get FFDC ID
    ดั้งนั้น
    cat /tmp/errout | fcfilter | read FIRST_FFDCID
else
    rm -f /tmp/errout
fi
```

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

สิ่งอ้างอิงที่เกี่ยวข้อง:

- “คำสั่ง *fcdispfid*” ในหน้า 500
- “คำสั่ง *fcreport*” ในหน้า 521
- “คำสั่ง *fcstkrpt*” ในหน้า 526

---

## คำสั่ง fcinit

### วัตถุประสงค์

สร้างหรือสืบทอดสถานะแวดล้อมการเรียกใช้งาน First Failure Data Capture

### ไวยากรณ์

สำหรับบอร์นและคอร์นเชลล์:

```
/usr/sbin/rsct/bin/fcinit.sh [[ -l ] [ -s { c | i } ] ] [ -h ]
```

สำหรับ C เชลล์:

```
source /usr/sbin/rsct/bin/fcinit.csh [[ -l ] [ -s { c | i } ] ] [ -h ]
```

### คำอธิบาย

อินเตอร์เฟซนี้ต้องใช้โดยสคริปต์โปรแกรมที่ต้องการใช้ อินเตอร์เฟซ FFDC สำหรับการบันทึกข้อมูลไปยัง AIX Error Log BSD System Log หรือ FFDC Error Stack

แอฟพลิเคชันอาจต้องการสร้าง FFDC Environment ด้วยเหตุผลข้อใดข้อหนึ่ง ต่อไปนี้:

- สคริปต์อาจต้องการบันทึกข้อมูลไปยัง AIX Error Log สคริปต์สามารถใช้ **fcinit** เพื่อสร้าง FFDC Environment พื้นฐาน
- สคริปต์ต้องการให้มีกระบวนการสคริปต์เอง และกระบวนการต่างๆ ที่สืบทอด ที่สร้างโดย กระบวนการนี้ หรือชายด์เพื่อ บันทึกข้อมูลความล้มเหลวไปยัง FFDC Error Stack ในกรณี สคริปต์จะพิจารณาว่าตัวเองเป็นแอฟพลิเคชัน "ระดับบนสุด" ที่จะทำให้เกิดแอฟพลิเคชัน "ระดับต่ำกว่า" อื่นๆ แอฟพลิเคชัน และความสำเร็ของ แอฟพลิเคชัน "ระดับบนสุด" นั้นขึ้นอยู่กับความสำเร็จของแอฟพลิเคชัน "ระดับต่ำกว่า" ทั้งหมดเหล่านี้ เมื่อใช้ **fcinit** ในลักษณะนี้ กระบวนการจะถูกแจ้งให้ จัดทำ หรือ สร้าง FFDC Error Stack Environment
- สคริปต์ใช้ FFDC Error Stack หรือ FFDC Trace ในกรณีเหล่านี้เท่านั้น เมื่อสคริปต์ถูกเรียกใช้โดยกระบวนการที่นำหน้าที่ ต้องการข้อมูลความล้มเหลว หรือติดตามข้อมูลที่บันทึกในอุปกรณ์เหล่านี้ ในกรณีอื่นๆ ทั้งหมด สคริปต์ ไม่ต้องการใช้ อุปกรณ์เหล่านี้ เมื่อใช้ **fcinit** ใน ลักษณะนี้ กระบวนการจะถูกแจ้งให้ *สืบทอด* FFDC Error Stack Environment

กระบวนการใดๆ ที่ต้องการบันทึกข้อมูลไปยัง AIX Error Log หรือ BSD System Log โดยอินเตอร์เฟซ FFDC ต้องสร้าง FFDC Environment หากกระบวนการไม่ต้องการใช้งาน FFDC Error Stack กระบวนการ จะสามารถสร้าง FFDC Environment พื้นฐานที่ไม่มีการใช้งาน FFDC Error Stack สำหรับ FFDC Error Stack Environment ซึ่งมี FFDC Error Stack จะถูกสร้างขึ้นโดย กระบวนการเมื่อต้องการให้ข้อมูลความล้มเหลว ของกระบวนการ เธรดที่สร้าง และกระบวนการที่สร้างในภายหลังถูก บันทึกไว้ใน FFDC Error Stack *FFDC Error Stack Environment* ซึ่งมี FFDC Error Stack นั้นถูกสืบทอดโดย กระบวนการเมื่อกระบวนการนั้นต้องการบันทึกข้อมูลความล้มเหลวไปยังไฟล์ FFDC Error Stack ต่อเมื่อมีกระบวนการที่เกิดก่อนกระบวนการหนึ่งได้ ร้องขอกระบวนการ ให้ทำ ในกรณีอื่นๆ กระบวนการจะไม่บันทึกข้อมูลความล้มเหลว ไปยัง FFDC Error Stack

FFDC Error Stack Environment ซึ่งมี FFDC Error Stack จะสงวน ไฟล์ FFDC Error Stack เพื่อที่ข้อมูลความล้มเหลวจะถูก บันทึกในไฟล์ ในไดเรกทอรี `/var/adm/ffdc/stacks` ไฟล์เหล่านี้ใช้รูปแบบการตั้งชื่อ `script_name.PID.date_and_time` โดยที่

*script\_name* คือ ชื่อของสคริปต์เอง *PID* คือ identifier กระบวนการของสคริปต์ และ *date\_and\_time* คือ วันที่และเวลาเมื่อสคริปต์ถูกเรียกใช้งาน เมื่อใดที่สคริปต์นี้ หรือกระบวนการลูก ของสคริปต์นี้บันทึกข้อมูลความล้มเหลวไปยัง FFDC Error Stack ข้อมูลนั้นจะถูกบันทึกในไฟล์นี้ด้วย

เพื่อให้กระบวนการทำการบันทึกข้อมูลใน FFDC Error Stack กระบวนการต้องใช้อินเตอร์เฟส *fcpushstk* FFDC และกระบวนการต้องมีการดำเนินการภายใน FFDC Error Stack Environment ที่สร้างขึ้น หากไม่มี FFDC Error Stack Environment หรือหากอินเตอร์เฟส *fcpushstk* ไม่ถูกใช้เมื่อมี FFDC Error Stack Environment จะไม่มีข้อมูลถูกบันทึกโดยกระบวนการนั้นใน FFDC Error Stack ฟังก์ชันนี้ยอมให้กระบวนการรันในโหมดปกติหรือ "ไม่โต้ตอบ" เมื่อไม่ต้องการหรือไม่จำเป็นต้องใช้ข้อมูลการดีบั๊กความล้มเหลว แต่ยังคงยอมให้ข้อมูลนี้พร้อมใช้ได้เมื่อกระบวนการถูกเรียกใช้ภายใน สภาวะแวดล้อมพิเศษสำหรับการดีบั๊ก

*fcinit* ต้องถูกเรียกใช้งานภายในสภาวะแวดล้อมกระบวนการ ("sourced") ของไคลเอ็นต์ FFDC เพื่อให้คำสั่งตั้งค่า FFDC Environment สำหรับสคริปต์ได้อย่างเหมาะสม ไคลเอ็นต์ FFDC ที่อิงตามสคริปต์ที่ใช้คำสั่งนี้ ต้อง "source" คำสั่งเพื่อให้ *fcinit* เรียกใช้งาน ภายในอิมเมจกระบวนการของไคลเอ็นต์ หากไม่ทำวิธีนี้ อินเตอร์เฟส FFDC จะถูกเรียกใช้งานภายในอิมเมจกระบวนการของตนเอง การตั้งค่าใดๆ ของ FFDC Environment จะสูญหายหลังจากอินเตอร์เฟส FFDC ดำเนินการเสร็จในการแสดงวิธีที่แอ็พพลิเคชันที่อิงตามสคริปต์จะ "source" คำสั่ง *fcinit* คอร์นเชลล์ โปรแกรมจะเรียกใช้คำสั่งต่อไปนี้:

```
. fcinit.sh <อ็อพชันและอาร์กิวเมนต์>
```

C เชลล์สคริปต์จะดำเนินการสิ่งต่อไปนี้:

```
source fcinit.csh <อ็อพชันและอาร์กิวเมนต์>
```

กระบวนการที่ใช้อินเตอร์เฟส *fclogerr* FFDC ต้อง สร้าง *FFDC Environment* หากกระบวนการต้องการ ใช้อินเตอร์เฟส *fclogerr* เท่านั้น *FFDC Environment* สามารถถูกสร้างโดยไม่มี FFDC Error Stack

หาก FFDC Environment นั้นมีอยู่แล้วเมื่อสคริปต์พยายามที่จะสร้างขึ้น สคริปต์จะสืบทอด FFDC Environment ที่มีอยู่แล้ว แทนการสร้างของตนเอง

## แฟล็ก

- h แสดงข้อความวิธีใช้ไปที่เอาต์พุตมาตรฐานและออก ไม่มีการประมวลผลอื่น ไม่ว่าจะมีการระบุอ็อพชันใด
- l ระบุว่ากระบวนการต้องการใช้งาน AIX Error Log เท่านั้น อ็อพชันนี้ไม่จำเป็นต้องใช้เมื่อระบุอ็อพชัน -s เนื่องจากการใช้ AIX Error Log ได้รับอนุญาตให้ใช้ได้ภายใน FFDC Error Stack Environment
- s ระบุว่า FFDC Error Stack Environment จะถูกสร้างขึ้น แอ็พพลิเคชันที่ต้องการใช้อินเตอร์เฟส *fcpushstk* ต้องระบุแฟล็กนี้ เมื่อการดำเนินการคำสั่งนี้สำเร็จเรียบร้อย FFDC ไฟล์ Error Stack file จะถูกสงวนไว้สำหรับสคริปต์ในไดเรกทอรี */var/adm/ffdc/stacks* แฟล็กนี้ต้องถูกระบุด้วยอ็อพชันที่เป็นไปได้หนึ่งในสองอ็อพชัน:
  - c ร้องขอให้ FFDC Error Stack Environment ถูก *สร้างขึ้น* หาก FFDC Error Stack Environment ไม่ถูกสร้างขึ้น โดยกระบวนการ ที่นำหน้า จะถูกสร้างขึ้นในตอนนี หากสภาวะแวดล้อมเช่นนั้นถูกสร้างขึ้นก่อนหน้านั้น โดยกระบวนการที่นำหน้า กระบวนการนี้จะ *สืบทอด* FFDC Error Stack Environment เหมือนกับที่ระบุระบุ *i*
  - i ระบุว่า FFDC Error Stack Environment จะถูก *สืบทอด* หากถูกสร้างขึ้นแล้วก่อนหน้านั้นโดยกระบวนการที่มา ก่อน หาก ก่อนหน้านั้น FFDC Error Stack Environment ยังไม่ถูกสร้างขึ้นโดยกระบวนการ ที่มาก่อน FFDC

Error Stack Environment จะไม่ถูกสร้างขึ้นสำหรับกระบวนการนี้ และกระบวนการนี้ไม่สามารถใช้งาน FFDC Error Stack ได้ (แม้ว่าจะใช้งาน AIX Error Log และ BSD System Log)

## พารามิเตอร์

*file\_name*

ชื่อไฟล์ที่จะถูกค้นหา FFDC Failure Identifier สามารถระบุได้มากกว่าหนึ่งไฟล์ ถ้าไม่มีการระบุชื่อไฟล์ *fcfilter* จะอ่านจากอินพุตมาตรฐาน

## สถานะออก

**fcinit** ส่งคืนโค้ดสถานะการออกต่อไปนี้เมื่อดำเนินการเสร็จสิ้น:

- 0 FFDC Environment สร้างขึ้นเรียบร้อย
- 1 FFDC Environment สืบทอดเรียบร้อย
- 2 มีการขอวิธีใช้แสดงและการประมวลผลสิ้นสุด

**fcinit** ส่งคืนโค้ดสถานะการออกต่อไปนี้เมื่อตรวจพบความล้มเหลว:

- 12 FFDC Environment ไม่ถูกสร้างหรือสืบทอด - กำหนดฟังก์ชันพารามิเตอร์ที่ไม่รู้จัก
- 13 FFDC Error Stack Environment ไม่ถูกสร้างหรือสืบทอด - ผู้เรียกใช้ระบุว่า FFDC Environment ควรทั้งถูกสร้างและสืบทอด
- 14 FFDC Environment ไม่ถูกสร้างในการเรียกใช้ - ผู้เรียกใช้มี FFDC Environment ถูกสร้างสำหรับผู้เรียกใช้อยู่แล้ว - รุทีนี้อาจถูกเรียกใช้งาน หลายครั้ง
- 15 FFDC Error Stack Environment ไม่ถูกสร้างหรือสืบทอด - ไม่มี FFDC Error Stack Environment และระบุอ็อปชัน FC\_INHERIT
- 16 FFDC Environment ไม่ถูกสร้างหรือสืบทอด - สถานะแวดล้อมกระบวนการ ของไคลเอ็นต์ไม่สามารถถูกแก้ไขโดยรุทีนี้
- 17 FFDC Environment ไม่ถูกสร้างหรือสืบทอด - FFDC Environment อาจเสียหาย และควรถูกพิจารณาว่าไม่สามารถใช้งานได้
- 18 FFDC Environment ไม่ถูกสร้างหรือสืบทอด - รุทีนี้ไม่สามารถจัดสรรหน่วยความจำที่จำเป็นต้องใช้เพื่อแก้ไขสถานะแวดล้อมกระบวนการของไคลเอ็นต์
- 19 FFDC Error Stack Environment ไม่ถูกสร้างหรือสืบทอด - ไม่สามารถ สงวนไฟล์ FFDC Error Stack สำหรับกระบวนการที่เรียกใช้ - ไม่มีไดเรกทอรี FFDC Error Stack หรือไม่สามารถใช้ได้
- 21 FFDC Error Stack Environment ไม่ถูกสร้างหรือสืบทอด - ไม่สามารถ สงวนไฟล์ FFDC Error Stack สำหรับกระบวนการที่เรียกใช้ - ไฟล์มีอยู่แล้ว
- 42 FFDC Error Stack Environment ไม่ถูกสร้างหรือสืบทอด - การสร้าง และใช้งาน FFDC Error Stacks ถูกปิดใช้งานโดยผู้ดูแลระบบ สคริปต์สามารถสร้างได้เฉพาะ FFDC Environment พื้นฐานที่ใช้งาน AIX Error Log และ BSD System Log เท่านั้น

- 99 FFDC Environment ไม่ถูกสร้างหรือสืบทอด - เกิดความล้มเหลวภายในที่ไม่คาดคิดขึ้นภายใน fcinit สถานะนี้อาจจำเป็นต้องได้รับความสนใจจากลูกค้าและฝ่ายบริการสนับสนุนแอปพลิเคชัน

## ตัวอย่าง

สำหรับคอร์เนลล์สคริปต์ที่จะสร้าง FFDC Environment พื้นฐานสำหรับใช้ AIX Error Log และ BSD System Log เท่านั้น (FFDC Error Stack ไม่ถูกใช้หรือสงวนไว้):

```
# Set up an FFDC Environment to use the AIX Error Log only. An FFDC Error
# Stack is not needed for this script.
. fcinit.sh -l
rc=$?
if ((rc != 0))
    ดังนั้น
        print "fcinit failed with exit code of $rc"
        exit 1
fi
# Normal processing starts
```

สำหรับคอร์เนลล์สคริปต์ที่จะสร้าง FFDC Error Stack Environment ที่ทำให้สคริปต์และกระบวนการที่จะตามมาใดๆ บันทึกข้อมูลความล้มเหลวไปยัง FFDC Error Stack:

```
# Set up FFDC Environment to record failure information to the FFDC Error
# Stack
. fcinit.sh -sc
rc=$?
if ((rc != 0))
    ดังนั้น
        print "fcinit failed with a code of $rc"
        exit 1
fi
# Normal processing starts
```

**หมายเหตุ:** โคลเอ็นต์ FFDC อาจได้รับการแจ้งว่า FFDC Error Stack Environment ถูกสืบทอดแทน การสร้างโดยการเรียกใช้ fcinit ซึ่งจะเกิดขึ้นเมื่อ FFDC Error Stack Environment ถูกสร้างไว้แล้วโดยกระบวนการหนึ่งของกระบวนการที่มาก่อน

ในการสืบทอด FFDC Error Stack Environment จากประมวลผลพาเรนต์ของสถานะแวดล้อม:

```
# Inherit an FFDC Environment from parent process if it exists - otherwise,
# operate in a normal "silent" mode
. fcinit.sh -si
rc=$?
if ((rc != 0))
    ดังนั้น
```

```
print "fcinit failed with a code of $rc"
exit 1
fi
# Normal processing starts
```

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fccheck” ในหน้า 493

“คำสั่ง fclogerr”

“คำสั่ง fcpushstk” ในหน้า 515

“คำสั่ง fcteststk” ในหน้า 528

---

## คำสั่ง fclogerr

### วัตถุประสงค์

บันทึกข้อมูลเกี่ยวกับสถานะความล้มเหลวหรือสำคัญไปที่บันทึกข้อผิดพลาด AIX และบันทึกระบบ BSD

### ไวยากรณ์

```
/usr/sbin/rsct/bin/fclogerr { -e event -t error_template_label -i error_template_headerfile -r resource -s
source_filename -p line_of_code_pos -v sidlevel -l lpp_name -a assoc_fid { [-d detail_data_item[,detail_data_item,...]
-x detail_data_type[,detail_data_type,...] -y detail_data_len[,detail_data_len,...] } | [-f detail_data_file] } -b
BSD_syslog_message_text } | -h
```

### คำอธิบาย

อินเตอร์เฟซนี้ถูกใช้โดยสคริปต์โปรแกรมที่ต้องการบันทึกข้อมูล ไปที่บันทึกข้อผิดพลาด AIX และบันทึกระบบ BSD ข้อมูลที่เขียน ไปที่อุปกรณ์นี้ถูกกำหนดไว้สำหรับใช้โดยผู้ดูแลระบบ หรือโอเปอเรเตอร์ เพื่อกำหนดสถานะของความล้มเหลวหรือสถานะสำคัญที่เกิดขึ้น บนระบบที่ต้องให้ความสนใจ จุดประสงค์ของบันทึกข้อผิดพลาด AIX และบันทึกระบบ BSD คือเพื่อบันทึกข้อมูลเกี่ยวกับสถานะ เพื่อที่สามารถระบุ ลักษณะ ผลกระทบ และการตอบสนองต่อสถานะ จากรายงาน โดยไม่ต้องมีการสร้างสถานะขึ้นใหม่เพื่อตรวจจับ สถานะอะไรที่เกิดขึ้นและเกิดขึ้นที่ใด ซอฟต์แวร์ที่พบสถานะความล้มเหลวถาวร ที่จะคงอยู่จนกว่าจะมีการเข้ามาแก้ไขโดยตรง หรือพบสถานะที่ควรนำมาพิจารณา ของผู้ดูแลระบบ ควรใช้ fclogerr เพื่อ บันทึกข้อมูลนี้ในบันทึกข้อผิดพลาด AIX และบันทึกระบบ BSD

สคริปต์ควรสร้างสถานะแวดล้อม FFDC พื้นฐานหรือ FFDC Error Stack Environment ก่อนใช้ fclogerr โดยการสร้าง หรือสืบทอดสถานะแวดล้อม fclogerr บันทึกข้อมูล ไปที่บันทึกข้อผิดพลาด AIX และบันทึกระบบ BSD แม้ว่าสถานะแวดล้อมเหล่านี้ไม่ถูกสร้างขึ้น แต่อินเตอร์เฟซจะไม่สามารถสร้าง FFDC Failure Identifier นอกจากหนึ่งในสถานะแวดล้อมเหล่านี้มีอยู่

กระบวนการที่ออกแบบให้ใช้ FFDC Error Stack ยังสามารถใช้อินเตอร์เฟซ fclogerr และควรใช้อินเตอร์เฟซ ถ้าพบ สถานะที่ต้องการการดูแลหรือเข้ามาแก้ไขของผู้ดูแลระบบ

เพื่อรับประกันการระบุอย่างถูกต้องของสถานะและตำแหน่ง ซึ่งพบข้อมูล, FFDC Policy แนะนำว่า **fclogerr** ควรถูกเรียกแบบ in-line ในโมดูลซอร์สโค้ดของสคริปต์ และเรียกใช้ทันทีที่ตรวจพบสถานะ **fclogerr** จะบันทึกชื่อซอร์สโค้ดและข้อมูลบรรทัดของโค้ด เพื่อช่วยเหลือในการระบุและหาตำแหน่งซอร์สโค้ดที่พบสถานะความล้มเหลว **fclogerr** สามารถถูกเรียกโดยรูทีนย่อยหรือรันไทม์ที่โหลดอัตโนมัติ เพื่อบันทึกข้อมูลนี้ ถ้าจำเป็น จะมีการจัดเตรียมข้อมูลตำแหน่งทั้งหมดและข้อมูลรายละเอียดความล้มเหลวที่จำเป็น จะถูกจัดเตรียมให้พร้อมสำหรับรูทีนภายนอกนี้ รูทีนการบันทึกภายนอก ต้องบันทึกตำแหน่งจริงซึ่งเหตุการณ์ถูกตรวจพบ

แม้ว่า **fclogerr** รายงานข้อมูลไปที่บันทึกข้อผิดพลาด AIX และบันทึกระบบ BSD, อาจมีการกำหนดอ็อปชันต่างไปให้กับ อินเตอร์เฟซนี้สำหรับแต่ละอุปกรณ์บันทึก ข้อมูล Detail Data ที่บันทึก ลงบันทึกข้อผิดพลาด AIX จะไม่บันทึกลงบันทึกระบบ BSD ข้อมูลบันทึกระบบ BSD ถูกจัดเตรียมผ่านอ็อปชันคำสั่งอื่น ซึ่ง อาจต้องการให้ผู้ใช้ **fclogerr** ทำซ้ำข้อมูลบางส่วน ในการเรียกนี้

## แฟล็ก

- a มี FFDC Failure Identifier สำหรับสถานะความล้มเหลวที่รายงานโดย ซอฟต์แวร์ที่ใช้แ็พพลิเคชันนี้ ซึ่งทำให้หรือมีผลให้สถานะ ถูกบันทึกในเวลานี้ identifier นี้ควรถูกส่งกลับไป แ็พพลิเคชันนี้เป็นส่วนหนึ่งของการแสดงผลลัพธ์ของซอฟต์แวร์ ส่วนที่เรียกจัดเตรียม identifier นี้ที่นี้ เพื่อที่ FFDC Error Stack จะสามารถเชื่อมโยงรายงานความล้มเหลวที่กำลังสร้างขึ้นในเวลานี้ พร้อมกับรายการความล้มเหลวที่บันทึกไว้ก่อนหน้านี้ ซึ่งทำให้ตัวตรวจสอบปัญหาสามารถติดตามสาเหตุของความล้มเหลวจากอาการ ต่างๆ ในแ็พพลิเคชันและส่วนอื่น ต่อสาเหตุหลักในซอฟต์แวร์อื่น ถ้าไม่มีความล้มเหลวของซอฟต์แวร์อื่น ที่เป็นต้นเหตุของสถานะนี้ หรือถ้าซอฟต์แวร์อื่นไม่ส่งกลับ FFDC Failure Identifier เป็นส่วนหนึ่ง ของข้อมูลผลลัพธ์, ไม่ควรมีอ็อปชันนี้
- b ระบุข้อความที่จะเขียนไปที่บันทึกระบบ BSD
- d รายการข้อมูลที่จัดเตรียมข้อมูลรายละเอียดเกี่ยวกับสถานะ ที่ใช้เพื่อจัดเตรียม Detail Data ในรายการบันทึกข้อผิดพลาด AIX ถ้ารายละเอียด ของข้อมูลยาวเกินไป รายละเอียดเหล่านี้สามารถถูกเขียนลงในไฟล์ และชื่อของไฟล์นั้นที่จัดเตรียมเป็นพารามิเตอร์ *detail\_data\_file* ถ้าชื่อไฟล์ข้อมูลรายละเอียดถูกจัดเตรียม อ็อปชันนี้ควร ถูกละเว้น ถ้าทั้งพารามิเตอร์ *detail\_data* หรือ *detail\_data\_file* ไม่ถูกจัดเตรียมหรือไม่ถูกต้อง ข้อมูล null จะถูกบันทึกสำหรับข้อมูลรายละเอียดในบันทึกข้อผิดพลาด AIX  
  
สามารถหน่วยข้อมูลให้กับอ็อปชันนี้ได้มากกว่าหนึ่งรายการ แต่ละหน่วยข้อมูล ต้องถูกแยกด้วยคอมมา (,) โดยไม่มีอักขระ whitespace ถ้าหน่วยข้อมูลมีอักขระ whitespace ฝังอยู่ หน่วยข้อมูลต้องถูกปิด ในอัญประกาศ (") ตัวหน่วยข้อมูลเองต้องไม่มีคอมมา (,) เนื่องจากคำสั่งแปลคอมมาเป็นตัวแยกฟิลด์  
  
อ็อปชันนี้ ต้อง ใ้รวมกับอ็อปชัน -x และ -y
- e ระบุ FFDC Log Event Type ค่าที่ถูกต้องของ Current คือ FFDC\_EMERG, FFDC\_ERROR, FFDC\_STATE, FFDC\_TRACE, FFDC\_RECOV และ FFDC\_DEBUG โค้ดนี้ ให้รายละเอียดทั่วไปของชนิดเหตุการณ์ที่ถูกบันทึก (สถานะฉุกเฉิน, สถานะ авар, การแจ้งเตือนข้อมูล, ข้อมูลการข้อมูล และอื่นๆ) และความรุนแรงของสถานะ ถ้าไม่ระบุอ็อปชันนี้ ชนิดเหตุการณ์ FFDC\_DEBUG จะถูกกำหนดให้กับเรีกคอร์ดที่เกิดขึ้นนี้
- f ชื่อของไฟล์ที่มีรายละเอียดเกี่ยวกับสถานะที่ถูกรายงาน อ็อปชันนี้ถูกใช้เมื่อรายละเอียดยาวเกินไปที่จะบันทึกภายใน 100 ไบต์ที่เหลือของข้อมูล Detail Data ที่เหลือไว้ให้กับแ็พพลิเคชันโดย **fclogerr**, หรือเมื่อมียูทิลิตี้ที่สามารถวิเคราะห์ ข้อมูลรายละเอียด เนื้อหาของไฟล์นี้ถูกตัดลอกไปที่ไดเรกทอรี /var/adm/ffdc/dumps และตำแหน่งใหม่ถูกบันทึก เป็น Detail Data ในรายการบันทึกข้อผิดพลาด AIX
- h แสดงข้อความวิธีใช้ที่เอาต์พุตมาตรฐานและออก ไม่มีการประมวลผล อื่น ไม่ว่าจะมีการระบุอ็อปชันใด

- i ระบุชื่อพารามิเตอร์ของไฟล์ส่วนหัว (.h) ที่มี หมายเลขการระบุเพิ่มเพลดการบันทึกข้อผิดพลาดที่ตรงกับ `error_template_label` ที่ระบุในอ็อปชัน `-i` เพิ่มเพลดนี้ต้องมีอยู่ในที่เก็บเพิ่มเพลดการบันทึกข้อผิดพลาด ของโหนด เช่นกัน (`/var/adm/ras/errtmpl`) ไฟล์ส่วนหัว นี้ถูกสร้างโดยคำสั่ง `errupdate` เป็นส่วนหนึ่ง ของโปรแกรมการก่อสร้างซอร์สโค้ด และควรถูกรวมไว้ใน การทำแพ็คเกจของ LPP ที่จะถูกติดตั้งบนโหนดกับซอฟต์แวร์ ถ้าอ็อปชันนี้ ไม่ถูกระบุหรือไม่พบไฟล์ส่วนหัวเมื่อสคริปต์ถูกดำเนินการ `fclogerr` จะบันทึกข้อมูลความล้มเหลว โดยใช้เพิ่มเพลดข้อผิดพลาดดีฟอลต์ของตัวเอง (เลเบล `FFDC_DEF_TPLT_TR`, โค้ด `identifier 2B4F5CAB`)
- l ระบุตัวย่อของชื่อของผลิตภัณฑ์โปรแกรมมิ่งที่มีไลเซนส์ ซึ่งซอฟต์แวร์นี้ถูกจัดส่ง ค่านี้ควรเป็นที่รู้จักต่อ ลูกค้าและ เซอร์วิสการสนับสนุนแอสพลิคชัน เป็นชื่อที่ยอมรับได้สำหรับ LPP ตัวอย่างของค่าดังกล่าว: `PSSP`, `GPFS™`, `LoadLeveler®` และ `RSCT` ถ้าอ็อปชันนี้ไม่มี หรือไม่ถูกต้อง สตริงอักขระ `PPS_PRODUCT` จะถูกใช้
- p ระบุบรรทัดของตำแหน่งโค้ด ภายในโมดูลซอร์สโค้ดที่สภาวะ ถูกรายงาน ค่าที่จัดเตรียมต้องเป็นค่าจำนวนเต็ม ที่ถูก ต้อง เพื่อให้การระบุและตำแหน่งของสภาวะ ที่ถูกต้องค่านี้ ควรใกล้เคียงกับบรรทัดของโค้ดที่ตรวจพบสภาวะ เทาที่เป็นไปได้ สคริปต์ Korn Shell สามารถใช้ค่าของ `$LINENO` ภาษาสคริปต์ที่ไม่มีตัวแปรจำนวนบรรทัดพิเศษ สามารถ จัดเตรียมค่าเชิงสัญลักษณ์ที่นี้ ซึ่งผู้พัฒนาสามารถใช้เพื่อค้นหาจุด ในซอร์สโค้ดซึ่ง `fclogerr` กำลังถูกใช้ ถ้า อ็อปชันนี้ ใช้ไม่ได้หรือไม่ มี จะใช้ค่า `0`
- q ยับยั้งการสร้างข้อความคำเตือนจากคำสั่ง คำเตือน ถูกสร้างเมื่อคำสั่งต้องแทนที่ข้อมูลดีฟอลต์สำหรับข้อมูลที่หายไป หรือเมื่อคำสั่งไม่สามารถใช้ได้ในการคัดลอก `detail_data_file` ไปที่ไดเรกทอรี `/var/adm/ffdc/dumps`
- r ระบุชื่อคอมโพเนนต์ซอฟต์แวร์นี้เป็นชื่อเชิงสัญลักษณ์สำหรับ ซอฟต์แวร์ที่สร้างรายงาน และควรเป็นที่รู้จักต่อ ลูกค้า และ เซอร์วิสการสนับสนุนแอสพลิคชัน สตริงอักขระถูกจำกัดอยู่ที่ 16 อักขระ
- s ระบุชื่อของไฟล์ต้นฉบับที่มีบรรทัดของโค้ดที่ พบสภาวะที่ถูกรายงาน สำหรับสคริปต์ Korn and Borne Shell อาร์กิวเมนต์สำหรับอ็อปชันนี้ควรถูกเซตเป็น `$0`; สคริปต์ C Shell จะเซตอาร์กิวเมนต์นี้เป็น `{0}` ถ้าไม่มีอ็อปชันนี้ หรือไม่ ถูกต้อง สตริงอักขระ `unknown_file` จะถูกใช้
- t ระบุเลเบลเชิงสัญลักษณ์ที่กำหนดให้กับเพิ่มเพลดการบันทึกข้อผิดพลาด AIX ในที่เก็บบันทึกข้อผิดพลาด คำสั่ง `errupdate` ที่สร้างเพิ่มเพลดการบันทึกข้อผิดพลาดสร้างแม่โครที่แม่เพลดนี้กับ โค้ดจำนวนเต็ม เลเบลนี้เริ่มต้น ด้วยอักขระ `ERRID_` และมีค่าสูงสุด 19 อักขระ ถ้าอ็อปชันนี้ ไม่ถูกระบุหรือไม่พบไฟล์ส่วนหัวเมื่อสคริปต์ถูกดำเนินการ `fclogerr` จะเรียก `errlogger` เพื่อสร้าง ข้อความในบันทึกข้อผิดพลาด AIX โดยใช้เพิ่มเพลด `OPMSG`
- v ระบุหมายเลขเวอร์ชัน SCCS ของโมดูลซอร์สโค้ดที่ตรวจพบ สภาวะที่ถูกบันทึก สำหรับซอร์สโค้ดที่สร้างภายใต้การควบคุม SCCS นี้ควร ถูกเซตเป็น "1.1" (อัญประกาศจำเป็นต้องมี) ถ้าไม่มีอ็อปชันนี้หรือไม่ถูกต้อง สตริงอักขระ `unknown` จะถูกใช้
- x ระบุวิธีที่หน่วยข้อมูลที่ระบุโดยอ็อปชัน `-d` จะถูกแปล เมื่อบันทึกข้อมูลนี้ไปที่บันทึกข้อผิดพลาด AIX ชนิดเหล่านี้ ต้องเข้ากันกับฟิลด์ที่ตรงกันของเพิ่มเพลดการบันทึกข้อผิดพลาด AIX ที่ระบุในอ็อปชัน `-t` แต่ละ ชนิดระบุวิธีที่หน่วยข้อมูลที่ตรงกันในรายการ `-d` ถูกแปล ค่าที่ยอมรับได้สำหรับอ็อปชันนี้คือ ALPHA, HEX และ DEC ต้องมีการจับคู่ ชนิดที่แสดงในอาร์กิวเมนต์ `-x` สำหรับแต่ละในรายการ `-d`  
อ็อปชันนี้ ต้อง ถูกระบุถ้ามีอ็อปชัน `-d`
- y ระบุความยาวของหน่วยข้อมูล (เป็นไบนารี) ที่ระบุโดยอ็อปชัน `-d` ความยาวเหล่านี้ต้องเข้ากันกับฟิลด์ที่ตรงกัน ของเพิ่มเพลดการบันทึกข้อผิดพลาด AIX ที่ระบุในอ็อปชัน `-t` ต้องมีการจับคู่ชนิดที่แสดงในอาร์กิวเมนต์ `-y` สำหรับแต่ละในรายการ `-d`  
อ็อปชันนี้ ต้อง ถูกระบุถ้ามีอ็อปชัน `-d`

## พารามิเตอร์

### *file\_name*

ชื่อไฟล์ที่จะถูกค้นหา FFDC Failure Identifier สามารถระบุได้มากกว่าหนึ่งไฟล์ ถ้าไม่มีการระบุชื่อไฟล์ *fcfilter* จะอ่านจากอินพุตมาตรฐาน

## สถานะออก

**fclogerr** ส่งคืนโค้ดสถานะ *exit* ต่อไปนี้ เมื่อดำเนินการสำเร็จ:

- 0 ข้อมูลที่เข้าคิวสำเร็จเพื่อทำการเขียนไปที่บันทึกข้อผิดพลาด AIX และบันทึกระบบ BSD FFDC Failure Identifier สำหรับเร็กคอร์ด ถูกแสดงไปที่เอาต์พุตมาตรฐาน ส่วนที่เรียกควรรับข้อมูลเอาต์พุตมาตรฐาน เพื่อรับค่านี้
- 2 มีการข้อมูลวิธีใช้แสดงและการประมวลผลสิ้นสุด
- 12 ไม่มีการบันทึกข้อมูลกับบันทึกข้อผิดพลาด AIX และไม่มี FFDC Failure Identifier ถูกจัดเตรียมโดยคำสั่งนี้ ผู้ใช้คำสั่งจัดเตรียมอ็อปชันที่ไม่ถูกต้องกับ คำสั่งนี้

บนแพลตฟอร์ม AIX ที่ไม่ใช่ AIX, **fclogerr** ส่งคืนโค้ดสถานะ *exit* ต่อไปนี้เมื่อมีความล้มเหลวเกิดขึ้น:

- 38 เร็กคอร์ดไม่สามารถถูกสร้างในบันทึกระบบ BSD สำหรับเหตุการณ์นี้ได้ บันทึกระบบพบสถานะล้มเหลว บนระบบ AIX, รายงานถูกบันทึกไปที่บันทึกข้อผิดพลาด AIX ; บนระบบอื่น นี้ควร ถูกพิจารณาเป็นความล้มเหลว

เมื่อ **fclogerr** ถูกจัดเตรียมด้วยข้อมูลที่ไม่สมบูรณ์ จะมีการแทนที่ด้วยข้อมูลดีฟอลต์ สำหรับข้อมูลที่หายไปและพยายาม ที่จะสร้างเร็กคอร์ดใน FFDC Error Stack คำเตือนถูกสร้างในกรณีเหล่านี้ และขอความเตือนสร้างขึ้น นอกจากนี้การระบุอ็อปชัน *-q* ในกรณีที่มีการพบ สถานะคำเตือนมากกว่าหนึ่งรายการ คำสั่งจะส่งคืนโค้ดสถานะ *exit* สำหรับสถานะ ที่พิจารณาว่าร้ายแรงที่สุด โค้ดสถานะ *exit* ต่อไปนี้ถูกส่งกลับโดย **fclogerr** เมื่อสถานะ คำเตือนถูกตรวจพบ:

- 10 ผู้ใช้คำสั่งล้มเหลวเพื่อจัดเตรียมอ็อปชัน *-i* ให้กับคำสั่งนี้ หรือไฟล์ส่วนหัวที่มีชื่อตามอาร์กิวเมนต์ให้กับอ็อปชัน *-i* คนหาไม่พบ คำสั่งจะบันทึกข้อมูลทั่วไปไปที่บันทึกข้อผิดพลาด AIX ในกรณีนี้ การใช้ดีฟอลต์เพิ่มเพลต First Failure Data Capture (เลเบล FFDC\_DEF\_TPLT\_TR, โค้ด identifier 2B4F5CAB)
- 26 ทั้งสตริงข้อมูลรายละเอียดและไฟล์ข้อมูลรายละเอียดถูกจัดเตรียมให้กับ รูทีนนี้ รูทีนเลือกสตริงข้อมูลรายละเอียด และละเว้นไฟล์ข้อมูล รายละเอียด
- 28 ไม่มีชื่อของรีซอร์สที่ตรวจพบเหตุการณ์ ชื่อ รีซอร์สดีฟอลต์ *ffdc* ถูกใช้แทนที่ ชื่อรีซอร์สที่หายไป
- 29 ไม่มีการจัดเตรียมข้อมูลอย่างน้อยหนึ่งคอมโพเนนต์ของ ชื่อไฟล์โค้ด *information—source*, เวอร์ชันไฟล์ซอร์สโค้ด, ชื่อ LPP, บรรทัดของโค้ด *position—* ของแอสเพคเคชันที่ทำการตรวจจับ ข้อมูลดีฟอลต์ถูกใช้แทนที่ข้อมูลที่หายไป
- 32 ชื่อไฟล์ในพารามิเตอร์ *detail\_data\_file* ไม่สามารถถูกคัดลอกไปที่ไดเรกทอรี */var/adm/ffdc/dumps* รายการ FFDC Error Stack อาจอิงเวอร์ชันต้นฉบับของไฟล์นี้ อย่า ละทิ้งนำเนาต้นฉบับของไฟล์นี้
- 33 ไม่มีการระบุอ็อปชัน *-e* หรือไม่ได้ระบุ ชนิดเหตุการณ์ FFDC ที่ถูกต้อง ชนิดเหตุการณ์ FFDC\_DEBUG ได้ถูกกำหนด ให้กับเร็กคอร์ดเหตุการณ์นี้
- 34 ไม่มีการกำหนดข้อความในพารามิเตอร์ *format* ดังนั้น ข้อความทั่วไปถูกบันทึกลงในบันทึกระบบ BSD สำหรับ เหตุการณ์นี้
- 35 ไม่มีข้อมูลรายละเอียดถูกจัดเตรียมให้กับเหตุการณ์นี้ การวิเคราะห์ปัญหา ในภายหลังอาจเป็นไปได้ยาก โดยไม่มีรายละเอียดเหล่านี้ในการระบุ เหตุการณ์ที่เจาะจง

- 36 ความยาวของสตริงข้อมูลรายละเอียดยาวเกินกว่าความจุของ ค่าจำกัดรายการบันทึกข้อผิดพลาด AIX ข้อมูลรายละเอียดถูกตัดท้ายเพื่อให้พอดีกับพื้นที่ที่มี ข้อมูลบางส่วนในเหตุการณ์อาจสูญหายในการตัดข้อมูลนี้
- 37 FFDC Error Identifier ไม่สามารถถูกสร้างขึ้นสำหรับรายงานที่สร้าง โดยรูทีนนี้ FFDC Failure Identifier ไม่ถูกเขียนไปที่เอาต์พุตมาตรฐาน แต่ข้อมูลเกี่ยวกับเหตุการณ์ถูกบันทึกไปที่บันทึกข้อผิดพลาด AIX และบันทึกระบบ BSD
- 38 เร็กคอร์ดไม่สามารถถูกสร้างในบันทึกระบบ BSD สำหรับเหตุการณ์นี้ได้ บันทึกระบบอาจไม่ถูกเปิดใช้ หรืออาจพบปัญหา บนระบบ AIX, รายงานถูกบันทึกไปที่บันทึกข้อผิดพลาด AIX ; บนระบบอื่น นี้ควร ถูกพิจารณาเป็นความล้มเหลว

## ตัวอย่าง

สำหรับตัวอย่างนี้ สคริปต์ Korn Shell พยายามเข้าถึงข้อมูลคอนฟิกูเรชัน จากไฟล์ ถ้าการพยายามนี้ล้มเหลว โค้ดจะบันทึกความล้มเหลว ไปที่บันทึกข้อผิดพลาด AIX โดยใช้ซอร์สโค้ดเต็มเพลตต่อไปนี้:

```
*! mymesgcat.cat
+ SP_FFDCXMPL_ER:
  Comment      = "Configuration Failed - Exiting"
  Class        = S
  Log          = true
  Report       = true
  Alert        = false
  Err_Type     = PERM
  Err_Desc     = {3, 10, "CONFIGURATION FAILURE - EXITING"}
  Prob_Causes  = E89B
  User_Causes  = E811
  User_Actions = 1056
  Fail_Causes  = E906, E915, F072, 108E
  Fail_Actions = {5, 14, "VERIFY USER HAS CORRECT PERMISSIONS TO ACCESS FILE"},
                 {5, 15, "VERIFY CONFIGURATION FILE"}
  Detail_Data  = 46, 00A2, ALPHA
  Detail_Data  = 42, EB2B, ALPHA
  Detail_Data  = 42, 0030, ALPHA
  Detail_Data  = 16, EB00, ALPHA
  Detail_Data  = 16, 0027, ALPHA
  Detail_Data  = 4, 8183, DEC
  Detail_Data  = 4, 8015, DEC
  Detail_Data  = 60, 8172, ALPHA
```

นิยามนี้ให้ผลเต็มเพลตการบันทึกข้อผิดพลาด AIX ต่อไปนี้:

```
LABEL:          ERRID_SP_FFDCXMPL_ER
IDENTIFIER:     <calculated by errupdate during source code build>

Date/Time:     <filled in by AIX Error Log subsystem>
Sequence Number: <filled in by AIX Error Log subsystem>
Machine Id:    <filled in by AIX Error Log subsystem>
Node Id:      <filled in by AIX Error Log subsystem>
Class:        S
Type:         PERM
Resource Name: <filled in by -r option to fcllogerr>
```

Description  
CONFIGURATION FAILURE - EXITING

Probable Causes  
COULD NOT ACCESS CONFIGURATION FILE

User Causes  
USER CORRUPTED THE CONFIGRATION DATABASE OR METHOD

Recommended Actions  
RE-CREATE FILE

Failure Causes  
COULD NOT ACCESS CONFIGURATION FILE  
PERMISSIONS ERROR ACCESSING CONFIGURATION DATABASE  
FILE READ ERROR  
FILE IS CORRUPT

Recommended Actions  
VERIFY USER HAS CORRECT PERMISSIONS TO ACCESS FILE  
VERIFY CONFIGURATION FILE

Detail Data  
DETECTING MODULE  
<filled in by **fclogerr** options>  
ERROR ID  
<The FFDC Failure Identifier created by **fclogerr**>  
REFERENCE CODE  
<The **-a** option value to **fclogerr**>  
FILE NAME  
<Must be supplied as part of **-d** option list to **fclogerr**>  
FUNCTION  
<Must be supplied as part of **-d** option list to **fclogerr**>  
RETURN CODE<Must be supplied as part of **-d** option list to **fclogerr**>  
ERROR CODE AS DEFINED IN sys/errno.h  
<Must be supplied as part of **-d** option list to **fclogerr**>  
USER ID<Must be supplied as part of **-d** option list to **fclogerr**>

สาม Detail Data Fields แรกถูกสร้างโดยรูทีน **fclogerr** จากข้อมูลที่ส่งในพารามิเตอร์ Detail Data ที่เหลือต้องถูกระบุกับอ็อปชัน **-d** และชนิดของข้อมูลที่ระบุต้องถูกแสดงโดยอ็อปชัน **-x** เซกเมนต์ซอร์สโค้ดตัวอย่างด้านล่างแสดงวิธีการทำงาน และการที่ **fclogerr** ถูกเรียกเพื่อบันทึกข้อมูลในบันทึกข้อผิดพลาด AIX และบันทึกระบบ BSD

```
typeset CONFIG_FNAME
typeset INBUF
typeset MINUSDOPTS
typeset MINUSXOPTS
typeset MINUSYOPTS
typeset FID
integer MYCLIENT
integer RC
:
MYCLIENT=$$
```

```

CONFIG_FNAME="/configfile.bin"
exec 3< $CONFIG_FNAME
:
read -u3 INBUF
RC=$?
if ((RC != 0))
    ดั้งนั้น
    # Create Detail Data Memory Block for AIX Error Log Template
    # Need to know the EXACT structure of the Template to do this correctly.
    #   Field 1 - filled in by fc_log_error
    #   Field 2 - filled in by fc_log_error
    #   Field 3 - filled in by fc_log_error
    #   Field 4 - name of configuration file being used - 16 bytes
    #   Field 5 - name of function call that failed - 16 bytes
    #   Field 6 - return code from failing function - 4 byte integer
    #   Field 7 - errno from failing function call (unused) - 4 byte integer
    #   Field 8 - user ID using this software - remaining space (62 bytes)
    # This source code supplied fields 4 through 8 in the "-d" option, and
    # describes the data types for each in the "-x" option.
    MINUSDOPTS=$CONFIG_FNAME
    MINUSXOPTS="ALPHA"
    MINUSYOPTS="16"
    MINUSDOPTS="$MINUSDOPTS,read"
    MINUSXOPTS="$MINUSXOPTS,ALPHA"
    MINUSYOPTS="$MINUSYOPTS,16"
    MINUSDOPTS="$MINUSDOPTS,$RC"
    MINUSXOPTS="$MINUSXOPTS,DEC"
    MINUSYOPTS="$MINUSYOPTS,4"
    MINUSDOPTS="$MINUSDOPTS,0"
    MINUSXOPTS="$MINUSXOPTS,DEC"
    MINUSYOPTS="$MINUSYOPTS,4"
    MINUSDOPTS="$MINUSDOPTS,$MYCLIENT"
    MINUSXOPTS="$MINUSXOPTS,DEC"
    MINUSYOPTS="$MINUSYOPTS,60"
    FID=$(fclogerr -e FFDC_ERROR -t ERRID_SP_FFDCXEMPL_ER -i /usr/lpp/ssp/inc/
myprog.h -r myprog -s myprog.ksh -p $LINEPOS -v "1.1" -l PSSP -d $MINUSDOPTS -x
$MINUSXOPTS -y $MINUSYOPTS -b "myprog Configuration Failure - Exiting")
    RC=$?
    if ((RC == 0))
        ดั้งนั้น
            fcdispfid $FID
            return 1
    else
        :
    fi
fi

```

ตอนนี้ให้พิจารณาข้อมูลที่ต่างไปจากตัวอย่างด้านบนเล็กน้อย โดยใช้เพิ่มเพลตบันทึกข้อผิดพลาด AIX เดียวกัน แต่คราวนี้ใช้คำสั่งภายนอกเพื่อรับข้อมูลคอนฟิกูเรชันจากไฟล์ที่ซอร์สโค้ดนี้ระบุ คำสั่งจบการทำงาน ด้วยสถานะ exit ที่ไม่เป็นศูนย์และพิมพ์ FFDC Failure Identifier ไปที่เอาต์พุตมาตรฐาน ถ้าพบสภาวะความล้มเหลว และ เพื่อแสดงการใช้ข้อมูลประกาศในรายการไฟล์คอนฟิกูเรชัน **-d** จะมีช่องว่างอยู่ในชื่อ:

```

typeset CONFIG_FNAME
typeset INBUF
typeset MINUSDOPTS
typeset MINUSXOPTS
typeset MINUSYOPTS
typeset FID
typeset OUTPUT
integer MYCLIENT
integer RC
:
MYCLIENT=$$
CONFIG_FNAME="This is a test"
OUTPUT=$(configdabeast $CONFIG_FNAME)
RC=$?
if ((RC != 0))
    ดั้งนั้น
        # Create Detail Data Memory Block for AIX Error Log Template
        # Need to know the EXACT structure of the Template to do this correctly.
        # Field 1 - filled in by fc_log_error
        # Field 2 - filled in by fc_log_error
        # Field 3 - filled in by fc_log_error
        # Field 4 - name of configuration file being used - 16 bytes
        # Field 5 - name of function call that failed - 16 bytes
        # Field 6 - return code from failing function - 4 byte integer
        # Field 7 - errno from failing function call (unused) - 4 byte integer
        # Field 8 - user ID using this software - remaining space (62 bytes)
        # This source code supplied fields 4 through 8 in the "-d" option, and
        # describes the data types for each in the "-x" option.
        MINUSDOPTS="\\"$CONFIG_FNAME\\"
        MINUSXOPTS="ALPHA"
        MINUSYOPTS="16"
        MINUSDOPTS="$MINUSDOPTS,configdabeast"
        MINUSXOPTS="$MINUSXOPTS,ALPHA"
        MINUSYOPTS="$MINUSYOPTS,16"
        MINUSDOPTS="$MINUSDOPTS,$RC"
        MINUSXOPTS="$MINUSXOPTS,DEC"
        MINUSYOPTS="$MINUSYOPTS,4"
        MINUSDOPTS="$MINUSDOPTS,0"
        MINUSXOPTS="$MINUSXOPTS,DEC"
        MINUSYOPTS="$MINUSYOPTS,4"
        MINUSDOPTS="$MINUSDOPTS,$MYCLIENT"
        MINUSXOPTS="$MINUSXOPTS,DEC"
        MINUSYOPTS="$MINUSYOPTS,60"
        FID=$(fclogerr -e FFDC_ERROR -t ERRID_SP_FFDCXEMPL_ER -i /usr/lpp/ssp/inc/
myprog.h -r myprog -s myprog.ksh -p $LINEPOS -v "1.1" -l PSSP -d $MINUSDOPTS -x
$MINUSXOPTS -y $MINUSYOPTS -a $OUTPUT -b "myprog Configuration Failure - Exiting")
        RC=$?
        if ((RC == 0))
            ดั้งนั้น
                fcdispfid $FID
                return 1
            else
                :
            fi
        fi
fi

```

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fcreport” ในหน้า 521

ข้อมูลที่เกี่ยวข้อง:

ไฟล์ ct\_ffdc.h

---

## คำสั่ง fcpushstk

### วัตถุประสงค์

บันทึกข้อมูลเกี่ยวกับสถานะ ความล้มเหลวหรือสำคัญ ไปที่ First Failure Data Capture Error Stack

### ไวยากรณ์

```
/usr/sbin/rsct/bin/fcpushstk { [-a assoc_fid] -c message_catalog_name -m message_set -n message_number [-o message_param [,message_param,...]] -l lpp_name -p line_of_code_pos -r resource -s source_filename -v sidlevel { [-d detail_data] | [-f detail_data_file] } } default_message | -h
```

### คำอธิบาย

**fcpushstk** ถูกใช้โดยสคริปต์เพื่อบันทึกข้อมูลความล้มเหลว ไปที่ FFDC Error Stack สคริปต์บันทึกข้อมูลที่มีรายละเอียดและข้อมูลการดีบั๊ก ไปที่ FFDC Error Stack เพื่อใช้ในการกำหนดปัญหาในภายหลัง

FFDC Error Stack ถูกใช้เพื่อช่วยทำความเข้าใจสถานะล้มเหลวที่เกิดขึ้นเมื่อ กระบวนการหรือ thread หลายๆ หน่วยที่เกี่ยวข้อง ถูกดำเนินการร่วมกัน บนโหนดเพื่อทำงานทั่วไป อุปกรณ์นี้ใช้ได้ดีที่สุดกับแอปพลิเคชัน ที่สร้าง threads หรือกระบวนการย่อย ซึ่งอาจสร้าง threads หรือกระบวนการย่อย ขึ้นมาด้วยตัวเอง เมื่อต้องการใช้ FFDC Error Stack สคริปต์สร้าง *FFDC Error Stack Environment* โดยใช้ อินเตอร์เฟซ **fcinit** หลังจากที่สถานะแวดล้อมนี้ถูกสร้าง แอปพลิเคชันและส่วนที่สืบทอดสามารถใช้ FFDC Error Stack

ไม่ใช่ซอฟต์แวร์แอปพลิเคชันทั้งหมดที่จะสร้าง FFDC Error Stack Environment อย่างไรก็ตาม แอปพลิเคชันเหล่านี้อาจถูกร้องขอโดยแอปพลิเคชันหรือสคริปต์อื่น ที่สร้าง FFDC Error Stack Environments ในกรณีเหล่านี้ สคริปต์ หรือแอปพลิเคชันที่เรียกซอฟต์แวร์นี้ อาจต้องการจับข้อมูลความล้มเหลว จากซอฟต์แวร์นี้ เพื่อทำการวิเคราะห์พร้อมกับข้อมูลความล้มเหลวอื่น จากซอฟต์แวร์อื่น ที่เรียกเพื่อค้นหาความสัมพันธ์หรือรูปแบบ ใน ความล้มเหลว สำหรับเหตุผลนี้ ซอฟต์แวร์ที่โดยปกติจะไม่ใช้ FFDC Error Stack ภายใต้สภาวะการดำเนินการปกติ อย่างน้อยควรสนับสนุน การใช้ FFDC Error Stack เมื่อ ถูกใช้โดยไคลเอนต์ที่เรียกใช้ ซอฟต์แวร์ ซึ่งกระทำได้โดย *การสืบทอด* FFDC Error Stack Environment จากกระบวนการพาเรนต์ผ่านอินเตอร์เฟซ **fcinit**

**fcpushstk** บันทึกคำอธิบายและรายละเอียดเกี่ยวกับ สถานะสำคัญไปที่ FFDC Error Stack ถ้า *FFDC Error Stack Environment* ยังไม่ได้ถูกสร้างขึ้นโดยสคริปต์ โดย การสร้างหรือการสืบทอด, **fcpushstk** จะไม่บันทึก ข้อมูลใด และส่งกลับการควบคุมไปที่ ส่วนที่เรียก การดำเนินการนี้อนุญาต ให้สคริปต์รันในโหมด "silent" ปกติ เมื่อข้อมูลการดีบั๊กไม่ถูกร้องขอ แต่ยังอนุญาตให้สคริปต์สนับสนุนการใช้ FFDC Error Stack เมื่อมีการร้องขอข้อมูลการดีบั๊ก

สคริปต์ต้องการการเรียกที่ชัดเจนไปที่ `fcpushstk` เพื่อ บันทึกข้อมูลไปที่ `FFDC Error Stack` เมื่อ `FFDC Error Stack Environment` ถูกสร้างขึ้น เพียงการสร้างสถานะแวดล้อมนั้นไม่เพียงพอในการให้ผลลัพธ์ การบันทึกข้อมูลความล้มเหลว คำสั่ง `fclogerr` จะไม่สร้างการบันทึกใดไปที่ `FFDC Error Stack`

เพื่อรับประกันการระบุอย่างถูกต้องของสถานะและตำแหน่ง ซึ่ง พบข้อมูล, `fcpushstk` ควรถูกเรียก in-line ในโมดูลซอร์สโค้ดของสคริปต์ ที่เรียกทันทีที่ตรวจพบสถานะที่ต้องการ `fcpushstk` จะบันทึกชื่อซอร์สโค้ดและข้อมูลบรรทัดของโค้ด เพื่อช่วยเหลือในการระบุและหาตำแหน่งซอร์สโค้ดที่ พบสถานะความล้มเหลว `fcpushstk` สามารถถูกเรียก โดยรูทีนย่อยหรือรันไทม์ที่โหลดอัตโนมัติ เพื่อบันทึกข้อมูลนี้ ถ้าจำเป็น จะมีการจัดเตรียมข้อมูลตำแหน่งทั้งหมดและข้อมูลรายละเอียดความล้มเหลวที่จำเป็นจะถูกจัดเตรียมให้พร้อมสำหรับรูทีนภายนอกนี้ รูทีนการบันทึกภายนอก ต้องบันทึกตำแหน่งจริง ซึ่งเหตุการณ์ถูกตรวจพบ

ขนาดสูงสุด ของรายการ `FFDC Error Stack` ถูกกำหนดโดยนิยาม `FC_STACK_MAX` ในไฟล์ส่วนหัว `<rset/ct_ffdc.h>` `FC_STACK_MAX` กำหนดความยาวเป็นไบต์ ค่านี้ควรถูกใช้เฉพาะเป็นแนวทางเบื้องต้น เนื่องจากความยาวนี้รวมข้อมูลที่จะถูกใช้โดย `fcpushstk` เพื่อบันทึกข้อมูลไฟล์การตรวจหา ข้อมูลคำอธิบาย และข้อมูล `FFDC Failure Identifier` เรียกคอร์ดใดที่ยาวเกิน `FC_STACK_MAX` ไบต์จะถูกตัดท้าย เพื่อให้พอดีกับค่าจำกัด `FC_STACK_MAX`

## แฟล็ก

- a ระบุ `FFDC Failure Identifier` สำหรับสถานะความล้มเหลวที่รายงานโดย ซอฟต์แวร์ที่ใช้แอปพลิเคชันนี้ ซึ่งทำให้หรือมีผลให้สถานะ ถูกบันทึกในเวลานี้ identifier นี้ควรถูกส่งกลับไป แอปพลิเคชันนี้เป็นส่วนหนึ่งของการแสดงผลลัพธ์ของซอฟต์แวร์ ส่วนที่เรียกจัดเตรียม identifier นี้ที่นี้ เพื่อที่ `FFDC Error Stack` จะสามารถเชื่อมโยงรายงานความล้มเหลวที่กำลังสร้างขึ้นในเวลานี้ พร้อมกับรายการความล้มเหลวที่บันทึกไว้ก่อนหน้านี้ ซึ่งทำให้ตัวตรวจสอบปัญหาสามารถติดตามสาเหตุของความล้มเหลวจากอาการ ต่างๆ ในแอปพลิเคชันและส่วนอื่น ต่อสาเหตุหลักในซอฟต์แวร์อื่น ถ้าไม่มีความล้มเหลวของซอฟต์แวร์อื่น ที่เป็นต้นเหตุของสถานะนี้ หรือถ้าซอฟต์แวร์อื่นไม่ส่งกลับ `FFDC Failure Identifier` เป็นส่วนหนึ่งของข้อมูลผลลัพธ์, ไม่ควรมีอ็อปชัน `-a` จัดเตรียมไว้
- c ระบุชื่อของแค็ตตาล็อกข้อความ `XPG/4-compliant` ที่มี รายละเอียดของความล้มเหลวที่บันทึกไว้ ชื่อนี้สัมพันธ์กับไดเรกทอรี `/usr/lib/nls/msg/$LANG` ถ้าไม่พบแค็ตตาล็อกข้อความ `default_message` จะถูกแสดง เพื่ออธิบายความล้มเหลว โปรดสังเกตว่า `default_message` จะไม่ถูกแปลระหว่างโลแคล
- d สตริงอักขระที่จัดเตรียมข้อมูลรายละเอียดเกี่ยวกับสถานะ เหมือนกับแนวคิด `Detail Data` ที่ใช้โดย `AIX Error Log` ถ้ารายละเอียด ของข้อมูลยาวเกินไป รายละเอียดเหล่านี้สามารถถูกเขียนลงในไฟล์ และชื่อของไฟล์นั้นที่จัดเตรียมเป็นอาร์กิวเมนต์กับอ็อปชัน `-f` อ็อปชัน `-d` และ `-f` ไม่สามารถถูกระบุได้ในเวลาเดียวกัน ถ้าไม่มีอ็อปชัน `-d` หรือ `-f` หรือใช้ได้ สตริงอักขระ ไม่มีข้อมูลรายละเอียด จะถูกบันทึก
- f ระบุชื่อของไฟล์ที่มีรายละเอียดเกี่ยวกับสถานะ ที่ถูกรายงาน เหมือนกับแนวคิด `Detail Data` ที่ใช้โดย `AIX Error Log` อ็อปชันนี้ถูกใช้เมื่อรายละเอียดยาวเกินไปที่จะบันทึก ภายในตัว `FFDC Error Stack` เอง หรือเมื่อมีวิธีการที่สามารถวิเคราะห์ ข้อมูลรายละเอียด เนื้อหาของไฟล์นี้ถูกคัดลอกไปที่ไดเรกทอรี `/var/adm/ffdc/dumps` และตำแหน่งใหม่ถูกบันทึก เป็น `Detail Data` ใน `FFDC Error Stack` ถ้าไฟล์ที่มีรายละเอียดของ สถานะไม่มีอยู่ อย่าระบุอ็อปชันนี้ อ็อปชัน `-d` และ `-f` ไม่สามารถถูกระบุได้ในเวลาเดียวกัน
- h แสดงข้อความวิธีใช้ไปที่เอาต์พุตมาตรฐานและออก ไม่มีการประมวลผลอื่น ไม่ว่าจะมีการระบุอ็อปชันใด
- l ระบุตัวย่อของชื่อของโปรแกรมที่มีไลเซนส์ ซึ่งมีซอฟต์แวร์นี้มาให้ ค่านี้ถูกค่าและ และเซอร์วิสการสนับสนุนผลิตภัณฑ์ควรทราบว่าเป็นชื่อที่ยอมรับได้ สำหรับโปรแกรมที่มีไลเซนส์ (`AIX` เป็นต้น) ถ้าอ็อปชันนี้ไม่ถูก ระบุ หรือปรากฏว่าไม่ถูกต้อง สตริงอักขระ `PPS_PRODUCT` จะถูกใช้แทน

- m ระบุชื่อข้อความที่มีข้อความที่อธิบายความล้มเหลว ในไฟล์แค็ตตาล็อกข้อความ ถ้าไม่พบชื่อข้อความนี้ `default_message` จะถูกแสดงเพื่ออธิบายความล้มเหลว หมายเหตุไว้ว่า `default_message` จะไม่ถูกแปลเป็นโลแคลของผู้ใช้
- n ระบุหมายเลขข้อความที่อธิบายความล้มเหลวที่กำลังถูกบันทึก ถ้าไม่พบข้อความนี้ `default_message` จะถูกแสดงเพื่ออธิบายความล้มเหลว หมายเหตุไว้ว่า `default_message` จะไม่ถูกแปลเป็นโลแคลของผู้ใช้
- o ระบุรายการของพารามิเตอร์การแทนค่าภายในข้อความที่ระบุโดย อ็อปชัน `-n fcpushstk` สนับสนุน เฉพาะสตริงอักขระเป็นพารามิเตอร์การแทนค่า (`%s`) เนื่องจากสภาวะแวดล้อม การปฏิบัติการเซลล์ ถ้ามีพารามิเตอร์การแทนค่าหลายพารามิเตอร์ แต่ละพารามิเตอร์ต้องถูกแยกด้วยคอมมา (,) ถ้ามีพารามิเตอร์การแทนค่า ที่มีช่องว่างที่ฝังตัว ต้องถูกปิดในอัญประกาศ ("`"`)
- q ยับยั้งการสร้างข้อความคำเตือนจากคำสั่ง คำเตือน ถูกสร้างเมื่อคำสั่งต้องแทนที่ข้อมูลดีฟอลต์สำหรับข้อมูลที่หายไป หรือเมื่อคำสั่งไม่สามารถใช้ได้ในการคัดลอก `detail_data_file` ไปที่ไดเรกทอรี `/var/adm/ffdc/dumps`
- r ระบุชื่อคอมโพเนนต์ซอฟต์แวร์นี้เป็นชื่อเชิงสัญลักษณ์สำหรับ ซอฟต์แวร์ที่สร้างรายงาน และควรเป็นชื่อที่รู้จักต่อลูกค้า และเซอร์วิสการสนับสนุนแ็พพลิเคชัน
- p ระบุบรรทัดของตำแหน่งโค้ด ภายในโมดูลซอร์สโค้ดที่สภาวะ ถูกรายงาน ค่าที่จัดเตรียมต้องเป็นค่าจำนวนเต็ม ที่ถูกต้อง เพื่อให้มีการระบุและตำแหน่งของสภาวะ ที่ถูกต้องค่านี้ ควรใกล้เคียงกับบรรทัดของโค้ดที่ตรวจพบสภาวะ เทาที่เป็นไปได้ สคริปต์ Korn Shell สามารถใช้ค่าของ `$LINENO` ภาษาสคริปต์ที่ไม่มีตัวแปรจำนวนบรรทัดพิเศษ สามารถจัดเตรียมค่าเชิงสัญลักษณ์นี้ ซึ่งผู้พัฒนาสามารถใช้เพื่อค้นหาจุด ในซอร์สโค้ดซึ่ง `fcpushstk` กำลังถูกใช้ ถ้า อ็อปชันนี้ใช้ไม่ได้หรือไม่มี จะใช้ค่า 0
- s ระบุชื่อของไฟล์ต้นฉบับที่มีบรรทัดของโค้ดที่พบสภาวะที่ถูกรายงาน สำหรับสคริปต์ Korn and Borne Shell อาร์กิวเมนต์สำหรับอ็อปชันนี้ควรถูกเซตเป็น `$0`; สคริปต์ C Shell จะเซตอาร์กิวเมนต์นี้เป็น `{0}` ถ้าไม่มีอ็อปชันนี้หรือไม่ถูกต้อง สตริงอักขระ `unknown_file` จะถูกใช้
- v ระบุหมายเลขเวอร์ชัน SCCS ของโมดูลซอร์สโค้ดที่ตรวจพบ สภาวะที่ถูกบันทึก สำหรับซอร์สโค้ดภายใต้การควบคุม SCCS นี้ควร ถูกเซตเป็น "1.1" (อัญประกาศจำเป็นต้องมี) ถ้าไม่มีอ็อปชันนี้หรือไม่ถูกต้อง สตริงอักขระ `unknown` จะถูกใช้

## พารามิเตอร์

### `default_message`

ระบุข้อความดีฟอลต์ที่จะถูกใช้เป็นคำอธิบายของความล้มเหลว เมื่อข้อมูลไม่สามารถถูกดึงมาจากข้อมูลแค็ตตาล็อกข้อความ ที่ระบุผ่านอ็อปชัน `-c`, `-m` และ `-n` ถ้าสตริงนี้มีพารามิเตอร์การกำหนดตำแหน่ง พารามิเตอร์การกำหนดตำแหน่ง ทั้งหมดต้องถูกระเบเป็นสตริงอักขระ (`%s`) ข้อความควรถูกปิดอยู่ในอัญประกาศ ("`"`) ถ้ามีช่องว่าง ฝังอยู่ `fcpushstk` จำกัดความยาวรวมของ สตริงนี้ที่ 72 อักขระ

## สถานะออก

`fcpushstk` ส่งคืนโค๊ดสถานะ `exit` ต่อไปนี้ เมื่อดำเนินการสำเร็จ:

- 0 มี FFDC Error Stack Environment และข้อมูลความล้มเหลวถูกบันทึก สำเร็จอยู่ใน FFDC Error Stack FFDC Failure Identifier สำหรับเรีกคอร์ด ถูกแสดงไปที่เอาต์พุตมาตรฐาน ส่วนที่เรียกควรรับข้อมูลเอาต์พุตมาตรฐาน เพื่อรับค่านี้
- 2 มีการข้อมูลวิธีใช้แสดงและการประมวลผลสิ้นสุด

fcpushstk ส่งคืนโค้ดสถานะ exit ต่อไปนี้ เมื่อมีความล้มเหลวเกิดขึ้น:

- 11 ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack และไม่มี FFDC Failure Identifier ถูกจัดเตรียมโดยคำสั่งนี้ โคลเอ็นต์ที่ร้องขอเพื่อใช้อ็อปชัน ไม่สนับสนุนในรีลีสนี้ของซอฟต์แวร์ FFDC
- 12 ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack และไม่มี FFDC Failure Identifier ถูกจัดเตรียมโดยคำสั่งนี้ มีการให้ฟังก์ชันพารามิเตอร์ที่ไม่รู้จัก แก่อินเตอร์เฟส
- 15 FFDC Error Stack Environment ไม่มีอยู่ ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack ไม่มีการสร้าง FFDC Failure Identifier โดยคำสั่งนี้ นี่เป็นโค้ดส่งคืนปกติกับโคลเอ็นต์ FFDC เมื่อ FFDC Error Stack Environment ไม่มีอยู่ให้สืบทอดผ่าน fcinit
- 17 ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack และไม่มี FFDC Failure Identifier ถูกจัดเตรียมโดยคำสั่งนี้ FFDC Error Stack Environment เสียหายและควรพิจารณาว่าไม่สามารถใช้งานได้
- 19 ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack - โดเร็กทอรี FFDC Error Stack ไม่มีอยู่หรือไม่สามารถใช้งานได้ ไม่มีการจัดเตรียม FFDC Failure Identifier โดยคำสั่งนี้
- 20 ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack และไม่มี FFDC Failure Identifier ถูกจัดเตรียมโดยคำสั่งนี้ ไม่สามารถเข้าถึงไฟล์ FFDC Error Stack ไฟล์อาจถูกลบ หรือสิทธิในไฟล์หรือโดเร็กทอรี ถูกเปลี่ยนแปลงเพื่อห้ามการเข้าถึง FFDC Error Stack
- 22 ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack - ไฟล์ FFDC Error Stack ไม่สามารถถูกล็อกเพื่อการใช้งานเฉพาะอินเตอร์เฟสนี้ได้ ได้มีการพยายาม ล็อกไฟล์นี้อีกครั้ง แต่ความพยายามทั้งหมดล้มเหลว กระบวนการอื่น อาจล็อกไฟล์นี้ อยู่และรีลีสไม่สำเร็จ หรือกระบวนการอื่นอาจ หยุดทำงาน และป้องกันไม่ให้กระบวนการอื่นใช้ FFDC Error Stack ได้ ไม่มีการจัดเตรียม FFDC Failure Identifier โดยคำสั่งนี้
- 24 ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack และไม่มี FFDC Failure Identifier ถูกจัดเตรียมโดยคำสั่งนี้ ไฟล์ FFDC Error Stack เสียหาย โคลเอ็นต์ควรพิจารณาว่า FFDC Error Stack Environment ใช้ไม่ได้
- 25 ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack และไม่มี FFDC Failure Identifier ถูกจัดเตรียมโดยคำสั่งนี้ ชื่อไฟล์ FFDC Error Stack ถูกเซต เป็นชื่อโดเร็กทอรี FFDC Error Stack Environment ควรพิจารณาว่า เสียหายและไม่สามารถใช้งานได้
- 32 ดัมพ์ไฟล์ไม่สามารถถูกตัดลอกไปที่โดเร็กทอรี /var/adm/ffdc/dumps มีพื้นที่ไม่เพียงพอในระบบไฟล์ที่มีโดเร็กทอรี /var/adm/ffdc คำสั่ง fcclear ควรถูกใช้เพื่อลบ FFDC Error Stacks และดัมพ์ไฟล์ที่ไม่จำเป็น หรือผู้ดูแลระบบจำเป็น ต้องเพิ่มพื้นที่ให้กับระบบไฟล์ ไม่มีการจัดเตรียม FFDC Failure Identifier โดยคำสั่งนี้
- 40 ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack - ข้อมูลไม่สามารถ ถูกบันทึกใน FFDC Error Stack มีพื้นที่ไม่เพียงพอในระบบไฟล์ที่มีโดเร็กทอรี /var/adm/ffdc คำสั่ง fcclear ควรถูกใช้เพื่อลบ FFDC Error Stacks และดัมพ์ไฟล์ที่ไม่จำเป็น หรือผู้ดูแลระบบจำเป็น ต้องเพิ่มพื้นที่ให้กับระบบไฟล์ ไม่มีการจัดเตรียม FFDC Failure Identifier โดยคำสั่งนี้
- 41 ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack และไม่มี FFDC Failure Identifier ถูกจัดเตรียมโดยคำสั่งนี้ ความล้มเหลวเกิดขึ้นเมื่อทำการอ่านข้อมูลควบคุม จาก FFDC Error Stack หรือการเขียนข้อมูลสำคัญไปที่ FFDC Error Stack โคลเอ็นต์ควรสรุปว่ารายการไม่ได้ถูกบันทึก สำหรับเหตุการณ์นี้
- 99 ไม่มีการบันทึกข้อมูลกับ FFDC Error Stack และไม่มี FFDC Failure Identifier ถูกจัดเตรียมโดยคำสั่งนี้ มีความล้มเหลวภายในที่ที่ไม่คาดคิดเกิดขึ้น ในรูทีน fc\_push\_stack ปัญหานี้ อาจต้องการ การดูแลจากเซอร์วิสการสนับสนุนแอฟพลิเคชัน

เมื่อ `fcpushstk` ถูกจัดเตรียมด้วยข้อมูลที่ไม่สมบูรณ์ จะมีการแทนที่ด้วยข้อมูลดีฟอลต์ สำหรับข้อมูลที่หายไปและพยายามที่จะสร้างเร็กคอร์ดใน FFDC Error Stack ค่าเตือนถูกสร้างในกรณีเหล่านี้ และข้อความเตือนถูกแสดงกับอุปกรณ์ข้อผิดพลาดมาตรฐาน นอกจากนี้ มีการระบุข้อผิดพลาด -q ในกรณีที่มีการพบ สภาวะค่าเตือนมากกว่าหนึ่งรายการ คำสั่งจะสร้างโค้ดสถานะ exit ที่ตรงกับสถานะค่าเตือนร้ายแรงที่สุดที่ตรวจพบ โค้ดสถานะ exit ต่อไปนี้ถูกส่งกลับโดย `fcpushstk` เมื่อสภาวะ ค่าเตือนถูกตรวจพบ:

- 26 ทั้งสตริงข้อมูลรายละเอียดและไฟล์ข้อมูลรายละเอียดถูกจัดเตรียมให้กับ รูทีนนี้ รูทีนเลือกสตริงข้อมูลรายละเอียดและละเว้นไฟล์ข้อมูล รายละเอียด
- 28 ไม่มีชื่อของรีซอร์สที่ตรวจพบเหตุการณ์ ชื่อ รีซอร์สดีฟอลต์ถูกใช้แทนที่ ชื่อรีซอร์สที่หายไป
- 29 ไม่มีการจัดเตรียมข้อมูลอย่างน้อยหนึ่งคอมโพเนนต์ของ ชื่อไฟล์โค้ด information—source, เวอร์ชันไฟล์ซอร์สโค้ด, ชื่อ LPP, บรรทัดของโค้ด position—ของแอฟพลิเคชันที่ทำการตรวจจับ ข้อมูลดีฟอลต์ถูกใช้แทนที่ข้อมูลที่หายไป
- 30 ไม่มีข้อความดีฟอลต์เพื่ออธิบายลักษณะทั่วไปของเหตุการณ์ ถ้าไม่พบแค็ตตาล็อกข้อความ XPG/4 ที่มีข้อความรายละเอียด ไม่มีรายละเอียดสำหรับสภาวะที่จะถูกแสดงโดยคำสั่ง `fcstkprpt`
- 31 ไม่มีข้อความเพื่ออธิบายลักษณะทั่วไปของเหตุการณ์ หรือคอมโพเนนต์ ของชื่อไฟล์ XPG/4 information—catalog, หมายเลขเซ็ทข้อความ, หมายเลขข้อความ—ไม่ได้ถูกจัดเตรียม ไม่มีรายละเอียดสำหรับสภาวะนี้ที่สามารถถูกแสดงได้โดยคำสั่ง `fcstkprpt`
- 32 ชื่อไฟล์ในพารามิเตอร์ `detail_data_file` ไม่สามารถถูกตัดลอกไปที่ใดเรียกทอรี `/var/adm/ffdc/dumps` รายการ FFDC Error Stack อ้างอิงเวอร์ชันต้นฉบับของไฟล์นี้ อย่า ละทิ้งนำเอาต้นฉบับของไฟล์นี้
- 35 ไม่มีข้อมูลรายละเอียดถูกจัดเตรียมให้กับเหตุการณ์นี้ การวิเคราะห์ปัญหา ในภายหลังอาจเป็นไปได้ยาก โดยไม่มีรายละเอียดเหล่านี้ในการระบุ เหตุการณ์ที่เจาะจง
- 37 FFDC Failure Identifier ไม่สามารถถูกสร้างขึ้นสำหรับรายงานที่สร้าง โดยรูทีนนี้ ไม่มี FFDC Failure Identifier ที่จัดเตรียมโดยคำสั่งนี้ แต่ข้อมูล เกี่ยวกับเหตุการณ์ถูกบันทึกไปที่ FFDC Error Stack
- 44 ข้อมูลที่จัดเตรียมกับคำสั่งนี้จะทำให้เร็กคอร์ด FFDC Error Stack เกินกว่าข้อจำกัด `FC_STACK_MAX` เร็กคอร์ดถูกตัดท้าย เพื่อให้สามารถถูกบันทึกภายในข้อจำกัดของระบบได้ ข้อมูลสำคัญเกี่ยวกับ ความล้มเหลวอาจสูญหายไป ระหว่างกระบวนการตัดท้าย ตัดแปลงสคริปต์ เพื่อให้มีข้อมูลน้อยลง หรือให้บันทึกข้อมูลไปที่ไฟล์ข้อมูลรายละเอียด และส่งชื่อไฟล์ข้อมูลรายละเอียดไปที่คำสั่งนี้แทน

## ตัวอย่าง

เมื่อต้องการบันทึกข้อมูลเกี่ยวกับความล้มเหลวไปที่ FFDC Error Stack เมื่อ FFDC Environment ถูกสร้างหรือสืบทอดโดยกระบวนการ:

```
#!/bin/ksh
:
:
cp /tmp/workfile $FILENAME
RC=$?
if ((RC != 0))
then
    FFDCID=$(fcpushstk -c mymsg.cat -m2 -n10 -o$FILENAME -r myprog
    -d"cp exit status $RC - file being copied /tmp/workfile" -s$0
    -p$LINENO -v"1.1" -lPSSP "Cannot update configuration file %1$s")
```

```

if (($? == 0))
then
    fcdispfid $FFDCID
    return 1
fi
fi
:
:

```

เมื่อต้องการทำให้การบันทึกที่เหมือนกันจากภาษาสคริปต์ที่ไม่มี บรรทัดของตัวแปรโค้ด พร้อมใช้งาน:

```

#!/bin/bsh
:
:
CODESCTN=14          # Used to identify where in the script code we are
cp /tmp/workfile $FILENAME
RC=$?
if test $RC -ne 0
then
    FFDCID=`fcpushstk -c mymsg.cat -m2 -n10 -o$FILENAME -r myprog
            -d"cp exit status $RC - file being copied /tmp/workfile" -s$0
            -p$CODESCTN -v"1.1" -lPSSP "Cannot update configuration file %1$s"`
    if test $? -eq 0
    then
        fcdispfid $FFDCID
        return 1
    fi
fi
CODESECTION=15      # New code section begins - a different task starts
:
:

```

เมื่อต้องการบันทึกข้อมูลเกี่ยวกับสถานะความล้มเหลวที่ถูกเชื่อมโยงกับสถานะความล้มเหลวอื่นที่บันทึกก่อนหน้านี้ไปที่ FFDC Error Stack โดยแอปพลิเคชันที่ใช้ FFDC:

```

#!/bin/ksh
:
:
ASSOC_FID=$(/usr/lpp/ssp/bin/somecmd -a -b)
RC=$?if ((RC != 0))
then
    FFDCID=$(fcpushstk -a$ASSOC_FID -c mymsg.cat -m2 -n10 -o$FILENAME -r myprog
            -d"cp exit status $RC - file being copied /tmp/workfile" -s$0
            -p$LINENO -v"1.1" -lPSSP "Cannot update configuration file %1$s")
    if (($? == 0))
    then
        fcdispfid $FFDCID
        return 1
    fi
fi
:
:

```

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fcinit” ในหน้า 503

“คำสั่ง fcreport”

---

## คำสั่ง fcreport

### วัตถุประสงค์

ค้นหาและแสดงรายงานความล้มเหลวและความล้มเหลวใดๆ ที่สัมพันธ์กับความล้มเหลวนั้น

### ไวยากรณ์

```
/usr/sbin/rsct/bin/fcreport { [-a ] FFDC_Failure_ID } | -h
```

### คำอธิบาย

**fcreport** ถอดรหัส FFDC Failure Identifier และ จัดทำรายงานเกี่ยวกับความล้มเหลวที่ระบุ คำสั่งยังตรวจหา ว่ามีความล้มเหลวใดบ้างที่สัมพันธ์กับ FFDC Failure Identifier และหากพบ จะจัดทำรายงานของความล้มเหลวนั้น คำสั่งยังทำการตรวจสอบรายงาน ของความล้มเหลวแต่ละอย่างที่ค้นหาความล้มเหลวที่สัมพันธ์กัน และจัดทำรายการของ ความล้มเหลวที่สัมพันธ์นั้นจนกระทั่งตรงตามเงื่อนไขข้อใดข้อหนึ่งต่อไปนี้:

- ไม่พบความล้มเหลวที่สัมพันธ์ใดๆ อีก
- รายงานความล้มเหลวที่สัมพันธ์กับที่หาไม่พบ เงื่อนไขนี้อาจเกิดขึ้นเมื่อ รายงานความล้มเหลวที่สัมพันธ์กันนั้นอยู่บนรีโมต โหนดที่ไม่สามารถเข้าถึงได้ ในขณะนั้น หรือเรียกคอร์ตของความล้มเหลวที่ถูกลบออกจากโหนด ที่มีอยู่

การใช้คำสั่งนี้ ผู้ใช้สามารถจัดทำรายงานสำหรับทั้งรายการ ของความล้มเหลวที่ทำให้เกิดความล้มเหลวที่ระบุได้ **fcreport** ไม่สามารถค้นหา รายงานความล้มเหลวใดๆ ที่อาจเป็นสาเหตุจาก ความล้มเหลวเริ่มต้นที่คำสั่งกำหนด โดยสามารถจัดทำเฉพาะ รายงาน ความล้มเหลวที่เป็นสาเหตุของความล้มเหลวนั้นเท่านั้น

### แฟล็ก

- a แสดงข้อมูลทั้งหมดที่มีในรายงานความล้มเหลว ค่าดีฟอลต์ คือแสดงเน็ตเวิร์กแอดเดรสของโหนดที่รายงานความล้มเหลว นั้นถูกสร้างขึ้น การประทับเวลาบนรายงานความล้มเหลว และรายละเอียดของ เหตุการณ์ที่บันทึกในรายงานความล้มเหลว
- h แสดงข้อความวิธีใช้ไปที่เอาต์พุตมาตรฐานและออก ไม่มีการประมวลผลอื่น ไม่ว่าจะมีการระบุอ็อปชันใด

### พารามิเตอร์

*FFDC\_Failure\_ID*

ระบุ FFDC Failure Identifier ของความล้มเหลวเพื่อเริ่มต้น รายงาน **fcreport** จะพยายามหาข้อมูล ความล้มเหลวสำหรับความล้มเหลวนั้น รวมถึงความล้มเหลวใดๆ ที่รายงานแสดงให้เห็นว่าเป็นความล้มเหลวที่เชื่อมโยง หนึ่งใน FFDC Failure Identifier เท่านั้นที่ถูกกำหนด ให้กับคำสั่งนี้

## ความปลอดภัย

fcreport ใช้ rsh เพื่อจัดการรายงานความล้มเหลวที่อาจ มีอยู่บนรีโมตโหนด ผู้ใช้ต้องมีสิทธิ์ที่เพียงพอในการเรียกใช้งาน คำสั่ง rsh ไปยังรีโมตโหนดเหล่านี้ หากผู้ใช้ไม่มีสิทธิ์นี้ fcreport จะสามารถติดตามรายการของความล้มเหลวที่เกี่ยวข้อง เท่านั้น ฐานรายใดที่ยังมีอยู่บนโลคอลโหนด

## สถานะออก

fcreport สร้างโค้ดสถานะการออกค่าใดค่าหนึ่ง ต่อไปนี้เมื่อดำเนินการเสร็จเรียบร้อย:

- 0 รายงานความล้มเหลวที่ค้นหาและแสดงสำหรับ FFDC Failure Identifier จะถูกจัดทำ รายงานความล้มเหลวที่เกี่ยวข้องที่ค้นหาและแสดงอาจไม่มีแสดงหรือแสดงมากกว่าหนึ่งรายงาน เช่นกัน
- 2 มีการข้อมูลวิธีใช้แสดงและการประมวลผลสิ้นสุด
- 10 ไม่ได้ระบุข้อพจน์หรืออาร์กิวเมนต์ที่จำเป็น
- 11 FFDC Failure Identifier ที่จัดให้แก่คำสั่งนี้ถูกสร้างโดย ซอฟต์แวร์ FFDC ที่มีรหัสใหม่กว่า ซึ่งคำสั่งไม่สามารถแปลความหมายของ identifier ได้อย่างถูกต้อง
- 12 มีการระบุข้อพจน์ที่ไม่รู้จักให้แก่คำสั่งนี้
- 20 FFDC Failure Identifier อ้างถึงรายการใดๆ ที่ทำใน FFDC Error Stack บนระบบนี้ แต่ไม่สามารถเข้าถึงไฟล์ FFDC Error Stack ไฟล์ อาจถูกลบออก หรืออาจปรับเปลี่ยนสิทธิ์บนไฟล์ เพื่อป้องกันการเข้าถึงไฟล์
- 27 FFDC Failure Identifier ที่จัดให้แก่คำสั่งนี้เป็น identifier ที่ไม่ถูกต้อง

## ตัวอย่าง

พิจารณากรณีที่หลากหลาย กระบวนการได้ถูกสร้างขึ้นตามลำดับ พาเรนต์-ชายด์ต่อไปนี้:

```
          PID = 562
          .
          .
          PID = 785
          .
          .
          PID = 2024      PID = 1042
          .
          .
          PID = 981      PID = 5012
```

ในตัวอย่างนี้ กระบวนการ 785 สร้าง FFDC Failure Identifier .3Iv04ZVVfvp.wtY0xRXQ7..... แสดงส่งค่ากลับไปที่กระบวนการ 562 ในการจัดทำรายงานรายละเอียดสำหรับ FFDC Failure Identifier .3Iv04ZVVfvp.wtY0xRXQ7..... และความล้มเหลวก่อนหน้านี้ใดๆ ที่นำไปสู่ความล้มเหลวที่ระบุนี้:

```
$ fcreport -a .3Iv04ZVVfvp.wtY0xRXQ7.....
```

รายงานนี้จะมีรายละเอียดของ FFDC Failure Identifier ที่ระบุ รวมถึงความล้มเหลวใดๆ ในกระบวนการ 2024, 1042, 981 และ 5012 ที่อาจเป็นสาเหตุของความล้มเหลว รายงานจะไม่มี ความล้มเหลวใดๆ ของกระบวนการ 562 ที่อาจเป็นสาเหตุจากผลลัพธ์ของความล้มเหลวของกระบวนการ 785

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fcclear” ในหน้า 495

“คำสั่ง fclogerr” ในหน้า 507

“คำสั่ง fcstkrpt” ในหน้า 526

---

## คำสั่ง fcstat

### วัตถุประสงค์

แสดงสถิติที่รวบรวมโดยไดรเวอร์อุปกรณ์ Fibre Channel ที่ระบุ

### ไวยากรณ์

```
fcstat [-z [-d|-c] | -d|-e [-d|-c] | -c] Device_Name
```

### คำอธิบาย

คำสั่ง `fcstat` แสดงสถิติที่รวบรวมโดยไดรเวอร์อุปกรณ์ Fibre Channel ที่ระบุ คุณสามารถเลือกระบุให้แสดงสถิติของอุปกรณ์เฉพาะ นอกเหนือจากสถิติอุปกรณ์ทั่วไป ถ้าคุณไม่ระบุแฟล็ก คำสั่ง `fcstat` แสดง สถิติอุปกรณ์ทั่วไปเท่านั้น คำสั่ง `fcstat` รวบรวมค่าสถิติโดยใช้ไพรซีเดนต์ต่อไปนี้:

1. เปิดแค็ตตาล็อกข้อความของ `fcstat` และตรวจดูรายการ พารามิเตอร์
2. ฐานข้อมูล Accesses the Object Data Manager (ODM) สำหรับข้อมูล ที่เกี่ยวกับอะแดปเตอร์ที่เลือก
3. เข้าถึงฐานข้อมูล ODM เพื่อรับข้อมูลที่เกี่ยวข้องกับพอร์ตของอะแดปเตอร์ที่เลือก
4. เปิดและเข้าถึงสถิติอะแดปเตอร์
5. รีเซ็ตสถิติบางสถิติหากคุณระบุแฟล็ก `-z`
6. รายงานสถิติและออกจากการทำงาน

ถ้าระบุ `Device_Name` ที่ไม่ถูกต้อง คำสั่ง `fcstat` ส่งคืน ข้อความแสดงข้อผิดพลาดแจ้งว่าไม่พบอุปกรณ์ในฐานข้อมูล ODM

คำสั่ง `fcstat` ยังรายงานสถิติถ้า `Device_Name` ที่ระบุ ไม่ได้เชื่อมต่อกับเครือข่าย (นั่นคือ ลิงก์ไม่ทำงาน) โดยการเปิด อุปกรณ์ขึ้นในโหมดการวินิจฉัยโดยใช้แฟล็ก `-d` เมื่อลิงก์ ตาวนและอุปกรณ์ถูกเปิดในโหมดที่ไม่ใช่การวินิจฉัย คำสั่ง `fcstat` จะหน่วงเวลาในการสร้างเอาต์พุต คุณสามารถใช้แฟล็ก `-c` เพื่อขจัดการหน่วงเวลานี้ ถ้าอุปกรณ์เปิดอยู่แล้วและ คำสั่ง `fcstat` เริ่มทำงานด้วยแฟล็ก `-d` การดำเนินการ `open` บนอุปกรณ์ล้มเหลวโดยมีข้อผิดพลาด EACCESS

เมื่อคำสั่ง `fcstat` ไม่สามารถแยกสถิติ จาก `Device_Name` ที่ระบุได้ คำสั่งจะยังคงรายงานข้อมูล ที่แยกมาจากฐานข้อมูล ODM

## แฟล็ก

### ไอเท็ม

-c

-d

-e

-z

### คำอธิบาย

จัดการช่วงเวลาในการสร้างเอาต์พุตเมื่ออุปกรณ์ถูกเปิดในโหมดที่ไม่ใช่โหมดการวินิจฉัย และลิงก์ down

แสดงสถิติโดยการเปิดอะแดปเตอร์ในโหมดการวินิจฉัย

แสดงสถิติทั้งหมด ซึ่งมี สถิติอุปกรณ์เฉพาะ (สถิติไดรวเวอร์, สถิติลิงก์ และชนิด FC4)

รีเซ็ตบางสถิติกลับเป็น ค่าเริ่มต้น เฉพาะผู้ใช้ที่ได้รับสิทธิ์พิเศษเท่านั้นที่สามารถเรียกใช้แฟล็กนี้

## พารามิเตอร์

### ไอเท็ม

Device\_Name

### คำอธิบาย

ชื่อของอุปกรณ์ Fibre Channel ตัวอย่างเช่น fcs0

## ฟิลด์สถิติ

**หมายเหตุ:** บางอะแดปเตอร์อาจไม่สนับสนุนเฉพาะบางสถิติ ค่าของฟิลด์สถิติที่ไม่สนับสนุนเป็น 0 เสมอ พารามิเตอร์ทั้งหมดที่ทำการเครื่องหมายด้วยเครื่องหมายดอกจัน (\*) ถูกรีเซ็ตกลับเป็น ค่าเริ่มต้นเมื่อคุณใช้คำสั่ง `fcstat` กับ แฟล็ก `-z`

ฟิลด์สถิติที่แสดง ในเอาต์พุตของคำสั่ง `fcstat` และ คำอธิบายเป็นดังนี้:

ไอเท็ม	คำอธิบาย
ชนิดของอุปกรณ์	แสดงรายละเอียดของอะแดปเตอร์
Serial Number	แสดงหมายเลขลำดับจากอะแดปเตอร์
Option ROM Version	แสดงเวอร์ชันของ Options ROM บนอะแดปเตอร์
ZA	แสดงฟิลด์ ZA จาก VPD ของอะแดปเตอร์
Node WWN	แสดงชื่อที่แพร่หลายทั่วโลกของอะแดปเตอร์
Port FC ID	แสดง SCSI ID ของอะแดปเตอร์
Port Type	แสดงชนิดการเชื่อมต่อของอะแดปเตอร์
Port Speed	แสดงความเร็วของอะแดปเตอร์
Port WWN	แสดงชื่อที่แพร่หลายทั่วโลกของพอร์ต
Seconds Since Last Reset	แสดงจำนวนวินาทีตั้งแต่ที่รีเซ็ตสถิติครั้งล่าสุดบน อะแดปเตอร์
* Frames	แสดงผลจำนวนเฟรมที่ส่งและรับ
* Words	แสดงจำนวนเวิร์ดที่ส่งและรับ
* LIP Count	แสดงจำนวน LIP
* NOS Count	แสดงจำนวน NOS
Error Frames	แสดงผลจำนวนเฟรมที่มีข้อผิดพลาด
* Dumped Frames	แสดงเฟรมที่ถูกดัมพ์
Link Failure Count	แสดง Link Failure Count
Loss of Sync Count	แสดงจำนวนครั้งที่ Sync สูญหาย
Loss of Signal	แสดงจำนวนครั้งที่สัญญาณสูญหาย
Primitive Seq Protocol Err Count	แสดงจำนวนครั้งที่ลำดับเดิมมีข้อผิดพลาด
Invalid Tx Word Count	แสดงจำนวนการส่งที่ไม่ถูกต้องที่เกิดขึ้น
Invalid CRC Count	แสดงจำนวนข้อผิดพลาด CRC ที่เกิดขึ้น
FC SCSI Adapter Driver Information: No DMA Resource Count	แสดงจำนวนครั้งที่รีซอร์ส DMA ไม่พร้อมใช้งาน
FC SCSI Adapter Driver Information: No Adapter Elements Count	แสดงจำนวนครั้งที่ไม่มีอะแดปเตอร์อีลีเมนต์ พร้อมใช้งาน
FC SCSI Adapter Driver Information: No Command Resource Count	แสดงจำนวนครั้งที่ไม่มีรีซอร์สคำสั่ง พร้อมใช้งาน
* FC SCSI Traffic Statistics: Input Requests	แสดงจำนวนการร้องขออินพุต
* FC SCSI Traffic Statistics: Output Requests	แสดงจำนวนการร้องขอเอาต์พุต
* FC SCSI Traffic Statistics: Control Requests	แสดงจำนวนการร้องขอการควบคุม



Loss of Sync Count: 1  
Loss of Signal: 1  
Primitive Seq Protocol Err Count: 1  
Invalid Tx Word Count: 1  
Invalid CRC Count: 1

FC SCSI Adapter Driver Information  
No DMA Resource Count: 0  
No Adapter Elements Count: 0  
No Command Resource Count: 0

FC SCSI Traffic Statistics  
Input Requests: 16289  
Output Requests: 48930  
Control Requests: 11791  
Input Bytes: 128349517  
Output Bytes: 209883136

## Location

`/usr/sbin/fcstat`

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `fddistat`” ในหน้า 531

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `atmstat`

คำสั่ง `netstat`

คำสั่ง `tokstat`

---

## คำสั่ง `fcstkrpt`

### วัตถุประสงค์

แสดงเนื้อหาของไฟล์ FFDC Error Stack

### ไวยากรณ์

```
/usr/sbin/rsct/bin/fcstkrpt { [-a] [-p] | -r } { -f FFDC_Failure_Identifier [-i] | -s FFDC_Error_Stack_File_Name } } |  
[-h]
```

### คำอธิบาย

`fcstkrpt` อ่านไฟล์ FFDC Error Stack ที่มี และแสดงเนื้อหาของไฟล์ไปยังอุปกรณ์เอาต์พุตมาตรฐาน ไฟล์ FFDC Error Stack ถูกระบุโดยใช้ชื่อของไฟล์เอง หรือโดยใช้ FFDC Failure Identifier ที่อ้างอิงเรีกคอร์ดที่ระบุภายในไฟล์นั้น

ข้อมูลจาก FFDC Error Stack สามารถแสดงโดยใช้รูปแบบหนึ่งในสองรูปแบบ: โดย *เงื่อนไขความล้มเหลวที่เกี่ยวข้อง* (ดีฟอลต์) หรือโดย *ซอฟต์แวร์เลย์เออร์*

## แฟล็ก

- a ระบุว่าข้อมูลทั้งหมดถูกแสดงสำหรับรายการใน FFDC Error Stack การดำเนินการดีฟอลต์คือแสดงการประทับเวลาสำหรับเร็กคอร์ด และรายละเอียดของเหตุการณ์
- f ระบุ FFDC Failure Identifier เพื่อใช้กำหนดตำแหน่ง FFDC Error Stack `fcstkprt` ถอดรหัส FFDC Failure Identifier กำหนดตำแหน่ง FFDC Error Stack ที่สัมพันธ์กับ FFDC Failure Identifier และประมวลผล FFDC Error Stack หนึ่ง FFDC Failure Identifier เท่านั้นที่สามารถ ถูกระบุโดยใช้แฟล็กนี้
- h แสดงข้อความวิธีใช้ไปที่เอาต์พุตมาตรฐานและออก ไม่มีการดำเนินการ อื่นๆ ไม่ว่าจะระบุอ็อปชันใด
- i แสดงเฉพาะข้อมูลที่สัมพันธ์กับรายการความล้มเหลวที่ระบุ ที่กำหนดโดยแฟล็ก -f โดยค่าดีฟอลต์ เร็กคอร์ดทั้งหมดใน FFDC Error Stack ถูกแสดง
- p แสดงข้อมูลจาก FFDC Error Stack ตามการจัดเรียงกระบวนการ เอาต์พุตถูกจัดลำดับเพื่อแสดงถึงลำดับที่กระบวนการ ถูกสร้างขึ้น (ความสัมพันธ์กระบวนการแบบพาเรนต์-ชาวด์) โดยแสดงข้อมูลกระบวนการ ชาวด์เป็นอันดับแรก ตามด้วยข้อมูลกระบวนการพาเรนต์ มุมมองนี้ใช้เพื่อให้เกิดความเข้าใจในเหตุการณ์ที่เกิดขึ้นอันดับแรก และเหตุการณ์ที่เกิดขึ้นภายหลัง อันสืบเนื่องจากเหตุการณ์แรก
- r แสดงข้อมูลจาก FFDC Error Stack ตามความสัมพันธ์ของเหตุการณ์ เหตุการณ์จะถูกนำเสนอพร้อมกับเหตุการณ์ที่สัมพันธ์ มุมมองนี้ใช้เพื่อให้เกิดความเข้าใจว่าเหตุการณ์ใดที่เกิดขึ้นเนื่องจากการเกิดขึ้นของ เหตุการณ์อื่นๆ นี้คือค่าดีฟอลต์
- s ระบุชื่อของ FFDC Error Stack ที่จะตรวจสอบ ชื่อนี้ อาจเป็นชื่อพาสซัมบูร์นหรือสัมพันธ์ของ FFDC Error Stack ชื่อไฟล์ FFDC Error Stack หนึ่งชื่อเท่านั้นที่สามารถระบุโดยแฟล็กนี้ หากใช้ชื่อไฟล์ สัมพันธ์ จะถือว่าไฟล์อยู่ในไดเรกทอรี `/var/adm/ffdc/stacks` ของโหนดที่ไฟล์นั้นอยู่

## พารามิเตอร์

### *FFDC\_Failure\_ID*

ระบุ FFDC Failure Identifier ของความล้มเหลวเพื่อเริ่มต้น รายงาน `fcreport` จะพยายามหาข้อมูล ความล้มเหลว สำหรับความล้มเหลวนี้ รวมถึงความล้มเหลวใดๆ ที่รายงานแสดงให้เห็นว่าเป็นความล้มเหลวที่เชื่อมโยง หนึ่ง FFDC Failure Identifier เท่านั้นที่ถูกกำหนด ให้กับคำสั่งนี้

## ความปลอดภัย

`fcreport` ใช้ `rsh` เพื่อจัดหารายงานความล้มเหลวที่อาจ มีอยู่บนรีโมตโหนด ผู้ใช้ต้องมีสิทธิ์ที่เพียงพอในการเรียกใช้งาน คำสั่ง `rsh` ไปยังรีโมตโหนดเหล่านี้ หากผู้ใช้ไม่มีสิทธิ์นี้ `fcreport` จะสามารถติดตามรายการของความล้มเหลวที่เกี่ยวข้อง เท่านั้นตราบไต่ที่ยังมีอยู่บนโลคัลโหนด

## สถานะออก

`fcstkprt` ใช้โค้ดสถานะการออก เลขจำนวนเต็มต่อไปนีเมื่อดำเนินการเสร็จเรียบร้อย:

- 0 ไฟล์ FFDC Error Stack ถูกกำหนดตำแหน่งสำเร็จ และเนื้อหาถูกแสดงที่ อุปกรณ์เอาต์พุตมาตรฐาน
- 2 มีการขอวิธีใช้แสดงและการประมวลผลสิ้นสุด
- 12 มีการระบุอ็อปชันไม่ถูกต้อง

- 14 ไม่มีข้อมูลถูกเขียนไปยังอุปกรณ์เอาต์พุตมาตรฐาน อีพซัน -f ถูกใช้และอาร์กิวเมนต์ *FFDC Error Identifier* ไม่ถูกต้อง
- 20 ไม่มีข้อมูลถูกเขียนไปยังอุปกรณ์เอาต์พุตมาตรฐาน อีพซัน -s ถูกใช้และไม่พบอาร์กิวเมนต์ *FFDC Error Stack File*
- 27 ไม่มีข้อมูลถูกเขียนไปยังอุปกรณ์เอาต์พุตมาตรฐาน ผู้เรียกให้ *FFDC Failure Identifier* ที่ถูกต้อง แต่ไฟล์ที่อ้างอิงโดย *FFDC Failure Identifier* ไม่ได้ถูกบันทึกบนโทนดนี้ ใช้คำสั่ง *fcdecode* เพื่อค้นหาโทนดที่มี *FFDC Error Stack* นี้อยู่
- 81 ไม่มีข้อมูลถูกเขียนไปยังอุปกรณ์เอาต์พุตมาตรฐาน เกิดความล้มเหลว ขึ้นขณะเขียนข้อมูลไปยังเอาต์พุตมาตรฐาน แอ็พพลิเคชันจะสรุปว่าเอาต์พุตมาตรฐานไม่สามารถรับเอาต์พุต
- 85 ไม่มีข้อมูลถูกเขียนไปยังอุปกรณ์เอาต์พุตมาตรฐาน ผู้เรียกให้ *FFDC Failure Identifier* ที่ถูกต้อง แต่ไฟล์ที่อ้างอิงโดย *FFDC Failure Identifier* ไม่มีอยู่

## ตัวอย่าง

ในการจัดการรายงานอย่างย่อของข้อมูลที่เกิดขึ้นในไฟล์ *FFDC Error Stack /var/adm/ffdc/stacks/myprog.562.19981001143052:*

```
$ fcstkrpt -r -s myprog.562.19981001143052
```

ในการจัดการรายงานอย่างละเอียดของข้อมูลที่มีอยู่ใน *FFDC Error Stack* โดยที่ *FFDC Failure Identifier .3Iv04ZVVfvp.wtY0xRXQ7.....* ถูกบันทึก และแสดงข้อมูลนี้ตามลำดับพาเรนต์-ชายด์:

```
$ fcstkrpt -p -f .3Iv04ZVVfvp.wtY0xRXQ7.....
```

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ *Reliable Scalable Cluster Technology (RSCT)*

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง *fcclear*” ในหน้า 495

“คำสั่ง *fcpushstk*” ในหน้า 515

“คำสั่ง *fcreport*” ในหน้า 521

---

## คำสั่ง *fcteststk*

### วัตถุประสงค์

ทดสอบการมีอยู่ของสภาวะแวดล้อม *First Failure Data Capture Error Stack*

### ไวยากรณ์

```
/usr/sbin/rsct/bin/fcteststk [-q] | [-h]
```

## คำอธิบาย

`fccteststk` สามารถถูกเรียกใช้โดยแอฟพลิเคชันโปรแกรมใดๆ ที่ต้องการใช้ `FFDC Error Stack` เพื่อทดสอบว่าโปรแกรมอำนวยความสะดวกเหล่านี้ถูกเรียกทำงานหรือไม่ โดยการทำการทดสอบนี้ แอฟพลิเคชันสามารถแจ้งภาวะ ผลการทำงานของ การรวบรวมข้อมูลความล้มเหลวในกรณีที่ *สภาวะแวดล้อม FFDC* ยังไม่ถูกสร้างขึ้น โดยอินเทอร์เฟซนี้จัดให้มี สำหรับใช้โดยโลบ ารารัฐที่เป็นหลัก ซึ่งจะไม่ทราบที่ โคลเอ็นต์แอฟพลิเคชันของตนได้สร้างหรือสืบทอด *สภาวะแวดล้อม FFDC*

An *FFDC Error Stack Environment* ถูกสร้าง โดยกระบวนการเมื่อกระบวนการนั้นต้องการข้อมูลความล้มเหลวของกระบวนการ เธรดที่สร้าง และกระบวนการที่สร้างในภายหลังใดที่สร้างจะถูก บันทึกไว้ใน `FFDC Error Stack` *FFDC Error Stack Environment* ถูกสืบทอดโดยกระบวนการเมื่อกระบวนการนั้นต้องการบันทึกข้อมูล ความล้มเหลวไปยังไฟล์ `FFDC Error Stack` ต่อเมื่อมีหนึ่งในกระบวนการที่มีมาก่อนหน้านั้นได้ ร้องขอให้กระบวนการทำ ในกรณีอื่นๆ ทั้งหมด กระบวนการจะไม่ บันทึกข้อมูลความล้มเหลวไปยัง `FFDC Error Stack` กระบวนการจะใช้ `fcinit` เพื่อสร้างหรือสืบทอด `FFDC Error Stack Environment`

`FFDC Error Stack Environment` จะสงวนไฟล์ `FFDC Error Stack` เพื่อ ที่ข้อมูลความล้มเหลวจะถูกบันทึกในไฟล์ในไดเรกทอรี `/var/adm/ffdc/stacks` ไฟล์เหล่านี้ใช้รูปแบบการตั้งชื่อ `script_name.PID.date_and_time` โดยที่ `script_name` คือ ชื่อของสคริปต์เอง `PID` คือ identifier กระบวนการของสคริปต์ และ `date_and_time` คือ วันที่และเวลาเมื่อสคริปต์ถูกเรียกใช้งาน เมื่อใดที่สคริปต์นี้ หรือกระบวนการลูก ของสคริปต์นี้บันทึกข้อมูลความล้มเหลวไปยัง `FFDC Error Stack` ข้อมูลนั้นจะถูกบันทึกในไฟล์นี้ด้วย

แอฟพลิเคชันจะใช้อินเทอร์เฟซ `fcpushstk` เพื่อเรียกคอร์ต ข้อมูลความล้มเหลวไปยัง `FFDC Error Stack` อย่างไรก็ตาม แอฟพลิเคชันอาจจำเป็นต้องรวบรวมข้อมูลจากหลายๆ ที่ก่อนทำการบันทึก ข้อมูล และการจัดหาข้อมูลนี้อาจส่งผลต่อผลการทำงานโดยรวม ของแอฟพลิเคชัน แอฟพลิเคชันจะไม่จำเป็นต้องรวบรวมข้อมูลนี้หาก *FFDC Error Stack Environment* ไม่ถูกสร้างขึ้นหรือถูกสืบทอด ในการหลีกเลี่ยงผลกระทบต่อผลการดำเนินงานนี้ แอฟพลิเคชันสามารถเรียกใช้ `fccteststk` เพื่อพิจารณา *FFDC Error Stack Environment* มีอยู่หรือไม่ และถ้ามี ให้เริ่มทำการรวบรวมข้อมูล ความล้มเหลว หาก *FFDC Error Stack Environment* ไม่มีอยู่ แอฟพลิเคชันสามารถหลีกเลี่ยงการรวบรวมข้อมูลนี้

กระบวนการที่ใช้อินเทอร์เฟซ `fclogerr` `FFDC` สามารถ ใช้ `fclogerr` เมื่อมี *FFDC Environment* อยู่ไม่ว่า `FFDC Error Stack` จะถูกใช้งานโดย *FFDC Environment* อยู่หรือไม่ เมื่อใดที่ใช้ `fclogerr` ข้อมูล ความล้มเหลวจะถูกบันทึกใน `AIX Error Log` and the `BSD System Log` โดยไม่คำนึงว่า `FFDC Error Stack` ถูกสงวนไว้หรือไม่ แอฟพลิเคชันใดๆ ที่ บันทึกข้อมูลโดยใช้อินเทอร์เฟซ `fclogerr` ต้อง รวบรวมข้อมูลความล้มเหลว *เสมอ* และบันทึกข้อมูล โดยไม่คำนึงว่า `FFDC Error Stack` ถูกใช้งานอยู่หรือไม่

## แฟล็ก

ไอเท็ม	คำอธิบาย
-h	แสดงข้อความการใช้งานสำหรับ คำสั่งนี้ ไม่มีการประมวลผลใดๆ อีก
-q	ไม่แสดงเอาต์พุตจากคำสั่งนี้ ซึ่งอธิบายว่าสภาวะแวดล้อม <code>FFDC</code> ถูกสร้างขึ้นหรือไม่ ผู้ใช้ คำสั่งจำเป็นต้องทดสอบสถานะการออกจากการทำงานคำสั่งเพื่อกำหนดว่า สภาวะแวดล้อม <code>FFDC</code> ถูกสร้างขึ้นสำหรับกระบวนการนี้หรือไม่

## พารามิเตอร์

`FFDC_Failure_ID`

ระบุ FFDC Failure Identifier ของความล้มเหลวเพื่อเริ่มต้น รายงาน `fcreport` จะพยายามหาข้อมูล ความล้มเหลว สำหรับความล้มเหลวนี้ รวมถึงความล้มเหลวใดๆ ที่รายงานแสดงให้เห็นว่าเป็นความล้มเหลวที่เชื่อมโยง หนึ่ง FFDC Failure Identifier เท่านั้นที่ถูกกำหนดให้กับคำสั่งนี้

## ความปลอดภัย

`fcreport` ใช้ `rsh` เพื่อจัดหารายงานความล้มเหลวที่อาจ มีอยู่บนรีโมตโหนด ผู้ใช้ต้องมีสิทธิ์ที่เพียงพอในการเรียกใช้งาน คำสั่ง `rsh` ไปยังรีโมตโหนดเหล่านี้ หากผู้ใช้ไม่มีสิทธิ์นี้ `fcreport` จะสามารถติดตามรายการของความล้มเหลวที่เกี่ยวข้อง เท่านั้นตรวจ ใดที่ยังมีอยู่บนโหนด

## สถานะออก

- 0 FFDC Error Stack Environment มีอยู่
- 2 มีการข้อมูลวิธีใช้แสดงและการประมวลผลสิ้นสุด
- 12 ไม่มีการดำเนินการประมวลผล มีการระบุข้อผิดพลาดไม่ถูกต้อง
- 15 FFDC Error Stack Environment ไม่ถูกสร้างขึ้นหรือสืบทอดโดย โคลเอ็นต์ในจุดนี้ได้ทันเวลา
- 17 FFDC Error Stack Environment อาจเสียหาย และควรถูกพิจารณาว่า ไม่สามารถใช้งานได้

## ตัวอย่าง

ในการทดสอบว่า FFDC Error Stack Environment มีสำหรับแอปพลิเคชันหรือไม่:

```
fcteststk -q
if (($? == 0))
then
    # Collect failure information
    :
    :
    # Use fcpushstk to record failure info
    :
    :
fi
```

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `fcinit`” ในหน้า 503

“คำสั่ง `fcdecode`” ในหน้า 498

---

## คำสั่ง `fddistat`

### วัตถุประสงค์

แสดงไทรเวอร์อุปกรณ์ FDDI และสถิติอุปกรณ์

### ไวยากรณ์

```
fddistat [ -r -t ] Device_Name
```

### คำอธิบาย

คำสั่ง `fddistat` แสดงสถิติที่รวบรวมโดยไทรเวอร์อุปกรณ์ FDDI ที่ระบุได้ไม่ได้ระบุแฟล็ก เฉพาะสถิติของไทรเวอร์อุปกรณ์จะ  
ถูกแสดง คำสั่งนี้ยังถูกใช้เมื่อคำสั่ง `netstat` ถูกรันพร้อมกับแฟล็ก `-v` คำสั่ง `netstat` ไม่เรียกแฟล็กคำสั่ง `fddistat`

ถ้ามีการระบุ `Device_Name` ที่ไม่ถูกต้อง คำสั่ง `fddistat` สร้างข้อความแสดงความผิดพลาด เริ่มต้นว่าไม่สามารถเชื่อมต่อไปที่  
อุปกรณ์

### แฟล็ก

ไอเท็ม คำอธิบาย

`-r` รีเซ็ตสถิติกลับเป็นค่าตั้งต้น แฟล็กนี้ สามารถถูกเรียกโดยผู้ใช้ที่ได้รับ privilege

`-t` สลับการติดตามการตีบในบางไทรเวอร์อุปกรณ์

### พารามิเตอร์

ไอเท็ม

`Device_Name`

คำอธิบาย

ชื่อของอุปกรณ์ FDDI ตัวอย่าง `fddi0`

### ฟิลด์สถิติ

หมายเหตุ: บางอะแดปเตอร์อาจไม่นับสนุนสถิติที่ระบุ ค่าของฟิลด์ของสถิติที่ไม่ได้รับการสนับสนุนจะเป็น 0 เสมอ

ฟิลด์สถิติที่แสดงในเอาต์พุต ของคำสั่ง `fddistat` และรายละเอียด คือ:

#### ฟิลด์หัวข้อ

ไอเท็ม

เวลาที่ผ่านไป

คำอธิบาย

แสดงระยะเวลาจริง ที่ผ่านไปตั้งแต่ครั้งสุดท้ายที่ สถิติถูกรีเซ็ต เนื่องจากส่วนของสถิติอาจถูกรีเซ็ต โดยไทรเวอร์อุปกรณ์  
ระหว่างการแก้ไขข้อผิดพลาด เมื่อตรวจพบข้อผิดพลาด ฮาร์ดแวร์ โดยจะมีข้อมูลเวลาที่ผ่านไปอื่นที่แสดงในส่วนกลาง ของ  
เอาต์พุต เมื่อสถานการณ์นี้เกิดขึ้นเพื่อแสดงถึง เวลาที่ต่างกันระหว่างสถิติ

#### ฟิลด์สถิติการส่ง

## ไอเท็ม

Packets  
Bytes  
Interrupt

ข้อผิดพลาดการส่งข้อมูล

แพ็กเก็ตที่ถูกครอบ

แพ็กเก็ตสูงสุดสำหรับคิวการส่งผ่านซอฟต์แวร์

คิวไอเวอร์โพล์การส่งผ่านซอฟต์แวร์

Current S/W+H/W Transmit Queue Length

แพ็กเก็ตการบรอดคาสต์  
มัลติคาสต์แพ็กเก็ต

## คำอธิบาย

จำนวนของแพ็กเก็ตที่ส่งสำเร็จโดยอุปกรณ์  
จำนวนไบต์ที่ส่งสำเร็จโดยอุปกรณ์  
จำนวนของอินเทอร์รับต์การส่งที่ได้รับโดยไดรเวอร์จาก  
อะแด็ปเตอร์  
จำนวนของข้อผิดพลาดเอาต์พุตที่พบบนอุปกรณ์นี้เป็น  
ตัวบ่งชี้การส่งข้อมูลไม่สำเร็จเนื่องจากข้อผิดพลาดทาง  
ฮาร์ดแวร์/เน็ตเวิร์ก  
จำนวนของแพ็กเก็ตที่ยอมรับโดยไดรเวอร์อุปกรณ์สำหรับการ  
การส่งข้อมูลซึ่งไม่ได้ให้กับอุปกรณ์ (สำหรับเหตุผลใดๆ)  
จำนวนสูงสุดของแพ็กเก็ตขาออกที่คิวกับคิวการส่ง  
ซอฟต์แวร์  
จำนวนแพ็กเก็ตขาออกที่ไอเวอร์โพล์คิวการส่งข้อมูล  
ซอฟต์แวร์  
จำนวนของแพ็กเก็ตขาออกขาออก บนคิวการส่งข้อมูล หรือ  
คิวการส่งข้อมูลซอฟต์แวร์  
จำนวนของแพ็กเก็ตกระจายที่ถูกส่งโดยไม่มีข้อผิดพลาด  
จำนวนของแพ็กเก็ตมัลติคาสต์ที่ถูกส่งโดยไม่มีข้อผิดพลาด

## ฟิลต์สติการรับ

### ไอเท็ม

Packets  
Bytes  
อินเทอร์รับต์  
ข้อผิดพลาดของการรับ

แพ็กเก็ตที่ถูกครอบ

แพ็กเก็ตที่ไม่ดี

แพ็กเก็ตการบรอดคาสต์  
มัลติคาสต์แพ็กเก็ต

### คำอธิบาย

จำนวนแพ็กเก็ตที่ได้รับสำเร็จโดยอุปกรณ์  
จำนวนไบต์ที่ได้รับสำเร็จโดยอุปกรณ์  
จำนวนของอินเทอร์รับต์การรับที่ได้รับโดยไดรเวอร์จาก อะแด็ปเตอร์  
จำนวนของข้อผิดพลาดอินพุตที่พบบนอุปกรณ์นี้เป็น ตัวบ่งชี้การรับข้อมูลไม่สำเร็จเนื่องจากข้อผิดพลาดทาง  
ฮาร์ดแวร์/เน็ตเวิร์ก  
จำนวนแพ็กเก็ตที่ได้รับโดยไดรเวอร์อุปกรณ์จากอุปกรณ์ซึ่งไม่ถูกกำหนด (ไม่ว่าด้วยเหตุใด) ให้กับ demuxer  
จำนวนของแพ็กเก็ตไม่ถูกต้องที่ได้รับ (หรือบันทึก) โดยไดรเวอร์อุปกรณ์  
จำนวนของแพ็กเก็ตกระจายที่รับโดยไม่มีข้อผิดพลาด  
จำนวนของมัลติคาสต์แพ็กเก็ตที่รับโดยไม่มีข้อผิดพลาด

## ฟิลต์สติทั่วไป

### ไอเท็ม

No mbuf Errors

SMT Error Word  
SMT Event Word  
Connection Policy Violation  
Port Event  
Set Count  
Adapter Check Code  
Purged Frames  
ECM State Machine  
PCM State Machine: Port A  
PCM State Machine: Port B  
CFM State Machine: Port A  
CFM State Machine: Port B  
CF State Machine  
MAC CFM State Machine  
RMT State Machine

### คำอธิบาย

จำนวนครั้งที่ mbufs ไม่พร้อมใช้งานกับไดรเวอร์อุปกรณ์ซึ่งปกติเกิดขึ้นระหว่างการ  
รับข้อมูล เมื่อไดรเวอร์ต้องรับ บัฟเฟอร์ mbuf เพื่อประมวลผลแพ็กเก็ตขาเข้า ถ้าพูล  
mbuf สำหรับขนาดที่ร้องขอว่างเปล่า แพ็กเก็ตจะถูกทิ้ง คำสั่ง netstat -m สามารถ  
ถูกใช้เพื่อคำสั่ง  
สถานะข้อผิดพลาด SMT ของอะแด็ปเตอร์  
สถานะเหตุการณ์ SMT ของอะแด็ปเตอร์  
สถานะของการเชื่อมต่อไปยังวงแหวนของอะแด็ปเตอร์  
สถานะพอร์ตของอะแด็ปเตอร์  
ค่าจำนวนเซตปัจจุบัน  
สถานะการตรวจสอบอะแด็ปเตอร์ล่าสุดของอะแด็ปเตอร์  
รับเฟรมที่ดรอปโดยอะแด็ปเตอร์เนื่องจากไม่มี descriptors อยู่  
เครื่องแสดงสถานะ Entity Coordination Management  
Physical Connection Management สำหรับเครื่องแสดงสถานะ อะแด็ปเตอร์หลัก  
Physical Connection Management สำหรับเครื่องแสดงสถานะ อะแด็ปเตอร์รอง  
Configuration Management สำหรับเครื่องแสดงสถานะอะแด็ปเตอร์หลัก  
Configuration Management สำหรับเครื่องแสดงสถานะอะแด็ปเตอร์รอง  
เครื่องแสดงสถานะคอนฟิกูเรชันโดยรวม  
Configuration Management สำหรับเครื่องแสดงสถานะ MAC  
เครื่องแสดงสถานะ Ring Management

## ตัวอย่าง

เมื่อต้องการแสดงสถิติไดรเวอร์อุปกรณ์สำหรับ fddi0 ให้ป้อน:

```
fddistat fddi0
```

ซึ่งจะได้ผลลัพธ์ตามที่แสดงดังต่อไปนี้:

```
-----  
FDDI STATISTICS (fddi0) :  
Elapsed Time: 0 days 0 hours 1 minutes 3 seconds  
  
Transmit Statistics:                Receive Statistics:  
-----  
Packets: 100                       Packets: 100  
Bytes: 113800                      Bytes: 104700  
Interrupts: 100                   Interrupts: 100  
Transmit Errors: 0                 Receive Errors: 0  
Packets Dropped: 0                Packets Dropped: 0  
Max Packets on S/W Transmit Queue: 0 Bad Packets: 0  
S/W Transmit Queue Overflow: 0  
Current S/W+H/W Transmit Queue Length: 0  
  
Broadcast Packets: 0              Broadcast Packets: 0  
Multicast Packets: 0             Multicast Packets: 0  
  
General Statistics:  
-----  
No mbuf Errors: 0  
SMT Error Word: 00040080          SMT Event Word: 000004a0  
Connection Policy Violation: 0000 Port Event: 0000  
Set Count Hi: 0000               Set Count Lo: 0003  
Adapter Check Code: 0000         Purged Frames: 0  
  
ECM State Machine:      IN  
PCM State Machine Port A: CONNECT  
PCM State Machine Port B: ACTIVE  
CFM State Machine Port A: ISOLATED  
CFM State Machine Port B: CONCATENATED  
CF State Machine:      C_WRAP_B  
MAC CFM State Machine: PRIMARY  
RMT State Machine:     RING_OP  
  
Driver Flags: Up Broadcast Running  
              Simplex DualAttachStation
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง fcstat” ในหน้า 523

“คำสั่ง entstat” ในหน้า 403

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง atmstat

คำสั่ง netstat

## คำสั่ง fdformat

### วัตถุประสงค์

คำสั่ง fdformat พอร์ม็แตต ดิสเก็ต

### ไวยากรณ์

```
fdformat [ Device ] [ -h ]
```

### คำอธิบาย

**ข้อควรสนใจ:** การพอร์ม็แตต ดิสเก็ต หรืออ่าน/เขียนอ็อบติคัลดิสก์จะทำลายข้อมูลใดๆ ที่มีอยู่บน ดิสก์นั้น

คำสั่ง fdformat พอร์ม็แตต ดิสเก็ตในดิสเก็ตไดรฟ์ที่ระบุเป็นความจุต่่ายกเว้นระบุแฟล็ก -h ไว้

ดิสเก็ตเปล่าแผ่นใหม่ทั้งหมดต้องถูกพอร์ม็แตตก่อนจึงจะใช้งานได้

ก่อนทำการพอร์ม็แตตดิสเก็ต หรืออ็อบติคัลดิสก์ อ่าน/เขียน คำสั่ง fdformat จะพร้อมต์การตรวจสอบ ซึ่ง อนุญาตให้คุณสิ้นสุดการดำเนินการได้อย่างเรียบร้อย

### แฟล็ก

ไอเท็ม คำอธิบาย

-h กำหนดให้ทำการพอร์ม็แตตความจุสูง แฟล็กนี้ใช้กับคำสั่ง fdformat เท่านั้น

### พารามิเตอร์

ไอเท็ม คำอธิบาย

Device ระบุอุปกรณ์ที่มีดิสเก็ตที่จะพอร์ม็แตต ดีฟอลต์ คืออุปกรณ์ /dev/rfd0 สำหรับไดรฟ์ 0

### ตัวอย่าง

ในการกำหนดการพอร์ม็แตตของดิสเก็ตเมื่อ ใช้คำสั่ง fdformat ให้ป้อน:

```
fdformat -h
```

### ไฟล์

ไอเท็ม  
/usr/sbin/fdformat  
/dev/rfd\*  
/dev/fd\*  
/dev/romd\*  
/dev/omd\*

คำอธิบาย  
มีคำสั่ง fdformat  
ระบุพารามิเตอร์อุปกรณ์  
ระบุพารามิเตอร์อุปกรณ์  
ระบุพารามิเตอร์อุปกรณ์  
ระบุพารามิเตอร์อุปกรณ์

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง flcopy” ในหน้า 583

“คำสั่ง format” ในหน้า 595

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง fd

---

## คำสั่ง fdpr

ไอเท็ม  
-analyse\_asm\_csects  
-extra\_safe\_analysis  
  
-ignore\_info  
  
-align bytes  
  
-lr\_opt  
-bt\_csect\_anchor\_removal  
-dead\_code\_removal  
-selective\_inline  
-sid\_fac percent  
  
-inline\_small\_funcs size  
-inline\_hot\_funcs percent  
  
-inline  
-hco\_resched  
-dcbt\_opt  
-killed\_regs  
  
-tb  
  
-pc  
-pp  
-RD  
-dpnf factor  
  
-dpht threshold  
  
-build\_dcg  
  
-tocload

### คำอธิบาย

วิเคราะห์ csects ที่เขียนในแอสเซมบลี (เมื่อใช้ ต้องระบุในเฟสทั้ง -1 และ -3)  
พยายามวิเคราะห์ csects ที่ไม่ใช่แบบธรรมดาที่มีแอสเซมบลีโค้ดที่เขียนด้วยมือ (เมื่อใช้ ต้องถูก  
ระบุที่เฟส -1 และ -3)  
ข้ามส่วน .info ที่สร้างด้วยอ็อพชัน -qfdpr ระหว่าง เวลาคอมไพล์ (เมื่อใช้ ต้องระบุในทั้งเฟส -1 และ -3  
)  
จัดเรียงโค้ดเรียกใช้งาน ตามจำนวนไบต์ที่กำหนด เพื่อปรับปรุงอัตราประสิทธิภาพการเตรียมโค้ดล่วงหน้า  
หากข้ามอ็อพชันนี้ คำสั่ง fdpr จะจัดเรียง โค้ดด้วยตัวแปรจำนวนไบต์ดีฟอลต์  
การจัดการเก็บและการเรียกคืน ลิงก์รีจิสเตอร์ในโปรซีเตอร์ที่เรียกใช้งานบ่อย  
กำจัดกาไรลด์คำสั่ง ที่สัมพันธ์กับการโยงของตารางสาขาในโค้ด  
ลบโค้ดที่ไม่สามารถเข้าถึงได้  
ดำเนินการอินไลน์แบบเลือก สำหรับฟังก์ชันที่ถูกเรียกใช้งานใช้โค้ดที่โดดเด่นใช้โค้ดเดียว  
เซตเปอร์เซ็นต์ปัจจัยการโดดเด่น สำหรับการออปติไมซ์อินไลน์แบบเลือก ช่วงที่อนุญาตอยู่ระหว่าง 50 -  
100 (ใช้ได้กับแฟล็ก -selective\_inline เท่านั้น)  
อินไลน์ฟังก์ชันทั้งหมดที่มีขนาดเล็กกว่าหรือเท่ากับขนาดที่กำหนดเป็นไบต์  
อินไลน์ฟังก์ชันทั้งหมดที่มีความถี่การเรียกใช้งานเท่ากับหรือมากกว่าเปอร์เซ็นต์ที่กำหนด ช่วง อินพุต  
อินพุตคือระหว่าง 0 - 100  
ดำเนินการ -inline\_small\_funcs 12 ด้วย -selective\_inline  
ย้ายที่คำสั่งกาไรลด์ โค้ดที่เรียกใช้งานบ่อยไปยังพื้นที่โค้ดที่เรียกใช้งานน้อย เมื่อทำได  
แทรกคำสั่ง dcbt เพื่อปรับปรุงผลการทำงานการแคชข้อมูล  
การจัดการเก็บและการเรียกคืน รีจิสเตอร์ที่ถูก คิล (เขียนทับ) หลังจากการเรียกใช้ ฟังก์ชันที่เรียกใช้งาน  
บ่อย  
บังคับให้มีการปรับโครงสร้างตาราง การติดตามย้อนกลับในโค้ดที่เรียงลำดับใหม่ หากข้ามอ็อพชัน -tb  
ตารางการติดตามย้อนกลับจะถูกเรียกคืนโดยอัตโนมัติสำหรับแอฟพลีเคชัน C++ โดยใช้กลไก Try &  
Catch  
สแกนขอบเขตของ csects ใน โค้ดที่เรียงลำดับใหม่  
สแกนขอบเขตของฟังก์ชันใน โค้ดที่เรียงลำดับใหม่  
ดำเนินการเรียงลำดับข้อมูลสแตตติกใหม่  
Data Placement Normalization Factor ระหว่าง 0 - 1 โดยที่ 0 ทำให้ตัวแปรสแตตติกถูกเรียงลำดับใหม่  
ไม่ว่าจะมีขนาดเท่าใด ในขณะที่ 1 จะค้นหาเฉพาะตัวแปรขนาดเล็กเป็นอันดับแรก (ใช้กับแฟล็ก -RD  
เท่านั้น)  
Data Placement Hotness Threshold ระหว่าง 0 - 1 โดยที่ 0 เรียงลำดับตัวแปรสแตตติกในกลุ่มขนาดใหญ่  
ใหม่อิงตาม โพล์การควบคุม ในขณะที่ 1 จะเรียงลำดับตัวแปรใหม่ในกลุ่มขนาดเล็กมาก อิงตาม  
ความถี่ในการเข้าถึง (ใช้กับแฟล็ก -RD เท่านั้น)  
สร้าง DCG (Data Connectivity Graph) สำหรับการเรียงลำดับข้อมูลที่ขยายเพิ่มเติม (ใช้กับแฟล็ก  
-RD เท่านั้น)  
ดำเนินการออปติไมซ์ tocload

ไอเท็ม	คำอธิบาย
<code>-reduce_toc_removal_factor</code>	ดำเนินการลบรายการ TOC ออกตามค่าปัจจัยการลบระหว่าง 0 - 1 โดยที่ 0 ลบเฉพาะรายการ TOC ที่ไม่มีการเข้าถึงเท่านั้น และ 1 ลบรายการ TOC ที่ไม่มีการเอ็กพอร์ตทั้งหมด
<code>-strip</code>	แบ่งเอาต์พุตไฟล์ (หากถูกสร้างขึ้น)
<code>-ptrgl_opt</code>	ดำเนินการถอดรหัสคำสั่งการเรียกใช้ทางอ้อมด้วยวิธีการรีจิสเตอร์โดยการแทนที่ด้วยการข้ามโดยตรง
<code>-no_ptrgl_r11</code>	อย่าดำเนินการลบคำสั่งการโหลด R11 ใน <code>_ptrgl csect</code> (การถอดรหัส <code>-ptrgl_r11</code> ถูกใช้โดยค่าดีฟอลต์)
<code>-O</code>	ดำเนินการเรียงลำดับโค้ดใหม่ด้วย การตั้งค่าคิดการคาดการณ์สาขา, การตัดสาขา และการลบคำสั่ง NOOP แฟล็ก <code>-O</code> ถูกนำไปเป็นค่าดีฟอลต์
<code>-O2</code>	สลับใช้แฟล็กการถอดรหัสที่มีความมั่นใจน้อยทั้งหมด
<code>-O3</code>	สลับใช้แฟล็กการถอดรหัสที่มีความมั่นใจทั้งหมด
<code>-O4</code>	สลับใช้แฟล็กการถอดรหัสที่มีความมั่นใจทั้งหมด

## Purpose

ยูทิลิตี้การปรับผลการทำงานเพื่อปรับปรุงเวลาการทำงาน และการใช้งานหน่วยความจำจริงของแอปพลิเคชันโปรแกรม `post-link` ระดับผู้ใช้

## Syntax

การใช้งานส่วนใหญ่:

```
fdpr -p ProgramFile -x WorkloadCommand
```

การใช้งานโดยละเอียด:

```
fdpr -p ProgramFile [-M SegNum ] [-fd Fdesc ] [-o OutputFile ] [-armember ArchiveMemberList ] [ OptimizationFlags ] [-map ] [-disasm ] [-disasm_data ] [-disasm_bss ] [-profcoun ] [-quiet] [-v ] [-1 | -2 | -3 | -12 | -23 | -123 ] [-x WorkloadCommand ]
```

การถอดรหัสแฟล็ก

```
[-tb ] [-pc ] [-pp ] [-O ] [-O2 ] [-O3 ] [-O4 ] [-selective_inline ] [-sid_fac percent ] [-inline_small_funcs size ] [-inline_hot_funcs percent ] [-hco_resched ] [-killed_regs ] [-lr_opt ] [-align bytes ] [-RD ] [-dnpf factor ] [-dpht threshold ] [-build_dcg ] [-tocload ] [-ptrgl_opt ] [-no_ptrgl_r11 ] [-dcbt_opt ] [-ignore_info ] [-dead_code_removal ] [-bt_csect_anchor_removal ] [-strip ] [-analyse_asm_csects ] [-extra_safe_analysis ] [-inline ] [-reduce_toc_removal_factor ]
```

## รายละเอียด

คำสั่ง `fdpr` (Feedback Directed Program Restructuring) คือยูทิลิตี้การปรับผลการทำงานที่อาจช่วยปรับปรุงเวลาการทำงาน และการใช้งานหน่วยความจำจริงของแอปพลิเคชันโปรแกรมระดับผู้ใช้ โปรแกรม `fdpr` ถอดรหัสเวลาการทำงานของโปรแกรม โดยการรวบรวมข้อมูลเกี่ยวกับลักษณะการทำงานของโปรแกรมขณะที่โปรแกรม ถูกใช้สำหรับเวิร์กโหลดบางประเภท และจากนั้นสร้างโปรแกรมเวอร์ชันใหม่ ที่ได้รับการถอดรหัสสำหรับเวิร์กโหลดนั้น โปรแกรมใหม่ที่สร้างโดย `fdpr` โดยปกติจะรันได้เร็วกว่าและใช้หน่วยความจำจริงน้อยกว่า

**ข้อควรสนใจ:** คำสั่ง `fdpr` ใช้เทคนิคการออปติไมซ์ระดับสูง กับโปรแกรมซึ่งอาจส่งผลให้โปรแกรมนั้นไม่ทำงานตามที่ คาดโปรแกรมซึ่งได้รับการออปติไมซ์โดยใช้เครื่องมือนี้ควรถูกใช้ ด้วยความระมัดระวัง และควรได้รับการทดสอบอย่างเข้มงวด อย่างน้อยกับชุดทดสอบเดียวกันกับ ที่ทดสอบโปรแกรมต้นฉบับเพื่อตรวจสอบว่ามีการทำงานตามที่คาด โปรแกรมที่ออปติไมซ์ได้รับการสนับสนุน

คำสั่ง `fdpr` สร้างโปรแกรมเรียกทำงานที่ถูกออปติไมซ์ใน 3 เฟสต่างกัน:

- เฟส 1 (แฟล็ก -1): สร้างโปรแกรมเรียกทำงาน ที่เป็นเครื่องมือและเพิ่มเพลตโปรไฟล์ไฟล์ว่าง
- เฟส 2 (แฟล็ก -2): รันโปรแกรมที่เป็นเครื่องมือ และอัปเดตข้อมูลโปรไฟล์
- เฟส 3 (แฟล็ก -3): สร้างโปรแกรมไฟล์เรียกทำงาน ที่ออปติไมซ์

เฟสเหล่านี้สามารถรันแยกกัน หรือในแบบรวมกันบางส่วนหรือรวมกันทั้งหมด แต่ต้องรันตามลำดับ (เช่น -1 จากนั้น -2 จากนั้น -3 หรือ -1 2 จากนั้น -3) ค่าดีฟอลต์คือรันทั้งสามเฟส

**หมายเหตุ:** โปรแกรมที่ที่รันได้ที่เป็นเครื่องมือซึ่งสร้างในเฟส 1 และรันในเฟส 2 โดยปกติ มีการรันหลายครั้งที่ช้ากว่าโปรแกรมต้นฉบับ เนื่องจากโปรแกรมที่เป็นเครื่องมือ ต้องการใช้เวลากระทำการเพิ่มสูงขึ้น โปรแกรมที่รันได้ควร ถูกเรียกใช้ ด้วยวิธีที่จะลดระยะเวลาการทำงานให้น้อยที่สุด ขณะที่ยังคง ดำเนินการส่วนโค้ดที่ต้องการได้อย่างเต็มประสิทธิภาพ ผู้ใช้คำสั่ง `fdpr` ควรพยายามลด ที่จะทำได้ เวลาใดๆ ที่ขึ้นกับ ลักษณะของโปรแกรม

## แฟล็ก

<b>ไอเท็ม</b>	<b>คำอธิบาย</b>
-1,-2,-3	ระบุเฟสที่จะรัน ค่าดีฟอลต์คือทั้ง 3 เฟส (-123) แฟล็ก -s ต้องถูกใช้เมื่อรันเฟสแยก เพื่อให้เฟสในภายหลังสามารถเข้าถึงไฟล์ระหว่างการดำเนินการที่จำเป็นได้ เฟสต้องรันตามลำดับ (ตัวอย่างเช่น -1 จากนั้น -2 จากนั้น -3 หรือ -1 จากนั้น -23) แฟล็ก -2 ต้องใช้ควบคู่กับแฟล็กการร้องขอ -x
-M SegNum	ระบุตำแหน่งที่จะแมปหน่วยความจำที่แบ่งใช้สำหรับการทำโปรไฟล์ ค่าดีฟอลต์ คือ 0x30000000 ระบุแอดเดรสหน่วยความจำที่แบ่งใช้ ทางเลือกหากโปรแกรมจะถูกออปติไมซ์ หรือสตริงคำสั่งเวิร์กโหลดใดๆ ที่เรียกใช้ด้วยแฟล็ก -x ใช้แอดเดรสหน่วยความจำที่แบ่งใช้ที่มีความขัดแย้ง โดยปกติค่าทางเลือกได้แก่ 0x40000000, 0x50000000, ... จนถึง 0xC0000000)
-fd Fdesc	ระบุหมายเลข descriptor ไฟล์ใดจะถูกใช้สำหรับไฟล์ profile ที่ถูกแมปกับพื้นที่หน่วยความจำที่แบ่งใช้ด้านบน ค่าดีฟอลต์ของ Fdesc ถูกเซตเป็น 1999
-o OutFile	ระบุชื่อของเอาต์พุตไฟล์จากตัวออปติไมซ์ ค่าดีฟอลต์ คือ <code>program.fdpr</code>
-p ProgramFile	มีชื่อของโปรแกรมไฟล์เรียกทำงานหรืออ็อบเจกต์ไฟล์ที่แบ่งใช้ หรือไลบรารีที่แบ่งใช้ที่มีอ็อบเจกต์/โปรแกรมที่รันได้ที่แบ่งใช้ เพื่อออปติไมซ์ โปรแกรมนี้ ต้องเป็นโปรแกรมที่รันได้แบบ unstripped
-armember	แสดงรายการสมาชิกที่เก็บถาวรที่จะถูกออปติไมซ์ ภายในไฟล์เก็บถาวรที่แบ่งใช้ ที่ระบุโดยแฟล็ก -p หากไม่ระบุ
ArchiveMemberList	-armember สมาชิกทั้งหมดของไฟล์เก็บถาวรจะถูกออปติไมซ์
-map	พิมพ์แมปของบล็อกพื้นฐาน และตัวแปรสตริง ที่มีแอดเดรสเก่า -> ใหม่ตามลำดับไปยังไฟล์ที่มีส่วนเติมท้าย <code>_mapper</code>
-disasm	พิมพ์ส่วนข้อมูลที่ถูกลอตแยกของเอาต์พุตที่ถูกออปติไมซ์ และโปรแกรมที่เป็นเครื่องมือไปยังไฟล์ที่ต่อท้ายด้วย <code>.dis_text</code>
-disasm_data	พิมพ์ส่วนข้อมูลที่ถูกลอตแยกของเอาต์พุตที่ถูกออปติไมซ์ และโปรแกรมที่เป็นเครื่องมือไปยังไฟล์ที่ต่อท้ายด้วย <code>.dis_data</code>
-disasm_bss	พิมพ์ส่วน bss ที่ถูกลอตแยกของเอาต์พุตที่ถูกออปติไมซ์ และโปรแกรมที่เป็นเครื่องมือไปยังไฟล์ที่ต่อท้ายด้วย <code>.dis_bss</code>
-profcount	พิมพ์ตัวนับการทำโปรไฟล์ไปยังไฟล์ที่ลงท้ายด้วย <code>.ncounts</code>
-quiet	โหมตเอาต์พุตแบบเงียบ
-v	เอาต์พุตแบบละเอียด
-x WorkloadCommand	ระบุคำสั่งที่ใช้สำหรับการเรียกใช้โปรแกรม ที่เป็นเครื่องมือ อาร์กิวเมนต์ทั้งหมดหลังแฟล็ก -x ถูกใช้สำหรับการร้องขอ ดังนั้น แฟล็ก -x ต้องปรากฏ เป็นอันดับสุดท้ายในบรรทัดคำสั่ง แฟล็ก -x จำเป็นต้องใช้เมื่อ ใช้แฟล็ก -2

## การออปติไมซ์แฟล็ก

## การอปติไมซ์

คำสั่ง `fdpr` ดำเนินการอปติไมซ์โดยค่าดีฟอลต์ การเรียงลำดับระดับที่เป็นไปได้ของโค้ดใหม่ รวมกับการอปติไมซ์ การตั้งค่า บิตการคาดการณ์สาขา, การตัดสาขา, การจัดเรียงโค้ด และการลบ คำสั่ง `NOOP` มากเกินไป แฟล็ก `-pc` เรียงลำดับ โค้ดทั้งหมด ใหม่ขณะที่ยังคงสงวนขอบเขตของ `csects` ไว้ ดังนั้นอาจส่งผล ให้มีการปรับปรุงผลการทำงานได้น้อยกว่าการเรียงลำดับ ดีฟอลต์โค้ดใหม่ เช่นเดียวกัน แฟล็ก `-pp` เรียงลำดับโค้ดทั้งหมดในขณะที่ยังคงสงวน ขอบเขตของโพธิ์เตอร์ไว้

การอปติไมซ์เพิ่มเติมที่ดำเนินการบนโปรแกรมไฟล์เรียกทำงานทั้งหมด ที่พร้อมใช้โดยแฟล็กการอปติไมซ์ด้านบน

โปรแกรมที่รันได้ที่สร้างขึ้นด้วยคอมไพเลอร์แฟล็ก `-qfdpr IBM xl` ที่มีข้อมูลที่จะช่วย `fdpr` ในการสร้างโปรแกรมที่เรียง ลำดับใหม่ โมดูลซึ่งไม่ถูกคอมไพล์ด้วยอ็อปชัน `-qfdpr` จะถูกเรียงลำดับใหม่ตามลายเซ็นคอมไพเลอร์ ในตารางสัญลักษณ์

การเพิ่มประสิทธิภาพผลการทำงานเพิ่มอาจทราบ โดยการใช้อัลกอริทึมสแตติกเมื่อสร้างโปรแกรมที่จะถูกเรียงลำดับใหม่ เนื่องจาก โปรแกรม `fdpr` เรียงลำดับคำสั่งภายในโปรแกรมเรียกทำงาน ที่ระบุ โลกาวารูทีนที่แบ่งใช้ที่ลิงก์แบบไดนามิกจะถูกเรียกใช้ โดยโปรแกรมที่ไม่ถูกอปติไมซ์ การลิงก์โกลบาวารูทีนแบบสแตติกกับ โปรแกรมที่รันได้ที่อนุญาตให้ทำการอปติไมซ์ทั้งคำสั่ง ในโปรแกรม และโกลบาวารูทีนทั้งหมดที่ใช้โดยโปรแกรม โดยมีข้อดีอื่นรวมถึง ข้อเสียในการสร้าง โปรแกรมที่ลิงก์แบบสแตติก

## เอาต์พุตไฟล์

ไฟล์ทั้งหมดที่สร้างโดยคำสั่ง `fdpr` จะถูกเก็บ อยู่ในไดเรกทอรีปัจจุบันโดยมีชื่อยกเว้นว่าไฟล์ใดๆ ซึ่งอาจถูกสร้างขึ้น โดยคำสั่ง ของเวิร์กโฟลด์ที่กำลังรันที่ระบุในแฟล็ก `-x` ระหว่างกระบวนการอปติไมซ์ โปรแกรมต้นฉบับจะถูกบันทึกโดยการเปลี่ยนชื่อ โปรแกรม และเก็บในชื่อโปรแกรมต้นฉบับต่อเมื่อดำเนินการจนถึงเฟสสุดท้าย สำเร็จแล้วเท่านั้น

โปรไฟล์ไฟล์ที่สร้างโดยคำสั่ง `fdpr` จะใช้ชื่อเต็ม ของไดเรกทอรีปัจจุบันโดยชัดเจน เนื่องจากสคริปต์ที่ใช้รันโปรแกรม อาจ เปลี่ยนแปลงไดเรกทอรีการทำงานก่อนการเรียกใช้งานโปรแกรม

ไฟล์ที่สร้างและ/หรือใช้โดยคำสั่ง `fdpr` ได้แก่:

### ไอเท็ม

`program`

`program.save`

`program.nprof`

`program.instr`

`program.fdpr`

`program.instr.dis_text`

`program.fdpr.dis_text`

`program.instr.dis_data`

`program.fdpr.dis_data`

`program.instr.dis_bss`

`program.fdpr.dis_bss`

`program.instr.mapper`

`program.fdpr.mapper`

`program.ncounts`

### คำอธิบาย

ชื่อของโปรแกรมเรียกทำงานที่ไม่ถูกแบ่งส่วน ซึ่งจะถูกอปติไมซ์

เวอร์ชันที่บันทึกไว้ของโปรแกรมเรียกทำงานต้นฉบับ

ชื่อของโปรไฟล์ไฟล์

ชื่อของเวอร์ชันที่เป็นเครื่องมือของโปรแกรม

ชื่อดีฟอลต์ของเอาต์พุตไฟล์ที่เรียกทำงานที่ถูกอปติไมซ์

ไฟล์ถอดแยกภาษาแอสเซมบลีดีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm` หลังเฟสกา รทำให้เป็นเครื่องมือ

ไฟล์ถอดแยกภาษาแอสเซมบลีดีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm` หลังเฟสกา รอปติไมซ์

ไฟล์ถอดแยกภาษาแอสเซมบลีดีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm_data` หลังเฟสการทำให้เป็นเครื่องมือ

ไฟล์ถอดแยกภาษาแอสเซมบลีดีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm_data` หลังเฟสการอปติไมซ์

ไฟล์ถอดแยกภาษาแอสเซมบลีดีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm_bss` หลังเฟสการทำให้เป็นเครื่องมือ

ไฟล์ถอดแยกภาษาแอสเซมบลีดีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm_bss` หลังเฟสการอปติไมซ์

ไฟล์การแมปดีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-map` หลังเฟสการทำให้เป็นเครื่อง มือ

ไฟล์การแมปดีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-map` หลังเฟสการอปติไมซ์

ไฟล์ตัวนับโปรไฟล์ดีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-profcnt`

## ความสามารถการดีบั๊กที่ขยายเพิ่ม

ในการเปิดใช้ระดับที่แน่นอนของความสามารถการดีบั๊กสำหรับโปรแกรมที่ ออपติไมซ์ **FDPR** จะอัปเดตตารางสัญลักษณ์เพื่อให้เห็น การเปลี่ยนแปลงได้ทำไปในส่วน `.text`

ฟิลด์รายการในตารางสัญลักษณ์ที่ระบุแอดเดรสของสัญลักษณ์ที่ถูกเปลี่ยนสำหรับระหว่างการจัดลำดับ **FDPR** ใหม่ จะถูกแก้ไข เพื่อชี้ไปยังแอดเดรสใหม่ในส่วน `.text`

นอกจากนี้ ในกรณีที่ฟังก์ชันหรือไฟล์ถูกแบ่งระหว่างการจัดอันดับใหม่ **FDPR** จะสร้างรายการใหม่ในตารางสัญลักษณ์สำหรับแต่ละส่วน ที่ถูกแบ่งใหม่ของฟังก์ชัน/ไฟล์ที่แบ่ง ส่วนใหม่เหล่านี้ที่ที่เป็นฟังก์ชันเดียวกัน จะได้รับการกำหนดชื่อสัญลักษณ์ใหม่ในตารางสัญลักษณ์ตามข้อกำหนดการตั้งค่า ต่อไปนี้:

```
<original function name>__fdpr_<function's part number>
```

หลังจากการจัดเรียงโค้ดใหม่เสร็จแล้ว รายการใหม่ทั้งหมดจะถูกเติมท้ายด้วยสตริง `__fdpr_`

ตัวอย่าง: ในเริ่มแรก ฟังก์ชัน "main" มีรายการต่อไปนี้ในตารางสัญลักษณ์:

[Index]	m	Value	Scn	Aux	Sclass	Type	Name
[456]	m	0x00000230	2	1	0x02	0x0000	.main

หลังจากจัดเรียงโค้ดใหม่ ฟังก์ชัน main ถูกแบ่งเป็น 3 ส่วน จึงมี 3 รายการในตารางสัญลักษณ์ หนึ่งรายการสำหรับหนึ่งส่วนดังต่อไปนี้:

[Index]	m	Value	Scn	Aux	Sclass	Type	Name
[456]	m	0x00000304	2	1	0x02	0x0000	.main
[1447]	m	0x00003328	2	1	0x02	0x0000	.main__fdpr_1
[1453]	m	0x000033b4	2	1	0x02	0x0000	.main__fdpr_2

## ตัวอย่าง

ต่อไปนี้เป็นตัวอย่างการใช้งานโดยทั่วไปของคำสั่ง `fdpr`

1. ตัวอย่างนี้ให้ผู้ใช้รันได้ทั้งสามเฟส ในตัวอย่างนี้ test1 เป็นโปรแกรมที่รันได้ที่ไม่ถูกแบ่งส่วนและ test2 เป็นเซลล์สคริปต์ที่เรียกใช้ test1 ไดรฟ์ทอริการทำงานขณะนี้เป็น `/tmp/fdpr`

สคริปต์ไฟล์ test2:

```
# code to exercise test1
test1 -expand 100 -root $PATH file.jpg -quit
# the end of test2
```

เรียกใช้งานคำสั่ง `fdpr` (โดยใช้ การออพติไมซ์ที่เป็นค่าดีฟอลต์):

```
fdpr -p test1 -x test2
```

คำสั่งนี้ให้ผลลัพธ์ เป็นโปรแกรมที่รันได้ที่ถูกจัดเรียงใหม่ `test1.f DPR`

2. ในการรันครั้งละหนึ่งเฟส ให้เรียกใช้เฟสหนึ่งใน `fdpr`

```
fdpr -1 -p test1
```

สตริงคำสั่งนี้สร้างเวอร์ชัน ที่เป็นเครื่องมือที่มีชื่อ `test1.instr` และเพิ่มเฟลตโปรไฟล์ไฟล์ว่าง `test1.nprof`

ในการเรียกใช้งานเฟสสอง:

```
fdpr -2 -p test1 -x test2
```

สตริงคำสั่งนี้เรียกใช้งานสคริปต์ไฟล์ test2 ที่รันเวอร์ชันที่เป็นเครื่องมือของ test1 เพื่อรวบรวมข้อมูลโปรไฟล์

ในการเรียกใช้งานเฟสสาม:

```
fdpr -3 -p test1
```

และคำสั่งให้ผลลัพธ์ เป็นโปรแกรมที่รันได้ที่จัดเรียงใหม่ test1.fdpr

3. ในการรันสองเฟสแรกตามด้วยเฟสที่สาม ให้เรียกใช้งานเฟสหนึ่ง และสอง

```
fdpr -12 -p test1 -x test2
```

เรียกใช้งานเฟสสามโดยใช้ ระดับออปติไมซ์ระดับสาม

```
fdpr -3 -O3 -p test1
```

4. หากมีข้อผิดพลาดเกิดขึ้นระหว่างการรันโปรแกรมที่ออปติไมซ์ fdpr สามารถใช้คำสั่ง dbx เพื่อหาว่า โพรซีเดอร์ใดที่เกิดข้อผิดพลาดขึ้นดังที่แสดงต่อไปนี้:

```
dbx program.fdpr
```

ซึ่งจะสร้างเอาต์พุตที่คล้ายต่อไปนี้:

```
Type 'help' for help.
```

```
reading symbolic information ...warning: no source compiled with -g
```

```
[using memory image in core]
```

```
Segmentation fault in proc_d at 0x10000634
```

```
0x10000634 (???) 98640000 stb r3,0x0(r4)
```

```
(dbx)
```

สแต็กการติดตามย้อนกลับ ซึ่งใช้กำหนดวิธีที่โปรแกรมจะมาที่ตำแหน่งปัจจุบัน โดยจะถูกสร้างดังนี้:

```
(dbx) where
```

ซึ่งจะสร้างเอาต์พุตต่อไปนี้:

```
proc_d(0x0) at 0x10000634
```

```
proc_c(0x0) at 0x10000604
```

```
proc_b(0x0) at 0x100005d0
```

```
proc_a(0x0) at 0x1000059c
```

```
main(0x2, 0x2ff7fba4) at 0x1000055c
```

```
(dbx)
```

5. คำสั่งย่อย dbx stepi ยังอาจใช้เพื่อดำเนินขั้นตอนเดียวตามคำสั่งเครื่องของโปรแกรมเรียกทำงานที่เรียงลำดับใหม่ ดังนี้:

```
(dbx) stepi
```

ซึ่งจะสร้างเอาต์พุตต่อไปนี้:

```
stopped in proc_d at 0x1000061c
```

```
0x1000061c (???) 9421ffc0 stwu r1,-64(r1)
```

```
(dbx)
```

ในตัวอย่างนี้ dbx ระบุว่า โปรแกรมหยุดทำงานในรูทีน proc\_d ที่แอดเดรส 0x1000061c ในส่วนข้อความที่จัดเรียงใหม่

## ข้อมูลจำเพาะการนำไปใช้งาน

Software Product/Option: AIX Performance Aide/ Local Performance Analysis & Control Commands.

Standards Compliance: ไม่มี.

## ไฟล์

### ไอเท็ม

`/usr/bin/fdpr`  
`program`  
`program.save`  
`program.nprof`  
`program.instr`  
`program.fdpr`  
`program.instr.dis_text`  
  
`program.fdpr.dis_text`  
  
`program.instr.dis_data`  
  
`program.fdpr.dis_data`  
  
`program.instr.dis_bss`  
  
`program.fdpr.dis_bss`  
  
`program.instr.mapper`  
  
`program.fdpr.mapper`  
`program.ncounts`

### คำอธิบาย

มีคำสั่ง `fdpr`  
ชื่อของโปรแกรมเรียกทำงานที่ไม่ถูกแบ่งส่วน ซึ่งจะถูกรวบรวม  
เวอร์ชันที่บันทึกไว้ของโปรแกรมเรียกทำงานฉบับ  
ชื่อของโปรไฟล์ไฟล์  
ชื่อของเวอร์ชันที่เป็นเครื่องมือของโปรแกรม  
ชื่อไฟล์ของเอาต์พุตไฟล์ที่เรียกทำงานที่ถูกรวบรวม  
ไฟล์ถอดแยกภาษาแอสเซมบลีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm` หลังเฟสกา  
รทำให้เป็นเครื่องมือ  
ไฟล์ถอดแยกภาษาแอสเซมบลีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm` หลังเฟสกา  
รรวบรวม  
ไฟล์ถอดแยกภาษาแอสเซมบลีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm_data`  
หลังเฟสการทำให้เป็นเครื่องมือ  
ไฟล์ถอดแยกภาษาแอสเซมบลีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm_data`  
หลังเฟสการรวบรวม  
ไฟล์ถอดแยกภาษาแอสเซมบลีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm_bss`  
หลังเฟสการทำให้เป็นเครื่องมือ  
ไฟล์ถอดแยกภาษาแอสเซมบลีฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-disasm_bss`  
หลังเฟสการรวบรวม  
ไฟล์การแมปฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-map` หลังเฟสการทำให้เป็นเครื่อง  
มือ  
ไฟล์การแมปฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-map` หลังเฟสการรวบรวม  
ไฟล์ตัวนับโปรไฟล์ฟอลต์ในรูปแบบ ASCII ที่สร้างโดยแฟล็ก `-profcnt`

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dbx`” ในหน้า 9

### ข้อมูลที่เกี่ยวข้อง:

การสร้างโปรแกรมที่ดำเนินการได้ขั้นใหม่ด้วยโปรแกรม `fdpr`

---

## คำสั่ง `fencevsd`

### วัตถุประสงค์

ป้องกันแอพลิเคชันที่กำลังบนโหนดหรือกลุ่มของโหนดมิให้ เข้าถึงดิสก์ที่แบ่งใช้แบบเสมือน หรือกลุ่มของดิสก์ที่แบ่งใช้แบบเสมือน

### ไวยากรณ์

```
fencevsd { -a | -v vsd_name_list } -n node_list
```

## คำอธิบาย

ภายใต้สถานการณ์บางอย่าง ระบบอาจเชื่อว่าโหนดได้หยุดการทำงานแล้ว และเริ่มต้นกระบวนการกู้คืน เมื่อจริงๆ แล้วโหนดกำลังดำเนินการอยู่แต่ ถูกตัดออกจากการสื่อสารกับโหนดอื่นๆ ที่กำลังรันแอสพลีเคชันเดียวกัน ในกรณีนี้โหนดที่เป็นปัญหาต้องไม่ได้รับอนุญาตให้ทำหน้าที่ตามการร้องขอสำหรับ ดิสก์ที่แบ่งใช้แบบเสมือน ที่โดยปกติทำหน้าที่จนกระทั่งการกู้คืนเสร็จสมบูรณ์และโหนดอื่นๆ ที่กำลังรันแอสพลีเคชันเข้าใจว่าโหนดที่เป็นปัญหากำลังดำเนินการอยู่ คำสั่ง `fencevsd` ป้องกันโหนดที่เป็นปัญหาจากการทำตามการร้องขอ สำหรับดิสก์ที่แบ่งใช้แบบเสมือน

คำสั่งนี้สามารถรันจากโหนดใดๆ ในเพียร์โตเมน RSCT โดยที่ ระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้กำลังรันอยู่

## แฟล็ก

-a ระบุดิสก์ที่แบ่งใช้แบบเสมือนทั้งหมด

-v *vsd\_name\_list*

ระบุชื่อดิสก์ที่แบ่งใช้แบบเสมือนอย่างน้อยหนึ่งชื่อ คั่นด้วยเครื่องหมายจุลภาค

-n *node\_list*

ระบุหมายเลขโหนดอย่างน้อยหนึ่งหมายเลข คั่นด้วยเครื่องหมายจุลภาค

## พารามิเตอร์

*logical\_volume\_name*

คือชื่อของโลจิคัลวอลุ่มที่คุณต้องการระบุเป็นดิสก์ที่แบ่งใช้แบบเสมือน วอลุ่มโลจิคัลนี้ต้องอยู่ภายในกลุ่มวอลุ่มโกลบอลที่ระบุ ความยาวชื่อต้องน้อยกว่าหรือเท่ากับ 15 อักขระ

*global\_group\_name*

คือชื่อของกลุ่มวอลุ่มเข้าถึงได้แบบโกลบอลที่กำหนดก่อนหน้านี้ โดยคำสั่ง `vsdvg` ที่คุณต้องการระบุดิสก์ที่แบ่งใช้แบบเสมือน ความยาวชื่อต้องน้อยกว่าหรือเท่ากับ 31 อักขระ

*vsd\_name*

ระบุชื่อเฉพาะสำหรับดิสก์ที่แบ่งใช้แบบเสมือนใหม่ ชื่อนี้ต้อง เป็นชื่อเฉพาะภายใน RSCT เพียร์โตเมน และเพื่อหลีกเลี่ยงความขัดแย้ง การตั้งชื่อในอนาคต ชื่อควรเป็นชื่อเฉพาะทั่วทั้งคลัสเตอร์ทั้งหมด ระเบียบในการตั้งชื่อ ที่แนะนำคือ `vsdnngvg_name` ความยาวชื่อต้องน้อยกว่าหรือเท่ากับ 31 อักขระ

**หมายเหตุ:** ถ้าคุณระบุ *vsd\_name* ที่เป็นชื่อของอุปกรณ์อื่นแล้ว คำสั่ง `cfgrvsd` จะทำงานไม่สำเร็จสำหรับดิสก์ที่แบ่งใช้แบบเสมือนนั้น ขอผิดพลาดนี้ประกันว่าไฟล์อุปกรณ์พิเศษที่สร้างสำหรับ ชื่อที่กำหนดไม่ซ้อนกันและทำลายไฟล์ที่มีชื่อเหมือนกัน ซึ่งมีอยู่ในชนิดอุปกรณ์อื่น (เช่น โลจิคัลวอลุ่ม)

## ความปลอดภัย

คุณต้องมีสิทธิ์การใช้งาน `root` เพื่อรันคำสั่งนี้

## ข้อจำกัด

คุณต้องเรียกใช้คำสั่งนี้จากโหนดในเพียร์โตเมนที่มีระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้แอดที่พออยู่

## ตัวอย่าง

ในการจำกัดดิสก์ที่แบ่งใช้แบบเสมือน vsd1 และ vsd2 จากโหนด 5 ให้ป้อน:

```
fencevsd -v vsd1,vsd2 -n 5
```

## Location

/opt/rsct/vsd/bin/fencevsd

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง lsvsd

คำสั่ง unfencevsd

---

## คำสั่ง ff

### วัตถุประสงค์

แสดงรายการชื่อไฟล์และสถิติสำหรับระบบไฟล์

### ไวยากรณ์

```
ff [-a Number] [-c Number] [-I] [-l] [-m Number] [-n File] [-o Options] [-p Prefix] [-s] [-u] [-V VFSName] [-i I-Number[,I-Number...]] [FileSystem|DeviceName]
```

### คำอธิบาย

คำสั่ง `ff` อ่าน `i-nodes` ในระบบไฟล์ที่ระบุโดยพารามิเตอร์ `FileSystem` จากนั้นเขียนข้อมูลที่เกี่ยวข้องไปยังเอาต์พุตมาตรฐาน โดยถือว่า `FileSystem` คือระบบไฟล์ ซึ่งถูกอ้างถึงในไฟล์ `/etc/filesystems` และบันทึกข้อมูล `i-node` สำหรับไฟล์ที่ระบุโดยแฟล็ก

เอาต์พุตจากคำสั่ง `ff` ประกอบด้วยชื่อพาธสำหรับแต่ละหมายเลข `i-node` ที่ร้องขอ นอกเหนือจาก ข้อมูลไฟล์อื่นที่คุณสามารถร้องขอโดยใช้แฟล็ก เอาต์พุตถูกแสดงตามลำดับหมายเลข `i-node` ที่มีแท็บระหว่างฟิลด์ทั้งหมด บรรทัด ดีฟอลต์ที่สร้างโดยคำสั่ง `ff` ประกอบด้วยฟิลด์ชื่อพาธ และหมายเลข `i-node` เมื่อเปิดใช้แฟล็กทั้งหมด เอาต์พุตฟิลด์จะประกอบด้วย ชื่อพาธ, หมายเลข `i-node`, ขนาด และ UID (ID ผู้ใช้)

พารามิเตอร์ `Number` เป็นเลขฐานสิบที่ระบุจำนวนวัน โดยนำหน้าด้วย `+` หรือ `-` (เครื่องหมาย บวก หรือลบ) ดังนั้น `+3` หมายความว่ามากกว่า 3 วัน, `-3` หมายความว่าน้อยกว่า 3 วัน และ `3` หมายถึง 3 วัน โดยที่วันถูกกำหนดเป็นรอบเวลา 24 ชั่วโมง

คำสั่ง `ff` แสดงชื่อพาธเดียวเท่านั้นจากชื่อที่เป็นไปได้หลายๆ ชื่อสำหรับ `i-node` ที่มีมากกว่าหนึ่งลิงก์ ยกเว้นคุณระบุแฟล็ก `-l` ด้วยแฟล็ก `-l` คำสั่ง `ff` จะแสดงรายการลิงก์ทั้งหมด

### แฟล็ก

ไอเท็ม	คำอธิบาย
-a Number	แสดงไฟล์หากถูกเข้าถึงภายในจำนวนวันที่ระบุโดยพารามิเตอร์ <i>Number</i>
-c Number	แสดงไฟล์หาก i-node ถูกเปลี่ยนแปลงภายในจำนวนวันที่ระบุโดยพารามิเตอร์ <i>Number</i>
-i I-Number	แสดงไฟล์ที่สอดคล้องกับหมายเลข i-node ที่ระบุ โดยพารามิเตอร์ <i>I-Number</i> หมายเลข i-node ที่แสดง ต้องค้นด้วยเครื่องหมายจุลภาค
-I	(แฟล็กนี้เป็น i ตัวพิมพ์ใหญ่) ไม่แสดง i-node หลัง ชื่อพาธแต่ละชื่อ

ไอเท็ม	คำอธิบาย
-l	(แฟล็กนี้เป็น L ตัวพิมพ์เล็ก) นอกจากนั้น แสดงรายการชื่อพาธสำหรับไฟล์ที่มีมากกว่าหนึ่งลิงก์
-m Number	แสดงไฟล์หากถูกแก้ไขภายในจำนวนวันที่ระบุโดยพารามิเตอร์ <i>Number</i>
-n File	แสดงไฟล์หากถูกแก้ไข ล่าสุดกว่าไฟล์ที่ระบุโดยพารามิเตอร์ <i>File</i>
-o Options	ระบุรายการที่ค้นด้วยเครื่องหมายจุลภาค ของอ็อพชันที่นำไปใช้งานโดยเฉพาะสำหรับระบบไฟล์เสมือน
	อ็อพชันต่อไปนี้ ระบุเฉพาะกับระบบไฟล์ที่ทำเจอร์นัลแล้ว (JFS2) ซึ่งได้รับการพัฒนาแล้ว:
-p Prefix	-o snapshot= <i>snapName</i> ระบุชื่อของ snapshot subject ภายในให้แกเป็น ff ระบบไฟล์ที่เป็นเจ้าของ snapshot ต้องถูก mount เพิ่มคำนำหน้าทีระบุโดยพารามิเตอร์ <i>Prefix</i> ให้กับชื่อพาธแต่ละชื่อ คำนำหน้าดีฟอลต์ คือ . (จุด)
-s	เขียนขนาดไฟล์เป็นไบต์ หลังชื่อพาธแต่ละชื่อ
-u	เขียนชื่อล็อกอินของเจ้าของหลัง ชื่อพาธแต่ละชื่อ
-V VFSName	สั่งให้คำสั่ง ff ถือว่าระบบไฟล์เป็นประเภท <i>VFSName</i> โดยแทนที่ค่าในไฟล์ /etc/filesystems

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

- ในการแสดงชื่อพาธของไฟล์ทั้งหมดใน ระบบไฟล์ที่กำหนดให้ป้อน:

```
ff -I /dev/hd0
```

คำสั่งนี้ แสดงชื่อพาธของไฟล์บนอุปกรณ์ /dev/hd0 หากคุณไม่ระบุแฟล็ก -I คำสั่ง ff จะแสดงหมายเลข i-node ของแต่ละไฟล์ด้วย

- ในการแสดงไฟล์ที่ถูกแก้ไขเมื่อเร็วๆ นี้ให้ป้อน:

```
ff -m -2 -u /dev/hd0
```

คำสั่งนี้ แสดงชื่อพาธ, หมายเลข i-node และชื่อผู้ใช้ของเจ้าของ (แฟล็ก -u) ของแต่ละไฟล์บนไฟล์ /dev/hd0 ที่ถูกแก้ไขภายในสองวันล่าสุด (-m -2)

- ในการแสดงไฟล์ที่ไม่ได้ถูกใช้งานเมื่อเร็วๆ นี้ให้ป้อน:

```
ff -a +30 /dev/hd0
```

คำสั่งนี้แสดงชื่อพาธและ i-node ของแต่ละไฟล์ที่ ถูกเข้าถึงครั้งสุดท้ายนานมากกว่า 30 วัน (-a +30)

- ในการค้นหาพาธที่เกี่ยวข้องกับ หมายเลข i-node ที่แน่นอน ให้ป้อน:

```
ff -l -i 451,76 /dev/hd0
```

คำสั่งนี้แสดงชื่อพาร์ติชันทั้งหมด (-l) ที่สัมพันธ์กับ i-nodes 451 และ 76

## ไฟล์

ไอเท็ม	คำอธิบาย
/etc/vfs	มีคำอธิบายของชนิดของระบบไฟล์เสมือน
/etc/filesystems	แสดงรายการของระบบไฟล์ที่รู้จัก และกำหนดคุณสมบัติไว้

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง find” ในหน้า 568

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง ncheck

ระบบไฟล์

---

## คำสั่ง fg

### วัตถุประสงค์

รันงานที่ส่วนหน้า

### ไวยากรณ์

```
fg [JobID]
```

### คำอธิบาย

หากการควบคุมงานถูกเปิดใช้ (ดูที่ “การควบคุมงานงานคอร์นเชลล์หรือ POSIX เชลล์” ใน *Operating system and device management*) คำสั่ง fg จะย้ายงานส่วนหลังในสถานะแวดล้อมปัจจุบัน ไปยังส่วนหน้า ใช้พารามิเตอร์ JobID เพื่อระบุงานที่เจาะจงที่จะรันที่ส่วนหน้า หากพารามิเตอร์นี้ไม่ได้กำหนด คำสั่ง fg จะใช้งานที่หยุดทำงานชั่วคราวล่าสุด ที่อยู่ในส่วนหลัง หรือ รันเป็นงานส่วนหลัง

พารามิเตอร์ JobID สามารถเป็น หมายเลข ID กระบวนการ หรือคุณสามารถใช้หนึ่งในการใช้สัญลักษณ์ร่วมกันต่อไปนี้:

ไอเท็ม	คำอธิบาย
%Number	อ้างถึงงานโดยใช้หมายเลขงาน
%String	อ้างถึงงานที่มีชื่อขึ้นต้นด้วยสตริงที่ระบุ
%?String	อ้างถึงงานที่มีสตริงที่ระบุ
%+ หรือ %%	อ้างถึงงานปัจจุบัน
%-	อ้างถึงงานก่อนหน้า

การใช้คำสั่ง fg เพื่อใส่ งานลงในส่วนหน้า ลบ ID กระบวนการของงานออกจากรายการของงาน ที่สถานะแวดล้อมเชลล์ปัจจุบัน รู้จัก

คำสั่ง /usr/bin/fg ไม่ทำงาน เมื่อดำเนินงานในสถานะแวดล้อมการเรียกใช้งานคำสั่งของตนเอง เนื่องจาก สถานะแวดล้อมไม่มีงานที่ใช้ได้สำหรับการจัดการ ด้วยเหตุนี้ คำสั่ง fg จะถูกนำไปใช้เป็นคำสั่งในตัวปกติของ คอร์นเชลล์หรือ POSIX เชลล์.

## สถานะออก

ค่าออกต่อไปนี้ถูกส่งคืน:

ไอเท็ม คำอธิบาย  
0 ดำเนินการเป็นผลสำเร็จ  
>0 มีข้อผิดพลาดเกิดขึ้น

หากการควบคุมงานถูกปิดใช้ คำสั่ง `fg` จะออกจากการทำงานพร้อมข้อผิดพลาด และไม่มีงานอยู่ในส่วนหน้า

## ตัวอย่าง

หากเอาต์พุตของคำสั่ง `jobs -l` แสดงงานที่กำลังรันในส่วนหลังต่อไปนี้:

```
[1] + 16477RunningSleep 100 &
```

ใช้ ID กระบวนการเพื่อรันคำสั่ง `sleep 100 &` ในส่วนหน้าโดยการป้อน:

```
fg 16477
```

หน้าจอจะแสดง:

```
sleep
```

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/ksh</code>	มีคำสั่งในตัว <code>fg</code> คอร์นเชลล์
<code>/usr/bin/fg</code>	มีคำสั่ง <code>fg</code>

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `bg`

คำสั่ง `csh`

คำสั่ง `wait`

การควบคุมงานใน Korn เชลล์หรือ POSIX เชลล์

---

## คำสั่ง `fgrep`

### Purpose

ค้นหาตรงตามตัวอักษรในไฟล์

### ไวยากรณ์

```
fgrep [-h] [-i] [-s] [-u] [-v] [-w] [-x] [-y] [[-b] [-n] | [-c | -l | -q]] [-p Separator] {Pattern | -e Pattern | -f StringFile} [File...]
```

## รายละเอียด

คำสั่ง `fgrep` จะค้นหาไฟล์อินพุตที่ระบุโดยพารามิเตอร์ `File` (อินพุตมาตรฐาน โดยดีฟอลต์) สำหรับบรรทัดที่ตรงกับรูปแบบคำสั่ง `fgrep` ค้นหาโดยเฉพาะพารามิเตอร์ `Pattern` ที่เป็นสตริงคงที่ คำสั่ง `fgrep` แสดงไฟล์ที่มีบรรทัดที่ตรงกัน หากคุณระบุมากกว่าหนึ่งไฟล์ในพารามิเตอร์ `File`

คำสั่ง `fgrep` แตกต่างจากคำสั่ง `grep` และ `egrep` เนื่องจาก ค้นหาสตริงแทนการค้นหารูปแบบที่ตรง ตามนิพจน์ คำสั่ง `fgrep` ใช้ลกอริทึมแบบ รวดเร็วและกระชับ อักขระ `$, *, [, ], (, )` และ `\` จะถูกแปลความหมาย ตามตัวอักษรโดยคำสั่ง `fgrep` อักขระเหล่านี้จะไม่ถูกแปลความหมายเป็นส่วนหนึ่งของนิพจน์ปกติ เนื่องจากถูกแปลความหมายในคำสั่ง `grep` และ `egrep` เนื่องจาก อักขระเหล่านี้มีความหมายพิเศษต่อเซลล์ ทั้งสตริงต้องอยู่ภายในเครื่องหมายคำพูดเดี่ยว ('...'). หากไม่ระบุไฟล์ คำสั่ง `fgrep` จะใช้อินพุต มาตรฐาน โดยปกติ แต่ละบรรทัดที่พบจะถูกคัดลอกไปยังเอาต์พุตมาตรฐาน ชื่อไฟล์ถูกพิมพ์ก่อนแต่ละบรรทัดที่พบ หากมีไฟล์อินพุตมากกว่าหนึ่งไฟล์

### Notes:

1. คำสั่ง `fgrep` จะเหมือนกับคำสั่ง `grep` ที่มีแฟล็ก `-F` ยกเว้นว่าข้อความแสดงข้อผิดพลาดและการใช้งานจะแตกต่างกัน และแฟล็ก `-s` ทำงานแตกต่างกัน
2. บรรทัดถูกจำกัดที่ 2048 ไบต์
3. ย่อหน้า (ภายใต้แฟล็ก `-p`) ขณะนี้ถูกจำกัดที่ความยาว 5000 อักขระ
4. ยารันคำสั่ง `grep` กับไฟล์พิเศษเนื่องจากจะสร้างผลที่คาดการณ์ไม่ได้
5. บรรทัดอินพุตต้องไม่มีอักขระ NULL
6. ไฟล์อินพุตต้องลงท้ายด้วยอักขระขึ้นบรรทัดใหม่
7. แม้ว่าสามารถระบุ บางแฟล็กได้แบบพร้อมกัน บางแฟล็กจะแทนที่แฟล็กอื่นๆ ตัวอย่าง ถ้าคุณระบุ `-I` และ `-n` พร้อมกัน เฉพาะชื่อไฟล์ ถูกเขียนไปที่เอาต์พุตมาตรฐาน

## แฟล็ก

แฟล็ก	คำอธิบาย
<code>-b</code>	นำหน้าแต่ละบรรทัดด้วยหมายเลขบล็อกที่พบ ใช้แฟล็กนี้เพื่อช่วยค้นหาหมายเลขบล็อกของคันทัสก์ตามบริบท แฟล็ก <code>-b</code> ไม่สามารถใช้ร่วมกับอินพุตจาก <code>stdin</code> หรือ <code>pipe</code>
<code>-c</code>	แสดงเฉพาะจำนวนบรรทัดที่ตรงกันเท่านั้น
<code>-e Pattern</code>	ระบุรูปแบบ ซึ่งทำงานเหมือนกับรูปแบบอย่างง่าย แต่ มีประโยชน์เมื่อรูปแบบขึ้นต้นด้วย <code>-</code> (เครื่องหมายลบ)
<code>-f StringFile</code>	ระบุไฟล์ที่มีสตริง หมายเหตุ: เมื่อต้องการ เพิ่มประสิทธิภาพของคำสั่ง <code>fgrep</code> (หรือ <code>grep -F</code> ) ที่มีอินพุตเป็นไฟล์ที่มีรูปแบบการค้นหา ให้เอ็กซ์พอร์ตตัวแปรสถานะแวดล้อม <code>ENABLE_FGREP_AC</code> ก่อนที่คุณจะรันคำสั่ง <code>fgrep</code> ตัวอย่างเช่น คุณสามารถรันคำสั่งต่อไปนี้เพื่อเอ็กซ์พอร์ตตัวแปรสถานะแวดล้อมนี้: <pre>export ENABLE_FGREP_AC=""</pre>
<code>-h</code>	ยับยั้งชื่อไฟล์เพื่อประมวลผลไฟล์หลายไฟล์
<code>-i</code>	ไม่สนใจขนาดตัวพิมพ์ของตัวอักษรเมื่อเปรียบเทียบ
<code>-l</code>	แสดงเพียงชื่อไฟล์ (ครั้งเดียว) กับบรรทัดที่พบ แต่ละ ชื่อไฟล์ถูกแยกด้วยอักขระขึ้นบรรทัดใหม่
<code>-n</code>	นำหน้าแต่ละบรรทัดด้วยหมายเลขบรรทัดสัมพัทธ์ในไฟล์
<code>-p Separator</code>	แสดงทั้งย่อหน้าที่มีบรรทัดที่ตรงกัน ย่อหน้า ถูกค้นด้วยตัวค้นย่อหน้า ตามที่ระบุโดยพารามิเตอร์ <code>Separator</code> ซึ่งรูปแบบ อยู่ในฟอร์มเดียวกับรูปแบบการค้นหา บรรทัดที่มีตัวค้นย่อหน้า ถูกใช้เฉพาะเป็นตัวค้น; จะไม่มีการรวมไว้ในเอาต์พุต ตัวค้นย่อหน้าดีฟอลต์คือบรรทัดว่าง
<code>-q</code>	ยับยั้งการเขียนทั้งหมดทั้งหมดไปที่เอาต์พุตมาตรฐาน ไม่ว่าบรรทัด จะตรงกันหรือไม่ จบ ด้วยสถานะ 0 ถ้าบรรทัดอินพุตถูกเลือก
<code>-s</code>	แสดงเฉพาะข้อความแสดงข้อผิดพลาด ซึ่งมีประโยชน์สำหรับการตรวจสอบสถานะ
<code>-u</code>	ทำให้เอาต์พุตถูกยกเลิกบัฟเฟอร์

แฟล็ก	คำอธิบาย
-v	แสดงบรรทัดทั้งหมดยกเว้นบรรทัดที่ตรงกับรูปแบบที่ระบุ
-w	ค้นหาคำ
-x	แสดงบรรทัดที่ตรงกับรูปแบบพอดี โดยไม่มีอักขระเพิ่มเติม
-y	ไม่สนใจขนาดตัวพิมพ์ของตัวอักษรเมื่อเปรียบเทียบ

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	พบข้อมูลตรงกัน
1	ไม่พบข้อมูลตรงกัน
>1	พบข้อผิดพลาดทางไวยากรณ์หรือไม่เข้าถึงไม่ได้ (แม้ว่าจะพบข้อมูลที่ตรงกัน)

## ตัวอย่าง

1. ในการค้นหาสตริงอักขระอย่างง่าย ในไฟล์หลายๆ ไฟล์

```
fgrep strcpy *.c
```

ซึ่งจะค้นหาสตริง strcpy ในไฟล์ทั้งหมดในไดเรกทอรีปัจจุบันที่มีชื่อลงท้ายด้วยสตริงอักขระ .c

2. ในการนับจำนวนบรรทัดที่ตรงตาม รูปแบบ:

```
fgrep -c "{" pgm.c
```

```
fgrep -c "}" pgm.c
```

โดยแสดงจำนวนบรรทัดทั้งหมดใน pgm.c ที่มีวงเล็บปีกกาซ้าย และขวา

หากคุณไม่ใส่มากกว่าหนึ่ง { (วงเล็บปีกกาซ้าย) หรือหนึ่ง } (วงเล็บปีกกาขวา) บนบรรทัด ในโปรแกรม C ของคุณ และหากวงเล็บปีกกาครบคู่พอดี จำนวน สองจะถูกแสดงและโดยปกติจะเหมือนกันหากตรงตามเงื่อนไขที่เหมาะสม หากจำนวนไม่เหมือนกัน คุณสามารถแสดงบรรทัดที่มีวงเล็บปีกกาตามลำดับที่เกิดขึ้นในไฟล์ด้วย:

```
egrep {\|} pgm.c
```

3. ในการแสดงชื่อของไฟล์ที่มีรูปแบบ:

```
fgrep -l strcpy *.c
```

ซึ่งจะค้นหาไฟล์ในไดเรกทอรีปัจจุบันที่ลงท้ายด้วย .c และแสดงชื่อของไฟล์ที่มีสตริง strcpy

## ไฟล์

File	คำอธิบาย
/usr/bin/fgrep	มีคำสั่ง fgrep
/bin/fgrep	ลิงก์สัญลักษณ์ไปยังคำสั่ง fgrep

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง egrep” ในหน้า 357

“คำสั่ง grep” ในหน้า 736

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง sed

## คำสั่ง file

### วัตถุประสงค์

พิจารณาประเภทไฟล์

### ไวยากรณ์

ในการจัดหมวดหมู่ประเภทไฟล์

```
file [-m MagicFile] [-d] [-h] [-i] [-M MagicFile] [-f FileList] [File...]
```

ในการตรวจสอบเมจิกไฟล์เพื่อหาข้อผิดพลาดของรูปแบบ

```
file -c [-m MagicFile]
```

### คำอธิบาย

คำสั่ง file อ่านไฟล์ที่ระบุโดยพารามิเตอร์ *File* หรือตัวแปร *FileList* แล้วดำเนินการชุดการทดสอบกับแต่ละไฟล์ และพยายามจัดหมวดหมู่ตามประเภท จากนั้นคำสั่งจะเขียนประเภทไฟล์ไปยังเอาต์พุต มาตรฐาน ไฟล์สามารถเป็นไฟล์ปกติ, ไตเร็กทอรี, FIFO (named pipe), บล็อกพิเศษ, อักขระพิเศษ, ลิงก์สัญลักษณ์ หรือประเภทซ็อกเก็ต

- หากเป็นไฟล์ปกติ และมีความยาวเป็นศูนย์ จะถูกระบุเป็นไฟล์ว่าง
- หากไฟล์เป็นลิงก์สัญลักษณ์ โดยค่าดีฟอลต์ ลิงก์จะตามด้วย ไฟล์ที่สัญลักษณ์นั้นอ้างอิงถึง

หากปรากฏว่าไฟล์อยู่ในรูปแบบ ASCII คำสั่ง file จะตรวจสอบ 1024 ไบต์แรก และพิจารณาประเภทไฟล์ หากว่าไฟล์ไม่ได้อยู่ในรูปแบบ ASCII คำสั่ง file จะพยายามต่อไปโดยแยกไฟล์ข้อมูลไบนารีจากไฟล์ข้อความ ที่มีอักขระส่วนขยาย

หากพารามิเตอร์ *File* ระบุ โมดูลไฟล์ที่รันได้หรือที่เป็นอ็อบเจกต์ และหมายเลขเวอร์ชันมากกว่า 0 คำสั่ง file จะแสดงเวลาประทับของเวอร์ชัน คำสั่ง `ld` อธิบายการใช้ไฟล์ `a.out`

หากสถานะแวดล้อมของภาษาเป็นภาษาโปรแกรมภาษา C คำสั่ง file จะใช้ไฟล์ `/etc/magic` เพื่อระบุ ไฟล์ที่มีข้อมูลจำพวกเลขเมจิก นั่นคือไฟล์ใดๆ ที่มี ค่าคงที่ตัวเลขหรือสตริงที่ระบุประเภท

อย่างไรก็ตาม ถ้าสถานะแวดล้อมภาษาเป็น ภาษาอื่นที่ไม่มีภาษาโปรแกรม C คำสั่ง file ใช้ไฟล์ `/usr/lib/nls/msg/<language_env>/magic.cat` เพื่อระบุไฟล์ที่มีหมายเลขเมจิก

หากไฟล์ไม่มีอยู่ ไม่สามารถอ่านได้ หรือไม่สามารถระบุค่าสถานะของไฟล์ได้ ไฟล์จะไม่ถูกพิจารณาว่ามีข้อผิดพลาดที่ส่งผลกระทบต่อสถานะ การออกจากการทำงาน เอาต์พุตจะระบุว่าไฟล์ถูกประมวลผลแต่ไม่สามารถระบุ ประเภทได้

เมื่อใช้แฟล็ก `-i` รูปแบบต่อไปนี้ จะถูกใช้เพื่อระบุตัวถูกดำเนินการ *file* ที่ระบุแต่ละตัว:

```
"%s: %s\n", file, type
```

ค่าสำหรับ *type* ไม่ถูกระบุยกเว้นในโลแคล POSIX หาก *file* ถูกระบุเป็นหนึ่งใน ประเภทที่แสดงในตารางต่อไปนี้ *type* จะมี (แต่ไม่จำกัด) ระบุที่สอดคล้องกัน แต่ละช่องว่างที่แสดงใน สตริงจะเป็นหนึ่ง *space* อย่างแท้จริง

ตารางที่ 5. File Utility Output Strings

หาก <i>file</i> เป็น:	<i>type</i> จะมีสตริง:
ไดเรกทอรี	ไดเรกทอรี
FIFO	fifo
ซ็อกเก็ต	socket
บล็อกพิเศษ	block special
อักขระพิเศษ	character special
ไบนารีที่รันได้	ที่รันได้
ไฟล์ปกติว่าง	empty
ลิงก์เชิงสัญลักษณ์	symbolic link to
ไลบรารีไฟล์เก็บถาวร <i>ar</i>	เก็บถาวร
รูปแบบ <i>cpio</i> ส่วนขยาย	cpio archive
รูปแบบ <i>tar</i> ส่วนขยาย	tar archive
เชลล์สคริปต์	commands text
ซอร์สภาษา C	c program text
ซอร์ส FORTRAN	fortran program text

หาก *file* ถูกระบุเป็นลิงก์สัญลักษณ์ จะใช้รูปแบบ เอาต์พุตทางเลือกต่อไปนี้:

"%s: %s %s\n", *file*, *type*, *contents of link*"

หากไฟล์ระบุชื่อโดยตัวถูกดำเนินการ *file* ไม่มีอยู่หรือไม่สามารถอ่านได้ สตริง *cannot open* จะถูกรวมเป็นส่วนหนึ่งของฟิลด์ *type* แต่ค่านี้จะ ไม่ถูกพิจารณาว่าเป็นข้อผิดพลาดที่มีผลต่อสถานะการออกจากการทำงาน หากประเภทของ ไฟล์ที่ระบุชื่อโดยตัวถูกดำเนินการ *file* ไม่สามารถระบุได้ สตริง *data* จะถูกรวมเป็นส่วนหนึ่งของฟิลด์ *type* แต่ค่านี้จะไม่ถูกพิจารณาว่าเป็นข้อผิดพลาดที่มีผลต่อ สถานะการออกจากการทำงาน

## แฟล็ก

ไอเท็ม	คำอธิบาย
-c	ตรวจสอบเมจิกไฟล์ที่ระบุ (ไฟล์ <i>/etc/magic</i> โดยค่าดีฟอลต์) เพื่อหาข้อผิดพลาดการจัดรูปแบบ การตรวจสอบความถูกต้องนี้ไม่ใช่ การทำตามปกติ การกำหนดประเภทไฟล์ไม่ได้ทำภายใต้การใช้แฟล็กนี้
-d	นำใช้การทดสอบระบบดีฟอลต์กับไฟล์
-f <i>FileList</i>	อ่านรายการไฟล์ที่ระบุ ไฟล์ต้องแสดงรายการหนึ่งไฟล์ต่อหนึ่งบรรทัด และต้องนำหน้าหรือลงท้ายด้วยช่องว่าง
-h	เมื่อพบลิงก์สัญลักษณ์ จะระบุไฟล์เป็นลิงก์ สัญลักษณ์ หากไม่ระบุแฟล็ก -h และ <i>file</i> เป็นลิงก์สัญลักษณ์ที่อ้างถึงไฟล์ที่ไม่มีอยู่ <i>file</i> จะระบุไฟล์เป็นลิงก์สัญลักษณ์ เหมือนกับที่แฟล็ก -h ระบุ
-i	หากไฟล์เป็นไฟล์ปกติ ไม่ต้องพยายามจัดหมวดหมู่ประเภท ของไฟล์อีกต่อไป แต่ให้ระบุไฟล์ดังที่ระบุใน “คำอธิบาย” ในหน้า 549
-m <i>MagicFile</i>	ระบุชื่อไฟล์ของเมจิกไฟล์ (ไฟล์ <i>/etc/magic</i> โดยค่าดีฟอลต์)
-M <i>MagicFile</i>	ระบุชื่อของไฟล์ที่มีการทดสอบที่จะถูกใช้กับ ไฟล์เพื่อจัดหมวดหมู่ไฟล์ จะไม่มีการนำใช้การทดสอบระบบดีฟอลต์

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย  
0 ดำเนินการเป็นผลสำเร็จ  
>0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. ในการแสดงประเภทของข้อมูลที่ไฟล์มีให้ป้อน:

```
file myfile
```

ค่านี้แสดง ประเภทไฟล์ของ *myfile* (เช่น ไตเร็กทอรี, ข้อมูล, ข้อความ ASCII, ซอร์สโปรแกรม C และไฟล์เก็บถาวร)

2. ในการแสดงประเภทของแต่ละไฟล์ที่ระบุชื่อในรายการ ชื่อไฟล์ให้ป้อน:

```
file -f filenames
```

ค่านี้แสดงประเภทของแต่ละไฟล์ที่ระบุชื่อใน รายการ filenames ชื่อไฟล์แต่ละชื่อต้องปรากฏ เพียงลำพังบนหนึ่งบรรทัด

หมายเหตุ: ในการรับข้อความที่กำหนดเองจากคำสั่ง `file` ให้ใช้เมจิกไฟล์ต่างหากกับอ็อปชัน `-m` ไม่แนะนำให้แก้ไขไฟล์ `/etc/magic` ที่เป็นแบบอ่านอย่างเดียว

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/file</code>	มีคำสั่ง <code>file</code>
<code>/etc/magic</code>	มีฐานข้อมูลประเภทไฟล์

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `find`” ในหน้า 568

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `ld`

คำสั่ง `Files`

โหมตการเข้าถึงไฟล์และไตเร็กทอรี

---

## คำสั่ง `filemon`

### Purpose

มอนิเตอร์ผลการทำงานของระบบไฟล์ และรายงานกิจกรรม I/O ในชื่อของ โลจิคัลไฟล์ เซกเมนต์หน่วยความจำเสมือน โลจิคัลวอลุ่ม และฟิสิคัลวอลุ่ม

## Syntax

```
filemon [-d] [-i Trace_File -n Gensyms_File] [-o File] [-O Levels] [-w] [-I count:interval] [-P] [-T n] [-u] [-v] [-@ WparList|ALL] [-r RootString] [-A -x User_Command]
```

## รายละเอียด

คำสั่ง **filemon** มอนิเตอร์ การติดตามระบบไฟล์และเหตุการณ์ระบบ I/O และรายงานเกี่ยวกับ ไฟล์ และผลการทำงานการเข้าถึง I/O ระหว่างช่วงเวลานั้น

ในโหมดปกติ คำสั่ง **filemon** รันในแบบเบื้องหลังขณะที่มีอย่างน้อยหนึ่งแอ็พพลิเคชัน โปรแกรม หรือคำสั่งระบบกำลังถูกรัน และมอนิเตอร์ คำสั่ง **filemon** เริ่มทำงานและมอนิเตอร์การติดตามระบบไฟล์ ของโปรแกรมและเหตุการณ์ I/O ณ เวลาจริงโดยอัตโนมัติ โดยดีฟอลต์ การติดตาม เริ่มทำงานในทันที หรืออาจเลือกให้การติดตามถูกหน่วงเวลาออกไปจนกระทั่งคุณ เรียกใช้ คำสั่ง **trcon** คุณสามารถเรียกใช้คำสั่ง **trcoff** และ **trcon** ขณะที่คำสั่ง **filemon** กำลังรันเพื่อ ปิดและเปิดการมอนิเตอร์ได้ตามต้องการ เมื่อ การตามรอยถูกหยุดโดยคำสั่ง **trcstop** คำสั่ง **filemon** จะสร้างรายงานกิจกรรม I/O และจบการทำงาน

คำสั่ง **filemon** ยังสามารถประมวลผลไฟล์การติดตามที่ถูกเรียกคอร์ด ก่อนหน้านั้นโดยใช้โปรแกรมอำนวยความสะดวกการติดตาม รายงานกิจกรรมไฟล์และ I/O อิง ตามเหตุการณ์ที่เรียกคอร์ดในไฟล์นั้น

เมื่อต้องการระบุความเข้าใจโดยสมบูรณ์ของประสิทธิภาพการทำงานของระบบไฟล์สำหรับ แอ็พพลิเคชัน คำสั่ง **filemon** จะมอนิเตอร์กิจกรรมไฟล์และ I/O ในสี่ระดับ:

ไอเท็ม	คำอธิบาย
ระบบไฟล์โลจิคัล	คำสั่ง <b>filemon</b> มอนิเตอร์การดำเนินการ I/O โลจิคัลเกี่ยวกับโลจิคัลไฟล์ การดำเนินการที่มอนิเตอร์ ประกอบด้วยคำสั่งเรียกใช้ระบบ <b>read</b> , <b>write</b> , <b>open</b> และ <b>lseek</b> ทั้งหมด ซึ่งอาจจะ หรืออาจจะไม่ ส่งผลต่อ I/O ฟิสิคัลจริง ทั้งนี้ขึ้นอยู่กับว่าไฟล์ ถูกบัฟเฟอร์ไว้ในหน่วยความจำเรียบร้อยแล้วหรือไม่ สถิติ I/O ถูก เก็บไว้ในรูปแบบต่อไฟล์ การเรียกเพื่ออะซิงโครนัสการเรียกระบบ I/O ไม่ถูกมอนิเตอร์โดย คำสั่ง <b>filemon</b> ดังนั้นรายงานโลจิคัลไฟล์ <b>filemon</b> ไม่รวมการร้องขอ asynchronous I/O (AIO)
ระบบหน่วยความจำเสมือน	คำสั่ง <b>filemon</b> มอนิเตอร์การดำเนินการฟิสิคัล I/O (การเพจ) ระหว่างเซกเมนต์และอิมเมจ บนดิสก์ สถิติ I/O ถูกเก็บไว้ในรูปแบบต่อเซกเมนต์
โลจิคัลวอลุ่ม	คำสั่ง <b>filemon</b> มอนิเตอร์การดำเนินการ I/O บนโลจิคัลวอลุ่ม สถิติ I/O ถูก เก็บไว้ในรูปแบบต่อโลจิคัลวอลุ่ม
ฟิสิคัลวอลุ่ม	คำสั่ง <b>filemon</b> มอนิเตอร์การดำเนินการ I/O บนฟิสิคัลวอลุ่ม ที่ระดับนี้ มีการรับการใช้ฟิสิคัล รีซอร์ส สถิติ I/O ถูกเก็บไว้ในรูปแบบต่อฟิสิคัลวอลุ่ม

คุณสามารถมอนิเตอร์โดยใช้ การรวมกันในระดับ ตามที่ระบุโดยแฟล็กบรทัดรับคำสั่ง โดยดีฟอลต์คำสั่ง **filemon** มอนิเตอร์การดำเนินการ I/O เท่านั้นที่ระดับ หน่วยความจำเสมือน โลจิคัลวอลุ่ม และฟิสิคัลวอลุ่ม ระดับเหล่านี้ทั้งหมดสัมพันธ์กับการร้องขอ ดิสก์ I/O จริง

คำสั่ง **filemon** ยังสร้างรายงานล่าสุดเกี่ยวกับไฟล์ โลจิคัลวอลุ่ม และฟิสิคัลวอลุ่ม รายงานแบบ hotness ยังสามารถสร้างได้โดยใช้อ็อปชันฮ็อต -O รายงานนี้สนับสนุนเฉพาะในโหมดออฟไลน์ แบบอัตโนมัติและโหมดออฟไลน์แบบแมนวล รายงานล่าสุดมีค่าสถิติสำหรับการดำเนินการ I/O ของไฟล์ โลจิคัลวอลุ่ม และฟิสิคัลวอลุ่ม รายงานนี้ ช่วยคุณพิจารณาว่าไฟล์หรือโลจิคัลวอลุ่มใดที่จะย้ายไปยังไดรฟ์ใด โดยมีคุณสมบัติ I/O ต่างกันขึ้นอยู่กับไฟล์/โลจิคัลวอลุ่ม ล่าสุด ข้อมูลล่าสุดถูกพิจารณาจากจำนวนการดำเนินการอ่าน จำนวนไบต์เฉลี่ยของการอ่านต่อหนึ่งการดำเนินการอ่าน จำนวนลำดับการอ่าน และความยาวลำดับเฉลี่ย

คำสั่ง **filemon** เขียน รายงานไปที่เอาต์พุตมาตรฐานหรือไฟล์ที่ระบุ โดยดีฟอลต์รายงาน มีสรุปของกิจกรรม I/O สำหรับแต่ละระดับที่ถูกมอนิเตอร์ รายงานโดยละเอียดถูก พิมพ์ต่อเมื่อเปิดใช้งานแฟล็ก **-O detailed** เนื้อหารายงานสรุปและรายงานรายละเอียด อธิบายอยู่ในส่วน รายงาน

#### Notes:

1. รายงานที่สร้างขึ้นโดยคำสั่ง **filemon** อาจยาวมาก ดังนั้น จะใช้อ็อปชัน **-o** เพื่อเขียนรายงาน ไปยังไฟล์เอาต์พุต เมื่ออุปกรณ์ฟิสิกส์ ถูกเปิดและถูกเข้าถึงโดยตรงโดยแอสพลีเคชัน, บล็อกการอ่านและการเขียนข้อมูลสมบูรณ์ขนาด 512-ไบต์เท่านั้นที่อยู่ในรายงาน การอ่านและเขียน “แบบย่อ” ที่ใช้โดยไดรเวอร์อุปกรณ์ เพื่อเรียกคำสั่งอุปกรณ์และสถานะอุปกรณ์ ถูกละเว้น CD-ROM ไม่มี concentric “แทร็ก” หรือ “ซิลินเดอร์,” เหมือนในฮาร์ดไฟล์ (มีหนึ่งสไปรัลแทร็ก) ดังนั้น จึงไม่สามารถรายงานรายละเอียดระยะไกลสำหรับ CD-ROMs ในแง่ของซิลินเดอร์
2. แฟล็ก **-u** ถูกใช้เพื่อสร้างรายงานบนไฟล์ที่เปิด ก่อนการเริ่มทำงาน **trace daemon** บางส่วนของข้อมูลนี้อาจมีประโยชน์ แต่ส่วนใหญ่ใช้กับ daemons และกิจกรรมที่ไม่เกี่ยวข้องอื่น ข้อมูลแบ็กกราวนด์นี้อาจมีมากเกินไป โดยเฉพาะในระบบขนาดใหญ่ ถ้าไฟล์ **/unix** และเคอร์เนลที่กำลังรันนั้นไม่ เหมือนกัน ดังนั้นเคอร์เนลแอดเดรสไม่ถูกต้อง ทำให้คำสั่ง **filemon** ออกจากการทำงาน เมื่อใช้คำสั่ง **filemon** จากภายใน เซลล์สคริปต์ ดูเนื้อหาของไฟล์เอาต์พุต **filemon** หลังจากช่วงเวลาไประยะหนึ่ง คำสั่ง **filemon** อาจใช้เวลาสักครู่เพื่อสร้างรายงานนี้
3. เมื่อคุณระบุพาสส์เวิร์ดในโปรแกรมประมวลผล I/O เพื่ออ่าน หรือเขียนไฟล์ คำสั่ง **filemon** แปลงพาสส์เวิร์ดนี้เป็นไอดีเรียกทอร์จากตำแหน่งที่คำสั่ง **filemon** ถูกรัน ในกรณีเช่นนั้น รายการกิจกรรม I/O อาจไม่แสดงข้อมูลลวดลุ่ม (i-node) ที่ถูกต้องสำหรับไฟล์นั้น เมื่อต้องการหลีกเลี่ยงปัญหานี้ ให้ใช้พาสส์เวิร์ดในโปรแกรมการประมวลผล I/O ทั้งหมด
4. คำสั่ง **filemon** ไม่สนับสนุนดิสก์โซลิดสเตตไดรฟ์ (SSD) ดังนั้น คำสั่ง **filemon** จะไม่รายงานสถิติของดิสก์ SSD

### ส่วนช่วยเหลือการติดตามระบบ

คำสั่ง **filemon** รับข้อมูลผลการทำงาน raw I/O โดยใช้ส่วนช่วยเหลือการติดตามระบบ ขณะนี้ ส่วนช่วยเหลือการติดตามสนับสนุนเพียงหนึ่งเอาต์พุต stream ดังนั้น มีเพียงหนึ่ง **filemon** หรือกระบวนการติดตาม ที่แอดดที่พีได้ในหนึ่งครั้ง ถ้า **filemon** หรือกระบวนการติดตาม อื่นรันอยู่แล้ว คำสั่ง **filemon** จะตอบสนอง ด้วยข้อความ:

```
/dev/systrace: Device busy
```

ขณะมอนิเตอร์แอสพลีเคชันที่เน้นด้าน I/O คำสั่ง **filemon** อาจไม่สามารถใช้เหตุการณ์การติดตาม ได้เร็วเท่าที่เหตุการณ์ถูกสร้างขึ้น ณ เวลาจริง เมื่อเหตุการณ์นี้เกิดขึ้น ข้อความแสดงความผิดพลาด:

```
Trace kernel buffers overflowed, N missed entries
```

ถูกแสดงบน **stderr** โดระบุ จำนวนเหตุการณ์การติดตามที่หายไปขณะที่บัฟเฟอร์การติดตามเต็ม คำสั่ง **filemon** ยังคงทำการมอนิเตอร์กิจกรรม I/O ต่อ แต่ความถูกต้องของรายงานจะลดลงในระดับที่ไม่ทราบ หนึ่งวิธีที่จะป้องกันโอเวอร์โฟลว์คือมอนิเตอร์ ระดับน้อยกว่าของไฟล์และระบบย่อย I/O: จำนวนของเหตุการณ์การติดตาม ที่สร้างเป็นสัดส่วนกับจำนวนระดับที่มอนิเตอร์ นอกจากนี้ ขนาดบัฟเฟอร์การติดตาม สามารถถูกเพิ่มโดยใช้อ็อปชัน **-T** เพื่อรับข้อมูลของเหตุการณ์การติดตามมากขึ้น ก่อนโอเวอร์โฟลว์ โปรดจำไว้ว่าการเพิ่ม ขนาดบัฟเฟอร์การติดตามจะส่งผลให้ใช้หน่วยความจำที่ทั้งหมดเพิ่มขึ้น ดังนั้นอาจส่งผลต่อการทำงาน I/O และการสลับหน้า

ในสภาวะแวดล้อมที่มีหน่วยความจำจำกัด (มีความ ต้องการหน่วยความจำมากกว่าที่มี) อ็อปชัน **-P** สามารถ ใช้เพื่อตรงข้อความและเพจข้อมูลของกระบวนการ **filemon** ตามเวลาจริง ในหน่วยความจำ เพื่อที่เพจไม่สามารถถูกสับเปลี่ยนออกไป ถ้าไม่ใช้อ็อปชัน **-P** จะเป็นการยอมให้กระบวนการ **filemon** ถูกสลับออก ความคืบหน้าของคำสั่ง **filemon** อาจ ล้าช้าในการที่จะระบุ ว่าไม่สามารถประมวลผลเหตุการณ์การติดตามได้เร็ว เพียงพอ สถานการณ์นี้ทำให้เกิดปัญหาบัฟเฟอร์การติดตามล้นตั้ง

อธิบาย ก่อนหน้านั้น ดังนั้น การปักหยุดกระบวนการนี้จะใช้หน่วยความจำจาก แอ็พพลิเคชัน (แม้ว่าคำสั่ง filemon จะไม่ใช่โปรแกรม ขนาดใหญ่ แต่อิมเพจของกระบวนการก็อาจใช้ถึง 500KB)

แฟล็ก `-i Trace_File` และ `-n Gensyms_File` ให้มีการประมวลผลออฟไลน์โดย filemon ของไฟล์ข้อมูลการติดตามที่สร้างโดยคำสั่ง `trace` ต้องระบุทั้งสอง แฟล็กถ้ามีแฟล็กหนึ่งอยู่ แฟล็กเหล่านี้มีประโยชน์ เมื่อจำเป็นที่จะต้อง postprocess ไฟล์การติดตามจากเครื่องรีโมต หรือดำเนินการติดตามการรวบรวมข้อมูล แล้ว postprocess ข้อมูลนี้ภายหลัง แฟล็กยังมีประโยชน์เมื่อภาวะของระบบสูง และ filemon พลัด hook การติดตาม คุณสามารถใช้แฟล็กเหล่านี้กับโหมดออฟไลน์ที่ทำให้เป็นอัตโนมัติ

แฟล็ก `-r RootString` ปฏิเสธแฟล็ก `-i Trace_File` และแฟล็ก `-n Gensyms_File` แยกจากการใช้แฟล็ก `-r RootString` สำหรับการประมวลผลออฟไลน์, และสามารถถูกใช้กับแฟล็ก `-A` ซึ่งเปิดใช้โหมดออฟไลน์ที่ทำให้เป็นอัตโนมัติ

ไฟล์ `gensyms` (ที่มี ข้อมูลระบบไฟล์) ต้องถูกใช้จากเครื่องที่ให้ข้อมูล การติดตาม และเป็นการดี ที่จะรัน `gensyms` ใกล้เคียงกับเวลาที่ไฟล์การติดตามระบบถูกสร้าง เพื่อที่คอนฟิกูเรชันระบบ จะเหมือนกัน

hooks การติดตามที่สัมพันธ์กับ filemon ต้องถูกรวบรวม โดยคำสั่ง `trace` และถูกระบุโดยแฟล็ก `trace -j Hooks` การติดตามที่สัมพันธ์กัน ถูกแสดงรายการเมื่อ filemon เริ่มทำงานด้วยแฟล็ก `-v` จากนั้นคำสั่ง `gensyms` ที่มีอ็อปชัน `-F` จะรันโดยเอาต์พุตถูกบันทึกใน `Gensyms_File` เพื่อรวบรวมข้อมูลเพิ่มเติมสำหรับ filemon อ็อปชัน `-F` ถูกใช้กับคำสั่ง `gensyms` เพื่อรวบรวมข้อมูลอุปกรณ์สำหรับฟิลิคัลและโลจิคัลวอลุ่ม และยังถูกใช้เพื่อรับข้อมูลระบบไฟล์เสมือนที่ใช้โดย ออฟไลน์ filemon ดังนั้นไฟล์นี้และ `Gensyms_File` อาจถูกระบุเป็น filemon

## รายงาน

แต่ละรายงานที่สร้างโดยคำสั่ง filemon มีส่วนหัวที่ระบุวันที่ ID เครื่อง และระยะเวลา การมอนิเตอร์เป็นวินาที การใช้งานตัวประมวลผลระหว่าง ช่วงเวลาการมอนิเตอร์จะถูกรายงานเช่นกัน

ต่อมา รายงานสรุปถูกสร้างสำหรับแต่ละ ระดับระบบไฟล์ที่ถูกมอนิเตอร์ โดยดีฟอลต์ รายงานโลจิคัลไฟล์ และหน่วยความจำเสมือน ถูกจำกัดไว้ที่ 20 ไฟล์และเซกเมนต์แอ็คทีฟล่าสุด โดยวัด ตามจำนวนข้อมูลทั้งหมดที่ถ่ายโอน ถ้าแฟล็ก `-v` ถูกระบุ กิจกรรมสำหรับไฟล์และเซกเมนต์ทั้งหมดถูกรายงาน มีหนึ่ง แถวสำหรับแต่ละไฟล์ เซกเมนต์หรือวอลุ่ม ที่รายงาน คอลัมน์ในแต่ละแถวสำหรับสี่รายงานสรุป ถูกอธิบายในรายการ ต่อไปนี้:

### รายงานไฟล์ที่แอ็คทีฟที่สุด

คอลัมน์	คำอธิบาย
#MBS	จำนวนทั้งหมดของเมกะไบต์ที่ถ่ายโอนไปยังหรือจากไฟล์ แถวมีการเรียงลำดับตามฟิลด์นี้ในลำดับจากมากไปน้อย
#opns	จำนวนครั้งที่ไฟล์ถูกเปิดระหว่างระยะเวลาการวัด
#rds	จำนวนการเรียกของระบบเพื่ออ่านข้อมูลกับไฟล์
#wrs	จำนวนการเรียกของระบบเพื่อเขียนข้อมูลกับไฟล์
file	ชื่อไฟล์ (ชื่อพารามิเตอร์อยู่ในรายงานรายละเอียด)
วอลุ่ม:inode	ชื่อของวอลุ่มที่มีไฟล์ และหมายเลข i-node ของไฟล์ ฟิลด์นี้สามารถถูกใช้เพื่อเชื่อมโยงไฟล์กับเซกเมนต์ persistent ที่ตรงกัน แสดงในรายงาน I/O หน่วยความจำเสมือน ฟิลด์นี้อาจว่าง ตัวอย่างเช่น สำหรับไฟล์ชั่วคราวที่สร้างและ ลบระหว่างการเรียกใช้งาน

### รายงานเซกเมนต์ที่แอ็คทีฟที่สุด

ไอเท็มคอลลิชัน	คำอธิบาย
#MBS	จำนวนรวมเป็นเมกะไบต์ที่ถ่ายโอน ไปที่/จาก เซกเมนต์ แลวมมีการเรียงลำดับตามฟิลด์นี้ในลำดับจากมากไปน้อย
#rpgs	จำนวนเพจ 4096-ไบต์ที่อ่านลงในเซกเมนต์จากดิสก์ (นั่น คือ เพจ)
#wpgs	จำนวนเพจ 4096-ไบต์จากเซกเมนต์ไปที่ดิสก์ (เพจ เอาต์)
segid	Internal ID ของเซกเมนต์
segtype	ชนิดของเซกเมนต์: เซกเมนต์ทำงาน เซกเมนต์ persistent (โลคัล ไฟล์) เซกเมนต์ไคลเอ็นต์ (รีโมตไฟล์) เซกเมนต์ตารางเพจ เซกเมนต์ระบบ หรือเซกเมนต์ persistent พิเศษที่มีข้อมูลระบบไฟล์ (log, โดเร็กทอรี root, .inode, .inodemap, .inodex, .inodexmap, .indirect, .diskmap)
วอลุ่ม: inode	สำหรับเซกเมนต์ถาวร ชื่อของวอลุ่มที่มีไฟล์ที่เกี่ยวข้อง และหมายเลข i-node ของไฟล์ ฟิลด์นี้สามารถถูกใช้เพื่อเชื่อมโยงเซกเมนต์ persistent กับไฟล์ที่ตรงกัน ที่แสดงในรายงาน I/O ไฟล์ ฟิลด์นี้ว่างสำหรับเซกเมนต์ non-persistent
	หมายเหตุ: เครื่องมือการวิเคราะห์หน่วยความจำเสมือน, svmon สามารถถูกใช้เพื่อแสดงข้อมูลเพิ่มเติม เกี่ยวกับเซกเมนต์ กำหนด segment ID (segid), ดังนี้:  svmon -S <segid>

### รายงานโลจิคัลวอลุ่มที่แอคทีฟที่สุด

ไอเท็มคอลลิชัน	คำอธิบาย
util	การใช้งานวอลุ่ม (ส่วนของเวลาไม่ว่าง) แลวมมีการเรียงลำดับตามฟิลด์นี้ในลำดับจากมากไปน้อย
#rblk	จำนวนบล็อก 512-ไบต์ที่อ่านจากวอลุ่ม
#wblk	จำนวนบล็อก 512-ไบต์ที่เขียนไปที่วอลุ่ม
KB/sec	ผลการถ่ายโอนรวม, เป็นกิโลไบต์ต่อวินาที
volume	ชื่อวอลุ่ม
คำอธิบาย	เนื้อหาของวอลุ่ม: ชื่อระบบไฟล์ หรือชนิดโลจิคัลวอลุ่ม (paging, jfslog, boot หรือ sysdump) รวมทั้ง ระบุว่า ระบบไฟล์ถูกแบ่งเป็นแฟร็กเมนต์หรือบีบอัด

### รายงานฟิลิคัลวอลุ่มที่แอคทีฟที่สุด

ไอเท็มคอลลิชัน	คำอธิบาย
util	การใช้งานวอลุ่ม (ส่วนของเวลาไม่ว่าง) แลวมมีการเรียงลำดับตามฟิลด์นี้ในลำดับจากมากไปน้อย
#rblk	จำนวนบล็อก 512-ไบต์ที่อ่านจากวอลุ่ม
#wblk	จำนวนบล็อก 512-ไบต์ที่เขียนไปที่วอลุ่ม
KB/sec	ผลวอลุ่มรวม, เป็นกิโลไบต์ต่อวินาที
volume	ชื่อวอลุ่ม
คำอธิบาย	ชนิดวอลุ่ม ตัวอย่าง, 120MB disk, 355MB SCSI หรือ CDRROM SCSI หมายเหตุ: การร้องขอโลจิคัลวอลุ่ม I/O เริ่มต้นก่อน และสิ้นสุดหลังจาก การร้องขอฟิลิคัลวอลุ่ม I/O ด้วย เหตุนั้น การใช้งานโลจิคัลวอลุ่มทั้งหมดจะแสดงสูงกว่า การใช้งานฟิลิคัลวอลุ่มทั้งหมด

### รายงานไฟล์ Process-Wise ที่แอคทีฟที่สุด

ไอเท็มคอลลิชัน	คำอธิบาย
#MBS	จำนวนรวมเป็นเมกะไบต์ที่ถ่ายโอนไปที่หรือมาจากไฟล์ แลวมมีการเรียงลำดับตามฟิลด์นี้ในลำดับจากมากไปน้อย
#opns	จำนวนครั้งที่ไฟล์ถูกเปิดระหว่างระยะเวลาการวัด
#rds	จำนวนการเรียกของระบบเพื่ออ่านข้อมูลกับไฟล์
#wrs	จำนวนการเรียกของระบบเพื่อเขียนข้อมูลกับไฟล์
file	ชื่อไฟล์ (ชื่อพารแบบเต็มอยู่ในรายงานรายละเอียด)
PID	ID ของกระบวนการซึ่งเปิดไฟล์
Process	ชื่อของกระบวนการซึ่งเปิดไฟล์
TID	ID ของ thread ซึ่งเปิดไฟล์

## รายงานไฟล์ Thread-Wise ที่แฉีกที่พีที่สุด

ไอเท็มคอลัมน์	คำอธิบาย
#MBS	จำนวนรวมเป็นเมกะไบต์ที่ถ่ายโอนไปที่หรือมาจากไฟล์ แลวมมีการเรียงลำดับตามฟิลด์นี้ในลำดับจากมากไปน้อย
#opns	จำนวนครั้งที่ไฟล์ถูกเปิดระหว่างระยะเวลาการวัด
#rds	จำนวนการเรียกของระบบเพื่ออ่านข้อมูลกับไฟล์
#wrs	จำนวนการเรียกของระบบเพื่อเขียนข้อมูลกับไฟล์
file	ชื่อไฟล์ (ชื่อพารแบบเต็มอยู่ในรายงานรายละเอียด)
TID	ID ของ thread ซึ่งเปิดไฟล์
Process	ชื่อของกระบวนการซึ่งเปิดไฟล์
PID	ID ของกระบวนการซึ่งเปิดไฟล์

ท้ายที่สุด รายงานรายละเอียดถูกสร้างสำหรับแต่ละ ระดับระบบไฟล์ที่ถูกมอนิเตอร์ โดยดีพอลต์ รายงานโลจิคัลไฟล์ และหน่วยความจำเสมือน ถูกจำกัดไว้ที่ 20 ไฟล์และเซกเมนต์แฉีกที่พีล่าสุด โดยวัดตามจำนวนข้อมูลทั้งหมดที่ถ่ายโอน ถ้าแฟล็ก -v ถูกระบุ ก็กิจกรรมสำหรับไฟล์และเซกเมนต์ทั้งหมดถูกรายงาน มีหนึ่ง รายการสำหรับแต่ละไฟล์ เซกเมนต์หรือวอลุ่ม ที่รายงาน

บางฟิลด์รายงาน ค่าเดียว ฟิลด์ อื่นรายงานสถิติที่กำหนดลักษณะการแจกแจงค่าจำนวนมาก ตัวอย่าง สถิติเวลาตอบสนองถูกเก็บไว้สำหรับการร้องขอการอ่านหรือเขียน ทั้งหมดที่ถูกมอนิเตอร์ เวลาตอบสนอง เฉลี่ย, ต่ำสุด และสูงสุด และความเบี่ยงเบนมาตรฐานของ เวลาตอบสนองถูกรายงาน ค่าความเบี่ยงเบนมาตรฐานจะถูกใช้ เพื่อแสดงจำนวนของเวลาตอบสนองแต่ละครั้งที่เบี่ยงเบนจากค่าเฉลี่ย โดยประมาณ สองในสามของ เวลาตอบสนองที่สุ่มจะอยู่ระหว่าง ค่าเฉลี่ย - ความเบี่ยงเบนมาตรฐาน และ ค่าเฉลี่ย + ความเบี่ยงเบน มาตรฐาน ถ้าการแจกแจงของเวลาตอบสนองถูกแยก ข้ามช่วงขนาดใหญ่ ความเบี่ยงเบนมาตรฐานจะมีค่าขนาดใหญ่เปรียบเทียบกับเวลาตอบสนองเฉลี่ย ที่รายงานรายละเอียดถูกอธิบาย ในรายการต่อไปนี้:

### รายงานสถิติไฟล์แบบละเอียด

ไอเท็มคอลัมน์	คำอธิบาย
FILE	ชื่อของไฟล์ ชื่อพารเต็มที่กำหนดไว้ หากเป็นไปได้
volume	ชื่อของโลจิคัลวอลุ่ม/ระบบไฟล์ที่มีไฟล์
inode	หมายเลข I-node สำหรับไฟล์ภายในระบบไฟล์
เปิด	จำนวนครั้งที่ไฟล์นั้นถูกเปิดขณะที่มอนิเตอร์
จำนวนไบต์ xfrd	จำนวนรวมของไบต์ ที่อ่าน/เขียน ไปที่/มาจาก ไฟล์
การอ่าน	จำนวนของการเรียกการอ่านพร้อมกับไฟล์
ขนาดของการอ่าน (ไบต์)	สถิติขนาดถ่ายโอนการอ่านข้อมูล (avg/min/max/sdev) เป็นไบต์
เวลาที่อ่าน (มิลลิวินาที)	สถิติเวลาตอบสนองการอ่านข้อมูล (avg/min/max/sdev) เป็นมิลลิวินาที
writes	จำนวนการเรียกการเขียนพร้อมกับไฟล์
ขนาดของการเขียน (ไบต์)	สถิติขนาดการถ่ายโอนการเขียนข้อมูล
เวลาที่เขียน (มิลลิวินาที)	สถิติเวลาตอบสนองการเขียน
ค้นหา	จำนวนของการเรียกที่น้อยย Iseek

### รายงานสถิติ VM เซกเมนต์

ไอเท็มคอลัมน์	คำอธิบาย
SEGMENT	segment ID ภายใน
segtype	ชนิดของเนื้อหาของเซกเมนต์
แฟล็กเซกเมนต์	เซกเมนต์แอตทริบิวต์ต่างๆ
volume	สำหรับเซกเมนต์ persistent ชื่อของโลจิคัลวอลุ่มมีไฟล์ที่ตรงกัน
inode	สำหรับเซกเมนต์ persistent หมายเลข i-node สำหรับไฟล์ที่ตรงกัน
การอ่าน	จำนวนเพจ 4096-ไบต์ที่อ่านลงในเซกเมนต์ (นั่นคือ เพจ อิน)
เวลาที่อ่าน (มิลลิวินาที)	สถิติเวลาตอบสนองการอ่านข้อมูล (avg/min/max/sdev) เป็นมิลลิวินาที
ลำดับของการอ่าน	จำนวนของลำดับของการอ่าน ลำดับคือสตริงของเพจที่ถูกอ่าน (เพจอิน) ต่อเนื่องกัน จำนวนลำดับการอ่านคือ ตัวบ่งชี้ของจำนวนการเข้าถึงตามลำดับ
ความยาวของลำดับของการอ่าน	สถิติที่อธิบายระยะเวลาของลำดับการอ่าน ใน เพจ
การบันทึก	จำนวนเพจที่เขียนจากเซกเมนต์ (นั่นคือ เพจเอาต์)
เวลาที่เขียน (มิลลิวินาที)	สถิติเวลาตอบสนองการเขียนข้อมูล
ลำดับของการเขียน	จำนวนของลำดับของการเขียน ลำดับคือสตริงของเพจที่ถูกเขียน (เพจเอาต์) ต่อเนื่องกัน
write seq. lengths	สถิติที่อธิบายระยะเวลาของลำดับการเขียน ใน เพจ

### รายงาน โลจิคัล/ฟิสิคัล วอลุ่มแบบละเอียด

ไอเท็มคอลัมน์	คำอธิบาย
VOLUME	คำอธิบายชื่อวอลุ่ม
คำอธิบาย	คำอธิบายของวอลุ่ม (อธิบายถึงเนื้อหา หากทำงานกับโลจิคัลวอลุ่ม อธิบายถึงชนิด หากทำงานกับฟิสิคัลวอลุ่ม)
การอ่าน	จำนวนของคำร้องขอการอ่านที่สร้างขึ้นพร้อมกัวอลุ่ม
ขนาดของการอ่าน (blks)	สถิติขนาดการถ่ายโอนการอ่านข้อมูล (avg/min/max/sdev) ในหน่วย บล็อก 512-ไบต์
เวลาที่อ่าน (มิลลิวินาที)	สถิติเวลาตอบสนองการอ่านข้อมูล (avg/min/max/sdev) เป็นมิลลิวินาที
ลำดับของการอ่าน	จำนวนของลำดับของการอ่าน ลำดับคือสตริงบล็อก 512-ไบต์ที่ถูกอ่านติดต่อกัน และระบุจำนวน การเข้าถึงตามลำดับ
ความยาวของลำดับของการอ่าน	สถิติที่อธิบายระยะเวลาของลำดับการอ่าน ใน บล็อก
การบันทึก	จำนวนของการร้องขอการเขียนที่ทักัวอลุ่ม
ขนาดของการเขียน (blks)	สถิติขนาดการถ่ายโอนการเขียนข้อมูล
เวลาที่เขียน (มิลลิวินาที)	สถิติเวลาตอบสนองการเขียนข้อมูล
ลำดับของการเขียน	จำนวนของลำดับของการเขียน ลำดับคือสตริงบล็อก 512-ไบต์ ที่ถูกเขียนติดต่อกัน
ความยาวของลำดับของการเขียน	สถิติที่อธิบายระยะเวลาของลำดับการเขียน ใน บล็อก
ค้นหา	จำนวนการค้นหาที่นำหน้าการร้องขอการอ่านหรือการเขียน; และ แสดงเป็นเปอร์เซ็นต์ของการอ่านและการเขียนรวมที่ต้องการ การค้นหา
ระยะทางการค้นหา (blks)	สถิติระยะการค้นหา ในหน่วยบล็อก 512-ไบต์ นอกจาก สถิติปกติ (avg/min/max/sdev), ระยะของการดำเนินการค้นหา เริ่มต้น (ถือว่าบล็อก 0 คือจุดเริ่มต้น) ถูกรายงาน แยก ระยะการค้นหานี้บางครั้งมีขนาดใหญ่ ดังนั้นจะถูกรายงาน แยกต่างหากเพื่อหลีกเลี่ยงการทำให้ค่าสถิติอื่นบิดเบือน
ระยะทางการค้นหา (cyls)	(ฮาร์ดไดรฟ์เท่านั้น) สถิติระยะการค้นหา ในหน่วย ของดิสก์ซิลินเดอร์
เวลาที่อยู่ถัดจากคำร้องขอ	สถิติ (avg/min/max/sdev) ที่อธิบายระยะเวลา เป็นมิลลิวินาที ระหว่างการร้องขอ การอ่านหรือการเขียนติดต่อกันไปที่ วอลุ่ม คอลัมน์นี้แสดงอัตราซึ่งวอลุ่มถูก เข้าถึง
throughput	ผลวอลุ่มรวม, เป็นกิโลไบต์ต่อวินาที
utilization	ส่วนของเวลาที่วอลุ่มไม่วาง รายการในรายงานนี้ ถูกเรียงลำดับตามฟิลด์นี้ ตามลำดับที่ลดลง

### รายงานสถิติ Process-wise แบบละเอียด

ไอเท็มคอลลิมน์	คำอธิบาย
ID การประมวลผล	ID ของกระบวนการซึ่งเปิดไฟล์
ชื่อ	ชื่อของไฟล์ที่เปิดรวมถึงพาท
ID เธรด	ID ของ thread ซึ่งเปิดไฟล์
Total Bytes	จำนวนรวมไบต์ที่อ่านหรือเขียน
# ของการค้นหา	จำนวนการค้นหา
# ของการอ่าน	จำนวนการอ่าน
ข้อผิดพลาดเกี่ยวกับการเขียน	จำนวนข้อผิดพลาดการอ่าน
# ของการเขียน	จำนวนการเขียน
ไบต์ที่อ่าน	จำนวนไบต์ที่อ่าน
	min จำนวนไบต์ต่ำสุดที่อ่านในหนึ่งครั้ง
	avr จำนวนไบต์เฉลี่ยที่อ่านในหนึ่งครั้ง
ไบต์ที่เขียน	max จำนวนไบต์สูงสุดที่อ่านในหนึ่งครั้ง
	จำนวนไบต์ที่เขียน
	min จำนวนไบต์ต่ำสุดที่เขียนในหนึ่งครั้ง
	avr จำนวนไบต์เฉลี่ยที่เขียนในหนึ่งครั้ง
เวลาที่อ่าน	max จำนวนไบต์สูงสุดที่เขียนในหนึ่งครั้ง
เวลาที่เขียน	เวลาที่ใช้ในการอ่านข้อมูล
	เวลาที่ใช้ในการเขียนข้อมูล

## รายงานสถิติ Thread-wise แบบละเอียด

ไอเท็มคอลลิมน์	คำอธิบาย
ID เธรด	ID ของ thread ซึ่งเปิดไฟล์
ชื่อ	ชื่อของไฟล์ที่เปิดรวมถึงพาท
ID การประมวลผล	ID ของ thread ซึ่งเปิดไฟล์
Total Bytes	จำนวนรวมไบต์ที่อ่านหรือเขียน
# ของการค้นหา	จำนวนการค้นหา
# ของการอ่าน	จำนวนการอ่าน
ข้อผิดพลาดเกี่ยวกับการเขียน	จำนวนข้อผิดพลาดการอ่าน
# ของการเขียน	จำนวนการเขียน
ไบต์ที่อ่าน	จำนวนไบต์ที่อ่าน
	min จำนวนไบต์ต่ำสุดที่อ่านในหนึ่งครั้ง
	avr จำนวนไบต์เฉลี่ยที่อ่านในหนึ่งครั้ง
ไบต์ที่เขียน	max จำนวนไบต์สูงสุดที่อ่านในหนึ่งครั้ง
	จำนวนไบต์ที่เขียน
	min จำนวนไบต์ต่ำสุดที่เขียนในหนึ่งครั้ง
	avr จำนวนไบต์เฉลี่ยที่เขียนในหนึ่งครั้ง
เวลาที่อ่าน	max จำนวนไบต์สูงสุดที่เขียนในหนึ่งครั้ง
เวลาที่เขียน	เวลาที่ใช้ในการอ่านข้อมูล
	เวลาที่ใช้ในการเขียนข้อมูล

## รูปแบบรายงานที่เรียงลำดับ

ไอเท็ม ID	คำอธิบาย
	กระบวนการ ID ของกระบวนการซึ่งดำเนินการอ่านหรือเขียน
	เธรด ID ของ thread ซึ่งดำเนินการอ่านหรือเขียน
	CPU ID ของ CPU ซึ่งมีการดำเนินการอ่านหรือเขียน
ชนิดธุรกรรม time	ชนิด transaction: SCSI, SSA และอื่นๆ
	เหตุการณ์ <b>bstart</b> เวลาที่เหตุการณ์ <b>bstart</b> สตาร์ท
	เหตุการณ์ <b>iodone</b> เวลาที่การดำเนินการ I/O เสร็จสมบูรณ์
	ระยะเวลา ระยะเวลารวมของการดำเนินการ I/O
อ่าน/เขียน ฟิลิคัลบล็อกแอดเดรส การเข้าถึงรูปแบบ ขนาดฟิลิคัลบล็อก ชื่อหรือแอดเดรสวอลุ่ม	ชนิดของการดำเนินการ: อ่านหรือเขียน ฟิลิคัลบล็อกแอดเดรส ชนิดของการเข้าถึง: pattern, sequential หรือ random ขนาดฟิลิคัลบล็อก
	ฟิลิคัล ชื่อหรือแอดเดรสฟิลิคัลวอลุ่ม
	โลจิคัล ชื่อหรือแอดเดรสโลจิคัลวอลุ่ม
ดัชนีธุรกรรม time	ID เฉพาะเพื่อระบุธุรกรรม
	เหตุการณ์ เวลาซึ่งเหตุการณ์เริ่มต้น
	ขยาย เวลาซึ่งเหตุการณ์ขยายออกไป
ID	กระบวนการ ID ของกระบวนการซึ่งดำเนินธุรกรรม
	เธรด ID ของ thread ซึ่งดำเนินธุรกรรม
	CPU ID ของ CPU ซึ่งดำเนินธุรกรรม
ลำดับขั้นโปรโตคอล name	แสดงการแยกเหตุการณ์
แอดเดรส/จำนวน การเข้าถึงรูปแบบ label values	ชื่ออุปกรณ์ บัฟเฟอร์ หรือบล็อก หรือจำนวนไบต์ แอดเดรสหรือจำนวนไบต์ของอุปกรณ์ บัฟเฟอร์ หรือ บล็อก ประเภทรูปแบบการเข้าถึง: เรียงลำดับหรือแบบสุ่ม ชนิดวอลุ่มหรือแฟล็กการถายโอน ชื่อวอลุ่มหรือคาแฟล็ก

## รายงานแบบ Hotness

รายงาน แบบ hotness ประกอบด้วยสามส่วน: ส่วนข้อมูล, ส่วน ข้อมูลสรุป และส่วนรายงานแบบ hotness ส่วนข้อมูลมี โมเดลระบบ คำสั่ง filemon ที่ใช้และคำสั่ง trace ที่ใช้ ส่วนข้อมูลสรุปมี: จำนวนการดำเนินการอ่าน หรือเขียนทั้งหมด, เวลาทั้งหมดที่ใช้, ข้อมูลทั้งหมดที่อ่านหรือเขียน และการใช้งาน CPU

## รายงานฮ็อตไฟล์

ไอเท็มคอลัมน์	คำอธิบาย
ชื่อ	คำอธิบายชื่อของไฟล์
Size	ขนาดของไฟล์ หน่วยดีฟอลต์คือ MB หน่วยดีฟอลต์ ถูกเขียนทับโดยหน่วยที่ระบุโดยอ็อปชันหน่วย -O
CAP_ACC	ความสามารถที่เข้าถึงได้ คำนี้นี้เป็นค่าเฉพาะที่เข้าถึงใน ไฟล์ หน่วยดีฟอลต์คือ MB หน่วยดีฟอลต์ ถูกเขียนทับโดยหน่วยที่ระบุโดยอ็อปชันหน่วย -O
IOP/#	จำนวนของการดำเนินการ I/O ต่อหน่วยของข้อมูล queuing หน่วยของข้อมูล ที่ใช้จากอ็อปชันหน่วย -O ค่าดีฟอลต์คือ MB ตัวอย่าง ของค่าสำหรับคอลัมน์นี้ได้แก่ 2560/T, 256/G, 0.256/M, 0.000/K ตัวอักษร K, M, G และ T หมายถึง KB, MB, GB และ TB
LV	ชื่อของโลจิคัลวอลุ่มที่เป็นเจ้าของไฟล์ ถ้าข้อมูลนี้ ไม่สามารถระบุได้ จะรายงาน "-"
#ROP	จำนวนทั้งหมดของการดำเนินการอ่านจะเกิดขึ้นบนไฟล์
#WOP	จำนวนทั้งหมดของการดำเนินการเขียนจะเกิดขึ้น บนไฟล์นั้น
B/ROP	<minimum, average, maximum> จำนวนไบต์ ที่อ่านต่อการดำเนินการอ่าน
B/WOP	<minimum, average, maximum> จำนวนไบต์ ที่อ่านต่อการดำเนินการเขียน
RTIME	<minimum, average, maximum> เวลาที่ใช้ต่อ การดำเนินการอ่านเป็นมิลลิวินาที
WTIME	<minimum, average, maximum> เวลาที่ใช้ต่อ การดำเนินการเขียนเป็นมิลลิวินาที
SeqLen	<minimum, average, maximum> ความยาวของ ลำดับการอ่าน
#Seq	จำนวนของลำดับของการอ่าน ลำดับคือสตริงของหน้าขนาด 4K ที่อ่าน (เพจอิน) ต่อเนื่องกัน จำนวนลำดับการอ่านคือ ตัวบ่งชี้ของจำนวนการเข้าถึงตามลำดับ

## รายงานโลจิคัลวอลุ่มแบบฮ็อต

ไอเท็มคอลัมน์	คำอธิบาย
ชื่อ	คำอธิบายชื่อของโลจิคัลไฟล์
Size	ขนาดของโลจิคัลวอลุ่ม หน่วยดีฟอลต์คือ MB หน่วยดีฟอลต์ ถูกเขียนทับโดยหน่วยที่ระบุไว้โดยอ็อปชันหน่วย -O หากคุณไม่สามารถขอรับค่านี้ได้ "-" จะถูกรายงาน
CAP_ACC	ความสามารถที่เข้าถึงได้ คำนี้นี้เป็นค่าเฉพาะที่เข้าถึงใน ไฟล์ หน่วยดีฟอลต์คือ MB หน่วยดีฟอลต์ ถูกเขียนทับโดยหน่วยที่ระบุโดยอ็อปชันหน่วย -O
IOP/#	จำนวนของการดำเนินการ I/O ต่อหน่วยของข้อมูล queuing หน่วยของข้อมูล ที่ใช้จากอ็อปชันหน่วย -O ค่าดีฟอลต์คือ MB ตัวอย่าง ของค่าสำหรับคอลัมน์นี้ได้แก่ 2560/T, 256/G, 0.256/M, 0.000/K ตัวอักษร K, M, G และ T หมายถึง KB, MB, GB และ TB ตามลำดับ
#Files	จำนวนของไฟล์ที่เข้าถึงในโลจิคัลวอลุ่มนี้
#ROP	จำนวนการดำเนินการอ่านทั้งหมดที่เกิดขึ้นบนโลจิคัลวอลุ่ม
#WOP	จำนวนการดำเนินการเขียนทั้งหมดที่เกิดขึ้นบน โลจิคัลวอลุ่มนั้น
B/ROP	<minimum, average, maximum> จำนวนไบต์ ที่อ่านต่อการดำเนินการอ่าน
B/WOP	<minimum, average, maximum> จำนวนไบต์ ที่อ่านต่อการดำเนินการเขียน
RTIME	<minimum, average, maximum> เวลาที่ใช้ต่อ การดำเนินการอ่านเป็นมิลลิวินาที
WTIME	<minimum, average, maximum> เวลาที่ใช้ต่อ การดำเนินการเขียนเป็นมิลลิวินาที
SeqLen	<minimum, average, maximum> ความยาวของ ลำดับการอ่าน
#Seq	จำนวนของลำดับของการอ่าน ลำดับคือสตริงของหน้าขนาด 4K ที่อ่าน (เพจอิน) ต่อเนื่องกัน จำนวนลำดับการอ่านคือ ตัวบ่งชี้ของจำนวนการเข้าถึงตามลำดับ

## รายงานฟิสิคัลวอลุ่มแบบฮ็อต

ไอเท็มคอลัมน์	คำอธิบาย
ชื่อ	ชื่อของฟิลต์ลวลุ่ม
Size	ขนาดของฟิลต์ลวลุ่ม หน่วยดีฟอลต์คือ MB หน่วยดีฟอลต์ ถูกเขียนทับโดยหน่วยที่ระบุโดยซอ์พชันหน่วย -O
CAP_ACC	ความสามารถที่เข้าถึงได้ คำนี้นี้เป็นค่าเฉพาะที่เข้าถึงใน ไฟล์ หน่วยดีฟอลต์คือ MB หน่วยดีฟอลต์ ถูกเขียนทับโดยหน่วยที่ระบุโดยซอ์พชันหน่วย -O
IOP/#	จำนวนของการดำเนินการ I/O ต่อหน่วยของข้อมูลที่เข้าถึง หน่วยของข้อมูล ที่ใช้จากอ็อพชันหน่วย -O ค่าดีฟอลต์คือ MB ตัวอย่างของค่าสำหรับคอลัมน์นี้ได้แก่ 2560/T, 256/G, 0.256/M, 0.000/K ตัวอักษร K, M, G และ T หมายถึง KB, MB, GB และ TB ตามลำดับ
#ROP	จำนวนทั้งหมดของการดำเนินการอ่านที่เกิดขึ้นบนฟิลต์ลวลุ่ม
#WOP	จำนวนทั้งหมดของการดำเนินการเขียนที่เกิดขึ้น บนฟิลต์ลวลุ่มนั้น
B/ROP	<minimum, average, maximum> จำนวนไบต์ ที่อ่านต่อการดำเนินการอ่าน
B/WOP	<minimum, average, maximum> จำนวนไบต์ ที่อ่านต่อการดำเนินการเขียน
RTIME	<minimum, average, maximum> เวลาที่ใช้ต่อ การดำเนินการอ่านเป็นมิลลิวินาที
WTIME	<minimum, average, maximum> เวลาที่ใช้ต่อ การดำเนินการเขียนเป็นมิลลิวินาที
SeqIen	<minimum, average, maximum> ความยาวของ ลำดับการอ่าน
#Seq	จำนวนของลำดับของการอ่าน ลำดับคือสตริงที่มีขนาด 512 ไบต์ต่อบล็อก ซึ่งจะถูกอ่านอย่างต่อเนื่อง จำนวนลำดับการอ่านคือตัวบ่งชี้ของจำนวนการเข้าถึงตามลำดับ

แต่ละรายงานแบบ hotness ที่อธิบายถูกรายงานซ้ำ หลายครั้งตามฟิลต์การเรียงลำดับ

รายงานแบบ hotness อื่นๆ ที่อ้างอิงตามฟิลต์การเรียงลำดับอื่นๆ คือ:

1. รายงานแบบ hotness เรียงลำดับตามคีย์หลัก
2. รายงานแบบ hotness เรียงลำดับตาม CAP\_ACC
3. รายงานแบบ hotness เรียงลำดับตาม IOP/#
4. รายงานแบบ hotness เรียงลำดับตาม #ROP
5. รายงานแบบ hotness เรียงลำดับตาม #WOP
6. รายงานแบบ hotness เรียงลำดับตาม RTIME
7. รายงานแบบ hotness เรียงลำดับตาม WTIME

แต่ละรายงานที่เรียงลำดับจากมากไปน้อยของฟิลต์ การเรียงลำดับที่สอดคล้องกัน

ถ้าคุณระบุอ็อพชัน -O hot=r ดังนั้นเฉพาะรายงานของการดำเนินการอ่าน และรายงานตาม ปัจจัยหลักเท่านั้นที่ถูกสร้าง หมายเลขรายงาน 1, 4 และ 6 จะถูก สร้าง

ถ้าผู้ใช้ระบุอ็อพชัน -O hot=w ดังนั้น เฉพาะรายงานของการดำเนินการเขียน และรายงานตามปัจจัยหลัก เท่านั้นที่ถูกสร้าง นั่นคือหมายเลขรายงาน 1, 5 และ 7 ถูกสร้าง

คีย์หลักถูกกำหนดไว้โดยค่าของคอลัมน์ต่อไปนี้: #ROP, B/ROP, SeqIen และ #Seq

## แฟล็ก

## ไอเอ็ม

`-i Trace_File`

### คำอธิบาย

อ่านข้อมูลการติดตาม I/O จาก *Trace\_File* ที่ระบุ แทนการอ่านจากกระบวนการติดตาม ณ เวลาจริง รายงาน **filemon** สรุป กิจกรรม I/O สำหรับระบบและระยะเวลาที่แสดง โดยไฟล์การติดตาม อ็อพชันนี้ถูกปฏิเสธ ใช้แฟล็ก `-r RootString` แทน

สำหรับ รายงานเพื่อให้มีความถูกต้อง ไฟล์การติดตามต้องมี hooks ทั้งหมดที่จำเป็นโดยคำสั่ง **filemon**

`-n Gensyms_File`

### อ็อพชัน -n ต้องถูกระบุ

ระบุ *Gensyms\_File* สำหรับการประมวลผล การติดตามออฟไลน์ ไฟล์นี้ถูกสร้างโดยการรันคำสั่ง **gensyms** พร้อมกับอ็อพชัน `-f` และเปลี่ยนทิศทางเอาต์พุต ไปที่ไฟล์ ดังนี้:

```
gensyms -F > file
```

### อ็อพชัน -i ต้องถูกระบุ

`-o File`

`-d`

`-T n`

แฟล็ก `-n` ถูก ปฏิเสธ ใช้แฟล็ก `-r RootString` แทน

เขียนรายงานกิจกรรม I/O ไปที่ *File*, ที่ระบุ แทนการเขียนไปที่ไฟล์ **stdout**

เริ่มทำงานคำสั่ง **filemon** แต่เลื่อนเวลาการติดตามไปจนกระทั่ง คำสั่ง **trcon** ถูกรันโดยผู้ใช้ โดยดีฟอลต์ การติดตามถูกเริ่มในทันที

ตั้งค่าขนาดบัฟเฟอร์การติดตามของเคอร์เนลเป็น *n* ไบต์ ขนาดดีฟอลต์คือ 64 000 ไบต์ต่อ CPU ขนาดบัฟเฟอร์สามารถเพิ่มขึ้นได้เพื่อให้เหมาะสมกับ การส่งข้อมูลเป็นชุดที่มีขนาดใหญ่กว่า หากมี (ขนาดเรกคอร์ด เหตุการณ์ทั่วไปคือ 30 ไบต์)

หมายเหตุ: ไตรเวอร์ การติดตามในเคอร์เนลใช้การบัฟเฟอร์คู่ ดังนั้นในความเป็นจริงจะมีสอง บัฟเฟอร์ที่จัดสรรให้มีขนาด *n* ไบต์ และ หมายเหตุ ว่าบัฟเฟอร์เหล่านี้ถูกตรึงในหน่วยความจำ ดังนั้นจึงไม่มีการนำมา ทำการเพจ บัฟเฟอร์ขนาดใหญ่ อาจส่งผลกระทบต่อประสิทธิภาพการสลับหน้า และ I/O อื่นๆ

`-P`

ปักกระบวนการมอนิเตอร์ในหน่วยความจำ แฟล็ก `-P` ทำให้หน้าตัวอักษรและข้อความคำสั่ง **filemon** ถูกปิดหมด ในหน่วยความจำสำหรับช่วงเวลาระหว่างการมอนิเตอร์ แฟล็กนี้สามารถถูกใช้เพื่อประกันว่ากระบวนการ **filemon** ณ เวลาจริง ไม่ถูกเพจเอาต์ เมื่อรันในสภาวะแวดล้อมที่มีหน่วยความจำจำกัด

`-v`

พิมพ์ข้อมูลพิเศษในรายงาน ผลที่ชัดเจนที่สุดของ แฟล็ก `-v` คือ โลจิคัลไฟล์ทั้งหมด และเชกเมนต์ทั้งหมดที่ถูกเข้าถึงถูกรวมอยู่ในรายงานกิจกรรม I/O แทนการรายงานเพียงไฟล์และเชกเมนต์ที่แอ็คทีฟที่สุด 20 รายการเท่านั้น เปิดโหมดออฟไลน์ที่ทำให้เป็นอัตโนมัติ คุณต้องใช้ แฟล็ก `-x` คู่กับแฟล็ก `-A` ที่การติดตามถูก รวบรวมไวจนกระทั่งคำสั่งที่ผู้ใช้ระบุเสร็จสิ้นการเรียกใช้งาน ตัวอย่างของคำสั่งผู้ใช้ทั่วไปคือ `sleep 10`

`-A -x User_Command`

`-r RootString`

ถ้าคุณรวมแฟล็กนี้กับแฟล็ก `-A` คำสั่ง **filemon** เก็บข้อมูลการติดตามใน ไฟล์ **RootString.trc** และสร้างไฟล์ **gensyms** และเก็บไว้ในไฟล์ **RootString.syms** เมื่ออ็อพชันนี้ถูกเปิดใช้โดยไม่มีแฟล็ก `-A` คำสั่ง **filemon** ทำการประมวลผลภายหลังไฟล์ **RootString.trc** และ **RootString.syms** เพื่อสร้างรายงานออฟไลน์ อ็อพชันนี้ไม่ยอมรับแฟล็ก `-n` และ `-i` ที่มีอยู่ คำสั่ง **filemon** ดำเนินต่อเพื่อสนับสนุนแฟล็ก `-i` และแฟล็ก `-n` สำหรับ ความเข้ากันได้ทางไบนารี

คำอธิบาย

มอนิเตอร์เฉพาะระดับระบบไฟล์ที่ระบุ อีอพชั่นที่ค้นด้วยจุลภาค ที่ถูกต้องคือ:

abbreviated

สร้างรายการธุรกรรมในรูปแบบย่อ หนึ่งบรรทัด ต่อธุรกรรม (แทนที่พวล "subpar" แบบเก่า) อีอพชั่นนี้สนับสนุนเฉพาะในโหมดออฟไลน์และไม่สามารถถูกรวมกับอีอพชั่น -O อื่น

**collated** สร้างรายการธุรกรรมในรูปแบบที่มีการเรียงลำดับ: เหตุการณ์ ถูกรวบรวมร่วมกันต่อธุรกรรม อีอพชั่นนี้สนับสนุนเฉพาะในโหมดออฟไลน์และไม่สามารถถูกรวมกับอีอพชั่น -O อื่น

**detailed** รายงานรายละเอียดถูกสร้างพร้อมกับโหมดสรุปสถิติ และไม่สามารถถูกรวมกับอีอพชั่น abbreviated หรืออีอพชั่น collated

**If=num** แสดงเฉพาะจำนวนรายการโลจิคัลไฟล์ที่ระบุและ ไม่สามารถรวมกับแฟล็ก -O abbreviated หรือแฟล็ก -O collated ถ้าอาร์กิวเมนต์ num ไม่ถูกระบุ จะแสดงรายการ ทั้งหมด

**vm=num** แสดงเฉพาะจำนวนรายการหน่วยความจำเสมือนที่ระบุและ ไม่สามารถรวมกับแฟล็ก -O abbreviated หรือแฟล็ก -O collated ถ้าอาร์กิวเมนต์ num ไม่ถูกระบุ จะแสดงรายการ ทั้งหมด

**lv=num** แสดงเฉพาะจำนวนรายการโลจิคัลลุ่มที่ระบุและ ไม่สามารถรวมกับแฟล็ก -O abbreviated หรือแฟล็ก -O collated ถ้าอาร์กิวเมนต์ num ไม่ถูกระบุ จะแสดงรายการ ทั้งหมด

**pv=num** แสดงเฉพาะจำนวนรายการฟิสิคัลลุ่มที่ระบุและ ไม่สามารถรวมกับแฟล็ก -O abbreviated หรือแฟล็ก -O collated ถ้าอาร์กิวเมนต์ num ไม่ถูกระบุ จะแสดงรายการ ทั้งหมด

**hot=r|w** สร้างรายงานแบบ hotness หากระบุ hot=r แล้ว รายงานแบบ hotness ที่อ้างอิงตามการดำเนินการอ่านเท่านั้นที่ถูกสร้าง หากระบุ hot=w รายงานแบบ hotness ที่อ้างอิง ตามการดำเนินการเขียนเท่านั้นที่ถูกสร้าง

**sz=num** ระบุขนาดสูงสุดของไฟล์ที่เข้าถึงซึ่งต้องถูกรายงาน ในรายงานแบบ hotness หน่วยสำหรับค่านี้ถูกระบุไว้โดยอีอพชั่นหน่วย -O หน่วยดีฟอลต์คือ MB unit={KB|MB|GB|TB} ระบุ หน่วยที่จะใช้กับอีอพชั่น sz และหน่วยที่จะใช้ กับ CAP\_ACC และฟิลด์ขนาด ในรายงานแบบ hotness

**th=num** แสดงเฉพาะจำนวนรายการสถิติ thread ที่ระบุและ ไม่สามารถรวมกับแฟล็ก -O abbreviated หรือแฟล็ก -O collated ถ้าอาร์กิวเมนต์ num ไม่ถูกระบุ จะแสดงรายการ ทั้งหมด

**pr=num** แสดงเฉพาะจำนวนรายการสถิติกระบวนการที่ระบุและ ไม่สามารถรวมกับแฟล็ก -O abbreviated หรือแฟล็ก -O collated ถ้าอาร์กิวเมนต์ num ไม่ถูกระบุ จะแสดงรายการ ทั้งหมด

**all=num** เช็ต If=num, vm=num, lv=num, pv=num, ts=num และ เขียนทับค่าเก่าสำหรับอีอพชั่นของ If, vm, lv, pv, th และ pr อีอพชั่นนี้ไม่สามารถ รวมกับแฟล็ก -O abbreviated หรือ แฟล็ก -O collated ถ้าอาร์กิวเมนต์ num ไม่ถูกระบุ จะแสดงรายการ ทั้งหมด และเป็นดีฟอลต์อีอพชั่น

คำอธิบาย

ระดับ vm, lv และ pv ถูกระบุโดยดีฟอลต์ เมื่อคุณรัน คำสั่ง filemon -O ในโกลบอล WPAR โดยไม่มีแฟล็ก -@ ระดับ If ถูกใช้โดยนัยโดย ดีฟอลต์เมื่อคุณรันคำสั่ง filemon -O ใน WPAR หรือเมื่อ คุณใช้แฟล็ก -@

ถ้าไม่ระบุอาร์กิวเมนต์ num ดีฟอลต์จะแสดง รายการของส่วนนั้นทั้งหมด อาร์กิวเมนต์ num ไม่สนับสนุนในรูปแบบ abbreviated และ collated และมีการสนับสนุนเฉพาะในข้อสรุป สถิติ ถ้ามีการระบุแฟล็ก -O detailed รายงาน อยู่ในรูปแบบสรุปสถิติ ตามด้วยรายงานรายละเอียด ในทั้งโหมดออนไลน์และออฟไลน์ โหมดดีฟอลต์ของการดำเนินการสำหรับ คำสั่ง filemon ถูกเปลี่ยนจากรายงานสรุป และรายงานสถิติแบบละเอียดเป็นรายงานสรุปเฉพาะ สถิติ ถ้าคำสั่ง filemon ถูกเรียกโดยไม่มี อีอพชั่นหรือมีเพียงแฟล็ก -O พร้อมกับหนึ่งใน อีอพชั่น If, vm, lv, pv, pr, th หรือ all เฉพาะรายงานสรุป เท่านั้นที่ถูกแสดงนอกจากมีการระบุแฟล็ก -O detailed

## ไอเอ็ม

-u

### คำอธิบาย

รายงานบนไฟล์ที่ถูกเปิดขึ้นก่อนการเริ่มทำงาน **trace daemon process ID (PID)** และ **file descriptor (FD)** ถูกแทนที่สำหรับชื่อไฟล์

หมายเหตุ: เนื่องจาก PIDs และ FDs สามารถนำมาใช้ใหม่ได้จึงเป็นไปได้ที่จะเห็นไฟล์ต่างกันในรายงานโดยมีฟิลด์ชื่อเหมือนกัน

-w

-I count:interval

พิมพ์รายงานแบบ hotness ในรูปแบบกว้าง อีพซันนี้ใช้ได้ต่อเมื่อระบุอีพซัน -O hot

ระบุการนับและช่วงที่ต้องใช้ สำหรับการติดตามแบบหลายสแน็ปช็อต หากระบุอีพซันนี้จำนวน การนับ

ของสแน็ปช็อตของการติดตามจะถูกรวบรวมไว้ด้วยช่องว่างของ ช่วง วินาทีระหว่างสองสแน็ปช็อต อีพซันนี้ใช้งานได้เท่านั้นในโหมดออฟไลน์แบบอัตโนมัติ ด้วยการระบุอีพซันฮ็อต -O

-@[WparList|ALL]

รายงานถูกจำกัดกับรายการของ WPARs ที่ส่งผ่านโดยอาร์กิวเมนต์

## ตัวอย่าง

1. เมื่อต้องการมอนิเตอร์กิจกรรมไฟล์ I/O ของ ระดับ หน่วยความจำเสมือน โลจิคัลวอลุ่มและไฟล์คัลวอลุ่มของ ระบบไฟล์ให้ป้อน:

```
filemon
```

คำสั่ง **filemon** เริ่มการติดตามระบบโดยอัตโนมัติและนำตัวเองไปไว้ในแบ็กกราวนด์ หลังจากคำสั่งนี้ให้ป้อนแอ็พพลิเคชันโปรแกรมและคำสั่งระบบ ที่จะถูกรันในเวลานี้แล้วป้อน:

```
trcstop
```

หลังจากเรียกใช้คำสั่ง **trcstop** รายงานกิจกรรม I/O จะถูกแสดงบนเอาต์พุตมาตรฐาน (แต่อาจปิดการเลื่อนหน้าจอ) รายงาน I/O หน่วยความจำเสมือนถูกจำกัดไว้ที่ 20 เซกเมนต์เปลี่ยนแปลงไม่ได้ใน I/O ส่วนใหญ่

2. เมื่อต้องการมอนิเตอร์กิจกรรมที่ระดับระบบไฟล์ ทั้งหมด และเขียนรายงานไปที่ไฟล์ **fmon.out** ให้ป้อน:

```
filemon -o fmon.out -0 all
```

คำสั่ง **filemon** เริ่มการติดตามระบบโดยอัตโนมัติและนำตัวเองไปไว้ในแบ็กกราวนด์ หลังจากคำสั่งนี้ให้ป้อนแอ็พพลิเคชันโปรแกรมและคำสั่งระบบ ที่จะถูกรันในเวลานี้แล้วป้อน:

```
trcstop
```

หลังจากคำสั่ง **trcstop** ถูกเรียกรายงานกิจกรรม I/O ถูกเขียนไปที่ไฟล์ **fmon.out** ทั้ง สี่ระดับของระบบไฟล์และ I/O (ระดับโลจิคัลไฟล์, หน่วยความจำ เสมือน, โลจิคัลวอลุ่ม และไฟล์คัลวอลุ่ม) ถูกมอนิเตอร์ รายงานโลจิคัลไฟล์และ I/O หน่วยความจำเสมือนถูกจำกัดที่ 20 ไฟล์และเซกเมนต์ (ตามลำดับ) ที่เปลี่ยนแปลงไม่ได้ใน I/O ส่วนใหญ่

3. เมื่อต้องการมอนิเตอร์กิจกรรมที่ระดับระบบไฟล์ ทั้งหมด และเขียนรายงาน verbose ไปที่ไฟล์ **fmon.out** ให้ป้อน:

```
filemon -v -o fmon.out -0 all
```

คำสั่ง **filemon** เริ่มการติดตามระบบโดยอัตโนมัติและนำตัวเองไปไว้ในแบ็กกราวนด์ หลังจากคำสั่งนี้ให้ป้อนแอ็พพลิเคชันโปรแกรมและคำสั่งระบบ ที่จะถูกรันในเวลานี้แล้วป้อน:

```
trcstop
```

ตัวอย่างนี้ เหมือนกับตัวอย่างก่อนหน้านี้ ยกเว้นรายงาน verbose ถูกสร้าง บนไฟล์ **fmon.out** ความแตกต่างหลักคือคำสั่ง **filemon** จะระบุขั้นตอนที่ใช้เพื่อเริ่มต้นการติดตาม และรายงานสรุปและ รายงานโดยละเอียดประกอบด้วยไฟล์และเซกเมนต์ทั้งหมดที่เปลี่ยนแปลง I/O ใดๆ ไม่ได้ (อาจมีจำนวนมาก) แทนการเก็บเพียง 20 อันดับสูงสุด

4. เมื่อต้องการรายงานกิจกรรม I/O ที่เก็บข้อมูลโดยเซสชัน การติดตามที่บันทึกก่อนหน้านี้ให้ป้อน:

```
filemon -i trcfile | pg
```

ในตัวอย่างนี้ คำสั่ง `filemon` อ่านเหตุการณ์การติดตามระบบไฟล์จากอินพุตไฟล์ `trcfile` เนื่องจากข้อมูลการติดตาม ถูกบันทึกไว้แล้วบนไฟล์ คำสั่ง `filemon` จึงไม่วางตนเองในแบบเบื้องหลัง เพื่อให้แอฟพลิเคชันโปรแกรมรัน หลังจากไฟล์ ทั้งไฟล์ถูกอ่าน รายงานกิจกรรม I/O สำหรับ ระดับ หน่วยความจำเสมือน โลจิคัลวอลุ่มและฟิสิคัลวอลุ่ม จะ ถูกแสดงบน เอาต์พุตมาตรฐาน (ซึ่งในตัวอย่างนี้ถูกโพพ ไปที่ `pg`)

5. เมื่อต้องการมอนิเตอร์กิจกรรม I/O สำหรับโลจิคัลและ ฟิสิคัลวอลุ่มเท่านั้น ขณะควบคุมระยะเวลาที่มอนิเตอร์ โดยใช้คำสั่ง `trcon` และ `trcoff` ให้ป้อน:

```
filemon -d -o fmon.out -O pv,lv
```

คำสั่ง `filemon` เริ่มการติดตามระบบโดยอัตโนมัติและนำตัวเองไปไว้ในแบ็กกราวนด์ หลังจากคำสั่งนี้ คุณสามารถป้อนแอฟพลิเคชันโปรแกรมที่ไม่ได้มอนิเตอร์และคำสั่งระบบ ที่จะถูกรันในเวลานี้แล้วป้อน:

```
trcon
```

หลังจากคำสั่งนี้ คุณสามารถป้อนแอฟพลิเคชันโปรแกรมที่มอนิเตอร์และคำสั่งระบบ ที่จะถูกรันในเวลานี้แล้วป้อน:

```
trcoff
```

หลังจากคำสั่งนี้ คุณสามารถป้อนแอฟพลิเคชันโปรแกรมที่ไม่ได้มอนิเตอร์และคำสั่งระบบ ที่จะถูกรันในเวลานี้แล้วป้อน:

```
trcon
```

หลังจากคำสั่งนี้ คุณสามารถป้อนแอฟพลิเคชันโปรแกรมที่มอนิเตอร์และคำสั่งระบบ ที่จะถูกรันในเวลานี้แล้วป้อน:

```
trcstop
```

ใน ตัวอย่างนี้แฟล็ก `-O` ถูกใช้เพื่อจำกัด การมอนิเตอร์กับโลจิคัลและฟิสิคัลวอลุ่มเท่านั้น เฉพาะเหตุการณ์การติดตาม เหล่านั้น ที่สัมพันธ์กับโลจิคัลและฟิสิคัลวอลุ่มที่จะถูกเปิดใช้งาน และจากผลของการใช้แฟล็ก `-d` การมอนิเตอร์ ถูกเลื่อนไปเมื่อเริ่มต้น จนกว่าคำสั่ง `trcon` จะถูกเรียก การติดตามระบบสามารถถูกปิดใช้งานและเปิดใช้อีกครั้ง อย่างไม่ต่อเนื่อง โดยใช้คำสั่ง `trcoff` และ `trcon` เพื่อที่เฉพาะระยะเวลาที่จะเจาะจงเท่านั้นที่จะถูกมอนิเตอร์

6. เมื่อต้องการรัน `filemon` ในโหมด ออฟไลน์ ให้รันคำสั่ง `trace` และ `gensyms` แยกกัน จากนั้นใช้เอาต์พุตจากคำสั่งเหล่านั้น เป็นอินพุตกับคำสั่ง `filemon` ดังนี้:

```
trace -a -T 768000 -L 10000000 -o trace.out -j 000,000,001,002,003,005,006,139,102,10C,106,00A,107,101,104,10D,15B,12E,130,163,19C,154,3D3,1BA,1BE,1BC,10B,221,1C9,222,228,232,45B
```

รัน แอฟพลิเคชันโปรแกรมที่มอนิเตอร์และคำสั่งระบบ แล้วป้อน:

```
trcstop
```

สร้าง ไฟล์ `gensyms` :

```
gensyms -F > gensyms.out
```

จากนั้น รัน `filemon` ด้วยทั้งแฟล็ก `-i` และ `-n` :

```
filemon -i trace.out -n gensyms.out -O all
```

7. หากต้องการสร้างรายงานแบบ `hotmess` ในโหมดออฟไลน์แบบอัตโนมัติ ด้วยหน่วยของข้อมูลที่เป็นเมกะไบต์ ให้ใช้คำสั่งต่อไปนี้:

```
filemon -O hot,unit=MB -r <rootstring> -A-x "<user command>"
```

8. เมื่อต้องการสร้างรายงานแบบ `hotmess` ที่มีสามสแน็ปช็อตของการติดตามในช่วงเวลา 5 วินาที ให้รันคำสั่งต่อไปนี้:

```
filemon -O hot -r <rootstring> -A-x "<user command>" -I 3:5
```

9. หากต้องการสร้างรายงานแบบ hotness ในโหมดออฟไลน์:

```
filemon -r <rootstring> -O hot
```

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง svmon

คำสั่ง trcrpt

คำสั่ง lseek

การมอนิเตอร์ดิสก์ I/O

---

## คำสั่ง fileplace

### Purpose

แสดงตำแหน่งของไฟล์บล็อกภายใน โลจิคัลหรือฟิสิคัลวอลุ่ม

### Syntax

```
fileplace [{ -l | -p [-o FragOffset] [-n FragNumber] } [-i] [-v]] File | [-m LogicalVolumeName]
```

### รายละเอียด

คำสั่ง **fileplace** แสดงตำแหน่งของไฟล์ที่ระบุภายในโลจิคัลหรือฟิสิคัลวอลุ่ม ที่มีไฟล์อยู่

โดยดีฟอลต์คำสั่ง **fileplace** แสดงช่องของโลจิคัลวอลุ่มแฟรกเมนต์ที่จัดสรรให้กับไฟล์ที่ระบุ ไปที่เอาต์พุตมาตรฐาน ลำดับซึ่งโลจิคัลวอลุ่มแฟรกเมนต์ ถูกแสดง เหมือนกับลำดับในไฟล์ ส่วนหัวย่อแสดงขนาดไฟล์ (เป็นไบต์) ชื่อของโลจิคัลวอลุ่ม ซึ่งมีไฟล์อยู่ ขนาดบล็อก (เป็นไบต์) สำหรับวอลุ่มนั้น ขนาดแฟรกเมนต์เป็นไบต์ และการบีบอัด ระบุว่าระบบไฟล์ถูกบีบอัดหรือไม่

บางครั้ง ส่วนของไฟล์อาจไม่ถูกแมพ กับแฟรกเมนต์ในวอลุ่ม พื้นที่เหล่านี้ ซึ่งขนาดเป็น จำนวนเต็มของแฟรกเมนต์ ซึ่งถูกใส่ข้อมูลศูนย์ทางอ้อม โดยระบบไฟล์ คำสั่ง **fileplace** ระบุ พื้นที่ในไฟล์ที่ไม่มีแฟรกเมนต์จัดสรรไว้

ทางเลือก คำสั่ง **fileplace** ยังแสดง:

- สถิติระบุระดับซึ่งไฟล์ถูกกระจายภายใน วอลุ่ม
- บล็อกแอดเดรสทางอ้อมสำหรับไฟล์
- ตำแหน่งไฟล์บนฟิสิคัล (as opposed to logical) วอลุ่ม สำหรับแต่ละสำเนาฟิสิคัลของไฟล์

### Notes:

1. คำสั่ง **fileplace** ไม่สามารถที่จะแสดง ตำแหน่งของไฟล์ Network File System (NFS) แบบรีโมต ถ้ามีการระบุรีโมตไฟล์ คำสั่ง **fileplace** ส่งคืน ข้อความแสดงความผิดพลาด ใดๆก็ตาม ตำแหน่งของรีโมตไฟล์สามารถถูก แสดง ถ้าคำสั่ง **fileplace** รันโดยตรง บนไฟล์เซิร์ฟเวอร์
2. คำสั่ง **fileplace** อ่านรายการบล็อกของไฟล์ โดยตรงจากโลจิคัลวอลุ่มบนดิสก์ ถ้าไฟล์ถูก สร้างใหม่ ถูกขยาย หรือตัดท้าย ข้อมูลระบบไฟล์ อาจยังไม่มีอยู่บนดิสก์ เมื่อคำสั่ง **fileplace** ถูกรัน ใช้คำสั่ง **sync** เพื่อส่งข้อมูลไฟล์ไปที่โลจิคัลวอลุ่ม

3. ไม่มีแนวคิดบล็อก Indirect/Double Indirect ในระบบไฟล์ JFS2 ไฟล์ถูกแสดงในรูปแบบขยาย ดังนั้นขนาดของการขยายสูงสุดขึ้นกับขนาดบล็อกรวม ด้วยขนาดบล็อกรวม 512 ไบต์ (ขนาดเล็กที่สุดที่ใช้ได้), ส่วนขยายมากที่สุดคือ  $512 * (2^{24} - 1)$  ไบต์ long (น้อยกว่า 8G เล็กน้อย) ด้วยขนาดบล็อกรวม 4096 ไบต์ (ขนาดใหญ่ที่สุดที่ใช้ได้), ส่วนขยายมากที่สุดคือ  $4096 * (2^{24} - 1)$  ไบต์ long (น้อยกว่า 64G เล็กน้อย)

ข้อจำกัดเหล่านี้ใช้ เฉพาะกับส่วนขยายเดียว; ไม่มีทางที่ข้อจำกัดเหล่านี้จะมีผล กับขนาดไฟล์โดยรวม

## แฟล็ก

### ไอเท็ม

- i คำอธิบาย แสดงบล็อกทางอ้อมสำหรับไฟล์ ถ้ามี บล็อกทางอ้อม ถูกแสดงในแบบของโลจิคัลหรือฟิสิคัลลวลุ่ม บล็อกแอดเดรส ขึ้นกับว่ามีกระบวนการแฟล็ก -I หรือ -p
- I แสดงตำแหน่งไฟล์ในแบบของโลจิคัลลวลุ่มแฟรกเมนต์ สำหรับโลจิคัลลวลุ่มที่มีไฟล์ แฟล็ก -I และ -p มีความเฉพาะตัวที่สัมพันธ์กัน  
หมายเหตุ: ถ้าไม่มีกระบวนการแฟล็ก -I หรือ -p แฟล็ก -I ถูกนำมาใช้โดยดีฟอลต์ ถ้ามีการระบุทั้ง สองแฟล็กจะใช้แฟล็ก -p
- m LogicalVolumeName แสดงโลจิคัลกับฟิสิคัลแม็พสำหรับ โลจิคัลลวลุ่ม
- n FragNumber แสดงไฟล์บล็อก โลจิคัลหรือฟิสิคัล มีช่วงจากบล็อกแรกไปที่บล็อกที่ตรงกับ FragNumber
- o FragOffset แสดงไฟล์บล็อก โลจิคัลหรือฟิสิคัล มีช่วงจากบล็อกที่ตรงกับ fragoffset + 1 กับบล็อกสุดท้าย คำสั่ง fileplace แสดง แอดเดรสของแฟรกเมนต์จำเพาะเมื่อทั้งแฟล็ก -n และ แฟล็ก -o ถูกระบุ
- p แสดงตำแหน่งไฟล์ในแบบของฟิสิคัลลวลุ่ม ที่จำเป็นสำหรับฟิสิคัลลวลุ่มที่มีไฟล์ ถ้าโลจิคัลลวลุ่ม ที่มี ไฟล์อยู่ถูกทำมีเรอร์ ตำแหน่งฟิสิคัลจะถูกแสดง สำหรับแต่ละสำเนาเรอร์ แฟล็ก -I และ -p มีความ เฉพาะตัวที่สัมพันธ์กัน
- v แสดงข้อมูลเพิ่มเติมเกี่ยวกับไฟล์และตำแหน่ง รวมถึงสถิติเกี่ยวกับการกระจายตัวข้ามลวลุ่มของไฟล์ และระดับของการแตกแฟรกเมนต์ในลวลุ่ม สถิติถูกแสดง ในแบบตัวเลข โลจิคัลหรือฟิสิคัลลวลุ่มแฟรกเมนต์ ขึ้นกับว่าระบุแฟล็ก -I หรือ -p

*ประสิทธิภาพของ พื้นที่ไฟล์* ถูกคำนวณเป็นจำนวนของแฟรกเมนต์ที่ไม่เป็นค่า null (N) ทหารด้วยช่อง ของแฟรกเมนต์ (R) ที่กำหนดให้กับไฟล์และคูณด้วย 100 หรือ  $(N/R) \times 100$  ช่วง ถูกคำนวณเป็น แอดเดรสที่กำหนดสูงสุดลบด้วยแอดเดรสที่กำหนดต่ำสุด บวก 1 หรือ  $MaxBlk - MinBlk + 1$  ตัวอย่าง โลจิคัลบล็อกที่เขียนสำหรับไฟล์คือ 01550 ถึง 01557 ดังนั้น N เท่ากับ 8 ช่วง R, (01557 - 01550 + 1) เท่ากับ 8 เช่นกัน ประสิทธิภาพ พื้นที่สำหรับไฟล์นี้คือ 100% หรือ  $8/8 \times 100$  ข้อความแฟล็ก -v พิมพ์ผลลัพธ์ของสมการ  $(N/R) \times 100$

ตามเมธอด การคำนวณประสิทธิภาพนี้ ไฟล์ที่ใหญ่กว่า 32KB จะไม่มีทางมีประสิทธิภาพ 100% เนื่อง จากการใช้บล็อกทางอ้อม

*ประสิทธิภาพโดยลำดับ* ถูกกำหนดเป็น 1 ลบ จำนวนช่องว่าง (nG) ที่หารด้วยจำนวน ช่องว่างที่เป็นไป ได้ (nPG) หรือ  $1 - (nG/nPG)$  จำนวนของช่องว่างที่เป็นไปได้เท่ากับ N ลบ 1 ( $nPG = N - 1$ ) ถ้าไฟล์ ถูกเขียนไปที่ 9 บล็อก (ใหญ่กว่า 32KB) และโลจิคัล แฟรกเมนต์คอลัมน์แสดง:

01550-01557  
01600

ไฟล์ถูกเก็บใน 2 แฟรกเมนต์จาก 9 แฟรกเมนต์ที่เป็นไปได้ การคำนวณประสิทธิภาพโดยลำดับสำหรับ ไฟล์นี้คือ:

$nG=1$   
 $nPG=9-1=8$   
 $(1-1/8) \times 100=87.5\%$

## ตัวอย่าง

1. เมื่อต้องการแสดงการจัดตำแหน่งไฟล์ในโลจิคัลลวลุ่ม ให้ป้อน:

fileplace data1

ตัวอย่าง นี้แสดงรายการของแฟรกเมนต์ และโลจิคัลวอลุ่มที่มีไฟล์ data1

2. เมื่อต้องการแสดงบล็อกทางอ้อมสำหรับไฟล์ให้ป้อน:

```
fileplace -i data1
```

นอกจาก รายการดีฟอลต์ของโลจิคัลวอลุ่มแฟรกเมนต์ บล็อกทางอ้อม (ถ้ามี) ที่ใช้เพื่อเก็บไฟล์บล็อกแอดเดรสในระบบไฟล์ ถูกแจกแจง

3. เมื่อต้องการแสดงข้อมูลการจัดตำแหน่งเพิ่มเติมสำหรับไฟล์ให้ป้อน:

```
fileplace -v data1
```

นอกจาก รายการดีฟอลต์ของโลจิคัลวอลุ่มแฟรกเมนต์ สถิติ เกี่ยวกับประสิทธิภาพของการจัดตำแหน่งถูกแสดง

4. เมื่อต้องการแสดงข้อมูลทั้งหมดเกี่ยวกับการจัดตำแหน่งของไฟล์บน ฟิสิคัลวอลุ่มให้ป้อน:

```
fileplace -piv data1
```

ตัวอย่าง นี้แสดงรายการของไฟล์และบล็อกทางอ้อมในแบบ ฟิสิคัลวอลุ่มที่จำเป็น และรวมถึงสถิติเกี่ยวกับ ประสิทธิภาพของการจัดตำแหน่ง

5. เมื่อต้องการแสดงตำแหน่งของฟิสิคัลวอลุ่มที่จำเป็นสำหรับ 18 บล็อกแรกในไฟล์ /usr/lib/boot/unix\_mp ให้ป้อน:

```
fileplace -n 18 -p /usr/lib/boot/unix_mp
```

6. เมื่อต้องการแสดงตำแหน่งของฟิสิคัลวอลุ่มที่จำเป็นจาก บล็อก 18th ถึงบล็อกสุดท้ายในไฟล์ /usr/lib/boot/unix\_mp ให้ป้อน:

```
fileplace -p -o 17 /usr/lib/boot/unix_mp
```

7. เมื่อต้องการแสดงตำแหน่งของฟิสิคัลวอลุ่มที่จำเป็นของ บล็อกที่ 18 ในไฟล์ /usr/lib/boot/unix\_mp ให้ป้อน:

```
fileplace -o 17 -n 1 -p /usr/lib/boot/unix_mp
```

## ไฟล์

ไอเท็ม

/dev/hd0, /dev/hd1, .../dev/hdn

คำอธิบาย

ระบุโลจิคัลวอลุ่ม

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง sync

การมอนิเตอร์ดิสก์ I/O

หน่วยเก็บโลจิคัลวอลุ่ม

---

## คำสั่ง find

### วัตถุประสงค์

ค้นหาไฟล์ที่มีนิพจน์ที่ตรงกัน

### ไวยากรณ์

```
find [ -H | -L ] Path ... [ Expression ]
```

## คำอธิบาย

คำสั่ง `find` ทำการค้นหา ลำดับไดเรกทอรีแบบทวนซ้ำ เพื่อหาพารามิเตอร์ *Path* ที่ระบุ ค้นไฟล์ที่ตรงกับนิพจน์ Boolean นิพจน์บูลีนถูกเขียน โดยใช้ประโยคที่กำหนดในข้อความต่อไปนี้ เมื่อคำสั่ง `find` ถูกทวนซ้ำโครงสร้างไดเรกทอรีจากมากไปหาน้อย จะไม่ค้นหาลงไปไดเรกทอรีที่ถูกลิงก์เชิงสัญลักษณ์ ในลำดับชั้นปัจจุบัน เอาต์พุตจากคำสั่ง `find` ขึ้นอยู่กับประโยคที่ระบุโดยพารามิเตอร์ *Expression*

คำสั่ง `find` ไม่สนับสนุนไวยากรณ์ 4.3 BSD `fast-find`

## แฟล็ก

ไอเท็ม คำอธิบาย

- H ทำให้ข้อมูลไฟล์และชนิดไฟล์ที่ถูกประเมิน สำหรับแต่ละลิงก์สัญลักษณ์ที่พบบนบรรทัดคำสั่ง เป็นไฟล์ที่อ้างถึงโดยลิงก์ และไม่ใช้ตัวลิงก์เอง หากไม่มีไฟล์ที่อ้างถึง ข้อมูลไฟล์และชนิดจะใช้สำหรับ ตัวลิงก์เอง ข้อมูลไฟล์สำหรับลิงก์เชิงสัญลักษณ์ทั้งหมดที่ไม่อยู่บนบรรทัดคำสั่ง เป็นของตัวลิงก์เอง
- L ทำให้ข้อมูลไฟล์และชนิดไฟล์ที่ถูกประเมิน สำหรับแต่ละลิงก์สัญลักษณ์ เป็นไฟล์ที่อ้างถึงโดยลิงก์ และไม่ใช้ตัวลิงก์เอง

## ประโยคนิพจน์

นิพจน์ Boolean และตัวแปรเหล่านี้ อธิบาย ขอบเขตการค้นหาของคำสั่ง `find` ตามที่ กำหนดในพารามิเตอร์ *Path* และ *Expression*

**หมายเหตุ:** ในนิยามต่อไปนี้ ตัวแปร  $n$  ระบุจำนวนเต็มฐานสิบที่สามารถถูกแสดงเป็น  $+n$  (มากกว่า  $n$ ),  $-n$  (น้อยกว่า  $n$ ) หรือ  $n$  (เท่ากับ  $n$ ) และ ตัวแปร *Number* ระบุจำนวนเต็มฐานสิบ ที่สามารถถูกแสดงเป็น  $+Number$  (มากกว่า *Number*),  $-Number$  (น้อยกว่า *Number*) หรือ *Number* (*Number*-1 ถึง *Number*)

ไอเท็ม

`\( Expression \)`  
`-amin n`

คำอธิบาย

หาค่าเป็นค่า True ถ้านิพจน์ในวงเล็บ เป็น true  
ค่าของ  $n$  สามารถเป็นหนึ่งใน ค่าต่อไปนี้:

- $n$  ประเมินค่าเป็น True ถ้าเวลาเข้าถึงไฟล์ลบจาก เวลาการกำหนดค่าเริ่มต้น หารด้วย 60 วินาที (โดยตัดเศษที่เหลือทิ้ง) คือ  $n$
- $-n$  หาค่าเป็น True ถ้าเวลาการเข้าถึงไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น หารด้วย 60 วินาที (ปัดเศษทิ้ง) น้อยกว่า  $n$
- $+n$  หาค่าเป็น True ถ้าเวลาการเข้าถึงไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น หารด้วย 60 วินาที (ปัดเศษทิ้ง) มากกว่า  $n$  (ในกรณีของ UNIX03, มากกว่า  $n+1$ )

ตัวอย่างเช่น `-amin 2` เป็น true หากไฟล์ถูกเข้าถึงภายใน 1 ถึง 2 นาที

**หมายเหตุ:** ไฟล์ที่ถูกเข้าถึงหลังจากเวลาเริ่มต้นของคำสั่ง `find` จะไม่ถูกพิจารณา อย่างไรก็ตาม เมื่อคำสั่ง `find` ถูกใช้ภายในตัวดำเนินการยูนารี NOT สำหรับลักษณะการทำงานที่ไม่ใช่ UNIX03 ไฟล์ที่ถูกแก้ไขหลังจาก เวลาเริ่มต้นของคำสั่งจะถูกแสดง จนถึงค่าของ  $n$

ไอเท็ม  
-atime *n*

#### คำอธิบาย

ค่าของ *n* สามารถเป็นหนึ่งใน ค่าต่อไปนี้:

- n*            หาค่าเป็น True ถ้าเวลาการเข้าถึงไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น ทหารด้วย 86400 วินาที (พิเศษทั้ง) คือ *n*
- n*            หาค่าเป็น True ถ้าเวลาการเข้าถึงไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น ทหารด้วย 86400 วินาที (พิเศษทั้ง) น้อยกว่า *n*
- +*n*            หาค่าเป็น True ถ้าเวลาการเข้าถึงไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น ทหารด้วย 86400 วินาที (พิเศษทั้ง) มากกว่า *n* (ในกรณีของ UNIX03, มากกว่า *n*+1)

หมายเหตุ: นิยามของ -atime ถูกเปลี่ยนเพื่อให้สอดคล้องกับข้อมูลจำเพาะ Single UNIX เวอร์ชัน 3 ลักษณะการทำงานก่อนหน้าของ -atime จะถูกประเมินเป็น True หากไฟล์ถูกเข้าถึงใน *n*-1 ถึง *n* คูณ 24 ชั่วโมง โดยดีพอลต์ find -atime ทำงานเหมือนกับที่ทำก่อน UNIX03 สามารถรับการทำงานของ UNIX03 ได้โดยการตั้งค่าตัวแปรสถานะแวดล้อม XPG\_SUS\_ENV เป็น ON และ XPG\_UNIX98 เป็น OFF

สามารถ รับการทำงานก่อนหน้าสำหรับอ็อพชันนี้ได้โดยตั้งค่าตัวแปร XPG\_UNIX98 เป็น ON

ไฟล์ที่ถูกเข้าถึงหลังจากเวลาเริ่มต้นของคำสั่ง find จะไม่ถูกพิจารณาอย่างไรก็ตาม เมื่อคำสั่ง find ถูกใช้ ภายในตัวดำเนินการยูนิารี NOT สำหรับลักษณะการทำงานที่ไม่ใช่ UNIX03 ไฟล์ที่ถูกแก้ไขหลังจาก เวลาเริ่มต้นของคำสั่งจะถูกแสดง จนถึงค่าของ *n* ค่าของ *n* สามารถเป็นหนึ่งใน ค่าต่อไปนี้:

- n*            ประเมินค่าเป็น True ถ้าเวลาการแก้ไข i-node ไฟล์ลบจาก เวลาการกำหนดค่าเริ่มต้น ทหารด้วย 60 วินาที (โดยตัดเศษที่เหลือทั้ง) คือ *n*
- n*            หาค่าเป็น True ถ้าเวลาการแก้ไข i-node ของไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น ทหารด้วย 60 วินาที (พิเศษทั้ง) น้อยกว่า *n*
- +*n*            หาค่าเป็น True ถ้าเวลาการแก้ไข i-node ของไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น ทหารด้วย 60 วินาที (พิเศษทั้ง) มากกว่า *n* (ในกรณีของ UNIX03, มากกว่า *n*+1)

หมายเหตุ: ไฟล์ที่มี i-nodes ที่ถูกแก้ไขหลังจากเวลาเริ่มต้นของคำสั่ง find จะไม่ถูกพิจารณา อย่างไรก็ตาม เมื่อคำสั่ง find ถูกใช้ภายในตัวดำเนินการยูนิารี NOT สำหรับลักษณะการทำงานที่ไม่ใช่ UNIX03 ไฟล์ที่มี i-node ที่ถูกแก้ไขหลังจาก เวลาเริ่มต้นของคำสั่งจะถูกแสดง จนถึงค่าของ *n* เขียนไฟล์ปัจจุบันไปที่อุปกรณ์ที่ระบุในรูปแบบคำสั่ง cpio

-cmin *n*

-cpio Device

ไอเท็ม  
-ctime *n*

### คำอธิบาย

ค่าของ *n* สามารถเป็นหนึ่งใน ค่าต่อไปนี้:

- n***            หาค่าเป็น True ถ้าเวลาการแก้ไข i-node ของไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น  
                  หารด้วย 86400 วินาที (ปัดเศษทิ้ง) คือ *n*
- n***            หาค่าเป็น True ถ้าเวลาการแก้ไข i-node ของไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น  
                  หารด้วย 86400 วินาที (ปัดเศษทิ้ง) น้อยกว่า *n*
- +*n***            หาค่าเป็น True ถ้าเวลาการแก้ไข i-node ของไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น  
                  หารด้วย 86400 วินาที (ปัดเศษทิ้ง) มากกว่า *n* (ในกรณีของ UNIX03, มากกว่า *n*+1)

หมายเหตุ: นิยามของ -ctime ถูกเปลี่ยนเพื่อให้สอดคล้องกับข้อมูลจำเพาะ Single UNIX เวอร์ชัน 3  
ลักษณะการทำงานก่อนหน้านี้ของ -ctime จะถูกประเมินเป็น True หากไฟล์ถูกเข้าถึงใน *n*-1 ถึง *n* คูณ  
24 ชั่วโมง โดยดีฟอลต์ find -ctime ทำงานเหมือนกับที่ทำก่อน UNIX03 สามารถรับการทำงานของ  
UNIX03 ได้โดยการตั้งค่าตัวแปรสถานะแวดล้อม XPG\_SUS\_ENV เป็น ON และ XPG\_UNIX98 เป็น  
OFF

สามารถรับการทำงานก่อนหน้านี้สำหรับอ็อปชันนี้ได้โดยตั้งค่าตัวแปร XPG\_UNIX98 เป็น ON

-depth

ไฟล์ที่มี i-nodes ที่ถูกแก้ไขหลังจากเวลาเริ่มต้นของคำสั่ง find จะไม่ถูกพิจารณา อย่างไรก็ตาม เมื่อคำ  
สั่ง find ถูกใช้ภายในตัวดำเนินการยูนารี NOT สำหรับลักษณะการทำงานที่ไม่ใช่ UNIX03 ไฟล์ที่มี i-  
node ที่ถูกแก้ไขหลังจาก เวลาเริ่มต้นของคำสั่งจะถูกแสดง จนถึงค่าของ *n*  
หาค่าได้เป็น True เสมอ ทำให้ข้อมูลที่สืบทอดของ ลำดับชั้นไดเรกทอรีถูกดำเนินการ เพื่อที่รายการทั้ง  
หมดในไดเรกทอรีได้รับผล ก่อนที่ตัวไดเรกทอรีเองจะได้รับผล ซึ่งมีประโยชน์ เมื่อคำสั่ง find ถูกใช้กับ  
คำสั่ง cpio เพื่อถ่ายโอนไฟล์ที่อยู่ใน ไดเรกทอรีที่ไม่มีสิทธิ์ในการเขียน

-ea

-exec *Command*

หาค่าเป็นค่า True ถ้าไฟล์มีข้อมูล access control (ACL) หรือ Extended attributes (EA) ตั้งไว้  
หาค่าเป็นค่า True ถ้าคำสั่งที่ระบุไว้ และส่งคืนค่า 0 เป็นสถานะ exit จุดสิ้นสุดของคำสั่งที่ระบุ ต้องถูก  
ค้นด้วยเซมิโคลอนในเครื่องหมายคำพูด, escaped เซมิโคลอน, หรือเครื่องหมายบวกรหัสอักขระที่มี  
สองอักขระ {} (เครื่องหมายปีกกา) ต้องตามด้วยเครื่องหมายบวกรหัสอักขระที่ค้นจุดสิ้นสุดของคำสั่ง ที่ระบุ พารามิเตอร์คำสั่ง {} (เครื่องหมายปีกกา) ถูกแทนที่ ด้วยชื่อพารามิเตอร์ปัจจุบัน

-follow

-fstype *Type*

ทำให้ลิงก์สัญลักษณ์และฮาร์ดลิงก์ถูกติดตาม  
หาค่าเป็นค่า True ถ้าระบบไฟล์ซึ่งไฟล์ใช้ เป็นชนิดที่ระบุ ตัวแปร *Type* มีค่าเป็น jfs (journaled file  
system), nfs (network file system), jfs2 (enhanced journaled file system), procfs (proc file  
system) หรือ namefs (name file system)

-group *Group*

หาค่าเป็นค่า True ถ้าไฟล์เป็นสมาชิกของกลุ่มที่ ระบุ ถ้าค่าของตัวแปร *Group* เป็นตัวเลขและไม่ได้อยู่ใน  
ไฟล์ /etc/group จะถูกแปลเป็น group ID

-inum *n*

-links *n*

-iregex *regular\_expression*

หาค่าเป็นค่า True ถ้าไฟล์มีจำนวนลิงก์ ที่ระบุ ดุที่คำสั่ง ln สำหรับรายละเอียด ของลิงก์

-long

ประเมินเป็นค่า True ถ้าชื่อพารามิเตอร์ทั้งหมดของไฟล์ตรงกับนิพจน์ ปกติ อ็อปชันนี้คล้ายกับอ็อปชัน -regex  
ยกเว้นว่าการจับคู่ ไม่คำนึงถึงตัวพิมพ์  
พิมพ์อักขระที่มีทั้งหมดของแต่ละชื่อ ผู้ใช้/กลุ่ม แทนการตัดท้ายเป็น 8 อักขระแรกเมื่อใช้ร่วมกับ -ls

ไอเอ็ม  
-ls

### คำอธิบาย

หาค่าได้เป็น True เสมอ ทำให้ชื่อพารปัจจุบัน ถูกพิมพ์ร่วมกับสถิติที่เกี่ยวข้อง สถิติเหล่านี้ รวมถึงค่าต่อไปนี้:

- หมายเลข I-node
- ขนาดหน่วยเป็น KB (1024 ไบต์)
- โหมดการปกป้อง
- จำนวนฮาร์ดลิงก์
- User
- กลุ่ม
- ขนาดเป็นไบต์
- เวลาแก้ไข

-mminn

หากไฟล์เป็นไฟล์พิเศษ พิเศษขนาด จะมีหมายเลขอุปกรณ์หลักและรอง ถ้าไฟล์เป็น ลิงก์เชิงสัญลักษณ์ ชื่อพารของไฟล์ที่ถูกลิงก์ไปจะถูกพิมพ์ ด้วยสัญลักษณ์ -> (เครื่องหมายชี้ดเค้น, มากกว่า) นำหน้า การ จัดรูปแบบเหมือนกับคำสั่ง ls -flids อย่างไรก็ตาม การจัดรูปแบบถูก ดำเนินการภายในโดยไม่มีการรัน คำสั่ง ls ดังนั้น อาจมีความแตกต่างในเอาต์พุตกับคำสั่ง ls เช่น โดยการใช้โหมดการป้องกัน คาของ n สามารถเป็นหนึ่งใน ค่าต่อไปนี้:

**n**        หาค่าเป็น True ถ้าเวลาการแก้ไขไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น ทารด้วย 60 วินาที (ปัดเศษทิ้ง) คือ n

**-n**        หาค่าเป็น True ถ้าเวลาการแก้ไขไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น ทารด้วย 60 วินาที (ปัดเศษทิ้ง) น้อยกว่า n

**+n**        หาค่าเป็น True ถ้าเวลาการแก้ไขไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น ทารด้วย 60 วินาที (ปัดเศษทิ้ง) มากกว่า n (ในกรณีของ UNIX03, มากกว่า n+1)

**หมายเหตุ:** ไฟล์ที่ถูกปรับเปลี่ยนหลังจากเวลาเริ่มต้นของคำสั่ง find จะไม่ถูกพิจารณา อย่างไรก็ตาม เมื่อคำสั่ง find ถูกใช้ภายในตัวดำเนินการยูนิริ NOT สำหรับลักษณะการทำงานที่ไม่ใช่ UNIX03 ไฟล์ที่ถูกแก้ไขหลังจาก เวลาเริ่มต้นของคำสั่งจะถูกแสดง จนถึงค่าของ n คาของ n สามารถเป็นหนึ่งใน ค่าต่อไปนี้:

-mtime n

**n**        หาค่าเป็น True ถ้าเวลาการแก้ไขไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น ทารด้วย 86400 วินาที (ปัดเศษทิ้ง) คือ n 86400 วินาทีคือ 24 ชั่วโมง

**-n**        หาค่าเป็น True ถ้าเวลาการแก้ไขไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น ทารด้วย 86400 วินาที (ปัดเศษทิ้ง) น้อยกว่า n

**+n**        หาค่าเป็น True ถ้าเวลาการแก้ไขไฟล์ที่แยกจาก เวลาการกำหนดค่าเริ่มต้น ทารด้วย 86400 วินาที (ปัดเศษทิ้ง) มากกว่า n (ในกรณีของ UNIX03, มากกว่า n+1)

**หมายเหตุ:** นิยามของ -mtime ถูกเปลี่ยนแปลงเพื่อให้สอดคล้องกับ Single UNIX Specification, Version 3 ลักษณะการทำงานก่อนหน้านี้ของ -mtime จะถูกประเมินเป็น True หากไฟล์ถูกปรับเปลี่ยน ใน n-1 ถึง n คูณด้วย 24 ชั่วโมง โดยดีฟอลต์ find -mtime ทำงานเหมือนกับที่ทำก่อน UNIX03 สามารถรับการทำงานของ UNIX03 ได้โดยการตั้งค่าตัวแปรสภาวะแวดล้อม XPG\_SUS\_ENV เป็น ON และ XPG\_UNIX98 เป็น OFF

สามารถ รับการทำงานก่อนหน้านี้สำหรับอ็อพชันนี้ได้โดยตั้งค่าตัวแปร XPG\_UNIX98 เป็น ON

ไฟล์ที่ถูกปรับเปลี่ยนหลังจากเวลาเริ่มต้นของคำสั่ง find จะไม่ถูกพิจารณา อย่างไรก็ตาม เมื่อคำสั่ง find ถูกใช้ภายในตัวดำเนินการยูนิริ NOT สำหรับลักษณะการทำงานที่ไม่ใช่ UNIX03 ไฟล์ที่ถูกแก้ไขหลังจาก เวลาเริ่มต้นของคำสั่งจะถูกแสดงจนถึงค่าของ n

## ไอเท็ม

-name File

### ค่าอธิบาย

หาค่าเป็นค่า True ถ้าค่าของตัวแปร File ตรงกับชื่อไฟล์ อักขระการสร้างชื่อไฟล์เซลล์โดยทั่วไป (ดูที่คำสั่ง sh) สามารถใช้ได้ รูปแบบต้อง อยู่ในเครื่องหมายคำพูดหรืออักขระ escape ใช้อักขระ escape เมื่อใช้คำสั่ง find จากเซลล์ backslash (\) ถูกใช้เป็น escape character ภายในรูปแบบ คุณสามารถใช้อักขระ wildcard (การจับคู่รูปแบบ) โดยอยู่ภายในเครื่องหมายคำพูด สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการใช้อักขระ wildcard ดูที่ การจับคู่รูปแบบกับ wildcards และอักขระเมตา ใน *Operating system and device management*

-newer File

-nogroup

-nouser

-ok Command

ใน นิพจน์เช่น [a-z] เครื่องหมายขีดคั่นหมายความว่า ถึง ตามลำดับการเรียงปัจจุบัน ลำดับการเรียงอาจกำหนดคลาสเทียบเท่า สำหรับใช้ในช่วงอักขระ สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการเรียงลำดับและคลาสที่เหมือนกัน ดูที่ “ภาพรวมการสนับสนุนภาษาประจำชาติ” ใน *National Language Support Guide and Reference*

จะประเมินเป็นค่า True หากไฟล์ปัจจุบันถูกแก้ไขล่าสุดกว่า ไฟล์ที่ระบุโดยตัวแปร File

หาค่าเป็นค่า True ถ้าไฟล์เป็นสมาชิกของ กลุ่มที่ไม่ได้อยู่ในฐานข้อมูล /etc/group

หาค่าเป็นค่า True ถ้าไฟล์เป็นสมาชิกของ ผู้ใช้ที่ไม่ได้อยู่ในฐานข้อมูล /etc/passwd

เหมือนกับนิพจน์ -exec ยกเว้นว่า คำสั่ง find จะตรวจสอบว่าต้องเริ่มต้น คำสั่งที่ระบุหรือไม่ การตอบยืนยันจะเริ่มคำสั่ง จุดสิ้นสุดของคำสั่งระบุต้องถูกคั่นด้วยเซมิโคลอน ที่อยู่ในเครื่องหมายคำพูดหรือ\;

(backslash-escape semicolon)

-perm [ - ] OctalNumber

หาค่าเป็นค่า True ถ้าโคดสิทธิการใช้งานของไฟล์ ตรงกับพารามิเตอร์ OctalNumber สำหรับรายละเอียดเกี่ยวกับสิทธิการใช้งานของไฟล์ อ้างอิงที่คำสั่ง chmod ถ้ามี - (เครื่องหมายขีดคั่น) ทางเลือกอยู่ นิพจน์นี้จะถูกหาค่าเป็นค่า true ถ้าสิทธิเหล่านี้ ถูกเซต พารามิเตอร์ OctalNumber สามารถเป็นเลขฐานแปดได้ถึง 9 หลัก

หมายเหตุ: สำหรับไฟล์ที่เป็นส่วนหนึ่งของ สภาวะแวดล้อม TCB บิตความปลอดภัยเพิ่มเติมจะถูกเพิ่มให้กับสิทธิการใช้งาน ของไฟล์ ไฟล์เหล่านี้มีบิตเซต S\_ITCB และบิตเซตความปลอดภัย ถูกกำหนดเป็น 0x010000000 ดังนั้น ค่าสิทธิ์เลขฐานแปดของไฟล์ที่ TCB เปิดใช้งานต้องรวมการตั้งค่าบิต 100000000 ตามด้วยบิตสิทธิ์ การใช้งานอื่น

-perm [ - ] Mode

ตัวอย่าง: เมื่อต้องการแสดงรายการไฟล์ซึ่งเป็นส่วนหนึ่งของสภาวะแวดล้อม TCB find -perm 100000600 -print ซึ่งจะแสดงชื่อไฟล์ ที่มีเฉพาะสิทธิ์ owner-read และ owner-write และเป็นส่วนหนึ่งของสภาวะแวดล้อม TCB ดูที่คำสั่ง chmod สำหรับการอธิบายโคดสิทธิการใช้งานอาร์กิวเมนต์ mode ถูกใช้เพื่อแสดงไฟล์โหมดบิต คำมีรูปแบบเหมือนกับตัวถูกดำเนินการ <symbolicmode> ที่อธิบาย ใน chmod และถูกแปลความหมายดังนี้:

ขั้นแรก เพิ่มเพลตจะถือว่าไฟล์โหมดบิตทั้งหมดเคลียร์แล้ว สัญลักษณ์ Op (-) มีฟังก์ชันต่อไปนี้:

+ ตั้งโหมดบิตที่เหมาะสมในเพิ่มเพลต

- เคลียร์บิตที่เหมาะสม

= เซตโหมดบิตที่เหมาะสม โดยไม่สนใจกับเนื้อหาของมาสก์การสร้างโหมดไฟล์ของกระบวนการ

สัญลักษณ์ op - ต้องไม่เป็นอักขระแรกของ โหมด ซึ่งช่วยหลีกเลี่ยงความกำกวมที่มีขีดคั่นนำหน้าที่เป็นทางเลือก เนื่องจากโหมดเริ่มต้นคือบิตทั้งหมดถูกปิด จึงไม่มีโหมดเชิงสัญลักษณ์ ที่ต้องใช้ - เป็นอักขระแรก

ถ้าเครื่องหมายขีดคั่น ถูกละเว้น การหาค่าหลักเป็น True เมื่อบิตสิทธิการใช้งานของไฟล์ เหมือนกับค่าของเพิ่มเพลตผลลัพธ์ มิฉะนั้น ถ้าโหมดถูกนำหน้าด้วยเครื่องหมายขีดคั่น การหาค่าหลักเป็น True ถ้าบิตทั้งหมดในเพิ่มเพลตผลลัพธ์ถูกเซตในบิตสิทธิการใช้งาน ของไฟล์

-print

พารามิเตอร์ Mode เหมือนกับ ไวยากรณ์คำสั่ง chmod นิพจน์นี้ หาค่าเป็นค่า True ถ้าไฟล์มีสิทธิการใช้งานเหล่านี้เหมือนกัน ถ้ามี - (เครื่องหมายขีดคั่น) ทางเลือกอยู่ นิพจน์นี้จะถูกหาค่าเป็นค่า True ถ้าสิทธิเหล่านี้ ถูกเซต

หาค่าได้เป็น True เสมอ แสดงชื่อพาร ปัจจุบัน คำสั่ง find จะใช้นิพจน์ -print โดยปกติ นอกจากจะมีนิพจน์ -exec, -ls หรือ -ok อยู่

ไอเท็ม	คำอธิบาย
-prune	หาค่าได้เป็น True เสมอ หยุดการสืบทอด ของชื่อพารปัจจุบัน ถ้าเป็นไดเรกทอรี ถ้าแฟล็ก -depth ถูก ระบุ แฟล็ก -prune ถูกละเว้น
-size n	หาค่าเป็นค่า True ถ้าไฟล์ถูกระบุ n ของความยาวบล็อก (512 ไบต์ต่อบล็อก) ขนาดไฟล์ ถูกปัดเป็น บล็อกที่ใกล้เคียงที่สุดเพื่อทำการเปรียบเทียบ
-regex regular_expression	ประเมินเป็นค่า True ถ้าชื่อพารทั้งหมดของไฟล์ตรงกับนิพจน์ปกติ อ็อพชันนี้ไม่ได้ค้นหานิพจน์ปกติ แต่จับคู่นิพจน์ปกติ กับชื่อพารที่สมบูรณ์ของไฟล์ ตัวอย่างเช่น เพื่อจับคูไฟล์ที่ชื่อ ./test คุณสามารถ ใช้นิพจน์ปกติ .test.* หรือ .*t.*t แต่ไม่ใช่ t.*t
-regextype Type	หาค่าได้เป็น True เสมอ อ็อพชันนี้ระบุชนิดของไวยากรณ์นิพจน์ปกติ สำหรับอ็อพชัน -regex และ -iregex และยังมีผลต่อ นิพจน์ปกติที่เกิดขึ้นภายหลังในบรรทัดรับคำสั่งด้วย
	ตัวแปร Type สามารถมีค่าอย่างใดอย่างหนึ่งต่อไปนี้:
	<b>พื้นฐาน</b> สำหรับไวยากรณ์นิพจน์ปกติพื้นฐาน
	<b>ขยาย</b> สำหรับไวยากรณ์นิพจน์ปกติที่ขยาย
	<b>หมายเหตุ:</b> ถ้าไม่ได้ใช้อ็อพชัน -regextype นิพจน์ปกติจะ ถูกตีความเป็น basic
-size nc	หาค่าเป็นค่า True ถ้าไฟล์ตรงกับความยาวไบต์ n ที่ระบุ การเพิ่ม c ให้กับ จุดสิ้นสุดของตัวแปร n ระบุ ขนาดของไฟล์ถูกวัดผลในแต่ละไบต์ไม่ใช่บล็อก
-type Type	หาค่าเป็นค่า True ถ้าตัวแปร Type ระบุหนึ่งในค่าต่อไปนี้:
	<b>b</b> บล็อกไฟล์พิเศษ
	<b>c</b> อักขระไฟล์พิเศษ
	<b>d</b> ไดเรกทอรี
	<b>f</b> ไฟล์ปกติ
	<b>l</b> ลิงก์เชิงสัญลักษณ์
	<b>p</b> FIFO (ไพล์ที่กำหนดชื่อ)
	<b>s</b> ซ็อกเก็ต
-user User	หาค่าเป็นค่า True ถ้าไฟล์เป็นสมาชิกของผู้ใช้ที่ระบุ ถ้าค่าของตัวแปร User เป็นตัวเลขและไม่มีอยู่ใน ไฟล์ /etc/passwd จะถูกแปลเป็น ID ผู้ใช้
-xdev	หาค่าได้เป็น True เสมอ ปกป้องคำสั่ง find จากการข้ามระบบไฟล์อื่น จากที่ระบุโดยพารามิเตอร์ Path

นิพจน์เหล่านี้สามารถถูกรวม โดยใช้โอเปอเรเตอร์ต่อไปนี้เพื่อลดลำดับนำหน้า:

1. ( Expression ) - กลุ่มในวงเล็บของนิพจน์และโอเปอเรเตอร์ (วงเล็บมีความหมายพิเศษกับเซลล์และต้องการลำดับ  
backslash-escape)
2. ! นิพจน์ - นิเสธของนิพจน์ ('!' คือโอเปอเรเตอร์ยูนิควารี NOT)
3. Expression [ - a ] Expression - การรวมกันของนิพจน์ (ตัวดำเนินการ AND หมายถึงการรวมกันของ สองข้อมูลหลักหรือ  
ระบุโดยตรงเป็น -a)
4. Expression - o Expression - การเปลี่ยน ข้อมูลหลัก; -o คือโอเปอเรเตอร์ OR นิพจน์ที่สองจะไม่ถูกประเมิน ถ้านิพจน์แรก  
เป็น true

**หมายเหตุ:** เมื่อใช้คำสั่ง find และ cpio ร่วมกัน คุณต้องใช้อ็อพชัน -follow และ -L กับคำสั่ง cpio การไม่ใช้สองอ็อพชันนี้รวม  
กันจะสร้างผลลัพธ์ที่ไม่ต้องการ ถ้านิพจน์ไม่มีอยู่, -print ตามที่ใช้ในนิพจน์ ดีฟอลต์ ตัวอย่าง หากนิพจน์ที่ระบุไม่มีข้อมูล  
-exec, -ok หรือ -print หลัก นิพจน์จะถูกแทนที่โดย (given\_expression) -print ข้อมูลหลัก -user, -group และ  
-newer แต่ละรายการ หาค่าอาร์กิวเมนต์ตามลำดับเพียงหนึ่งครั้ง การใช้คำสั่ง ที่ระบุโดย -exec หรือ -ok ไม่มีผลกับข้อมูล  
หลัก ในภายหลัง บนไฟล์เดียวกัน

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย

0 พารามิเตอร์ *Path* ทั้งหมดถูกดำเนินการสำเร็จแล้ว  
>0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการแสดงระบบไฟล์ที่ระบุชื่อไฟล์ฐาน ให้พิมพ์:

```
find / -name .profile -print
```

คำสั่งนี้ค้นหาทั้งระบบไฟล์และเขียนชื่อพารามิเตอร์ของไฟล์ทั้งหมดที่มีชื่อ `.profile / (slash)` กำหนดให้คำสั่ง `find` ค้นหาไดเรกทอรีรากและไดเรกทอรีย่อยในนั้นทั้งหมดเพื่อไม่ให้เสียเวลา เป็นการดีที่สุดที่จะจำกัดการค้นหาโดยระบบไดเรกทอรีที่คุณคิดว่าอาจจะมีไฟล์อยู่

2. เมื่อต้องการแสดงไฟล์ที่มีโค้ดสิทธิ์เฉพาะ ในแผนผังไดเรกทอรีในปัจจุบัน ให้พิมพ์:

```
find . -perm 0600 -print
```

คำสั่งนี้แสดงชื่อของไฟล์ที่มีสิทธิ์การใช้งาน เฉพาะ `owner-read` และ `owner-write` . (จุด) กำหนดให้คำสั่ง `find` ค้นหาไดเรกทอรีปัจจุบันและไดเรกทอรีย่อย ดูที่คำสั่ง `chmod` สำหรับการอธิบายโค้ดสิทธิ์การใช้งาน

3. เมื่อต้องการค้นหาหลายไดเรกทอรีเพื่อหาไฟล์ที่มีโค้ดสิทธิ์การใช้งานที่ต้องการ ให้พิมพ์:

```
find manual clients proposals -perm -0600 -print
```

คำสั่งนี้แสดงชื่อของไฟล์ที่มีสิทธิ์ `owner-read` และ `owner-write` และสิทธิ์อื่นที่เป็นไปได้ไดเรกทอรี `manual, clients` และ `proposals` และไดเรกทอรีย่อยจะถูกค้นหา ในตัวอย่างก่อนหน้านี้ `-perm 0600` เลือกเฉพาะไฟล์ที่มีโค้ดสิทธิ์การใช้งาน ที่ตรงกับ `0600` เท่านั้น ในตัวอย่างนี้ `-perm -0600` จะเลือก ไฟล์ที่มีโค้ดสิทธิ์ที่อนุญาตการเข้าถึงที่ระบุโดย `0600` และการเข้าถึงอื่นที่มากกว่าระดับ `0600` และยังคงตรงกับโค้ด สิทธิ์การใช้งาน `0622` และ `0744`

4. เมื่อต้องการแสดงไฟล์ทั้งหมดในไดเรกทอรีปัจจุบันที่ ถูกเปลี่ยนระหว่างระยะเวลา 24 ชั่วโมงในปัจจุบัน ให้พิมพ์:

```
find . -ctime 1 -print
```

5. เมื่อต้องการค้นหาไฟล์ทั่วไปกับหลายลิงก์ ให้พิมพ์:

```
find . -type f -links +1 -print
```

คำสั่งนี้แสดงชื่อของไฟล์ทั่วไป (`-type f`) ที่มีมากกว่าหนึ่งลิงก์ (`-links +1`)

หมายเหตุ: ทุกไดเรกทอรีมีอย่างน้อยสองลิงก์: รายการในพาเรนต์ไดเรกทอรีและตัวไดเรกทอรีเอง . (จุด) คำสั่ง `ln` อธิบายหลายไฟล์ลิงก์

6. เมื่อต้องการค้นหาไฟล์ที่เข้าถึงได้ทั้งหมด ซึ่งชื่อพารามีคำว่า `find` ให้พิมพ์:

```
find . -name '*find*' -print
```

7. เมื่อต้องการลบไฟล์ทั้งหมดที่มีชื่อ `a.out` หรือ `*.o` ที่ไม่ถูกเข้าถึงเป็นเวลาหนึ่งสัปดาห์ และที่ไม่ถูกเมาท์โดยใช้ `nfs` ให้พิมพ์:

```
find / \( -name a.out -o -name '*.o' \) -atime +7 ! -fstype nfs -exec rm {} \;
```

หมายเหตุ: จำนวนที่ใช้ภายในนิพจน์ -atime คือ +7 ซึ่งเป็นรายการที่ถูกต้องหากคุณต้องการให้คำสั่งทำงานกับไฟล์ที่ไม่ถูกเข้าถึงมากกว่าหนึ่งสัปดาห์ (ระยะเวลา 7 คูณ 24 ชั่วโมง)

8. เมื่อต้องการพิมพ์ชื่อพารของไฟล์ทั้งหมดในหรือด้านล่างของไดเรกทอรีปัจจุบัน ยกเว้นไดเรกทอรีชื่อ SCCS หรือไฟล์ในไดเรกทอรี SCCS ให้พิมพ์:

```
find . -name SCCS -prune -o -print
```

เมื่อต้องการ พิมพ์ชื่อพารของไฟล์ทั้งหมดในหรือด้านล่างของไดเรกทอรีปัจจุบัน รวมทั้งชื่อของไดเรกทอรี SCCS ให้พิมพ์:

```
find . -print -name SCCS -prune
```

9. เมื่อต้องการค้นหาไฟล์ทั้งหมดที่มีความยาว 414 ไบต์พอดี ให้พิมพ์:

```
find . -size 414c -print
```

10. เมื่อต้องการค้นหาหรือลบไฟล์ทั้งหมดในไดเรกทอรีโฮมของคุณ ด้วยคำต่อท้าย .c ให้พิมพ์:

```
find /u/arnold -name "*.c" -exec rm {} \;
```

ทุกครั้ง ที่คำสั่ง find ระบุไฟล์ด้วย คำต่อท้าย .c คำสั่ง rm ลบไฟล์นั้น คำสั่ง rm เป็นพารามิเตอร์เดียว ที่ถูกระบุสำหรับนิพจน์ -exec แสดง {} (เครื่องหมายปีกกา) ที่แสดงชื่อพารปัจจุบัน

11. ในตัวอย่างนี้ dirlink เป็นลิงก์เชิงสัญลักษณ์กับไดเรกทอรี dir เมื่อต้องการแสดงไฟล์ใน dir โดยการอ้างอิง ไปที่ลิงก์เชิงสัญลักษณ์ dirlink บนบรรทัดคำสั่ง ให้พิมพ์:

```
find -H dirlink -print
```

12. ในตัวอย่างนี้ dirlink เป็นลิงก์เชิงสัญลักษณ์กับไดเรกทอรี dir เมื่อต้องการแสดงไฟล์ใน dirlink, การแหวผ่านลำดับชั้นไฟล์ภายใต้ dir รวมทั้งลิงก์เชิงสัญลักษณ์ ให้พิมพ์:

```
find -L dirlink -print
```

13. เมื่อต้องการกำหนดว่าไฟล์ dir1 ที่อ้างอิงโดย ลิงก์เชิงสัญลักษณ์ dirlink ใหม่กว่า dir2 หรือไม่ ให้พิมพ์:

```
find -H dirlink -newer dir2
```

หมายเหตุ: เนื่องจากการใช้แฟล็ก -H ข้อมูลเวลาที่รวบรวมไม่ได้มาจาก dirlink แต่มาจาก dir1 ซึ่งพบโดยการแหวผ่านลิงก์เชิงสัญลักษณ์

14. เมื่อต้องการสร้างรายการของไฟล์ในไดเรกทอรีปัจจุบันในรูปแบบ ls ที่มีชื่อผู้ใช้และชื่อกลุ่มที่ขยาย ให้พิมพ์:

```
find . -ls -long
```

15. เมื่อต้องการแสดงไฟล์ที่มีเซต ACL/EA ในไดเรกทอรีปัจจุบัน ให้พิมพ์:

```
find . -ea
```

16. เมื่อต้องการแสดงไฟล์ที่แก้ไขภายใน 60 นาที ให้พิมพ์:

```
find . -mmin -60
```

17. เมื่อต้องการค้นหาชื่อพารทั้งหมดในไดเรกทอรี /home ที่มี รูปแบบ afile ในชื่อพาร ให้พิมพ์คำสั่ง ต่อไปนี้:

```
find /home -regextype basic -regex ".*afile.*"
```

18. เมื่อต้องการค้นหาชื่อพารทั้งหมดในไดเรกทอรี /home ที่มี รูปแบบ afile หรือ cap ในชื่อพาร ให้พิมพ์คำสั่ง ต่อไปนี้:

```
find /home -regextype extended -regex ".*afile.*|.*cap.*"
```

19. เมื่อต้องการค้นหาชื่อพารทั้งหมดในไดเรกทอรี /home ที่มี รูปแบบ afile, AFILE, cap หรือ CAP ในชื่อพาร ให้พิมพ์คำสั่ง ต่อไปนี้:

```
find /home -regextype extended -iregex ".*afile.*|.*cap.*"
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/find	มีคำสั่ง <code>find</code>
/bin/find	ลิงก์เชิงสัญลักษณ์ไปที่คำสั่ง <code>find</code>
/etc/group	มีรายการของกลุ่มที่รู้จักทั้งหมด
/etc/passwd	มีรายการของผู้ใช้ที่รู้จักทั้งหมด

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `ln`

เมธอดการสำรองข้อมูล

ชนิดของไฟล์

การเปลี่ยนทิศทางอินพุตและเอาต์พุต

คำสั่ง Shells

---

## คำสั่ง `finger`

### วัตถุประสงค์

แสดงข้อมูลผู้ใช้ คำสั่งนี้เหมือนกับ คำสั่ง `f`

### ไวยากรณ์

```
{ finger | f } [ -b ] [ -h ] [ -l ] [ -p ] [ -i ] [ -q ] [ -s ] [ -w ]
```

```
[ -f ] [ -m ] [ User | User@Host | @Host ]
```

### คำอธิบาย

คำสั่ง `/usr/bin/finger` แสดง ข้อมูลเกี่ยวกับผู้ใช้ที่ขณะนี้ล็อกอินอยู่ในโฮสต์ รูปแบบของ เอาต์พุตแตกต่างกันตามอ็อปชันของข้อมูลที่แสดง

### รูปแบบดีฟอลต์

รูปแบบดีฟอลต์ประกอบด้วยรายการต่อไปนี้:

- ชื่อล็อกอิน
- ชื่อนามสกุลผู้ใช้
- ชื่อเทอร์มินัล
- สถานะการเขียน (\* (เครื่องหมายดอกจัน) หน้าชื่อ เทอร์มินัลระบุว่าสิทธิในการเขียนถูกปฏิเสธ)

สำหรับผู้ใช้แต่ละคนบนโฮสต์ รายการข้อมูลดีฟอลต์ ยังประกอบด้วยรายการต่อไปนี้ ถ้าทราบ:

- เวลาเดินเครื่องเปล่า (เวลาเดินเครื่องเปล่าเป็นนาทีหากเป็นเลขจำนวนเต็ม เดียว เป็นชั่วโมงและนาที หาก : (โคลอน) แสดงอยู่ หรือวันและชั่วโมงหาก "d" แสดง)

- เวลาล็อกอิน
- ข้อมูลเฉพาะไซต์

ข้อมูลเฉพาะไซต์ถูกเรียกข้อมูลจาก ไฟล์ `gecos` ในไฟล์ `/etc/passwd` ไฟล์ `gecos` อาจมีชื่อนามสกุลผู้ใช้ตามด้วยเครื่องหมายจุลภาคหรือ / (อักขระสแลช) ข้อมูลทั้งหมดที่ตามหลัง เครื่องหมายจุลภาคหรืออักขระสแลชถูกแสดงโดยคำสั่ง `finger` ที่มีข้อมูลเฉพาะไซต์

## รูปแบบยาว

รูปแบบยาวใช้โดยคำสั่ง `finger` เมื่อใดที่กำหนดรายการชื่อของผู้ใช้ (ชื่อแอดเดสรวมทั้งชื่อและนามสกุลของผู้ใช้ที่ยอมรับได้) รูปแบบนี้เป็นแบบหลายบรรทัด และประกอบด้วยข้อมูลทั้งหมดที่อธิบายข้างต้นรวมถึงต่อไปนี้:

- ไตเร็กทอรี `$HOME` ของผู้ใช้
- ล็อกอินเชลล์ของผู้ใช้
- เนื้อหาของไฟล์ `.plan` ในไตเร็กทอรี `$HOME` ของผู้ใช้
- เนื้อหาของไฟล์ `.project` ในไตเร็กทอรี `$HOME` ของผู้ใช้

คำสั่ง `finger` ยังอาจ ถูกใช้เพื่อค้นหาผู้ใช้บนระบบรีโมต รูปแบบคือเพื่อระบุผู้ใช้ เป็น `User@Host` หากคุณไม่ระบุชื่อผู้ใช้ คำสั่ง `finger` จะจัดให้มีการแสดงรายการรูปแบบมาตรฐานบนระบบรีโมต

สร้างไฟล์ `.plan` และ `.project` โดยใช้เท็กซ์เอดิเตอร์ที่ต้องการของคุณและวางไฟล์ใน ไตเร็กทอรี `$HOME` ของคุณ คำสั่ง `finger` ใช้รูทีนย่อย `toascii` เพื่อแปลงอักขระภายนอก ขอบเขตอักขระ ASCII ปกติ เมื่อแสดงเนื้อหาของไฟล์ `.plan` และ `.project` คำสั่ง `finger` และ M- หน้าอักขระ แต่ละตัวที่ถูกแปลง

เมื่อคุณระบุผู้ใช้ด้วยพารามิเตอร์ `User` คุณสามารถระบุด้วยชื่อ นามสกุล หรือแอดเดสของผู้ใช้ เมื่อคุณระบุผู้ใช้ คำสั่ง `finger` ที่โฮสต์ที่ระบุ จะส่งกลับข้อมูลเกี่ยวกับ ผู้ใช้เหล่านั้นในรูปแบบยาวเท่านั้น

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับคำสั่ง `finger` โปรดดูที่ "การติดตั้ง TCP/IP" ใน *Networks and communication management*

## แฟล็ก

ไอเท็ม คำอธิบาย

- b แสดงรายการรูปแบบยาวอย่างย่อ
- f ระวังการพิมพ์บรรทัดส่วนหัวบนเอาต์พุต (บรรทัดแรกที่กำหนดฟิลด์ที่จะถูกแสดง)
- h ระวังการพิมพ์ไฟล์ `.project` ด้วยรูปแบบยาว และรูปแบบยาวอย่างย่อ
- i แสดงรายการที่มีเวลาเดินเครื่องเปล่าอย่างรวดเร็ว
- l แสดงรายการรูปแบบยาว
- m ถือว่าพารามิเตอร์ `User` ระบุ ID ผู้ใช้ (ใช้สำหรับค่าควบคุมการเข้าถึงอย่างรอบคอบ) *ไม่ใช่* ชื่อล็อกอินผู้ใช้
- p ระวังการพิมพ์ไฟล์ `.plan` ในรูปแบบยาว และรูปแบบยาวอย่างย่อ
- q แสดงรายการแบบรวดเร็ว
- s แสดงรายการรูปแบบสั้น
- w แสดงรายการรูปแบบสั้นแบบจำกัด

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
@Host	ระบุผู้ใช้ที่ล็อกอินทั้งหมดบนรีโมตโฮสต์
User	ระบุ ID ผู้ใช้โลคัล (ใช้สำหรับค่าควบคุมการเข้าถึงอย่างรอบคอบ) หรือชื่อล็อกอินผู้ใช้โลคัล ดังที่ระบุในไฟล์ /etc/passwd
User@Host	ระบุ ID ผู้ใช้บนรีโมตโฮสต์ โดยแสดงในรูปแบบยาว

## ตัวอย่าง

- ในการรับข้อมูลเกี่ยวกับผู้ใช้ทั้งหมดที่ล็อกอิน บนโฮสต์ alcatraz ให้ป้อน:

```
finger @alcatraz
```

ข้อมูล จะคล้ายกับที่แสดงต่อไปนี้:

```
[alcatraz.austin.ibm.com]
Login      Name          TTY Idle      When        Site Info
brown      Bob Brown     console  2d   Mar 15 13:19
smith      Susan Smith   pts0    11:   Mar 15 13:01
jones      Joe Jones     tty0    3     Mar 15 13:01
```

ผู้ใช้ brown ล็อกอินที่ console ผู้ใช้ smith ล็อกอินจาก pseudo teletype line pts0 และผู้ใช้ jones ล็อกอินจาก tty0

- ในการรับข้อมูลเกี่ยวกับผู้ใช้ brown ที่ alcatraz ให้ป้อน:

```
finger brown@alcatraz
```

ข้อมูล จะคล้ายกับที่แสดงต่อไปนี้:

```
Login name: brown
Directory: /home/brown   Shell: /home/bin/xinit -L -n Startup
On since May 8 07:13:49 on console
No Plan.
```

- ในการรับข้อมูลเกี่ยวกับผู้ใช้ brown ที่โลคัลโฮสต์ในรูปแบบสั้น ให้ป้อน:

```
finger -q brown
```

ข้อมูล จะคล้ายกับที่แสดงต่อไปนี้:

```
Login      TTY          When
brown      pts/6        Mon Dec17 10:58
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/finger	มีคำสั่ง finger
/etc/utmp	มีรายการของผู้ใช้ที่ล็อกอินขณะนี้
/etc/passwd	กำหนดแอดเดสส์ และโฮมไดเรกทอรี
/etc/security/passwd	กำหนดรหัสผ่านผู้ใช้
/var/adm/lastlog	มีเวลาล็อกอินล่าสุด
\$HOME/.plan	ไฟล์ทางเลือกที่มีรายละเอียดแบบหนึ่งบรรทัดของแผนงานของผู้ใช้
\$HOME/.project	ไฟล์ทางเลือกที่มีการมอบหมายโปรเจกต์ของผู้ใช้

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง hostname” ในหน้า 802

“fingerd Daemon” ในหน้า 580

ข้อมูลที่เกี่ยวข้อง:

คำสั่งสำหรับการแสดงข้อมูลเกี่ยวกับผู้ใช้ที่ล็อกอิน  
การสื่อสารและเน็ตเวิร์ก

---

## fingerd Daemon

### วัตถุประสงค์

จัดให้มีเซิร์ฟเวอร์ฟังก์ชันสำหรับคำสั่ง finger

### ไวยากรณ์

หมายเหตุ: โดยปกติ fingerd daemon เริ่มทำงานโดย inetd daemon รวมทั้งสามารถถูกควบคุมจากบรรทัดคำสั่ง โดยใช้คำสั่ง System Resource Controller (SRC)

```
/usr/sbin/fingerd [-s] [-f]
```

### คำอธิบาย

/usr/sbin/fingerd daemon คือโปรโตคอลอย่างง่ายที่จัดให้มีอินเทอร์เน็ตเฟสไปยังคำสั่ง finger ที่หลายๆ เน็ตเวิร์กไซต์ คำสั่ง finger ส่งกลับ รายงานสถานะของระบบปัจจุบัน หรือผู้ใช้ fingerd daemon รอรับการร้องขอ Transmission Control Protocol (TCP) requests ที่พอร์ต 79 ดังที่แสดงรายการในไฟล์ /etc/services และไฟล์ /etc/inetd.conf

เพื่อการรักษาความปลอดภัยของแต่ละไซต์ที่เกี่ยวกับ fingerd daemon โดยค่าดีฟอลต์จะไม่ส่งต่อการร้องขอ finger ใดๆ ไปยังระบบอื่นๆ ถ้าได้รับการร้องขอเพื่อส่งต่อ fingerd daemon จะตอบกลับด้วยความ Finger forwarding service denied ไปยังคำสั่ง finger ผู้ดูแลระบบมีตัวเลือกในการเปิดการส่งต่อ finger เป็นค่าดีฟอลต์เมื่อรัน fingerd daemon โดยใช้แฟล็ก -f

การเปลี่ยนแปลงใน fingerd daemon สามารถทำได้โดยใช้ System Management Interface Tool (SMIT) หรือ SRC หรือโดยการแก้ไขไฟล์ /etc/inetd.conf หรือไฟล์ /etc/services ไม่แนะนำให้ทำการป้อน fingerd ที่บรรทัดคำสั่ง fingerd ถูกเริ่มทำงานเป็นค่าดีฟอลต์เมื่อ ไม่ถูกทำเป็นความคิดเห็นในไฟล์ /etc/inetd.conf

โดย inetd daemon รับข้อมูลจาก ไฟล์ /etc/inetd.conf และไฟล์ /etc/services

หลังการเปลี่ยนไฟล์ /etc/inetd.conf หรือ /etc/services รันคำสั่ง refresh -s inetd หรือ kill -1 InetdPID เพื่อ แจ้งให้ inetd daemon ทราบถึงการเปลี่ยนแปลงในไฟล์คอนฟิกูเรชัน

fingerd daemon ควร มี ID ผู้ใช้ที่มีสิทธิเป็นอย่างน้อยที่สุดที่เป็นได้ nobody ID อนุญาตให้ใช้สิทธิอย่างน้อยที่สุด การกำหนด fingerd daemon ด้วย ID ผู้ใช้ nobody อนุญาตให้ daemon ถูกใช้ บนโฮสต์ของคุณ เปลี่ยนไฟล์ /etc/services เพื่อแสดง ID ผู้ใช้ที่คุณต้องการใช้

### การจัดการ fingerd Daemon ด้วย System Resource Controller

fingerd daemon เป็น เซิร์ฟเวอร์ย่อยของ inetd daemon ซึ่งเป็น ระบบย่อยของ SRC fingerd daemon เป็นสมาชิกของกลุ่มระบบย่อย tcpip SRC daemon นี้จะถูกเปิดใช้งานเมื่อมันถูกยกเลิกหมายเหตุในไฟล์ /etc/inetd.conf และสามารถจัดการโดยคำสั่ง SRC ต่อไปนี้:

ไอเท็ม	คำอธิบาย
startsrc	เริ่มต้นระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย
stopsrc	หยุดทำงานระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย
lssrc	รับสถานะระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย

## แฟล็ก

ไอเท็ม คำอธิบาย  
-s เปิดใช้งานการตีบกระดับของซ็อกเก็ต

ไอเท็ม คำอธิบาย  
-f เปิดเซอวิสเซอการส่งต่อ finger สำหรับ fingerd daemon นี้

## ตัวอย่าง

หมายเหตุ: อาร์กิวเมนต์สำหรับ fingerd daemon สามารถระบุโดยใช้ SMIT หรือโดยการแก้ไขไฟล์ `/etc/inetd.conf`

1. ในการเริ่มทำงาน fingerd daemon ให้พิมพ์:

```
startsrc -t finger
```

คำสั่งนี้เริ่มทำงานเซิร์ฟเวอร์ย่อย fingerd

2. ในการหยุดทำงาน fingerd daemon โดยทั่วไปให้พิมพ์:

```
stopsrc -t finger
```

คำสั่งนี้ อนุญาตให้การเชื่อมต่อที่ค้างอยู่เริ่มต้น และยังคงการเชื่อมต่อต่อไปเพื่อให้เสร็จสิ้น แต่จะไม่ให้มีการเชื่อมต่อใหม่

3. ในการบังคับให้หยุดทำงาน fingerd daemon และการเชื่อมต่อ fingerd ทั้งหมด ให้พิมพ์:

```
stopsrc -f -t finger
```

คำสั่งนี้จะยกเลิกการเชื่อมต่อที่ค้างอยู่ทั้งหมด และการเชื่อมต่อที่มีอยู่โดยทันที

4. ในการแสดงรายงานสถานะอย่างย่อเกี่ยวกับ fingerd daemon ให้พิมพ์:

```
lssrc -t finger
```

คำสั่งนี้ ส่งกลับชื่อของ daemon, ID กระบวนการ และสถานะ (แอ็คทีฟหรือไม่แอ็คทีฟ)

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง kill

คำสั่ง startsrc

TCP/IP daemons

คำสั่ง /etc/inetd.conf

---

## คำสั่ง fish

### วัตถุประสงค์

เล่นเกมไฟ go fish

### ไวยากรณ์

fish

### คำอธิบาย

วัตถุประสงค์ของเกม go fish คือสะสมชุด ไพ่สีใบที่มีค่านำใบเหมือนกัน คุณและโปรแกรม (คู่ต่อสู้ของคุณ) จะผลัดกันขอไพ่จากมือของอีกฝ่าย หากคู่ต่อสู้ของคุณมี ไพ่อย่างน้อยหนึ่งใบของค่าที่ร้องขอ คู่ต่อสู้ของคุณต้องส่งไพ่นั้นมาให้ หากไม่มี คู่ต่อสู้ของคุณจะพร้อม GO FISH! และคุณ จั่วไพ่จากกองไพ่ที่ยังไม่ได้แจก หากคุณจั่วได้ไพ่ที่คุณต้องการ คุณจะจั่วอีกครั้ง ขณะที่เรียงชุดชุดไพ่จะถูกเรียงบนโต๊ะ เล่นต่อไปจนกระทั่งไม่มีไพ่เหลือ ผู้เล่นที่มีชุดไพ่สูงสุดจะเป็นผู้ชนะ เกม คำสั่ง fish จะบอกให้คุณทราบผู้ชนะและ ออกจากเกม

คำสั่ง fish พร้อมด้วย instructions? ก่อนเล่นอีกครั้ง ในการดูวิธีเล่น ให้ป้อน Y (ใช่)

การป้อน p เมื่อคุณ เดินตาแรกของของคุณจะให้คุณเล่นเกมระดับผู้ชำนาญ ค่าตีพอลต์คือเกมระดับ สมัครเล่น

เมื่อเล่น go fish คุณป้อนไฟที่คุณต้องการ เมื่อคู่ต่อสู้ของคุณพร้อม:

you ask me for:

หากคุณกดปุ่ม Enter เท่านั้นเมื่อได้รับพร้อมท์ คุณ จะได้รับข้อมูลเกี่ยวกับจำนวนไพ่ในมือของคุณและ ในกองไพ่

เกมจะแสดง:

- ไพ่ที่มีในมือคุณขณะนั้น พร้อมชุดไพ่ที่คุณสะสมไว้ได้
- GO FISH! เมื่อ คุณหรือคู่ต่อสู้ขอไพ่ที่อีกฝ่ายไม่มี
- ไพ่ที่จั่วหลังจากพร้อมท์ GO FISH! t
- ไพ่ที่คู่ต่อสู้ของคุณขอจากคุณ
- ชุดไพ่ที่ครบสมบูรณ์ (ของคุณ หรือของคู่ต่อสู้)
- ไพ่ที่ร้องขอเมื่อคุณหรือคู่ต่อสู้ของคุณ เดาไพ่อื่น

### ตัวอย่าง

ต่อไปนี้เป็นตัวอย่างของการแสดงหน้าจอ fish :

your hand is: A 5 5 7 10 J Q

you ask me for: 5

I say "GO FISH!"

You draw A

I ask you for: 5

Made a book of 5's

I get another guess

I ask you for 6  
You say "GO FISH!"  
your hand is: A A 7 10 J Q  
you ask me for:

ในการออกจากเกมก่อนที่จะเล่นเสร็จ ให้กด ลำดับปุ่มอินเทอร์รัปต์ (Ctrl-C)

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/games	ตำแหน่งของเกมของระบบ

ข้อมูลที่เกี่ยวข้อง:

คำสั่งทางคณิตศาสตร์

คำสั่ง bj

คำสั่ง moo

คำสั่ง quiz

---

## คำสั่ง **flcopy**

### วัตถุประสงค์

คัดลอกไปยังและจากดิสเก็ต

### ไวยากรณ์

**flcopy** [ *-f Device* ] [ *-h | -r* ] [ *-t Number* ]

### คำอธิบาย

คำสั่ง **flcopy** คัดลอกดิสเก็ต (เปิดเป็น `/dev/rfd0`) ไปยังไฟล์ชื่อ **floppy** ที่สร้างในไดเรกทอรีปัจจุบัน จากนั้นพิมพ์ข้อความ: Change floppy, hit return when done จากนั้นคำสั่ง **flcopy** คัดลอกไฟล์ **floppy** ไปยัง ดิสเก็ต คุณสามารถระบุแฟล็ก `-f`, `-h`, `-r` หรือ `-tNumber` เพื่อแก้ไขลักษณะการทำงานของคำสั่ง **flcopy**

หมายเหตุ: คุณไม่สามารถใช้คำสั่ง **flcopy** เพื่อคัดลอกข้อมูลจากดิสเก็ต หนึ่งไปอีกดิสเก็ตหนึ่งที่มีขนาดต่างกัน

### แฟล็ก

ไอเท็ม	คำอธิบาย
-f Device	อนุญาตให้คุณระบุไดรฟ์อื่นนอกเหนือจาก /dev/rfd0
-h	ทำให้คำสั่ง flcopy เพื่อไฟล์ floppy ในไดเรกทอรีปัจจุบันและคัดลอกไปยัง /dev/rfd0
-r	ให้คำสั่ง flcopy ออกจากการทำงานหลังจากคัดลอก ดิสเก็ตไปยังไฟล์ floppy ในไดเรกทอรีปัจจุบัน
-t Number	ทำให้ Number ที่ระบุของแทร็ก แทนที่ที่ถูกคัดลอก แทร็กที่ถูกคัดลอกจะขึ้นต้นด้วยแทร็กแรกบน ดิสเก็ตเสมอ

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. ในการคัดลอก /dev/rfd1 ไปยังไฟล์ floppy ในไดเรกทอรีปัจจุบัน ให้ป้อน:

```
flcopy -f/dev/rfd1 -r
```

2. ในการคัดลอก 100 แทร็กแรกของดิสเก็ต ให้ป้อน:

```
flcopy -f/dev/rfd1 -t100
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/sbin/flcopy	มีคำสั่ง flcopy

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง format” ในหน้า 595

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง fd

## คำสั่ง flush-secdapclntd

### วัตถุประสงค์

คำสั่ง flush-secdapclntd ล้างค่าแคชสำหรับการประมวลผล secdapclntd daemon

### ไวยากรณ์

```
//usr/sbin/flush-secdapclntd
```

### คำอธิบาย

คำสั่ง flush-secdapclntd ล้างค่าแคชสำหรับการประมวลผล secdapclntd daemon

## ตัวอย่าง

1. ในการล้างค่าแคช `secdapclntd` daemon ให้พิมพ์:

```
/usr/sbin/flush-secdapclntd
```

## ไฟล์

ไอเท็ม

`/etc/security/ldap/ldap.cfg`

คำอธิบาย

มีข้อมูลที่จำเป็นสำหรับ `secdapclntd` daemon ในการเชื่อมต่อกับเซิร์ฟเวอร์

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `mksecdap`

คำสั่ง `start-secdapclntd`

คำสั่ง `restart-secdapclntd`

คำสั่ง `/etc/security/ldap/ldap.cfg`

---

## คำสั่ง `fmt`

### วัตถุประสงค์

จัดรูปแบบข้อความเมลก่อนการส่ง

### ไวยากรณ์

```
/usr/bin/fmt [ -Width ] [ File ... ]
```

### คำอธิบาย

คำสั่ง `fmt` เริ่มทำงานตัวจัดรูปแบบข้อความที่อ่านอินพุต *Files* ต่อเนื่องกัน (หรือ อินพุตมาตรฐานหากไม่ระบุ *Files*) จากนั้นสร้าง เวอร์ชันของเอาต์พุตที่มีความยาวขีดตามค่าของ `-Width` บนเอาต์พุตมาตรฐาน หากไม่มี ค่ารูกับแฟล็ก `-Width` จะใช้ค่าดีฟอลต์ 72 อักขระ ช่องว่างที่เริ่มต้นของ บรรทัดอินพุตจะถูกสงวนไว้ในเอาต์พุต เนื่องจากเป็นบรรทัดแบ่งและช่องว่างระหว่างคำ

คำสั่ง `fmt` โดยทั่วไป ถูกใช้จัดรูปแบบข้อความเมลเพื่อปรับปรุงลักษณะที่ปรากฏก่อนที่จะถูกส่ง อย่างไรก็ตาม คำสั่ง `fmt` ยังการเป็นประโยชน์สำหรับงาน การจัดรูปแบบอย่างเดียวย ตัวอย่างเช่น ภายในโมดแสดงภาพของโปรแกรมแก้ไขข้อความ เช่นเอดิเตอร์ `vi` คำสั่ง `!fmt` จัดรูปแบบย่อหน้า ดังเพื่อบรรทัดทั้งหมดถูกขีดเป็นค่าที่ระบุด้วยแฟล็ก `-Width` หากไม่มีค่าระบุกับแฟล็ก `-Width` จะใช้ค่าดีฟอลต์คือ 72 อักขระ โปรแกรมแก้ไขข้อความมาตรฐานมีความเหมาะสมมากกว่า `fmt` สำหรับการดำเนินการจัดรูปแบบที่ซับซ้อน

หมายเหตุ: ห้ามใช้คำสั่ง `fmt` ถ้าข้อความมีข้อความฝังอยู่หรือข้อมูลที่จัดรูปแบบล่วงหน้าจากไฟล์อื่น คำสั่งนี้จะจัดรูปแบบข้อมูล ส่วนหัวในข้อความที่ถูกฝัง และอาจเปลี่ยนแปลงรูปแบบของข้อความ ที่จัดรูปแบบล่วงหน้า

### แฟล็ก

ไอเท็ม	คำอธิบาย
ไฟล์	ระบุชื่อของไฟล์ที่จะจัดรูปแบบ
-Width	ระบุความยาวบรรทัด คาค่าดีฟอลต์สำหรับ Width คือ 72 อักขระ

## ตัวอย่าง

1. ในการจัดรูปแบบข้อความที่คุณสร้างขึ้นด้วย เมลเอ็ดเตอร์ให้ป้อน:

```
~| fmt
```

~| ถูกใส่ที่ขอบซ้ายของข้อความ หลังกจากคุณออกคำสั่ง ~| fmt ข้อความจะถูกจัดรูปแบบ คำว่า (continue) ถูกแสดงเพื่อ  
บ่งชี้ว่าคุณสามารถป้อน ข้อมูลเพิ่มเติมหรือส่งข้อความ

2. ในการจัดรูปแบบไฟล์และแสดงเอาต์พุตบน หน้าจอของคุณให้ป้อน:

```
fmt file1
```

ในตัวอย่างนี้ไฟล์ file1 ถูกจัดรูปแบบและแสดงบน หน้าจอของคุณ

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/fmt	มีคำสั่ง fmt

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mail

คำสั่ง nroff

คำสั่ง vi

แอ็พพลิเคชั่นเมลล์

## คำสั่ง fold

### วัตถุประสงค์

ตัดบรรทัดที่ยาวเกินสำหรับอุปกรณ์เอาต์พุตที่มีความกว้างคงที่

### ไวยากรณ์

```
fold [-b] [-s] [-w Width] [File...]
```

### คำอธิบาย

คำสั่ง fold คือตัวกรอง ที่ตัดบรรทัดสำหรับอุปกรณ์เอาต์พุตที่มีความกว้างที่จำกัด โดยคาค่าดีฟอลต์ คำสั่ง จะตัดเนื้อหาของอินพุตมาตรฐาน โดยแบ่งบรรทัดออกเป็นบรรทัดที่มีความกว้าง 80 (แปดสิบ) คุณยังสามารถระบุหนึ่งหรือหลายบรรทัดเป็นอินพุตในคำสั่ง

คำสั่ง fold จะแทรกอักขระบรรทัดใหม่ ในบรรทัดอินพุตเพื่อให้บรรทัดเอาต์พุตแต่ละบรรทัดมีความกว้างเท่าที่เป็นไปได้ โดยไม่เกินค่าที่ระบุโดยพารามิเตอร์ Width หากระบุแฟล็ก -b ความกว้างบรรทัดจะถูกนับเป็น ไบต์ หากไม่ระบุ -b :

- *Width* ถูกนับเป็นคอลัมน์ ตามที่กำหนดโดยตัวแปรสถานะแวดล้อม `LC_CTYPE`
- อักขระถอยกลับจะลดความยาวของ บรรทัดเอาต์พุตทีละ 1
- อักขระแท็บจะไปยังคอลัมน์ถัดไปโดยที่ ตำแหน่งคอลัมน์คือ 1 บวกผลคูณของ 8

คำสั่ง `fold` ยอมรับค่า `-w Width` ที่เป็นผลคูณของ 8 หากไฟล์มีแท็บ ในการใช้ค่าความกว้างอื่นๆ เมื่อไฟล์มีแท็บ ให้ใช้คำสั่ง `expand` ก่อนการใช้คำสั่ง `fold`

#### Notes:

1. คำสั่ง `fold` อาจส่งผล กับการขีดเส้นใต้ใดๆ ที่มีอยู่
2. คำสั่ง `fold` ไม่ แทรกอักขระบรรทัดใหม่ตรงระหว่างกลางของอักขระหลายไบต์แม้ว่า จะใช้แฟล็ก `-b`

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-b</code>	นับ <i>Width</i> เป็นไบต์ ค่าดีฟอลต์ คือนับเป็นคอลัมน์
<code>-s</code>	แบ่งบรรทัดหลังช่องว่างด้านขวาสุดที่อยู่ภายในขีดจำกัด <i>Width</i> หากเซกเมนต์บรรทัดเอาต์พุตมีอักขระช่องว่างใดๆ อยู่ ค่าดีฟอลต์คือ แบ่งบรรทัดเพื่อให้แต่ละบรรทัดเอาต์พุตมีความกว้างเท่าที่เป็นไปได้
<code>-w Width</code>	ระบุความกว้างบรรทัดสูงสุดเป็นค่าตัวแปร <i>Width</i> ค่าดีฟอลต์คือ 80

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	อินพุตไฟล์ทั้งหมดถูกประมวลผลเสร็จสมบูรณ์
>0	มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

ในการตัดบรรทัดของไฟล์ชื่อ `longlines` ให้มีความกว้าง 72 (เจ็ดสิบสอง) ให้ป้อน:

```
fold -w 72 longlines
```

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/fold</code>	มีคำสั่ง <code>fold</code>

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `expand`” ในหน้า 460

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `tab`

# คำสั่ง folder

## วัตถุประสงค์

เลือกและแสดงรายการโฟลเดอร์และข้อความ

## ไวยากรณ์

```
folder [ + Folder ] [ Message ] [ -all ] [ -nopack | -pack ] [ -nofast | -fast ] [ -norecurse | -recurse ] [ -print | -noprnt ] [ -header | -noheader ] [ -nototal | -total ] [ -push | -pop ] [ -list | -nolist ]
```

## คำอธิบาย

คำสั่ง **folder** เช็ตโฟลเดอร์ ปัจจุบันและข้อความปัจจุบันสำหรับโฟลเดอร์นั้น และแสดงข้อมูลเกี่ยวกับ โฟลเดอร์ของคุณ โดยค่าดีฟอลต์ คำสั่ง **folder** แสดง ชื่อโฟลเดอร์ปัจจุบัน จำนวนข้อความ ช่วงของหมายเลขข้อความ และข้อความปัจจุบัน

โฟลเดอร์ที่ระบุโดยแฟล็ก **+Folder** จะกลายเป็นโฟลเดอร์ปัจจุบัน ข้อความที่ระบุโดยพารามิเตอร์ **Message** จะกลายเป็นข้อความปัจจุบัน สำหรับโฟลเดอร์ ใช้แฟล็ก **-pack** เพื่อปรับหมายเลขข้อความในโฟลเดอร์

## แฟล็ก

ไอเท็ม	คำอธิบาย
-all	แสดงบรรทัดของข้อมูลเกี่ยวกับแต่ละโฟลเดอร์ในเมลไตรีกทอรีของคุณ
-fast	แสดงเฉพาะชื่อของโฟลเดอร์
+Folder	ระบุข้อมูลโฟลเดอร์ที่จะแสดง
-header	แสดงส่วนหัวคอลัมน์สำหรับข้อมูลโฟลเดอร์
-help	แสดงรายการไวยากรณ์คำสั่ง การสับเปลี่ยน (สลับ) ที่มีอยู่ และข้อมูลเวอร์ชัน หมายเหตุ: สำหรับ Message Handler (MH), ชื่อของแฟล็กนี้ระบุแบบเต็ม
-list Message	แสดงโฟลเดอร์ปัจจุบันตามด้วยเนื้อหาของโฟลเดอร์สแต็ก เช็ตข้อความที่ระบุเป็นข้อความปัจจุบัน ยกเว้นคุณระบุ แฟล็ก <b>+Folder</b> คำสั่ง จะเช็ตตั้งค่าที่ระบุสำหรับโฟลเดอร์ปัจจุบัน ใช้การอ้างอิงต่อไปนี้ เพื่อระบุข้อความ:  <i>Number</i> จำนวนของข้อความ  <i>cur or .</i> (จุด) ข้อความ ปัจจุบัน นี้คือ ค่าดีฟอลต์  <i>first</i> ข้อความแรกในโฟลเดอร์  <i>last</i> ข้อความสุดท้ายในโฟลเดอร์  <i>next</i> ข้อความหลังจากข้อความปัจจุบัน  <i>new</i> ข้อความใหม่ที่ถูกรสร้าง  <i>prev</i> ข้อความที่นำหน้าข้อความปัจจุบัน
-nofast	แสดงข้อมูลเกี่ยวกับแต่ละโฟลเดอร์ แฟล็กนี้คือ ค่าดีฟอลต์
-noheader	ไม่แสดงส่วนหัวคอลัมน์สำหรับข้อมูลโฟลเดอร์ แฟล็กนี้คือ ค่าดีฟอลต์
-nolist	ยับยั้งการแสดงผลเนื้อหาของโฟลเดอร์-สแต็ก แฟล็กนี้คือ ค่าดีฟอลต์
-nopack	ป้องกันการกำหนดหมายเลขใหม่ของข้อความในโฟลเดอร์ แฟล็กนี้คือ ค่าดีฟอลต์
-noprnt	ป้องกันการแสดงผลข้อมูลโฟลเดอร์ หากระบุแฟล็ก <b>-push</b> , <b>-pop</b> หรือ <b>-list</b> แฟล็ก <b>-noprnt</b> จะเป็นค่าดีฟอลต์
-norecurse	แสดงข้อมูลเกี่ยวกับโฟลเดอร์ระดับบนในโฟลเดอร์ ปัจจุบันของคุณเท่านั้น ข้อมูลเกี่ยวกับโฟลเดอร์ย่อยจะไม่ถูกแสดง แฟล็กนี้คือ ค่าดีฟอลต์

ไอเท็ม	คำอธิบาย
-nototal	ป้องกันการแสดงผลรวมของข้อความและโฟลเดอร์ทั้งหมดในโครงสร้างไดเรกทอรีเมลของคุณ เมื่อระบุแฟล็ก -all ค่าดีฟอลต์เป็นแฟล็ก -total มิฉะนั้นแฟล็ก -nototal จะเป็นค่าดีฟอลต์
-pack	กำหนดหมายเลขข้อความใหม่ในโฟลเดอร์ที่ระบุ การกำหนดหมายเลขใหม่ช่วยกำจัดช่องว่าง ในการกำหนดหมายเลขข้อความหลังจากข้อความถูกลบ
-pop	ลบโฟลเดอร์ออกจากด้านบนของโฟลเดอร์สแต็กและทำให้เป็นโฟลเดอร์ปัจจุบัน แฟล็ก +Folder ไม่สามารถระบุด้วยแฟล็ก -pop
-print	แสดงข้อมูลเกี่ยวกับโฟลเดอร์ หากระบุแฟล็ก -push, -pop หรือ -list แฟล็ก -noprint จะเป็นค่าดีฟอลต์ มิฉะนั้น แฟล็ก -print จะเป็นค่าดีฟอลต์
-push	ย้ายโฟลเดอร์ปัจจุบันไปที่ด้านบนของโฟลเดอร์สแต็ก และเซต โฟลเดอร์ที่ระบุเป็นโฟลเดอร์ปัจจุบัน หากไม่ระบุโฟลเดอร์แฟล็ก -push จะสลับโฟลเดอร์ปัจจุบันเป็นโฟลเดอร์ที่อยู่ด้านบน ของโฟลเดอร์สแต็ก
-recurse	แสดงข้อมูลเกี่ยวกับโฟลเดอร์และโฟลเดอร์ย่อยทั้งหมดในโฟลเดอร์ ปัจจุบันของคุณ
-total	แสดงข้อความและโฟลเดอร์ทั้งหมดในโครงสร้างเมลไดเรกทอรีของคุณ แฟล็ก -total ไม่แสดงข้อมูลสำหรับโฟลเดอร์ย่อย ยกเว้นคุณระบุแฟล็ก -recurse แฟล็ก -total เป็นค่าดีฟอลต์หากระบุแฟล็ก -all

## รายการโปรไฟล์

รายการต่อไปนี้ถูกป้อนลงในไฟล์ `UserMhDirectory/.mh_profile`:

ไอเท็ม	คำอธิบาย
Current-Folder:	ตั้งค่าโฟลเดอร์ปัจจุบันสำหรับค่าดีฟอลต์
Folder-Protect:	เซตระดับการปกป้องสำหรับโฟลเดอร์ไดเรกทอรีใหม่
Folder-Stack:	ระบุโฟลเดอร์สแต็ก
Isproc:	ระบุโปรแกรมที่ใช้แสดงเนื้อหาโฟลเดอร์
Path:	ระบุไดเรกทอรี MH ของผู้ใช้

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ตัวอย่าง

- ในการแสดงข้อมูลเกี่ยวกับโฟลเดอร์ปัจจุบัน ให้ป้อน:

```
โฟลเดอร์
```

ระบบตอบกลับด้วยข้อความที่คล้ายกับ ต่อไปนี้:

```
inbox+ has 80 messages (1-82); cur = 7; (others).
```

ในตัวอย่างนี้ โฟลเดอร์ปัจจุบันคือ inbox ในโฟลเดอร์มี 80 ข้อความ อยู่ในช่วง ตั้งแต่ข้อความ 1 ถึงข้อความ 82 หมายเลขข้อความปัจจุบันคือ 7

- ในการแสดงข้อมูลเกี่ยวกับโฟลเดอร์ทั้งหมด ให้ป้อน:

```
folder -all
```

ระบบตอบกลับด้วยข้อความที่คล้ายกับ ต่อไปนี้:

```
Folder # of messages (range); cur msg (other files)
inbox+ has 80 messages (1-82); cur= 7; (others).
test has 5 messages (1-5); cur= 5; (others).
```

Total= 85 messages in 2 folders

ในตัวอย่างนี้มี 2 โฟลเดอร์ที่มีทั้งหมด 85 ข้อความ โฟลเดอร์ปัจจุบันคือ inbox ซึ่งระบุโดย+ (เครื่องหมายบวก) ที่ตามหลัง

3. ในการทำให้โฟลเดอร์ test เป็นโฟลเดอร์ปัจจุบันและแสดงข้อมูลเกี่ยวกับ test ให้ป้อน:

```
folder +test
```

ระบบตอบกลับด้วยข้อความที่คล้ายกับ ต่อไปนี้:

```
test+ has 5 messages (1-5); cur = 5; (others)
```

4. ในการทำให้ข้อความ 2 เป็นข้อความปัจจุบันในโฟลเดอร์ปัจจุบัน ให้ป้อน:

```
folder 2
```

ระบบตอบกลับด้วยข้อความที่คล้ายกับ ต่อไปนี้:

```
test+ has 5 messages (1-5); cur = 2; (others)
```

5. ในการสร้างโฟลเดอร์ชื่อ group และทำให้เป็นโฟลเดอร์ปัจจุบัน ให้ป้อน:

```
folder +group
```

ระบบตอบกลับด้วยข้อความที่คล้ายกับ ต่อไปนี้:

```
Create folder "/home/dawn/Mail/group"? _
```

ให้ป้อน:

```
yes
```

ระบบตอบกลับด้วยข้อความที่คล้ายกับ ต่อไปนี้:

```
group+ has no messages.
```

6. ในการกำหนดหมายเลขข้อความใหม่ในโฟลเดอร์ปัจจุบัน ให้ป้อน:

```
folder -pack
```

ระบบตอบกลับด้วยข้อความที่คล้ายกับ ต่อไปนี้:

```
inbox+ has 80 messages (1-80); cur= 7; (others).
```

ในตัวอย่างนี้ข้อความถูกกำหนดหมายเลขใหม่เพื่อกำจัดช่องว่างในการกำหนดหมายเลขหลังจากข้อความถูกลบ

ไฟล์

ไอเท็ม  
\$HOME/.mh\_profile  
/usr/bin/folder

คำอธิบาย  
มีโปรไฟล์ผู้ใช้ MH  
มีคำสั่ง folder

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mhpah

คำสั่ง refile

ไฟล์ mh\_profile

แอปพลิเคชันเมล

---

## คำสั่ง folders

### วัตถุประสงค์

แสดงรายการโฟลเดอร์และข้อความทั้งหมดในเมลไต่เรียกทอรี

### ไวยากรณ์

```
folders [ +Folder ] [ Message ] [ -all ] [ -pack | -nopack ] [ -fast | -nofast ] [ -recurse | -norecurse ] [ -print | -noprnt ] [ -header | -noheader ] [ -total | -nototal ] [ -push | -pop ] [ -list | -nolist ]
```

### คำอธิบาย

คำสั่ง folders แสดงรายการ โฟลเดอร์และข้อความทั้งหมดในเมลไต่เรียกทอรีของคุณ คำสั่งนี้เทียบเท่ากับ คำสั่ง folder ที่ระบุด้วยแฟล็ก -all

### แฟล็ก

ไอเท็ม	คำอธิบาย
-all	แสดงบรรทัดของข้อมูลเกี่ยวกับแต่ละโฟลเดอร์ในเมลไต่เรียกทอรีของคุณ
-fast	แสดงเฉพาะชื่อของโฟลเดอร์
+Folder	ระบุข้อมูลโฟลเดอร์ที่จะแสดง
-header	แสดงส่วนหัวคอลัมน์สำหรับข้อมูลโฟลเดอร์ แฟล็กนี้คือ คำดีพอลต์
-help	แสดงรายการไวยากรณ์คำสั่ง การสับเปลี่ยน (สลับ) ที่มีอยู่ และข้อมูลเวอร์ชัน
	หมายเหตุ: สำหรับ Message Handler (MH), ชื่อของแฟล็กนี้ระบุแบบเต็ม
-list	แสดงโฟลเดอร์ปัจจุบันตามด้วยเนื้อหาของ โฟลเดอร์สแต็ก

ไอเท็ม  
Message

คำอธิบาย

เซ็ทข้อความที่ระบุเป็นข้อความปัจจุบัน ยกเว้นคุณระบุ แฟล็ก +Folder คำสั่ง จะเซ็ทตั้งค่าที่ระบุสำหรับโฟลเดอร์ปัจจุบัน ใช้การอ้างอิงต่อไปนี้ เพื่อระบุข้อความ:

Number จำนวนของข้อความ

cursor . (จุด)

ข้อความ ปัจจุบัน นี้คือ ค่าดีฟอลต์

first ข้อความแรกในโฟลเดอร์

last ข้อความสุดท้ายในโฟลเดอร์

next ข้อความหลังจากข้อความปัจจุบัน

new ข้อความใหม่ที่ถูกรสร้าง

prev ข้อความที่นำหน้าข้อความปัจจุบัน

-nofast แสดงข้อมูลเกี่ยวกับแต่ละโฟลเดอร์ แฟล็กนี้คือ ค่าดีฟอลต์

-noheader ไม่แสดงส่วนหัวคอลัมน์สำหรับข้อมูลโฟลเดอร์

-nolist ยับยั้งการแสดงเนื้อหาของโฟลเดอร์-สแต็ก แฟล็กนี้คือ ค่าดีฟอลต์

-nopack ป้องกันการกำหนดหมายเลขใหม่ของข้อความในโฟลเดอร์ แฟล็กนี้คือ ค่าดีฟอลต์

-noprint ป้องกันการแสดงผลโฟลเดอร์ หากระบุแฟล็ก -push, -pop หรือ -list แฟล็ก -noprint จะเป็นค่าดีฟอลต์

-norecurse แสดงข้อมูลเกี่ยวกับโฟลเดอร์ในเมลไอดีเร็กทอรีของคุณ ข้อมูลเกี่ยวกับโฟลเดอร์ย่อยจะไม่ถูกแสดง แฟล็กนี้คือ ค่าดีฟอลต์

-nototal ป้องกันการแสดงผลข้อความและโฟลเดอร์ทั้งหมดในโครงสร้างเมลไอดีเร็กทอรี ของคุณ

-pack กำหนดหมายเลขใหม่ของข้อความในโฟลเดอร์ การกำหนดหมายเลขใหม่ช่วยกำจัดช่องว่าง ในการกำหนดหมายเลขข้อความหลังจากข้อความถูกลบ

-pop ลบโฟลเดอร์ออกจากส่วนบนของโฟลเดอร์สแต็กและทำให้เป็น โฟลเดอร์ปัจจุบัน

-print แสดงจำนวนข้อความในแต่ละโฟลเดอร์ ข้อความปัจจุบัน สำหรับแต่ละโฟลเดอร์ และโฟลเดอร์ปัจจุบัน หากระบุแฟล็ก -push, -pop หรือ -list แฟล็ก -noprint จะเป็นค่าดีฟอลต์ มิฉะนั้น แฟล็ก -print จะเป็นค่าดีฟอลต์

-push ย้ายโฟลเดอร์ปัจจุบันไปที่ด้านบนของโฟลเดอร์สแต็ก และเซ็ท โฟลเดอร์ที่ระบุเป็นโฟลเดอร์ปัจจุบัน หากไม่ระบุโฟลเดอร์ แฟล็ก -push จะสลับโฟลเดอร์ปัจจุบันเป็นโฟลเดอร์ที่อยู่ด้านบน ของโฟลเดอร์สแต็ก

-recurse แสดงข้อมูลเกี่ยวกับโฟลเดอร์และโฟลเดอร์ย่อยทั้งหมดในโครงสร้าง เมลไอดีเร็กทอรีของคุณ

-total แสดงข้อความและโฟลเดอร์ทั้งหมดในโครงสร้างเมลไอดีเร็กทอรีของคุณ แฟล็ก -total ไม่แสดงข้อมูลสำหรับโฟลเดอร์ย่อย ยกเว้นคุณระบุแฟล็ก -recurse แฟล็ก -total เป็นดีฟอลต์

## รายการโปรไฟล์

รายการต่อไปนี้ถูกป้อนลงในไฟล์ `UserMhDirectory/.mh_profile`:

ไอเท็ม

Current-Folder:

Folder-Protect:

Folder-Stack:

Isproc:

Path:

คำอธิบาย

ตั้งค่าโฟลเดอร์ปัจจุบันสำหรับค่าดีฟอลต์

เซ็ทระดับการปกป้องสำหรับโฟลเดอร์ไอดีเร็กทอรีใหม่

ระบุโฟลเดอร์สแต็ก

ระบุโปรแกรมที่ใช้แสดงเนื้อหาโฟลเดอร์

ระบุไอดีเร็กทอรี MH ของผู้ใช้

## ตัวอย่าง

1. ในการแสดงข้อมูลเกี่ยวกับโฟลเดอร์ทั้งหมด ให้ป้อน:

โฟลเดอร์

ระบบตอบกลับด้วยข้อความที่คล้ายกับ ต่อไปนี้:

Folder # of messages (range); cur msg (other files)  
inbox+ has 80 messages (1-82); cur= 7; (others).  
test has 5 messages (1-6); cur= 5; (others).

Total= 85 messages in 2 folders.

ในตัวอย่างนี้มี 2 โฟลเดอร์ที่มีทั้งหมด 85 ข้อความ โฟลเดอร์ปัจจุบันคือ inbox ซึ่งระบุโดย+ (เครื่องหมายบวก) ตามหลัง

2. ในการแสดงรายชื่อของโฟลเดอร์ทั้งหมดให้ป้อน:

```
folders -fast
```

ระบบตอบกลับด้วยข้อความที่คล้ายกับ ต่อไปนี้:

```
inbox  
test
```

3. ในการปรับหมายเลขข้อความในโฟลเดอร์ทั้งหมดให้ป้อน:

```
folders -pack
```

ระบบตอบกลับด้วยข้อความที่คล้ายกับ ต่อไปนี้:

```
inbox+ has 80 messages (1-80); cur= 7; (others).  
test has 5 messages (1-5); cur= 5; (others).
```

ในตัวอย่างนี้ ข้อความ ในโฟลเดอร์ inbox และในโฟลเดอร์ test ได้ถูกปรับหมายเลขเพื่อกำจัดช่องว่างในการกำหนดหมายเลขข้อความ หลังจากข้อความถูกลบ

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>\$HOME/.mh_profile</code>	มีโปรไฟล์ผู้ใช้ MH
<code>/usr/bin/folders</code>	มีคำสั่ง folders

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `mhpath`

คำสั่ง `packf`

คำสั่ง `refile`

คำสั่ง `mh_profile`

---

## คำสั่ง `forcerpoffline`

### วัตถุประสงค์

บังคับให้เฟอร์รีดเมนออฟไลน์

### ไวยากรณ์

`forcerpoffline [-h] domain_name`

## คำอธิบาย

**ข้อควรระวัง:** ใช้คำสั่งนี้ด้วยความระมัดระวังอย่างยิ่ง

คำสั่ง `forcerpoffline` ต้องถูกใช้ต่อเมื่อโหนดอยู่ในสถานะออนไลน์ค้างอยู่ และคุณ ไม่สามารถนำกลับมาออนไลน์โดยใช้คำสั่ง `startprdomain` สถานการณ์นี้จะเกิดขึ้นถ้าคุณพยายามนำโหนดกลับมาออนไลน์ขณะที่โหนดกำลังดำเนินการภายใน quorum ถ้าคุณไม่แน่ใจว่าทำไม โหนดจึงติดอยู่ในสถานะออนไลน์ค้างอยู่ ให้รันคำสั่ง `ctsnap` ก่อนใช้คำสั่ง `forcerpoffline` เนื่องจากผลของการรันคำสั่ง `forcerpoffline` ระบบย่อยตัวจัดการรีซอร์สคอนฟิกูเรชัน (IBM.ConfigRM) และระบบย่อย RMC (`ctrmc`) ถูกรีไซเคิล

## พารามิเตอร์

`domain_name`

ระบุชื่อของเพียร์โหนดเมื่อกำหนดก่อนหน้านี้นี้ที่ถูก บังคับให้ออฟไลน์

## แฟล็ก

`-h` เขียนข้อความสั่งการใช้งานคำสั่งไปยังเอาต์พุตมาตรฐาน

## ไฟล์

ไฟล์ `/var/ct/cfg/current_cluster` และไฟล์ `/var/ct/cfg/default_cluster` ถูกแก้ไข

## เอาต์พุตมาตรฐาน

เมื่อแฟล็ก `-h` ถูกระบุ ข้อความสั่งการใช้งานคำสั่งนี้จะถูกเขียนไปยังเอาต์พุต มาตรฐาน

## สถานะออก

- 0 คำสั่งที่รันเป็นผลสำเร็จ
- 1 คำสั่งยุติการทำงานเนื่องจากมีข้อผิดพลาด RMC จำเป็น
- 2 คำสั่งยุติการทำงานเนื่องจากมีข้อผิดพลาดสำคัญในสคริปต์ คำสั่ง
- 3 คำสั่งยุติการทำงานเนื่องจากผู้ใช้ระบุแฟล็ก ไม่ถูกต้อง
- 4 คำสั่งยุติการทำงานเนื่องจากผู้ใช้ระบุพารามิเตอร์ไม่ถูกต้อง
- 5 คำสั่งยุติการทำงานเนื่องจากข้อผิดพลาดผู้ใช้ (การระบุชื่อ โดเมนที่ไม่มีอยู่ เป็นต้น)

## ความปลอดภัย

คุณต้องมีสิทธิ์การใช้งาน `root` เพื่อรันคำสั่งนี้

## ข้อมูลจำเพาะการนำไปปฏิบัติ

คำสั่งนี้ เป็นส่วนหนึ่งของชุดไฟล์ `rsct.basic.rte` สำหรับแพลตฟอร์ม AIX และ `rsct.basic-3.1.0.0-0.แพ็คเกจ.rpm` สำหรับ Linux, Solaris และ Windows โดย *แพลตฟอร์ม* ได้แก่ `i386`, `ppc`, `ppc64`, `s390` หรือ `x86_64`

## Location

/usr/sbin/rsct/bin/forcerpoffline

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง ctsnap

คำสั่ง startrpdomain

คำสั่ง stoprpdomain

---

## คำสั่ง format

### วัตถุประสงค์

ฟอร์แมตดิสก์หรืออ่าน/เขียนดิสก์สื่อบันทึก แบบอ็อปติคัล

### ไวยากรณ์

`format [-d Device] [-f] [-l]`

### คำอธิบาย

ข้อควรสนใจ: การฟอร์แมต ดิสก์หรืออ่าน/เขียนอ็อปติคัลดิสก์จะทำลายข้อมูลใดๆ ที่มีอยู่บน ดิสก์นั้น

คำสั่ง `format` จัดรูปแบบดิสก์ในดิสก์ไดรฟ์ที่ระบุโดยพารามิเตอร์ `Device` คำสั่ง `format` พิจารณาชนิดอุปกรณ์ ซึ่งเป็นแบบใดแบบหนึ่งต่อไปนี้:

- ดิสก์ความจุต่ำขนาด 5.25 นิ้ว (360KB) ที่มี 40x2 แทร็ก แต่ละแทร็กมี 9 เซ็กเตอร์
- ดิสก์ความจุสูงขนาด 5.25 นิ้ว (1.2MB) ที่มี 80x2 แทร็ก แต่ละแทร็กมี 15 เซ็กเตอร์
- ดิสก์ความจุต่ำขนาด 3.5 นิ้ว (720KB) ที่มี 80x2 แทร็ก แต่ละแทร็กมี 9 เซ็กเตอร์
- ดิสก์ความจุสูงขนาด 3.5 นิ้ว (1.44MB) ที่มี 80x2 แทร็ก แต่ละแทร็กมี 18 เซ็กเตอร์
- ดิสก์ความจุสูงขนาด 3.5 นิ้ว (2.88MB) ที่มี 80x2 แทร็ก แต่ละแทร็กมี 36 เซ็กเตอร์

ขนาดเซ็กเตอร์คือ 512 ไบต์สำหรับดิสก์ทุกชนิด

คำสั่ง `format` ฟอร์แมต ดิสก์ที่มีความจุสูงสุดที่สนับสนุนโดยดิสก์ไดรฟ์ ยกเว้นว่า พารามิเตอร์ `Device` จะระบุค่าความจุอื่น

คำสั่ง `format` ฟอร์แมต อ็อปติคัลดิสก์อ่าน/เขียน ทำให้ไดรฟ์สนับสนุนการตั้งค่าบิต Format Options Valid (FOV) ของส่วนหัวรายการ defect เป็น 0 ในการฟอร์แมตอ็อปติคัลดิสก์ อ่าน/เขียน ให้ใช้ชื่อของอ็อปติคัลไดรฟ์อ่าน/เขียน (เช่น `/dev/romd0`) หลังแฟล็ก `-d` สำหรับข้อมูลเพิ่มเติม โปรดดูที่การดำเนินการ `DKFORMAT` ของรูทีนย่อย `ioctl` ใน "scdisk SCSI Device Driver" ใน *Technical Reference: Kernel and Subsystems, Volume 2*

ก่อนทำการฟอร์แมตดิสก์หรืออ็อปติคัลดิสก์ อ่าน/เขียน คำสั่ง `format` จะพร้อมกับการตรวจสอบซึ่ง อนุญาตให้คุณสิ้นสุดการดำเนินการได้อย่างเรียบร้อย

## แฟล็ก

### ไอเท็ม

-d Device

### คำอธิบาย

ระบุอุปกรณ์ที่ใช้ฟอร์แมตดิสเก็ต หากชื่ออุปกรณ์ลงท้ายด้วยตัวอักษร h ไดรฟ์จะฟอร์แมตดิสเก็ตด้วยความจุสูง หากชื่ออุปกรณ์ลงท้ายด้วยตัวอักษร l ไดรฟ์จะฟอร์แมตดิสเก็ตด้วยความจุต่ำ อ้างอิงไปยังไฟล์พิเศษ fd เพื่อดูข้อมูลเกี่ยวกับชนิดอุปกรณ์ที่ถูกต้อง แฟล็กนี้ใช้กับคำสั่ง format เท่านั้น

**ข้อควรใส่ใจ:** หากดิสเก็ตไดรฟ์สนับสนุนความจุที่สูงกว่าความจุสูงสุดที่ซึ่งดิสเก็ตจะไม่สามารถทำงานได้ปกติ ความจุของดิสเก็ตควรถูกระบุไว้อย่างชัดเจนในพารามิเตอร์ Device (แฟล็ก -d Device) ของคำสั่ง format ตัวอย่างเช่น ในการฟอร์แมตดิสเก็ต 1MB บนดิสเก็ตไดรฟ์ 4MB ให้ระบุความจุดิสเก็ตในแฟล็ก -d ดังนี้:

```
-d /dev/fd0.9 for a 1MB diskette
```

-f

หากไม่ทำเช่นนี้จะทำให้เกิดข้อผิดพลาดในการอ่านและเขียน ฟอร์แมตดิสเก็ตโดยไม่ตรวจสอบเทร็กที่เสียหาย ดังนั้นการฟอร์แมตดิสเก็ตทำได้รวดเร็วยิ่งขึ้น แฟล็กนี้ใช้กับดิสเก็ตเท่านั้น ไม่สามารถใช้กับฮาร์ดดิสก์ อ่าน/เขียน ใช้กับคำสั่ง format เท่านั้น

-l

(L ตัวพิมพ์เล็ก) ฟอร์แมตดิสเก็ต 360KB ในดิสเก็ตไดรฟ์ขนาด 5.25 นิ้ว 1.2MB ฟอร์แมตดิสเก็ต 720KB ในดิสเก็ตไดรฟ์ขนาด 3.5 นิ้ว 1.4MB แฟล็กนี้ใช้กับดิสเก็ตเท่านั้น ไม่สามารถใช้กับฮาร์ดดิสก์ อ่าน/เขียน ใช้กับคำสั่ง format เท่านั้น

**ข้อควรสนใจ:** ดิสเก็ตไดรฟ์ 360KB อาจไม่สามารถอ่าน ดิสเก็ต 360KB ที่ถูกฟอร์แมตในไดรฟ์ 1.2MB

## พารามิเตอร์

### ไอเท็ม

Device

### คำอธิบาย

ระบุอุปกรณ์ที่มีดิสเก็ตที่ฟอร์แมต ดีฟอลต์คืออุปกรณ์ /dev/rfd0 สำหรับไดรฟ์ 0

## ตัวอย่าง

1. ในการฟอร์แมตดิสเก็ตในอุปกรณ์ /dev/rfd0 ให้ป้อน:

```
format -d /dev/rfd0
```

2. ในการฟอร์แมตดิสเก็ตที่ไม่มีการตรวจหาเทร็กที่เสียหาย ให้ป้อน:

```
format -f
```

3. ในการฟอร์แมตดิสเก็ต 360KB ในดิสเก็ตไดรฟ์ขนาด 5.25 นิ้ว 1.2MB ในอุปกรณ์ /dev/rfd1 ให้ป้อน:

```
format -l -d /dev/rfd1
```

4. ในการฟอร์แมตดิสเก็ตความจุต่ำ 3.5 นิ้ว (720KB) ให้ป้อน:

```
format -d /dev/fd0.9
```

5. ในการฟอร์แมตดิสเก็ตความจุสูง 3.5 นิ้ว (1.44MB) ให้ป้อน:

```
format -d /dev/fd0.18
```

6. ในการฟอร์แมตฮาร์ดดิสก์อ่าน/เขียนในอุปกรณ์ /dev/romd0 ให้ป้อน:

```
format -d /dev/romd0
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/sbin/format	มีคำสั่ง format
/dev/rfd*	ระบุพารามิเตอร์อุปกรณ์
/dev/fd*	ระบุพารามิเตอร์อุปกรณ์
/dev/romd*	ระบุพารามิเตอร์อุปกรณ์
/dev/omd*	ระบุพารามิเตอร์อุปกรณ์

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง flcopy” ในหน้า 583

“คำสั่ง fdformat” ในหน้า 534

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง fd

## คำสั่ง fortune

### วัตถุประสงค์

แสดงโชคชะตาสุ่มจากฐานข้อมูลโชคชะตา

### ไวยากรณ์

```
fortune [-] [-s|-l|-a [-w]] [File]
```

### คำอธิบาย

คำสั่ง fortune แสดง คำทำนายโชคชะตาจากไฟล์ fortunes.dat หรือไฟล์ ที่ระบุโดยพารามิเตอร์ *File* หลังจากแสดง คำทำนายโชคชะตา คำสั่ง fortune จะออกจากการทำงาน

### แฟล็ก

ไอเท็ม	คำอธิบาย
-	แสดงข้อมูลสรุปการใช้งาน
-a	แสดงประเภทของโชคชะตา
-l	แสดงโชคชะตาแบบยาวเท่านั้น
-s	แสดงโชคชะตาแบบสั้นเท่านั้น
-w	รอหลังจากแสดงคำทำนายโชคชะตาเพื่อให้เวลาผู้ใช้ได้อ่าน คำทำนาย

### ไฟล์

ไอเท็ม  
/usr/games  
/usr/games/lib/fortune/fortunes.dat

คำอธิบาย  
ตำแหน่งของเกมของระบบ  
ตำแหน่งของฐานข้อมูล fortune ดีฟอลต์

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง hangman” ในหน้า 778

ข้อมูลที่เกี่ยวข้อง:

คำสั่งทางคณิตศาสตร์

คำสั่ง moo

คำสั่ง ttt

---

## คำสั่ง forw

### วัตถุประสงค์

ส่งต่อข้อความ

### ไวยากรณ์

**forw** [ +Folder ] [ -draftfolder +Folder | -nodraftfolder ] [ Message ] [ -draftmessage Message ] [ -digest Name [ -issue Number ] [ -volume Number ] ] [ -form FormFile ] [ -editor Editor | -noedit ] [ -whatnowproc Program | -nowhatnowproc ] [ -filterFile ] [ -annotate [ -inplace | -notinplace ] | -noannotate ] [ -format | -noformat ] [ -help ]

### คำอธิบาย

คำสั่ง **forw** เริ่มต้นอินเทอร์เน็ตเฟสสำหรับการส่งต่อข้อความ โดยค่าดีฟอลต์ อินเทอร์เน็ตเฟสคำสั่ง **forw** :

- เปิดขึ้นเพื่อแก้ไขไฟล์ *UserMhDirectory/draft*
- พร้อมให้ผู้ใช้ป้อนข้อมูลการส่งต่อ ตามเพิ่มเพลตที่กำหนดในไฟล์ */etc/mh/mhl.forward*
- พร้อมดีให้ผู้ใช้ป้อนข้อความเพิ่มเติม ที่ควรแนบไปพร้อมกับข้อความที่ส่งต่อ

ในการดำเนินการแก้ไขไฟล์ *UserMhDirectory/draft* ให้เสร็จสมบูรณ์ให้กดลำดับ Ctrl-D คำสั่ง **forw** จะผนวกข้อความปัจจุบันจากโพลเดอร์ ปัจจุบันต่อท้ายไฟล์ **draft** หากคุณต้องการผนวกมากกว่า หนึ่งข้อความ ให้ใช้พารามิเตอร์ *Messages*

**หมายเหตุ:** บรรทัดของเส้นประ หรือบรรทัดว่างต้องอยู่ ระหว่างส่วนหัวและเนื้อความของข้อความสำหรับข้อความที่ ระบุเมื่อถูกส่ง

เมื่อออกจากเอดิเตอร์ คำสั่ง **forw** จะเริ่มต้นพร้อมดี What Now? กด คีย์ Enter เพื่อดูรายการของคำสั่งย่อย **whatnow** ที่มี คำสั่งย่อยเหล่านี้ทำให้คุณสามารถดำเนินการแก้ไขข้อความ แสดงข้อความ กำหนดการจัดการข้อความหรือจบการประมวลผลของคำสั่ง **forw**

คำสั่ง **forw** อนุญาตให้คุณ เปลี่ยนรูปแบบของข้อความที่ส่งต่อด้วยแฟล็ก **-form** โดยค่าดีฟอลต์ คำสั่งใช้รูปแบบข้อความดีฟอลต์ที่อยู่ในไฟล์ *UserMhDirectory/forwcomps* ของคุณ หากคุณไม่ได้กำหนดไฟล์ **forwcomps** ของคุณเอง จะใช้ไฟล์ */etc/mh/forwcomps* แทน

ใช้แฟล็ก `-annotate` เพื่อเพิ่มความคิดเห็น ในข้อความต้นฉบับด้วยข้อมูลการส่งต่อ เพื่อให้แน่ใจในการเพิ่มหมายเหตุ ประกอบให้ส่ง หมายเหตุที่ส่งต่อก่อนออกจากอินเตอร์เฟซคำสั่ง `forw`

หมายเหตุ: แฟล็ก `-annotate` ไม่ถูกเก็บรักษาไว้ระหว่างการเรียกใช้งานคำสั่ง `forw` หลายๆ ครั้ง บนแบบร่างเดียวกัน

## แฟล็ก

### ไอเท็ม

`-annotate`

#### คำอธิบาย

เพิ่มความคิดเห็นข้อความที่ส่งต่อด้วยบรรทัด:

Forwarded: Date

Forwarded: Addresses

`-digest Name`

ใช้แฟล็ก `-inplace` เพื่อกำหนดให้การเพิ่มความคิดเห็นอยู่ในตำแหน่ง คำนี้จะสวนลงไปถึงข้อความที่เพิ่มความคิดเห็นไว้

ใช้โปรแกรมอำนวยความสะดวก `digest` เพื่อสร้างฉบับใหม่สำหรับส่วนย่อยที่ระบุ โดยตัวแปร `Name` คำสั่ง `forw` ขยายสตริงรูปแบบในไฟล์ `components` (โดยใช้ กลไกสตริงรูปแบบเดียวกันกับที่ใช้โดยคำสั่ง `repl`) และเขียนแบบร่างใหม่โดยใช้อัลกอริทึมการครอบคลุมส่วนย่อยมาตรฐาน หลังจากแบบร่างได้ถูกเขียนใหม่แล้ว คำสั่ง `forw` จะเขียนรายการวอลุ่มและฉบับสำหรับส่วนย่อยและเริ่มทำงานเอดิเตอร์

`-draftfolder +Folder`

ยกเว้นว่าคุณระบุแฟล็ก `-form` ไว้ คำสั่ง `forw` จะใช้รูปแบบในไฟล์ `UserMhDirectory/digestcomps` หากไฟล์นี้ไม่มีอยู่ คำสั่งจะใช้ค่าดีฟอลต์ที่ระบุ ในไฟล์ `/etc/mh/digestcomps` วางข้อความแบบร่างในโฟลเดอร์ที่ระบุเฉพาะ หากคุณไม่ระบุ แฟล็กนี้ คำสั่ง `forw` จะเลือกโฟลเดอร์แบบร่าง ดีฟอลต์ตามข้อมูลที่ให้ในโปรไฟล์ Message Handler (MH) หากไม่ระบุ `+Folder` จะถือเป็น `Current-Folder` คุณสามารถกำหนด โฟลเดอร์แบบร่างดีฟอลต์ได้ในไฟล์ `$HOME/.mh_profile`

หมายเหตุ: ถ้า `-draftfolder +Folder` ตามด้วยพารามิเตอร์ `Message` ค่าจะ เหมือนกับการระบุแฟล็ก `-draftmessage`

`-draftmessage Message`

ระบุข้อความแบบร่าง หากคุณระบุ `-draftfolder` โดยไม่มีแฟล็ก `-draftmessage` ดังนั้นข้อความดีฟอลต์จะเป็น `new`

`-editor Editor`

ระบุเอดิเตอร์เริ่มต้นสำหรับการจัดเตรียมข้อความ

`-filter File`

ปรับรูปแบบแต่ละข้อความที่จะถูกส่งต่อ และวางข้อความที่ได้รับการปรับรูปแบบในข้อความแบบร่าง แฟล็ก `-filter` ยอมรับ รูปแบบที่ใช้โดยคำสั่ง `mhl`

`+Folder`

ระบุโฟลเดอร์ที่มีข้อความที่คุณต้องการส่งต่อ หากไม่ระบุโฟลเดอร์ จะใช้ `Current-Folder`

`-form FormFile`

แสดงเอาต์พุตคำสั่ง `forw` ในรูปแบบที่ ระบุโดยตัวแปร `FormFile` คำสั่ง `forw` ปฏิบัติต่อแต่ละบรรทัดในไฟล์ที่ระบุเป็นสตริงรูปแบบ หากระบุแฟล็ก `-digest` ด้วย คำสั่ง `forw` จะใช้รูปแบบที่ระบุโดยตัวแปร `File` เป็นรูปแบบสำหรับส่วนย่อย หากไม่ระบุแฟล็ก `-form` เมื่อใช้แฟล็ก `-digest` ไฟล์ตัวกรองส่วนย่อยจะกลายเป็นรูปแบบดีฟอลต์

`-format`

การใช้คำสั่ง `mhl` และไฟล์รูปแบบดีฟอลต์ จะปรับรูปแบบแต่ละข้อความที่จะถูกส่งต่อ และวางข้อความที่ได้รับการปรับรูปแบบในข้อความแบบร่าง หากมีไฟล์ `UserMhDirectory/mhl.forward` อยู่

`-help`

ไฟล์จะมีรูปแบบดีฟอลต์ มิฉะนั้น ไฟล์ `/etc/mh/mhl.forward` จะมีรูปแบบดีฟอลต์ แสดงรายการไวยากรณ์คำสั่ง การสับเปลี่ยน (สลับ) ที่มีอยู่ และข้อมูลเวอร์ชัน

`-inplace`

หมายเหตุ: สำหรับ MH ชื่อของแฟล็กนี้ ต้องถูกสะกดคำ

บังคับให้การเพิ่มความคิดเห็นเสร็จสิ้นในจุดที่กำหนดเพื่อรักษาลิงก์ไปยังข้อความ ที่เพิ่มความคิดเห็น

`-issue Number`

ระบุหมายเลขฉบับของส่วนย่อย หมายเลขฉบับดีฟอลต์ คือค่าที่มากกว่าค่าปัจจุบันอยู่หนึ่งของรายการ `DigestName-issue-list` ในไฟล์ `UserMhDirectory/context`

## ไอเท็ม Message

### คำอธิบาย

ระบุข้อความ คุณสามารถระบุข้อความต่างๆ ของข้อความ หรือข้อความเดียวได้ ใช้การอ้างอิงต่อไปนี่เมื่อระบุข้อความ:

*Number* จำนวนของข้อความ

*Sequence* กลุ่มข้อความที่ระบุโดยผู้ใช้ ค่าที่จัดจำ ได้แก่:

**ทั้งหมด** ข้อความทั้งหมดในโฟลเดอร์

**cur or . (จุด)**

ข้อความ ปัจจุบัน นี้คือ ค่าตีฟอลด์

**first** ข้อความแรกในโฟลเดอร์

**last** ข้อความสุดท้ายในโฟลเดอร์

**new** ข้อความใหม่ที่สร้างขึ้น

**next** ข้อความหลังจากข้อความปัจจุบัน

**prev** ข้อความก่อนหน้าข้อความปัจจุบัน

ข้อความตีฟอลด์คือข้อความปัจจุบันใน โฟลเดอร์ปัจจุบัน เมื่อคุณระบุหลายข้อความ ข้อความแรกที่ส่งต่อ จะกลายเป็นข้อความปัจจุบัน เมื่อคุณระบุโฟลเดอร์ โฟลเดอร์นั้นจะกลายเป็น โฟลเดอร์

ปัจจุบัน

ป้องกันการเพิ่มความคิดเห็นในข้อความต้นฉบับ แฟล็กนี้คือ ค่าตีฟอลด์

วางแบบร่างในไฟล์ *UserMhDirectory/draft*

ไม่แสดงการแก้ไขเริ่มต้น

ป้องกันการปรับรูปแบบของข้อความที่จะถูกส่งต่อ แฟล็กนี้คือ ค่าตีฟอลด์

ป้องกันการเพิ่มความเห็นเห็นในตำแหน่ง แฟล็กนี้คือ ค่าตีฟอลด์

ป้องกันการประมวลผลแบบโต้ตอบของคำสั่ง *forw* ด้วยแฟล็กนี้ จะไม่มีการแก้ไขเกิดขึ้น

ระบุหมายเลขขอลุ่มของส่วนย่อย หมายเลขขอลุ่มตีฟอลด์ คือค่าปัจจุบันของรายการ *DigestName-*

*volume-list* ในไฟล์ *UserMhDirectory/context*

สตาท์โปรแกรมที่ระบุเพื่อแนะนำคุณผานงาน การส่งต่อข้อมูล

หมายเหตุ: ถ้าคุณระบุคำสั่ง *whatnow* สำหรับ *Program* คำสั่ง *forw* เริ่มทำงานโปรแกรม *whatnow*

ภายใน แทนโปรแกรมที่มีชื่อไฟล์ *whatnow*

-noannotate

-nodraftfolder

-noedit

-noformat

-noinplace

-nowhatnowproc

-volume *Number*

-whatnowproc *Program*

## รายการโปรไฟล์

รายการต่อไปนี้เป็นถูกป้อนลงในไฟล์ *UserMhDirectory/.mh\_profile*:

### ไอเท็ม

Current-Folder:

Draft-Folder:

Editor:

fileproc:

mh1proc:

Msg-Protect:

Path:

whatnowproc:

### คำอธิบาย

ตั้งค่าโฟลเดอร์ปัจจุบันสำหรับค่าตีฟอลด์

ตั้งค่าตีฟอลด์โฟลเดอร์สำหรับแบบร่าง

เซตตีฟอลด์เอดิเตอร์

ระบุโปรแกรมที่ใช้เพื่อรีไฟล์ข้อความ

ระบุโปรแกรมที่ใช้กรองข้อความที่จะส่งต่อ

ตั้งการระดับของการปกป้องสำหรับไฟล์ข้อความใหม่

ระบุ *UserMhDirectory*

ระบุโปรแกรมที่ใช้เพื่อพร้อมคำถาม *What now?*

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

1. ในการส่งต่อข้อความปัจจุบันไปยังบุคคลอื่นให้ป้อน:

```
forw
```

ระบบพร้อมต์ให้คุณป้อนข้อมูลในฟิลด์ส่วนหัวในการข้ามฟิลด์ให้กดปุ่ม Enter คุณต้องป้อนข้อมูลในไฟล์ To: ระบบตอบกลับ ด้วย:

```
-----Enter initial text
```

ป้อนข้อความที่คุณต้องการให้แสดง หน้าข้อความที่จะถูกส่งต่อ และกดลำดับปุ่ม Ctrl-D ข้อความของข้อความที่ส่งต่อจะถูกแสดง และคุณได้รับพร้อมต์ที่มี What now? กด send หลัง พร้อมต์ What now? เพื่อส่งต่อข้อความ

2. ในการส่งต่อข้อความ 5 จาก โฟลเดอร์ inbox ให้ป้อน:

```
forw +inbox 5
```

## ไฟล์

### ไอเท็ม

```
/etc/mh/digestcomps
/etc/mh/mhl.forward
UserMhDirectory/digestcomps

UserMhDirectory/forwcomps
UserMhDirectory/mhl.forward

/usr/bin/forw
$HOME/.mh_profile
UserMhDirectory/draft
/etc/mh/forwcomps
```

### คำอธิบาย

กำหนดรูปแบบข้อความดีฟอลต์ MH เมื่อระบุแฟล็ก **-digest**  
มีตัวกรองข้อความ MH ดีฟอลต์  
ระบุรูปแบบข้อความดีฟอลต์ของผู้ใช้เมื่อระบุแฟล็ก **-digest** (หากมีอยู่จะแทนที่ตัวกรองข้อความ MH ดีฟอลต์)  
มีรูปแบบข้อความดีฟอลต์ของผู้ใช้  
มีตัวกรองข้อความดีฟอลต์ของผู้ใช้ (หากมีอยู่จะแทนที่ตัวกรองข้อความ MH ดีฟอลต์)  
มีรูปแบบที่รันได้ของคำสั่ง **forw**  
มีไฟล์ที่กำหนด MH สำหรับผู้ใช้แต่ละคนเอง  
มีแบบร่างที่สร้างสำหรับการแก้ไขข้อความ  
กำหนดคอมโพเนนต์สำหรับข้อความที่สร้างโดยคำสั่ง **forw**

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง anno

คำสั่ง whatnow

ไฟล์ mh\_alias

แอ็พพลิเคชันเมล

---

## คำสั่ง fpm

### วัตถุประสงค์

จัดการสิทธิ์บนคำสั่งและ daemons ที่เป็นของผู้ใช้ที่มีสิทธิ์ ด้วยสิทธิ์ setuid หรือ setgid

### ไวยากรณ์

```
fpm [-l level] [-f file] [[-c] [-p]] [-v] | [-s] | [-q] | [-?]
```

### คำอธิบาย

คำสั่ง fpm อนุญาตให้ผู้ดูแลระบบทำให้ระบบ แข็งแกร่งขึ้นโดยการปิดใช้งาน bit setuid และ setgid บนหลายๆ คำสั่งใน ระบบปฏิบัติการ คำสั่งนี้มีเจตนาลบสิทธิ์ setuid ออกจากคำสั่ง และ daemons ที่เป็นของผู้ใช้ที่มีสิทธิ์ แต่คุณยังสามารถกำหนดเอง เพื่อระบุ ความต้องการที่เจาะจงของสถานะแวดล้อมของคอมพิวเตอร์เฉพาะ

โปรแกรม setuid บนระบบปฏิบัติการของ AIX ฐานได้ถูก จัดกลุ่มเพื่ออนุญาตให้ระดับการทำให้เข้มแข็งขึ้น การจัดกลุ่มนี้ ช่วยให้ผู้ใช้ดูแลระบบ เลือกระบบของการทำให้เข้มแข็งขึ้นตามสถานะแวดล้อมระบบของตน นอกจากนั้น คุณสามารถใช้คำสั่ง fpm เพื่อกำหนดรายการของ โปรแกรมที่จำเป็นต้องถูกปิดใช้งานในสถานะแวดล้อมของคุณ คุณต้องตรวจทาน ระดับการของ ปิดใช้งาน และเลือกระดับที่ถูกต้องสำหรับสถานะแวดล้อมของคุณ

การเปลี่ยนสิทธิ์การเรียกใช้งานของคำสั่งและ daemons ด้วยคำสั่ง fpm จะมีผลต่อผู้ใช้ที่ไม่มีสิทธิ์ ทำการปฏิเสธการเข้าถึงของผู้ใช้ ไปยังคำสั่งและ daemons เหล่านี้ หรือของคำสั่งและ daemons นอกจากนั้น คำสั่งอื่นๆ ที่เรียกใช้ หรือขึ้นอยู่กับคำสั่ง และ daemons เหล่านี้สามารถได้รับผลกระทบ สคริปต์ที่ผู้ใช้สร้างใดๆ ที่ขึ้นอยู่กับคำสั่งและ daemons ที่มีสิทธิ์ ที่ถูกปรับเปลี่ยนโดย คำสั่ง fpm ไม่สามารถดำเนินการ ตามที่คาดหวังเมื่อรันโดยผู้ใช้ที่ไม่มีสิทธิ์ ให้ขอควรพิจารณาโดยสมบูรณ์สำหรับ ผลลัพธ์และ ผลกระทบร้ายแรงของการแก้ไขสิทธิ์ดีฟอลต์ของคำสั่งและ daemons

คุณต้องดำเนินการทดสอบอย่างเหมาะสมก่อนใช้คำสั่งนี้เพื่อเปลี่ยนแปลง สิทธิ์การเรียกใช้งานของคำสั่งและ daemons ใน สถานะแวดล้อมของคอมพิวเตอร์ วิกิตูตใดๆ หากคุณพบปัญหาในสถานะแวดล้อมที่สิทธิ์การเรียกใช้งาน ได้ถูกแก้ไข ให้เรียกคืน สิทธิ์ดีฟอลต์และสร้างปัญหานั้นอีกครั้ง ในสถานะแวดล้อมดีฟอลต์เพื่อให้แน่ใจว่าปัญหาที่เกิดขึ้นไม่ได้มาจากการไม่มีสิทธิ์ การเรียกใช้งานที่เหมาะสม

คำสั่ง fpm ยังจัดให้มีความสามารถในการเรียกคืน สิทธิ์ดีฟอลต์การติดตั้ง AIX เดิมโดยใช้แฟล็ก `-l default`

นอกจากนั้น คำสั่ง fpm จะบันทึกสถานะสิทธิ์ ของไฟล์ก่อนที่จะเปลี่ยนแปลงสิทธิ์ ไฟล์บันทึกการทำงาน fpm ถูกสร้างขึ้นในไฟล์ `/var/security/fpm/log/date_time` หากจำเป็น คุณสามารถใช้ไฟล์บันทึกการทำงานเหล่านั้นเพื่อเรียกคืนสิทธิ์ของไฟล์ของ ระบบที่บันทึกใน ไฟล์บันทึกการทำงานที่บันทึกไว้ก่อนหน้านี้

เมื่อใช้คำสั่ง fpm บนไฟล์ที่ขยายสิทธิ์ คำสั่งจะปิดใช้งานสิทธิ์ที่ขยายเพิ่ม แม้ว่าสิทธิ์ที่ขยายเพิ่มใดๆ ที่มีอยู่ก่อนหน้าการร้องขอ fpm จะยังคงอยู่ใน ACL ที่ขยายเพิ่ม

ไฟล์คอนฟิกูเรชันที่กำหนดเองสามารถถูกสร้างและถูกกำหนดเป็นส่วนหนึ่งของ ค่ากำหนดสูง กลาง ต่ำ และดีฟอลต์ รายการ ไฟล์สามารถระบุในไดเรกทอรี `/usr/lib/security/fpm/custom/high/*` ไดเรกทอรี `/usr/lib/security/fpm/custom/medium/*` และไดเรกทอรี `/usr/lib/security/fpm/custom/default/*` เมื่อต้องการใช้ประโยชน์ของ คุณลักษณะนี้ ให้สร้างไฟล์ที่มีรายการ ไฟล์ที่คุณต้องการให้ ประมวลผลโดยอัตโนมัตินอกเหนือจากรายการภายในคำสั่ง fpm เมื่อรันคำสั่ง fpm จะ ประมวลผล

รายการในไดเรกทอรีที่กำหนดเองที่สอดคล้องเช่นกัน เมื่อต้องการดูตัวอย่างของรูปแบบสำหรับไฟล์ที่กำหนดเองให้ดูไฟล์ `/usr/lib/security/fpm/data/high_fpm_list` รูปแบบดีฟอลต์สามารถดูได้ในไฟล์ `/usr/lib/security/fpm/data/default_fpm_list.example` สำหรับการกำหนดค่าของแฟล็ก `-l low` คำสั่ง `fpm` จะอ่านไฟล์เดียวกันในไดเรกทอรี `/usr/lib/security/fpm/custom/medium` แต่จะลบสิทธิ์ `setgid` เท่านั้น ในขณะที่แฟล็ก `-l medium` ลบ ทั้งสิทธิ์ `setuid` และ `setgid`

คำสั่ง `fpm` ไม่สามารถรันบนโฮสต์ที่เปิดใช้งาน TCB

## แฟล็ก

ไอเท็ม

`-l level`

คำอธิบาย

ระบุวาลิตี้ไฟล์ถูกเปลี่ยนตาม ระดับที่ระบุ

**-l high** การรักษาความปลอดภัยระดับสูง แฟล็กนี้ลบสิทธิ์ `setuid` และ `setgid` สำหรับระบบคอมพิวเตอร์ที่ล้มเหลวในหมวดหมู่ของการรักษาความปลอดภัย ระดับสูง แฟล็กนี้ใช้รายการของไฟล์ในไฟล์ `/usr/lib/security/fpm/data/high_fpm_list` และไฟล์ `/usr/lib/security/fpm/custom/high/*.*` เป็นอินพุตเป็นค่าดีฟอลต์ แต่อินพุตไฟล์ทางเลือกสามารถถูกเลือกโดย แฟล็ก `-f`

**-l medium**

การรักษาความปลอดภัยระดับกลาง แฟล็กนี้ลบสิทธิ์ `setuid` และ `setgid` สำหรับระบบคอมพิวเตอร์ที่ล้มเหลวในหมวดหมู่ของการรักษาความปลอดภัย ระดับกลาง แฟล็กนี้ใช้รายการของไฟล์ในไฟล์ `/usr/lib/security/fpm/data/med_fpm_list` และไฟล์ `/usr/lib/security/fpm/custom/med/*.*` เป็นอินพุตเป็นค่าดีฟอลต์ อินพุตไฟล์ทางเลือกสามารถเลือกด้วยแฟล็ก `-f`

**-l low**

การรักษาความปลอดภัยระดับต่ำ แฟล็กนี้ลบสิทธิ์ `setuid` เท่านั้นสำหรับระบบคอมพิวเตอร์ที่ล้มเหลวในหมวดหมู่ของการรักษาความปลอดภัยระดับต่ำ แฟล็กนี้ใช้รายการของไฟล์ในไฟล์ `/usr/lib/security/fpm/data/med_fpm_list` และไฟล์ `/usr/lib/security/fpm/custom/med/*.*` เป็นอินพุตเป็นค่าดีฟอลต์ อินพุตไฟล์ทางเลือกสามารถเลือกด้วยแฟล็ก `-f`

**-l default**

ส่งคืนคำสั่งระบบที่แก้ไขก่อนหน้านี้โดยคำสั่ง `fpm` เป็นสิทธิ์เริ่มแรกที่กำหนดให้ค่าดีฟอลต์ หากคำสั่ง ถูกปรับเปลี่ยนก่อนหน้านี้โดยใช้ระดับสูง กลาง หรือต่ำ อ็อพชันนี้ อ่านไฟล์ `/usr/lib/security/fpm/custom/default/*.*` และตั้งคาลิทธิที่กำหนดในไฟล์

`-s`

แสดงสถานะของการเปลี่ยนแปลงล่าสุดที่ทำโดยคำสั่ง `fpm` สถานะถูกเขียนในไฟล์ `/usr/lib/security/fpm/data/status_fpm` ระดับการรักษาความปลอดภัยถูกแทนด้วยเลขจำนวนเต็มตั้งแต่ 1-5

`-f file`

อนุญาตข้อกำหนดคุณลักษณะของรายการไฟล์แทนที่อินพุตไฟล์ โดยที่พารามิเตอร์ `file` คือชื่อไฟล์ที่มีรายการไฟล์ที่จะใช้เป็นอินพุต แฟล็กนี้ต้องใช้คู่กับแฟล็ก `-l highmediumlowdefault` หรือ `-c` เมื่อใช้ระดับสูง กลาง หรือต่ำ รูปแบบอินพุตไฟล์จะเป็นดังนี้:

`full_path/filename`

ตัวอย่างเช่น `For /usr/sbin/foo`

เมื่อใช้กับแฟล็ก `-l default` รูปแบบอินพุตไฟล์เป็นดังนี้:

`octet_permissions full_path/filename`

โดยต้องมีช่องว่างระหว่าง ตัวแปร `octet_permissions` และตัวแปร `full_path` ตัวอย่างเช่น `0750 /usr/sbin/foo`

`-c`

รูปแบบ `-f` อนุญาตสำหรับการควบคุมที่เจาะจงของ รายการไฟล์ที่จะได้รับผล ตรวจสอบสิทธิ์ไฟล์ แต่ไม่ดำเนินการใดๆ คำสั่ง `fpm` ส่งกลับค่า 0 หากไม่พบไฟล์ที่ไม่เป็นไปตามที่กำหนด หากมีอย่างน้อยหนึ่งไฟล์ที่มีสิทธิ์ที่ไม่เป็นไปตามที่กำหนด อ็อพชันนี้จะแสดงไฟล์ที่ไม่เป็นไปตามที่กำหนด และส่งคืนค่า 1 แฟล็กนี้ต้องใช้กับอ็อพชัน `-l level` ตัวอย่างเช่น หากใช้แฟล็ก `-c` และ `-l high` ร่วมกัน คำสั่ง `fpm` จะตรวจสอบไฟล์ที่แสดงในไฟล์ `/usr/lib/security/fpm/data/high_fpm_list` และลบสิทธิ์ `setuid` และ `setgid` ออก แฟล็ก `-f file` ยังสามารถใช้กับอ็อพชัน `-c`

`-v`

เอาต์พุตแบบละเอียด

`-p`

แสดงตัวอย่างการเปลี่ยนแปลงคำสั่ง `fpm` ที่จะทำให้ แต่ไม่ดำเนินการใดๆ แฟล็กนี้ต้องใช้ร่วมกับแฟล็ก `-l level`

ไอเท็ม  
-q  
-?

คำอธิบาย  
ออกจากโหมดซึ่งจะย่อเอาต์พุตให้เล็กสุด และไม่แสดงค่าเตือน  
พิมพ์คำสั่งการใช้งาน

## สถานะออก

ไอเท็ม	คำอธิบาย
0	สำเร็จ
Non-zero	ล้มเหลว หรือล้มเหลวบางส่วน ใช้แฟล็ก -v เพื่อดูรายละเอียดเพิ่มเติม

## ความปลอดภัย

คำสั่ง **fpm** ลดจำนวนคำสั่งที่มีสิทธิ์ **setuid** และ **setgid**

### ตัวอย่าง

1. ในการใช้การตั้งค่าการรักษาความปลอดภัยระดับต่ำของ **fpm** ให้ป้อน:

```
fpm -l low
```

คำสั่งนี้ยังประมวลผลรายการไฟล์ใดๆ ในไดเรกทอรี **/usr/lib/security/fpm/custom/med/**

2. ในการตรวจสอบว่าคำสั่งระบบถูกตั้งค่าเป็น **fpm** สิทธิระดับต่ำ หรือไม่ ให้ป้อน:

```
fpm -c -l low
```

คำสั่งนี้รายงานให้ทราบว่าไฟล์ใดๆ ที่มีสิทธิ์ไม่สอดคล้องกัน

3. ในการเรียกคืนค่าดีฟอลต์สิทธิ์เริ่มแรกแบบเดิม ให้ป้อน:

```
fpm -l default
```

คำสั่งนี้ยังประมวลผลรายการไฟล์ใดๆ ในไดเรกทอรี **/usr/lib/security/fpm/custom/default/**

4. ในการแสดงรายการ หรือแสดงตัวอย่างของสิทธิ์ที่เปลี่ยนแปลง หรือที่จะถูกดำเนินการโดย ทำให้ระบบเข้ากันได้กับการรักษาความปลอดภัยระดับสูงของคำสั่ง **fpm** โดยไม่มีการเปลี่ยนแปลงสิทธิ์ไฟล์ใดๆ ให้ป้อน:

```
fpm -l high -p
```

คำสั่งนี้ยังแสดงตัวอย่างรายการไฟล์ใดๆ ที่แสดงในไดเรกทอรี **/usr/lib/security/fpm/custom/high/**

5. ในการใช้การตั้งค่าการรักษาความปลอดภัยระดับสูงของ **fpm** ให้ป้อน:

```
fpm -l high
```

คำสั่งนี้ยังประมวลผลรายการไฟล์ใดๆ ในไดเรกทอรี **/usr/lib/security/fpm/custom/high/**

6. ในการแสดงสถานะปัจจุบันของระบบที่เปลี่ยนแปลงโดยคำสั่ง **fpm** ให้ป้อน:

```
fpm -s
```

7. หากคำสั่ง **fpm -l level** รันในวันที่ 7 มกราคม 2550 เวลา 8:00 น. ดังนั้นสถานะสิทธิ์ของไฟล์ที่ได้รับผลจะถูกบันทึกโดยคำสั่ง **fpm** ก่อนทำการเปลี่ยนแปลงใดๆ ในการเรียกคืนสิทธิ์ไฟล์ให้เป็นสถานะของวันที่ 7 มกราคม 2550 เวลา 8:00 น. ให้ป้อน:

```
fpm -l default -f /var/security/fpm/log/01072007_08:00:00
```

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/lib/security/fpm/data/default_list_example</code>	มีสิทธิ์ตั้งต้นค่าดีฟอลต์และไฟล์
<code>/usr/lib/security/fpm/data/high_fpm_list</code>	มีรายการไฟล์ที่สิทธิ์สามารถ เปลี่ยนแปลงโดยใช้แฟล็ก <code>-I high</code>
<code>/usr/lib/security/fpm/data/med_fpm_list</code>	มีรายการไฟล์ที่สิทธิ์สามารถ เปลี่ยนแปลงโดยใช้แฟล็ก <code>-I medium</code> หรือ <code>-I low</code>
<code>/usr/lib/security/fpm/custom/high/*</code>	ไฟล์ในไดเรกทอรีนี้สามารถใช้เป็นอินพุตที่ผู้ใช้ กำหนดค่าเองเมื่อระดับ <code>-I high</code> ถูกเลือก ไฟล์เหล่านี้ ต้องมีรายการไฟล์ที่คำสั่ง <code>fpm</code> จะ ลบสิทธิ์ <code>setuid</code> และ <code>setgid</code> ออก
<code>/usr/lib/security/fpm/custom/medium/*</code>	ไฟล์ในไดเรกทอรีนี้ทำหน้าที่เหมือนกับไดเรกทอรี ระดับสูง แต่ใช้กับแฟล็ก <code>-I medium</code> และแฟล็ก <code>-I low</code>
<code>/usr/lib/security/fpm/custom/default/*</code>	ไฟล์ในไดเรกทอรีนี้ทำหน้าที่เหมือนกับไดเรกทอรี ระดับสูง แต่ใช้กับแฟล็ก <code>-I default</code>
<code>/usr/lib/security/fpm/data/status_fpm</code>	หมายเหตุ: ไฟล์เหล่านี้ต้องอยู่ในรูปแบบเดียวกัน กับไฟล์ <code>/usr/lib/security/fpm/data/default_list_example</code> มีสถานะของสิทธิ์ไฟล์ที่เปลี่ยนแปลง จาการรันล่าสุดของคำสั่ง <code>fpm</code>
<code>/var/security/fpm/log/date_time</code>	มีรายการของไฟล์ที่เปลี่ยนแปลงโดยคำสั่ง <code>fpm</code> ที่สอดคล้องกับข้อมูลและเวลาที่คำสั่ง รัน ไฟล์นี้สามารถใช้เป็นอินพุตไฟล์ของแฟล็ก <code>-f</code> เพื่อเรียกคืนสิทธิ์ให้กับอินสแตนซ์นี้

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `aixpert`

ผู้เชี่ยวชาญด้านการรักษาความปลอดภัย

---

## คำสั่ง `asa`, `fpr`

### วัตถุประสงค์

พิมพ์ไฟล์ FORTRAN ในรูปแบบของเครื่องพิมพ์รายบรรทัด

### ไวยากรณ์

```
{ asa | fpr } [ File ... ]
```

### คำอธิบาย

คำสั่ง `asa` และ `fpr` พิมพ์ไฟล์ FORTRAN เพื่อให้สอดคล้องกับแนวทางพริ้นเตอร์รายบรรทัด ของระบบปฏิบัติการนี้ คำสั่งทั้งสองทำงานเหมือนตัวกรองในการแปลงไฟล์ ที่จัดรูปแบบตามแนวทางรหัสควบคุมการแสดงผล FORTRAN ให้เป็น ไฟล์ที่จัดรูปแบบตามแนวทางเครื่องพิมพ์รายบรรทัด

ตัวแปร *File* ระบุชื่อของอินพุตไฟล์ที่คำสั่ง `asa` และ `fpr` จะอ่านแทนอินพุตมาตรฐาน คำสั่ง `asa` และ `fpr` อ่านไฟล์ แทนที่ อักขระรหัสควบคุมการแสดงผลด้วยอักขระที่อักขระรู้จัก และพิมพ์ไฟล์ไปยังเอาต์พุตมาตรฐาน

คำสั่งทั้งสองอ่านอักขระแรกของแต่ละบรรทัด จากอินพุตไฟล์ แปลอักขระ และเว้นบรรทัดตาม นิยามของอักขระแรก หากอักขระตัวแรกเป็น ช่องว่าง, 0, เส้นประ (-), 1 หรือเครื่องหมายบวก (+) คำสั่งจะทำต่อไปนี้:

ไอเท็ม	คำอธิบาย
Blank	เลื่อนการขึ้นบรรทัดไปหนึ่งบรรทัด และพิมพ์บรรทัดอินพุต
0	เลื่อนการขึ้นบรรทัดไปสองบรรทัด และพิมพ์บรรทัดอินพุต
-	เลื่อนการขึ้นบรรทัดไปสามบรรทัด และพิมพ์บรรทัดอินพุต
1	เลื่อนการขึ้นบรรทัดไปที่ด้านบนของหน้าถัดไป
+	ไม่เลื่อนการขึ้นบรรทัดและเริ่มพิมพ์บรรทัดอินพุต ในที่ว่างแรกของเอาต์พุตไฟล์

คำสั่งแปลบรรทัดว่างเหมือนอักขระ ตัวแรกของบรรทัดคือเป็นช่องว่าง และลบช่องว่างที่แสดงเป็นอักขระ รหัสควบคุมการแสดงผล โดยถือเสมือนบรรทัดขึ้นต้นด้วยอักขระอื่นนอกเหนือจากอักขระ ควบคุมที่กำหนดเสมือนว่าขึ้นต้นด้วยอักขระช่องว่าง อักขระตัวแรกของบรรทัดจะไม่ถูกพิมพ์ หากมีบรรทัดอื่นแสดง การวินิจฉัยที่เหมาะสม จะแสดงในข้อผิดพลาดมาตรฐาน

หมายเหตุ: ผลลัพธ์จะไม่ถูกกำหนดสำหรับบรรทัดอินพุตที่เกิน 170 อักขระ

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

- ใช้คำสั่ง `fpr` ใน ลักษณะต่อไปนี้เพื่อเปลี่ยนอักขระรหัสควบคุมการแสดงผลในไฟล์ `a.out` ที่สร้างโดยคอมไพเลอร์ FORTRAN ให้เป็นอักขระรหัสควบคุมการแสดงผล และพิมพ์ไฟล์ผลลัพธ์:
 

```
a.out | fpr | qprt
```
- ใช้คำสั่ง `asa` ใน ลักษณะต่อไปนี้เพื่อรันไฟล์ `f77.output` จาก คำสั่ง `asa` เพื่อเปลี่ยนอักขระรหัสควบคุมการแสดงผล จาก FORTRAN เป็นระบบปฏิบัติการและพิมพ์ไฟล์ผลลัพธ์
 

```
asa f77.output | qprt
```

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/ucb/fpr</code>	มีคำสั่ง <code>fpr</code>
<code>/usr/bin/asa</code>	มีคำสั่ง <code>asa</code>

## คำสั่ง `fractrl`

### วัตถุประสงค์

ควบคุมและกำหนดค่า FRCA

### ไวยากรณ์

```
fractrl { load | unload } fractrl open Ip_Address Port [ Virtual_Host ] Server_Name Virtual_Root Log_File fractrl close
Ip_Address Port [ Virtual_Host ] fractrl loadfile Ip_Address Port [ Virtual_Host ] Document_Root File ... fractrl stats [
reset ] [ Interval ] fractrl logging Ip_Address Port [ Virtual_Host ] { on | off } [ Format ] [ CPU_Id ] fractrl { start | stop }
```

*Ip\_Address Port [ Virtual\_Host ] frcactrl revaltimeout Ip\_Address Port [ Virtual\_Host ] [ Seconds ] frcactrl pctionintr [ Percentage ] frcactrl set { option=value } frcactrl getfrcactrl default [ option ]*

## คำอธิบาย

คำสั่ง **frcactrl** ควบคุมและกำหนดค่า ส่วนขยายเคอร์เนล FRCA ส่วนขยายเคอร์เนลต้องถูกโหลดก่อนเริ่มทำงาน เว็บเซิร์ฟเวอร์ใดๆ ที่ต้องการใช้ FRCA

## คำสั่งย่อย

**load** โหลดส่วนขยายเคอร์เนล FRCA หากยังไม่โหลด

**unload** ยกเลิกการโหลดส่วนขยายเคอร์เนล FRCA หากโหลดอยู่

**open** *Ip\_Address Port [ Virtual\_Host ] Server\_Name Virtual\_Root Log\_File*  
เปิดและกำหนดคอนฟิกอินสแตนซ์ FRCA ภายใต้ชื่อ *Server\_Name* สำหรับ IP แอดเดรส *Ip\_Address* บนพอร์ต *Port* พารามิเตอร์ *Virtual\_Root* ระบุไดเรกทอรีที่ข้อมูลเว็บเริ่มต้น การร้องขอจะถูกบันทึกการทำงาน ในไฟล์ที่ระบุโดย *Log\_File* ชื่อไฟล์นี้ต้อง ระบุแบบเต็ม

หมายเหตุ: FRCA สนับสนุนหนึ่งไฟล์บันทึกการทำงานเท่านั้น เมื่อรัน มากกว่าหนึ่งเว็บเซิร์ฟเวอร์บนระบบที่มี FRCA การร้องขอทั้งหมดจะถูกบันทึกการทำงาน ลงในไฟล์เดียวกัน

**close** *Ip\_Address Port [ Virtual\_Host ]*

ปิดอินสแตนซ์ FRCA ที่สัมพันธ์กับ IP แอดเดรสและ พอร์ตที่ระบุ

**loadfile** *Ip\_Address Port [ Virtual\_Host ] Document\_Root File ...*

โหลดไฟล์ที่ระบุไปยัง FRCA / Network Buffer Cache IP และหมายเลขพอร์ตที่อินสแตนซ์ FRCA ได้ถูกเปิดขึ้นก่อนหน้าต้อง ถูกระบุในที่นี้พร้อมกับ root เอกสารและไฟล์ที่จะถูกโหลด

**stats** [ **reset** ] [ *Interval* ]

แสดงสถิติ FRCA คำสั่งย่อย **reset** ทางเลือก จะล้างค่า (ทำให้เป็นศูนย์) สถิติ คุณสามารถแสดงสถิติในช่วงเวลา ปกติ โดยการระบุระยะเวลาเป็นวินาทีด้วยพารามิเตอร์ *Interval*

**logging** *Ip\_Address Port [ Virtual\_Host ] { on | off } [ Format ] [ CPU\_Id ]*

เปิดหรือปิดการบันทึกการร้องขอที่รับหน้าที่โดยอินสแตนซ์ FRCA ที่โยงอยู่กับ *Ip\_Address* และ *Port* ที่ระบุ รูปแบบสามารถเป็นหนึ่งใน CLF, V-CLF หรือ ECLF (Common Log Format, Virtual Host & CLF, Extended CLF) เเรดการบันทึกการทำงาน FRCA ยังถูกโยงกับ CPU เฉพาะโดยการระบุพารามิเตอร์ *CPU\_Id* ทางเลือก บนเครื่องมัลติโพรเซสเซอร์

**start** *Ip\_Address Port [ Virtual\_Host ]*

เปิดใช้งานเคอร์เนลที่รับค่าเอ็นจินเพื่อรับหน้าที่ให้บริการการร้องขอที่ส่งไปยัง IP และพอร์ตที่ระบุ

**stop** *Ip\_Address Port [ Virtual\_Host ]*

ปิดใช้งานเคอร์เนลที่รับค่าเครื่องสำหรับ IP และพอร์ตที่ระบุ

**revaltimeout** *Ip\_Address Port [ Virtual\_Host ] [ Seconds ]*

เปลี่ยนค่าการหมดเวลาใช้งานการตรวจสอบความถูกต้องใหม่สำหรับอินสแตนซ์ FRCA ที่แอดเดรส และพอร์ตที่ระบุ ค่าการหมดเวลาใช้งานต้องระบุเป็นวินาที

## pctonintr [ Percentage ]

ควบคุมเปอร์เซ็นต์ของเวลา CPU ที่สามารถใช้ในอินเทอร์พรีตคอนเท็กซ์ หากค่านี้ต่ำเกินไป FRCA จะส่งการร้องขอไปยังเว็บเซิร์ฟเวอร์บ่อยขึ้น เนื่องจากเรียกใช้งานในอินเทอร์พรีตคอนเท็กซ์เสมอ ค่าใด  $\geq 100$  จะส่งผลให้ FRCA ให้บริการทุกการร้องขอที่ แคมในแคช FRCA

## set {option=value}

ตั้งค่าอ็อปชัน FRCA ที่ระบุให้เป็นค่า อ็อปชันที่ขณะนี้ ใช้ได้เท่านั้นคือ `frca_hashsz` ซึ่งตั้งค่าจำนวนสล็อต ในตารางการแฮช FRCA ให้เป็นค่าที่ระบุ ค่าดีฟอลต์ของ `frca_hashsz` คือ 12841 หากเปลี่ยนแปลง ค่าที่ใช้ต้องเป็นจำนวนเฉพาะเนื่องจาก ค่านี้ให้ผลลัพธ์การแจกแจงของรายการตารางการแฮชที่สม่ำเสมอ

**get** แสดงอ็อปชัน FRCA ทั้งหมดที่มีพร้อมกับค่าปัจจุบัน หนึ่งอ็อปชันชื่อ `frca_hashsz` เท่านั้นที่มีอยู่ขณะนี้

## default [option]

ตั้งค่าของอ็อปชันทั้งหมดเป็นค่าดีฟอลต์เมื่อไม่มี การระบุชื่ออ็อปชัน หากระบุชื่ออ็อปชัน จะตั้ง ค่าของอ็อปชันที่ระบุ ให้เป็นค่าดีฟอลต์เท่านั้น

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ตัวอย่าง

1. ต่อไปนี้คือตัวอย่างของการใช้คำสั่งย่อย `open`:

```
frctrll open 9.1.1.1 80 ici imgcache01 /htdocs /logs/frca.log bin
frctrll open 9.1.1.2 80 ici imgcache02 /htdocs /logs/frca.log bin
```

ใน ตัวอย่างข้างต้น "ici" คือชื่อโฮสต์เสมือนซึ่งสามารถใช้เพื่อเข้าถึง หนึ่งใน `imgcache01` หรือ `imgcache02` ที่ทำมีเรอร์ IP แอดเดรสอาจเป็น 0.0.0.0 หากเว็บเซิร์ฟเวอร์ไม่ถูกโยงกับ IP แอดเดรสที่เจาะจง

2. ในการปิดอินสแตนซ์ FRCA ที่สัมพันธ์กับ IP แอดเดรส 9.1.1.1 และพอร์ต 80 ให้พิมพ์:

```
frctrll close 9.1.1.1 80
```

3. ในการโหลดเนื้อหาของไฟล์ `/a/b/c/d` and `/a/b/c/e` ที่มี URLs `/d` และ `/e` ให้พิมพ์:

```
frctrll loadfile /a/b/c /a/b/c/d e
```

4. ในการแสดงสถิติ FRCA ให้พิมพ์:

```
frctrll stats
```

ซึ่งจะทำให้แสดงสถิติ FRCA โดยจะมีลักษณะ คล้ายกับต่อไปนี้:

Total Requests	Deferred Requests	Cache Hits	Cache Misses	Resource Errors
1024065396	227	1024065168	1	0

5. ตัวอย่างนี้แสดงวิธีใช้คำสั่งย่อย `start` สำหรับโฮสต์เสมือน "ici":

```
frctrll start 9.1.1.1 80 ici
```

หมายเหตุ: พารามิเตอร์โฮสต์เสมือนเป็นทางเลือก

- ในการปิดใช้งานเคอร์เนลการรับค่าเอ็นจินสำหรับพอร์ต 80 บน IP แอดเดรส 9.1.1.1 บน โฮสต์เสมือน "ici" ให้พิมพ์:  

```
frctrll stop 9.1.1.1 80 ici
```
- ตัวอย่างต่อไปนี้จะตั้งค่าการหมดเวลาใช้งานการตรวจสอบความถูกต้องใหม่สำหรับอินสแตนซ์ FRCA ที่พอร์ต 80 ของ IP แอดเดรส 9.1.1.1 เป็น 100 วินาที:  

```
frctrll revaltimeout 9.1.1.1 80 100
```
- ในการอนุญาตให้ CPU ใช้เวลา 98 เปอร์เซ็นต์ในอินเตอร์รัปต์คอนเท็กซ์ให้พิมพ์:  

```
frctrll pconintr 98
```
- ในการตั้งค่าของอ็อปชัน `frca_hashsz` เป็น 24499 ให้พิมพ์:  

```
frctrll set frca_hashsz=24499
```
- ในการตั้งค่าของ `frca_hashsz` เป็นค่าดีฟอลต์ให้พิมพ์:  

```
frctrll default frca_hashsz
```

## ไฟล์

/usr/bin/frctrll

---

## คำสั่ง from

### วัตถุประสงค์

ในการพิจารณาว่าเมลมาจากใคร

### ไวยากรณ์

```
from [ -d Directory ] [ -s Sender ] [ user ]
```

### คำอธิบาย

คำสั่ง `from` แสดง ส่วนหัวข้อความในเมลบ็อกซ์ของคุณเพื่อแสดงให้คุณทราบว่าเมลมาจากไหน หากคุณ ระบุ `user` เมลบ็อกซ์ `user` จะถูกตรวจสอบแทนของคุณ (โดยคุณต้องมีสิทธิ์การอ่าน ในเมลบ็อกซ์ของผู้ใช้)

### แฟล็ก

ไอเท็ม	คำอธิบาย
-d Directory	ระบุไดเรกทอรีของเมลบ็อกซ์ระบุ
-s Sender	พิมพ์ส่วนหัวข้อความสำหรับเมลที่ส่งโดย Sender เท่านั้น

## พารามิเตอร์

ไอเท็ม  
ผู้ใช้

คำอธิบาย  
ระบุเมลบ็อกซ์ user ที่ถูกตรวจสอบแทนของคุณ (โดยคุณต้องมีสิทธิ์การอ่าน ในเมลบ็อกซ์ของระบุ)

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

1. ในการแสดงส่วนหัวข้อความในเมลบ็อกซ์ของคุณ ให้ป้อน:

```
from
```

ชื่อของผู้ส่งและวันที่ของ ข้อความจะถูกแสดง

2. ในการแสดงส่วนหัวข้อความสำหรับเมลที่ส่ง โดยผู้ใช้ที่ระบุ ให้ป้อน:

```
from -s dale
```

ในตัวอย่างนี้ เฉพาะส่วนหัว ข้อความของข้อความที่ส่งจากผู้ใช้ dale เท่านั้น ที่แสดง

3. ในการแสดงส่วนหัวข้อความในเมลบ็อกซ์ของผู้ใช้ที่เจาะจง ให้ป้อน:

```
from dawn
```

ในตัวอย่างนี้ ส่วนหัวข้อความจากเมลบ็อกซ์ของผู้ใช้ dawn จะแสดง (โดยที่ คุณต้องมีสิทธิ์การอ่านในเมลบ็อกซ์ของ dawn)

4. ในการดูข้อความทั้งหมดที่ bob ได้รับจาก jane ให้ป้อน:

```
from -d /var/spool/mail -s jane bob
```

คำสั่งนี้อุญาตให้คุณเห็นข้อความทั้งหมด ที่ bob ได้รับจาก jane โดยคุณมีสิทธิ์ (เช่น root)

## ไฟล์

ไอเท็ม

/var/spool/mail/\*  
/usr/bin/from

คำอธิบาย

เมลบ็อกซ์ระบบสำหรับผู้ใช้ทั้งหมด  
ไฟล์เมลบ็อกซ์ของผู้ใช้

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mail

แอปพลิเคชันเมล

---

## คำสั่ง fsck

### วัตถุประสงค์

ตรวจสอบความสอดคล้องกันของระบบไฟล์และซ่อมแซมระบบไฟล์ แบบโต้ตอบ

# ไวยากรณ์

```
fsck [-n] [-p] [-y] [-dBlockNumber] [-f] [-i-NodeNumber] [-o Options] [-tFile] [-V VfsName] [FileSystem1 -  
FileSystem2 ...]
```

## คำอธิบาย

**ข้อควรสนใจ:** รันคำสั่ง `fsck` บนระบบไฟล์เสมอ หลังจากระบบทำงานไม่ปกติ การดำเนินการแก้ไขอาจส่งผลให้เกิดข้อมูลสูญหาย การดำเนินการดีฟอลต์สำหรับการแก้ไขความสอดคล้องกันแต่ละครั้งคือการขอให้ผู้ควบคุมเครื่องป้อน `yes` หรือ `no` หากคุณไม่มีสิทธิ์ในการเขียนในระบบไฟล์ที่ได้รับผลกระทบ คำสั่ง `fsck` จะใช้ค่าดีฟอลต์เป็นการตอบกลับ `no` แม้ว่าจะไม่ใช่การตอบกลับจริงของคุณ

### Notes:

1. คำสั่ง `fsck` ไม่ได้ทำการแก้ไขในระบบไฟล์ที่ติดตั้ง
2. คำสั่ง `fsck` สามารถรันบนระบบไฟล์ที่เมาท์ ด้วยเหตุผลอื่นที่ไม่ใช่การซ่อมแซม อย่างไรก็ตาม ข้อความแสดงความผิดพลาดที่ไม่ถูกต้องอาจถูกส่งกลับเมื่อระบบไฟล์ถูกเมาท์

คำสั่ง `fsck` จะตรวจสอบและซ่อมแซมแบบโต้ตอบ ระบบไฟล์ที่ไม่สอดคล้องกัน คุณควรรันคำสั่งนี้ก่อนการเมาท์ ระบบไฟล์ใดๆ คุณต้องสามารถอ่านไฟล์อุปกรณ์ที่มีระบบไฟล์อยู่ (ตัวอย่างเช่น อุปกรณ์ `/dev/hd0`) โดยปกติ ระบบไฟล์จะสอดคล้องกัน และคำสั่ง `fsck` เพียงรายงานเกี่ยวกับจำนวนไฟล์ บล็อกที่ใช้ และบล็อกที่ว่างในระบบไฟล์ หากระบบไฟล์ไม่สอดคล้อง คำสั่ง `fsck` จะแสดงข้อมูลเกี่ยวกับความไม่สอดคล้องกันที่พบและพร้อมที่จะให้คุณเพื่อขอสิทธิ์ในการซ่อมแซม

คำสั่ง `fsck` ะมัดระวังเมื่อพยายามจะทำการซ่อมแซม และพยายามหลีกเลี่ยงการดำเนินการที่อาจส่งผลให้เกิดการสูญเสียข้อมูลที่ถูกต้อง อย่างไรก็ตาม ก็มีบางกรณี ที่คำสั่ง `fsck` จะแนะนำให้ทำลายไฟล์ที่เสียหายนั้น หากคุณไม่อนุญาต คำสั่ง `fsck` จะทำการซ่อมแซมตามที่จำเป็น ซึ่งอาจส่งผลต่อระบบไฟล์ที่ไม่สอดคล้องกัน การเมาท์ระบบไฟล์ที่ไม่สอดคล้องกันอาจส่งผลให้ระบบขัดข้อง

หากระบบไฟล์ JFS2 มีสแน็ปช็อต คำสั่ง `fsck` จะพยายาม สงวนไว้ หากการดำเนินการนี้ล้มเหลว สแน็ปช็อตไม่สามารถรับประกันได้ว่า มีข้อมูลทั้งหมดของอิมเมจก่อนหน้าจากระบบไฟล์สแน็ปช็อต คำสั่ง `fsck` จะลบสแน็ปช็อตและสแน็ปช็อตโลจิคัลวอลุ่ม สแน็ปช็อตภายในถูกลบหากคำสั่ง `fsck` แก้ไขระบบไฟล์

หากคุณไม่ระบุระบบไฟล์ด้วยพารามิเตอร์ `FileSystem` คำสั่ง `fsck` จะตรวจสอบระบบไฟล์ทั้งหมด ที่แสดงในไฟล์ `/etc/filesystems` ที่ซึ่งแอตทริบิวต์ `check` ถูกตั้งค่าเป็น `True` คุณสามารถ เปิดใช้งานการตรวจสอบชนิดนี้ได้โดยการเพิ่มบรรทัดใน stanza ดังนี้:

```
check=true
```

คุณยังสามารถดำเนินการตรวจสอบบนหลายระบบไฟล์ได้โดยการจัดกลุ่ม ระบบไฟล์ในไฟล์ `/etc/filesystems` ในการทำนั้น ให้เปลี่ยนแอตทริบิวต์การตรวจสอบในไฟล์ `/etc/filesystems` ดังนี้:

```
check=Number
```

พารามิเตอร์ `Number` แจ้งให้คำสั่ง `fsck` ทราบว่ากลุ่มใดมีระบบไฟล์เฉพาะ ระบบไฟล์ที่ใช้อุปกรณ์บันทึกการทำงานทั่วไป ควรจัดให้อยู่ในกลุ่มเดียวกัน ระบบไฟล์ จะถูกตรวจสอบทีละระบบ ตามลำดับกลุ่ม จากนั้นตามลำดับที่แสดงในไฟล์ `/etc/filesystems` ระบบไฟล์ `check=true` ทั้งหมดอยู่ในกลุ่ม 1 คำสั่ง `fsck` พยายามตรวจสอบระบบไฟล์แรกก่อน ระบบไฟล์อื่นโดยไม่คำนึงถึงลำดับที่ระบุไว้บนบรรทัดรับคำสั่ง หรือในไฟล์ `/etc/filesystems`

คำสั่ง fsck จะตรวจสอบความไม่สอดคล้องกันต่อไปนี้:

- บล็อกหรือแฟรกเมนต์ที่จัดสรรให้แก่ไฟล์หลายไฟล์
- i-nodes ที่มีบล็อกหรือแฟรกเมนต์จำนวนหนึ่งที่ซ้อนทับกัน
- i-nodes ที่มีบล็อกหรือแฟรกเมนต์จำนวนหนึ่งที่อยู่นอกขอบเขต
- ความแตกต่างกันระหว่างจำนวนการอ้างอิงไต่เรียกทอรีไปยังไฟล์และจำนวนลิงก์ของไฟล์
- บล็อกหรือแฟรกเมนต์ที่จัดสรรไม่ถูกต้อง
- i-nodes ที่มีบล็อกหรือแฟรกเมนต์จำนวนหนึ่งที่ถูกละทิ้งในดิสก์แม็พ
- i-nodes ที่มีบล็อกหรือแฟรกเมนต์จำนวนหนึ่งที่เสียหาย
- แฟรกเมนต์ที่ไม่ใช้ดีสก์แอดเดรสสุดท้ายใน i-node การตรวจสอบนี้ไม่ใช่กับระบบไฟล์บีบอัด
- ไฟล์ที่มีขนาดใหญ่กว่า 32KB ที่มีแฟรกเมนต์ การตรวจสอบนี้ไม่ใช่กับระบบไฟล์บีบอัด
- การตรวจสอบขนาด:
  - จำนวนบล็อกไม่ถูกต้อง
  - ขนาดไต่เรียกทอรีไม่เป็นจำนวนผลคูณของ 512 ไบต์

การตรวจสอบนี้ไม่ใช่กับระบบไฟล์บีบอัด

- การตรวจสอบไต่เรียกทอรี:
  - รายการไต่เรียกทอรีมีหมายเลข i-node ที่ทำเครื่องหมายว่างในแม็พ i-node
  - หมายเลข i-node ไม่อยู่ในช่วงที่กำหนด
  - ไม่มีลิงก์จุด(.) หรือไม่ได้ชี้ไปยังตัวเอง
  - ไม่มีลิงก์จุดจุด(..) หรือไม่ได้ชี้ไปยังไต่เรียกทอรีพารেন্ট
  - ไฟล์ที่ไม่ถูกอ้างอิง หรือไต่เรียกทอรีที่ไม่สามารถเข้าถึงได้
- ดิสก์แม็พไม่สอดคล้องกัน
- แม็พ i-node ไม่สอดคล้องกัน

ไฟล์และไต่เรียกทอรี Orphaned (คือไม่สามารถเข้าถึงได้) หากคุณ อนุญาต จะถูกเชื่อมต่ออีกครั้งโดยการวางไฟล์และไต่เรียกทอรีในไต่เรียกทอรีย่อย lost+found ในไต่เรียกทอรี root ของระบบไฟล์ ชื่อที่กำหนดคือหมายเลข i-node หากคุณไม่อนุญาต คำสั่ง fsck จะเชื่อมต่อกับ ไฟล์ orphaned ใหม่ โดยร้องขอสิทธิเพื่อทำลายไฟล์

นอกจากข้อความแล้ว คำสั่ง fsck ยังบันทึก ผลลัพธ์ที่ได้ของการตรวจสอบและซ่อมแซมจากค่าสถานะการออก ค่าสถานะการออกนี้ สามารถเป็นผลรวมใดๆ ของสภาวะต่อไปนี้:

ไอเท็ม คำอธิบาย

- |   |  |
|---|--|
| 0 | ระบบไฟล์ที่ตรวจสอบทั้งหมดเรียบร้อยดี   |
| 2 | คำสั่ง fsck ถูกขัดจังหวะก่อนที่จะดำเนินการตรวจสอบและซ่อมแซมให้เสร็จเรียบร้อย |
| 4 | คำสั่ง fsck เปลี่ยนแปลงระบบไฟล์ ผู้ใช้ต้องรีสตาร์ทระบบในทันที                |
| 8 | ระบบไฟล์มีความเสียหายที่ไม่ได้รับการซ่อมแซม                                  |

คำสั่ง fsck ต้องการการเข้าถึงพิเศษ ไปยังอุปกรณ์วอลุ่มแบบโลจิคัลของระบบไฟล์ที่สำคัญกว่า หาก fsck ล้มเหลวเนื่องจากอุปกรณ์ที่สำคัญกว่าไม่พร้อมใช้งานแล้ว fsck จะถูกเรียกทำงานใหม่อีกครั้งหลังจากอุปกรณ์พร้อมที่จะเปิด

เมื่อระบบบูตจากดิสก์ กระบวนการบูตจะรัน คำสั่ง `fsck` อย่างชัดเจน ที่ระบุด้วยแฟล็ก `-f` และ `-p` บนระบบไฟล์ `/`, `/usr`, `/var` และ `/tmp` หากคำสั่ง `fsck` ดำเนินการไม่สำเร็จ บนระบบไฟล์ใดๆ เหล่านี้ ระบบจะไม่บูต การบูตจากสื่อบันทึก แบบถอดออกได้ และการดำเนินการดูแลรักษาจะเป็นสิ่งที่ต้องทำก่อนที่ระบบนั้น จะบูต

หากคำสั่ง `fsck` รันบน `/`, `/usr`, `/var` และ `/tmp` สำเร็จ การกำหนดค่าเริ่มต้นระบบปกติจะดำเนินการต่อ ระหว่างการกำหนดค่า เริ่มต้นระบบปกติ คำสั่ง `fsck` ที่ระบุด้วยแฟล็ก `-f` และ `-p` จะรันจากไฟล์ `/etc/rc` คำสั่งนี้ทำการตรวจสอบลำดับระบบไฟล์ทั้งหมดที่ แอ็ททริบิวต์ `check` ถูกตั้งค่าเป็น `True` (`check=true`) หากคำสั่ง `fsck` ถูกเรียกใช้งานจาก ไฟล์ `/etc/rc` ไม่สามารถรับประกันความสอดคล้องกัน ของระบบไฟล์ใดๆ ได้ การกำหนดค่าเริ่มต้นระบบจะดำเนินการต่อ อย่างไรก็ตาม การเมาท์ระบบไฟล์ที่ไม่สอดคล้องกันอาจล้มเหลว การเมาท์ล้มเหลวอาจทำให้การกำหนดค่าเริ่มต้นระบบ ไม่สมบูรณ์

หมายเหตุ: โดยค่าดีฟอลต์ ระบบไฟล์ `/`, `/usr`, `/var` และ `/tmp` มีแอ็ททริบิวต์ `check` ถูกตั้งค่า เป็น `False` (`check=false`) ใน `/etc/filesystem` stanzas แอ็ททริบิวต์ถูกตั้งค่าเป็น `False` สำหรับเหตุผล ต่อไปนี้:

1. กระบวนการบูตรันคำสั่ง `fsck` บนระบบไฟล์ `/`, `/usr`, `/var` และ `/tmp` อย่างเด่นชัด
2. ระบบไฟล์ `/`, `/usr`, `/var` และ `/tmp` ถูกเมาท์เมื่อไฟล์ `/etc/rc` ถูกเรียกใช้งาน คำสั่ง `fsck` จะไม่แก้ไขระบบไฟล์ที่เมาท์ นอก จากนั้น คำสั่ง `fsck` ที่รันระบบไฟล์ที่เมาท์จะให้ผลลัพธ์ที่ไม่น่าเชื่อถือ

คุณสามารถใช้พาด่วน `smit fsck` ของ System Management Interface Tool (SMIT) เพื่อรันคำสั่งนี้

## แฟล็ก

### ไอเท็ม

`-dBlockNumber`

### คำอธิบาย

ค้นหาการอ้างอิงไปที่บล็อกดิสก์ที่ระบุ เมื่อใดก็ตามที่คำสั่ง `fsck` พบไฟล์ที่มีบล็อกที่ระบุ ซึ่งแสดงหมายเลข `i-node` และชื่อพาททั้งหมดที่อ้างอิงถึง สำหรับระบบไฟล์ JFS2 หมายเลข `i-node` ที่อ้างอิงบล็อกที่ระบุจะ ถูกแสดงแต่ไม่แสดงชื่อพาท

`-f`

ดำเนินการตรวจสอบอย่างรวดเร็ว ภายใต้สถานการณ์ปกติ ระบบไฟล์เดียว ที่ได้รับผล โดยการหยุดระบบโดยไม่ต้องมีการปิดระบบอย่างถูกต้อง คือระบบที่ถูกเมาท์เมื่อระบบหยุด แฟล็ก `-f` พร้อมคำสั่ง `fsck` ไม่ต้องตรวจสอบ ระบบไฟล์ที่ถูกยกเลิกการเมาท์เสร็จสมบูรณ์ คำสั่ง `fsck` ระบุได้โดยอาศัยการตรวจสอบแฟล็ก `s_fmmod` ใน `superblock` ระบบไฟล์

`-ii-NodeNumber`

แฟล็กนี้ถูกเซตเมื่อใดก็ตาม ที่ระบบไฟล์ถูกเมาท์และเคลียร์ เมื่อมีการยกเลิกการเมาท์เสร็จสมบูรณ์ ถ้า ระบบไฟล์ถูกยกเลิกการเมาท์เสร็จสมบูรณ์ แสดงว่าไม่มีปัญหาใด เนื่องจากระบบไฟล์ส่วนใหญ่จะถูกยกเลิกการเมาท์เสร็จสมบูรณ์ การไม่ตรวจสอบระบบไฟล์เหล่านี้ สามารถลดเวลาการตรวจสอบได้

`-n`

`-o Options`

ค้นหาการอ้างอิงไปที่ `i-node` ที่ระบุ เมื่อใดก็ตามที่คำสั่ง `fsck` พบการอ้างอิงใดเรียกทอริไปที่ `i-node` ที่ระบุ, จะมีการแสดงชื่อพาทแบบเต็มของการอ้างอิง

ถือว่ามีการตอบกลับ `no` ไปที่คำถามที่ถาม ทั้งหมดโดยคำสั่ง `fsck`; ไม่เปิดระบบไฟล์ ที่ระบุเพื่อการเขียนส่งอ็อพชั่นโดยแยกด้วยคอมมาไปที่คำสั่ง `fsck` ขณะนี้อ็อพชั่นต่อไปนี้ได้รับการสนับสนุนสำหรับ JFS (อ็อพชั่นต่างๆ ที่ไม่ใช่แล้ว สำหรับระบบไฟล์ที่ใหม่กว่าและสามารถข้ามได้):

### mountable

ทำให้คำสั่ง `fsck` จบการทำงานสำเร็จ ส่งคืนค่า 0, ถ้าระบบไฟล์ดังกล่าวสามารถเมาท์ได้ (clean) ถ้าระบบไฟล์ไม่สามารถเมาท์ได้ คำสั่ง `fsck` จบการทำงานโดยส่งคืนค่า 8

### mytype

ทำให้คำสั่ง `fsck` จบการทำงานสำเร็จ (0) ถ้าระบบไฟล์ดังกล่าวเป็นชนิดเดียวกับที่ระบุไว้ในไฟล์ `/etc/filesystems` หรือโดยแฟล็ก `-V` บน บรรทัดคำสั่ง มิฉะนั้น ค่า 8 จะถูกส่งคืน ตัวอย่าง, `fsck -omytype -V jfs /` จบการทำงานด้วยค่า 0 ถ้า / (ระบบไฟล์ `root`) เป็นระบบไฟล์ `journaled`

`-p`

ไม่แสดงข้อความเกี่ยวกับปัญหาบ่อย แต่ทำการแก้ไขโดยอัตโนมัติ แฟล็กนี้ไม่ได้รับไลเซนส์ทั้งหมดเหมือนกับแฟล็ก `-y` และมีประโยชน์สำหรับการตรวจสอบอัตโนมัติ เมื่อระบบถูกเริ่มต้นตามปกติ คุณควรใช้แฟล็กนี้เป็นส่วนหนึ่งของขั้นตอนการเริ่มต้นระบบ เมื่อใดก็ตามที่ระบบถูกรันโดยอัตโนมัติ หากซูเปอร์บล็อกหลักเสียหาย ซูเปอร์บล็อกรองจะถูกตรวจสอบ และคัดลอกไปยังซูเปอร์บล็อกหลัก

ไอเท็ม	คำอธิบาย
-t <i>File</i>	ระบุพารามิเตอร์ <i>File</i> เป็นไฟล์ว่าง บนระบบไฟล์อื่นที่ไม่ใช่ระบบที่ถูกตรวจสอบ, ถ้าคำสั่ง <code>fsck</code> ไม่สามารถรับหน่วยความจำได้อย่างเพียงพอเพื่อคงความเสถียรไว้ ถ้าคุณไม่ ระบุแฟล็ก <code>-t</code> และคำสั่ง <code>fsck</code> ต้องการไฟล์เริ่มต้น คำสั่งจะพร้อมต่อชื่อของไฟล์ เริ่มต้น อย่างไรก็ตาม ถ้าคุณได้ระบุแฟล็ก <code>-p</code> คำสั่ง <code>fsck</code> จะไม่เสร็จสมบูรณ์ ถ้าไฟล์เริ่มต้นไม่ใช่ไฟล์พิเศษ ไฟล์จะถูกเอาออกเมื่อคำสั่ง <code>fsck</code> สิ้นสุด
-V <i>VfsName</i>	ใช้คำอธิบายของระบบไฟล์เสมือนที่ระบุโดยตัวแปร <i>VFSName</i> สำหรับระบบไฟล์แทนที่ใช้ไฟล์ <code>/etc/filesystems</code> ในการระบุรายละเอียด ถ้าไม่ระบุแฟล็ก <code>-V VfsName</code> ในบรรทัดคำสั่งไฟล์ <code>/etc/filesystems</code> จะถูกตรวจสอบและ <code>vfs=Attribute</code> ของ stanza ที่ตรงกันจะถือว่าเป็นชนิดระบบไฟล์ที่ต้องการ
-y	ถือว่าตอบกลับ <code>yes</code> กับคำถามทั้งหมดที่ถามโดยคำสั่ง <code>fsck</code> แฟล็กนี้ให้คำสั่ง <code>fsck</code> ดำเนินการที่คำสั่งเห็นว่าเป็นใช้แฟล็กนี้เฉพาะกับระบบไฟล์ ที่เสียหายอย่างร้ายแรง

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ตัวอย่าง

1. ในการตรวจสอบระบบไฟล์ดีฟอลต์ทั้งหมดให้ป้อน:

```
fsck
```

คำสั่งนี้จะตรวจสอบระบบไฟล์ทั้งหมดที่ทำเครื่องหมาย `check=true` ในไฟล์ `/etc/filesystems` คำสั่ง `fsck` รูปแบบนี้จะขอสิทธิจากคุณ ก่อนทำการเปลี่ยนแปลงใดๆ ในระบบไฟล์

2. ในการแก้ไขปัญหาเล็กๆ กับระบบไฟล์ดีฟอลต์โดยอัตโนมัติให้ป้อน:

```
fsck -p
```

3. ในการตรวจสอบระบบไฟล์ที่เจาะจงให้ป้อน:

```
fsck /dev/hd1
```

คำสั่งนี้ ตรวจสอบระบบไฟล์ที่ไม่ถูกเมาท์ที่อยู่บนอุปกรณ์ `/dev/hd1`

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/sbin/fsck</code>	มีคำสั่ง <code>fsck</code>
<code>/etc/filesystems</code>	แสดงรายการของระบบไฟล์ที่รู้จัก และกำหนดคุณสมบัติไว้
<code>/etc/vfs</code>	มีคำอธิบายของชนิดของระบบไฟล์เสมือน
<code>/etc/rc</code>	มีคำสั่ง (รวมทั้งคำสั่ง <code>fsck</code> ) ที่ถูกรันเมื่อระบบเริ่มทำงาน

## สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dfscck`” ในหน้า 127

“คำสั่ง `fsdb`” ในหน้า 615

## ข้อมูลที่เกี่ยวข้อง:

ไฟล์ `filsys.h`

ระบบไฟล์

---

## คำสั่ง `fsck_cachefs`

### Purpose

ตรวจสอบความสมบูรณ์ของข้อมูลที่แคชด้วย CacheFS

### Syntax

```
fsck_cachefs [ -m ] [ -o noclean ] cache_directory
```

### รายละเอียด

เวอร์ชัน CacheFS ของคำสั่ง `fsck` ตรวจสอบ ความสมบูรณ์ของแคชไดเรกทอรีโดยค่าดีฟอลต์จะแก้ไขปัญหา CacheFS ใดๆ ที่พบ ไม่มีโหมดการโต้ตอบ การร้องขอที่อาจเป็นไปได้ของ `fsck_cachefs` สำหรับระบบไฟล์ CacheFS คือตอนบูตจากรายการใน `/etc/rc.nfs`

### แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-m</code>	ตรวจสอบแต่ไม่ซ่อมแซม
<code>-o noclean</code>	บังคับให้มีการตรวจสอบบนแคช แม้ว่าจะไม่เห็นผลให้ต้องสงสัยว่า มีปัญหา

### ตัวอย่าง

ในการบังคับให้มีการตรวจสอบแคชไดเรกทอรีให้บ่อย:

```
fsck_cachefs -o noclean /cache3
```

---

## คำสั่ง `fsdb`

### วัตถุประสงค์

ดีบักระบบไฟล์

### ไวยากรณ์

```
fsdb FileSystem [ - ]
```

### คำอธิบาย

คำสั่ง `fsdb` เปิดให้คุณ ตรวจสอบ ปรับเปลี่ยน และดีบักระบบไฟล์ ที่ระบุโดยพารามิเตอร์ `FileSystem` คำสั่งจัดให้มีการเข้าถึงออบเจกต์ระบบไฟล์ เช่น บล็อก, `i-nodes` หรือไดเรกทอรี คุณสามารถใช้คำสั่ง `fsdb` เพื่อตรวจสอบและแพ็ทช์ระบบไฟล์ที่เสียหาย คีย์คอมโพเนนต์ของระบบไฟล์สามารถถูกอ้างอิงในแบบสัญลักษณ์ คุณลักษณะนี้ช่วยให้โปรแกรมเมอร์แก้ไขรายการของบล็อกควบคุมได้ง่าย และเรียงลำดับแผนผังระบบไฟล์ จากมากไปหาน้อย

ในการตรวจสอบระบบไฟล์ ให้ระบุด้วยชื่ออุปกรณ์แบบ บล็อก ชื่ออุปกรณ์จริง หรือชื่อระบบไฟล์ที่เมาท์ในกรณีสุดท้าย คำสั่ง `fsdb` จะพิจารณาชื่อระบบไฟล์ที่สัมพันธ์กัน โดยการอ่านไฟล์ `/etc/filesystems` ระบบไฟล์ที่ถูกเมาท์ไม่สามารถแก้ไขได้

คำสั่ง `fsdb` มีอินเตอร์เฟซแตกต่างกันสำหรับ ระบบไฟล์ JFS และระบบไฟล์ JFS2 ต่อไปนี้อธิบายวิธีใช้ `fsdb` กับระบบไฟล์ JFS โปรดดูที่ คำสั่งย่อย JFS2 เพื่อดูข้อมูลเกี่ยวกับคำสั่งย่อย JFS2

หากระบบไฟล์ที่ระบุเป็นสแน็ปช็อต JFS2 คำสั่ง `fsdb` จะเปิดใช้งานการตรวจสอบและการแก้ไขของสแน็ปช็อตซูเปอร์บล็อก สแน็ปช็อตเม็พ สำเนา `xtree` บล็อกเม็พ และส่วนหัวเซกเมนต์ โปรดดูที่ คำสั่งย่อยสแน็ปช็อต JFS2 เพื่อดูข้อมูลเกี่ยวกับคำสั่งย่อยสแน็ปช็อต JFS2

คำสั่งย่อย สำหรับคำสั่ง `fsdb` อนุญาตให้คุณเข้าถึง ดู หรือเปลี่ยนข้อมูล ในระบบไฟล์ หมายเลขใดๆ ที่คุณป้อนในคำสั่งย่อยจะถูกพิจารณาเป็นฐานสิบ โดยค่าดีฟอลต์ ยกเว้นว่าคุณใส่ค่านำหน้าด้วย 0 เพื่อระบุตัวเลขฐานแปด หรือ 0x เพื่อระบุตัวเลขฐานสิบหก แอดเดรสทั้งหมดจะถูกพิมพ์ในเป็นเลขฐานสิบหก

เนื่องจากคำสั่ง `fsdb` อ่าน และเขียนครั้งละหนึ่งบล็อก โดยทำงานกับข้อมูลดิบรวมถึงกับ I/O บล็อก

## แฟล็ก

ไอเท็ม คำอธิบาย

- เปิดใช้งานไจรูทีนตรวจสอบข้อผิดพลาดที่ใช้เพื่อตรวจสอบแอดเดรส i-nodes และ บล็อก คำสั่งย่อย O จะเปิดและปิดรูทีนเหล่านี้ เมื่อรูทีนเหล่านี้กำลังรัน คำสั่ง `fsdb` จะอ่านข้อมูลระบบไฟล์วิกฤตจากซูเปอร์บล็อก ข้อมูลที่ได้รับจะช่วยให้คำสั่ง `fsdb` เข้าถึงอ็อบเจกต์ระบบไฟล์ที่แตกต่างกันได้สำเร็จและดำเนินการตรวจสอบข้อผิดพลาดต่างๆ ได้

## คำสั่งย่อย

คำสั่งย่อย `fsdb` คือการร้องขอ เพื่อระบุตำแหน่งและแสดงหรือแก้ไขข้อมูลในระบบไฟล์ หมวดหมู่หลัก ของคำสั่งย่อยได้แก่:

ไอเท็ม	คำอธิบาย
หมวดหมู่	ฟังก์ชัน
Location	เข้าถึงข้อมูลในระบบไฟล์
Display	ดูข้อมูลในระบบไฟล์
Modification	เปลี่ยนแปลงข้อมูลในระบบไฟล์

นอกจากนั้น ยังมีคำสั่งย่อย miscellaneous บางคำสั่ง

## คำสั่งย่อยตำแหน่ง

มีคำสั่งย่อยตำแหน่งสองชนิด:

`Number[ I | M | i | b ]`

หรือ

`dDirectorySlot`

ชนิดแรกประกอบด้วยตัวเลข และอาจตามด้วย ข้อกำหนดคุณลักษณะแอดเดรส ข้อกำหนดคุณลักษณะแอดเดรสจะกำหนดวิธีการแปลความหมายตัวเลขที่นำหน้า โดยมีข้อกำหนดคุณลักษณะแอดเดรสสี่แบบที่สอดคล้องกับการแปลความหมายตัวแปร `Number` แตกต่างกันไป:

ไอเท็ม	คำอธิบาย
I	หมายเลขบล็อกแม่ I-node
M	หมายเลขบล็อกแม่พดิสก์
i	หมายเลข I-node
b	หมายเลขแฟร็กเมนต์

โดยขึ้นอยู่กับข้อกำหนดคุณลักษณะแอดเดรส (หรือการไม่มีอยู่) คำสั่งย่อยตำแหน่งชนิดนี้จะเข้าถึงข้อมูลดังนี้:

ไอเท็ม	คำอธิบาย
<i>Number</i>	เข้าถึงข้อมูลที่ไบต์ออฟเซตสัมบูรณ์ที่ระบุโดยตัวแปร <i>Number</i>
<i>MapBlockNumberI</i>	เข้าถึงบล็อกแม่ I-node ที่กำหนดโดยตัวแปร <i>MapBlockNumber</i>
<i>MapBlockNumberM</i>	เข้าถึงบล็อกแม่พดิสก์ที่กำหนดโดยตัวแปร <i>MapBlockNumber</i>
<i>InodeNumberi</i>	เข้าถึง i-node ที่ระบุโดยตัวแปร <i>InodeNumber</i>
<i>FragmentNumberb</i>	เข้าถึงบล็อกระบบไฟล์ที่ระบุโดยตัวแปร <i>FragmentNumber</i> หมายเลขแฟร็กเมนต์ประกอบด้วยบล็อกแอดเดรส และความยาวที่เข้ารหัส แฟร็กเมนต์แอดเดรสโดยสมบูรณ์จะยาว 32 บิต โดย 28 บิตในลำดับต่ำคือแฟร็กเมนต์แอดเดรสเริ่มต้น ความยาวแฟร็กเมนต์ถูกเข้ารหัสใน 4 บิตที่เหลือ โดยเข้ารหัสเป็นจำนวนแฟร็กเมนต์ ที่น้อยกว่าบล็อกขนาดเต็ม ตัวอย่างเช่น บนระบบไฟล์ที่ประกอบด้วยแฟร็กเมนต์ 1024 ไบต์ แอดเดรส 0x2000010f อ้างถึง บล็อกที่เริ่มต้นด้วยหมายเลขบล็อกขนาด 1KB 0x10f และมีความยาว 2KB ในทางกลับกัน บนระบบไฟล์ที่มีแฟร็กเมนต์ 512 ไบต์ แอดเดรส 0x2000010f อ้างอิงบล็อกที่ เริ่มต้นที่บล็อก 512 ไบต์ 0x10f และความยาว 3072 (512 * 6) ไบต์

ชนิดที่สองของคำสั่งย่อยตำแหน่งถูกใช้เพื่อเข้าถึง รายการไดเร็กทอรี คำสั่งย่อยประกอบด้วยอักขระ **d** ตามด้วยหมายเลข ไดเร็กทอรี-สล็อต หมายเลขไดเร็กทอรี-สล็อตเริ่มต้นที่ 0 สำหรับแต่ละบล็อกของ i-node ที่เชื่อมโยงกัน

คำสั่งย่อยตำแหน่งชนิดนี้จะเข้าถึงข้อมูล ดังนี้:

ไอเท็ม	คำอธิบาย
<i>dDirectorySlot</i>	เข้าถึงรายการไดเร็กทอรีที่ตำแหน่งนี้โดยตัวแปร <i>DirectorySlot</i> สำหรับ i-node ปัจจุบัน เฉพาะรายการไดเร็กทอรีที่จัดสรรเท่านั้น ที่สามารถจัดการโดยใช้คำสั่งย่อยตำแหน่งนี้

### คำสั่งย่อยการแสดงผล

ในการดูข้อมูลสัมพันธ์กับข้อกำหนดคุณลักษณะแอดเดรส ให้ใช้คำสั่งย่อยการแสดงผลที่ประกอบด้วยหนึ่งในโปรแกรมอำนวยความสะดวกการแสดงผล ร่วมกับ รูปแบบการแสดงผลแบบหนึ่ง ดังนี้:

```
p[Number]{ i|d|o|e|c|b|y|M|I|x|s|D }
```

หรือ

```
f[Number]{ i|d|o|e|c|b|y|M|I|x|s|D }
```

โปรแกรมอำนวยความสะดวกการแสดงผลได้แก่:

**ไอเท็ม คำอธิบาย**

- p** ระบุโปรแกรมอำนวยความสะดวกทั่วไป ใช้คำสั่งย่อยการแสดงผลทั่วไป เพื่อแสดงข้อมูลที่สัมพันธ์กับแอดเดรสปัจจุบัน หากคุณป้อนหมายเลขหลังสัญลักษณ์ **p** คำสั่ง **fsdb** จะแสดงรายการหมายเลขนั้น การตรวจสอบถูกดำเนินการเพื่อตรวจหาโอเวอร์โฟลว์ ของขอบเขตบล็อก หากคุณป้อน 0 หรือ \* (เครื่องหมายดอกจัน) คำสั่ง **fsdb** จะแสดงรายการทั้งหมดจนถึงสิ่งสุดท้ายที่ปรากฏในปัจจุบัน
- f** ระบุโปรแกรมอำนวยความสะดวกไฟล์ ใช้คำสั่งย่อยการแสดงผล เพื่อแสดง บล็อกข้อมูลที่สัมพันธ์กับ **i-node** ปัจจุบัน หากคุณป้อนหมายเลขหลังสัญลักษณ์ **f** คำสั่ง **fsdb** จะแสดงบล็อกของไฟล์ การกำหนดหมายเลขบล็อกจะเริ่มต้นที่ 0 รูปแบบ การแสดงตามหลังหมายเลขบล็อก หากคุณป้อน **f** โดยไม่มี หมายเลขบล็อก คำสั่ง **fsdb** จะใช้ค่าดีฟอลต์เพื่อแสดง บล็อก 0 ของ **i-node** ปัจจุบัน

รูปแบบการแสดงผลสำหรับโปรแกรมอำนวยความสะดวกได้แก่:

**ไอเท็ม คำอธิบาย**

- i** แสดงเป็น **i-nodes**
- d** แสดงเป็นไดเรกทอรี
- o** แสดงเป็นเวิร์ดฐานแปด
- e** แสดงเป็นเวิร์ดฐานสิบ
- c** แสดงเป็นอักขระ
- b** แสดงเป็นไบตฐานแปด
- y** แสดงเป็นไบตฐานสิบหก
- M** แสดงรายการแม่พิดส์
- I** แสดงเป็นรายการแม่พ **i-node**
- x** แสดงเป็นเวิร์ดฐานสิบหก
- S** แสดงเป็นบล็อกทางอ้อมบล็อกเดียว
- D** แสดงเป็นบล็อกทางอ้อมบล็อกคู่

โปรแกรมอำนวยความสะดวกการแสดงผลที่เลือกและรูปแบบการแสดงผล ยังมีผลอยู่ระหว่างประมวลผลคำสั่ง **fsdb** จนกว่าจะมีการเปลี่ยนแปลงอย่างชัดเจน คุณอาจได้รับข้อความแสดงความผิดพลาดที่ระบุการจัดตำแหน่ง ไม่เหมาะสมหากแอดเดรสที่คุณระบุไม่อยู่ภายในขอบเขตที่เหมาะสม

หากคุณใช้คำสั่งย่อยตำแหน่ง **Number**, **MapBlockNumber** หรือ **FragmentNumber** เพื่อเข้าถึง ข้อมูล **i-node** คุณสามารถตรวจสอบข้อมูล ตรวจสอบแต่ละไบต์ เวิร์ด หรือดับเบิลเวิร์ด เลือกโหมดการแสดงผลที่ต้องการโดยการป้อนคำสั่งย่อยใดคำสั่งย่อยหนึ่ง ต่อไปนี้:

**ไอเท็ม คำอธิบาย**

- B** เริ่มต้นการแสดงในไบต์โหมด
- D** เริ่มต้นการแสดงในดับเบิลเวิร์ดโหมด
- W** เริ่มต้นการแสดงในเวิร์ดโหมด

คุณสามารถเลื่อนขึ้นหน้าหรือย้อนหลังเพื่อดูข้อมูล ขอบเขตจะเลื่อนเพื่อกับหน้าจอแสดงผล และอยู่ที่แอดเดรส ที่รายการสุดท้ายแสดง เอาต์พุตสามารถสิ้นสุดได้ตลอดเวลาโดยการกด ปุ่ม **INTERRUPT** สัญลักษณ์ต่อไปนี้อัญญาตีให้เคลื่อนย้ายในข้อมูล:

**ไอเท็ม คำอธิบาย**

- + Number** ย้ายขึ้นหน้าไปยังจำนวนยูนิตที่ระบุที่มีผลอยู่ในขณะนี้
- Number** ย้ายย้อนกลับจำนวนยูนิตที่ระบุที่มีผลอยู่ในขณะนี้

สัญลักษณ์ต่อไปนี้อัญญาตีให้คุณเก็บแอดเดรสปัจจุบัน และกลับไปแอดเดรสนั้นได้โดยง่าย:

ไอเท็ม	คำอธิบาย
>	เก็บค่าแอดเดรสปัจจุบัน
<	กลับไปยังแอดเดรสที่เก็บก่อนหน้า

คุณสามารถใช้จุด แท็บ และช่องว่าง เป็นตัวคั่นคำสั่งย่อย แต่จำเป็นต้องเมื่อใช้คั่นเลขฐานสิบหกจากคำสั่งย่อย ที่อาจถูกแปลเป็นตัวเลขฐานสิบหก การกดปุ่ม Enter (การป้อน บรรทัดว่าง) จะเพิ่มค่าแอดเดรสปัจจุบันตามขนาดของชนิดข้อมูลที่แสดงล่าสุด นั่นคือ แอดเดรสถูกตั้งเป็นเป็นไบต์ เวิร์ด ดับเบิลเวิร์ด รายการไดเรกทอรี หรือ i-node ถัดไป ซึ่งอนุญาตให้คุณตรวจสอบขอบเขตของ ระบบไฟล์

คำสั่ง `fsdb` แสดงข้อมูล ในรูปแบบที่เหมาะสมกับชนิดข้อมูล ไบต์ เวิร์ด และดับเบิลเวิร์ดจะถูกแสดง เป็นแอดเดรสเลขฐานสิบหกตามด้วยการแสดงค่าเลขฐานสิบหก ของข้อมูลที่แอดเดรสและค่าฐานสิบที่เทียบเท่าที่อยู่ในวงเล็บ คำสั่ง `fsdb` เพิ่มคำต่อท้าย `.B` หรือ `.D` ที่ท้ายแอดเดรสเพื่อระบุการแสดงเป็นค่าไบต์ หรือดับเบิลเวิร์ด ซึ่งแสดงไดเรกทอรีเป็นไดเรกทอรีสี่อต่อฟเช็ดตามด้วยหมายเลข i-node ฐานสิบ และการแทนค่าอักษรของชื่อรายการ โดยแสดง i-nodes ด้วยฟิลด์ที่เลเบลอธิบายแต่ละอิลีเมนต์ ตัวแปรสภาวะแวดล้อม `ควบคุมรูปแบบของฟิลด์วันที่และเวลา`

### คำสั่งย่อยการแก้ไข

คุณสามารถแก้ไขข้อมูลที่สัมพันธ์กับข้อกำหนดคุณลักษณะ แอดเดรสโดยการใช้ข้อกำหนดคุณลักษณะฟิลด์ (สำหรับฟิลด์ใน i-node และ ฟิลด์ในไดเรกทอรี) รูปแบบทั่วไปสำหรับการกำหนดค่าใหม่คือ: *mnemonic operator new-value* โดยที่พารามิเตอร์ *mnemonic* แทนฟิลด์ใดฟิลด์หนึ่งที่อธิบายในรายการ ต่อไปนี้:

`mnemonics` ต่อไปนี้ถูกใช้สำหรับชื่อของ ฟิลด์ ของ i-node และอ้างถึง i-node การทำงานปัจจุบัน:

ไอเท็ม	คำอธิบาย
md	โหมดสิทธิ์
ln	จำนวนการลิงก์
uid	หมายเลขผู้ใช้
gid	หมายเลขกลุ่ม
sz	ขนาดไฟล์
aNumber	หมายเลขบล็อกข้อมูล (0 ถึง 8) โดยที่พารามิเตอร์ <i>Number</i> สามารถเป็นคำสั่งย่อยตำแหน่ง
at	เวลาการเข้าถึง
mt	เวลาแก้ไข
maj	หมายเลขอุปกรณ์หลัก
min	หมายเลขอุปกรณ์รอง

`mnemonics` ต่อไปนี้อ้างอิงแม่พ i-node และ ดิสก์:

ไอเท็ม	คำอธิบาย
mf	จำนวนแม่พรี
ms	ขนาดแม่พ
mp	บิตแม่พการจัดสรรถาวร
mw	บิตแม่พการจัดสรรที่ใช้งาน

`mnemonics` ต่อไปนี้ใช้สำหรับชื่อของ ฟิลด์ในไดเรกทอรี:

ไอเอ็ม	คำอธิบาย
r	ความยาวของเร็กคอร์ดรายการไดเร็กทอรี
n	ความยาวของชื่อไดเร็กทอรี
nm	ชื่อไดเร็กทอรี

ค่าที่ใช้ได้ของพารามิเตอร์ *Operator* ได้แก่:

**หมายเหตุ:** ระบบไฟล์ต้อง ถูกเลิกเมาทก่อนที่จะแก้ไข

ไอเอ็ม	คำอธิบาย
=	กำหนดพารามิเตอร์ <i>New-Value</i> เป็นพารามิเตอร์ <i>Mnemonic</i> ที่ระบุ
+=	เพิ่มค่าพารามิเตอร์ <i>Mnemonic</i> ด้วยพารามิเตอร์ <i>New-Value</i> ที่ระบุ พารามิเตอร์ <i>New-Value</i> ดีฟอลต์คือค่าหนึ่ง
-=	ลดค่า <i>Mnemonic</i> ด้วย <i>New-Value</i> ที่ระบุ <i>New-Value</i> ดีฟอลต์ คือค่าหนึ่ง
="	กำหนดสตริงอักขระที่ระบุโดยพารามิเตอร์ <i>New-Value</i> เป็นพารามิเตอร์ <i>Mnemonic</i> ที่ระบุ หากรูปแบบการแสดงผลปัจจุบันเป็นข้อกำหนดคุณสมบัติแอดเดรส <i>d</i> สำหรับไดเร็กทอรีและ mnemonic ไม่ถูกระบุ ชื่อไดเร็กทอรีจะถูกเปลี่ยน ชื่อไดเร็กทอรีใหม่ไม่สามารถยาวเกิน ชื่อไดเร็กทอรีก่อนหน้านี้

### คำสั่งย่อยเบ็ดเตล็ด

คำสั่งย่อยเบ็ดเตล็ด ได้แก่:

ไอเอ็ม	คำอธิบาย
q	ออก

ไอเอ็ม	คำอธิบาย
xn	ขยายไดเร็กทอรี <i>n</i> ไบต์โดย <i>n</i> บวกขนาดปัจจุบันของไดเร็กทอรีต้องไม่มากกว่า แพรกเมนต์เป็นจำนวนไบต์ของไดเร็กทอรีปัจจุบัน
!	ออกไปที่เซสล
O	สลับการตรวจสอบข้อผิดพลาด

### คำสั่งย่อย JFS2

คำสั่งย่อยเหล่านี้สามารถป้อนโดยใช้ชื่อเต็ม หรือโดยใช้ชื่อย่อ ของชื่อ โดยต้องป้อนตัวอักษรตัวหนาเป็นอย่างน้อย

ไอเอ็ม	คำอธิบาย
a[lter] <block> <offset> <hex string>	ปรับเปลี่ยนข้อมูลดิสก์
b[map] [<block number>]	แสดงแผนผังการจัดสรรบล็อก
dir[ectory] <inode number> [<fileset>] [R]	แสดงรายการไดเร็กทอรี
d[isplay] [<block> [<offset> [<format> [<count>]]]]	แสดงข้อมูล
dt[ree] { <block number> <inode number> { a   f } }	แสดงผลโหนด dtree
h[elp] [<command>]	แสดงวิธีใช้บนคำสั่งย่อย
ia[g] [<IAG number>] [a   <fileset>]	แสดงหน้า IAG
im[ap] [a   <fileset>]	แสดงแผนผังการจัดสรร inode
i[node] [<inode number>] [a   <fileset>]	แสดง inodes
q[uit]	ออกจาก fsdb
su[perblock] [p   s]	แสดงซูเปอร์บล็อก
x[tree] { <block number> <inode number> { a   f } }	แสดงโหนด xtree

a[lter] <block> <offset> <hex string>  
โดยที่:

ไอเท็ม  
<block>  
<offset>  
<hex string>

คำอธิบาย  
หมายเลขบล็อก (ฐานสิบ)  
ออฟเซตภายในบล็อก (hex)  
สตริงของเลข hex

ปรับเปลี่ยนข้อมูลดิสก์ <hex string> ควรมีจำนวนหลักเป็นเลขคู่

**b[map]** [<block numbers>]

แสดงแม่พิมพ์การจัดสรรบล็อก

<block number> แสดงหน้า **dmap** ซึ่งอธิบาย หมายเลขบล็อกนี้

คำสั่งย่อย:

ไอเท็ม	คำอธิบาย
m	แก้ไขไหนดปัจจุบัน
u	เยียมขมระดับบนของหน้า <b>bmap</b>
l	เยียมขม sibling ฝั่งซ้าย
r	เยียมขม sibling ฝั่งขวา
w	แสดง wmap
p	แสดง pmap
s	แสดง stree
x	ออกจากโหมดคำสั่งย่อย

**dir[ectory]** [<inode number> [<fileset>]][R]

ไอเท็ม	คำอธิบาย
<inode number>	หมายเลข inode ของไดเร็กทอรี (ฐานสิบ)
<fileset>	หมายเลข ขณะนี้ต้องเป็นศูนย์
R	แสดงรายการไดเร็กทอรีย่อยทั้งหมดแบบซ้ำ

แสดงรายการไดเร็กทอรี

**d[isplay]** [<block> [<offset> [<format> [<count>]]]]

ไอเท็ม	คำอธิบาย
<block>	หมายเลขบล็อก (ฐานสิบ)
<offset>	ออฟเซตภายในบล็อก (hex)
<format>	จัดรูปแบบที่แสดงข้อมูล (ดูด้านล่าง)
<count>	จำนวนอ็อบเจกต์ที่จะแสดง (ฐานสิบ)

แสดงข้อมูลในรูปแบบต่างๆ

รูปแบบอาจเป็นหนึ่งในรูปแบบใดต่อไปนี้:

ไอเท็ม	คำอธิบาย	
a	ascii	
i	inode	struct dinode
I	inode allocation map	iag_t
s	superblock	struct superblock
x	hexadecimal	

**dt[ree] {<block number> | <inode number>{a | f}}**

ไอเท็ม	คำอธิบาย
<block number>	หมายเลขบล็อกที่มีหน้า dtree
<inode number>	หมายเลข inode ของไดเร็กทอรี (ฐานสิบ)
{a   f}	'a' ระบุหมายเลขที่เป็น inode รวม 'f' ระบุหมายเลข inode ที่เป็น inode ชุดไฟล์

แสดง root ของ btree ไดเร็กทอรี และใช้โหมดคำสั่งย่อย เพื่อสำรวจ btree

คำสั่งย่อย:

ไอเท็ม	คำอธิบาย
m	แก้ไขไหนดปัจจุบัน
f	สำรวจรายการที่วาง
s	แสดงช่องเสียบสล็อตที่ระบุ
[0-9]+	แสดงรายการ stbl ที่ระบุ
t	แสดง stbl ที่จัดรูปแบบ
u	เยี่ยมชมไหนดบน (ไม่ใช่พาร์เนตไดเร็กทอรี)
d	เยี่ยมชมไหนดลูก
x	ออกจากโหมดคำสั่งย่อย

**h[elp] [<command>]**

ไอเท็ม	คำอธิบาย
<command>	ชื่อคำสั่ง

พิมพ์ข้อความวิธีใช้ แสดงคำสั่งทั้งหมดหากไม่มีพารามิเตอร์

**ia[g] [<IAG number>] [a | <fileset>]**

ไอเท็ม	คำอธิบาย
<IAG number>	หมายเลข IAG (ฐานสิบ)
a	ใช้ตาราง inode รวม
<fileset>	หมายเลขชุดไฟล์ (ขณะนี้ต้องเป็นศูนย์)

แสดงข้อมูล iag และเข้าสู่โหมดคำสั่งย่อย

คำสั่งย่อย:

ไอเท็ม	คำอธิบาย
e	แสดง/แก้ไขแม่พจนานุกรม inode
m	แก้ไข iag
p	แสดง/แก้ไขแม่พจนานุกรม
w	แสดง/แก้ไขแม่พจนานุกรมที่ทำงาน

**im[ap] [a | <fileset>]**

ไอเท็ม	คำอธิบาย
a	ใช้ตาราง inode รวม
<fileset>	หมายเลขชุดไฟล์ (ขณะนี้ต้องเป็นศูนย์)

แสดงแม่พจนานุกรม inode ที่ระบุและเข้าสู่โหมดคำสั่งย่อย

คำสั่งย่อย:

ไอเท็ม	คำอธิบาย
e	แสดง/แก้ไขแม่พจนานุกรม inode
m	แก้ไข iag
p	แสดง/แก้ไขแม่พจนานุกรม

**i[node] [<inode number>] [a | <fileset>]**

ไอเท็ม	คำอธิบาย
<inode number>	หมายเลข Inode (ฐานสิบ)
a	ใช้ตาราง inode รวม
fileset	หมายเลขชุดไฟล์ (ขณะนี้ต้องเป็นศูนย์)

แสดงข้อมูล inode และเข้าสู่โหมดคำสั่งย่อย

คำสั่งย่อย:

ไอเท็ม	คำอธิบาย
m	แก้ไข inode
t	แสดง / แก้ไข b-tree ของ inode
e	แสดง / แก้ไข EAs ของ inode

**หมายเหตุ:** คำสั่ง `fsdb` เข้าใจทั้งแอตทริบิวต์ส่วนขยาย v1 และ v2 ลักษณะการทำงานเมื่อดู EAs จะขึ้นอยู่กับรูปแบบ สำหรับ inode ที่กำลังดู

สำหรับ v1 หลังจากแสดง EAs ของ inode คุณสามารถแก้ไขรายการ `pxdTable` หรือ `eaDirectory` ระบุออฟชันการแก้ไข จากนั้นดับงชี้ `pxdTable` หรือ `eaDirectory` และออฟเซตลงในตาราง

สำหรับ v2 EAs ถูกแสดงโดยใช้รูปแบบคำสั่งย่อย `dtree` คำสั่งย่อย `dtree` ทั้งหมดจะพร้อมสำหรับการดำเนินการต่างๆ บน EAs

`q[uit]` ออกจาก `fsdb`

`su[perblock] [p | s]`

ไอเท็ม	คำอธิบาย
p	แสดงซูเปอร์บล็อกหลัก
s	แสดงซูเปอร์บล็อกรอง

แสดงข้อมูลซูเปอร์บล็อก

`x[tree] { <block number> | <inode number> {a | f} }`

ไอเท็ม	คำอธิบาย
<block number>	หมายเลขบล็อก (ฐานสิบ)
<inode number>	หมายเลข inode
{a   f}	'a' ระบุหมายเลขที่เป็น inode รวม 'f' ระบุหมายเลข inode ที่เป็น inode ชุดไฟล์

แสดงหนึ่งโหนดของ `xtree` และใช้โหมดคำสั่งย่อยเพื่อสำรวจ `xtree`

คำสั่งย่อย:

ไอเท็ม	คำอธิบาย
m	แก้ไขโหมดปัจจุบัน
u	เยี่ยมชมโหมดบน
d	เยี่ยมชมโหมดลูก
n	เยี่ยมชม sibling พี่ชาย
p	เยี่ยมชม sibling พี่ชาย
s	เลือกรายการ xad เพื่อดู
x	ออกจากโหมดคำสั่งย่อย

## คำสั่งย่อยสแน็ปช็อต JFS2

คำสั่งย่อยเหล่านี้สามารถป้อนโดยใช้ชื่อเต็ม หรือโดยใช้เซตย่อย ของชื่อ โดยต้องป้อนตัวอักษรตัวหนาเป็นอย่างน้อย

ไอเท็ม	คำอธิบาย
a[iter] <block> <offset> <hex string>	ปรับเปลี่ยนข้อมูลดิสก์
b[map]	แสดงสำเนา xtree บล็อกแม็พ
d[isplay] [<block> [<offset> [<format> [<count>]]]]	แสดงข้อมูล
h[elp] [<command>]	แสดงวิธีใช้บนคำสั่งย่อย
q[uit]	ออกจาก fsdb
st[able] [<block number>]	แสดงตารางสแน็ปช็อตสรุป
s[map] <block number>	แสดงสแน็ปช็อตบิตแม็พ
su[perblock]	แสดงซูเปอร์บล็อก

**a[iter] <block> <offset> <hex string>**

โดยที่:

ไอเท็ม	คำอธิบาย
<block>	หมายเลขบล็อก (ฐานสิบ)
<offset>	ออฟเซตภายในบล็อก (hex)
<hex string>	สตริงของเลข hex

ปรับเปลี่ยนข้อมูลดิสก์ <hex string> ควรมีจำนวนหลักเป็นเลขคู่

**b[map]**

แสดงสำเนา xtree บล็อกแม็พ

**d[isplay] [<block> [<offset> [<format> [<count>]]]]**

ไอเท็ม	คำอธิบาย
<block>	หมายเลขบล็อก (ฐานสิบ)
<offset>	ออฟเซตภายในบล็อก (hex)
<format>	จัดรูปแบบที่แสดงข้อมูล (ดูด้านล่าง)
<count>	จำนวนอ็อบเจกต์ที่จะแสดง (ฐานสิบ)

แสดงข้อมูลในรูปแบบต่างๆ

รูปแบบอาจเป็นหนึ่งในรูปแบบใดต่อไปนี้:

ไอเท็ม	คำอธิบาย
a	ascii
s	ส่วนหัวสแน็ปช็อตเซกเมนต์
t	หน้าต่างสแน็ปช็อต
x	หน้า xtree

**h[elp][<command>]**

ไอเท็ม	คำอธิบาย
<command>	ชื่อคำสั่ง

แสดงวิธีใช้บนคำสั่งย่อย

**q[uit]** ออกจาก fsdb

**st[able][<block number>]**

โดยที่:

ไอเท็ม	คำอธิบาย
<block number>	หมายเลขบล็อก (ฐานสิบ)

แสดงตารางสแน็ปช็อตสรุป

**s[map][<block number>]**

โดยที่:

ไอเท็ม  
<block number>

คำอธิบาย  
หมายเลขบล็อก (ฐานสิบ)

แสดงสแน็ปช็อตบิตแม็พ

su[perblock]

แสดงซูเปอร์บล็อก

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

ตัวอย่างต่อไปนี้แสดงคำสั่งย่อยที่คุณสามารถใช้ หลังเริ่มทำงานคำสั่ง **fsdb** บนระบบไฟล์ JFS

1. ในการแสดง i-node ป้อน:

```
386i
```

คำสั่งนี้แสดง i-node 386 ในรูปแบบ i-node ซึ่งในขณะนี้จะเป็น i-node ปัจจุบัน

2. ในการเปลี่ยนจำนวนลิงก์ i-node ปัจจุบันให้มีค่าเป็น 4 ให้ป้อน:

```
ln=4
```

3. ในการเพิ่มจำนวนลิงก์ของ i-node ปัจจุบันด้วยค่า 1 ให้ป้อน:

```
ln+=1
```

4. ในการแสดงส่วนของไฟล์ที่เชื่อมโยงกับ i-node ปัจจุบัน ให้ป้อน:

```
fc
```

คำสั่งนี้แสดงบล็อก 0 ของไฟล์ที่เชื่อมโยงกับ i-node ปัจจุบันในไบต์ ASCII

5. ในการแสดงรายการของไดเร็กทอรีให้ป้อน:

```
2i.fd
```

คำสั่งนี้เปลี่ยนแปลง i-node ปัจจุบันเป็น i-node root (i-node 2) จากนั้นแสดงรายการไดเร็กทอรีในบล็อกแรกที่สัมพันธ์กับ i-node นั้น รายการสุดท้ายอย่างน้อยหนึ่งรายการที่แสดงอาจมีหมายเลข i-node เป็น 0 (ศูนย์) เหล่านี้เป็นไดเร็กทอรีบล็อกที่ไม่ใช่ซึ่งรายการไม่สามารถจัดการได้เหมือนในตัวอย่างถัดไป

6. ในการลงไปที่ระดับล่างของลำดับไดเร็กทอรีให้ป้อน:

```
d5i.fc
```

คำสั่งนี้จะเปลี่ยน i-node ปัจจุบันเป็นโหนดที่สัมพันธ์กับรายการไดเรกทอรี 5 จากนั้นแสดงบล็อกแรกของไฟล์เป็นข้อความ ASCII (fc) รายการไดเรกทอรีถูกกำหนดตัวเลขโดยเริ่มต้นจาก 0

7. ในการแสดงบล็อกเมื่อคุณทราบหมายเลขบล็อกให้ป้อน:

1b.p0o

คำสั่งนี้แสดงซูเปอร์บล็อก (บล็อก 1) ของระบบไฟล์เป็นเลขฐานแปด

8. ในการเปลี่ยน i-node ของรายการไดเรกทอรีให้ป้อน:

2i.a0b.d7=3

คำสั่งนี้เปลี่ยน i-node ของรายการไดเรกทอรี 7 ในไดเรกทอรี root (2i) เป็น 3 ตัวอย่างนี้ยังแสดงให้เห็นว่าหลายๆ การดำเนินการถูกรวมกันบน บรรทัดเดียวได้อย่างไร

9. ในการเปลี่ยนชื่อไฟล์ของรายการไดเรกทอรีให้ป้อน:

d7.nm="chap1.rec"

คำสั่งนี้เปลี่ยนฟิลด์ชื่อของรายการไดเรกทอรี 7 เป็น chap1.rec

10. ในการแสดงบล็อกที่กำหนดของไฟล์ที่สัมพันธ์กับ i-node ปัจจุบันให้ป้อน:

a2b.p0d

คำสั่งนี้แสดงบล็อก 2 ของ i-node ปัจจุบันเป็นรายการไดเรกทอรี

11. ในการแสดงเนื้อหาของบล็อกทางอ้อมเดียวที่บล็อก 7 ให้ป้อน:

7b.p0S

คำสั่งนี้แสดงหมายเลขบล็อกที่จัดสรรแก่ i-node ที่มีบล็อกทางอ้อมเดียวที่บล็อก 7

12. ในการแสดงหน้าแรกของดิสก์แม่พิมพ์ให้ป้อน:

0M

13. ในการแสดง 10 เวิร์ดแรกของแม่พิมพ์การจัดสรรบล็อกถาวรเป็นเลขฐานสิบหกให้ป้อน:

mp1.p10x

คำสั่งนี้แสดงบิตแม่พิมพ์ การจัดสรรที่แอดเดรสปัจจุบัน ตัวอย่างเช่น ที่ 0M

ตัวอย่างต่อไปนี้จะแสดงบางคำสั่งย่อยที่คุณสามารถใช้บนระบบไฟล์ JFS2

**ข้อควรสนใจ:** อย่าใช้คำสั่งย่อย JFS2 เพื่อแก้ไขระบบไฟล์

1. ในการแสดง i-node ป้อน:

inode 2

คำสั่งนี้แสดง i-node 2 ในรูปแบบ i-node

2. ในการแสดงรายการของไดเรกทอรีให้ป้อน:

dir 2

คำสั่งนี้แสดงรายการไดเรกทอรีที่สัมพันธ์กับ i-node 2

3. ในการแสดงบล็อกที่มีหมายเลขบล็อก 0x1000 ให้ป้อน:

display 0x1000

คำสั่งนี้แสดงบล็อกที่ระบบไฟล์ในรูปแบบเลขฐานสิบหก

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/sbin	มีคำสั่ง fsdb
/etc/filesystems	มีข้อมูลบนระบบไฟล์

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dfsc” ในหน้า 127

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง dir

ไฟล์ filsys.h

รูทีนย่อย read

---

## คำสั่ง fsplit

### วัตถุประสงค์

แบ่งซอร์สโค้ด FORTRAN ออกเป็นรูทีนไฟล์แยก

### ไวยากรณ์

```
fsplit[ -e SubprogramUnit ] ...[ File ]
```

### คำอธิบาย

คำสั่ง `fsplit` รับอินพุตเป็น ไฟล์ หรืออินพุตมาตรฐานที่มีซอร์สโค้ด FORTRAN และแบ่ง อินพุตออกเป็นรูทีนไฟล์แยกในรูปแบบของ `name.f` โดยที่ `name` คือชื่อ ของโปรแกรมยูนิต (ตัวอย่างเช่น ฟังก์ชัน รูทีนย่อย ข้อมูลบล็อก หรือโปรแกรม)

ชื่อสำหรับโปรแกรมย่อยข้อมูลบล็อกที่ไม่ได้ตั้งชื่อ มีรูปแบบ `blkdtaNNN.f` โดยที่ `NNN` คือเลขสามหลัก และไฟล์ของที่ยังไม่มีชื่อนี้ สำหรับโปรแกรมหลัก ที่ไม่ได้ตั้งชื่อ ชื่อจะมีรูปแบบ `mainNNN.f` หากมีข้อผิดพลาดในการจัดหมวดหมู่โปรแกรมยูนิต หรือหาก `name.f` มีอยู่แล้ว โปรแกรมยูนิตจะถูกเก็บ ในไฟล์ที่มีรูปแบบ `zzzNNN.f` โดยที่ `zzzNNN.f` ยังไม่มีอยู่

หมายเหตุ: คำสั่ง `fsplit` สมมติว่าชื่อ โปรแกรมย่อยอยู่บนค่าแรกที่ไม่ใช่บรรทัดรับคำสั่งของยูนิตโปรแกรมย่อย รูปแบบซอร์สที่ไม่เป็นไปตามมาตรฐานอาจทำให้คำสั่งไม่สามารถทำงานได้ และอาจก่อให้เกิดผลลัพธ์ที่ไม่คาดคิด

### แฟล็ก

ไอเท็ม  
-e SubprogramUnit

คำอธิบาย  
ทำให้เฉพาะยูนิทโปรแกรมย่อยที่ระบุที่ถูกแบ่งออกเป็นไฟล์ แยก โดยปกติแต่ละยูนิทโปรแกรมย่อยจะแบ่งออกเป็นไฟล์แยก

แฟล็ก -e สามารถใช้สำหรับโปรแกรมหลักที่ระบุชื่อ และโปรแกรมย่อยข้อมูลแบบบล็อก หากไม่พบชื่อที่ระบุผ่านอ็อปชัน -e การวินิจฉัยจะเขียนไปยังข้อผิดพลาดมาตรฐาน

## ตัวอย่าง

คำสั่ง **fsplit** ต่อไปนี้ แบ่งโปรแกรมย่อย readit และ doit ออกเป็นไฟล์แยก:

```
fsplit -e readit -e doit prog.f
```

## ไฟล์

ไอเท็ม  
/usr/bin/fsplit

คำอธิบาย  
มีคำสั่ง **fsplit**

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **asa**

คำสั่ง **struct**

---

## คำสั่ง ftp

### วัตถุประสงค์

ถ่ายโอนไฟล์ระหว่างโฮสต์โลคัลและรีโมต

### ไวยากรณ์

```
ftp [ -d ] [ -D DataConnTimeOut ] [ -g ] [ -i ] [ -n ] [ -v ] [ -f ] [ -K ] [ -k realm ] [ -q ] [ -C ] [ -s ] [ -M ] [ HostName [ Port ] ] [ -H ]
```

### คำอธิบาย

คำสั่ง **ftp** ใช้ File Transfer Protocol (FTP) ในการถ่ายโอนไฟล์ระหว่างโฮสต์โลคัลโฮสต์ และรีโมตโฮสต์หรือระหว่างสองรีโมตโฮสต์ การดำเนินการแบบรีโมต ของคำสั่ง **ftp** ไม่แนะนำ

โปรโตคอล FTP อนุญาตให้มีการถ่ายโอนข้อมูลระหว่าง โฮสต์ที่ใช้ระบบไฟล์ไม่เหมือนกัน ถึงแม้ว่าโปรโตคอลมีความยืดหยุ่นสูงในการถ่ายโอนข้อมูล โปรโตคอลไม่ รักษาไฟล์แอตทริบิวต์ (เช่นโหมดการป้องกันหรือเวลาการแก้ไข ไฟล์) ที่จำเพาะกับระบบไฟล์ นอกจากนี้ โปรโตคอล FTP ยังสร้างสมมุติฐานบางส่วนเกี่ยวกับโครงสร้างโดยรวม ของระบบไฟล์และไม่จัดเตรียมหรืออนุญาตฟังก์ชันเช่นการ คัดลอกไดเร็กทอรีย่อยแบบวนซ้ำ

หมายเหตุ: ถ้าคุณกำลังถ่ายโอนไฟล์ระหว่าง ระบบ และต้องส่งวนไฟล์แอตทริบิวต์ไว้ หรือคัดลอกไดเร็กทอรีย่อยแบบเรียกซ้ำ โดยใช้คำสั่ง **recp**

### การเรียกคำสั่งย่อย

ที่พร้อมต์ ftp> คุณสามารถป้อนคำสั่งย่อยเพื่อดำเนินงานเช่น การแสดงรายการไดเรกทอรีรีโมต การเปลี่ยนแปลงไดเรกทอรีโลคัลและรีโมตปัจจุบัน การถ่ายโอนไฟล์หลายๆ ไฟล์ในการร้องขอเดียว การสร้างและการลบไดเรกทอรี และการ escape โลคัลเซลล์เพื่อดำเนินคำสั่งเซลล์ ดูที่ส่วน คำสั่งย่อย สำหรับรายละเอียดของแต่ละ คำสั่งย่อย

ถ้าคุณเรียกใช้งานคำสั่ง ftp และไม่ได้ระบุพารามิเตอร์ HostName สำหรับโฮสต์รีโมต คำสั่ง ftp แสดงพร้อมต์ ftp> ทันทีและรองคำสั่งย่อย ftp เมื่อต้องการเชื่อมต่อ กับโฮสต์รีโมต ให้เรียกใช้งานคำสั่งย่อย open เมื่อคำสั่ง ftp เชื่อมต่อ กับโฮสต์รีโมต ดังนั้นคำสั่ง ftp จะพร้อมต์รับชื่อล็อกอิน และรหัสผ่านก่อนแสดงพร้อมต์ ftp> อีกครั้ง คำสั่ง ftp จะไม่สำเร็จถ้าไม่มีรหัสผ่านถูกกำหนดไว้ที่รีโมตโฮสต์ สำหรับชื่อล็อกอิน

ตัวแปลคำสั่ง ftp ซึ่งรองรับคำสั่งย่อยทั้งหมดที่ป้อน ที่พร้อมต์ ftp> จะมีโปรแกรมอำนวยความสะดวกที่ไม่มีสำหรับ โปรแกรมถ่ายโอนไฟล์ส่วนใหญ่ เช่น:

- การจัดการพารามิเตอร์ file-name กับคำสั่งย่อย ftp
- การรวบรวมกลุ่มคำสั่งย่อยลงใน แมโครคำสั่งย่อยเดียว
- การโหลดแมโครจากไฟล์ \$HOME/.netrc

เครื่องมือเหล่านี้ช่วยให้ทำซ้ำๆ ง่ายขึ้น และอนุญาตให้คุณใช้คำสั่ง ftp ในโหมด unattended

interpreter คำสั่งจัดการพารามิเตอร์ file-name ตามกฎต่อไปนี้:

- ถ้ามีการระบุ - (เครื่องหมายขีดคั่น) สำหรับพารามิเตอร์ อินพุตมาตรฐาน (stdin) จะถูกใช้ในการอ่านการดำเนินการและเอาต์พุตมาตรฐาน (stdout) จะถูกใช้ในการเขียนข้อมูล
- ถ้าไม่มีการตรวจสอบก่อนหน้า และมีการเปิดใช้ ส่วนขยาย file-name (ดูที่แฟล็ก -g หรือคำสั่งย่อย glob ) interpreter ขยายชื่อไฟล์ ตามกฎของเซลล์ C เมื่อมีการเปิดใช้ globbing และอักขระที่มีการจับคู่รูปแบบถูกใช้ในคำสั่งย่อย ที่ต้องการชื่อไฟล์เดียว ผลลัพธ์อาจต่างไปจากที่ต้องการ

ตัวอย่างคำสั่งย่อย append และ put ดำเนินการกับส่วนขยาย file-name แล้วใช้เฉพาะชื่อไฟล์แรก ที่สร้าง คำสั่งย่อย ftp อื่น เช่น cd, delete, get, mkdir, rename และ rmdir ไม่ดำเนินการ ส่วนขยาย file-name และรับอักขระที่มีการจับคู่ตามรูปแบบตามตัวอักษร

- สำหรับคำสั่งย่อย get, put, mget และ mput interpreter มีความสามารถในการแปลและแม้กระทั่งระหว่างลักษณะไวยากรณ์โลคัลและรีโมต file-name ต่างกัน (ดูที่คำสั่งย่อย case, ntrans, และ nmap ) และความสามารถในการแก้ไขชื่อไฟล์โลคัล ถ้าชื่อมีการซ้ำกัน (ดูที่ คำสั่งย่อย runique ) นอกจากนี้ คำสั่ง ftp สามารถส่งคำแนะนำไปที่รีโมต ftpd เซิร์ฟเวอร์เพื่อแก้ไขชื่อรีโมตไฟล์ ถ้าไฟล์มีการซ้ำกัน (ดูที่คำสั่งย่อย sunique )
- ใช้ัญประกาศ (" ") เพื่อระบุ พารามิเตอร์ที่มีอักขระช่องว่าง

หมายเหตุ: ตัวแปลคำสั่ง ftp ไม่รองรับไฟฟ์ และไม่จำเป็นในการสนับสนุนชื่อไฟล์ อักขระมัลติไบต์ทั้งหมด

เมื่อต้องการสิ้นสุดเซลล์ ftp เมื่อคุณ กำลังรันแบบโต้ตอบ ให้ใช้คำสั่งย่อย quit หรือ bye หรือลำดับปุ่ม End of File (Ctrl-D) ที่พร้อมต์ ftp> เมื่อต้องการจบการถ่ายโอนไฟล์ ก่อนที่จะสมบูรณ์ ให้กดลำดับคีย์ Interrupt ดีโพลต์ลำดับคีย์ Interrupt คือ Ctrl-C คำสั่ง stty สามารถถูก ใช้เพื่อกำหนดลำดับคีย์นี้ใหม่ได้

โดยปกติคำสั่ง ftp หยุดการถ่ายโอนที่กำลังส่ง (จากโลคัลโฮสต์ไปที่รีโมตโฮสต์) ในทันที คำสั่ง ftp หยุดการถ่ายโอน ที่ได้รับอยู่ (จากรีโมตโฮสต์ไปที่โลคัลโฮสต์) โดยการส่ง คำสั่ง FTP ABOR ไปที่รีโมต FTP เซิร์ฟเวอร์และไม่สนใจแฟ็กเกิดการถ่ายโอน

ไฟล์ที่เข้ามาจนกว่ารีโมตเซิร์ฟเวอร์หยุดการส่งข้อมูล ถ้าเซิร์ฟเวอร์รีโมตไม่สนับสนุนคำสั่งเครื่อง คำสั่ง ftp จะไม่แสดง พร้อมทั้ง ftp> จะกว่าเซิร์ฟเวอร์รีโมตจะส่งไฟล์ที่ร้องขอ ทั้งหมดแล้ว นอกจากนี้ ถ้ารีโมตเซิร์ฟเวอร์ทำงาน ที่ไม่คาดคิด คุณอาจจำเป็นต้องจบกระบวนการโลคัล ftp

## ความปลอดภัยและการล็อกอินอัตโนมัติ

ถ้า Standard คือเมธอดการพิสูจน์ตัวตนปัจจุบัน:

คำสั่ง ftp จะจัดการ ความปลอดภัยด้วย โดยการส่งรหัสผ่านไปที่รีโมตโฮสต์และอนุญาต การล็อกอินอัตโนมัติ, การถ่ายโอนไฟล์ และการล็อกออฟ

ถ้าคุณเรียกใช้งานคำสั่ง ftp และระบุชื่อโฮสต์ (HostName) ของ รีโมตโฮสต์คำสั่ง ftp พยายามสร้าง การเชื่อมต่อไปที่โฮสต์ที่ระบุ ถ้าคำสั่ง ftp เชื่อมต่อสำเร็จ คำสั่ง ftp ค้นหา ไฟล์โลคัล \$HOME/.netrc ในไดเรกทอรีปัจจุบัน หรือไดเรกทอรีโฮมของคุณ ถ้ามีไฟล์อยู่ คำสั่ง ftp ค้นหาไฟล์เพื่อการเริ่มต้นกระบวนการล็อกอินและนิยาม แมโครคำสั่งสำหรับรีโมตโฮสต์ ถ้าไฟล์ \$HOME/.netrc หรือรายการล็อกอินอัตโนมัติไม่มีอยู่ หรือถ้าระบบของคุณ รักษาความปลอดภัยด้วยคำสั่ง securetcip คำสั่ง ftp พร้อมผู้ใช้เพื่อขอชื่อผู้ใช้ และรหัสผ่าน คำสั่งแสดงพร้อมแต่ไม่ว่าจะมีการระบุ พารามิเตอร์ HostName หรือไม่ บนบรรทัดคำสั่ง

หมายเหตุ: ระบบการคิว ไม่สนับสนุนชื่อโฮสต์มัลติไปต์

ถ้าคำสั่ง ftp พบ รายการล็อกอินอัตโนมัติ \$HOME/.netrc สำหรับ โฮสต์ที่ระบุ คำสั่ง ftp จะพยายามใช้ ข้อมูลในรายการนั้น เพื่อล็อกอินไปที่รีโมตโฮสต์ คำสั่ง ftp ยังโหลดแมโครคำสั่งทั้งหมดที่กำหนด ในรายการ ในบางกรณี (ตัวอย่างเช่น เมื่อรหัสผ่านที่ร้องขอนั้นไม่แสดงรายการอยู่ในรายการล็อกอิน อัตโนมัติ) คำสั่ง ftp จะพร้อมรับรหัสผ่านก่อน แสดงพร้อม ftp>

เมื่อคำสั่ง ftp ดำเนินการล็อกอินอัตโนมัติสมบูรณ์ คำสั่ง ftp เรียกใช้งานแมโคร init ถ้าแมโครถูกกำหนด ในรายการล็อกอินอัตโนมัติ ถ้าแมโคร init ไม่มีอยู่ หรือไม่มีคำสั่งย่อย quit หรือ bye คำสั่ง ftp จะแสดงพร้อม ftp> และ รอคำสั่งย่อย

หมายเหตุ: ชื่อผู้ใช้รีโมตที่ระบุที่ พร้อมต์ หรือในไฟล์ \$HOME/.netrc ต้องมีอยู่ และมี รหัสผ่านกำหนดที่โฮสต์รีโมต มิฉะนั้น คำสั่ง ftp จะล้มเหลว

หาก Kerberos 5 คือเมธอดการพิสูจน์ตัวตนปัจจุบัน

คำสั่ง ftp จะใช้ส่วนขยายกับ คำกำหนด ftp ตามที่กำหนดในเอกสารแบบร่าง IETF "draft-ietf-cat-ftpsec-09.txt" ส่วนขยาย ความปลอดภัย FTP จะถูกนำมาใช้โดยใช้กลไกความปลอดภัย Generic Security Service API (GSSAPI) GSSAPI มี เซอร์วิสที่อิสระต่อการรักษาความปลอดภัยภายใน และกลไก การสื่อสาร GSSAPI ถูกนิยามใน rfc 1508 และ 1509

คำสั่ง ftp จะใช้คำสั่ง AUTH และ ADAT เพื่อพิสูจน์ตัวตนกับ ftpd daemon ถ้าทั้งสองสนับสนุนการพิสูจน์ตัวตน Kerberos จะใช้ DCE credentials ของผู้ใช้โลคัลเพื่อพิสูจน์ตัวตนผู้ใช้บนระบบรีโมต ถ้าล้มเหลวและการพิสูจน์ตัวตน Standard ถูกตั้งค่าไว้ในทั้งสอง ระบบ กระบวนการที่อธิบายไว้ทางด้านบนจะถูกใช้

พารามิเตอร์ HostName parameter คือชื่อของเครื่องโฮสต์ ซึ่งไฟล์ถูกถ่ายโอน พารามิเตอร์ Port ทางเลือกระบุ ID ของ พอร์ตที่จะทำการถ่ายโอน (ไฟล์ /etc/services ระบุดีพอลต์พอร์ต)

**หมายเหตุ:** ถ้าค่าของรีจิสทรี ถูกตั้งค่าเป็นแบบแผนการพิสูจน์ตัวตนปัจจุบันที่ถูกต้อง การพิสูจน์ตัวตน FTP จะทำงานกับรหัสผ่านไอดีเร็กทอรีที่แอดทีฟ ถ้าค่าของรีจิสทรี ถูกตั้งค่าเป็น null ดังนั้นค่าดีฟอลต์ของไฟล์ (การพิสูจน์ตัวตนผู้ใช้โลคัล) จะถูกใช้

### การสนับสนุน Transport Layer Security

คำสั่ง ftp สนับสนุน Transport Layer Security (TLS) ดังกำหนดใน RFC 4217 TLS เป็นโปรโตคอล การเข้ารหัสที่จัดให้มีการสื่อสารแบบปลอดภัยระหว่างไคลเอ็นต์และเซิร์ฟเวอร์

คำสั่ง ftp ใช้คำสั่ง AUTH TLS และ PROT P เพื่อรักษาความปลอดภัยการสื่อสารกับ ftpd daemon ถ้าคำสั่ง AUTH TLS และ PROT P ทั้งสองสนับสนุน โปรโตคอล TLS ดังนั้นจะสร้างช่องสัญญาณที่มีการรักษาความปลอดภัยขึ้น เฉพาะเมธอด การพิสูจน์ตัวตนแบบมาตรฐาน เท่านั้นที่ได้รับการสนับสนุน

ถ้าแฟล็ก -s ถูก ระบุเมื่อคุณรันคำสั่ง ftp ดังนั้นคำสั่ง ftp จะค้นหาไฟล์ \$HOME/.ftpcnf โลคัลในโฮม ไดเร็กทอรีของคุณ ถ้าพบไฟล์ คำสั่ง ftp จะใช้ พารามิเตอร์คอนฟิกูเรชันต่อไปนี้เพื่อตั้งค่าเซสชัน TLS กับ เซิร์ฟเวอร์ ถ้าไม่พบไฟล์ หรือพารามิเตอร์คอนฟิกูเรชัน หายไป คำสั่ง ftp จะพยายามเชื่อมต่อกับเซิร์ฟเวอร์ โดยไม่ใช้พารามิเตอร์คอนฟิกูเรชัน

### CRL\_PATH

พารามิเตอร์ CRL\_PATH จัดให้มีพารไปยังไฟล์รายการ เรียกคืนใบรับรอง ซึ่งต้องอยู่ในรูปแบบ privacy enhanced mail (PEM) ถ้า ระบุใบรับรองดิจิทัลที่มีให้โดยเซิร์ฟเวอร์ จะถูกตรวจสอบกับรายการเรียกคืนใบรับรอง ถ้าใบรับรองถูกเรียกคืน เซสชัน TLS จะล้มเหลว ถ้าไม่ระบุ ใบรับรอง ดิจิทัลจะไม่ถูกเก็บรายการเรียกคืนใบรับรอง

### CA\_PATH

พารามิเตอร์ CA\_PATH จัดเตรียมพารไปยังไฟล์สิทธิของใบรับรอง ที่ต้องอยู่ในรูปแบบ PEM ถ้าระบุ ใบรับรองเซิร์ฟเวอร์ จะถูกตรวจสอบกับผู้ออกใบรับรอง ถ้าใบรับรองดิจิทัล ถูกกำหนดให้มีโดยเซิร์ฟเวอร์ไม่ได้รับการลงนามโดยเจ้าหน้าที่รักษาความปลอดภัย เซสชัน TLS จะล้มเหลว ถ้าไม่ระบุ ใบรับรองดิจิทัลที่มีให้โดยเซิร์ฟเวอร์จะไม่ถูกตรวจสอบกับรายการเรียกคืน ใบรับรอง

### CIPHER\_LIST

หากระบุพารามิเตอร์ CIPHER\_LIST ไว้ รายการจะถูกใช้ในระหว่างเซสชัน TLS ถ้าไม่ใช้ รายการการเข้ารหัสดีฟอลต์ จะถูกนำมาใช้

### DEPTH

ถ้าพารามิเตอร์คอนฟิกูเรชัน CA\_PATH ถูกระบุ ค่า DEPTH จะถูกใช้เพื่อตรวจสอบว่าใบรับรองที่มีให้โดยเซิร์ฟเวอร์ ftpd อยู่ในลำดับชั้นใบรับรองดิจิทัล ถ้าไม่มีการจัดเตรียม ค่าดีฟอลต์คือ 9 จะถูกนำมาใช้

### CERTIFICATE

พารามิเตอร์ CERTIFICATE มีพารไปยังไฟล์เชื่อมโยงของ ใบรับรองดิจิทัลที่ถูกต้องในรูปแบบ PEM ไฟล์นี้ ถูกใช้ในเซสชัน TLS

### CERTIFICATE\_PRIVATE\_KEY

พารามิเตอร์ CERTIFICATE\_PRIVATE\_KEY มีพารไปยังไพรเวตคีย์ ของใบรับรองในรูปแบบ PEM ซึ่งถูกใช้ในระหว่างเซสชัน TLS เมื่อต้องการสนับสนุน TLS คุณต้องติดตั้งเวอร์ชันล่าสุดของ เครื่องมือ OpenSSL จากเว็บไซต์ AIX Web Download Pack Programs

สำหรับระบบ Trusted AIX

ผู้ใช้จะถูกกำหนดลีดออกอนดีฟอลต์ Sensitivity Label (SL) และ Integrity Label (TL) ซึ่งเป็น SL ที่มีประสิทธิภาพและ TL ที่มีประสิทธิภาพของกระบวนการของผู้ใช้หลังจากที่ล็อกอินเป็นผลสำเร็จ หากผู้ใช้ไม่ต้องการ ล็อกอินที่ล็อกอินดีฟอลต์ SL ผู้ใช้สามารถเลือก เพื่อจัดหา SL อื่นในเวลาทีล็อกอินโดยใช้ชื่อพจน -e SL ที่ระบุโดยผู้ใช้ต้องถูกควบคุมโดย clearance ของผู้ใช้ที่มีในขอบเขตการแต่งตั้งของระบบ TL ไม่สามารถระบุได้โดยผู้ใช้งาน เวลาทีล็อกอิน ล็อกอินดีฟอลต์ SL และ TL ถูกนิยามอยู่ในไฟล์ /etc/security/user พร้อมกับชื่อผู้ใช้และการกลางข้อมูลสำหรับแต่ละผู้ใช้ หากต้องการใช้ชื่อพจน -e เคอร์เนลบน บิตเน็ตเวิร์กที่ ไว้วางใจได้ของฝั่งเซิร์ฟเวอร์ ต้องถูกปิด

หมายเหตุ: ผู้ใช้ที่มี ID ผู้ใช้น้อยกว่าหรือเท่ากับ 128 ไม่สามารถล็อกอินเข้าสู่ระบบ Trusted AIX รีโมต

## แฟล็ก

### ไอเท็ม

- C** คำอธิบาย  
อนุญาตให้ผู้ใช้ระบุไฟล์ขาออกที่ส่งโดยใช้ คำสั่ง `send_file` ต้องถูกแคชใน Network Buffer Cache (NBC) แฟล็กนี้ไม่สามารถ ใช้ได้นอกจากมีการระบุแฟล็ก `-q` แฟล็กนี้ใช้ได้เฉพาะ เมื่อไฟล์กำลังถูกส่งออกในโหมดไบนารี โดยไม่มีการป้องกัน
- d** ส่งข้อมูลดีบั๊กเกี่ยวกับการดำเนินการคำสั่ง `ftp` ไปที่ `syslogd` daemon ถ้าคุณระบุแฟล็ก `-d` คุณต้องแก้ไขไฟล์ `/etc/syslog.conf` และเพิ่มหนึ่งในรายการ ต่อไปนี้:
- ```
user.info FileName
```
- หรือ
- ```
user.debug FileName
```
- หมายเหตุ:** ระดับการดีบั๊ก `syslogd` daemon ประกอบด้วยข้อความระดับข้อมูล
- ถ้าคุณไม่แก้ไขไฟล์ `/etc/syslog.conf` จะไม่มี การสร้างข้อความขึ้น หลังจากการเปลี่ยนไฟล์ `/etc/syslog.conf` วันคำสั่ง `refresh -s syslogd` หรือ `kill -1 SyslogdPID` เพื่อ แจ้ง `syslogd` daemon ถึงการเปลี่ยนแปลง กับไฟล์คอนฟิกูเรชัน สำหรับข้อมูลเพิ่มเติมเกี่ยวกับระดับดีบั๊ก อ้างอิง ที่ไฟล์ `/etc/syslog.conf` รวมทั้ง อ้างอิงคำสั่งย่อย `debug`
- D DataConnTimeOut**  
ระบุจำนวนสูงสุดของวินาทีที่ คำสั่ง `ftp` คงการเชื่อมต่อข้อมูลไว้ ค่าดีฟอลต์ คือ 300 วินาทีและมีช่วงได้จาก 300 วินาทีจนถึง 3600 วินาที
- f** เป็นสาเหตุทำให้หนึ่งสื่อบรรณถูกส่งต่อ แฟล็กนี้จะถูกละเว้นหาก Kerberos 5 ไม่ใช้เมธอดของการพิสูจน์ตัวตนในปัจจุบัน
- g** ปิดใช้งานส่วนขยายของ metacharacters ในชื่อไฟล์ การแปล metacharacters สามารถถูกอ้างอิงเป็นการขยาย (บางครั้งเรียกว่า globbing) ชื่อไฟล์ ดูที่คำสั่งย่อย `glob`
- H** เปิดการล็อกการตรวจสอบสำหรับเหตุการณ์ FILE\_Unlink ถ้าเปิดใช้งานเหตุการณ์สำหรับ ผู้ใช้
- i** ปิดการพร้อมต์แบบโต้ตอบระหว่างการถ่ายโอนไฟล์หลายไฟล์ ดูที่คำสั่งย่อย `prompt`, `mget`, `mput` และ `mdelete` สำหรับรายละเอียดของการพร้อมต์ระหว่างการถ่ายโอนไฟล์หลายไฟล์
- K** ปิดใช้งานอ็อพชัน `SO_KEEPALIVE` ที่กำหนดในไฟล์ `sys/socket.h` บนทั้งควบคุมและการเชื่อมต่อข้อมูล
- k realm** อนุญาตให้ผู้ใช้ระบุขอบเขตของตำแหน่งรีโมต หากแตกต่างจากขอบเขตของระบบโลคัล สำหรับวัตถุประสงค์เหล่านี้ ขอบเขตมีความหมายเหมือนกับเซลล์ DCE แฟล็กนี้จะถูกละเว้นหาก Kerberos 5 ไม่ใช้เมธอดของการพิสูจน์ตัวตนในปัจจุบัน
- M** ป้องกันมิให้คำสั่ง `ftp` ถูกบล็อก หลังจากไฟล์ถูกถ่ายโอนระหว่างโฮสต์โลคัลและรีโมต
- n** ป้องกันการล็อกอินอัตโนมัติในการเชื่อมต่อเริ่มต้น มิฉะนั้น คำสั่ง `ftp` จะคนหารายการ `$HOME/.netrc` ที่อธิบายกระบวนการล็อกอินและการกำหนดค่าเริ่มต้นสำหรับรีโมต โฮสต์ โปรดดูคำสั่งย่อย `user`
- q** อนุญาตให้ผู้ใช้ระบุวูรทีนย่อย `send_file` ต้องถูกใช้สำหรับการส่ง ไฟล์บนเน็ตเวิร์ก แฟล็กนี้ใช้ได้เฉพาะ เมื่อไฟล์กำลังถูกส่งออกในโหมดไบนารี โดยไม่มีการป้องกัน
- v** แสดงการตอบสนองทั้งหมดจากรีโมตเซิร์ฟเวอร์และจัดเตรียม สถิติการถ่ายโอนข้อมูล โหมดแสดงผลนี้เป็นดีฟอลต์เมื่อ เอาต์พุตของคำสั่ง `ftp` ส่งไปที่เทอร์มินัล เช่นคอนโซลหรือจอแสดงผล
- ถ้า `stdim` ไม่ใช่เทอร์มินัล คำสั่ง `ftp` จะปิดใช้งานโหมด `verbose` นอกจากผู้ใช้เรียกคำสั่ง `ftp` ด้วยแฟล็ก `-v` หรือเรียกคำสั่งย่อย `verbose`
- s** เริ่มต้นเซสชัน TLS ด้วยเซิร์ฟเวอร์ โดยส่งคำสั่ง `AUTH TLS` และคำสั่ง `PROTP` ไปยัง `ftpd` daemon ถ้าเซสชัน TLS ถูกสร้างขึ้น และคุณได้รับการพิสูจน์ตัวตน โดยใช้เมธอด Standard Authentication การถ่ายโอนข้อมูล และคำสั่งจะถูกเข้ารหัส

## คำสั่งย่อ

คำสั่งย่อ ftp ต่อไปนี้สามารถป้อนที่พร้อมท์ ftp> ใช้อัญประกาศ (" ") เพื่อระบุ พารามิเตอร์ที่มีอักขระช่องว่าง

### ไอเท็ม

![Command [Parameters]]

### คำอธิบาย

เรียกเซลล์แบบโต้ตอบบนโลคัลโฮสต์ คำสั่งเป็นทางเลือก โดยมีพารามิเตอร์ทางเลือกหนึ่งพารามิเตอร์หรือมากกว่านั้น สามารถถูกกำหนด กับคำสั่งเซลล์

\$Macro [Parameters]

เรียกใช้งานแมโครที่ระบุ ที่กำหนดก่อนหน้านี้กับคำสั่งย่อ macdef พารามิเตอร์ไม่ถูกขยาย

?[Subcommand]

แสดงขอความช่วยเหลือที่อธิบายคำสั่งย่อ ถ้าคุณ ไม่ระบุพารามิเตอร์ Subcommand คำสั่ง ftp จะแสดงรายการของคำสั่งย่อที่รู้จัก

account [Password]

ส่งรหัสผ่านเพิ่มเติมที่รีโมตโฮสต์อาจต้องการ การให้สิทธิ์การเข้าถึงรีซอร์สของโฮสต์ ถ้าไม่มีการระบุรหัสผ่านกับคำสั่ง ผู้ใช้จะได้รับพร้อมท์ขอรหัสผ่าน รหัสผ่าน ไม่ถูกแสดงบนจอภาพ

append LocalFile [RemoteFile]

ผนวกไฟล์กับไฟล์บนรีโมตโฮสต์ ถ้าชื่อไฟล์รีโมต ไม่ถูกระบุ ชื่อไฟล์โลคัลจะถูกใช้ โดยเปลี่ยนตาม การตั้งค่าที่กำกับคำสั่งย่อ ntrans หรือคำสั่งย่อ nmap คำสั่งย่อ append ใช้ค่าปัจจุบัน สำหรับคำสั่งย่อ form, mode, struct และ type ขณะผนวกไฟล์

ascii

คำพ้องสำหรับคำสั่งย่อ type ascii

bell

ส่งเสียงกระดิ่งหลังจากการถ่ายโอนไฟล์แต่ละไฟล์สมบูรณ์

binary

คำพ้องสำหรับคำสั่งย่อ type binary

block

คำพ้องสำหรับคำสั่งย่อ mode block

bye

สิ้นสุดเซสชันการถ่ายโอนไฟล์และออกจากคำสั่ง ftp เหมือนกับคำสั่งย่อ quit

carriage-control

คำพ้องสำหรับคำสั่งย่อ form carriage-control

case

เซตการสลับสำหรับตัวพิมพ์ของชื่อไฟล์ เมื่อคำสั่งย่อ case เป็น On คำสั่ง ftp จะเปลี่ยนชื่อไฟล์รีโมตที่แสดงในตัวอักษรตัวพิมพ์ใหญ่ทั้งหมดจากตัวพิมพ์ใหญ่ เป็นตัวพิมพ์เล็ก เมื่อเขียนข้อมูลไปที่ไดเรกทอรีโลคัล ดีฟอลต์คือ Off (ดังนั้นคำสั่ง ftp จะเขียนชื่อไฟล์รีโมตตัวพิมพ์ใหญ่ เป็นตัวพิมพ์ใหญ่ในโลคัลไดเรกทอรี)

cd RemoteDirectory

เปลี่ยนไดเรกทอรีทำงานบนรีโมตโฮสต์เป็นไดเรกทอรี ที่ระบุ

cdup

เปลี่ยนไดเรกทอรีทำงานบนรีโมตโฮสต์เป็นพาเรนทของ ไดเรกทอรีปัจจุบัน

close

จบเซสชันการถ่ายโอนไฟล์ แต่ไม่ออกจากคำสั่ง ftp แมโครที่กำหนดถูกลบ เหมือนกับคำสั่งย่อ disconnect

copylocal

สลับสำเนาโลคัล copylocal ดีฟอลต์ เป็น off ftp ทำการตรวจสอบว่าคุณไม่ได้ปล่อยวางไฟล์ โดยการ ftp มาที่ตัวเอง (ตัวอย่าง ชื่อโฮสต์เดียวกัน ชื่อพาธเดียวกัน) การเปิด copylocal ON ข้ามการตรวจสอบนี้

cr

เอาอักขระปิดแครอออกจากลำดับการปิดแคร และการป้อนบรรทัด เมื่อได้รับเรีกคอร์ดระหว่างการถ่ายโอนไฟล์ ชนิด ASCII (คำสั่ง ftp จบท้ายแต่ละเรีกคอร์ด ชนิด ASCII ด้วยการปิดแครและการป้อนบรรทัด ระหว่าง การถ่ายโอนไฟล์)

เรีกคอร์ดบนรีโมตโฮสต์ที่มีระบบปฏิบัติการอื่น ที่ไม่ใช่ที่คุณกำลังรันอยู่ อาจมีการป้อนบรรทัดเดี่ยวที่ฝังในตัวในเรีกคอร์ด เมื่อต้องการแยกการป้อนบรรทัดฝังตัวเหล่านี้จากตัวคั่นเรีกคอร์ด, ให้เซตคำสั่งย่อ cr เป็น Off คำสั่งย่อ cr สลับระหว่าง On และ Off

debug [O|1]

สลับการเก็บดีบักเรีกคอร์ด On และ Off ระบุ debug หรือ debug 1 เพื่อพิมพ์ แต่ละคำสั่งที่ส่งไปที่รีโมตโฮสต์และบันทึกไฟล์ควบคุม การรีสตาร์ท ระบุ debug อีกครั้ง หรือ debug 0, เพื่อหยุดการเก็บดีบักเรีกคอร์ด ลำดับคีย์ Ctrl-C ยังบันทึกไฟล์ควบคุมการรีสตาร์ทด้วย

การระบุคำสั่งย่อ debug ส่งข้อมูลการดีบักเกี่ยวกับการดำเนินการคำสั่ง ftp ไปที่ syslogd daemon ถ้าคุณระบุคำสั่งย่อ debug คุณต้องแก้ไขไฟล์ /etc/syslog.conf และเพิ่มหนึ่งในรายการ ต่อไปนี้:

user.info FileName

หรือ

user.debug FileName

หมายเหตุ: ระดับการดีบัก syslogd daemon ประกอบด้วยข้อความระดับข้อมูล

ถ้าคุณไม่แก้ไขไฟล์ /etc/syslog.conf จะไม่มี การสร้างข้อความขึ้น หลังจากการเปลี่ยนไฟล์ /etc/syslog.conf รันคำสั่ง refresh -s syslogd หรือ kill -1 SyslogdPID เพื่อ แจ้ง syslogd daemon ถึงการเปลี่ยนแปลง กับไฟล์คอนฟิกูเรชัน สำหรับข้อมูลเพิ่มเติมเกี่ยวกับระดับดีบัก อ้างอิง ที่ไฟล์ /etc/syslog.conf และอ้างอิงถึง แฟล็ก ftp -d ลบรีโมตไฟล์ที่ระบุ

delete RemoteFile

เขียนรายการของเนื้อหาของไดเรกทอรีรีโมตที่ระบุ (RemoteDirectory) ไปที่โลคัลไฟล์ ที่ระบุ (LocalFile) ถ้าพารามิเตอร์ the RemoteDirectory ไม่ได้ระบุไว้ คำสั่งย่อ dir แสดง เนื้อหาของรีโมตไดเรกทอรีปัจจุบัน ถ้าพารามิเตอร์ LocalFile ไม่ถูกระบุหรือเป็น - (เครื่องหมายขีดคั่น) คำสั่งย่อ dir แสดงรายการของโลคัลเทอร์มินัล

dir [RemoteDirectory][LocalFile]

## ไอเท็ม

**disconnect**

**ebcdic**

**exp\_cmd**

**file**

**form [ carriage-control | non-print |**

**telnet ]**

## คำอธิบาย

จบเซสชันการถ่ายโอนไฟล์ แต่ไม่ออกจากคำสั่ง ftp แมโครที่กำหนดถูกลบ เหมือนกับคำสั่งย่อย close

คำพ้องสำหรับคำสั่งย่อย type ebcdic

สลับระหว่าง คำสั่งโปรโตคอล ตามปกติและเชิงการทดลอง คำดีฟอลต์คือปิด

คำพ้องสำหรับคำสั่งย่อย struct file

ระบุฟอร์มของการถ่ายโอนไฟล์ คำสั่งย่อย form ดัดแปลงคำสั่งย่อย type ให้ส่งการถ่ายโอนไฟล์ในฟอร์มที่กำหนด อารกิวเมนต์ที่ใช้ได้คือ carriage-control, non-print และ telnet

## carriage-control

เซตฟอร์มของการถ่ายโอนไฟล์เป็น carriage-control

## non-print

เซตฟอร์มของการถ่ายโอนไฟล์เป็น non-print

**telnet** เซตฟอร์มของการถ่ายโอนไฟล์เป็น Telnet Telnet คือโปรโตคอล Transmission Control Protocol/Internet Protocol (TCP/IP) ที่เปิดการเชื่อมต่อกับระบบ

**get RemoteFile [LocalFile]**

คัดลอกรีโมตไฟล์ไปที่โลคัลโฮสต์ ถ้าพารามิเตอร์ LocalFile ไม่ถูกระบุ ชื่อไฟล์รีโมตถูกใช้แบบโลคัล และถูกเปลี่ยนแปลงโดยการตั้งค่าที่ทำโดยคำสั่งย่อย case, ntrans และ nmap คำสั่ง ftp ใช้การตั้งค่า ปัจจุบันสำหรับคำสั่งย่อย type, form, mode และ struct ขณะทำการถ่ายโอนไฟล์

**glob**

สลับส่วนขยาย file-name (globbing) สำหรับคำสั่งย่อย mdelete, mget และ mput ถ้า globbing ถูกปิด พารามิเตอร์ file-name สำหรับคำสั่งย่อยเหล่านี้จะไม่ถูกขยาย เมื่อมีการเปิดใช้ globbing และอักขระที่มีการจับคู่รูปแบบถูกใช้ในคำสั่งย่อย ที่ต้องการชื่อไฟล์เดียว ผลลัพธ์อาจต่างไปจากที่ต้องการ

ตัวอย่างคำสั่งย่อย append และ put ดำเนินการกับส่วนขยาย file-name แล้วใช้เฉพาะชื่อไฟล์แรก ที่สร้าง คำสั่งย่อย ftp อื่น เช่น cd, delete, get, mkdir, rename และ rmdir ไม่ดำเนินการ ส่วนขยาย file-name และรับอักขระที่มีการจับคู่ตามรูปแบบ ตามตัวอักษร

Globbering สำหรับคำสั่งย่อย mput ดำเนินการแบบโลคัลเหมือนกับ คำสั่ง csh สำหรับคำสั่งย่อย mdelete และ mget แต่ละชื่อไฟล์ถูกขยายแยกกันที่เครื่องรีโมตและ รายการไม่ถูกรวมกัน ส่วนขยายของชื่อใดเรียกหรือสามารถต่างจาก ส่วนขยายของชื่อไฟล์ ขึ้นกับรีโมตโฮสต์และ เซิร์ฟเวอร์ ftp

เมื่อต้องการ แสดงตัวอย่างส่วนขยายของชื่อใดเรียกหรือ ให้ใช้คำสั่งย่อย mls :

mls RemoteFile

เมื่อต้องการถ่ายโอนไดเรกทอรีซึบหรือของไฟล์ทั้งหมด ให้ถ่ายโอนไฟล์เก็บถาวร tar ของซึบหรือในไบนารีฟอร์ม

แทนการใช้คำสั่งย่อย mget หรือ mput

สลับการพิมพ์เครื่องหมาย แฮช (#) เมื่อคำสั่งย่อย hash เปิด คำสั่ง ftp แสดงหนึ่งเครื่องหมายแฮช สำหรับแต่ละ

**hash**

บล็อกข้อมูล (1024 ไบต์) ที่ถ่ายโอน

แสดงข้อมูลความช่วยเหลือ ดูที่ ? ก่อน

**help [Subcommand]**

**image**

คำพ้องสำหรับคำสั่งย่อย type image

**lcd [Directory]**

เปลี่ยนไดเรกทอรีทำงานบนโลคัลโฮสต์ ถ้าคุณไม่ได้ ระบุไดเรกทอรี คำสั่ง ftp จะใช้ ไดเรกทอรีโฮมของคุณ

**local M**

คำพ้องสำหรับคำสั่งย่อย type local M

**ls [RemoteDirectory] [LocalFile]**

เขียนรายการไฟล์ที่เป็นตัวอักษรรีโมตไดเรกทอรีไปที่โลคัลไฟล์ ถ้าพารามิเตอร์ RemoteDirectory ไม่ถูกระบุ คำสั่ง ftp แสดง ไดเรกทอรีรีโมตปัจจุบัน ถ้าพารามิเตอร์ LocalFile ไม่ถูกระบุหรือเป็น - (เครื่องหมายขีดค้น) คำสั่ง ftp แสดงรายการของโลคัลเทอร์มินัล

**macdef Macro**

กำหนดแมโครคำสั่งย่อย บรรทัดต่อมาถึงบรรทัดว่าง (มีการบั่นบรรทัดต่อกันสองครั้ง) ถูกบันทึกเป็นข้อความของแมโคร ไม่มากถึง 16 แมโคร มีอักขระได้มากที่สุด 4096 อักขระสำหรับแมโคร ที่สามารถกำหนดได้ แมโครยังคงกำหนดไว้ จนกว่ามีการกำหนดใหม่หรือคำสั่งย่อย close ถูกเรียกใช้งาน

\$ (เครื่องหมายดอลลาร์) และ \ (backslash) เป็นอักขระพิเศษในแมโคร ftp สัญลักษณ์ \$ ที่ตามด้วยตัวเลขถูกแทนที่ด้วยพารามิเตอร์แมโครที่ตรงกัน บนบรรทัดที่มีการเรียกใช้ (ดูที่คำสั่งย่อย \$) สัญลักษณ์ \$ ที่ตามด้วยตัวอักษร i แสดงว่าแมโครมีการวนซ้ำ อักขระที่รวมกัน \$i ถูกแทนที่ด้วยพารามิเตอร์ตามลำดับ ในการส่งผ่านแต่ละครั้ง

พารามิเตอร์แมโครแรกถูกใช้ในการส่งครั้งแรก พารามิเตอร์ที่สองถูกใช้ในการส่งครั้งที่สอง และต่อไป สัญลักษณ์ \ ป้องกันการปฏิบัติเป็นพิเศษ ของอักขระถัดไป ใช้สัญลักษณ์ \ เพื่อปิดความหมายพิเศษของสัญลักษณ์ \$ และ \ (จุดแบคสแลช)

**mdelete RemoteFiles**

ขยายไฟล์ที่ระบุโดยพารามิเตอร์ RemoteFiles ที่รีโมตโฮสต์และลบรีโมตไฟล์

## ไอเท็ม

**mkdir** [*RemoteDirectories LocalFile*]

### คำอธิบาย

ขยายไดเรกทอรีที่ระบุโดยพารามิเตอร์ *RemoteDirectories* ที่รีโมตโฮสต์และเขียนรายการของเนื้อหาของไดเรกทอรีเหล่านี้ไปที่ไฟล์ที่ระบุในพารามิเตอร์ *LocalFile* ถ้าพารามิเตอร์ *RemoteDirectories* มีอีกชระ pattern-matching คำสั่งย่อย **mkdir** จะพร้อมต่อขอโลคัลไฟล์ ถ้าไม่มีการระบุ ถ้าพารามิเตอร์ *RemoteDirectories* คือรายการของรีโมตไดเรกทอรีที่แยกด้วยช่องว่าง อาร์กิวเมนต์สุดท้าย ในรายการต้องเป็นชื่อไฟล์โลคัลหรือ - (เครื่องหมายขีดค้น)

**mget** *RemoteFiles*

ถ้าพารามิเตอร์ *LocalFile* เป็น - (เครื่องหมายขีดค้น) คำสั่งย่อย **mkdir** แสดง รายการบนโลคัลเทอร์มินัล ถ้าการพร้อมต่อแบบโต้ตอบเปิดอยู่ (ดูที่คำสั่งย่อย **prompt**) คำสั่ง **ftp** จะพร้อมต่อผู้ใช้เพื่อตรวจสอบว่า พารามิเตอร์สุดท้ายคือโลคัลไฟล์ไม่ใช่รีโมตไดเรกทอรี

**mkdir** [*RemoteDirectory*]

**mls** [*RemoteDirectories LocalFile*]

ขยายพารามิเตอร์ *RemoteFiles* ที่รีโมตโฮสต์และคัดลอกรีโมตไฟล์ที่กำหนดไปที่ไดเรกทอรีปัจจุบัน บนโลคัลโฮสต์ ดูที่คำสั่งย่อย **glob** สำหรับข้อมูลเพิ่มเติมเกี่ยวกับส่วนขยายชื่อไฟล์ ชื่อรีโมตไฟล์ถูกใช้แบบโลคัลและถูกเปลี่ยนแปลง โดยการตั้งค่าที่ทำโดยคำสั่งย่อย **case**, **ntrans** และ **nmap** คำสั่ง **ftp** ใช้การตั้งค่าปัจจุบันสำหรับคำสั่งย่อย **form**, **mode**, **struct** และ **type** ขณะทำการถ่ายโอนไฟล์

สร้างไดเรกทอรีที่ระบุในพารามิเตอร์ *RemoteDirectory* บนรีโมตโฮสต์

ขยายไดเรกทอรีที่ระบุในพารามิเตอร์ *RemoteDirectories* ที่รีโมตโฮสต์และเขียนรายการไฟล์แบบย่อของไดเรกทอรีรีโมตที่ระบุให้กับโลคัลไฟล์ ถ้าพารามิเตอร์ *RemoteDirectories* มีอีกชระ pattern-matching คำสั่งย่อย **mls** จะพร้อมต่อขอโลคัลไฟล์ ถ้าไม่มีการระบุ ถ้าพารามิเตอร์ *RemoteDirectories* คือรายการของรีโมตไดเรกทอรีที่แยกด้วยช่องว่าง อาร์กิวเมนต์สุดท้าย ในรายการต้องเป็นชื่อไฟล์โลคัลหรือ - (เครื่องหมายขีดค้น)

**mode** [*stream | block*]

ถ้าพารามิเตอร์ *LocalFile* เป็น - (เครื่องหมายขีดค้น) คำสั่งย่อย **mls** แสดง รายการบนโลคัลเทอร์มินัล ถ้าการพร้อมต่อแบบโต้ตอบเปิดอยู่ (ดูที่คำสั่งย่อย **prompt**) คำสั่ง **ftp** จะพร้อมต่อผู้ใช้เพื่อตรวจสอบว่า พารามิเตอร์สุดท้ายคือโลคัลไฟล์ไม่ใช่รีโมตไดเรกทอรี

เซตโหมดการถ่ายโอนไฟล์ ถ้าไม่มีการระบุอาร์กิวเมนต์ดีฟอลต์คือ **stream**

**block** เซตโหมดการถ่ายโอนไฟล์เป็นบล็อก

**stream** เซตโหมดการถ่ายโอนไฟล์เป็น stream

## ไอเท็ม

**modtime**

### คำอธิบาย

แสดงเวลาการแก้ไขล่าสุดของไฟล์ที่ระบุบน เครื่องรีโมต ถ้าคำสั่ง **ftp** ไม่ถูก เชื่อมต่อกับโฮสต์ก่อนการเรียกใช้งาน คำสั่งย่อย **modtime** จบการทำงานโดยมีข้อความแสดงความผิดพลาด คำสั่ง **ftp** ละเว้น พารามิเตอร์หลังจากพารามิเตอร์แรก ถ้าพารามิเตอร์ *FileName* ไม่ถูกระบุคำสั่ง **ftp** จะพร้อมต่อขอชื่อไฟล์ ถ้าไม่มีการกำหนดชื่อไฟล์ คำสั่ง **ftp** ส่งขอความการใช้งานไปที่เอาต์พุตมาตรฐานและจบคำสั่งย่อย

**mput** [*LocalFiles*]

ถ้าชื่อที่ระบุโดยพารามิเตอร์ *FileName* มีอยู่บนรีโมตโฮสต์ และชื่อระบุไฟล์ จากนั้นคำสั่ง **ftp** จะส่งข้อความที่มีเวลาการแก้ไขล่าสุด ของไฟล์ไปที่เอาต์พุตมาตรฐานและจบการทำงานของ คำสั่งย่อย ถ้า *FileName* ระบุไดเรกทอรี คำสั่ง **ftp** ส่งขอความแสดงความผิดพลาดไปที่ เอาต์พุตมาตรฐานและจบคำสั่งย่อย

หมายเหตุ: คำสั่งย่อย **modtime** แปลความหมายอีกชระเมตาเมื่อได้รับอนุญาต

ขยายไฟล์ที่ระบุในพารามิเตอร์ *LocalFiles* ที่โลคัลโฮสต์และคัดลอกไฟล์โลคัลที่ระบุไปที่รีโมต โฮสต์ ดูที่คำสั่งย่อย **glob** สำหรับข้อมูลเพิ่มเติมเกี่ยวกับส่วนขยายชื่อไฟล์ ชื่อ รีโมตไฟล์ถูกใช้แบบโลคัลและถูกเปลี่ยนแปลงโดยการตั้งค่าที่ทำโดยคำสั่งย่อย **ntrans** และ **nmap** คำสั่ง **ftp** ใช้การตั้งค่าปัจจุบันสำหรับคำสั่งย่อย **type**, **form**, **mode** และ **struct** ขณะทำการถ่ายโอนไฟล์

**nlist** [*RemoteDirectory*][*LocalFile*]

เขียนรายการของเนื้อหาของไดเรกทอรีรีโมตที่ระบุ (*RemoteDirectory*) ไปที่โลคัลไฟล์ ที่ระบุ (*LocalFile*) ถ้าพารามิเตอร์ *RemoteDirectory* ไม่ได้ระบุไว้ คำสั่งย่อย **nlist** แสดง เนื้อหาของรีโมตไดเรกทอรีปัจจุบัน ถ้าพารามิเตอร์ *LocalFile* ไม่ถูกระบุหรือเป็น - (เครื่องหมายขีดค้น) คำสั่งย่อย **nlist** แสดงรายการของโลคัลเทอร์มินัล

## ไอเท็ม

**nmap** [*InPattern OutPattern*]

### คำอธิบาย

เปิดหรือปิดกลไกการแม็พชื่อไฟล์ ถ้าไม่มีการระบุ พารามิเตอร์ การแม็พชื่อไฟล์จะถูกปิด ถ้ามีการระบุพารามิเตอร์ ชื่อไฟล์ต้นฉบับจะถูกแม็พสำหรับคำสั่งย่อย **mget** และ **mput** และสำหรับคำสั่งย่อย **get** และ **put** เมื่อไม่มีการระบุชื่อไฟล์ปลายทาง คำสั่งย่อยนี้มีประโยชน์ เมื่อโฮสต์โลคัลและโฮสต์รีโมตใช้ระเบียบหรือแนวปฏิบัติ การตั้งชื่อไฟล์ต่างกัน การแม็พเป็นไปตามรูปแบบที่เซ็ทโดยพารามิเตอร์ *InPattern* และ *OutPattern*

พารามิเตอร์ *InPattern* ระบุเพิ่มเพลตสำหรับ ชื่อไฟล์ขาเข้า ซึ่งอาจถูกประมวลผลแล้วตาม การตั้งค่า **case** และ **ntrans** ตัวแปร เพิ่มเพลต \$1 ถึง \$9 สามารถถูกรวมไว้ในพารามิเตอร์ *InPattern* อีกซะทั้งหมดในพารามิเตอร์ *InPattern* ที่ไม่ใช่ \$ (เครื่องหมายดอลลาร์) และ \ (backslash, เครื่องหมายดอลลาร์), ถูกปฏิบัติตามตัวอักษร และถูกใช้เป็นตัวคั่นระหว่างตัวแปร *InPattern* ตัวอย่าง ถ้าพารามิเตอร์ *InPattern* คือ \$1.\$2 และชื่อไฟล์รีโมตคือ mydata.dat, ค่าของ \$1 คือ mydata และค่าของ \$2 คือ dat

พารามิเตอร์ *OutPattern* กำหนดชื่อไฟล์ผลลัพธ์ ตัวแปร \$1 ถึง \$9 ถูกแทนที่โดยค่าที่สืบทอด จากพารามิเตอร์ *InPattern* และตัวแปร \$0 ถูกแทนที่โดยชื่อไฟล์ต้นฉบับ นอกจากนี้ ลำดับ [*Sequence1,Sequence2*] ถูกแทนที่โดยค่าของ *Sequence1* ถ้า *Sequence1* ไม่ใช่ null; มิฉะนั้น ถูกแทนที่โดยค่าของ *Sequence2* ตัวอย่าง คำสั่งย่อย:

```
nmap $1.$2.$3 [$1,$2].[$2,file]
```

จะให้ผล myfile.data จาก myfile.data หรือ myfile.data.old, myfile.file จาก myfile, และ myfile.myfile จาก .myfile ใช้สัญลักษณ์ \ (backslash) เพื่อป้องกันความหมายพิเศษของ \$ (เครื่องหมายดอลลาร์), (วงเล็บเหลี่ยมซ้าย), (วงเล็บเหลี่ยมขวา) และ, (คอมมา) ในพารามิเตอร์ *OutPattern*

## non-print

**ntrans** [*InCharacters OutCharacters*]

### คำพ้องสำหรับคำสั่งย่อย form non-print

เปิดหรือปิดกลไกการเปลือกรหัสชื่อไฟล์ ถ้าไม่มีการระบุพารามิเตอร์ การเปลือกรหัสจะถูก ปิด ถ้ามีการระบุพารามิเตอร์ รหัสในชื่อไฟล์ต้นฉบับ ถูกแปลสำหรับคำสั่งย่อย **mget** และ **mput** และสำหรับคำสั่งย่อย **get** และ **put** เมื่อชื่อไฟล์ปลายทางไม่ถูกระบุ

คำสั่งย่อยนี้มีประโยชน์ เมื่อ โฮสต์โลคัลและโฮสต์รีโมตใช้ระเบียบหรือแนวปฏิบัติ การตั้งชื่อไฟล์ต่างกัน การเปลือกรหัสเป็นไปตาม รูปแบบที่เซ็ทโดยพารามิเตอร์ *InCharacters* และ *OutCharacters* รหัสในชื่อไฟล์ต้นฉบับ จับคู่อักขระในพารามิเตอร์ *InCharacters* ถูกแทนที่โดยอักขระที่ตรงกัน ในพารามิเตอร์ *OutCharacters*

ถ้าสตริงที่ระบุโดยพารามิเตอร์ *InCharacters* ยาวกว่าสตริงที่ระบุโดยพารามิเตอร์ *OutCharacters* อักขระในพารามิเตอร์ *InCharacters* ถูกลบ ถ้าไม่มีอักขระที่ตรงกันในพารามิเตอร์ *OutCharacters*

**open** *HostName* [*Port*]

สร้างการเชื่อมต่อไปที่เซิร์ฟเวอร์ FTP ที่โฮสต์ที่ระบุโดย พารามิเตอร์ *HostName* ถ้ามีการระบุ หมายเลขพอร์ตทางเลือก คำสั่ง **ftp** จะพยายาม เชื่อมต่อไปที่เซิร์ฟเวอร์ที่พอร์ตนั้น ถ้าคุณลักษณะการล็อกอินอัตโนมัติ ถูกเซ็ท (นั่นคือ ไม่มีการระบุแฟล็ก **-n** บนบรรทัดคำสั่ง) คำสั่ง **ftp** พยายาม ล็อกอินที่เซิร์ฟเวอร์ FTP

คุณ ต้องมีไฟล์ **\$HOME/.netrc** ที่มี ข้อมูลที่ถูกต้อง และสิทธิ์ที่ถูกต้อง ไฟล์ **.netrc** ต้องอยู่ในไดเรกทอรีโฮมของคุณ

**passive**

สลับโหมด passive สำหรับการถ่ายโอนไฟล์ เมื่อคำสั่งการถ่ายโอนไฟล์ (เช่น **get**, **mget**, **put** หรือ **mput**) ถูกร้องขอด้วย การเปิดโหมด passive เซิร์ฟเวอร์ ftp จะเปิด การเชื่อมต่อข้อมูลกลับไปทีโคลเอ็นต์ ในโหมด passive โคลเอ็นต์เปิด การเชื่อมต่อข้อมูลไปที่เซิร์ฟเวอร์ เมื่อทำการส่งหรือรับข้อมูล

**ส่วนตัว**

ตั้งการระดับการป้องกันเป็นไพรเวตต่อเมื่อเมธอด การพิสูจน์ตัวตนถูกตั้งค่า ที่ระดับนี้ บุรณภาพของข้อมูลและการรักษาความลับจะ ได้รับการป้องกัน

**t**

สลับการพร้อมต์แบบโต้ตอบ ถ้าการพร้อมต์แบบโต้ตอบถูก เปิด (ดีฟอลต์) คำสั่ง **ftp** พร้อมต์เพื่อทำการตรวจสอบ ก่อนการรับ การส่ง หรือการลบไฟล์หลายไฟล์ ระหว่างคำสั่งย่อย **mget**, **mput** และ **mdelete** มิฉะนั้น คำสั่ง **ftp**

**protect**

จะทำงานตาม ไฟล์ทั้งหมดที่ระบุ คำสั่งนี้ส่งกลับระดับปัจจุบันของ การป้องกัน

ไอเท็ม  
proxy [Subcommand]

### คำอธิบาย

เรียกใช้งานคำสั่ง ftp บนการเชื่อมต่อ ควบคุมรอง คำสั่งย่อยนี้อนุญาตให้คำสั่ง ftp เชื่อมต่อพร้อมกันไปที่สองรีโมต FTP เซิร์ฟเวอร์สำหรับการถ่ายโอน ไฟล์ระหว่างสองเซิร์ฟเวอร์ คำสั่งย่อย proxy แรก ควรเป็นคำสั่งย่อย open เพื่อสร้างการเชื่อมต่อควบคุมรอง บ่อนคำสั่งย่อย proxy ? เพื่อดูคำสั่งย่อย ftp อื่น ที่รันได้บนการเชื่อมต่อรอง

คำสั่งย่อย ต่อไปนี้ทำงานต่างไป เมื่อเริ่มโดยคำสั่งย่อย proxy :

- คำสั่งย่อย open ไม่กำหนดแม่โครใหม่ ระหว่างกระบวนการล็อกอิน
- คำสั่งย่อย close ไม่ลบ macro definitions ที่มีอยู่
- คำสั่งย่อย get และ mget ถ่ายโอนไฟล์จากโฮสต์บนการเชื่อมต่อหลักไปที่โฮสต์ บนการเชื่อมต่อรอง
- คำสั่งย่อย put, mput และ append ถ่ายโอนไฟล์ จากโฮสต์บนการเชื่อมต่อรองไปที่โฮสต์บน การเชื่อมต่อหลัก
- คำสั่งย่อย restart สามารถถูกจัดการโดยคำสั่ง proxy
- คำสั่งย่อย status แสดงข้อมูลที่ถูกต้อง

put LocalFile [RemoteFile]

การถ่ายโอนไฟล์ต้องการให้เซิร์ฟเวอร์ FTP บนการเชื่อมต่อรองต้องสนับสนุนคำสั่ง PASV (passive) เก็บไฟล์บนรีโมตโฮสต์ ถ้าคุณไม่ระบุ พารามิเตอร์ RemoteFile คำสั่ง ftp ใช้ชื่อไฟล์โลคัลเพื่อตั้งชื่อรีโมตไฟล์ และชื่อรีโมตไฟล์ ถูกเปลี่ยนโดยการตั้งค่าที่ทำได้โดยคำสั่งย่อย ntrans และ nmap คำสั่ง ftp ใช้การตั้งค่าปัจจุบัน สำหรับคำสั่งย่อย type, form, mode และ struct ขณะทำการถ่ายโอนไฟล์

pwd

แสดงชื่อของไดเรกทอรีปัจจุบันบนรีโมต โฮสต์

quit

ปิดการเชื่อมต่อและออกจากคำสั่ง ftp เหมือน กับคำสั่งย่อย bye

quote String

ส่งสตริงที่ระบุโดยพารามิเตอร์ String ตามตัวอักษรไปที่รีโมตโฮสต์ เรียกใช้งานคำสั่งย่อย remohelp หรือ quote help เพื่อแสดงรายการของค่าที่ใช้ได้ สำหรับพารามิเตอร์ String

record

หมายเหตุ: คำสั่ง "Quoting" ที่เกี่ยวกับการถ่ายโอนข้อมูล สามารถก่อให้เกิดผลลัพธ์ที่ไม่คาดคิด

recv RemoteFile [LocalFile]

คำพ้องสำหรับคำสั่งย่อย struct record

reinitialize

คัดลอกรีโมตไฟล์ไปที่โลคัลโฮสต์ เหมือนกับคำสั่งย่อย get

remohelp [Subcommand]

กำหนดค่าเริ่มต้นเซชัน FTP ใหม่โดยขจัด I/O ทั้งหมดและ อนุญาตให้การถ่ายโอนสมบูรณ์ รีเซ็ตฟิลด์ทั้งหมด เหมือนกับว่าผู้ใช้เพิ่งเริ่ม เซชัน FTP โดยไม่มีการล็อกอินไปที่รีโมตโฮสต์

rename FromName ToName

ร้องขอวิธีใช้จากรีโมต FTP เซิร์ฟเวอร์

reset

เปลี่ยนชื่อไฟล์บนรีโมตโฮสต์

restart get | put | append

เคลียร์คิวตอบกลับ คำสั่งย่อยนี้ resynchronizes การ วิเคราะห์ค่าของคำสั่ง รีสตาร์ทการถ่ายโอนไฟล์ที่จุดซึ่งจุดตรวจสอบสุดท้าย ถูกสร้างขึ้น เมื่อต้องการรันให้สมบูรณ์ คำสั่งย่อยต้องเหมือนกับ คำสั่งย่อยที่ยกเลิก รวมถึง โครงสร้าง ชนิด และฟอร์ม อาร์กิวเมนต์ ที่ใช้ได้คือ get, put, และ append

rmdir RemoteDirectory

ลบรีโมตไดเรกทอรีที่ระบุโดยพารามิเตอร์ RemoteDirectory ที่รีโมตโฮสต์

runique

(ReceiveUnique) สลับส่วนช่วยเหลือสำหรับการสร้างชื่อไฟล์ เฉพาะสำหรับไฟล์ปลายทางโลคัลระหว่างคำสั่งย่อย get และ mget ถ้าส่วนช่วยเหลือนี้ถูกปิด (ดีฟอลต์) คำสั่ง ftp เขียนทับโลคัลไฟล์ มิฉะนั้น ถ้าโลคัลไฟล์มีชื่อเหมือนกับ ที่ระบุสำหรับไฟล์ปลายทางโลคัล คำสั่ง ftp แก้ไข ชื่อที่ระบุของไฟล์ปลายทางโลคัลด้วย . 1 ถ้าโลคัลไฟล์มีชื่อใหม่อยู่แล้ว คำสั่ง ftp จะผนวก . 2 ต่อท้ายกับชื่อที่ระบุ ถ้าไฟล์โลคัลใช้ชื่อที่สองนี้อยู่แล้ว คำสั่ง ftp ดำเนินการเพิ่ม คำต่อท้ายจนกว่าจะพบชื่อไฟล์เฉพาะหรือถึง .99 ถ้าไม่พบชื่อไฟล์เฉพาะ ถ้าคำสั่ง ftp ไม่พบชื่อไฟล์เฉพาะ คำสั่ง ftp จะรายงานข้อผิดพลาดและการถ่ายโอนจะไม่เกิดขึ้น หมายเหตุว่า คำสั่งย่อย runique ไม่มีผลกับชื่อไฟล์โลคัล ที่สร้างจากคำสั่งเซลล์

safe

เซ็ระดับการปกป้องเป็น "safe" ที่ระดับ นี้ข้อมูลมีการปกป้อง integrity

send LocalFile [RemoteFile]

เก็บโลคัลไฟล์บนรีโมตโฮสต์ เหมือนกับคำสั่งย่อย put

sendport

สลับการใช้คำสั่ง FTP PORT โดยดีฟอลต์คำสั่ง ftp ใช้คำสั่ง PORT เมื่อทำการ เชื่อมต่อสำหรับแต่ละการถ่ายโอน ข้อมูล เมื่อการใช้คำสั่ง PORT ถูกปิดใช้งาน คำสั่ง ftp จะไม่ใช้คำสั่ง PORT สำหรับการถ่ายโอนข้อมูล คำสั่ง PORT มีประโยชน์เมื่อจัดการ กับเซิร์ฟเวอร์ FTP ที่ละเว้นคำสั่ง PORT ขณะมีการระบุ ว่าคำสั่งได้รับแล้ว อย่างไม่ถูกต้อง แสดงหรือเซ็ช่วงการหมดเวลาสูญเปล่า แสดงหรือเซ็ umask การสร้างไฟล์ หรือเปลี่ยนสิทธิของไฟล์ โดยใช้

site Args

คำสั่ง chmod ค่าที่เป็นไปได้สำหรับพารามิเตอร์ Args คือ umask และ chmod

size RemoteFile

แสดงขนาดเป็นไบต์ของรีโมตไฟล์ที่ระบุโดย พารามิเตอร์ RemoteFile

status

แสดงสถานะปัจจุบันของคำสั่ง ftp เช่นเดียวกับสถานะของคำสั่งย่อย

stream

คำพ้องสำหรับคำสั่งย่อย mode stream

struct [ file | record ]

เซ็ชนิดโครงสร้างการถ่ายโอนข้อมูล อาร์กิวเมนต์ที่ใช้ได้คือ file และ record

file เซ็ชนิดโครงสร้างการถ่ายโอนข้อมูลเป็นไฟล์

record เซ็ชนิดโครงสร้างการถ่ายโอนข้อมูลเป็นเร็กคอร์ด

ไอเท็ม  
sunique

**คำอธิบาย**

(Send/Store Unique) สลับส่วนช่วยเหลือสำหรับการสร้างชื่อไฟล์ เฉพาะสำหรับไฟล์ปลายทางรีโมตระหว่างคำสั่งย่อย **put** และ **mput** ถ้าส่วนช่วยเหลือนี้ถูกปิด (ดีฟอลต์) คำสั่ง **ftp** เขียนที่รีโมตไฟล์มีจะนั้น ถ้ารีโมตไฟล์มีชื่อเหมือนกับไฟล์ที่ระบุสำหรับไฟล์ปลายทางรีโมต รีโมต FTP เซิร์ฟเวอร์จะแก้ไขชื่อของไฟล์ปลายทางรีโมต หมายเหตุว่า รีโมตเซิร์ฟเวอร์ต้องสนับสนุนคำสั่ง **STOU**

system

แสดงชนิดของระบบปฏิบัติการที่รันบนเครื่อง รีโมต

telnet

คำพ้องสำหรับคำสั่งย่อย **form telnet**

tenex

คำพ้องสำหรับคำสั่งย่อย **type tenex**

ติดตาม

สลับการติดตามแฟ้มเกิด

type [ascii | binary | ebcdic | image | local M | tenex]

เซตชนิดการถ่ายโอนไฟล์ อาร์กิวเมนต์ที่ใช้ได้คือ **ascii**, **binary**, **ebcdic**, **image**, **local M**, และ **tenex** ถ้าไม่มีการระบุอาร์กิวเมนต์ ชนิดปัจจุบันจะถูกพิมพ์ ชนิดดีฟอลต์คือ **ascii**; ชนิด **binary** สามารถมีประสิทธิภาพมากกว่า **ascii** ได้

**ascii** เซตชนิดการถ่ายโอนไฟล์เป็นเน็ตเวิร์ก ASCII ชนิดนี้เป็น ดีฟอลต์ การถ่ายโอนไฟล์อาจมีประสิทธิภาพมากกว่าด้วยการถ่ายโอน แบบ **binary-image** ดูที่อาร์กิวเมนต์ **binary** สำหรับ ข้อมูลเพิ่มเติม

**binary** เซตชนิดการถ่ายโอนไฟล์เป็น **binary image** ชนิดนี้มีประสิทธิภาพมากกว่าการถ่ายโอน ASCII ได้

**ebcdic** เซตชนิดการถ่ายโอนไฟล์เป็น EBCDIC

**image** เซตชนิดการถ่ายโอนไฟล์เป็น **binary image** ชนิดนี้มีประสิทธิภาพมากกว่าการถ่ายโอน ASCII ได้

**local M** เซตชนิดการถ่ายโอนไฟล์เป็นโลคัล พารามิเตอร์ **M** กำหนดหมายเลขฐานสิบของบิตต่อเวิร์ดของเครื่อง พารามิเตอร์นี้ ไม่มีดีฟอลต์

**tenex** เซตชนิดการถ่ายโอนไฟล์เป็นชนิดที่ต้องการสำหรับเครื่อง TENEX

user User [Password] [Account]

ระบุผู้ใช้โลคัล (**User**) เป็นรีโมต FTP เซิร์ฟเวอร์ ถ้าพารามิเตอร์ **Password** หรือ **Account** ไม่ถูกระบุและรีโมต เซิร์ฟเวอร์ขอข้อมูลนี้ คำสั่ง **ftp** จะพร้อมตัว XOR ที่สผ่านหรือแอดเคดแบบโลคัล ถ้าพารามิเตอร์ **Account** จำเป็น คำสั่ง **ftp** จะส่งไปที่ รีโมตเซิร์ฟเวอร์หลังจากกระบวนการล็อกอินรีโมตสมบูรณ์

หมายเหตุ: ยกเว้นการล็อกอินอัตโนมัติ ถูกปิดใช้งานโดยการระบุแฟล็ก **-n** บนบรรทัดรับคำสั่ง คำสั่ง **ftp** ส่งพารามิเตอร์ **User**, **Password** และ **Account** สำหรับการเชื่อมต่อเริ่มต้น ไปยังเซิร์ฟเวอร์รีโมตโดยอัตโนมัติ และคุณจำเป็นต้องมีไฟล์ **.netrc** อยู่ในไดเรกทอรีโฮมของคุณ เพื่อเรียกการล็อกอินอัตโนมัติ

verbose

สลับโหมด **verbose** เมื่อโหมด **verbose** ถูกเปิด (ดีฟอลต์), คำสั่ง **ftp** แสดงการตอบสนองทั้งหมดจาก รีโมต FTP เซิร์ฟเวอร์ นอกจากนี้คำสั่ง **ftp** แสดงสถิติเกี่ยวกับการถ่ายโอนไฟล์ทั้งหมด เมื่อการถ่ายโอนสมบูรณ์

## ตัวอย่าง

1. เมื่อต้องการเรียกคำสั่ง **ftp** ให้ล็อกอินสู่ระบบ **canopus**, แสดง ข้อมูลวิธีใช้โลคัล, แสดงข้อมูลวิธีใช้รีโมต, แสดงสถานะ, สลับคำสั่งย่อย **bell**, **prompt**, **runique**, **trace** และ **verbose**, แล้วจบการทำงานให้ป้อน:

```
$ ftp canopus
Connected to canopus.austin.century.com.
220 canopus.austin.century.com FTP server (Version 4.1 Sat Nov 23 12:52:09 CST 1991) ready.
Name (canopus:eric): dee
331 Password required for dee.
Password:
230 User dee logged in.
ftp> help
คำสั่งอาจถูกย่อ คำสั่งได้แก่:
!      delete      mdelete      proxy        runique
$      debug        mdir          sendport     send
account dir           mget          put           size
append disconnect   mkdir         pwd           status
ascii  form          mls           quit          struct
bell   get           mode          quote         sunique
binary glob         modtime      recv          system
bye    hash          mput         remotehelp   tenex
case   help         nmap         rstatus      trace
```

```

cd      image      nlist      rhelp      type
cdup    lcd         ntrans     rename     user
close   ls            open       reset      verbose
cr      macdef      prompt     rmdir     ?
clear   private     protect    safe

```

```
ftp> remoteshelp
```

```
214-The following commands are recognized(* =>'s unimplemented).
```

```

USER  PORT  RETR  MSND*  ALLO  DELE  SITE*  XMKD  CDUP
PASS  PASV  STOR  MSOM*  REST*  CWD  STAT*  RMD  XCUP
ACCT*  TYPE  APPE  MSAM*  RNFR  XCWD  HELP  XRMD  STOU
REIN*  STRU  MLFL*  MRSQ*  RNTD  LIST  NOOP  PWD
QUIT  MODE  MAIL*  MRCP*  ABOR  NLST  MKD  XPWD
AUTH  ADAT  PROT  PBSZ  MIC  ENC  CCC

```

```
214 Direct comments to ftp-bugs@canopus.austin.century.com.
```

```
ftp> status
```

```
Connected to canopus.austin.century.com.
```

```
No proxy connection.
```

```
Mode: stream; Type: ascii; Form: non-print; Structure: file
```

```
Verbose: on; Bell: off; Prompting: on; Globbing: on
```

```
Store unique: off; Receive unique: off
```

```
Case: off; CR stripping: on
```

```
Ntrans: off
```

```
Nmap: off
```

```
Hash mark printing: off; Use of PORT cmds: on
```

```
ftp> bell
```

```
Bell mode on.
```

```
ftp> prompt
```

```
Interactive mode off.
```

```
ftp> runique
```

```
Receive unique on.
```

```
ftp> trace
```

```
Packet tracing on.
```

```
ftp> verbose
```

```
Verbose mode off.
```

```
ftp> quit
```

```
$
```

- เมื่อต้องการเรียกคำสั่ง ftp ให้ล็อกอินเข้าสู่ระบบ canopus พิมพ์ไดเรกทอรีทำงาน เปลี่ยนไดเรกทอรีทำงาน เช็ตชนิดการถ่ายโอนไฟล์ เป็น ASCII ส่งการถ่ายโอนไฟล์ไฟล์ไปที่รีโมตโฮสต์ เปลี่ยนไดเรกทอรีทำงานเป็นพาเรนท์ไดเรกทอรีแล้วจบการทำงานให้ป้อน:

```
$ ftp canopus
```

```
Connected to canopus.austin.century.com.
```

```
220 canopus.austin.century.com FTP server (Version 4.1 Sat Nov 23 12:52:09 CST 1991) ready.
```

```
Name (canopus:eric): dee
```

```
331 Password required for dee.
```

```
Password:
```

```
230 User dee logged in.
```

```
ftp> pwd
```

```
257 "/home/dee" is current directory.
```

```
ftp> cd desktop
```

```
250 CWD command successful.
```

```
ftp> type ascii
```

```
200 Type set to A.
```

```
ftp> send typescript
```

```
200 PORT command successful.
```

```

150 Opening data connection for typescript (128.114.4.99,1412).
226 Transfer complete.
ftp> cdup
250 CWD command successful.
ftp> bye
221 Goodbye.
$

```

3. เมื่อต้องการเรียกคำสั่ง **ftp** ด้วยการล็อกออนอัตโนมัติ (โดยใช้ไฟล์ **.netrc**), ให้เปิดเซสชันด้วยระบบ **canopus** ล็อกอิน เปลี่ยนไดเรกทอรีทำงานเป็นพาเรนท์ไดเรกทอรี **พิมพ์** ไดเรกทอรีทำงาน แสดงเนื้อหาของไดเรกทอรีปัจจุบัน ลบไฟล์ เขียนรายการของเนื้อหาของไดเรกทอรีปัจจุบัน ไปที่โลคัลไฟล์ ปิดเซสชัน แล้วจบการทำงานให้ป้อน:

```

$ ftp canopus
Connected to canopus.austin.century.com.
220 canopus.austin.century.com FTP server (Version 4.1 Sat Nov 23 12:52:09 CST 1991) ready.
331 Password required for dee.
230 User dee logged in.
ftp> cdup
250 CWD command successful.
ftp> pwd
257 "/home" is current directory.
ftp> dir
200 PORT command successful.
150 Opening data connection for /usr/bin/ls (128.114.4.99,1407)
(0 bytes).
total 104
drwxr-xr-x  2 system      32 Feb 23 17:55 bin
Drwxr-xr-x 26 rios       4000 May 30 17:18 bin1
drwxr-xr-x  2 system      32 Feb 23 17:55 books
drwxrwxrwx 18 rios       1152 Jun  5 13:41 dee
-r--r--r--  1 system     9452 May 17 12:21 filesystems
drwxr-xr-x  2 system      32 Feb 23 17:55 jim
drwxr-xr-x  5 system      80 Feb 23 17:55 krs
drwxrwxrwx  2 rios       16432 Feb 23 17:36 lost+found
-rwxr-xr-x  1 rios       3651 May 24 16:45 oldmail
drwxr-xr-x  2 system     256 Feb 23 17:55 pubserv
drwxrwxrwx  2 system     144 Feb 23 17:55 rein989
drwxr-xr-x  2 system     112 Feb 23 17:55 reinstall
226 Transfer complete.
ftp> delete oldmail
250 DELE command successful.
ftp> mdir /home/dee/bin binlist
output to local-file: binlist? y
200 PORT command successful.
150 Opening data connection for /usr/bin/ls (128.114.4.99,1408) (0 bytes).
226 Transfer complete.
ftp> close
221 Goodbye.
ftp> quit
$

```

## ไฟล์

ไอเท็ม  
/usr/samples/tcpip/netrc  
/etc/syslog.conf

คำอธิบาย  
มีไฟล์ .netrc ตัวอย่าง  
มีข้อมูลคอนฟิกูเรชันสำหรับ syslogd daemon.

สิ่งอ้างอิงที่เกี่ยวข้อง:

“ftpd Daemon”

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง stty

คำสั่ง telnet

คำสั่ง tftp

คำสั่ง syslogd

การโอนย้ายไฟล์โดยใช้คำสั่ง ftp และ rcp

---

## ftpd Daemon

### วัตถุประสงค์

จัดเตรียมเซิร์ฟเวอร์ฟังก์ชันสำหรับอินเทอร์เน็ต FTP โพรโตคอล

### ไวยากรณ์

หมายเหตุ: โดยปกติ ftpd daemon ถูก เริ่มทำงานโดย inetd daemon คำสั่งยังสามารถควบคุมได้จากบรรทัดรับคำสั่ง โดยใช้ คำสั่ง SRC

```
/usr/sbin/ftpd [-d] [-D DataConnTimeOut] [-e] [-f] [-ff] [-k] [-l] [-U] [-t TimeOut] [-T MaxTimeOut] [-s] [-u OctalVal] [-q [-C]] [-c] [-H]
```

### คำอธิบาย

/usr/sbin/ftpd daemon คือกระบวนการเซิร์ฟเวอร์ DARPA Internet File Transfer Protocol (FTP) ftpd daemon ใช้ Transmission Control Protocol (TCP) เพื่อรอการติดต่อที่พอร์ตที่ระบุ ด้วยค่ากำหนดเซอร์วิสคำสั่ง ftp ในไฟล์ /etc/services

การเปลี่ยนแปลง กับ ftpd daemon สามารถทำได้โดยใช้ System Management Interface Tool (SMIT) หรือ System Resource Controller (SRC) โดยการแก้ไข ไฟล์ /etc/inetd.conf หรือ /etc/services ไม่แนะนำให้พิมพ์ ftpd ที่บรรทัดคำสั่ง ftpd daemon ถูกสตาร์ทโดยดีฟอลต์ เมื่อถูกเอามาเหตุผลออกจากในไฟล์ /etc/inetd.conf

inetd daemon รับข้อมูลจากไฟล์ /etc/inetd.conf และไฟล์ /etc/services

ถ้าคุณเปลี่ยนไฟล์ /etc/inetd.conf หรือ /etc/services ให้รัน คำสั่ง refresh -s inetd or kill -1 InetdPID เพื่อแจ้ง แก่ inetd daemon ถึงการเปลี่ยนแปลง กับไฟล์คอนฟิกูเรชัน

ftpd daemon ขยาย ชื่อไฟล์ตามระเบียบของคำสั่ง csh คำสั่งนี้อ่อนุญาตให้คุณใช้ metacharacters เช่น \* (เครื่องหมายดอกจัน), ? (เครื่องหมายคำถาม), [] (วงเล็บเหลี่ยมซ้ายและขวา), { } (เครื่องหมายปีกกาซ้ายและขวา) และ ~ (tilde)

## ftppaccess.ctl File

ไฟล์ `/etc/ftppaccess.ctl` ถูกค้นหาเพื่อหาบรรทัดที่เริ่มด้วย `allow:`, `deny:`, `readonly:`, `writeonly:`, `readwrite:`, `useronly:`, `grouponly:`, `herald:` และ/หรือ `motd:` บรรทัดอื่นจะถูกข้ามไป ถ้าไฟล์ไม่มีอยู่จากนั้นการเข้าถึง ftp ทำได้จากทุกโฮสต์ บรรทัด `allow:` และ `deny:` มีไว้สำหรับจำกัดการเข้าถึงโฮสต์ บรรทัด `readonly:`, `writeonly:` และ `readwrite:` มีไว้สำหรับจำกัด การอ่าน (get) และเขียน (put) ftp บรรทัด `useronly:` และ `grouponly:` มีไว้สำหรับการกำหนดผู้ใช้ anonymous บรรทัด `herald:` และ `motd:` มีไว้สำหรับข้อความหลายบรรทัด ก่อนและหลังการล็อกอิน

ไวยากรณ์สำหรับบรรทัดทั้งหมดใน `/etc/ftppaccess.ctl` อยู่ในฟอร์ม:

```
keyword: value, value, ...
```

ซึ่งคุณสามารถระบุ ค่าหนึ่งค่าหรือมากกว่านั้น สำหรับทุกคีย์เวิร์ด คุณมีบรรทัดหลายบรรทัดโดยมี คีย์เวิร์ดเหมือนกันได้ บรรทัดใน `/etc/ftppaccess.ctl` ถูก จำกัดที่ 1024 อักขระ อักขระที่เกิน 1024 อักขระจะ ถูกละเว้น

ไวยากรณ์สำหรับบรรทัด `allow:` และ `deny:` คือ:

```
allow: host, host, ...
```

```
deny: host, host, ...
```

ถ้าบรรทัด `allow:` ถูกระบุแล้ว เฉพาะโฮสต์ที่แสดงในบรรทัด `allow:` ทั้งหมด ได้รับอนุญาตให้เข้าถึง ftp โฮสต์อื่นทั้งหมดจะถูก ปฏิเสธไม่ให้เข้าถึง ftp ถ้าไม่มีบรรทัด `allow:` โฮสต์ทั้งหมดจะ ถูกกำหนดการเข้าถึง ftp ยกเว้นโฮสต์ที่ระบุในบรรทัด `deny:` โฮสต์สามารถถูกระบุเป็นชื่อโฮสต์หรือ IP แอดเดรส

ไวยากรณ์สำหรับบรรทัด `readonly:`, `writeonly:` and `readwrite:` คือ:

```
readonly: dirname, dirname, ...
```

```
writeonly: dirname, dirname, ...
```

```
readwrite: dirname, dirname, ...
```

บรรทัด `readonly:` แสดงไดเรกทอรีอ่านอย่างเดียวและบรรทัด `writeonly:` แสดงไดเรกทอรีเขียนอย่างเดียว การเข้าถึงเพื่ออ่าน ถูกปฏิเสธในไดเรกทอรีแบบเขียน อย่างเดียวและการเข้าถึงเพื่อเขียนถูกปฏิเสธในไดเรกทอรีแบบอ่านอย่างเดียว ไดเรกทอรี อื่นทั้งหมดได้รับการเข้าถึงยกเว้นบรรทัด `readwrite:` ถูกระบุ ถ้าบรรทัด `readwrite:` ถูกระบุ เฉพาะไดเรกทอรีที่แสดงใน บรรทัด `readwrite:` และ/หรือที่แสดงในบรรทัด `readonly:` ได้รับการเข้าถึง เพื่อการอ่าน และเฉพาะไดเรกทอรีที่แสดงใน บรรทัด `readwrite:` และ/หรือ ที่แสดงในบรรทัด `writeonly:` ที่ได้รับการเข้าถึง เพื่อการเขียน นอกจากนี้ บรรทัดเหล่านี้สามารถ มีค่า "ALL" หรือ "NONE"

ไวยากรณ์สำหรับบรรทัด `useronly:`, `puseronly:`, `grouponly:`, และ `pgrouponly:` คือ:

```
useronly: username, username, ...
```

```
puseronly: username, username, ...
```

```
grouponly: groupname, groupname, ...
```

```
pgrouponly: groupname, groupname, ...
```

ชื่อผู้เข้ามาจาก `/etc/passwd` และชื่อกลุ่มมาจาก `/etc/group` บรรทัด `useronly:` และ `puseronly:` กำหนดผู้ใช้ anonymous บรรทัด `grouponly:` และ `pgrouponly:` กำหนดกลุ่มของผู้ใช้ anonymous ผู้ใช้ anonymous เหล่านี้เหมือนกับ ผู้ใช้ anonymous ในกิจกรรม ftp นั้นที่ถูกจำกัดกับไดเรกทอรีโฮมของผู้ใช้ บรรทัด `useronly:` และ `grouponly:` กำหนดผู้ใช้ anonymous เหมือน กับผู้ใช้ anonymous ตรงที่ ไม่มีการป้องกันด้วยรหัสผ่าน บรรทัด `puseronly:` และ `pgrouponly:` กำหนดผู้ใช้ anonymous ที่มี การป้องกันด้วยรหัสผ่าน

หมายเหตุ: สำหรับผู้ใช้ `puseronly:` และ `pgrouponly:` รหัสผ่านต้องถูกสร้างและล็อกอินต้องถูกปิดใช้งาน

ไวยากรณ์สำหรับบรรทัด `herald:` และ `motd:` คือ:

```
herald: path
motd: on|off
```

พารามิเตอร์เพิ่มเติมของไฟล์ที่มี multiline herald ที่แสดงก่อนการล็อกอิน เมื่อบรรทัด `motd:` มีค่า 'on' ไฟล์ `$HOME/motd` จะมีข้อความหลายบรรทัด ที่แสดงหลังการล็อกอิน ถ้าผู้ใช้ถูกกำหนดเป็นผู้ใช้ anonymous ไฟล์ `/etc/motd` จะมีข้อความหลายบรรทัดที่แสดงหลังการล็อกอิน (หมายเหตุไว้ว่า `/etc/motd` อยู่ในไดเรกทอรีโฮมของ chroot ของผู้ใช้ anonymous) ดีฟอลต์สำหรับ บรรทัด `motd:` คือ off

ถ้าเมธอดการพิสูจน์ตัวตนระบบปฏิบัติการมาตรฐาน คือเมธอดการพิสูจน์ตัวตนปัจจุบัน:

ก่อนที่ `ftpd` daemon จะสามารถถ่ายโอนไฟล์สำหรับกระบวนการโคลเอ็นต์ ต้องมีการพิสูจน์ตัวตน กระบวนการโคลเอ็นต์ `ftpd` daemon พิสูจน์ตัวตน กระบวนการโคลเอ็นต์ตามกฎเหล่านี้:

- ผู้ใช้ต้องมีรหัสผ่านในฐานข้อมูลรหัสผ่าน `/etc/security/passwd` (ถ้ารหัสผ่านของผู้ใช้ไม่ใช่ null, กระบวนการโคลเอ็นต์ต้องจัดเตรียม รหัสผ่านนั้น)
- ชื่อผู้ใช้ต้องไม่อยู่ในไฟล์ `/etc/ftpusers`
- login shell ของผู้ใช้ต้องแสดงในเซลล์แอตทริบิวต์ของไฟล์ `/etc/security/login.cfg`
- ถ้าชื่อผู้ใช้คือ anonymous, ftp หรือถูก กำหนดผู้ใช้ anonymous ในไฟล์ `/etc/ftppaccess.ctl` แอคเคาต์ anonymous FTP ต้องถูกกำหนดในไฟล์รหัสผ่าน ใน กรณีนี้ กระบวนการโคลเอ็นต์ได้รับอนุญาตให้ล็อกอินโดยใช้รหัสผ่านตามต้องการ ตามระเบียบแล้ว รหัสผ่านคือชื่อของโคลเอ็นต์โฮสต์ `ftpd` daemon ใช้การวัดผลพิเศษเพื่อกำจัดการเข้าถึง โดยกระบวนการโคลเอ็นต์กับแอคเคาต์ anonymous

ถ้า Kerberos คือเมธอดการพิสูจน์ตัวตนปัจจุบัน:

`ftpd` daemon อนุญาตการเข้าถึงเฉพาะถ้า เจ็อนไซต่อไปนี้ทั้งหมดถูกระบุ:

- ผู้ใช้โลคัลของโคลเอ็นต์ ftp มี DCE credentials ปัจจุบัน
- ทั้งระบบโลคัลและรีโมตสนับสนุนคำสั่ง AUTH
- ระบบรีโมตยอมรับ DCE credentials สำหรับการเข้าถึงแอคเคาต์รีโมต ดูที่ฟังก์ชัน `kvalid_user` สำหรับข้อมูลเพิ่มเติม

### การสนับสนุน Transport Layer Security

`ftpd` daemon สนับสนุน Transport Layer Security (TLS) ตามที่กำหนดใน RFC 4217 TLS คือโปรโตคอลการเข้ารหัส ที่มีการสื่อสารที่ปลอดภัยระหว่างโคลเอ็นต์และเซิร์ฟเวอร์

จุดประสงค์หลักของการใช้คือเพื่อรักษาความปลอดภัย การควบคุม และการเชื่อมต่อข้อมูล โดยใช้การเข้ารหัส โคลเอ็นต์จำเป็นต้องถูกพิสูจน์ตัวตน โดยวิธีอื่น เมธอดที่สนับสนุนมีเพียงเมธอด Standard Authentication

เมื่อได้รับการร้องขอให้เริ่มทำงานเซสชัน TLS `ftpd` daemon ดำเนินการอ่านไฟล์ `/etc/ftpd.cnf` โหลด พารามิเตอร์คอนฟิกูเรชันต่อไปนี้จะใช้เพื่อตั้งค่า เซสชัน TLS:

ไอเท็ม CRL_PATH	คำอธิบาย พารามิเตอร์ CRL_PATH จัดเตรียม พาทไปที่ไฟล์รายการการร้องขอใบรับรอง ซึ่งต้องอยู่ในรูปแบบ PEM ถ้ามีการระบุ ใบรับรองดิจิทัลที่จัดเตรียมโดยไคลเอ็นต์ จะถูกตรวจสอบกับรายการการร้องขอใบรับรอง ถ้าไคลเอ็นต์ ftp ไม่ได้ใช้ใบรับรองดิจิทัล การเชื่อมต่อจะล้มเหลว ถ้าไคลเอ็นต์จัดเตรียมใบรับรองดิจิทัล แต่ใบรับรอง ถูกร้องขอ เซสชัน TLS จะล้มเหลว ถ้าพารามิเตอร์นี้ไม่ถูกระบุ ไคลเอ็นต์จะไม่ต้องจัดเตรียมใบรับรองดิจิทัล
CA_PATH	พารามิเตอร์ CA_PATH จัดเตรียม พาทไปที่ไฟล์สิทธิ์ของใบรับรอง ซึ่งต้องอยู่ในรูปแบบ PEM ถ้ามีการระบุ ใบรับรองไคลเอ็นต์จะถูกตรวจสอบกับ สิทธิ์ของใบรับรอง ถ้าไคลเอ็นต์ไม่ได้จัดเตรียมใบรับรองดิจิทัล การเชื่อมต่อจะล้มเหลว ถ้าไคลเอ็นต์จัดเตรียมใบรับรองดิจิทัล แต่ใบรับรองไม่ได้ถูกลงชื่อโดยผู้มีสิทธิ์ด้านความปลอดภัย เซสชัน TLS จะล้มเหลว ถ้าพารามิเตอร์นี้ไม่ถูกระบุ ไคลเอ็นต์จะไม่ต้องจัดเตรียมใบรับรองดิจิทัล
CIPHER_LIST	ถ้ามีการระบุพารามิเตอร์ CIPHER_LIST รายการถูกใช้ระหว่างเซสชัน TLS ถ้าไม่ใช้รายการการเข้ารหัสที่ปลอดภัย จะถูกนำมาใช้
DEPTH	ถ้าพารามิเตอร์คอนฟิกเรชัน CA_PATH ถูกระบุ ค่า DEPTH ถูกใช้เพื่อ ตรวจสอบใบรับรองที่จัดเตรียมโดยไคลเอ็นต์ ftp ในลำดับชั้น ใบรับรองดิจิทัล ถ้าไม่มีการจัดเตรียม ค่าดีฟอลต์คือ 9 จะถูกนำมาใช้
CERTIFICATE	พารามิเตอร์ CERTIFICATE จัดเตรียม พาทไปที่ไฟล์เชื่อมโยงใบรับรองดิจิทัลที่ใช้ได้ในรูปแบบ PEM ไฟล์นี้ ถูกใช้ในเซสชัน TLS พารามิเตอร์นี้ต่อถูกระบุให้เริ่มทำงานเซสชัน TLS
CERTIFICATE_PRIVATE_KEY	ถ้าพารามิเตอร์นี้ไม่ถูกระบุ เซิร์ฟเวอร์ ftpd จะปฏิเสธการร้องขอ TLS ทั้งหมด พารามิเตอร์ CERTIFICATE_PRIVATE_KEY มีพาทไปที่คีย์ไพรเวทใบรับรอง ซึ่งอยู่ในรูปแบบ PEM และถูกใช้ระหว่างเซสชัน TLS พารามิเตอร์นี้ต่อถูกระบุให้เริ่มทำงานเซสชัน TLS
DH_PARAMETERS_DIR	ถ้าพารามิเตอร์นี้ไม่ถูกระบุ เซิร์ฟเวอร์ ftpd จะปฏิเสธการร้องขอ TLS ทั้งหมด พารามิเตอร์ DH_PARAMETERS_DIR มี พาทไปที่ไดเรกทอรีที่มีพารามิเตอร์ Diffie Helman ในรูปแบบ PEM ไฟล์มากกว่าหนึ่งไฟล์ที่มีพารามิเตอร์ Diffie Helman ในรูปแบบ PEM สามารถถูกรวมไว้ในไดเรกทอรีนี้ ftpd daemon ค้นหาพารามิเตอร์ที่เหมาะสม สำหรับใช้งานถ้าจำเป็น

เมื่อต้องการสนับสนุน TLS คุณต้องติดตั้งเวอร์ชันล่าสุดของ เครื่องมือ OpenSSL จากเว็บไซต์ AIX Web Download Pack Programs

### คำแนะนำ File Transfer Protocol Subtree

เมื่อจัดการกับผู้ใช้ anonymous FTP เซิร์ฟเวอร์ ดำเนินคำสั่ง **chroot** ในไดเรกทอรีโฮมของแอดเคาต์ FTP สำหรับการรักษาความปลอดภัยที่สูงขึ้น ให้นำกฎต่อไปนี้มาใช้ เมื่อคุณสร้าง FTP subtree:

ไอเท็ม	คำอธิบาย
~ftp	สร้างไดเรกทอรีโฮมที่เป็นของ root และโหมด r-xr-xr-x (555)
~ftp/bin	สร้างไดเรกทอรีนี้เป็นของผู้ใช้ root และผู้อื่น ไม่สามารถเขียนข้อมูลได้ โปรแกรม ls ต้องอยู่ใน ไดเรกทอรีนี้เพื่อสนับสนุนคำสั่ง list โปรแกรมนี้ต้องมีโหมด 111
~ftp/etc	ทำไดเรกทอรีนี้เป็นของผู้ใช้ root และผู้อื่น ไม่สามารถเขียนข้อมูลได้
~ftp/pub	ทำไดเรกทอรีนี้เป็นโหมด 777 และเป็นของ FTP จากนั้นผู้ใช้ ต้องนำไฟล์ที่จะถูกเข้าถึง ผ่านแอดเคาต์ anonymous มาไว้ในไดเรกทอรีนี้

**หมายเหตุ:** เซลล์สคริปต์ `/usr/samples/tcpip/anon.ftp` ใช้ กฎทางด้านบนเพื่อเซตอัฟแอดเคาต์ anonymous FTP ให้กับคุณ

เมื่อจัดการกับผู้ใช้ FTP ที่กำหนดใน `/etc/ftppaccess.ctl`, เซิร์ฟเวอร์จะดำเนินการคำสั่ง **chroot** ใน ไดเรกทอรีโฮมของแอดเคาต์ สำหรับการรักษาความปลอดภัยที่สูงขึ้น ให้นำกฎต่อไปนี้มาใช้ เมื่อคุณสร้าง subtree ของผู้ใช้:

~user ทำไดเรกทอรีโฮมให้เป็นของ root และโหมด r-xr-xr-x (555)

~user/bin

ทำไดเรกทอรีให้เป็นของผู้ใช้ root และผู้อื่นไม่สามารถ เขียนข้อมูลได้ โปรแกรม ls ต้องอยู่ใน ไดเรกทอรีนี้เพื่อสนับสนุนคำสั่ง list โปรแกรมนี้ต้องมีโหมด 111

~user/etc

ทำไดเรกทอรีให้เป็นของผู้ใช้ root และผู้อื่นไม่สามารถ เขียนข้อมูลได้

~user/pub

ทำไดเรกทอรีนี้ให้เป็นโหมด 777 และเป็นของผู้ใช้ จากนั้นผู้ใช้ ต้องนำไฟล์ที่จะถูกเข้าถึง ผ่านแอดเคาต์ anonymous มาไว้ในไดเรกทอรีนี้

หมายเหตุ: เซลล์สคริปต์ /usr/samples/tcpip/anon.users.ftp ใช้ กฎทางด้านบนเพื่อเซตอัพแอดเคาต์ anonymous FTP ให้กับคุณ

เซิร์ฟเวอร์ต้องรันเป็นผู้ใช้ root เพื่อสร้าง ชื่อเกี่ยวกับหมายเลขพอร์ตที่มี privilege เซิร์ฟเวอร์ดูแล ID ผู้ใช้ที่ใช้งานได้ ของผู้ใช้ ที่ล็อกอิน แปลงกับเป็นผู้ใช้ root เฉพาะเมื่อ ทำการเชื่อมโยงแอดเดรสกับชื่อเกิด

### การร้องขอของ File Transfer Protocol ที่สนับสนุน

ftpd daemon ในขณะนี้ สนับสนุนการร้องขอ FTP ต่อไปนี้:

ไอเท็ม	คำอธิบาย
ABOR	จบคำสั่งก่อนหน้านี้
ACCT	ระบุแอดเคาต์ (ละเว)
ADAT	ระบุ Authentication/Security Data.
ALLO	จัดสรรหน่วยเก็บ (วาง)
APPE	ผนวกกับไฟล์
AUTH	ระบุกลไก การพิสูจน์ตัวตน/การรักษาความปลอดภัย
CCC	ระบุ Clear Command Channel
CDUP	เปลี่ยนพาเรนต์ไดเรกทอรีของไดเรกทอรีทำงานปัจจุบัน
CWD	เปลี่ยนไดเรกทอรีทำงาน
DELE	ลบไฟล์
ENC	ระบุ Privacy Protected Command
HELP	ให้ข้อมูลวิธีใช้
ไอเท็ม	คำอธิบาย
LIST	ให้ไฟล์รายการในไดเรกทอรี (การร้องขอ FTP นี้ เหมือนกับคำสั่ง ls -lA)
MKD	สร้างไดเรกทอรี
MDTM	แสดงเวลาการแก้ไขไฟล์ล่าสุด
MIC	ระบุ Integrity Protected Command
MODE	ระบุโหมดการถ่ายโอนข้อมูล
NLST	ให้รายชื่อไฟล์ในไดเรกทอรี (การร้องขอ FTP นี้ เหมือนกับคำสั่ง ls)
NOOP	ไม่ดำเนินการ
PASS	ระบุรหัสผ่าน
PASV	เตรียมการถ่ายโอน เซิร์ฟเวอร์ต่อเซิร์ฟเวอร์
PBSZ	ระบุ Protection Buffer Size
PORT	ระบุพอร์ตการเชื่อมต่อข้อมูล
PROT	ระบุ Data Channel Protection Level
PWD	พิมพ์ไดเรกทอรีทำงานปัจจุบัน
QUIT	จบเซสชัน
RETR	รับไฟล์
RMD	ลบไดเรกทอรี
RNFR	ระบุชื่อไฟล์ก่อนถูกเปลี่ยนชื่อ

ไอเท็ม	คำอธิบาย
RNTO	ระบุชื่อไฟล์ที่จะเปลี่ยน
SITE	คำสั่งต่อไปนี้ไม่ใช่คำสั่งมาตรฐานหรือ UNIX-specific ซึ่งมีการสนับสนุนโดยการร้องขอ SITE :
	UMASK เปลี่ยน umask (SITE UMASK 002)
	IDLE เซ็ตเวลาสูญเปล่า (SITE IDLE 60)
	CHMOD เปลี่ยนโหมดของไฟล์ (SITE CHMOD 755 FileName)
	HELP ให้ข้อมูลวิธีใช้ (SITE HELP)
SIZE	ส่งกลับขนาดของไฟล์ปัจจุบัน
STAT	ส่งกลับสถานะของเซิร์ฟเวอร์
STOR	เก็บไฟล์
STOU	เก็บไฟล์โดยใช้ชื่อไฟล์เฉพาะ
STRU	ระบุโครงสร้างของการถ่ายโอนข้อมูลเป็นโครงสร้างไฟล์
SYST	แสดงชนิดระบบปฏิบัติการของระบบเซิร์ฟเวอร์
TYPE	ระบุชนิดการถ่ายโอนข้อมูลด้วยพารามิเตอร์ Type
USER	ระบุชื่อผู้ใช้
XCUP	เปลี่ยนพาเรนทไดเรกทอรีของไดเรกทอรีทำงานปัจจุบัน (โดยปกติจะไม่ใช้)
XCWD	เปลี่ยนไดเรกทอรีปัจจุบัน (โดยปกติจะไม่ใช้)
XMKD	สร้างไดเรกทอรี (โดยปกติจะไม่ใช้)
XPWD	พิมพ์ไดเรกทอรีทำงานปัจจุบัน (โดยปกติจะไม่ใช้)
XRMD	ลบไดเรกทอรี (โดยปกติจะไม่ใช้)

การร้องขอ FTP ที่เหลือที่กำหนดใน Internet RFC 959 เป็นที่รู้จัก แต่ไม่มีการนำมาใช้ การร้องขอ MDTM และ SIZE ไม่ถูกระบุโดย RFC 959 แต่ ถูกกำหนดการไว้ให้มีใน FTP RFC ที่อัปเดตในครั้งถัดไป

ถ้าการร้องขอ STAT ได้รับ ระหว่างการถ่ายโอนข้อมูลและนำหน้ามาโดยสัญญาณ Telnet IP และสัญญาณ SYNCH สถานะการถ่ายโอนจะถูกส่งคืน

ftpd daemon ต้องถูก ควบคุมโดยใช้ System Management Interface Tool (SMIT) หรือ โดยการเปลี่ยนไฟล์ /etc/inetd.conf ไม่แนะนำให้พิมพ์ ftpd ที่ บรรทัดคำสั่ง

### การจัดการ ftpd Daemon ด้วย System Resource Controller

ftpd daemon เป็น subserver ของ inetd daemon, ซึ่งเป็นระบบย่อย ของ System Resource Controller (SRC) ftpd daemon เป็นสมาชิกของกลุ่มระบบย่อย tcpip SRC daemon นี้ถูกเปิดใช้โดยตีพอลต์ในไฟล์ /etc/inetd.conf และสามารถถูกดูแลโดย คำสั่ง SRC ต่อไปนี้:

ไอเท็ม	คำอธิบาย
startsrc	เริ่มต้นระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย
stopsrc	หยุดทำงานระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย
lssrc	รับข้อมูลสถานะของระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย

## แฟล็ก

## ไอเท็ม

- C คำอธิบาย  
อนุญาตให้ผู้ใช้นิวว่าไฟล์ขาออกที่ส่งโดยใช้คำสั่ง `send_file` ต้องถูกแคชใน Network Buffer Cache (NBC) แฟล็กนี้ไม่สามารถใช้ได้นอกจากมีการระบุแฟล็ก `-q` แฟล็กนี้ใช้ได้เฉพาะ เมื่อไฟล์กำลังถูกส่งออกในโหมดไบนารี โดยไม่มีการป้องกัน
- c ยับยั้งการค้นหาชื่อโฮสต์ย้อนกลับ
- d ส่งข้อมูลดีบั๊กเกี่ยวกับการดำเนินการ `ftpd daemon` ไปที่ `syslogd daemon` ถ้าคุณระบุแฟล็ก `-d` คุณต้องแก้ไขไฟล์ `/etc/syslog.conf` และเพิ่มรายการต่อไปนี้:  
  
daemon.debug FileName  
  
หมายเหตุ: `syslogd daemon` ที่ระดับ debug มีข้อความระดับ info  
  
ถ้าคุณไม่แก้ไขไฟล์ `/etc/syslog.conf` จะไม่มี การสร้างข้อความขึ้น หลังการเปลี่ยนแปลง ไฟล์ `/etc/syslog.conf` วันคำสั่ง `refresh -s syslogd` หรือคำสั่ง `kill -1 SyslogdPID` เพื่อแจ้ง `syslogd daemon` ทราบถึงการเปลี่ยนแปลงในไฟล์คอนฟิกูเรชัน สำหรับข้อมูลเพิ่มเติมเกี่ยวกับระดับดีบั๊ก อ้างอิง ที่ไฟล์ `/etc/syslog.conf`
- D DataConnTimeOut  
ระบุจำนวนสูงสุดของวินาทีที่ `ftpd daemon` คงการเชื่อมต่อข้อมูลไว้ ค่าดีฟอลต์คือ 300 วินาทีและค่า 0 ระบุการรอ ไม่สิ้นสุด ค่าสำหรับพารามิเตอร์ `DataConnTimeOut` สามารถมีช่วงจาก 0 ถึง `MAXINT`
- e เปิดใช้เฉพาะ TLS ที่เปิดใช้โคลเอ็นต์เพื่อสร้าง การเชื่อมต่อกับเซิร์ฟเวอร์
- f ปิดการตรวจสอบพอร์ตที่มี privilege เมื่อโคลเอ็นต์ร้องขอ เซิร์ฟเวอร์ให้เชื่อมต่อกลับไปพอร์ตที่ระบุ โดยดีฟอลต์ `ftpd` ไม่อนุญาตให้โคลเอ็นต์ร้องขอการเชื่อมต่อ ไปที่พอร์ตที่มี privilege เป็นการเผื่อระวังด้านความปลอดภัย
- ff ปิดการตรวจสอบสำหรับทั้งพอร์ตที่มี privilege และ IP แอดเดรส ที่ตรงกับที่ใช้เพื่อควบคุมการเชื่อมต่อ เมื่อโคลเอ็นต์ ร้องขอให้เซิร์ฟเวอร์เชื่อมต่อกลับไปที่โคลเอ็นต์พอร์ตที่ระบุ การใช้ แฟล็ก `ftpd` ไม่ อนุญาตให้การดำเนินการนี้เป็นการเผื่อระวังด้านความปลอดภัย
- H เปิดการล็อกการตรวจสอบสำหรับเหตุการณ์ `FILE_Rename`, `FS_Rmdir` และ `FILE_Unlink` ถ้าเปิดใช้งานเหตุการณ์เหล่านี้สำหรับผู้ใช้ `root`
- k เช็ทอ็อพชัน `SO_KEEPAIVE` ที่กำหนด ในไฟล์ `sys/socket.h` บนซ็อกเก็ตการถ่ายโอนข้อมูล เพื่อเปิดใช้การถ่ายโอนข้อมูลหมดเวลาใช้งานเมื่อ event TCP/IP หยุดทำงาน ช่วงเวลาสูญเปล่ามาจากค่าที่ระบบที่กำหนดโดย อ็อพชัน `tcp_keepidle` และ `tcp_keepintvl` ของคำสั่ง `no` ถ้าไม่มี แฟล็ก การถ่ายโอนข้อมูล `ftpd` จะไม่มีการหมดเวลาใช้งาน
- l ส่งข้อมูลการบันทึกเกี่ยวกับการดำเนินการ `ftpd daemon` ไปที่ `syslogd daemon` ถ้าคุณระบุแฟล็ก `-l`, คุณต้อง แก้ไขไฟล์ `/etc/syslog.conf` และเพิ่มรายการต่อไปนี้:  
  
daemon.info FileName  
  
ถ้า คุณไม่แก้ไขไฟล์ `/etc/syslog.conf` จะไม่มี การสร้างข้อความขึ้น หลังการเปลี่ยนแปลง ไฟล์ `/etc/syslog.conf` วันคำสั่ง `refresh -s syslogd` หรือคำสั่ง `kill -1 SyslogdPID` เพื่อแจ้ง `syslogd daemon` ทราบถึงการเปลี่ยนแปลงในไฟล์คอนฟิกูเรชัน สำหรับข้อมูลเพิ่มเติมเกี่ยวกับระดับดีบั๊ก อ้างอิง ที่ไฟล์ `/etc/syslog.conf`
- q อนุญาตให้ผู้ใช้นิวาร์ที่ขอยย `send_file` ต้องถูกใช้สำหรับการส่ง ไฟล์บนเน็ตเวิร์ก แฟล็กนี้ใช้ได้เฉพาะ เมื่อไฟล์กำลังถูกส่งออกในโหมดไบนารี โดยไม่มีการป้องกัน
- t TimeOut  
ออกจากเซสชันที่ไม่แอ็คทีฟ หลังจากเวลาเป็นวินาทีที่ระบุ โดยตัวแปร `TimeOut` ค่าจำกัดดีฟอลต์คือ 15 นาที (900 วินาที) ค่าหมดเวลาใช้งานใช้กับทั้ง การเชื่อมต่อข้อมูล และการควบคุม
- T MaxTimeOut  
ออกจากเซสชันโคลเอ็นต์ที่ไม่แอ็คทีฟ หลังจากเวลาสูงสุดเป็นวินาทีที่ระบุ โดยตัวแปร `MaxTimeOut` ค่าจำกัดดีฟอลต์คือ 2 ชั่วโมง (7200 วินาที)
- s เปิดใช้งานการดีบั๊กระดับของซ็อกเก็ต
- u OctalVal  
เช็ท umask ของ `ftpd daemon` ตัวแปร `OctalVal` ต้องถูกระบุเป็นค่าฐานแปด เพื่อกำหนด umask Umask ดีฟอลต์คือค่าฐานแปดของ 027 ซึ่งส่งผลให้สิทธิ์ในไฟล์เป็น `rw-r---`
- U คงการปลดล็อกของไฟล์ ขณะทำการถ่ายโอน ถ้าแฟล็กนี้ถูกระบุ กับ `/usr/sbin/ftpd` ไฟล์จะสามารถถูกเปิด ได้ขณะอยู่ในการถ่ายโอน

## ความปลอดภัย

`ftpd daemon` เป็นแอ็พพลิเคชัน PAM-enabled ที่มีชื่อเซอร์วิส `ftp` System-wide configuration เพื่อใช้การพิสูจน์ตัวตน PAM ถูกเช็ทโดยการแก้ไขค่าแอ็ททริบิวต์ `auth_type` ใน `usw stanza` ของ `/etc/security/login.cfg` เป็น `PAM_AUTH` ในฐานะ ผู้ใช้ `root`

กลไกการพิสูจน์ตัวตนที่ใช้เมื่อ PAM ถูกเปิดใช้ขึ้นกับ configuration สำหรับเซิร์ฟเวอร์ ftp ใน `/etc/pam.conf` ftpd daemon ต้องการรายการ `/etc/pam.conf` สำหรับแอคเคาต์ `auth`, `account` และชนิดโมดูล `session` รายการด้านล่างคือ configuration ที่แนะนำใน `/etc/pam.conf` สำหรับเซิร์ฟเวอร์ ftp:

```
#
# AIX ftp configuration
#
ftp auth      required    /usr/lib/security/pam_aix

ftp account  required    /usr/lib/security/pam_aix

ftp session  required    /usr/lib/security/pam_aix
```

## ตัวอย่าง

**หมายเหตุ:** อาร์กิวเมนต์สำหรับ ftpd daemon สามารถถูกระบุโดยใช้ SMIT หรือโดยการแก้ไขไฟล์ `/etc/inetd.conf`

1. เมื่อต้องการสตาร์ท ftpd daemon, ให้พิมพ์ดังต่อไปนี้:

```
startsrc -t ftp
```

คำสั่ง `startsrc` พร้อมกับแฟล็ก `-t` สตาร์ท ftpd subserver คุณต้องใช้แฟล็ก `-t` เพื่อระบุ subserver มิฉะนั้น คำสั่ง จะทำงานไม่ถูกต้อง

2. เมื่อต้องการหยุด ftpd daemon, โดยปกติให้พิมพ์ดังต่อไปนี้:

```
stopsrc -t ftp
```

คำสั่ง `stopsrc` พร้อมกับแฟล็ก `-t` หยุด ftpd subserver คำสั่ง `stopsrc` อนุญาตให้การเชื่อมต่อที่ค้างอยู่เริ่มต้น และการเชื่อมต่อที่มีอยู่ดำเนินการให้สมบูรณ์ แต่ป้องกันไม่ให้เกิดการเริ่มการเชื่อมต่อใหม่ คุณต้องใช้แฟล็ก `-t` เพื่อระบุ subserver มิฉะนั้น คำสั่ง จะทำงานไม่ถูกต้อง

3. เมื่อต้องการบังคับ ftpd daemon และการเชื่อมต่อ ftpd ทั้งหมดหยุด ให้พิมพ์ดังต่อไปนี้:

```
stopsrc -f -t ftp
```

คำสั่ง `stopsrc` พร้อมกับแฟล็ก `-t` และ `-f` บังคับให้ ftpd subserver หยุด ซึ่งจบ การเชื่อมต่อที่ค้างอยู่และการเชื่อมต่อที่มีอยู่ทั้งหมดในทันที

4. เมื่อต้องการแสดงรายงานสถานะแบบสั้น เกี่ยวกับ ftpd daemon ให้พิมพ์ดังต่อไปนี้:

```
lssrc -t ftp
```

คำสั่ง `lssrc` พร้อมกับแฟล็ก `-t` ส่งกลับ ชื่อ process ID และสถานะของ daemon (แอ็คทีฟหรือไม่แอ็คทีฟ) คุณต้องใช้แฟล็ก `-t` เพื่อระบุ subserver มิฉะนั้น คำสั่ง จะทำงานไม่ถูกต้อง

## ไฟล์

ไอเท็ม	คำอธิบาย
/etc/locks/ftpd	มี interlock และที่เก็บ process ID (PID)
/etc/group	มีรหัสผ่านสำหรับกลุ่ม
/etc/passwd	มีรหัสผ่านสำหรับผู้ใช้
/etc/security/login.cfg	มีข้อมูล configuration สำหรับล็อกอินและการพิสูจน์ตัวตนผู้ใช้
/etc/security/passwd	มีรหัสผ่านที่เข้ารหัส
/etc/syslog.conf	มีข้อมูลคอนฟิกูเรชันสำหรับ syslogd daemon.
/usr/samples/tcpip/anon.ftp	มีตัวอย่างเซลล์สคริปต์เพื่อเชื่อมต่อพีแอกเคต anonymous FTP และไฟล์นี้ยังมีคำแนะนำการใช้งาน
/etc/ftpd.cnf	มี configuration พารามิเตอร์สำหรับการสนับสนุน TLS

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง rlogin  
 คำสั่ง telnet  
 คำสั่ง syslogd  
 รูทีนย่อย kvalid\_user  
 ไฟล์ /etc/ftpdusers

---

## คำสั่ง fuser

### วัตถุประสงค์

ระบุกระบวนการโดยใช้ไฟล์หรือโครงสร้างไฟล์

### ไวยากรณ์

```
fuser [-c|-C|-f] [-x] [-d] [-k|-K { SignalNumber|SignalName }] [-u] [-V] File ...
```

### คำอธิบาย

คำสั่ง fuser แสดงรายการหมายเลข กระบวนการที่ใช้ไลคัลหรือรีโมตไฟล์ที่ระบุโดย พารามิเตอร์ *File* สำหรับอุปกรณ์พิเศษแบบบล็อก คำสั่งแสดงกระบวนการที่ใช้ไฟล์ใดๆ บนอุปกรณ์นั้น

หมายเลขกระบวนการแต่ละหมายเลขจะตามด้วยตัวอักษรที่ระบุ การใช้ไฟล์ของกระบวนการ:

ไอเท็ม	คำอธิบาย
c	ใช้ไฟล์เป็นไดเรกทอรีปัจจุบัน
e	ใช้ไฟล์เป็นอ็อบเจกต์ที่รันได้ของโปรแกรม
r	ใช้ไฟล์เป็นไดเรกทอรี root
s	ใช้ไฟล์เป็นไลบรารีที่แบ่งใช้ (หรืออ็อบเจกต์ที่โหลดได้อื่นๆ)

หมายเลขกระบวนการถูกเขียนไปยังเอาต์พุตมาตรฐาน ในหนึ่งบรรทัดโดยมีช่องว่างระหว่างหมายเลขกระบวนการ อีกขระบรรทัดใหม่จะถูกเขียน ไปยังข้อผิดพลาดมาตรฐานหลังจากเอาต์พุตสุดท้ายสำหรับแต่ละตัวถูกดำเนินการไฟล์ เอาต์พุตอื่นทั้งหมด จะถูกเขียนไปยังข้อผิดพลาดมาตรฐาน

คำสั่ง fuser จะไม่ตรวจหากระบวนการที่มีส่วน mmap ที่ descriptor ไฟล์เชื่อมโยงอยู่ตั้งแต่ที่ถูกปิดไป รวมทั้งกระบวนการที่ใช้ FIFOs (ไพพ์ที่ระบุชื่อ) จะไม่ถูกตรวจหาจนกว่า FIFO จะเปิดอย่างสมบูรณ์ ตัวอย่างเช่น กระบวนการที่รอการเรียกระบบเปิดเพื่อดำเนินการให้เสร็จสมบูรณ์จะไม่ถูกพบโดยคำสั่ง fuser

คำสั่ง `fuser` ใช้เพื่อระบุกระบวนการที่กำลังใช้ระบบไฟล์ ถ้าระบบไฟล์เป็นระบบไฟล์เครือข่าย (NFS) และเซิร์ฟเวอร์ NFS ไม่ตอบกลับ คำสั่ง `fuser` อาจค้าง เพื่อหลีกเลี่ยงสถานการณ์ดังกล่าว คุณสามารถตั้งค่าตัวแปรสถานะแวดล้อม `FUSER_VERSION` เป็น 1

## แฟล็ก

ไอเท็ม	คำอธิบาย
-c	รายงานเกี่ยวกับไฟล์ที่เปิดใดๆ ในระบบไฟล์ที่มี <i>File</i>
-C	รายงานเกี่ยวกับไฟล์ที่เปิดใดๆ ในระบบไฟล์ซึ่งมีการติดตั้งไอดีเร็กทอรี ซึ่งระบุโดย พารามิเตอร์ <i>File</i> ถ้าพารามิเตอร์ <i>File</i> ไม่ใช่จุดติดตั้ง คำสั่งจะรายงานข้อผิดพลาด
-d	รายงานเกี่ยวกับไฟล์เปิดใดๆ ที่ไม่ได้ล็อก (ลบทิ้ง) จากระบบไฟล์ที่มี <i>File</i> เมื่อใช้ร่วมกับแฟล็ก -V จะรวมงานหมายเลข inode และ ขนาดของไฟล์ที่ลบด้วย
-f	รายงานเกี่ยวกับอินสแตนซ์ที่เปิดอยู่ของ <i>File</i> เท่านั้น
-K <i>SignalNumber</i>   <i>SignalName</i>	ส่งสัญญาณที่ระบุไปยังแต่ละกระบวนการโลคัล ผู้ใช้ root เท่านั้นที่สามารถคลิกกระบวนการของผู้ใช้อื่น สัญญาณสามารถมีกรระบุเป็น <i>SignalName</i> เช่น KILL สำหรับสัญญาณ SIGKILL หรือ <i>SignalNumber</i> เช่น 9 ค่าที่ถูกต้องสำหรับ <i>SignalName</i> คือค่าที่แสดงโดยคำสั่ง <code>kill -l</code>
-k	ส่งสัญญาณ SIGKILL ไปยังแต่ละกระบวนการโลคัล ผู้ใช้ root เท่านั้นที่สามารถคลิกกระบวนการของผู้ใช้อื่น หมายเหตุ: <code>fuser -k</code> หรือ <code>-K</code> อาจไม่สามารถตรวจหาและคลิกกระบวนการใหม่ที่ถูกสร้างในทันทีหลังจากโปรแกรมเริ่มต้น การรัน
-u	แสดงชื่อล็อกอินสำหรับกระบวนการโลคัลในวงเล็บหลัง หมายเลขกระบวนการ
-V	ให้อาตปุตรรายละเอียด
-x	ใช้ร่วมกับ -c หรือ -f เพื่อรายงานเกี่ยวกับอ็อบเจกต์ที่ที่รันได้และที่โหลดได้นอกเหนือจากเอาต์พุต <code>fuser</code> มาตรฐาน

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ตัวอย่าง

- ในการแสดงหมายเลขกระบวนการของกระบวนการโลคัลโดยใช้ไฟล์ `/etc/passwd` ให้ป้อน:
 

```
fuser /etc/passwd
```
- ในการแสดงหมายเลขกระบวนการและชื่อล็อกอินผู้ใช้โดยใช้ไฟล์ `/etc/filesystems` ให้ป้อน:
 

```
fuser -u /etc/filesystems
```
- ในการยุติกระบวนการทั้งหมดโดยใช้ระบบไฟล์ที่กำหนดให้ป้อน:
 

```
fuser -k -x -u -c /dev/hd1
```

หรือ

```
fuser -kxuc /home
```

คำสั่งแสดงรายการหมายเลขกระบวนการ และชื่อผู้ใช้ จากนั้น ยุติแต่ละกระบวนการที่กำลังใช้ระบบไฟล์ `/dev/hd1` (`/home`) ผู้ใช้ root เท่านั้นที่สามารถยุติกระบวนการที่เป็นของ ผู้ใช้อื่น คุณอาจต้องการใช้คำสั่งนี้หากคุณกำลังยกเลิกการเมาท์ ระบบไฟล์ `/dev/hd1` และกระบวนการที่กำลังเข้าถึงระบบไฟล์ `/dev/hd1` เพื่อการป้องกันนี้

- ในการแสดงรายการกระบวนการทั้งหมดที่กำลังใช้ไฟล์ซึ่งถูกลบออกจาก ระบบไฟล์ที่กำหนดให้ป้อน:
 

```
fuser -d /usr
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/dev/kmem	ใช้สำหรับอิมเมจระบบ
/dev/mem	ใช้สำหรับอิมเมจระบบเช่นกัน

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง kill

คำสั่ง mount

คำสั่ง ps

ความปลอดภัย

---

## คำสั่ง fwtmp

### วัตถุประสงค์

จัดการเรกคอร์ดแอดเคาต์เวลาเชื่อมต่อ โดยการอ่านไบนารีเรกคอร์ดในรูปแบบ **wtmp** จาก อินพุตมาตรฐานและแปลงเป็นเรกคอร์ด ASCII ที่จัดรูปแบบ คุณสามารถใช้เวอร์ชัน ASCII เพื่อแก้ไขเรกคอร์ดที่ไม่ถูกต้อง

### ไวยากรณ์

```
/usr/sbin/acct/fwtmp [ -i ] [ -c ] [ -X ] [ -L ]
```

### คำอธิบาย

คำสั่ง **fwtmp** จัดการเรกคอร์ดแอดเคาต์ โดยการอ่านไบนารีเรกคอร์ดในรูปแบบ **wtmp** จากอินพุต มาตรฐานและแปลงเป็นเรกคอร์ด ASCII ที่จัดรูปแบบ

### แฟล็ก

ไอเท็ม	คำอธิบาย
-i	รับเรกคอร์ด ASCII ในรูปแบบ <b>utmp</b> เป็นอินพุต
-c	แปลงเอาต์พุตเป็นฐานสองที่จัดรูปแบบ <b>utmp</b>
-ic	แปลงอินพุตเรกคอร์ดที่จัดรูปแบบ ASCII <b>utmp</b> เป็นไบนารีเอาต์พุตเรกคอร์ด
-X	พิมพ์อักขระที่มีอยู่ทั้งหมดของชื่อผู้ใช้แต่ละชื่อแทน การตัดเหลือ 8 อักขระแรก
-L	พิมพ์อักขระที่มีอยู่ทั้งหมดของชื่อโฮสต์แต่ละชื่อแทนการตัดเหลือ 32 อักขระแรก

### ความปลอดภัย

ค่าควบคุมการเข้าถึง: คำสั่งเหล่านี้ควรให้สิทธิ์การเข้าถึง execute (x) เฉพาะกับสมาชิกของกลุ่ม **adm**

### ตัวอย่าง

- ในการแปลงไบนารีเรกคอร์ดในรูปแบบ **wtmp** เป็นเรกคอร์ด ASCII ชื่อ **dummy.file** ให้ป้อน:

```
/usr/sbin/acct/fwtmp < /var/adm/wtmp > dummy.file
```

เนื้อหาของไฟล์ **wtmp** ไบนารีถูกเปลี่ยนทิศทาง ไปยังไฟล์ ASCII ดัมมี่

2. ในการแปลง ASCII dummy.file เป็น ไบนารีไฟล์ในรูปแบบ wtmp ชื่อ /var/adm/wtmp ให้ป้อนคำสั่ง fwtmp ที่มีอ็อปชัน -ic:

```
/usr/sbin/acct/fwtmp -ic < dummy.file > /var/adm/wtmp
```

ไฟล์ ASCII ดัมมีถูกเปลี่ยนทิศทางไปยังไฟล์ wtmp ไบนารี

## ไฟล์

### ไอเท็ม

/usr/sbin/acct/fwtmp  
/var/adm/wtmp  
/usr/include/utmp.h

### คำอธิบาย

มีคำสั่ง fwtmp  
มีเรกคอร์ดของการเปลี่ยนแปลงวันที่ที่รวมวันที่เก่า และวันที่ใหม่  
มีเรกคอร์ดประวัติที่รวม เหตุผล วันที่และเวลา

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง runacct

คำสั่ง wtmpfix

การตั้งค่าระบบการบัญชี

---

## คำสั่ง fxfer

### วัตถุประสงค์

ถ่ายโอนไฟล์ระหว่างระบบโวลคัลและโฮสต์คอมพิวเตอร์ที่เชื่อมต่อโดย HCON

### ไวยากรณ์

เมื่อต้องการรีสตาร์ท Interrupted File Transfer

```
fxfer -R [-n SessionName]
```

เมื่อต้องการดาวน์โหลดไฟล์จากโฮสต์

```
fxfer [-n SessionName] [-a|-r] [-d] [-c|-C] [-J] [-f FileName] [-F] [-H HostType] [-I InputField] [-q] [-t [[-l] [-s]]|-T [[-l] [-s] [-b]]]
```

```
[-v] [-x HostLogin] [-e] [-X CodeSet] SourceFile DestFile
```

เมื่อต้องการอัปโหลดไฟล์ไปที่โฮสต์

```
fxfer [-n SessionName] [-a|-r] [-u] [-c|-C] [-J] [-f FileName] [-H HostType] [-q] [-t [[-l] [-s]]|-T [[-l] [-s]]] [-l] [-s] [-v] [-x HostLogin] [-X CodeSet] [-F|-V|-U] [-B BlockSize] [-L LogIRecLength] [-I InputField] [-S NumberUnits [,IncreaseUnits|,IncreaseUnits,UnitType|,UnitType]] [-M Volume] [-N Unit] [-k] SourceFile DestFile
```

เมื่อต้องการแสดงจอภาพวิธีใช้

fxfer -h

## คำอธิบาย

คำสั่ง **fxfer** ถ่ายโอนไฟล์ ระหว่าง local system และเมนเฟรมโฮสต์ที่เชื่อมต่อโดย Host Connection Program (HCON) ไฟล์อาจถ่ายโอนจาก local system ไปที่โฮสต์ (การอัปโหลด) หรือจากโฮสต์ไปที่ local system (การดาวน์โหลด) คำสั่ง **fxfer** ถ่ายโอนไฟล์ชื่อโดยใช้พารามิเตอร์ *SourceFile* ไปยังไฟล์ชื่อโดยใช้พารามิเตอร์ *DestFile* การถ่ายโอนเริ่มขึ้นผ่านเซสชัน HCON ที่ต้องการ โปรไฟล์เซสชันจำเพาะ หรือเซสชันที่มีอยู่

โฮสต์ระบบปฏิบัติการอาจเป็น VM/CMS, MVS/TSO, CICS/VS (สำหรับ CICS/MVS หรือ CICS/VSE), VSE/ESA, หรือ VSE/SP, ที่มีเวอร์ชันของ 3270 File Transfer Program (**IND\$FILE** หรือเทียบเท่า) ที่ตรงกันติดตั้งอยู่ เวอร์ชันของโปรแกรมการถ่ายโอนไฟล์ของ โฮสต์ถูกกำหนดโดยค่า File Transfer Program ในเซสชันโปรไฟล์ คำสั่ง **fxfer** สนับสนุนการถ่ายโอนของข้อความหรือข้อมูลไบนารี ไฟล์จะถ่ายโอนไปที่หรือจาก โฮสต์โดยมีหรือไม่มี การแปลง ASCII หรือ EBCDIC

กลไกความปลอดภัยป้องกันการเข้าถึงที่ไม่ได้รับอนุญาต การทำลายไฟล์ที่มีอยู่ หรือการสูญเสียข้อมูล ถ้าผู้ใช้ที่ไม่ใช่ HCON เรียก คำสั่ง **fxfer** คำสั่งจะล้มเหลว ถ้าคำสั่ง **fxfer** ถูกอินเตอร์รัปต์ก่อนเสร็จสมบูรณ์ สถานะของการถ่ายโอนจะถูกบันทึกในไฟล์ RESTART

ถ้าคำสั่ง **fxfer** ถูกเรียก ด้วยแฟล็ก **-h** จะมีการแสดงจรรยาวิธีใช้ ถ้าคำสั่งถูกเรียกด้วยแฟล็ก **-R** จะมีการค้นหาไดเรกทอรี **\$HOME** สำหรับไฟล์รีสตาร์ท ถ้าไฟล์รีสตาร์ทที่มีอยู่ เมนูรีสตาร์ทจะแสดง เปิดใช้การ รีสตาร์ทของการถ่ายโอนไฟล์ ถ้าแฟล็ก **-h** และ **-R** ไม่ถูกระบุ คำสั่งจะพยายามดำเนินการถ่ายโอนไฟล์ ที่ระบุ

ข้อมูลคำสั่ง **fxfer** รวมถึง:

- แฟล็ก
- แฟล็กสำหรับคุณสมบัติไฟล์โฮสต์
- ตัวอย่าง
- ไฟล์

คำสั่งนี้ต้องการ:

- อะแดปเตอร์ที่ใช้เพื่อเชื่อมต่อเมนเฟรมโฮสต์
- หนึ่งในระบบปฏิบัติการเมนเฟรมต่อไปนี้ถูกติดตั้งบนโฮสต์:
  - VM/SP CMS
  - VM/XA CMS
  - MVS/SPTSO/E
  - MVS/XA TSO/E
  - CICS/VS (สำหรับ CICS/MVS หรือ CICS/VSE)
  - VSE/ESA
- โปรแกรมเมนเฟรม Host-Supported File Transfer (**IND\$FILE** หรือเทียบเท่า) ถูกติดตั้งบนเมนเฟรม

เซสชันโปรไฟล์สำหรับการใช้คำสั่ง **fxfer**

คำสั่ง **fxfer** สื่อสาร กับเซสชัน HCON และอาจต้องการเซสชันโปรไฟล์จำเพาะ เซสชัน โปรไฟล์กำหนด:

- พาดการสื่อสารไปที่โฮสต์
- ชนิดโฮสต์
- ทิศทางการถ่ายโอนไฟล์ดีฟอลต์ (ลงหรือขึ้น)
- เวลาการกู้คืน
- ช่วงเวลารอการถ่ายโอนไฟล์

เมื่อคำสั่ง `fxfer` ดำเนินการ ล็อกออนอัตโนมัติไปไฟล์ยังสามารถกำหนด:

- Host logon ID
- AUTOLOG node ID
- การติดตาม AUTOLOG เปิดหรือไม่
- ค่าหมดเวลาใช้งาน AUTOLOG

โดยปกติผู้ใช้ระบบเซสชันโปรไฟล์เมื่อเรียก คำสั่ง `fxfer` ข้อยกเว้นเกิดขึ้น เมื่อคำสั่ง ถูกรันจาก subshell ของเซสชันที่มีอยู่ในกรณีนี้ ถ้าผู้ใช้ไม่ระบุเซสชันโปรไฟล์ คำสั่ง `fxfer` จะใช้เซสชันที่มีอยู่ ถ้าเซสชันที่เหมาะสมไม่ได้รับอยู่ คำสั่ง `fxfer` จะพยายามเรียกเซสชันใหม่

คำสั่ง `fxfer` ค้นหา เซสชัน HCON ดังนี้:

- เมื่อเรียกใช้โดยไม่มีแฟล็ก `-n SessionName` :
  - ถ้าคำสั่ง `fxfer` ถูกเรียกจาก subshell ของเซสชันที่มีอยู่ คำสั่งใช้เซสชันที่เชื่อมโยงกับ subshell (ที่กำหนดโดยตัวแปรสถานะแวดล้อม `$SNAME`)
  - ถ้า ไม่เรียกจาก subshell ของอิมูเลเตอร์เซสชัน คำสั่ง `fxfer` เรียก ข้อความแสดงความผิดพลาดและจบการทำงาน
- เมื่อเรียกด้วยแฟล็ก `-n SessionName` การถ่ายโอนไฟล์ ดำเนินในเซสชันที่ระบุ ถ้าเซสชันที่ระบุไม่มีอยู่ คำสั่งจะค้นหาโปรไฟล์สำหรับเซสชันนั้น ถ้าไม่พบ โปรไฟล์เซสชันที่ระบุ คำสั่ง `fxfer` จะแสดงข้อความแสดงความผิดพลาดและจบการทำงาน ถ้าโปรไฟล์ที่ระบุ มีอยู่ คำสั่ง `fxfer` พยายามล็อกออนอัตโนมัติ ไปที่โฮสต์โดยใช้ค่า AUTOLOG ที่กำหนดในเซสชันโปรไฟล์ค่าที่กำหนดด้วยแฟล็ก `-x` หรือโดยการพร้อมต์ ผู้ใช้เพื่อของข้อมูลล็อกออนที่จำเป็น

### การอินเตอร์รัปต์และการรีสตาร์ทการถ่ายโอนไฟล์

คำสั่ง `fxfer` สามารถถูกอินเตอร์รัปต์โดยโอเปอเรเตอร์หรือข้อผิดพลาดการสื่อสารที่กู้คืนไม่ได้ ก่อนทำงานสมบูรณ์ ถ้าอินเตอร์รัปต์ คำสั่งจะบันทึกสถานะของการถ่ายโอนในไฟล์ RESTART การถ่ายโอนสามารถถูกรีสตาร์ทจากการเริ่มต้นโดยไม่มี การสูญเสียข้อมูล

ถ้าคุณรันการถ่ายโอนไฟล์ใหม่หลังจากการถ่ายโอนที่อินเตอร์รัปต์ คำสั่ง `fxfer` จะส่งสัญญาณว่าไฟล์ RESTART ได้ถูกสร้าง และแสดงอ็อปชันเหล่านี้:

- รีสตาร์ทการถ่ายโอนไฟล์ที่อินเตอร์รัปต์
- บันทึกไฟล์ RESTART และออกจากโปรแกรม การถ่ายโอนไฟล์
- ลบไฟล์ RESTART และออกจากโปรแกรม การถ่ายโอนไฟล์
- ลบไฟล์ RESTART และดำเนินการถ่ายโอนปัจจุบัน ต่อ

คำสั่ง `fxfer` พร้อมกับแฟล็ก `-R` รีสตาร์ทการถ่ายโอนไฟล์ ที่อินเตอร์รัปต์

ถ้าการสื่อสารโฮสต์สูญหายหรือมีการยกเลิกการเชื่อมต่อระหว่างการถ่ายโอนไฟล์ที่เริ่มต้นด้วยการลือกอนอัตโนมัติ การการถ่ายโอนไฟล์พยายาม กู้คืนโดยทำการเชื่อมต่อใหม่และลือกอนกลับสู่โฮสต์ เวลาการกู้คืน สำหรับความพยายามนี้ถูกกำหนดโดยค่า File Transfer Recovery Time ในเซสชันโปรไฟล์ เมื่อการเชื่อมต่อของโฮสต์ถูกสร้างขึ้นใหม่ การถ่ายโอนไฟล์ ดำเนินต่อจากจุดเริ่มต้น ถ้าไม่สามารถสร้างการสื่อสารใหม่ได้ โปรแกรมการถ่ายโอนไฟล์จะสร้างไฟล์ RESTART

เมื่อการถ่ายโอนไฟล์สูญเสียการสื่อสารกับโฮสต์ ผู้ใช้ต้องรีสตาร์ทที่มิวเลเตอร์เซสชันและลือกอนกลับสู่โฮสต์ก่อนพยายามรีสตาร์ทการถ่ายโอนไฟล์

## ไฟล์ต้นทางและปลายทาง

คำสั่ง `xfxer` ต้องใช้พารามิเตอร์ `SourceFile` และ `DestFile` พารามิเตอร์ `SourceFile` ระบุไฟล์ต้นฉบับ สำหรับการถ่ายโอนไฟล์ พารามิเตอร์ `DestFile` ระบุไฟล์ปลายทาง สำหรับการถ่ายโอนไฟล์ ชื่อไฟล์ local system อยู่ในรูปแบบธรรมดา ชื่อโฮสต์เป็นไปตามระเบียบการตั้งชื่อโฮสต์ ซึ่งเป็นหนึ่งในรูปแบบต่อไปนี้:

ชนิดโฮสต์	รูปแบบชื่อไฟล์
VM/CMS	" <i>FileName FileType FileMode</i> " หมายเหตุ: ต้องใช้ "" (เครื่องหมายคำพูด) สำหรับชื่อไฟล์ VM/CMS ทั้งหมด เพื่อให้แน่ใจว่าถ่ายโอนไฟล์อย่างเหมาะสม
MVS/TSO	"['] <i>DataSetName</i> [ ( <i>MemberName</i> ) ][ / <i>Password</i> ][']"

โดยที่:

*DataSetName*

ระบุชุดข้อมูลต่อเนืองทางฟิลิคัลหรือเซตข้อมูล พาร์ติชัน

(*MemberName*)

ระบุชื่อของสมาชิกในไดเรกทอรีของเซตข้อมูล พาร์ติชัน () (วงเล็บ) ปิด *MemberName* จำเป็นต้องมี

*Password*

จำเป็น ถ้าการปกป้องรหัสผ่านถูกระบุสำหรับเซตข้อมูล MVS/TSO จำเป็นต้องมี / (slash) นำหน้า *Password*

Notes:

1. "" (เครื่องหมายคำพูด) จำเป็นสำหรับชื่อไฟล์ MVS/TSO ทั้งหมด เพื่อประกันความถูกต้องของการถ่ายโอนไฟล์
2. เมื่อระบุชื่อพารามิเตอร์สำหรับชื่อไฟล์ MVS/TSO ให้ใช้ ' (เครื่องหมายคำพูดเดี่ยว) ภายใน " (เครื่องหมายคำพูดคู่) อย่าใส่ช่องว่างระหว่างเครื่องหมายคำพูด คู่และเดี่ยว หรือระหว่างเครื่องหมายคำพูดและชื่อ ไฟล์

CICS/VS	" <i>FileName</i> "
VSE/ESA	" <i>FileName FileType</i> "

Notes:

1. "" (เครื่องหมายคำพูด) จำเป็น สำหรับ CICS/VS, VSE/ESA ทั้งหมด และชื่อไฟล์ VSE/SP เพื่อ ประกันความถูกต้องของการถ่ายโอนไฟล์
2. CICS/VS, VSE/ESA, และระเบียบชื่อไฟล์ VSE/SP อนุญาตให้ชื่อไฟล์ยาวได้ถึง 8 อักขระ
3. ในสถานะแวดล้อม DBCS, HCON ไม่สนับสนุนโฮสต์ VSE

## แฟล็ก

หมายเหตุ: สำหรับการสนับสนุน Double-Byte Character Set (DBCS) ที่รวม ภาษาญี่ปุ่น-อังกฤษ, ภาษาญี่ปุ่นคาตากานะ, ภาษาเกาหลี หรือภาษาจีนดั้งเดิม คำนี้ถึงข้อควรพิจารณา เหล่านี้:

- ถ้า DBCS -I หรือแฟล็ก -s ถูกระบุ หนึ่งในแฟล็กการแปล (-t, -T หรือ -J) ยังต้องถูก ระบุหรือไม่แล้วแฟล็ก DBCS จะถูกละเว้น
- แฟล็ก -M, -N และ -k ถูกใช้เฉพาะกับโฮสต์ MVS/TSO

- แฟล็ก -e ใช้ได้กับโปรแกรม CICS® สำหรับการดาวน์โหลดเท่านั้น
- แฟล็ก -b ใช้ได้เฉพาะสำหรับการดาวน์โหลด

**ไอเท็ม**

	<b>คำอธิบาย</b>
-a	ผนวกไฟล์ที่กำหนดโดย <i>SourceFile</i> ไปที่ไฟล์ที่กำหนดโดย <i>DestFile</i> , ถ้าไฟล์ปลายทางมีอยู่ แฟล็กนี้ ถูกละเว้นและไฟล์ปลายทางถูกสร้าง ถ้าไฟล์ที่กำหนดโดย <i>DestFile</i> ไม่มีอยู่ หมายเหตุ: แฟล็ก -a ไม่ถูกต้องเมื่ออัปโหลดไฟล์ไปยังโฮสต์ CICS/VS สำหรับ VSE/ESA แฟล็ก -a ใช้ได้เฉพาะการอัปโหลดไปที่ CICS หน่วยเก็บชั่วคราว (FILE=TS)
-b	คงช่องว่างที่ท้ายของแต่ละเรกคอร์ด เมื่อใช้กับแฟล็ก -t, -T, -c, หรือ -C แฟล็ก -b ถูกสนับสนุน เฉพาะในสภาวะแวดล้อม DBCS
-c	ในสภาวะแวดล้อม DBCS แฟล็ก -c เปลี่ยนโค้ด LF (line-feed) ของไฟล์ไปเป็นโค้ด CRLF (carriage return line-feed) ถ้า การถ่ายโอนไฟล์เป็นการอัปโหลด สำหรับการดาวน์โหลดการถ่ายโอนไฟล์ แฟล็ก -c เปลี่ยนโค้ด CRLF ของไฟล์เป็นโค้ด LF
-C	ในสภาวะแวดล้อม DBCS แฟล็ก -C flag ยับยั้ง การส่งโค้ด EOF (end-of-file) ของไฟล์ PC-DOS ถ้าการถ่ายโอนไฟล์ คืออัปโหลด สำหรับการดาวน์โหลดการถ่ายโอนไฟล์ แฟล็ก -C ผนวกโค้ด EOF: x'1A' ที่ตำแหน่งสุดท้ายของไฟล์ PC-DOS
-d	ดาวน์โหลดไฟล์โดยการถ่ายโอนไฟล์จากโฮสต์ไปที่ local system ถ้าแฟล็กนี้หรือแฟล็ก -u ไม่ถูกระบุ คุณสมบัติ File Transfer Direction ใน เซสชันโปรไฟล์จะกำหนดทิศทางของการถ่ายโอน หมายเหตุ: เมื่อดาวน์โหลด ไฟล์ที่แปลจาก VSE/ESA host file transfer (FILE=HTF) ไฟล์จะถูกลบออกจากระบบโฮสต์ ยกเว้นคุณระบุแฟล็ก -I "KEEP"
-e	ลบคิวหน่วยเก็บชั่วคราวเมื่อเสร็จสิ้นการถ่ายโอน ไฟล์ ใช้แฟล็กนี้เฉพาะกับ CICS โฮสต์สำหรับการดาวน์โหลด แฟล็ก -e ถูกสนับสนุนเฉพาะในสภาวะแวดล้อม DBCS
-f <i>FileName</i>	นำเอาดพุดการวินิจฉัยกระบวนการถ่ายโอนไฟล์ (หรือสถานะ การถ่ายโอนไฟล์) ไปไว้ในไฟล์ที่ระบุโดยตัวแปร <i>FileName</i>  ถ้าแฟล็ก -f ไม่ถูกระบุสำหรับการถ่ายโอนอะซิงโครนัสข้อความถูกนำไปไว้ในไฟล์ \$HOME/hconerrors ถ้าแฟล็ก -f ไม่ถูกระบุ สำหรับการถ่ายโอนซิงโครนัส ข้อความจะถูกส่งไปที่เอาต์พุดมาตรฐาน
	ข้อความที่เกี่ยวข้องกับข้อผิดพลาดในการระบุพารามิเตอร์ การถ่ายโอนไฟล์หรือชื่อไฟล์ หรือความล้มเหลวในกระบวนการถ่ายโอนไฟล์ ถูกส่งไปที่เอาต์พุดมาตรฐาน (ถ้าเป็นจอภาพ local system) หรือไปที่ไฟล์ \$HOME/hconerrors (ถ้าเอาต์พุดมาตรฐานไม่ใช่จอภาพ local system)
-h	แสดงจอภาพวิธียูสสำหรับคำสั่ง <i>fxfer</i> จอภาพนี้สรุปแต่ละแฟล็กคำสั่งและการดำเนินการของคำสั่ง เมื่อมีการระบุแฟล็กนี้ แฟล็กอื่นทั้งหมดจะถูกละเว้นและไฟล์ไม่ถูกถ่ายโอน
	<b>Notes:</b>
	1. ถ้าแฟล็ก -h ถูกใช้ แฟล็กอื่นทั้งหมด จะถูกละเว้น ไม่มีการถ่ายโอนไฟล์
	2. ถ้าคำสั่ง <i>fxfer</i> ไม่ได้ถูกเริ่มต้น จาก subshell ของเซสชัน HCON ที่มีอยู่ จำเป็นต้องมีแฟล็ก -h หรือแฟล็ก -n ระบุชนิดของโฮสต์ ตัวแปร <i>HostType</i> อาจมีค่าเหล่านี้:
-H <i>HostType</i>	
	CMS VM/SPCMS หรือ VM/XA CMS
	TSO MVS/SP TSO หรือ MVS/XA TSO
	CICS CICS/VS ( CICS ชนิดโฮสต์รวมถึง CICS/VSE, CICS/MVS, CICS/ESA และ CICS/MVS/ESA)
	VSE VSE/ESA (ไม่สนับสนุนในสภาวะแวดล้อม DBCS)
	ถ้าแฟล็ก -H ถูกละเว้น ค่าที่ระบุโดยคุณสมบัติ Host Type ในเซสชันโปรไฟล์จะถูกใช้ ผู้ใช้ต้องระบุระบบปฏิบัติการของโฮสต์ที่ต้องการ
	<b>Notes:</b>
	1. ถ้าคุณระบุค่า CICS หรือ VSE และระบบส่งคืนข้อผิดพลาด ให้ทำซ้ำคำสั่งด้วยค่าอื่น โปรแกรม CICS และ VSE IND\$FILE ทำงานแทนกันได้ แต่ มีความแตกต่างขนาดเฮดเดอร์ 6-ไบต์ ที่ทำให้เวอร์ชันทำงาน เข้ากันไม่ได้ โฮสต์ปลายทางอาจใช้เวอร์ชันอื่นของ โปรแกรม
	2. เมื่อต้องการถ่ายโอนไฟล์ไปที่โฮสต์ MVS/TSO คุณอาจจำเป็นต้องออกจากโหมดผู้จัดการเซสชัน ก่อนการเริ่มการถ่ายโอนไฟล์

ไอเอ็ม

-I InputField

คำอธิบาย

ระบุชื่อพื้นที่การถ่ายโอนไฟล์โฮสต์ที่กำหนดไว้โดยตรงในคำสั่ง IND\$FILE และอนุญาตให้หมายเหตุในคำสั่ง IND\$FILE ถูกวางไว้หลัง (วงเล็บขวา) ค่าที่ระบุโดยตัวแปร InputField ถูกกำหนดไว้ในเครื่องหมายคำพูด ดังนี้:

-I "FILE=TS) This is a comment"

-J

หมายเหตุ: ฟิวต์ -I ไม่สนับสนุนให้ใช้ในสภาวะแวดล้อม DBCS

อนุญาตการแปลงข้อมูลระหว่าง EBCDIC และ ASCII และการทำข้อมูลอักขระ SI/SO ให้เป็นข้อมูลปกติ การเปลี่ยนกับทิศทางของการถ่ายโอน:

Upload แปลอักขระ 1-ไบต์ของไฟล์เป็นโค้ด EBCDIC สำหรับประเทศ DBCS โค้ดส่วนขยายถูกแปลเป็นโค้ด DBCS ที่เหมาะสม อักขระ SO/SI ถูกแทรกลงในฟิวต์ DBCS ที่มีอักขระ DBCS ถ้าไฟล์มีโค้ดควบคุม 0x1E หรือ 0x1F, โค้ดจะถูกแทนที่ด้วยอักขระ SO และ SI ตามลำดับ

Download

แปลโค้ด EBCDIC เป็นอักขระ 1-ไบต์ของไฟล์; สำหรับ DBCD โค้ด DBCS ถูกแปลเป็นโค้ดส่วนขยาย ลบอักขระ SO/SI ออกจากฟิวต์ DBCS

-k

หมายเหตุ: ฟิวต์ -J สนับสนุนให้ใช้ในสภาวะแวดล้อม DBCS เท่านั้น

รีลีสรีกคอร์ดที่ไม่ได้ใช้ในชุดข้อมูลเมื่อการโอนย้ายไฟล์เสร็จสมบูรณ์ ใช้แฟล็กนี้เฉพาะในสภาวะแวดล้อม MVS/TSO แฟล็ก -k ถูกสนับสนุนเฉพาะในสภาวะแวดล้อม DBCS

-I

ระบุภาษาโฮสต์ในสภาวะแวดล้อม DBCS อีพซันนี้ ต้องถูกใช้กับหนึ่งในแฟล็กการแปล (-t, -T, or -J) ถ้า -t, -T หรือ -J ถูกละเว้น แฟล็ก -I ถูกละเว้น ถ้าแฟล็ก -I ไม่ถูกระบุ ภาษาโฮสต์ที่กำหนดในเซชันโปรไฟล์ถูกใช้ ถ้าแฟล็ก -I ถูกระบุ ภาษาโฮสต์ที่ใช้คือภาษาสำรองของภาษาที่กำหนดในเซชันโปรไฟล์ ตัวอย่าง ถ้าคุณสมบัติภาษาในเซชันโปรไฟล์คือ JPK (ภาษาญี่ปุ่น คาตาคานะ) ภาษาโฮสต์ที่ใช้สำหรับการถ่ายโอนไฟล์จะเป็น Japanese-English แฟล็ก -I ถูกสนับสนุนเฉพาะในสภาวะแวดล้อม DBCS

-M Volume

ระบุหมายเลขอนุกรมวอลุ่มของโฮสต์ดีสก์สำหรับการจัดสรรชุดข้อมูล ใช้แฟล็กนี้เฉพาะในสภาวะแวดล้อม MVS/TSO แฟล็ก -M ถูกสนับสนุนเฉพาะในสภาวะแวดล้อม DBCS

-n SessionName

ระบุชื่อของเซชันที่กำหนดก่อนหน้าซึ่งคุณสมบัติ ควบคุมการถ่ายโอนไฟล์ ชื่อเซชันเป็นอักขระเดี่ยวในช่วง a ถึง z ตัวพิมพ์ใหญ่จะถูกแปลเป็นตัวพิมพ์เล็ก

แฟล็ก -n SessionName จำเป็น ยกเว้นเมื่อผู้ใช้กำลังกำหนดค่าเริ่มต้นคำสั่ง fxfer จาก subshell ของเซชันที่มีอยู่ในกรณีนี้ ถ้าแฟล็ก -n ไม่ถูกใช้ คำสั่ง fxfer จะดีฟอลต์เป็น เซชันที่มีอยู่

Notes:

1. เซชันที่ระบุต้องถูกกำหนดไว้ก่อนหน้าโดยใช้คำสั่งพารามิเตอร์ smit hcon หรือคำสั่ง mkhcons

-N Unit

2. ถ้าคำสั่ง fxfer ไม่ได้ถูกเริ่มต้นจาก subshell ของเซชัน HCON ที่มีอยู่จำเป็นต้องมีแฟล็ก -h หรือแฟล็ก -n ระบุชนิดหน่วยของโฮสต์ดีสก์สำหรับการจัดสรรชุดข้อมูล ใช้แฟล็กนี้เฉพาะในสภาวะแวดล้อม MVS/TSO แฟล็ก -N ถูกสนับสนุนเฉพาะในสภาวะแวดล้อม DBCS

-q

รับการถ่ายโอนไฟล์แบบอะซิงโครนัส เป็นการประมวลผลส่วนหลัง ถ้า การถ่ายโอนไฟล์โดยยังไม่เสร็จสิ้น การร้องขอการถ่ายโอนปัจจุบันจะถูกคิว ถ้าแฟล็ก -q ไม่ถูกระบุ การดำเนินการถ่ายโอนไฟล์ เป็นซิงโครนัส ถ้าแฟล็ก -f ไม่ถูกระบุเอาต์พุตการวินิจฉัยและสถานะถูกกำหนดไว้ในไฟล์ \$HOME/hconerrors

-r

หมายเหตุ: ระบบจำกัดจำนวนไบต์ที่อนุญาตให้หนึ่งคิวข้อความ Interprocess Communication (IPC) ดังนั้น จำนวนการถ่ายโอนไฟล์มากที่สุดที่สามารถถูก คิวในหนึ่งครั้งคือ 580 โดยประมาณ ระบุการแทนที่ไฟล์ที่มีอยู่บนโฮสต์ (ออฟโหลด) หรือ โฮสต์ที่มีอยู่บน local system (ดาวน์โหลด) เมื่อดาวน์โหลด การแทนที่ ถูกดำเนินการเมื่อการถ่ายโอนสำเร็จ ซึ่งประกันว่าไฟล์ที่มีอยู่จะไม่สูญหายหรือถูกทำลาย ถ้าการถ่ายโอนไม่สมบูรณ์ไม่ว่าด้วยเหตุใด ถ้าแฟล็ก -r ถูกระบุและ ไฟล์ไม่มีอยู่ ไฟล์จะถูกสร้างระหว่างการถ่ายโอนไฟล์ ถ้าแฟล็ก -r ไม่ถูกระบุและมีไฟล์ปลายทางอยู่ข้อความแสดงข้อผิดพลาดจะเกิดขึ้น

สำหรับการออฟโหลด แฟล็ก -r ต้องถูกระบุเมื่อใช้ เวอร์ชันของโปรแกรมการถ่ายโอนไฟล์ของโฮสต์ที่ต่ำกว่า PTF UR20455

สำหรับ MVS/TSO หรือ PTF UR90118 สำหรับ VM/CMS สำหรับ VSE และ CICS แฟล็ก -r ถูกละเว้น

หมายเหตุ: โปรแกรมการถ่ายโอนไฟล์โฮสต์โดยปกติมีค่าดีฟอลต์ เพื่อแทนที่สำหรับไฟล์ ถ้าไม่มีการแทนที่ให้เพิ่ม -I "replace" ให้กับคำสั่ง fxfer เพื่อระบุการแทนที่ ข้อควรใส่ใจ: เมื่อทำการแทนที่ไฟล์บนโฮสต์ คุณต้องระบุความยาวเรีกคอร์ดเชิงตรรกะ (แฟล็ก -L) และ รูปแบบเรีกคอร์ด (แฟล็ก -F หรือ -V) เท่ากับความยาว เรีกคอร์ดเชิงตรรกะและรูปแบบเรีกคอร์ดของไฟล์ที่มีอยู่ ถ้าคุณไม่ทำดังนี้ อาจเกิดข้อมูลเสียหายขึ้น ซึ่งไม่ใช้กับ VSE/ESA

## ไอเท็ม

- R** คำอธิบาย  
รีสตาร์ทการถ่ายโอนไฟล์ก่อนหน้า (ซึ่งถูกอินเตอร์รัปต์โดยผู้ใช้ หรือพยายามกู้คืนไม่สำเร็จ) โดยใช้ข้อมูลที่บันทึกจากหนึ่งในไฟล์ RESTART: ไฟล์ \$HOME/x\_fxfer.r หรือไฟล์ \$HOME/i\_fxfer.r ถ้าการถ่ายโอนไฟล์ไม่ถูกเรียก จาก subshell ของเซสชันที่มีอยู่ แฟล็ก -n SessionName ต้องถูกรวมไว้เพื่อระบุเซสชันที่จะถูกใช้ ถ้าแฟล็ก -R ถูก ระบุร่วมกับแฟล็กการถ่ายโอนไฟล์อื่น แฟล็กเหล่านั้น จะถูกละเว้นและเมนูการถ่ายโอนไฟล์ RESTART จะถูกแสดง  
หมายเหตุ: ดวยแฟล็ก -R แฟล็กอื่นทั้งหมดยกเว้นแฟล็ก -n SessionName จะถูกข้าม เมนูการถ่ายโอนไฟล์ RESTART จะถูกแสดง
- s** ระบุการจัดการ SO/SI ในสภาวะแวดล้อม DBCS แฟล็ก -s ต้องถูกใช้กับหนึ่งในแฟล็กการแปล (-t, -T หรือ -J) ถ้า -t, -T หรือ -J ถูกละเว้น แฟล็ก -s ถูกละเว้น เมื่อแฟล็ก -s ถูกระบุ ฟังก์ชันต่อไปนี้จะถูกดำเนินการสำหรับการถ่ายโอนไฟล์:  
**Upload** อักขระ SO/SI ไม่ถูกแทรกในฟิลด์ DBCS  
**Download**  
อักขระ SO/SI ถูกแทนที่ด้วยอักขระควบคุม (0x1E/0x1F) ในฟิลด์ DBCS
- t** แฟล็ก -s ถูกสนับสนุนเฉพาะในสภาวะแวดล้อม DBCS  
ทำการแปล ASCII-EBCDIC สำหรับไฟล์ ถ้าการดาวน์โหลด คำสั่ง fxfer แปล EBCDIC เป็น ASCII ถ้าการอัปโหลด คำสั่ง fxfer แปล ASCII เป็น EBCDIC ภาษาถูกระบุ โดยคุณสมบัติภาษาในเซสชันโปรไฟล์ แฟล็ก -t ถือว่าไฟล์เป็นไฟล์ข้อความ อักขระ new-line เป็นตัวค้นบรรทัด  
เมื่อแฟล็ก -t ถูกใช้ในสภาวะแวดล้อม DBCS กับ DBCS อื่นที่สนับสนุนแฟล็ก การทำงานของแฟล็ก -t เปลี่ยนเป็นดังนี้:  
**Upload** แปล JISCH (ภาษาญี่ปุ่น) หรือ ASCII (ภาษาเกาหลี, จีนไต้หวัน) ไปเป็น EBCDIC แทรกอักขระ SO/SI ในฟิลด์ DBCS  
**Download**  
แปล EBCDIC JISCH (ภาษาญี่ปุ่น) หรือ ASCII (ภาษาเกาหลี, จีนไต้หวัน) ลบอักขระ SO/SI ออกจากฟิลด์ DBCS
- T** ทำการแปล ASCII-EBCDIC สำหรับไฟล์ disk operating system ลำดับอักขระ, CRLF, ที่ใช้เป็นตัวค้น และอักขระ disk operating system EOF (end-of-file) ถูกแทรกที่จุดสิ้นสุดของไฟล์ ที่ดาวน์โหลด ภาษาที่ถูกใช้สำหรับการแปล EBCDIC เป็น ASCII ถูกระบุ โดยคุณสมบัติภาษาในเซสชันโปรไฟล์ แฟล็ก -T ถูกใช้เพื่อแปลไฟล์ disk operating system  
หมายเหตุ: ถ้าไม่มีแฟล็ก -T, -t หรือ -J ถูกระบุ การถ่ายโอนไฟล์จะถือว่า ไม่มีการแปล และถ่ายโอนข้อมูลในรูปของไบนารี
- u** อัปโหลดไฟล์โดยการถ่ายโอนไฟล์จาก local system ไปที่โฮสต์ ถ้าแฟล็กนี้หรือแฟล็ก -d ไม่ถูกระบุ คุณสมบัติ File Transfer Direction ใน เซสชันโปรไฟล์จะกำหนดทิศทางของการถ่ายโอน
- v** เขียนสถานะของการถ่ายโอนไฟล์เป็นระยะๆ ไปที่ หน้าจอหรือไปที่ไฟล์สถานะที่ระบุโดยแฟล็ก -f สถานะรวมถึงจำนวนไบต์ที่ถ่ายโอน และเวลาที่ใช้ไป ตั้งแต่กระบวนการถ่ายโอนไฟล์เริ่มถ่ายโอนข้อมูล

## ไอเท็ม

-x HostLogin

### คำอธิบาย

ใช้ล็อกอิน ID ที่ระบุโดยตัวแปร HostLogin เพื่อล็อกอินไปที่โฮสต์ ผู้ใช้ถูกพร้อมต์เพื่อให้ป้อนรหัสผ่าน

สตริง HostLogin ประกอบด้วยโฮสต์ล็อกอิน ID, AUTOLOG node ID, และค่า AUTOLOG เป็นทางเลือกอื่น สตริงต้องไม่มีช่องว่างและต้องมี AUTOLOG node ID จัดรูปแบบสตริง AUTOLOG ดังนี้:

```
UserID,AutologNodeID[,Trace,Time . . . .]
```

ถ้าแฟล็ก -x ไม่ถูกระบุ ข้อมูลสำหรับ สตริง HostLogin ถูกนำมาจากเซสชันโปรไฟล์ ดังนี้:

- ถ้าโฮสต์ล็อกอิน ID ถูกเซตในเซสชันโปรไฟล์ คุณจะถูกรวมต์ขอรหัสผ่าน พารามิเตอร์ที่เหลือถูกเรียกมาจาก โปรไฟล์
- ถ้าโฮสต์ล็อกอิน ID ไม่ถูกเซตในโปรไฟล์ คุณจะได้รับพร้อมต์สำหรับทั้งโฮสต์ล็อกอินสตริงและรหัสผ่าน
- การตอบกลับของคุณไปที่พร้อมต์ จะเขียนทับโปรไฟล์พารามิเตอร์เสมอ ตัวอย่าง ถ้าเวลา AUTOLOG ถูกเซตในโปรไฟล์แต่คุณป้อนค่าอื่น ที่พร้อมต์ ค่าที่ป้อนที่พร้อมต์จะถูกใช้

ถ้าคุณละเว้นบางพารามิเตอร์ จากโฮสต์ล็อกอิน สตริง พารามิเตอร์จะถูกเรียกออกมาจากโปรไฟล์ ถ้ากำหนดไว้ ตัวอย่าง ถ้าคุณเซตพารามิเตอร์ AUTOLOG Node ID, AUTOLOG Trace, และ AUTOLOG Time ในโปรไฟล์ เฉพาะโฮสต์ล็อกอิน ID ต้องถูกป้อนที่พร้อมต์

กระบวนการถ่ายโอนไฟล์ล็อกอินกับโฮสต์ และสร้าง อิมูเลชันเซสชัน โดยใช้เซสชันโปรไฟล์ที่ระบุกับแฟล็ก -n เมื่อกระบวนการถูกล็อกอินสำเร็จ การถ่ายโอนไฟล์จะเริ่มขึ้น

พารามิเตอร์ File Transfer Wait Period ในเซสชันโปรไฟล์กำหนดระยะเวลาที่จะคงล็อกอินเซสชัน ไว้ การใช้พารามิเตอร์นี้ โฮสต์ล็อกอินเซสชันจะถูกรักษาไว้ เพื่อการถ่ายโอนไฟล์ที่มีต่อมา ไม่จำเป็นที่จะต้องมีการล็อกอินอีกครั้ง ระบุชุดโคดสำรองที่ใช้สำหรับการแปล ASCII-EBCDIC ถ้าแฟล็ก -X ถูกละเว้น ชุดโคดที่ระบุโดย โลแคลระบบจะถูกใช้ ชุดโคดต่อไปนี้ได้รับการสนับสนุน:

-X CodeSet

Default ใช้โคดเพจของ ASCII ระบบ

IBM-932 ใช้โคดเพจ IBM 932 สำหรับการแปลในสภาวะแวดล้อม DBCS

ISO8859-1

ใช้โคดเพจตัวอักษร ISO 8859-1 Latin หมายเลข 1

ISO8859-7

ใช้ตัวอักษร ISO 8859-7 Greek

ISO8859-9

ใช้ตัวอักษร ISO 8859-9 Turkish

IBM-eucJP

ใช้โคด IBM Extended UNIX สำหรับการแปลในสภาวะแวดล้อม ภาษาญี่ปุ่น

IBM-eucKR

ใช้โคด IBM Extended UNIX สำหรับการแปลในสภาวะแวดล้อม ภาษาเกาหลี

IBM-eucTW

ใช้โคด IBM Extended UNIX สำหรับการแปลในภาษาจีนไต้หวัน

## แฟล็กสำหรับคุณสมบัติไฟล์โฮสต์

แฟล็กต่อไปนี้จะระบุคุณสมบัติโฮสต์ไฟล์ และสามารถถูกใช้เฉพาะเพื่ออ็อปโหลดไฟล์ (โดยมีข้อยกเว้นของแฟล็ก -F ซึ่งสามารถถูกใช้เมื่อทำการดาวน์โหลดจากโฮสต์ VSE):

## ไอเอ็ม

**-B BlockSize**

### คำอธิบาย

ระบุขนาดบล็อกของชุดข้อมูลโฮสต์ แฟล็ก **-B** สามารถถูกใช้เฉพาะในสภาวะแวดล้อม MVS/TSO สำหรับชุดข้อมูลที่ต่อเนื่องกัน ตัวแปร *BlockSize* ต้องไม่เกิน ความจุของแตรีกเดียว แฟล็ก **-B** ถูกละเว้นถ้า ไฟล์กำลังถูกผนวก ค่าขนาดบล็อกเป็น 0 ทำให้เกิดข้อผิดพลาด

**-F**

ระบุเร็กคอร์ดความยาวคงที่ นี่เป็นดีฟอลต์ ถ้าไม่มีการระบุแฟล็ก **-V**, **-t**, **-T**, **-c** หรือ **-C** แฟล็ก **-F** ถูกละเว้นถ้า ไฟล์กำลังถูกผนวก

**-L LoglRecLength**

บน CICS หรือ VSE โฮสต์ หนึ่งในแฟล็กการแปล (**-t** หรือ **-T**) หรือหนึ่งในแฟล็ก CRLF (**-c** หรือ **-C**) ต้องถูกระบุตามด้วยแฟล็ก **-F** เนื่องจาก CICS และโปรแกรมการถ่ายโอนไฟล์ VSE โฮสต์ ไม่สนับสนุนเร็กคอร์ด ความยาวคงที่ การรวมกันของแฟล็ก **-F** และ แฟล็กการแปลทำให้โปรแกรมการถ่ายโอนเพิ่มช่องว่างให้กับเร็กคอร์ด ที่จุดสิ้นสุดของความยาวเร็กคอร์ดเชิงตรรกะ ค่าดีฟอลต์คือ 80

หมายเหตุ: ใช้แฟล็ก **-F** เมื่อดาวน์โหลดจากโฮสต์ VSE เพื่อป้องกันการลบช่องว่าง ท้ายไฟล์ที่แปล ระบุความยาวเร็กคอร์ดเชิงตรรกะเป็นไบต์ของโฮสต์ไฟล์ สำหรับ ไฟล์ใหม่ ดีฟอลต์คือ 80 สำหรับเร็กคอร์ดความยาวตัวแปร *LoglRecLength* คือขนาดสูงสุดของเร็กคอร์ด แฟล็ก **-L** ถูกละเว้นถ้า ไฟล์กำลังถูกผนวก ค่า *LoglRecLength* เป็น 0 ทำให้เกิดข้อผิดพลาด

เนื่องจาก MVS™ โอเวอร์เฮด จำนวนไบต์จริงที่เก็บในเร็กคอร์ดความยาว ตัวแปรบนโฮสต์ MVS/TSO น้อยกว่าค่าที่ระบุโดยตัวแปร *LoglRecLength* ไบต์

CICS และโปรแกรมการถ่ายโอนไฟล์โฮสต์ VSE ไม่สนับสนุนความยาว เร็กคอร์ดเชิงตรรกะ สำหรับการถ่ายโอนไปที่ หรือมาจาก CICS หรือโฮสต์ VSE แฟล็ก **-L** ต้องร่วมกับแฟล็ก **-F** การรวมกันของแฟล็ก **-F** และแฟล็ก **-L** ทำให้โปรแกรมการถ่ายโอนเพิ่มช่องว่างให้กับเร็กคอร์ด ที่จุดสิ้นสุดของความยาวเร็กคอร์ดเชิงตรรกะ ค่าดีฟอลต์คือ 80

หมายเหตุ: แฟล็ก **-L** จำเป็นต้องใช้ ถ้าความยาวเร็กคอร์ดมากกว่าความยาวเร็กคอร์ดดีฟอลต์คือ 80

**-S NumberUnits [ ,IncreaseUnits | ,IncreaseUnits,UnitType | ,,UnitType ]**

ระบุจำนวนของพื้นที่ที่จะถูกจัดสรรสำหรับชุดข้อมูล ต่อเนื่องใหม่บน TSO สำหรับไฟล์ MVS ขนาดใหญ่ขนาดบล็อกสูงสุดที่อนุญาต บนโฮสต์ถูกใช้เพื่อประกันว่าแตรีกดิสก์ทั้งหมดถูกบรรจุเต็ม แฟล็ก **-S** สามารถใช้ได้กับโฮสต์ MVS/TSO เท่านั้น

ตัวแปรต่อไปนี้สามารถใช้กับแฟล็ก **-S** ถ้าใช้ แฟล็กต้องถูกระบุตามลำดับที่กำหนดและแยกกันด้วยคอมมา ถ้าตัวแปรที่นำหน้าตัวแปรอื่นถูกละเว้น คอมมาต้องถูกใส่ไว้ เป็นตัวยึดตำแหน่ง ต้องมีช่องว่างระหว่างแฟล็ก **-S** และตัวแปร *NumberUnits* อย่างไรก็ตาม ช่องว่างจะมี ในตัวแปรสตริงไม่ได้

### *NumberUnits*

ระบุจำนวนหน่วยของพื้นที่ที่จะถูกเพิ่มเมื่อเริ่มต้น ค่า 0 คือค่าลบไม่สามารถใช้ได้กับตัวแปร *NumberUnits*

### *IncreaseUnits*

ระบุจำนวนหน่วยของพื้นที่ที่จะถูกเพิ่มให้กับชุดข้อมูล แต่ละครั้งที่พื้นที่ที่จัดสรรถูกบรรจุ (ทางเลือก)

*UnitType* กำหนดหน่วยของพื้นที่และ อาจเป็น T สำหรับ tracks, C สำหรับ cylinders หรือหมายเลขระบุ ขนาดบล็อกเฉลี่ย (เป็นไบต์) ของเร็กคอร์ดที่เขียนให้ชุดข้อมูล ถ้าตัวแปร *UnitType* ไม่ถูกระบุ ดีฟอลต์จะเป็นค่าที่ระบุโดยแฟล็ก **-B** ถ้าแฟล็ก **-B BlockSize** ไม่ถูกระบุ ค่าดีฟอลต์คือ 80

ต่อไปนี้เป็นกรรวมตัวแปรที่เป็นไปได้ ที่ใช้กับแฟล็ก **-S**:

**-S** *NumberUnits, IncreaseUnits, UnitType*

**-S** *NumberUnits, IncreaseUnits*

**-S** *NumberUnits*

**-S** *NumberUnits, UnitType*

**-U**

ระบุเร็กคอร์ดขอความยาวที่ไม่กำหนด แฟล็ก **-U** สามารถถูกใช้เฉพาะในสภาวะแวดล้อม MVS/TSO แฟล็ก **-U** ถูกละเว้นถ้า ไฟล์กำลังถูกผนวก

**-V**

ระบุเร็กคอร์ดของความยาวตัวแปร นี่เป็นดีฟอลต์ถ้าไม่ระบุแฟล็ก **-F** และมีกรระบุแฟล็ก **-t**, **-T**, **-c** หรือ **-C** แฟล็ก **-V** ถูกละเว้นถ้า ไฟล์กำลังถูกผนวก

แฟล็ก **-V** ไม่สนับสนุน โดย CICS หรือโปรแกรมการถ่ายโอนไฟล์โฮสต์ VSE เนื่องจากความยาว เร็กคอร์ดของตัวแปรเป็นค่าดีฟอลต์

## ตัวอย่าง

ตัวอย่างต่อไปนี้สมมุติว่าเซสชันโปรไฟล์สำหรับ เซสชัน a คือ:

```
Session type          DFT
Communication device   3270c0
Language               English (U.S.A.)
Host type              CMS
File transfer direction up
File transfer wait period 10
File transfer recovery time 30
```

โดยที่:

- โฮสต์คือ VM/CMS
- การเชื่อมต่อถูกสร้างขึ้นโดยใช้อุปกรณ์เชื่อมต่อ DFT 3270
- ทิศทางดีฟอลต์ของการถ่ายโอนไฟล์คืออัปโหลด (เมื่อต้องการใช้เซสชันโปรไฟล์ a สำหรับการดาวน์โหลดไฟล์ ผู้ใช้ต้องระบุแฟล็ก `-d` กับคำสั่ง `fxfer`)
- กระบวนการถ่ายโอนไฟล์คงการลือกอินเป็นเวลา 10 นาที
- ถ้าการถ่ายโอนถูกอินเตอร์รัปต์ กระบวนการจะพยายาม กู้คืนเป็นเวลา 30 นาที ก่อนทำการบันทึกข้อมูลในไฟล์ RESTART เพื่อการถ่ายโอนในภายหลัง
- ภาษาการแปลคือ U.S.A. ASCII-EBCDIC

1. เมื่อต้องการอัปโหลดไฟล์ `samplefile` (ในไดเรกทอรีปัจจุบัน) ไปที่โฮสต์และแปลเป็น EBCDIC โดยใช้ตารางการแปล U.S.A. ให้ป้อน:

```
fxfer -n a -t samplefile "test file a"
- -n กำหนดให้คำสั่ง fxfer ใช้เซสชัน a ในการถ่ายโอนไฟล์
- -t กำหนดให้คำสั่งทำการแปลโดยใช้อักขระ new-line
```

ข้อมูลที่แปลถูกนำไปไว้ใน `test file a` บนโฮสต์ เนื่องจากชื่อไฟล์ของโฮสต์มีช่องว่าง จำเป็นต้องมีเครื่องหมายคำพูดรอบชื่อไฟล์

2. เมื่อต้องการอัปโหลดไฟล์ `file2` เป็นโฮสต์ VM/CMS `test file b` ให้ป้อน:

```
fxfer -urv -L 132 -V -H CMS file2 "test file b"
- -u กำหนดให้คำสั่ง fxfer ทำการอัปโหลดไฟล์
- -H ระบุว่าชนิดของโฮสต์คือโฮสต์ VM/CMS ถ้ามีไฟล์ปลายทางอยู่ ไฟล์จะถูกแทนที่ (เนื่องจากการระบุแฟล็ก -r) โดยไฟล์ที่ถ่ายโอน
- -v ทำให้ fxfer แสดงจำนวนไบต์ ที่ถ่ายโอนและเวลาที่ใช้ไป เอาต์พุตสถานะหรือวินิจฉัย ถูกแสดง บนเทอร์มินัล
- ถ้าโฮสต์ไฟล์ไม่มีอยู่ ความยาวเรีกคอร์ดเชิงตรรกะสูงสุดของโฮสต์ไฟล์ถูกเซตเป็น 132 ไบต์ (-L)
- รูปแบบเรีกคอร์ดโฮสต์ไฟล์เป็นตัวแปร (แฟล็ก -V) ไม่มีการแปล
```

3. เมื่อต้องการอัปโหลดจาก `subshell` ของอิมูเลเตอร์เซสชัน a ไฟล์ `local system /etc/motd` ไปที่โฮสต์ไฟล์ CICS `motdfile` พร้อมกับการแปลและการเพิ่มช่องว่าง ให้ป้อน:

```
fxfer -utFH CICS -I ")This is a comment" /etc/motd "motdfile"
- -u กำหนดให้คำสั่งอัปโหลดไฟล์
- -t ทำให้มีการแปลจาก ASCII เป็น EBCDIC
```

- F ทำให้โปรแกรมการถ่ายโอนเพิ่มช่องว่างกับไฟล์ที่อัปโหลด เป็นคอลัมน์ 80 (ความยาวเร็กคอร์ดดีฟอลต์) เมื่อต้องการเปลี่ยนดีฟอลต์คอลัมน์ให้ใช้แฟล็ก -L กับความยาว เร็กคอร์ดอื่น (คอลัมน์)
- H ระบุโฮสต์เป็นชนิด CICS
- I ระบุว่าค่า *InputField* ถูกเพิ่มให้กับคำสั่ง **IND\$FILE**

ในตัวอย่างนี้, "This is a comment" เป็น ฟิลด์หมายเหตุของโฮสต์

เมื่อต้องการอัปโหลดหรือดาวน์โหลดไฟล์ด้วยคำสั่ง **fxfer** ไปที่หรือจาก สภาวะแวดล้อม TSO อื่นที่ไม่ใช่สภาวะแวดล้อมปัจจุบันของคุณ คุณต้องได้รับการอนุญาตสำหรับสภาวะแวดล้อมอื่น คุณต้องระบุไฟล์ (หรือชุดข้อมูล) อย่างสมบูรณ์ภายในเครื่องหมายคำพูดเดี่ยว (') ตามด้วยเครื่องหมาย คำพูดคู่ (" ")

- ตัวอย่าง เพื่อต้องการอัปโหลดไฟล์ **newfile** ไปที่สภาวะแวดล้อม TSO ซึ่งชื่อที่ระบุแบบสมบูรณ์คือ **sys4.parmlib.samplefile** ให้ป้อน:

```
fxfer -urtvH TSO 'newfile' "sys4.parmlib.samplefile"
```

- u กำหนดให้คำสั่งอัปโหลดไฟล์
- ถ้าไฟล์ **sys4.parmlib.samplefile** มีอยู่จะถูกแทนที่ (แฟล็ก **-r**) ด้วยเนื้อหาที่แปลของไฟล์ **newfile** (แฟล็ก **-t**)
- v กำหนดให้คำสั่ง **fxfer** เขียนสถานะ การถ่ายโอนไฟล์ไปที่หน้าจอโลคัลทุกๆ สองสามวินาที
- H กำหนดให้คำสั่ง **fxfer** ที่โฮสต์ เป็นโฮสต์ MVS/TSO

หมายเหตุ: ตัวอย่างนี้ถือว่า คำสั่ง **fxfer** ถูกเรียกใช้จากเซลล์ย่อยของเซสชัน ที่สร้าง (ใช้คำสั่ง **e789** เพื่อสร้างเซสชัน)

- เมื่อต้องการดาวน์โหลดไฟล์ **spfuser.test** จากโฮสต์ MVS/TSO ไปที่ local system ให้ป้อน:

```
fxfer -n a -d -r -H TSO spfuser.test samplefile1
```

- n กำหนดให้คำสั่ง **fxfer** ใช้เซสชัน a ในการถ่ายโอนไฟล์ ถ้าเซสชัน a ยังไม่ได้ถูกสร้างขึ้น คำสั่งจะพยายามล็อกอินอัตโนมัติ เนื่องจากไม่มีการระบุ ล็อกอิน ID คำสั่ง **fxfer** จะตรวจสอบเซสชันโปรไฟล์สำหรับ ล็อกอิน ID ถ้าไม่มีการระบุ ผู้ใช้จะได้รับพร้อมท์เพื่อขอ ล็อกอิน ID และรหัสผ่าน
- d เขียนทับทิศทางการถ่ายโอนไฟล์ดีฟอลต์ของการอัปโหลด
- ถ้าไฟล์ **samplefile1** มีอยู่แล้ว ไฟล์จะถูกแทนที่ (แฟล็ก **-r**) ด้วยไฟล์ที่ดาวน์โหลดจากโฮสต์
- H กำหนดให้คำสั่ง **fxfer** ที่โฮสต์ เป็นโฮสต์ MVS/TSO แทน VM/CMS (ดีฟอลต์จากเซสชันโปรไฟล์)

ไฟล์ที่ถ่ายโอนถูกกำหนดไว้ที่ไฟล์ **samplefile1** บน local system การถ่ายโอนไฟล์ถูกดำเนินการพร้อมกัน

- เมื่อต้องการดาวน์โหลดโฮสต์ VM/CMS **test file a** และผนวกกับไฟล์ local system **mydir/samplefile** โดยใช้เซสชันโปรไฟล์ a และการล็อกอินอัตโนมัติ ให้ป้อน:

```
fxfer -n a -dat -q -f status.out
-x laura,vm1,trace "test file a" mydir/samplefile
```

- n กำหนดให้คำสั่ง **fxfer** ให้ใช้เซสชัน โปรไฟล์ a ในการถ่ายโอนไฟล์
- x มีโฮสต์ล็อกอิน ID คำสั่ง **fxfer** ขึ้นแรกตรวจสอบเพื่อดูว่าเซสชันถูกสร้างบน local system หรือไม่ ถ้ามี คำสั่งจะถ่ายโอนไฟล์ผ่านเซสชันที่มีอยู่ ถ้าเซสชัน a ไม่ถูกสร้าง คำสั่ง **fxfer** จะดำเนินการ ล็อกอินอัตโนมัติโดยใช้โฮสต์ล็อกอิน ID **laura** และ สคริปต์ **AUTOLOG vm1** และติดตามกิจกรรมล็อกอิน ผู้ใช้จะได้รับพร้อมท์เพื่อขอรหัสผ่าน คำสั่งถ่ายโอนไฟล์

- **-dat** กำหนดให้ **fxfer** คำสั่งดาวน์โหลดไฟล์ (แฟล็ก **-d**) แปลข้อมูลจาก EBCDIC เป็น ASCII (แฟล็ก **-t**) โดยใช้ตารางการแปล U.S.A. (ที่กำหนดในเซสชันโปรไฟล์) และผนวก (แฟล็ก **-a**) ไฟล์ที่แปลไปที่ไฟล์ **mydir/samplefile** บน local system ถ้าไฟล์ **mydir/samplefile** ไม่มีอยู่ คำสั่ง **fxfer** จะละเว้นแฟล็ก **-a** และสร้างไฟล์
- เอาต์พุตสถานะหรือการวินิจฉัยถูกกำหนดไว้ในไฟล์ **status.out** ในไดเรกทอรีโลคัลปัจจุบัน (แฟล็ก **-f**)
- **-q** กำหนดให้คำสั่ง **fxfer** ทำการถ่ายโอนไฟล์ไม่พร้อมกัน

เมื่อผู้ใช้ป้อนรหัสผ่าน พร้อมต์ถูกส่งกลับและการถ่ายโอนไฟล์ ถูกดำเนินการอยู่เบื้องหลัง

เมื่อต้องการคิวการถ่ายโอนไฟล์ อื่นที่จะถูกดำเนินการโดยกระบวนการถ่ายโอนไฟล์เดียวกัน ให้ป้อน:

```
fxfer -n a -daq -f status.out "test file b"
mydir/samplefile
```

- **-n** กำหนดให้คำสั่ง **fxfer** ใช้เซสชัน **a** ในการถ่ายโอนไฟล์ เนื่องจากเซสชัน **a** ได้ถูกสร้างขึ้นโดยคำสั่งก่อนหน้านี้ คำสั่ง **fxfer** ไม่จำเป็นต้องล็อกอินไปที่โฮสต์อีกครั้ง
- **-d** กำหนดให้คำสั่งดาวน์โหลดไฟล์จากโฮสต์
- **-a** กำหนดให้คำสั่งทำการผนวกโฮสต์ไฟล์ **test file b** กับไฟล์ **mydir/samplefile** บน local system
- **-q** กำหนดให้คำสั่ง **fxfer** ทำการถ่ายโอนไฟล์ไม่พร้อมกัน

คำสั่ง **fxfer** ทำต่อการส่งข้อมูลสถานะ ไปที่ไฟล์ **status.out** บน local system (แฟล็ก **-f**)

#### Notes:

- ถ้าข้อความสำหรับคำสั่ง **fxfer** ขยายเกินค่าจำกัดของ ข้อความจะถูกตัดโดยอัตโนมัติไปที่ บรรทัดถัดไป การกดคีย์ Enter เพื่อตัดข้อความจะทำให้เกิดข้อผิดพลาด
  - การพยายามเริ่มการถ่ายโอนไฟล์พร้อมกัน เมื่อมีการถ่ายโอนที่ไม่พร้อมกันในคิวทำให้เกิดข้อผิดพลาด
  - ผู้ใช้จะไม่ได้รับพร้อมต์เพื่อขอล็อกอิน ID หรือ รหัสผ่าน トラบที่เซสชันยังคงรันอยู่และกระบวนการ **dfxfer** ยังคงล็อกอินกับโฮสต์ เวลาที่ กระบวนการยังคงล็อกอินอยู่ ถูกกำหนดโดย File Transfer Wait Period ใน เซสชันโปรไฟล์
7. เมื่อต้องการเริ่มการถ่ายโอนไฟล์ที่อินเทอร์รัปต์จาก อีเมลเตอร์ subshell ให้ป้อน:

```
fxfer -R
```

**-R** กำหนดให้คำสั่ง **fxfer** ใช้ข้อมูลที่บันทึกในหนึ่งในไฟล์ **RESTART** เพื่อเรียกใช้งาน การถ่ายโอนไฟล์ ไฟล์ **RESTART** คือไฟล์รีสตาร์ททางตรง **\$HOME/x\_fxfer.r** หรือไฟล์รีสตาร์ททางอ้อม **\$HOME/i\_fxfer.r** ถ้ามีการระบุแฟล็ก **-R** ร่วมกับแฟล็กการถ่ายโอนไฟล์ อื่น แฟล็กอื่นจะถูกละเว้น เมนูการถ่ายโอนไฟล์ **RESTART** ถูกแสดง การใช้เมนูนี้ กำหนดให้คำสั่ง **fxfer** ถ่ายโอนไฟล์ที่อินเทอร์รัปต์

8. เมื่อต้องการรีสตาร์ทการถ่ายโอนไฟล์จากบรรทัดคำสั่ง แทนจากอีเมลเตอร์ subshell ให้ป้อน:

```
fxfer -R -n a
```

แฟล็ก **-n** กำหนด ให้คำสั่ง **fxfer** ใช้เซสชัน **a** เพื่อดำเนินการถ่ายโอนที่รีสตาร์ท

ไฟล์

## ไอเท็ม

/usr/bin/xfxfer  
/usr/bin/dfxfer  
\$HOME/i\_fxfer.r  
\$HOME/x\_fxfer.r  
\$HOME/hconerrors  
/usr/lib/libfxfer.a

## คำอธิบาย

มีคำสั่ง fxfxfer  
มีกระบวนการ dxfxfer  
มีข้อมูล RESTART สำหรับคิวการลือกอินอัตโนมัติ ไฟล์ชั่วคราว ที่สร้างโดยคำสั่ง fxfxfer  
มีข้อมูล RESTART สำหรับคิวการลือกอินด้วยตัวเอง ไฟล์ชั่วคราว ที่สร้างโดยคำสั่ง fxfxfer  
มีเอาต์พุตการวินิจฉัย HCON และและการถ่ายโอนไฟล์ ไฟล์ชั่วคราว ที่สร้างโดยคำสั่ง HCON  
มีไลบรารีสำหรับการถ่ายโอนไฟล์ทางโปรแกรม

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง smit

---

## g

คำสั่ง AIX ต่อไปนี้เริ่มต้นด้วยตัวอักษร g

---

### gated Daemon

#### วัตถุประสงค์

จัดให้มีฟังก์ชันการจัดการเส้นทางแก่เครือข่ายสำหรับโปรโตคอล RIP, RIPng, EGP, BGP, BGP4+, HELLO, IS-IS, ICMP, ICMPv6 และ SNMP

หมายเหตุ: ใช้คำสั่ง SRC เพื่อควบคุม gated daemon จากบรรทัดรับคำสั่ง ใช้ไฟล์ rc.tcpip เพื่อสตาร์ท daemon ด้วยการเริ่มทำงานของระบบแต่ละระบบ

#### ไวยากรณ์

```
/usr/sbin/gated [-c] [-C] [-n] [-N] [-t TraceOptions] [-f ConfigFile] [TraceFile]
```

#### คำอธิบาย

/usr/sbin/gated daemon จัดการ หลายโปรโตคอลการจัดการเส้นทางและแทนที่ routed และ daemon การจัดการเส้นทางใดๆ ที่ใช้โปรโตคอลการจัดการเส้นทาง (HELLO) ขณะนี้ /usr/sbin/gated daemon จัดการ Routing Information Protocol (RIP), Routing Information Protocol Next Generation (RIPng), Exterior Gateway Protocol (EGP), Border Gateway Protocol (BGP) and BGP4+, Defense Communications Network Local-Network Protocol (HELLO) และ Open Shortest Path First (OSPF), Intermediate System to Intermediate System (IS-IS) และโปรโตคอลการจัดการเส้นทาง Internet Control Message Protocol (ICMP)/Router Discovery นอกจากนี้ gated daemon สนับสนุน Simple Network Management Protocol (SNMP) กระบวนการ gated สามารถกำหนดค่าเพื่อให้ดำเนินการโปรโตคอลทั้งหมดหรือโปรโตคอลใดๆ ร่วมกัน ไฟล์คอนฟิกูเรชันดีฟอลต์สำหรับ gated daemon คือไฟล์ /etc/gated.conf gated daemon เก็บ ID กระบวนการของตนในไฟล์ /etc/gated.pid

หมายเหตุ: อาจเกิดผลลัพธ์ที่ไม่คาดคิดเมื่อ gated และ routed daemons ถูกรันด้วยกันบน โฮสต์เดียวกัน

หาก ระบุไฟล์การติดตามบนบรรทัดคำสั่ง หรือไม่ระบุแฟล็กการติดตาม gated daemon จะแยกจากเทอร์มินัลและรัน ในแบบเบื้องหลัง หากระบุแฟล็กการติดตามโดยไม่ระบุไฟล์ การติดตาม gated จะถือว่าการติดตามกำหนดเป็น stderr และยังคงอยู่รันแบบเบื้องหน้า

หมายเหตุ: โปรโตคอลการจัดการเส้นทาง IS-IS ไม่สามารถรันบนเคอร์เนล 64 บิต

#### สัญญาณ

เซิร์ฟเวอร์ gated ดำเนินการ ต่อไปนี้เมื่อคุณใช้คำสั่ง kill เพื่อส่งสัญญาณ

ไอเท็ม SIGHUP	<p>คำอธิบาย</p> <p>อาการกำหนดค่าใหม่</p> <p>SIGHUP ทำให้ gated ต้องอ่านไฟล์คอนฟิกูเรชันใหม่ อันดับแรก gated daemon ทำการล้างค่าโครงสร้างนโยบายการจัดสรรทั้งหมด เพียร์ BGP และ EGP ทั้งหมดถูกแฟล็กเพื่อการลบ และไฟล์คอนฟิกูเรชัน ถูกวิเคราะห์ใหม่</p> <p>หากการวิเคราะห์ใหม่สำเร็จ เพียร์ BGP และ EGP ใดๆ ที่ไม่อยู่ในการกำหนดค่าจะถูกปิดการทำงานและเพียร์ใหม่ถูกเริ่มทำงาน gated daemon พยายามกำหนดว่าการเปลี่ยนแปลงที่ทำกับเพียร์ที่มีอยู่แล้วจำเป็นต้องปิดการทำงาน และรีสตาร์ทหรือไม่</p> <p>หมายเหตุ: การกำหนดค่าใหม่ถูกปิดใช้งานเมื่อ OSPF (Open Shortest Path First) ถูกเปิดใช้งาน</p>
SIGINT	<p>สแน็ปช็อตของสถานะปัจจุบัน</p> <p>สถานะปัจจุบัน ของงาน gated ตัวจับเวลา โปรโตคอล และตารางทั้งหมดถูกเขียนไปยัง /var/tmp/gated_dump</p>
SIGTERM	<p>ซึ่งดำเนินการโดยการ fork กระบวนการย่อยเพื่อตีพิมพ์ข้อมูลตารางเพื่อมีให้มีผลกระทบต่อฟังก์ชันการจัดเส้นทางของ gated daemon ปิดระบบแบบเรียบรอย</p> <p>เมื่อได้รับ สัญญาณ SIGTERM gated daemon จะพยายามปิดระบบ อย่างเรียบรอย งานและโปรโตคอลทั้งหมดถูกขอให้ปิดระบบ ส่วนใหญ่จะสิ้นสุดการทำงาน ในทันที ยกเว้นสำหรับเพียร์ EGP ที่ต้องรอค้ายืนยัน ซึ่ง อาจจำเป็นต้องทำซ้ำ SIGTERM ครั้งหรือสองครั้ง หากกระบวนการนี้ใช้เวลานานเกินไป</p> <p>เส้นทางโปรโตคอล ทั้งหมดถูกลบออกจากตารางการจัดเส้นทางของเคอร์เนลเมื่อได้รับ SIGTERM เส้นทางอินเตอร์เฟซ เส้นทางที่มี RTF_STATIC ถูกตั้งค่า (จากคำสั่ง route ที่สับบสนุน) และเส้นทาง สเตติกที่ระบุ retain จะยังคงอยู่ ในการยุติการทำงาน gated daemon ที่มีผลต่อเส้นทางภายนอก ให้ใช้สัญญาณ SIGKILL หรือ SIGQUIT (ซึ่งทำให้เกิดคอร์ตัมพ์)</p>
SIGUSR1	<p>สลับการติดตาม</p> <p>เมื่อได้รับสัญญาณ SIGUSR1 gated daemon จะปิดไฟล์ การติดตาม SIGUSR1 ที่ตามมาจะทำให้ ถูกเปิดใหม่ ซึ่งจะอนุญาตให้ไฟล์ถูกย้ายตามปกติ</p> <p>หมายเหตุ: คุณไม่สามารถใช้สัญญาณ SIGUSR1 หากไม่ได้รับไฟล์การติดตาม หรือการติดตามถูกดำเนินการไปยัง stderr</p>
SIGUSR2	<p>ตรวจสอบการเปลี่ยนแปลงอินเตอร์เฟซ</p> <p>เมื่อ ได้รับสัญญาณ SIGUSR2 gated daemon จะสแกน รายการอินเตอร์เฟซเคอร์เนลอีกครั้งเพื่อหาการเปลี่ยนแปลง</p>

## gated และ snmpd Daemons

gated daemon ถูกกำหนดค่า ภายในให้เป็นโปรโตคอลเพียร์ SNMP multiplexing (SMUX) หรือพรีอิกซ์เอเจนต์ ของ snmpd daemon สำหรับข้อมูลเพิ่มเติม โปรดอ้างอิง "การประมวลผล SNMP daemon" ใน *Networks and communication management*

## การจัดการ gated Daemon ด้วย System Resource Controller

gated daemon สามารถถูกควบคุม โดย System Resource Controller (SRC) gated daemon เป็นสมาชิกของกลุ่มระบบ SRC tcpip daemon นี้ถูกปิดใช้งานตามค่าดีฟอลต์ และสามารถจัดการได้ด้วยคำสั่ง SRC ต่อไปนี้:

ไอเท็ม	คำอธิบาย
startsrc	เริ่มต้นระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย
stopsrc	หยุดทำงานระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย
refresh	ทำให้ระบบย่อยหรือกลุ่มของระบบย่อยอ่าน ไฟล์คอนฟิกูเรชันอีกครั้ง
lssrc	รับข้อมูลสถานะของระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย

หมายเหตุ: ในการเริ่มทำงานตั้งต้นจากคำสั่ง startsrc gated daemon ไม่เริ่มทำการตอบกลับคำสั่ง SRC อื่นๆ จนกว่าการกำหนดค่าเริ่มต้น gated ทั้งหมดจะเสร็จสมบูรณ์ ไฟล์ /etc/gated.conf ขนาดใหญ่มากอาจต้องใช้เวลาลักพักเพื่อวิเคราะห์ค่าให้เสร็จสมบูรณ์

## แฟล็ก

### ไอเท็ม

- c คำอธิบาย  
ระบุการวิเคราะห์ค่าของไฟล์คอนฟิกูเรชัน เพื่อดูข้อมูลผิดพลาดทางไวยากรณ์ หลังจากที่ถูก `gated` daemon ออกจากการทำงาน หากไม่มีข้อมูลผิดพลาด `gated` daemon ไม่พิมพ์ไฟล์ลงในไฟล์ `/var/tmp/gated_dump` แฟล็ก `-c` แสดงนัยแฟล็ก `-tgeneral, kernel, nostamp` ถ้าแฟล็ก `-c` ถูกระบุ `gated` daemon ซ้ำคำสั่งย่อย `traceoption` และ `tracefile` ทั้งหมดในไฟล์คอนฟิกูเรชัน
- C ระบุว่าไฟล์คอนฟิกูเรชันถูกวิเคราะห์ค่าเพื่อหาข้อมูลผิดพลาดทางไวยากรณ์เท่านั้น `gated` daemon ที่มีอยู่ด้วยสถานะ 1 หากพบข้อผิดพลาดใดๆ และด้วยสถานะ 0 หากไม่พบ แฟล็ก `-C` แสดงนัยแฟล็ก `-tnostamp`
- f ConfigFile ระบุไฟล์คอนฟิกูเรชันทางเลือก โดยค่าดีฟอลต์ `gated` daemon ใช้ไฟล์ `/etc/gated.conf`
- n ระบุว่า `gated` daemon จะไม่แก้ไข ตารางการกำหนดเส้นทางของเคอร์เนล นี้ใช้สำหรับการทดสอบการกำหนดค่า `gated` ด้วยข้อมูลการกำหนดเส้นทางจริง
- N ระบุว่า `gated` daemon จะไม่ `daemonize` โดยปกติ หากไม่ระบุการติดตามไปยัง `stderr` และ ID กระบวนการพาเรนต์ไม่เป็น 1 `gated` daemon จะ `daemonize` แฟล็กนี้อินนุญาติให้ใช้วิธีที่คล้ายกับ `/etc/inittab` ในการเรียกใช้ `gated` daemon ที่ไม่มี ID กระบวนการค่า 1
- tTraceOptions ระบุว่าอ็อพชันการติดตามใดที่ถูกเปิดใช้งานเมื่อเริ่มทำงานระบบ เมื่อใช้โดยไม่มีตัวแปร `TraceOptions` แฟล็กนี้จะเริ่มทำงาน อ็อพชันการติดตาม `general` ค้นแต่ละอ็อพชันการติดตาม ด้วยเครื่องหมายจุลภาค อย่าแทรกช่องว่างระหว่างแฟล็กและ อ็อพชันการติดตามแรก

แฟล็ก `-t` ต้องใช้เพื่อติดตามเหตุการณ์ที่เกิดขึ้นก่อนไฟล์ `/etc/gated.conf` จะถูกวิเคราะห์ค่า เช่นการพิจารณาการกำหนดค่าอินเตอร์เฟซ และการอ่านเส้นทางจากเคอร์เนล

บทความไฟล์ `gated.conf` จะอธิบายอ็อพชันการติดตาม ที่มีอยู่

## ตัวอย่าง

1. ในการเริ่มทำงาน `gated` daemon ให้ป้อนคำสั่งที่คล้ายกับต่อไปนี้:

```
startsrc -s gated -a "-tall /var/tmp/gated.log"
```

คำสั่งเริ่มทำงาน `gated` daemon และบันทึกการทำงาน ข้อความ ข้อความจะถูกส่งไปยังไฟล์ `/var/tmp/gated.log`

2. ในการหยุดทำงาน `gated` daemon ตามปกติให้ป้อน:

```
stopsrc -s gated
```

คำสั่งนี้หยุด `daemon` แฟล็ก `-s` ระบุว่า ระบบย่อยที่ตามมาต้องหยุดทำงาน

3. ในการรับค่าสถานะอย่างย่อจาก `gated` daemon ให้ป้อน:

```
lssrc -s gated
```

คำสั่งนี้ ส่งกลับชื่อของ `daemon` ID กระบวนการของ `daemon` และสถานะของ `daemon` (แ็คทีฟหรือไม่แ็คทีฟ)

## ไฟล์

ไอเท็ม  
/etc/gated.pid  
/var/tmp/gated\_dump  
/var/tmp/gated.log

คำอธิบาย  
มี ID กระบวนการ gated  
ระบุดัมพ์ไฟล์หน่วยความจำ  
ระบุไฟล์บันทึกการทำงานสำหรับข้อความแสดงความผิดพลาด

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง gdc”

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง kill

คำสั่ง gated.conf

วิธีกำหนดค่า gated daemon

---

## คำสั่ง gdc

### วัตถุประสงค์

จัดให้มีส่วนการติดต่อกับผู้ใช้ในการดำเนินงานสำหรับ gated

### ไวยากรณ์

`gdc [-q] [-n] [-c coresize] [-f filesize] [-m datasize] [-s stacksize] [-t seconds] Subcommands`

### คำอธิบาย

คำสั่ง `gdc` จัดให้มีส่วนติดต่อกับผู้ใช้สำหรับการดำเนินการของ daemon การจัดเส้นทาง `gated` โดยให้การสนับสนุนสำหรับ:

- การเริ่มทำงานและการหยุดทำงาน daemon
- การนำส่งสัญญาณเพื่อจัดการ daemon ขณะกำลังดำเนินการ
- การดูแลรักษาและการตรวจสอบไวยากรณ์ของไฟล์คอนฟิกูเรชัน
- สำหรับการใช้งานจริง และการลบการดัมพ์สถานะ และคอร์ดัมพ์

คำสั่ง `gdc` สามารถไว้วางใจในการกำหนดสถานการณ์ของ `gated` และสร้างสถานะการออกที่ไว้วางใจได้ เมื่อเกิดข้อผิดพลาด ทำให้มีประโยชน์ในการใช้ในเชลล์สคริปต์ ซึ่งจะจัดการ `gated` คำสั่งที่เรียกใช้ โดยใช้ `gdc` และอาจมีข้อความแสดงความผิดพลาดสร้างขึ้น โดยการเรียกใช้ของคำสั่งเหล่านี้ จดบันทึกการทำงานผ่านโปรแกรมอำนวยความสะดวก `syslogd` เดียวกันซึ่ง `gated` เอง ใช้ การจัดให้มีหลักฐานการตรวจสอบของการดำเนินการที่ทำบน daemon

### แฟล็ก

ไอเท็ม	คำอธิบาย
-n	รันโดยไม่เปลี่ยนตารางการส่งต่อเคอร์เนล นี่เป็น ประโยชน์สำหรับการทดสอบ และเมื่อดำเนินการเป็นเซิร์ฟเวอร์เส้นทางซึ่ง ไม่ทำ การส่งต่อ
-q	รันแบบเงียบ โดยข้อความนี้ข้อความที่ให้อ้อมูลซึ่ง โดยปกติจะถูกพิมพ์ไปยังเอาต์พุตมาตรฐานจะไม่ถูกแสดงและข้อความแสดง ความผิดพลาดจะถูกบันทึกการทำงานด้วย syslogd แทน ถูกพิมพ์ไปยังเอาต์พุตข้อผิดพลาดมาตรฐาน สิ่งนี้ช่วยอำนวยความสะดวก เมื่อ รัน gdc จากเซลล์สคริปต์
-tseconds	ระบุเวลาเป็นวินาทีที่ gdc จะรอ gated ดำเนินงานที่ต้องการเสร็จเรียบร้อย โดยปกติในการจบการทำงานและการเริ่มทำงาน โดยค่า ดีฟอลต์ ค่านี้ ถูกตั้งค่าเป็น 10 วินาที
-c coresize	ตั้งค่าขนาดสูงสุดของการคอร์ตัมพี gated ที่เริ่มต้น ด้วยการสร้าง gdc นี้เป็นประโยชน์ต่อระบบ ที่ขนาดคอร์ตัมพีสูงสุดดีฟอลต์เล็ก เกินไปสำหรับ gated ที่สร้างคอร์ตัมพีทั้งหมดของข้อผิดพลาด
-f filesize	ตั้งค่าขนาดไฟล์สูงสุดที่ gated ที่เริ่มทำงาน ด้วย gdc จะสร้าง เป็นประโยชน์ต่อระบบ ที่ขนาดคอร์ตัมพีสูงสุดดีฟอลต์เล็กเกินไป สำหรับ gated ที่สร้างคอร์ตัมพีสถานะทั้งหมดเมื่อร้องขอ
-m datasize	ตั้งค่าขนาดสูงสุดของเซกเมนต์ข้อมูลของ gated ที่เริ่มทำงาน ด้วย gdc เป็นประโยชน์ต่อระบบเมื่อขนาด เซกเมนต์ข้อมูลดีฟอลต์เล็ก เกินไปสำหรับ gated ที่จะ รัน
-s stacksize	ตั้งค่าขนาดสูงสุดของสแต็กของ gated ที่เริ่มทำงาน ด้วย gdc เป็นประโยชน์ต่อระบบเมื่อขนาด สแต็กดีฟอลต์เล็กเกินไปสำหรับ gated ที่จะ รัน

## คำสั่งย่อ

คำสั่งย่อต่อไปนี้ทำให้มีสัญญาณส่งไปยัง gated ด้วยวัตถุประสงค์ต่างๆ:

ไอเท็ม	คำอธิบาย
COREDUMP dump interface	ส่งสัญญาณยกเลิกไปยัง gated ทำให้ ยุติคอร์ตัมพี สัญญาณ gated เพื่อตีพิมพ์สถานะปัจจุบัน ลงในไฟล์ /var/tmp/gated_dump สัญญาณ gated เพื่อตรวจสอบการกำหนดค่า อินเทอร์เน็ตอีกครั้ง โดยปกติ gated ทำเป็นระยะ ในเหตุการณ์ใดๆ แต่สามารถใช้โปรแกรมอำนวยความสะดวกเพื่อบังคับให้ daemon ตรวจสอบสถานะอินเทอร์เน็ตในทันทีเมื่อ ทราบมีการการเปลี่ยนแปลงเกิดขึ้น
KILL reconfig term	ทำให้ gated ยุติการทำงานอย่างประหลาด สัญญาณ gated เพื่ออ่านไฟล์คอนฟิกูเรชัน อีกครั้ง กำหนดค่าสถานะปัจจุบันตามความเหมาะสมอีกครั้ง สัญญาณ gated เพื่อยุติการทำงานหลักการทำงาน โปรโตคอลการแจ้งเตือนทางดำเนินการทั้งหมดอย่างเรียบ ร้อย การเรียกใช้งานคำสั่งนี้ ครั้งที่สองทำให้ gated ต้องยุติการทำงานแม้ว่า จะมีบางโปรโตคอลที่ยังไม่หยุดทำงาน ให้เรียบร้อย
toggletrace	ทำให้การติดตามถูกหยุดทำงานชั่วคราว และหาก gated กำลัง ทำการติดตามไปยังไฟล์ในขณะนี้ ให้ปิดไฟล์การติด ตาม หากการติดตาม gated ขณะนี้ถูกหยุดทำงานชั่วคราว คำสั่งย่อนี้จะทำให้ไฟล์การติดตามถูกเปิด อีกครั้งและ เริ่มต้นการติดตาม นี่เป็นประโยชน์สำหรับการย้ายไฟล์การติดตาม

คำสั่งย่อต่อไปนี้ดำเนินการที่เกี่ยวกับไฟล์คอนฟิกูเรชัน:

ไอเท็ม	คำอธิบาย
checkconf	ตรวจสอบ /etc/gated.conf เพื่อหาข้อผิดพลาดทางไวยากรณ์ นี่เป็นประโยชน์เมื่อทำเสร็จหลังการเปลี่ยนแปลงใน ไฟล์คอนฟิกูเรชัน แต่ ก่อนการส่งสัญญาณ reconfig ไปยัง gated ที่กำลังรันขณะนี้ เพื่อให้แน่ใจว่าไม่มีข้อผิดพลาด ใดๆ ในการกำหนดค่าที่อาจทำให้ gated ที่กำลังรันอยู่ต้อง ยุติการกำหนดค่าใหม่ เมื่อใช้คำสั่งนี้ gdc จะสร้างข้อ ความให้อ้อมูลที่ระบุว่า มีข้อผิดพลาดในวิเคราะหค่าหรือไม่ และหากมีให้บันทึกเอาต์พุตข้อผิดพลาดใน ไฟล์เพื่อ ใช้ในการตรวจสอบ
checknew newconf	เหมือนกับ checkconf ยกเว้นว่าไฟล์คอนฟิกูเรชัน new /etc/gated.conf+ ถูกตรวจสอบแทน ย้ายไฟล์ /etc/gated.conf+ ไปยัง ตำแหน่งเป็น /etc/gated.conf โดยคงเวอร์ชัน เกากว่าของไฟล์ไว้ตั้งอธิบายด้ำน บน gdc จะ ปฏิเสธในการทำสิ่งใดๆ เมื่อกำหนดคำสั่งนี้ หากไฟล์คอนฟิกูเรชันใหม่ ไม่มีอยู่หรือมีจะนั้นจะค้นหาสิ่ง ที่สงสัย
backout	ย้ายไฟล์คอนฟิกูเรชันในทิศทาง newer มีผลให้ย้ายไฟล์คอนฟิกูเรชันเก่า ไปยัง /etc/gated.conf คำสั่งจะปฏิเสธ การดำเนินการหาก /etc/gated.conf- ไม่มีอยู่ หรือมีความยาวเป็นศูนย์ หรือหากการดำเนินการนั้นจะลบ ไฟล์ /etc/gated.conf+ ที่มีความยาวไม่เป็นศูนย์ที่มีอยู่
BACKOUT modeconf createconf	ดำเนินการ backout แม้ว่า /etc/gated.conf+ มีอยู่ และความยาวไม่เป็นศูนย์ ตั้งค่าไฟล์คอนฟิกูเรชันทั้งหมดเป็นโหมด 664 root เจ้าของ, ระบบ กลุ่ม หาก /etc/gated.conf+ ไม่มีอยู่ ให้สร้างไฟล์ความยาวเป็นศูนย์ ที่มีไฟล์โหมดถูกตั้งค่าเป็น 664, root เจ้าของ, ระบบกลุ่ม

คำสั่งย่อต่อไปนี้จะทำให้การสนับสนุนสำหรับการเริ่มทำงานและ การหยุดทำงาน **gated** และเพื่อพิจารณาสถานะ การรัน:

ไอเท็ม	คำอธิบาย
การรัน	พิจารณาว่า <b>gated</b> กำลังรันอยู่ในขณะนี้หรือไม่ ทำโดยการตรวจสอบเพื่อดูว่า <b>gated</b> ได้ล็อกไฟล์ที่มี pid หรือไม่ หาก pid ในไฟล์มีความอ่อนไหว และหากมีกระบวนการที่กำลังรันด้วย pid นั้น ออกด้วยสถานะ ศูนย์ หาก <b>gated</b> กำลังรัน มิฉะนั้นเป็นไมเท่ากับศูนย์
start	เริ่มทำงาน <b>gated</b> คำสั่งส่งคืน ขอดผิดพลาดหาก <b>gated</b> กำลังรันอยู่ มิฉะนั้น จะเรียกใช้งานไบนารี <b>gated</b> และรอจนกว่า ถึงช่วงระยะเวลา (10 วินาทีโดยค่าดีฟอลต์ มิฉะนั้นตั้งที่ค่าด้วยอ็อปชัน -t) จนกระทั่งกระบวนการที่เริ่มทำงานใหม่ จัดการการล็อกบนไฟล์ pid สถานะการออกที่ไม่เป็นศูนย์จะถูกส่งกลับ หากพบข้อผิดพลาดขณะเรียกใช้งานไบนารี หรือหากการล็อก ไม่สามารถจัดการข้อมูลบนไฟล์ pid ภายในช่วงเวลาเวลาที่ระบุหยุดทำงาน <b>gated</b> อย่างเรียบร้อยหากเป็นไปได้ อย่างไรก็ตามไม่เรียบร้อยหากทำไม่ได้ คำสั่งส่งคืนขอดผิดพลาด (ที่มีสถานะการออก ไม่เป็นศูนย์) หาก <b>gated</b> ไม่ได้อาศัยอยู่ในขณะนี้ มิฉะนั้นจะส่งสัญญาณการสิ้นสุดไปยัง <b>gated</b> และ รอจนครบช่วงระยะเวลา (10 วินาทีโดยค่าดีฟอลต์ มิฉะนั้นตามที่ระบุ ด้วยอ็อปชัน -t) สำหรับกระบวนการ ที่จะออกจากการทำงาน หาก <b>gated</b> ในกรออกจากการทำงานภายใน ระยะเวลาช่วงเวลา โดยจะถูกส่งสัญญาณอีกครั้งโดยมีสัญญาณการยุติการทำงานครั้งที่สอง หากล้มเหลวในการออกจากการทำงานเมื่อสิ้นสุดระยะเวลาครั้งที่สอง จะส่งสัญญาณครั้งที่สามด้วยสัญญาณการคิล ซึ่งจะบังคับให้ ยุติการทำงานในทันทียกเว้นว่ามีบางสิ่งที่ไม่ถูกต้องอย่างยิ่ง คำสั่ง สิ้นสุดด้วยสถานะการออกเป็นศูนย์เมื่อพบว่า <b>gated</b> ยุติการทำงาน มิฉะนั้นจะเป็นค่าไมเท่ากับศูนย์
stop	หาก <b>gated</b> กำลังรัน จะถูกทำให้ยุติการทำงาน ผ่านโทรซีเตอร์เดียวกันกับที่ใช้สำหรับคำสั่ง <b>stop</b> ดับบน เมื่อ <b>gated</b> ก่อนหน้ายุติลง หรือ หากไม่ได้กำลังรันอยู่ก่อนหน้าที่จะเรียก ใช้งานคำสั่ง กระบวนการ <b>gated</b> ใหม่ จะถูกเรียกใช้งานโดยใช้โทรซีเตอร์ที่อธิบายสำหรับคำสั่ง <b>start</b> ดับบน ค่าสถานะการออกที่ไม่เท่ากับศูนย์ถูกส่งกลับหากมีขั้นตอนใดๆ ในโทรซีเตอร์นี้ ล้มเหลว

คำสั่งย่อต่อไปนี้อนุญาตให้ลบไฟล์ที่สร้างออก โดยการเรียกใช้งานบางคำสั่งดับบน:

ไอเท็ม	คำอธิบาย
rmcore	ลบคอร์ดัมพ์ไฟล์ <b>gated</b> ใดๆ ที่มีอยู่
rmdump	ลบดัมพ์ไฟล์สถานะ <b>gated</b> ใดๆ ที่มีอยู่
rmparse	ลบไฟล์ข้อผิดพลาดการวิเคราะห์ค่าที่สร้างเมื่อคำสั่ง <b>checkconf</b> หรือ <b>checknew</b> ถูกเรียกใช้งานและพบข้อผิดพลาดทางไวยากรณ์ ในไฟล์คอนฟิกูเรชันที่กำลังตรวจสอบ

โดยค่าดีฟอลต์ **gated** จัดการกำหนดค่า จากไฟล์ที่ปกติชื่อ **/etc/gated.conf** โปรแกรม **gdc** ยังดูแลรักษาเวอร์ชันอื่นๆ หลายเวอร์ชัน ของไฟล์คอนฟิกูเรชันไว้ในชื่อเฉพาะ:

ไอเท็ม	คำอธิบาย
<b>/etc/gated.conf+</b>	ไฟล์คอนฟิกูเรชันใหม่ เมื่อ <b>gdc</b> ได้รับ การร้องขอให้ติดตั้งไฟล์คอนฟิกูเรชันใหม่ ไฟล์นี้จะถูกเปลี่ยนชื่อเป็น <b>/etc/gated.conf</b>
<b>/etc/gated.conf-</b>	ไฟล์คอนฟิกูเรชันเก่า เมื่อ <b>gdc</b> ได้รับ การร้องขอให้ติดตั้งไฟล์คอนฟิกูเรชันใหม่ <b>/etc/gated.conf</b> ก่อนหน้าจะถูกเปลี่ยนชื่อเป็นชื่อนี้
<b>/etc/gated.conf~</b>	ไฟล์คอนฟิกูเรชันเก่ามาก <b>gdc</b> ยังคงเก็บ ไฟล์คอนฟิกูเรชันเก่า ก่อนหน้าไว้ภายใต้ชื่อนี้

## ไฟล์

ไอเท็ม	คำอธิบาย
<b>/usr/sbin/gated</b>	ไบนารี <b>gated</b>
<b>/etc/gated.conf</b>	ไฟล์คอนฟิกูเรชัน Current <b>gated</b>
<b>/etc/gated.conf+</b>	ไฟล์คอนฟิกูเรชันใหม่กว่า
<b>/etc/gated.conf-</b>	ไฟล์คอนฟิกูเรชันเก่ากว่า
<b>/etc/gated.conf~</b>	ไฟล์คอนฟิกูเรชันเก่ากว่ามาก
<b>/etc/gated.pid</b>	ที่ <b>gated</b> เก็บค่า pid
<b>/var/tmp/gated_dump</b>	ดัมพ์ไฟล์สถานะของ <b>gated</b>
<b>/var/tmp/gated.log</b>	ที่เก็บข้อผิดพลาดวิเคราะห์ค่าของไฟล์คอนฟิกูเรชัน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“gated Daemon” ในหน้า 667

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง syslogd

---

## คำสั่ง gencat

### วัตถุประสงค์

สร้างและแก้ไขแค็ตตาล็อกข้อความ

### ไวยากรณ์

`gencat CatalogFile SourceFile ...`

### คำอธิบาย

คำสั่ง `gencat` สร้างไฟล์แค็ตตาล็อกข้อความ (โดยปกติเป็น \*.cat) จากไฟล์ต้นฉบับข้อความ (โดยปกติเป็น \*.msg) คำสั่ง `gencat` จะผสานไฟล์ต้นฉบับข้อความ ที่ระบุโดยพารามิเตอร์ `SourceFile` ลงในแค็ตตาล็อกข้อความที่จัดรูปแบบ ที่ระบุโดยพารามิเตอร์ `CatalogFile` หลังจากป้อนข้อความลงในไฟล์ต้นฉบับ ให้ใช้คำสั่ง `gencat` เพื่อประมวลผลไฟล์ต้นฉบับเพื่อสร้างแค็ตตาล็อกข้อความ คำสั่ง `gencat` สร้างไฟล์แค็ตตาล็อกหากยังไม่มีไฟล์นั้นอยู่ หากไฟล์แค็ตตาล็อกมีอยู่แล้ว คำสั่ง `gencat` จะรวมข้อความใหม่ในไฟล์แค็ตตาล็อก

คุณสามารถระบุไฟล์ต้นฉบับ ข้อความเป็นจำนวนใดๆ คำสั่ง `gencat` ประมวลผลหลายๆ ไฟล์ต้นฉบับ ทีละหนึ่งไฟล์ ตามลำดับที่ระบุ แต่ละไฟล์ต้นฉบับ ต่อๆ มาจะแก้ไขแค็ตตาล็อก หากชุดและหมายเลขข้อความชนกัน ข้อความใหม่ที่กำหนดในพารามิเตอร์ `SourceFile` จะแทนที่ข้อความเก่าที่มีอยู่ในพารามิเตอร์ `CatalogFile` ในขณะนั้น หมายเลขข้อความต้อง อยู่ในช่วง 1 ถึง NL\_MSGMAX หมายเลขชุดต้องอยู่ในช่วง 1 ถึง NL\_SETMAX

คำสั่ง `gencat` ไม่ ยอมรับ identifiers ข้อความสัญลักษณ์ คุณต้องรันคำสั่ง `mecatdefs` หากคุณต้องการใช้ identifier ข้อความสัญลักษณ์

หมายเหตุ: เอาต์พุตมาตรฐานถูกใช้ถ้าอักขระ - (dash) ถูก ระบุเป็นพารามิเตอร์ `CatalogFile` อินพุต มาตรฐานถูกใช้หากระบุอักขระ - (เส้นประ) เป็นพารามิเตอร์ `SourceFile`

### สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย  
0 ดำเนินการเป็นผลสำเร็จ  
>0 มีข้อผิดพลาดเกิดขึ้น

### ตัวอย่าง

เมื่อต้องการสร้างแค็ตตาล็อก test.cat จากไฟล์ต้นฉบับ test.msg ให้ป้อน:

```
gencat test.cat test.msg
```

ไฟล์ test.msg ไม่มี identifiers สัญลักษณ์

## ไฟล์

ไอเท็ม  
/usr/bin/gencat

คำอธิบาย  
มีคำสั่ง gencat

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง dspcat” ในหน้า 229

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง runcat

คำสั่ง catopen

ภาพรวม Message Facility

---

## คำสั่ง gencopy

### วัตถุประสงค์

อนุญาตให้ผลิตภัณฑ์ซอฟต์แวร์ที่มีรูปแบบแพ็คเกจแตกต่างกัน (installp, RPM, ISMP) ถูกคัดลอก

### ไวยากรณ์

ในการคัดลอกซอฟต์แวร์จากสื่อบันทึกไปยังตำแหน่งปลายทาง

```
gencopy -d Media [ -t TargetLocation ] [ -D ] [ -b bffcreateFlags ] [ -U ] [ -X ] -f File | CopyList... | all
```

ในการแสดงรายการผลิตภัณฑ์ซอฟต์แวร์และแพ็คเกจบนสื่อบันทึก

```
gencopy -L -d Media [ -D ]
```

### คำอธิบาย

คำสั่ง **gencopy** คือ wrapper ของคำสั่ง **bffcreate** โดยกำหนดว่าอิมเมจใดที่จะต้องถูกคัดลอกและเรียกใช้ คำสั่งที่เหมาะสมสำหรับ RPM, ISMP หรืออิมเมจชนิดอื่นๆ ที่ไม่รู้จักรายการของไฟล์ที่จำเป็น ไฟล์ทั้งหมดในไดเรกทอรีจะถูกรวบรวมไปยังตำแหน่งปลายทาง

### แฟล็ก

ไอเท็ม	คำอธิบาย
-b <i>bffcreateFlags</i>	ระบุแฟล็กต่อไปนี้ที่ใช้ได้: l, q, v, w และ S
-d <i>Media</i>	ระบุอุปกรณ์หรือไดเรกทอรีที่มีอิมเมจติดตั้งอยู่ สื่อบันทึก สามารถเป็นอุปกรณ์ (/dev/cd0, /dev/rmt0) หรือไดเรกทอรี
-D	ระบุดีบักโหมด แฟล็กนี้สำหรับการดีบักสคริปต์นี้ โดยสร้างเอาต์พุตจำนวนมากและไม่ควรใช้สำหรับการดำเนินการปกติ
-f <i>File</i>	ระบุไฟล์ที่มีรายการอิมเมจเพื่อคัดลอกไปยังตำแหน่ง ปลายทาง อิมเมจ <i>installp</i> , RPM และ ISMP ควรนำหน้าด้วย <b>I</b> , <b>R</b> และ <b>J</b> : ตามลำดับ นำหน้าแฟกเกจการแก้ไขชั่วคราวด้วย <b>E</b> :
-L	แสดงรายการแฟกเกจการติดตั้งบนสื่อบันทึก รายการานี้ค้นด้วยโคลอน และมีข้อมูลต่อไปนี้:  file_name:package_name:fileset:V.R.M.F:type:platform:คำอธิบาย  bos.sysmgt:bos.sysmgt:bos.sysmgt.nim.client:4.3.4.0:I:R:Network Install Manager – Client Tools  bos.sysmgt:bos.sysmgt:bos.sysmgt.smit:4.3.4.0:I:R:System Management Interface Tool (SMIT)
-t <i>TargetLocation</i>	ระบุไดเรกทอรีที่อิมเมจไฟล์การติดตั้งถูกเก็บอยู่ หากไม่ระบุแฟล็ก -t ไฟล์จะถูกบันทึกในไดเรกทอรี /usr/sys/inst.images
-U	อัปเดตโครงสร้างไดเรกทอรีของที่เก็บปลายทาง เป็นมาตรฐานปัจจุบัน หากจำเป็น มาตรฐานปัจจุบันจำเป็นต้องจัดการแบ่ง อิมเมจให้ เป็นไดเรกทอรีย่อยตามชนิดแฟกเกจและสถาปัตยกรรม ตัวอย่างเช่น อิมเมจ <i>installp</i> อยู่ในไดเรกทอรี <i>SaveDir/installp/ppc</i> เมื่อคัดลอกจากต้นทางที่มี โครงสร้างนี้ ปลายทางจำเป็นต้องได้รับการปรับให้สอดคล้องกัน การระบุแฟล็ก -U อนุญาตให้คำสั่ง <i>gencopy</i> สร้างโครงสร้างไดเรกทอรีย่อยที่เหมาะสมในที่เก็บของคุณและย้าย อิมเมจที่มีอยู่ไปยังตำแหน่งที่เหมาะสม ยกเว้นว่ามีการคัดลอก ด้วยตนเองที่ไม่ถูกต้อง ดังนั้นแฟล็กนี้ควรใช้เพียงครั้งเดียวเท่านั้น
-X	ขยายระบบไฟล์โดยอัตโนมัติหากจำเป็นต้องใช้พื้นที่เพิ่ม

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

ในการคัดลอกอิมเมจทั้งหมดจากซีดี (/dev/cd0) ไปยัง การใช้ **LPP\_SOURCE** (/export/lpp\_source/500) ให้พิมพ์:  
gencopy -d /dev/cd0 -t /export/lpp\_source/500 all

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/sbin/gencopy	
/usr/sys/inst.data/sys_bundles	
/usr/sys/inst.data/user_bundles	

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง *bffcreate*

## คำสั่ง gencore

## วัตถุประสงค์

สร้างไฟล์แกนสำหรับกระบวนการที่กำลังรัน

## ไวยากรณ์

`gencore ProcessID FileName`

### คำอธิบาย

คำสั่ง `gencore` สร้างไฟล์แกนของกระบวนการที่ระบุโดย ID กระบวนการ `ProcessID` โดยไม่ยุติการทำงาน กระบวนการ ไฟล์แกนที่สร้างมีอิมเมจหน่วยความจำของกระบวนการ ซึ่งสามารถใช้กับคำสั่ง `dbx` เพื่อใช้ในการ ดีบั๊ก ไฟล์แกนที่สร้างจะถูกตั้งชื่อตามที่ระบุโดยพารามิเตอร์ `FileName`

คำสั่ง `gencore` ไม่สามารถไฟล์แกนในตำแหน่งที่ตั้งค่าโดยคำสั่ง `chcore` หรือ `syscorepath` ไฟล์แกน ถูกเก็บในพาร์ที่ระบุโดยพารามิเตอร์ `FileName` หาก `FileName` ระบุเฉพาะชื่อของไฟล์ ไฟล์แกน จะเก็บอยู่ในแกนปัจจุบัน

### พารามิเตอร์

ไอเท็ม	คำอธิบาย
<code>FileName</code>	ระบุชื่อไฟล์ของไฟล์แกนที่คำสั่ง <code>gencore</code> สร้าง
<code>ProcessID</code>	ระบุ ID กระบวนการของกระบวนการที่ <code>gencore</code> จะใช้สร้างไฟล์แกน

### สถานะออก

- 0 ไฟล์แกนถูกสร้างเสร็จสมบูรณ์
- >0 มีข้อผิดพลาดเกิดขึ้น อาจสร้างไฟล์แกนเพียงบางส่วน

### ตัวอย่าง

- ในการสร้างไฟล์แกนชื่อ "core.1095" สำหรับกระบวนการที่มี ID กระบวนการ 1095 ให้ป้อน:  
`gencore 1095 core.1095`

ซึ่งจะสร้างไฟล์แกนโดยไม่ยุติการทำงานกระบวนการ

### ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/gencore</code>	มีคำสั่ง <code>gencore</code>

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `dbx`” ในหน้า 9

---

## คำสั่ง `genfilt`

### วัตถุประสงค์

เพิ่มกฎตัวกรอง

# ไวยากรณ์

```
genfilt -v 4|6 [ -n fid ] [ -a D|P|I|L|E|H|S ] -s s_addr -m s_mask [ -d d_addr ] [ -M d_mask ] [ -g Y|N ] [ -c protocol ] [ -o s_opr ] [ -p s_port ] [ -O d_opr ] [ -P d_port ] [ -r R|L|B ] [ -w I|O|B ] [ -I Y|N ] [ -f Y|N|O|H ] [ -t tid ] [ -i interface ] [ -D description ] [ -e expiration_time ] [ -x quoted_pattern ] [ -X pattern_filename ] [ -C antivirus_filename ]
```

## คำอธิบาย

ใช้คำสั่ง **genfilt** เพื่อเพิ่มกฎการกรองลงในตารางกฎการกรอง กฎการกรองที่สร้างขึ้น โดยคำสั่งนี้ถูกเรียกเป็นกฎการกรองด้วยตนเอง กฎตัวกรอง IPsec สามารถกำหนดค่าได้โดยใช้คำสั่ง **genfilt** หรือ IPsec smit (IP เวอร์ชัน 4 หรือ IP เวอร์ชัน 6)

## แฟล็ก

### ไอเท็ม

-a Action

### คำอธิบาย

ค่า Action ต่อไปนี้ที่ได้รับอนุญาต:

- D (ปฏิเสธ) บล็อกการรับส่งข้อมูล
- P (อนุญาต) อนุญาตการรับส่งข้อมูล
- I ทำให้เป็นกฎตัวกรอง IF
- L ทำให้เป็นกฎตัวกรอง ELSE
- E ทำให้เป็นกฎตัวกรอง ENDIF
- H ทำให้เป็นกฎตัวกรอง SHUN\_HOST
- S ทำให้เป็นกฎตัวกรอง SHUN\_PORT

-C antivirus\_filename

-c protocol

-D คำอธิบาย

-d d\_addr

-e expiration\_time

-f

-g

-i interface

-l

-M

กฎ IF ทั้งหมดต้องปิดด้วยกฎ ENDIF ที่สัมพันธ์กัน กฎที่มีเงื่อนไข เหล่านี้สามารถซ้อนกันได้ แต่การซ้อนภายในที่ถูกต้องยึดตาม หรือ กฎจะต้องไม่ไหลต่ออย่างถูกต้องกับคำสั่ง **mkfilt**

ระบุชื่อไฟล์ป้องกันไวรัส แฟล็ก -C เขาใจในบางเวอร์ชันของ ClamAV Virus Database (<http://www.clamav.net>)

ค่าที่ใช้ได้คือ: **udp, icmp, icmpv6, tcp, tcp/ack, ospf, ipip, esp, ah** และ **all** ค่า **all** ระบุว่ากฎตัวกรองจะ ใช้กับโปรโตคอลทั้งหมด โปรโตคอลยังสามารถระบุเป็นตัวเลข (ระหว่าง 1 ถึง 252) ค่าดีฟอลต์คือ **all** ค่า **tcp/ack** แสดงนัยถึงการตรวจสอบแพ็กเก็ต TCP ที่มีชุดแฟล็ก ACK

ขอความรายละเอียดอย่างย่อสำหรับกฎตัวกรอง นี้เป็น แฟล็กทางเลือกสำหรับกฎตัวกรองแบบสแตติก ซึ่งไม่สามารถใช้กับกฎตัวกรอง ไดนามิก

ระบุแอดเดรสปลายทาง โดยสามารถเป็น IP แอดเดรส หรือชื่อ โฮสต์ หากระบุชื่อโฮสต์, IP แอดเดรสแรกที่ส่งกลับโดยเนมเซิร์ฟเวอร์ สำหรับโฮสต์นั้นจะถูกใช้ คำนี้พร้อมกับ subnet mask ปลายทางจะถูกเปรียบเทียบกับแอดเดรสปลายทางของแพ็กเก็ต IP

ระบุเวลาหมดอายุ เวลาหมดอายุคือ ระยะเวลาที่กฎควรยังคงแอ็คทีฟอยู่เป็นวินาที **expiration\_time** ไม่ลบกฎตัวกรองออกจากฐานข้อมูล **expiration\_time** สัมพันธ์กับระยะเวลาที่กฎตัวกรองแอ็คทีฟขณะประมวลผล ปริมาณรับส่งข้อมูลเน็ตเวิร์ก หากไม่ระบุ **expiration\_time** ดังนั้นเวลาจริง ของกฎตัวกรองจะเป็นไม่สิ้นสุด หากระบุ **expiration\_time** ร่วมกับกฎตัวกรอง SHUN\_PORT (-a S) หรือ SHUN\_HOST (-a H) ดังนั้นนี่จะเป็นระยะเวลาที่รีโมตพอร์ต หรือรีโมตโฮสต์ จะถูกปฏิเสธ หรือ shun เมื่อพารามิเตอร์กฎตัวกรองตรงตามเงื่อนไข หากระบุ **expiration\_time** นี้เป็นอิสระจากกฎ shun ดังนั้นนี่จะเป็น ระยะเวลาที่กฎตัวกรองจะยังคงแอ็คทีฟเมื่อกฎตัวกรอง ถูกโหลดเข้าสู่เคอร์เนลและเริ่มต้นการประมวลผลปริมาณรับส่งข้อมูลเน็ตเวิร์ก

ระบุการควบคุมการแตกแฟรกเมนต์ แฟล็กนี้ระบุว่าจะใช้กับแพ็กเก็ตทั้งหมด (Y), ชิ้นส่วน แฟรกเมนต์ และแพ็กเก็ตที่ไม่ถูกแฟรกเมนต์เท่านั้น (H), แฟรกเมนต์ และส่วนหัวแฟรกเมนต์เท่านั้น (O) หรือแพ็กเก็ตที่ไม่ถูกแฟรกเมนต์ เท่านั้น (N) ค่าดีฟอลต์คือ Y

ใช้กับการจัดเส้นทางผ่านทาง? ต้องระบุเป็น Y (ใช่) หรือ N (ไม่ใช่) หากระบุ Y กฎตัวกรองนี้สามารถใช้กับแพ็กเก็ต IP ที่ใช้การจัดเส้นทางผ่านทาง ค่าดีฟอลต์ คือ yes (Y) ฟิวด์นี้ใช้เพื่ออนุญาตกฎ เท่านั้น

ระบุชื่อของอินเตอร์เฟซ IP ที่ซึ่งกฎตัวกรองใช้ ตัวอย่างของชื่อได้แก่: **all, tr0, en0, lo0** และ **pp0** ค่าดีฟอลต์คือ **all** ระบุการควบคุมบันทึกการทำงาน ต้องระบุเป็น Y (ใช่) หรือ N (ไม่ใช่) หากระบุเป็น Y แพ็กเก็ตที่ตรงกับกฎตัวกรองนี้ จะถูกรวมในบันทึกการใช้ กฎ ค่าดีฟอลต์คือ N (ไม่ใช่)

ระบุ subnet mask ปลายทาง นี้ถูกใช้เปรียบเทียบ แอดเดรสปลายทางของแพ็กเก็ต IP กับแอดเดรสปลายทางของ กฎตัวกรอง

ไอเอ็ม	คำอธิบาย
-m	ระบุ subnet mask ต้นทาง ค่านี้ถูกใช้เปรียบเทียบ แอดเดรสต้นทางของแพ็กเก็ต IP กับแอดเดรสต้นทางของกฎตัวกรอง
-n	ระบุ ID กฎตัวกรอง กฎใหม่จะถูกเพิ่ม ก่อนหน้า กฎตัวกรองที่คุณระบุ สำหรับ IP version 4 นั้น ID ต้องมีค่ามากกว่า 1 เนื่องจากกฎตัวกรองแรกคือกฎที่ระบบสร้างขึ้น และไม่สามารถย้ายได้ หากไม่ใช่แฟล็กนี้ กฎใหม่จะถูกเพิ่มที่ท้ายของตาราง กฎตัวกรอง
-O	ระบุพอร์ตปลายทางหรือการดำเนินการโค้ด ICMP นี่คือการดำเนินการที่จะใช้ในการเปรียบเทียบระหว่างพอร์ตปลายทาง/โค้ด ICMP ของแพ็กเก็ตที่มีพอร์ตปลายทางหรือโค้ด ICMP (แฟล็ก -P) ค่าที่ใช้ได้คือ: lt, le, gt, ge, eq, neq และ any ค่าดีฟอลต์คือ any ค่านี้ต้องเป็น any เมื่อแฟล็ก -c เป็น ospf
-o	ระบุพอร์ตต้นทางหรือการดำเนินการชนิด ICMP นี่คือการดำเนินการที่จะใช้ในการเปรียบเทียบระหว่างพอร์ตต้นทาง/ชนิด ICMP ของ แพ็กเก็ตกับพอร์ตต้นทาง หรือชนิด ICMP (แฟล็ก -p) ที่ระบุในกฎตัวกรองนี้ ค่าที่ใช้ได้คือ: lt, le, gt, ge, eq, neq และ any ค่าดีฟอลต์คือ any ค่านี้ต้องเป็น any เมื่อแฟล็ก -c เป็น ospf
-p	ระบุพอร์ตต้นทาง หรือชนิด ICMP นี่คือการ/ชนิดที่จะถูกเปรียบเทียบกับพอร์ตต้นทาง (หรือชนิด ICMP) ของแพ็กเก็ต IP
-P	ระบุพอร์ตปลายทาง/โค้ด ICMP นี่คือการ/โค้ดที่จะถูกเปรียบเทียบกับพอร์ตปลายทาง (หรือโค้ด ICMP) ของแพ็กเก็ต IP
-r	การจัดเส้นทาง ค่านี้ระบุว่าจะใช้กับแพ็กเก็ตที่ส่งต่อ (R), แพ็กเก็ตที่มีปลายทางหรือต้นทางจากโลคัล โฮสต์ (L) หรือทั้งสอง (B) ค่าดีฟอลต์คือ B
-s s_addr	ระบุแอดเดรสต้นทาง โดยสามารถเป็น IP แอดเดรส หรือชื่อ โฮสต์ หากระบุชื่อโฮสต์, IP แอดเดรสแรกที่ส่งกลับโดยเนมเซิร์ฟเวอร์ สำหรับโฮสต์นั้นจะถูกใช้ ค่านี้พร้อมกับ subnet mask ต้นทางจะถูกเปรียบเทียบกับแอดเดรสต้นทางของแพ็กเก็ต IP
-t	ระบุ ID ของช่องสัญญาณที่สัมพันธ์กับกฎตัวกรองนี้ แพ็กเก็ต ทั้งหมดที่ตรงกับกฎตัวกรองนี้ต้องไปยังช่องสัญญาณที่ระบุ หากไม่ระบุแฟล็กนี้ กฎนี้จะใช้กับการรับส่งข้อมูลที่ไม่ใช่ช่องสัญญาณเท่านั้น
-v	ระบุเวอริชัน IP ของกฎตัวกรอง ค่าที่ใช้ได้คือ 4 และ 6
-w Direction	ระบุว่าจะใช้กับแพ็กเก็ตขาเข้า (I), แพ็กเก็ตขาออก (O) หรือทั้งสอง (B) ค่าดีฟอลต์คือ B ไม่สามารถใช้ทิศทางขาออก (O) กับอ็อปชันรูปแบบ -x, -X หรือ -C จะใช้ได้เมื่อระบุทั้งสองทิศทาง (B) ที่มีอ็อปชันรูปแบบ แต่เฉพาะแพ็กเก็ตขาเข้าเท่านั้นที่ถูกตรวจสอบ แพ็กเก็ต
-X pattern_filename	ระบุชื่อไฟล์รูปแบบ หากมีมากกว่าหนึ่งรูปแบบ สัมพันธ์กับกฎตัวกรองนี้ ดังนั้นชื่อไฟล์รูปแบบจะต้องถูกใช้ชื่อไฟล์รูปแบบต้องอยู่ในรูปแบบของหนึ่งรูปแบบต่อหนึ่งบรรทัด รูปแบบ คือสตริงอักขระที่ไม่อยู่ในเครื่องหมายคำพูด ไฟล์นี้ถูกอาจเมื่อกฎตัวกรอง ถูกเรียกทำงาน สำหรับข้อมูลเพิ่มเติม โปรดดูที่คำสั่ง mkfilt
-x pattern	ระบุสตริงอักขระที่อยู่ในเครื่องหมายคำพูด หรือรูปแบบ สตริงนี้ ที่ถูกระบุจะถูกแปลงเป็นสตริง ASCII ยกเว้นจะถูกนำหน้าด้วย 0x ไม่วาร์ณี่ได้จะถูกแปลงเป็นสตริงเลขฐานสิบหก -xpattern ถูกเปรียบเทียบกับปริมาณรับส่งข้อมูลเน็ตเวิร์ก

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## คำสั่ง geninstall

### วัตถุประสงค์

โปรแกรมติดตั้งทั่วไปที่ติดตั้งผลิตภัณฑ์ซอฟต์แวร์ที่มีรูปแบบแพ็คเกจ แตกต่างกัน ตัวอย่างเช่น installp, RPM, SI และ ISMP

### ไวยากรณ์

```
geninstall -d Media [-I installpFlags] [-E|-T] [-t ResponseFileLocation] [-e LogFile] [-p] [-F] [-Y] [-Z] [-D] {
-f File|Install_List}|all}
```

หรือ

```
geninstall -k [-d Media] [-p] [-Y] [-f File|install_list|all|update_all]
```

หรือ

```
geninstall -u [-e LogFile] [-E|-T] [-t ResponseFileLocation] [-D] {-f File|Uninstall_List...}
```

หรือ

```
geninstall -L -d Media [-e LogFile] [-D]
```

## คำอธิบาย

ยอมรับแฟล็ก `installp` ปัจจุบันทั้งหมดและส่งไปยัง `installp` บางแฟล็ก (ตัวอย่างเช่น `-L`) ถูกโอเวอร์โหลดเพื่อแสดงรายการผลิตภัณฑ์ทั้งหมดบนสื่อบันทึก แฟล็กที่ไม่มีควมหมายสำหรับผลิตภัณฑ์แพ็คเกจ ISMP จะถูกข้าม คำนี้อนุญาตให้โปรแกรม (เช่น NIM) ดำเนินการส่งแฟล็ก `installp` ไปยัง `geninstall` ได้เสมอ แต่เฉพาะแฟล็กที่มีความหมายเท่านั้น ที่ถูกใช้

คำสั่ง `geninstall` จัดให้มีวิธีการอย่างง่ายในการดูการแก้ไข ที่ทำกับไฟล์คอนฟิกูเรชันที่แสดงรายการใน `/etc/check_config.files` เมื่อไฟล์เหล่านี้ถูกเปลี่ยนแปลงระหว่างการติดตั้ง `geninstall` หรือการอัปเดต ความแตกต่างระหว่างไฟล์เก่าและใหม่จะถูกบันทึกใน `/var/adm/ras/config.diff` หาก `/etc/check_config.files` ร้องขอให้บันทึกไฟล์เก่า ไฟล์เก่าสามารถพบได้ในไดเรกทอรี `/var/adm/config`

ไฟล์ `/etc/check_config.files` สามารถถูกแก้ไข และสามารถใช้เพื่อระบุว่าไฟล์คอนฟิกูเรชันเก่าที่มีการเปลี่ยนแปลงควรถูกบันทึก (ระบุโดย `s`) หรือลบ (ระบุโดย `d`) และมีรูปแบบต่อไปนี้:

```
d /etc/inittab
```

ข้อสรุปของกิจกรรมการติดตั้งของคำสั่ง `geninstall` จะถูกเก็บไว้ใน `/var/adm/sw/geninstall.summary` ไฟล์นี้มีรายการที่ค้นด้วยโคลนของชุดไฟล์ที่ติดตั้งโดย `installp` และคอมโพเนนต์ที่ติดตั้งโดย ISMP คำนี้ถูกใช้เป็นหลักเพื่อให้ข้อมูลสรุปสำหรับการติดตั้งแบบไม่โต้ตอบ

**หมายเหตุ:** อ้างอิงไฟล์ `README.ISMP` ในไดเรกทอรี `/usr/lpp/bos` เพื่อศึกษาเพิ่มเติมเกี่ยวกับการติดตั้ง แพ็คเกจ ISMP และการใช้ไฟล์การตอบกลับ รวมถึง คำสั่ง `geninstall` ยังสามารถทำการติดตั้งไฟล์ interim fix ที่มี การอัปเดตพร้อมกัน โปรแกรมฟิกซ์เฉพาะกิจใดๆ ที่มีการอัปเดตพร้อมกันต้องมีการวางไว้ภายในไดเรกทอรีย่อยที่ชื่อ `updates` ในไดเรกทอรีที่มีอิมเมจการติดตั้ง และคำสั่ง `geninstall` จะติดตั้งโปรแกรมฟิกซ์ดังกล่าวอย่างเหมาะสม

## แฟล็ก

ไอเท็ม	
-d Device or Directory	<p>คำอธิบาย</p> <p>ระบุอุปกรณ์หรือไดเรกทอรีที่มีอิมเมจที่ต้องการติดตั้ง</p> <p>คำสั่ง <b>geninstall</b> จะค้นหาอิมเมจในพาทต่อไปนี้:</p> <ul style="list-style-type: none"> <li>• /mount_point/installp/ppc (แพ็คเกจ installp)</li> <li>• /mount_point/RPMS/ppc (แพ็คเกจ RPM)</li> <li>• /mount_point/emgr/ppc (แพ็คเกจโปรแกรมพีคซ์เฉพาะกิจสำหรับ AIX)</li> <li>• /mount_point/ISMP/ppc (แพ็คเกจ ISMP สำหรับ AIX)</li> </ul> <p>หากไม่มีพาท คำสั่ง <b>geninstall</b> จะค้นหาอิมเมจใน ไดเรกทอรีฐานของอุปกรณ์เฉพาะ หากชื่ออิมเมจไม่นำหน้าด้วยส่วนนำหน้าที่ระบุ ชนิดของอิมเมจ คำสั่ง <b>geninstall</b> จะระบุชนิดสำหรับ อิมเมจ</p> <p>หากมีพาท และไม่พบอิมเมจในพาทใดๆ และหากอิมเมจไม่มีส่วนนำหน้า อิมเมจจะถือว่าเป็นอิมเมจในรูปแบบ RPM</p>
-D	ระบุดีบักโหมด แฟล็กนี้สำหรับการดีบักสคริปต์นี้ โดยสร้างเอาต์พุตจำนวนมากและไม่ควรใช้สำหรับการดำเนินการปกติ
-e LogFile	เปิดใช้งานการบันทึกการทำงานเหตุการณ์ แฟล็ก -e เปิดให้ผู้ใช้ผนวกส่วนที่แน่นอนของเอาต์พุตคำสั่ง <b>geninstall</b> ไปยังไฟล์ที่ระบุโดยตัวแปร <i>LogFile</i> ตัวแปร <i>LogFile</i> ต้องระบุไฟล์ที่สามารถเขียนได้ที่มีอยู่แล้ว และระบบที่มี ไฟล์นั้นอยู่ต้องมีพื้นที่เพียงพอต่อการเก็บบันทึกการทำงาน ไฟล์บันทึกการทำงานไม่มีการตัดทับค่าเก่า
-E	สร้างไฟล์การตอบกลับ ISMP ที่บันทึกในตำแหน่งดีฟอลต์ ซึ่งคือไดเรกทอรีที่มีไฟล์การติดตั้งผลิตภัณฑ์ อีอ็อปชันนี้จำเป็นต้อง รับการติดตั้ง ISMP หรือการถอนการติดตั้งแบบโต้ตอบ และโดยสมบูรณ์ ไฟล์การตอบกลับผลลัพธ์จะถูกใช้เพื่อให้อีอ็อปชันเหมือนกันกับการติดตั้ง ในอนาคตหรือการถอนการติดตั้งของผลิตภัณฑ์เดียวกัน การสร้างการบันทึก ไฟล์การตอบกลับจะส่งผลให้เกิดการติดตั้ง หรือการถอนการติดตั้งของผลิตภัณฑ์
-f File	ระบุไฟล์ที่มีรายการอิมเมจเพื่อคัดลอกไปยังตำแหน่ง ปลายทาง อิมเมจ <b>installp</b> , RPM และ ISMP ควรนำหน้าด้วย <b>I</b> , <b>R</b> ; และ <b>J</b> : ตามลำดับ นำหน้าแพ็คเกจการแก้ไข ชั่วคราวด้วย <b>E</b> :
-F	อนุญาตให้ผู้ใช้ติดตั้งแพ็คเกจใหม่ที่ไดติดตั้งไว้แล้ว หรือ ติดตั้งแพ็คเกจที่เก่ากว่าเวอร์ชันที่ติดตั้งปัจจุบัน
-I installpFlags	ระบุแฟล็ก <b>installp</b> เพื่อใช้เมื่อเรียกใช้ คำสั่ง <b>installp</b> แฟล็กที่ถูกใช้ระหว่าง การดำเนินการติดตั้งสำหรับ <b>installp</b> ได้แก่แฟล็ก a, b, c, D, e, E, F, g, I, J, M, N, O, p, Q, q, S, t, v, V, w และ X แฟล็ก <b>installp</b> ที่ไม่ถูกใช้ระหว่าง การติดตั้งได้แก่แฟล็ก C, i, r, z, A และ I คำสั่ง <b>installp</b> ควรถูกเรียกใช้ โดยตรงเพื่อดำเนินการฟังก์ชันเหล่านี้ แฟล็ก -u, -d, -L และ -f ควร ถูกกำหนดภายนอกแฟล็ก -I

ไอเอ็ม  
-k

### คำอธิบาย

คุณสามารถใช้แฟล็กนี้ใน IBM AIX 7.2 Technology Level 1 หรือใหม่กว่า เพื่อใช้การดำเนินการ Live Update เพื่ออัปเดต Service Packs (SPs), Technology levels (TLs) กลุ่มหรือแต่ละการอัปเดต และโปรแกรมพิกซ์เฉพาะกิจที่ทำงานบนเครื่องเป็น LU CAPABLE เมื่อต้องการกำหนดว่า โปรแกรมพิกซ์เฉพาะกิจคือ LU CAPABLE คุณสามารถดำเนินการติดตั้งเพื่อแสดงตัวอย่าง ก่อนที่คุณจะเริ่มดำเนินการดำเนินการ Live Update คุณต้องคอมมิตการอัปเดตใดๆ ที่มีอยู่บนระบบก่อน การดำเนินการ Live Update จะคอมมิตอัปเดตทั้งหมดที่นำมาใช้ระหว่างกระบวนการขยายระบบไฟล์ใดๆ ที่จำเป็น และติดตั้งซอฟต์แวร์ใดๆ ที่จำเป็นต้องมีเสมอ กระบวนการการติดตั้งอื่นทั้งหมดจะถูกป้องกันไม่ให้เริ่มทำงาน ระหว่างการดำเนินการ Live Update

หมายเหตุ: หากการดำเนินการ Live Update ของคุณ รวมถึงการติดตั้งอัปเดต คุณต้องทำการสำรองข้อมูลระบบก่อนที่จะดำเนินการ Live Update การดำเนินการ Live Update จะไม่สร้าง อิมเมจการสำรองข้อมูลให้

- การดำเนินการ Live Update จะล้มเหลว หากคุณกำลังดำเนินการงานต่อไปนี้ในสภาวะแวดล้อมของคุณ:
- การติดตั้งโปรแกรมพิกซ์เฉพาะกิจในสภาวะแวดล้อมพิเศษ เช่น เวิร์กโหลดพาร์ติชัน (wpar), multibos หรือในสภาวะแวดล้อมที่เป็นทางเลือกอื่น
  - การติดตั้งระบบปฏิบัติการ
  - ชื่อโลจิคัลวอลุ่มสำหรับโลจิคัลวอลุ่มของระบบปฏิบัติการที่ต้องการ (/ , /var, /opt, /usr, /etc) และโลจิคัลวอลุ่มสำหรับบูตไม่ได้ใช้เป็น ชื่อโลจิคัลวอลุ่มดีฟอลต์

การดำเนินการ Live Update ต้องการ อินพุตเพิ่มเติมที่ระบุไว้ในไฟล์ /var/adm/ras/liveupdate/lvupdate.data สำหรับข้อมูลเพิ่มเติมเกี่ยวกับไฟล์นี้ โปรดดูที่ /var/adm/ras/liveupdate/lvupdate.template

หากการอัปเดต ถูกนำไปใช้และคอมมิตได้สำเร็จ และหากกระบวนการ Live Update ล้มเหลว คุณสามารถรันกระบวนการ Live Update อีกครั้งโดยการรันคำสั่ง `geninstall` โดยมีเฉพาะแฟล็ก -k ในสถานการณ์นี้ คุณไม่จำเป็นต้องรวมอุปกรณ์และซอฟต์แวร์ในคำสั่ง

ใน AIX 7200-01 หรือใหม่กว่า คุณสามารถใช้และคอมมิตอัปเดตหรือโปรแกรมพิกซ์เฉพาะกิจใดๆ ได้โดยใช้ SMIT หรือเมธอดใดๆ ที่คุณต้องการและดำเนินการ Live Update โดยไม่ใช้แฟล็ก -d เพื่อให้คุณไม่ต้องรีสตาร์ทระบบ

อ็พชัน `all` จะติดตั้งซอฟต์แวร์ทั้งหมดที่มีในอุปกรณ์หรือไดเรกทอรีที่ระบุโดยแฟล็ก -d กระบวนการนี้จะติดตั้งซอฟต์แวร์ทั้งหมดใน แหล่งที่มาของซอฟต์แวร์ที่ระบุโดยแฟล็ก -d แม้ว่าซอฟต์แวร์ไม่ได้ติดตั้งบนระบบก่อนหน้านี้ อ็พชัน `update_all` จะติดตั้งเฉพาะเวอร์ชันที่สูงกว่าของ ซอฟต์แวร์ที่ระบุที่ติดตั้งอยู่บนระบบแล้ว อ็พชัน `update_all` จะติดตั้งซอฟต์แวร์ใหม่เฉพาะหากเวอร์ชันใหม่ของซอฟต์แวร์ที่มีอยู่ มีสิ่งที่จะต้องสำหรับซอฟต์แวร์ใหม่

-L ชดไฟล์ `bos.liveupdate.rte` ต้องได้รับการติดตั้งเพื่อดำเนินการ Live Update แสดงรายการเนื้อหาของสื่อบันทึก รูปแบบเอาต์พุตเหมือนกันรูปแบบ `installp -Lc` โดยมีฟิลด์เพิ่มเติมที่ท้ายสำหรับ ผลิตภัณฑ์ที่จัดรูปแบบ ISMP และ RPM

-p ดำเนินการแสดงตัวอย่างของการดำเนินการโดยการรันการติดตั้งล่วงหน้า ทั้งหมดเพื่อตรวจหาการดำเนินการที่ระบุ

-t *ResponseFileLocation* อนุญาตให้ระบุตำแหน่งทางเลือกสำหรับไฟล์ การตอบกลับหรือเพิ่มเพลตไฟล์การตอบกลับ ตำแหน่งดีฟอลต์คือไดเรกทอรีที่มี ไฟล์การติดตั้งผลิตภัณฑ์ แฟล็กนี้สามารถใช้เพื่อสร้างไฟล์ การตอบกลับหรือเพิ่มเพลตในเพิ่มเพลตอื่น *ResponseFileLocation* สามารถเป็นไฟล์หรือชื่อไดเรกทอรี ใดๆ หนึ่ง ถ้า *ResponseFileLocation* คือไดเรกทอรี ไดเรกทอรีนี้ต้องมีอยู่ หาก *ResponseFileLocation* ไม่ใช้ไดเรกทอรีที่มีอยู่แล้ว จะถูกกว่าค่าที่ระบุนั้นเป็นชื่อไฟล์

-T สร้างเพิ่มเพลตไฟล์การตอบกลับ ISMP ในตำแหน่งดีฟอลต์ ซึ่งคือไดเรกทอรีที่มีไฟล์การติดตั้งผลิตภัณฑ์ เพิ่มเพลตผลลัพธ์สามารถใช้เพื่อสร้างไฟล์การตอบกลับ สำหรับการติดตั้ง หรือการถอนการติดตั้งในขนาดของผลิตภัณฑ์เดียวกันด้วยอ็พชันที่ต้องการ การสร้าง เพิ่มเพลตไฟล์การตอบกลับจะไม่มีผลในการติดตั้งหรือการถอนการติดตั้ง ผลิตภัณฑ์

-u ดำเนินการถอนการติดตั้งซอฟต์แวร์ที่ระบุ สำหรับผลิตภัณฑ์ ISMP โปรแกรมถอนการติดตั้งที่แสดงรายการในฐานข้อมูลผู้ค้าจะถูกเรียกใช้ นำหน้าด้วย "J:"

-Y ยอมรับข้อตกลงการอนุญาตใช้สิทธิ์ซอฟต์แวร์ที่จำเป็นเพื่อให้ซอฟต์แวร์ได้รับการติดตั้ง แฟล็กนี้ยังถูกยอมรับเป็นแฟล็ก `installp` ที่มี อ็พชัน -I

ไอเอ็ม  
-Z

คำอธิบาย  
แจ้ง `geninstall` ให้เรียกใช้การติดตั้งใน โหมดไม่ได้ตอบ

## ตัวอย่าง

1. เมื่อต้องการติดตั้งผลิตภัณฑ์ทั้งหมดบนสื่อบันทึกซีดีที่อยู่ในไดรฟ์ `cd0`, ให้พิมพ์:

```
geninstall -d /dev/cd0 all
```

หากอิมเมจ ISMP มีแสดงอยู่บนสื่อบันทึก จะแสดงอินเตอร์เฟซแบบกราฟิก อิมเมจ `installp`, SI หรือ RPM ใดๆ ที่ติดตั้ง โดยไม่มีการแสดงพร้อมตัวยกเว้นอิมเมจ `installp` จะถูกกระจายอยู่หลายซีดี

2. เมื่อต้องการติดตั้งโปรแกรมฟิซเฉพาะกิจ ที่ชื่อ `IV12345.160101.epkg.Z` ที่อยู่ในไดเรกทอรี `/images/emgr/ppc` ให้ป้อนคำสั่งต่อไปนี้:

```
geninstall -d /images IV12345.160101.epkg.Z
```

หมายเหตุ: หากมีไดเรกทอรี `/images/emgr/ppc` แต่แพ็คเกจอยู่ในไดเรกทอรี `/images (/images/IV12345.160101.epkg.Z)` คำสั่ง `geninstall` จะไม่พิจารณาแพ็คเกจเป็นโปรแกรมฟิซเฉพาะกิจ และพยายาม ติดตั้งเป็นอิมเมจที่มีรูปแบบ RPM สำหรับข้อมูลเพิ่มเติม โปรดดูที่ แฟล็ก `-d`

## ไฟล์

- `/usr/sbin/gencopy`
- `/usr/sys/inst.data/sys_bundles`
- `/usr/sys/inst.data/user_bundles`

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `installp`

การติดตั้งอัปเดตแบบ live

ข้อจำกัด

การติดตั้งแพ็คเกจซอฟต์แวร์ต่างๆ ที่มีการจัดรูปแบบไว้

---

## คำสั่ง `genkex`

### Purpose

คำสั่ง `genkex` แยก รายการของส่วนขยายเคอร์เนลที่ขณะนี้ถูกโหลดอยู่บนระบบ และแสดงแอดเดรส ขนาด และชื่อพารสำหรับแต่ละส่วนขยายเคอร์เนลในรายการ

### Syntax

```
genkex [ -dh ]
```

## รายละเอียด

สำหรับส่วนขยายเคอร์เนลที่ถูกโหลดอยู่บนระบบ เคอร์เนล จะคอยดูแลรักษารายการที่ลิงก์ที่ประกอบด้วยโครงสร้างข้อมูลที่เรียกใช้รายการตัวโหลด รายการตัวโหลดมีชื่อของส่วนขยาย แอดเดรสเริ่มต้น และขนาด ข้อมูลนี้ถูกรวบรวมและรายงานโดยคำสั่ง `genkex`

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-d</code>	แสดงแอดเดรสและขนาดของส่วน Data นอกเหนือจาก แอดเดรสและขนาดของส่วน Text
<code>-h</code>	แสดงคำสั่งการใช้งาน

## ตัวอย่าง

ในการสร้างรายการของส่วนขยายเคอร์เนลที่โหลดให้บ้อน:

```
genkex
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `genkld`”

“คำสั่ง `genld`” ในหน้า 684

ข้อมูลที่เกี่ยวข้อง:

การมอนิเตอร์และปรับคำสั่งและรูทีนย่อย

---

## คำสั่ง `genkld`

### Purpose

คำสั่ง `genkld` แยก รายการของอ็อบเจกต์ที่แบ่งใช้ที่ขณะนี้ถูกโหลดอยู่บนระบบ และแสดงแอดเดรสขนาด และชื่อพาทสำหรับแต่ละอ็อบเจกต์บนรายการ

### Syntax

```
genkld [ -dh ]
```

## รายละเอียด

สำหรับอ็อบเจกต์ที่แบ่งใช้ที่ถูกโหลดอยู่บนระบบ เคอร์เนล จะคอยดูแลรักษารายการที่ลิงก์ที่ประกอบด้วยโครงสร้างข้อมูลที่เรียกใช้รายการตัวโหลด รายการตัวโหลดมีชื่อของอ็อบเจกต์ แอดเดรสเริ่มต้น และ ขนาด ข้อมูลนี้ถูกรวบรวมและรายงานโดยคำสั่ง `genkld`

## แฟล็ก

ไอเท็ม	คำอธิบาย
-d	แสดงแอดเดรสและขนาดของส่วน Data นอกเหนือจาก แอดเดรสและขนาดของส่วน Text
-h	แสดงคำสั่งการใช้งาน

## ตัวอย่าง

ในการจัดการรายการของอ็อบเจ็กต์ที่แบ่งใช้ที่ถูกลบให้ป้อน:

```
genkld
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง genkex” ในหน้า 682

“คำสั่ง genld”

ข้อมูลที่เกี่ยวข้อง:

การมอนิเตอร์และปรับคำสั่งและรูทีนย่อย

## คำสั่ง genld

### Purpose

คำสั่ง `genld` รวบรวม รายการของกระบวนการทั้งหมดที่ขณะนี้กำลังรันบนระบบ และอาจเลือกรายงาน รายการของอ็อบเจ็กต์ที่ถูกลบที่สอดคล้องกับแต่ละกระบวนการ

### Syntax

```
genld [ -h | -l [ -d ] ] [ -a Area ] [-u]
```

### รายละเอียด

สำหรับแต่ละกระบวนการที่กำลังรันอยู่ขณะนี้ คำสั่ง `genld` จะพิมพ์รายงานที่ประกอบด้วย ID กระบวนการและชื่อ อาจเลือกตามด้วยรายการอ็อบเจ็กต์ที่ถูกลบสำหรับกระบวนการนั้น แอดเดรสของอ็อบเจ็กต์และชื่อพารามิเตอร์จะถูกแสดง สมาชิกของไลบรารีจะถูกแสดงใน วงเล็บเหลี่ยม ตัวอย่างเช่น `/usr/lib/libc.a[shr.o]` หมายความว่า `shr.o` เป็นสมาชิกที่ถูกลบของไลบรารี `libc.a`

- | คุณสามารถกรองเอาต์พุตของคำสั่ง `genld` ได้โดยใช้แฟล็ก `-u` เพื่อแสดงกระบวนการที่มีเวอร์ชันเก่าของอ็อบเจ็กต์ที่ถูกลบ
- | อ็อบเจ็กต์จะถูกพิจารณาว่าเป็นอ็อบเจ็กต์เก่า หากอิมเมจอ็อบเจ็กต์แตกต่างจากอิมเมจที่ติดตั้งอยู่บนระบบไฟล์ ในปัจจุบัน
- | แฟล็ก `-u` ถูกใช้หลังจากการนำอัปเดตไปใช้กับเพื่อแสดงรายการกระบวนการที่ต้องการ การรีสตาร์ทเพื่อใช้ไบนารีและไลบรารี

### Notes:

- ผู้ใช้ที่ไม่มีสิทธิ์สามารถดูอ็อบเจ็กต์ที่ถูกลบได้เฉพาะสำหรับกระบวนการนั้น
- | • หากไม่สามารถตรวจสอบชื่อพารามิเตอร์ไปยังอ็อบเจ็กต์ที่ถูกลบได้ คำสั่ง `genld` อาจไม่รายงานอัปเดตสำหรับอ็อบเจ็กต์นี้
- | หากอยู่บนระบบไฟล์อื่นที่ไม่ใช่ ระบบไฟล์แบบเจอร์นัล (JFS2) อ็อบเจ็กต์ยังอาจถูกรายงานเป็น อัปเดตแล้ว หาก อ็อบเจ็กต์ถูกแทนที่ด้วยสำเนาที่เหมือนกัน

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a Area	แสดงเฉพาะกระบวนการที่ใช้พื้นที่ไลบรารีที่แบ่งใช้ที่ระบุโดยพารามิเตอร์ Area
-d	แสดงแอดเดรสและขนาดของส่วน Data นอกเหนือจาก แอดเดรสและขนาดของส่วน Text อีอพชั่นนี้ไม่มีผลหากไม่มี แฟล็ก -l
-h	แสดงคำสั่งการใช้งาน
-l	รายงานรายการของอ็อบเจกต์ที่โหลดสำหรับแต่ละกระบวนการที่กำลังรันบน ระบบ
-u	แสดงรายการเฉพาะกระบวนการที่มีเวอร์ชันเก่าของอ็อบเจกต์ที่โหลด

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

- เมื่อต้องการดูรายการของอ็อบเจกต์ที่โหลดสำหรับแต่ละกระบวนการที่รันอยู่ให้ป้อน คำสั่งต่อไปนี้:

```
genld -l
```

- เมื่อต้องการดูรายการของกระบวนการที่มีเวอร์ชันเก่าของอ็อบเจกต์ที่โหลดให้ป้อน คำสั่งต่อไปนี้:

```
genld -u
```

- เมื่อต้องการดูรายการของกระบวนการที่มีเวอร์ชันเก่าของไลบรารี **libcrypt.a** ที่โหลดให้ป้อนคำสั่งต่อไปนี้:

```
genld -lu | grep -p libcrypt.a
```

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง genkex” ในหน้า 682

“คำสั่ง genld” ในหน้า 683

### ข้อมูลที่เกี่ยวข้อง:

การมอนิเตอร์และปรับคำสั่งและรูทีนย่อย

---

## คำสั่ง gennames

### Purpose

รวบรวมข้อมูลทั้งหมดที่จำเป็นสำหรับการรันคำสั่ง **filemon** and **netpmon** ในโหมดออฟไลน์

### Syntax

```
gennames[-f]
```

### รายละเอียด

คำสั่ง **gennames** รวบรวมข้อมูลการแม่พื้ชื่อกับแอดเดรสที่จำเป็นสำหรับคำสั่ง **filemon** และ **netpmon** เพื่อทำงานในโหมดออฟไลน์ ข้อมูลที่รวบรวมประกอบด้วย:

- รายการของส่วนขยายเคอร์เนลที่โหลดทั้งหมด คล้ายกับที่คำสั่ง `genkex` รายงาน
- รายการของไลบรารีที่แบ่งใช้ที่โหลดทั้งหมด คล้ายกับที่คำสั่ง `genkld` รายงาน
- รายการของกระบวนการที่โหลดทั้งหมด คล้ายกับที่คำสั่ง `genld` รายงาน
- สำหรับ `/unix` และส่วนขยายเคอร์เนลทั้งหมด และไลบรารีเอาต์พุตของคำสั่ง `stripnm -z` จะถูกรวบรวม

## แฟล็ก

ไอเท็ม คำอธิบาย  
 -f รวบรวมข้อมูลอุปกรณ์สำหรับสำหรับและโลจิคัลวอลุ่ม รวมทั้ง พิมพ์ข้อมูลระบบไฟล์เสมือนที่ใช้โดย filemon ออฟไลน์

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ตัวอย่าง

ในการรวบรวมข้อมูลที่เป็นสำหรับคำสั่ง `filemon` ในโหมดออฟไลน์ให้พิมพ์:

```
gennames -f > gen.out
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `genkex`” ในหน้า 682

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `netpmon`

คำสั่ง `stripnm`

การมอนิเตอร์และปรับคำสั่งและรูทีนย่อย

## คำสั่ง `gensyms`

### วัตถุประสงค์

รวมรวบรวมข้อมูลที่เป็นทั้งหมดเพื่อรันคำสั่ง `curt`, `splat` และ `tprof` ในโหมดออฟไลน์

### ไวยากรณ์

```
gensyms [-o] [-f] [-F] [-h] [-s] [-g] [I] [-N] [-k kernel] [-i file] [-b binary[,binary[,...]]] [-P pid[,pid[,...]]] [-S path]
```

### คำอธิบาย

คำสั่ง `gensyms` จะรวบรวมชื่อเพื่อระบุข้อมูลการแม็ปที่จำเป็นสำหรับ คำสั่ง `curt`, `splat` และ `tprof` เพื่อทำงาน ในโหมดออฟไลน์ ข้อมูลที่รวบรวมรวมถึงข้อมูลต่อไปนี้:

- รายการของส่วนขยายเคอร์เนลที่โหลดทั้งหมด
- รายการของไลบรารีแบบแบ่งใช้ที่โหลดทั้งหมด
- รายการของกระบวนการที่โหลดทั้งหมด
- สำหรับ `/unix`, ส่วนขยายเคอร์เนลทั้งหมด, ไลบรารี และไฟล์อ็อบเจกต์ทั้งหมดที่สอดคล้องกับกระบวนการ เอาต์พุตของคำสั่ง `stripnm` จะถูกรวบรวม

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-b binary</code>	ระบุรายการไบนารีทางเลือกที่จะใช้ ค้นหาสัญลักษณ์
<code>-f</code>	ไม่แสดงการพิมพ์ชื่อไฟล์ต้นฉบับ
<code>-F</code>	รวบรวมข้อมูลอุปกรณ์สำหรับสำหรับและ โจรจี้ลวอลุ่ม
<code>-g</code>	ถอดรหัสชื่อสัญลักษณ์
<code>-h</code>	พิมพ์ข้อความวิธีใช้
<code>-i file</code>	อ่านสัญลักษณ์จากไฟล์ที่ระบุ
<code>-I</code>	พิมพ์ค่าพื้นฐานสองของสัญลักษณ์
<code>-k kernel</code>	ระบุชื่อของเคอร์เนลอิมเมจ (ค่าดีฟอลต์ <code>/unix</code> )
<code>-N</code>	พิมพ์หมายเลขบรรทัดต้นทางของสัญลักษณ์
<code>-o</code>	พิมพ์ออฟเซตแทนแอดเดรส
<code>-P pid[,pid[,...]]</code>	พิมพ์สัญลักษณ์ของโมดูลที่ฟังพาที่โหลดโดยกระบวนการ ที่ระบุ แฟล็กนี้คือแฟล็กเพื่อเลือก
<code>-s</code>	ค้นหาสัญลักษณ์สำหรับไฟล์ที่ระบุโดยแฟล็ก <code>-k</code> และ <code>-b</code> เท่านั้น
<code>-S path</code>	ระบุรายการพาธการค้นหา ซึ่งใช้เพื่อ ค้นหาไบนารี

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ตัวอย่าง

1. เมื่อต้องการรวบรวมข้อมูลที่เป็นสำหรับคำสั่ง `tprof` ในโหมดดออปไลน์ ที่มีการทำโปรไฟล์ของการทดสอบโปรแกรมผู้ใช้ ให้ป้อนคำสั่งต่อไปนี้:

```
gensyms > test.syms
```

2. เมื่อต้องการรวบรวมข้อมูลเกี่ยวกับ ID กระบวนการที่ระบุ และกระบวนการที่ฟังพา ให้ป้อนคำสั่งต่อไปนี้:

```
gensyms -P pid > test.syms
```

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `gennames`” ในหน้า 685

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `splat`

คำสั่ง `tprof`

การมอนิเตอร์และปรับคำสั่งและรูทีนย่อย

# คำสั่ง gentun

## วัตถุประสงค์

สร้างนิยามช่องสัญญาณในฐานข้อมูลช่องสัญญาณ

## ไวยากรณ์

```
gentun -s src_host_IP_address -d dst_host_IP_address -v 4|6 [-t tun_type] [-m pkt_mode] [-t IBM] [-t manual] [-m tunnel] [-m transport] [-f fw_address] [-x dst_mask]] [-e [src_esp_algo]] [-a [src_ah_algo]] [-p src_policy] [-A [dst_ah_algo]] [-P dst_policy] [-k src_esp_key] [-h src_ah_key] [-K dst_esp_key] [-H dst_ah_key] [-n src_esp_spi] [-u src_ah_spi] [-N dst_esp_spi] [-U dst_ah_spi] [-b src_enc_mac_algo] [-c src_enc_mac_key] [-B dst_enc_mac_algo] [-C dst_enc_mac_key] [-g] [-z] [-E]
```

## คำอธิบาย

คำสั่ง **gentun** สร้างนิยามของช่องสัญญาณ ระหว่างโหนดโฮสต์และพาร์ทเนอร์โฮสต์ช่องสัญญาณ กฎตัวกรองที่สร้างขึ้นโดยอัตโนมัติ ที่เชื่อมโยงสำหรับช่องสัญญาณสามารถถูกเลือกสร้างโดยคำสั่งนี้

## แฟล็ก

ไอเท็ม	คำอธิบาย
-a	อัลกอริทึมการพิสูจน์ตัวตน ที่ใช้โดยต้นทางสำหรับการพิสูจน์ตัวตนแพ็กเก็ต IP ค่าที่ใช้ได้สำหรับ -a ขึ้นอยู่กับว่าอัลกอริทึมการพิสูจน์ตัวตน ใดถูกติดตั้งบนโฮสต์ รายการของอัลกอริทึมการพิสูจน์ตัวตน ทั้งหมดสามารถแสดงโดยเรียกใช้คำสั่ง <b>ipsecstat -A</b> ค่าที่พอลต์คือ HMAC_MD5 สำหรับช่องสัญญาณที่กำหนด ด้วยตนเอง
-A	(ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) อัลกอริทึมการพิสูจน์ตัวตน ใช้โดยปลายทางสำหรับการพิสูจน์ตัวตนแพ็กเก็ต IP ค่าที่ใช้ได้สำหรับ -A ขึ้นอยู่กับว่าอัลกอริทึมการพิสูจน์ตัวตน ใดถูกติดตั้งบนโฮสต์ รายการของอัลกอริทึมการพิสูจน์ตัวตน ทั้งหมดสามารถแสดงโดยเรียกใช้คำสั่ง <b>ipsecstat -A</b> หาก ไม่ใช้แฟล็กนี้ ค่าที่ใช้โดยแฟล็ก -a จะถูกใช้
-b	(ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) Source ESP Authentication Algorithm (รูปแบบส่วนหัวใหม่เท่านั้น) ค่าที่ใช้ได้สำหรับ -b ขึ้นอยู่กับว่าอัลกอริทึมการพิสูจน์ตัวตน ใดถูกติดตั้งบนโฮสต์ รายการของอัลกอริทึมการพิสูจน์ตัวตน ทั้งหมดสามารถแสดงโดยเรียกใช้คำสั่ง <b>ipsecstat -A</b>
-B	(ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) Destination ESP Authentication Algorithm (รูปแบบส่วนหัวใหม่เท่านั้น) ค่าที่ใช้ได้สำหรับ -B ขึ้นอยู่กับว่าอัลกอริทึมการพิสูจน์ตัวตน ใดถูกติดตั้งบนโฮสต์ รายการของอัลกอริทึมการพิสูจน์ตัวตนทั้งหมดสามารถ แสดงโดยเรียกใช้คำสั่ง <b>ipsecstat -A</b> หากไม่ใช้แฟล็กนี้ จะถูกตั้งค่าเป็นค่าเดียวกับแฟล็ก -b
-c	(ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) Source ESP Authentication Key (รูปแบบส่วนหัวใหม่เท่านั้น) โดยต้องเป็นสตริงเลขฐานสิบหกที่ขึ้นต้นด้วย "0x" หากไม่ใช้แฟล็กนี้ ระบบจะสร้างให้คุณ
-C	(ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) Destination ESP Authentication Key (รูปแบบส่วนหัวใหม่เท่านั้น) โดยต้องเป็นสตริงเลขฐานสิบหกที่ขึ้นต้นด้วย "0x" หากไม่ใช้แฟล็กนี้ จะถูกตั้งค่าเป็นค่าเดียวกับแฟล็ก -c
-d	IP แอดเดรสของโฮสต์ปลายทาง ในกรณีโฮสต์-โฮสต์ นี้คือ IP แอดเดรสของอินเตอร์เฟซโฮสต์ปลายทางที่จะใช้ช่องสัญญาณ ในกรณีโฮสต์-ไฟร์วอลล์-โฮสต์ นี้คือ IP แอดเดรสของโฮสต์ปลายทางที่อยู่เบื้องหลังไฟร์วอลล์ ชื่อโฮสต์สามารถใช้ได้ และ IP แอดเดรสแรกที่ใช้ส่งกลับโดยเนมเซิร์ฟเวอร์ สำหรับชื่อโฮสต์จะถูกใช้
-e	อัลกอริทึมการเข้ารหัส ใช้โดยต้นทางสำหรับการเข้ารหัสแพ็กเก็ต IP ค่าที่ใช้ได้สำหรับ -e ขึ้นอยู่กับว่าอัลกอริทึมการเข้ารหัส ใดถูกติดตั้งบนโฮสต์ รายการของอัลกอริทึมการเข้ารหัสทั้งหมดสามารถ แสดงโดยเรียกใช้คำสั่ง <b>ipsecstat -E</b>
-E	(ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) อัลกอริทึมการเข้ารหัส ใช้โดยปลายทางสำหรับการเข้ารหัสแพ็กเก็ต IP ค่าที่ใช้ได้สำหรับ -E ขึ้นอยู่กับว่าอัลกอริทึมการเข้ารหัส ใดถูกติดตั้งบนโฮสต์ รายการของอัลกอริทึมการเข้ารหัสทั้งหมดสามารถ แสดงโดยเรียกใช้คำสั่ง <b>ipsecstat -E</b> หากไม่ใช้แฟล็กนี้ ค่าที่ใช้โดยแฟล็ก -e จะถูกใช้
-f	IP แอดเดรสของไฟร์วอลล์จะอยู่ระหว่างโฮสต์ต้นทางและ ปลายทาง ช่องสัญญาณจะถูกสร้างขึ้นระหว่างโฮสต์นี้และไฟร์วอลล์ ดังนั้น นิยามช่องสัญญาณที่เกี่ยวข้องต้องถูกจัดทำบนไฟร์วอลล์ โฮสต์ ชื่อโฮสต์อาจถูกใช้สำหรับแฟล็กนี้และ IP แอดเดรสแรก ที่ส่งกลับโดยเนมเซิร์ฟเวอร์สำหรับชื่อโฮสต์นั้นจะถูกใช้

ไอเอ็ม

- g คำอธิบาย  
แฟล็กกฎตัวกรองที่ระบบสร้างโดยอัตโนมัติ หากไม่ใช้แฟล็กนี้ คำสั่งจะสร้างกฎตัวกรองสองข้อสำหรับช่องสัญญาณโดยอัตโนมัติ กฎตัวกรองที่สร้างโดยอัตโนมัติจะอนุญาตให้มีการรับส่งข้อมูล IP ระหว่างจุดปลายทางสองจุด ของช่องสัญญาณเพื่อผ่านทางช่องสัญญาณ
- h หากระบุแฟล็ก -g คำสั่งจะสร้างนิยามช่องสัญญาณเท่านั้น และผู้ใช้จะต้องเพิ่มกฎตัวกรองที่ใช้กำหนดเพื่อให้ช่องสัญญาณทำงานได้ นี่คือนิยาม AH Key String สำหรับช่องสัญญาณที่กำหนด ด้วยตนเอง อินพุตต้องเป็นสตริงเลขฐานสิบหกที่ขึ้นต้นด้วย "0x" หากไม่ใช้แฟล็กนี้ ระบบจะสร้างคีย์โดยใช้ตัวสร้างเลขสุ่ม
- H (ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) Key String สำหรับ AH ปลายทาง อินพุตต้องเป็นสตริงเลขฐานสิบหกที่ขึ้นต้นด้วย "0x" หากไม่ใช้แฟล็กนี้ ระบบจะสร้างคีย์โดยใช้ตัวสร้างเลขสุ่ม
- k นี่คือนิยาม ESP Key String สำหรับช่องสัญญาณที่กำหนด ด้วยตนเอง ถูกใช้โดยต้นทางเพื่อสร้างช่องสัญญาณ อินพุตต้องเป็นสตริง เลขฐานสิบหกที่ขึ้นต้นด้วย "0x" หากไม่ใช้แฟล็กนี้ ระบบจะสร้างคีย์โดยใช้ตัวสร้างเลขสุ่ม
- K (ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) Key String สำหรับ ESP ปลายทาง อินพุตต้องเป็นสตริง เลขฐานสิบหกที่ขึ้นต้นด้วย "0x" หากไม่ใช้แฟล็กนี้ ระบบจะสร้างคีย์โดยใช้ตัวสร้างเลขสุ่ม
- l Key Lifetime ระบุเป็นนาที

สำหรับช่องสัญญาณที่กำหนดด้วยตนเอง ค่านี้ระบุเวลาของการสามารถดำเนินการได้ก่อนที่ช่องสัญญาณ จะหมดอายุ

- m ค่าที่ใช้สำหรับช่องสัญญาณที่กำหนด ด้วยตนเอง คือ 0 - 44640 ค่า ระบุว่าช่องสัญญาณที่กำหนด ด้วยตนเอง จะไม่มีการหมดอายุ ค่าดีฟอลต์สำหรับช่องสัญญาณที่กำหนด ด้วยตนเอง คือ 480
- m Secure Packet Mode ค่านี้ต้องถูกระบุเป็น tunnel หรือ transport ค่าดีฟอลต์คือ tunnel โหมดช่องสัญญาณจะห่อหุ้มทั้งแพ็กเก็ต IP ขณะที่โหมดการส่งผ่าน จะห่อหุ้มเฉพาะส่วนข้อมูลของแพ็กเก็ต IP เท่านั้น เมื่อสร้างช่องสัญญาณโฮสต์-ไฟร์วอลล์-โฮสต์ (สำหรับโฮสต์ที่อยู่เบื้องหลังไฟร์วอลล์) ค่าของ tunnel ต้องใช้สำหรับแฟล็กนี้

- n แฟล็ก -m ถูกบังคับให้ใช้ค่าดีฟอลต์ (tunnel) หากระบุแฟล็ก -f (ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) Security Parameter Index สำหรับ ESP ต้นทาง นี้เป็นค่าตัวเลขที่ควบคู่กับ IP แอดเดรส ปลายทาง ระบุว่าการเชื่อมโยงการรักษาความปลอดภัยใดที่จะใช้สำหรับแพ็กเก็ต ที่ใช้ ESP หากไม่ใช้แฟล็กนี้ ระบบจะสร้าง SPI ให้แก่คุณ

- N (ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) Security Parameter Index สำหรับ ESP ปลายทาง โดยต้องถูกป้อนสำหรับช่องสัญญาณที่กำหนด ด้วยตนเอง หากระบุนโยบายในแฟล็ก -P ที่มี ESP แฟล็กนี้ไม่ใช่กับช่องสัญญาณ IBM
- p นโยบายต้นทาง ระบุวิธีในการพิสูจน์ตัวตนแพ็กเก็ต IP และ/หรือ การเข้ารหัสที่จะใช้โดยโฮสต์นี้ หากระบุเป็น ea แพ็กเก็ต IP จะถูกเข้ารหัสก่อนการพิสูจน์ตัวตน หากระบุเป็น ae จะถูกเข้ารหัสหลังการพิสูจน์ตัวตน โดยที่การระบุ e ตามลำพัง หรือ a ตามลำพังสอดคล้องกับแพ็กเก็ต IP ที่ถูกเข้ารหัสเท่านั้นหรือถูกพิสูจน์ตัวตนเท่านั้น ค่าดีฟอลต์ สำหรับแฟล็กนี้จะขึ้นอยู่กับว่ากำหนดแฟล็ก -e และ -a หรือไม่ นโยบายดีฟอลต์จะเป็น ea หากทั้งสองหรือไม่กำหนดทั้งแฟล็ก -e และ -a มิฉะนั้น นโยบายจะแสดงผลว่าแฟล็กใดของ -e และ -a ที่จะถูกกำหนด

- P (ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) นโยบายปลายทาง ระบุวิธีการพิสูจน์ตัวตน และ/หรือการเข้ารหัสแพ็กเก็ต IP ที่จะใช้โดยปลายทาง หากระบุเป็น ea แพ็กเก็ต IP จะถูกเข้ารหัสก่อนการพิสูจน์ตัวตน ถ้าระบุเป็น ae จะถูกเข้ารหัส หลังการพิสูจน์ตัวตน ขณะที่การระบุ e หรือ a สอดคล้อง กับแพ็กเก็ต IP ที่จะถูกเข้ารหัสเท่านั้น หรือถูกพิสูจน์ตัวตนเท่านั้น นโยบายดีฟอลต์จะเป็น ea หากทั้งสองหรือไม่กำหนดทั้งแฟล็ก -E และ -A มิฉะนั้น นโยบายจะแสดงผลว่าแฟล็กใดของ -E และ -A ที่จะถูกระบุ
- s IP แอดเดรสของโฮสต์ต้นทาง, IP แอดเดรสของโหนดโฮสต์อินเตอร์เฟสจะถูก ใช้โดยช่องสัญญาณ ชื่อโฮสต์สามารถใช้ได้ และ IP แอดเดรสแรกที่ส่งกลับโดยเนมเซิร์ฟเวอร์ สำหรับชื่อโฮสต์จะถูกใช้
- t ชนิดของสัญญาณ ตอระบุเป็น manual

คีย์ช่องสัญญาณเริ่มต้นและการอัปเดตคีย์ภายหลังใดๆ จำเป็นต้องถูกดำเนินการ ด้วยตนเองเมื่อใช้ช่องสัญญาณที่กำหนด ด้วยตนเอง เมื่อคีย์ ถูกติดตั้งด้วยตนเอง คีย์เดียวกันจะถูกใช้สำหรับการดำเนินการช่องสัญญาณทั้งหมดจนกว่า จะถูกเปลี่ยนด้วยตนเอง

- u ค่าช่องสัญญาณที่กำหนด ด้วยตนเอง ควรถูกเลือกเมื่อคุณต้องการสร้างช่องสัญญาณด้วยโฮสต์ IP Security ที่ไม่ใช่ IBM หรือจุดปลาย IP version 6 ใดๆ โดยที่จุดปลายสนับสนุน RFCs 1825 - 1829 หรือ IETF drafts สำหรับรูปแบบ IP Security encapsulation ใหม่สำหรับช่องสัญญาณ IP
- u (ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) Security Parameter Index สำหรับ AH ต้นทาง ใช้ SPI และ IP แอดเดรสปลายทางเพื่อ กำหนดว่า การเชื่อมโยงการรักษาความปลอดภัยแบบใดที่จะใช้สำหรับ AH หากไม่ใช้แฟล็กนี้ ค่าของ -n SPI จะถูกใช้
- U (ช่องสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) Security Parameter Index สำหรับ AH ปลายทาง หากไม่ใช้แฟล็กนี้จะใช้ -N spi
- v เวอร์ชัน IP ที่ช่องสัญญาณจะถูกสร้าง สำหรับช่องสัญญาณ IP version 4 ใช้ค่า 4 สำหรับช่องสัญญาณ IP version 6 tunnels ใช้ ค่า 6
- x เน็ตเวิร์กมาร์สำหรับเน็ตเวิร์กที่ได้รับการรักษาความปลอดภัยหลังไฟร์วอลล์ โฮสต์ ปลายทางคือสมาชิกของเน็ตเวิร์กที่ได้รับการรักษาความปลอดภัย การใช้ -d และ -x รวมกันอนุญาตให้โฮสต์ต้นทางสื่อสาร กับหลายๆ โฮสต์ ในเน็ตเวิร์กที่มีการรักษาความปลอดภัยผ่านทางฮอรัสที่มีการรักษาความปลอดภัยด้วยไฟร์วอลล์ ซึ่งต้องเป็นโหมดช่องสัญญาณ

แฟล็กนี้จะใช้โดยอัตโนมัติเมื่อใช้แฟล็ก -f

ไอเอ็ม	คำอธิบาย
-y	(ของสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) แฟล็กการป้องกันการทำซ้ำ การป้องกันการซ้ำจะใช้ได้ต่อเมื่อส่วนหัว ESP หรือ AH กำลังใช้รูปแบบ ส่วนหัวแบบใหม่ (ดูที่แฟล็ก -z) ค่าที่ใช้ได้สำหรับ แฟล็ก -y คือ Y (ใช่) และ N (ไม่ใช่) การห่อหุ้มทั้งหมด ที่ใช้ในช่องสัญญาณนี้ (AH, ESP, การส่ง และการรับ) จะใช้ ฟิวด์ replay หากค่าแฟล็กนี้เป็น Y ค่าดีฟอลต์คือ N
-z	(ของสัญญาณที่กำหนดด้วยตนเอง เท่านั้น) แฟล็กรูปแบบส่วนหัวใหม่ รูปแบบส่วนหัวใหม่ช่วยสงวนฟิวด์ในส่วนหัว ESP และ AH สำหรับการป้องกัน การทำซ้ำ และยังอนุญาตให้ทำการพิสูจน์ตัวตน ESP ฟิวด์ replay จะใช้ ต่อเมื่อแฟล็ก replay (-y) ถูกตั้งค่าเป็น Y ค่าที่ใช้ได้สำหรับแฟล็ก -z คือ Y (ใช่) และ N (ไม่ใช่) ค่าดีฟอลต์เมื่อไม่ใช้แฟล็ก -z จะขึ้นอยู่กับ อัลกอริทึมที่คุณเลือกสำหรับช่องสัญญาณ โดยมีค่าดีฟอลต์เป็น N ยกเว้นว่าใช้อัลกอริทึมอื่นนอกเหนือจาก KEYED_MD5 สำหรับแฟล็ก -a หรือ -A หรือหากใช้แฟล็ก -b หรือ -B

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **Issecattr** command or the **getcmdattr** subcommand.

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `exptun`” ในหน้า 478

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `chtun`

คำสั่ง `imptun`

คำสั่ง `lstun`

## คำสั่ง `genxlt`

### วัตถุประสงค์

สร้างตารางการแปลงชุดโค้ดสำหรับใช้ โดยไลบรารี `lconv`

### ไวยากรณ์

`genxlt` [OutputFile ]

### คำอธิบาย

คำสั่ง `genxlt` อ่าน ไฟล์ตารางการแปลงชุดซอร์สโค้ดจากอินพุตมาตรฐานและเขียน เวอร์ชันคอมไพล์ของไฟล์ที่ระบุโดยพารามิเตอร์ `OutputFile` หากไม่ระบุค่าสำหรับพารามิเตอร์ `OutputFile` จะใช้เอาต์พุตมาตรฐาน ไฟล์ตารางการแปลงชุดซอร์สโค้ดมีส่วนทำงานที่ทำหน้าที่ตามคำสั่ง `genxlt` เพื่อสร้างเวอร์ชันคอมไพล์

รูปแบบของไฟล์ต้นฉบับตารางการแปลงชุดโค้ดคือ:

- บรรทัดที่มีอักขระ nonwhite space เริ่มต้น คือ # (เครื่องหมายสี่เหลี่ยม) ถูกถือเป็นบรรทัดความคิดเห็น
- บรรทัด Null และบรรทัดที่มีเฉพาะอักขระ white-space เท่านั้นจะถูกถือเป็นบรรทัดความคิดเห็น
- บรรทัดที่ไม่ใช่ความคิดเห็นต้องอยู่ในรูปแบบ ต่อไปนี้:

```

%token <blank> # <tab> and <space>
%token <hex>   # <zero>, <one>, <two>, <three>, <four>,
               # <five>, <six>, <seven>, <eight>, <nine>,
               # <a>, <b>, <c>, <d>, <e>, <f>,
               # <A>, <B>, <C>, <D>, <E>, <F>,
%token <any>  # any character but '\n'
line   : offset blank value blank comment '\n'
       | 'SUB' blank value blank comment '\n'
       ;

blank  : <blank>
       | blank <blank>
       ;

offset : '0x' <hex>
       | offset <hex>
       ;

value  : offset
       | 'invalid'
       | 'substitution'
       ;

comment : '#' <any>
        | comment <any>
        ;

```

บรรทัดที่ออฟเซตเป็น 'SUB' ถูกใช้เพื่อระบุอักขระการแทนค่าดีฟอลต์

หากตารางถูกตั้งค่าเป็น 'substitution' ตัวแปลง **iconv** ที่ใช้ตารางนี้จะใช้ค่า SUB สำหรับออฟเซตนี้

หากค่าถูกตั้งค่าเป็น 'invalid' ตัวแปลง **iconv** ที่ใช้ตารางนี้จะส่งคืน ข้อผิดพลาดสำหรับออฟเซต

หากพบออฟเซตในไฟล์ตารางการแปลง ชุดซอร์สโค้ดหลายครั้ง รายการสุดท้ายจะถูกใช้ในการคอมไพล์ของตารางการแปลง

ออฟเซตและค่าต้องอยู่ในช่วง 0x00 ถึง 0xff

ต่อไปนี้เป็นส่วนของตารางการแปลงชุดโค้ด ที่ตัดตอนมา:

SUB	0x1a	substitute character
0x80	0xc7	C cedilla
0x81	0xfc	u diaeresis
0x82	0xe9	e acute
0x83	0xe2	a circumflex
0x84	0xe4	a diaeresis
0x85	0x40	a grave
0x9F		substitution
0xff		invalid

หากสำเร็จ คำสั่ง `genxlt` จะออกพร้อมค่า 0 หากไม่สามารถเปิดเอาต์พุตไฟล์ คำสั่ง `genxlt` จะดำเนินการไม่สำเร็จ และออกพร้อมค่า 1 หากตรวจพบข้อผิดพลาดทางไวยากรณ์ในอินพุตสตรีม คำสั่ง `genxlt` จะออกในทันทีด้วยค่า 2 และเขียนหมายเลขบรรทัดที่เกิดข้อผิดพลาดทางไวยากรณ์ไปยังข้อผิดพลาดมาตรฐาน

ชื่อของไฟล์ที่สร้างโดยคำสั่ง `genxlt` ต้องเป็นไปตามระเบียบการตั้งค่าด้านล่างเพื่อให้ระบบย่อย `iconv` รู้จักว่าเป็นไฟล์การแปลง:

```
fromcode: "IBM-850"  
tocode: "ISO8859-1"  
conversion table file: "IBM-850_ISO8859-1"
```

ชื่อไฟล์ตารางการแปลงถูกจัดรูปแบบโดยการต่อชื่อไฟล์ `tocode` เข้ากับชื่อไฟล์ `fromcode` โดยมีอักขระขีดเส้นใต้ ระหว่างชื่อทั้งสอง

## ตัวอย่าง

ในการสร้างตารางการแปลงชุดโค้ดที่ผู้ใช้กำหนดเอง ที่ไม่ใช่ภาษาอังกฤษ ให้ป้อน:

```
cp /usr/lib/nls/loc/iconvTable/ISO8859-1-IBM-850_src $HOME  
vi $HOME/ISO8859-1-IBM-850_src  
genxlt < $HOME/ISO8859-1-IBM-850_src > cs1_cs2
```

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `iconv`

คำสั่ง `iconv_open`

คำสั่ง `iconv_close`

ภาพรวมตัวแปลงสำหรับโปรแกรมมิง

---

## คำสั่ง `get`

### วัตถุประสงค์

สร้างเวอร์ชันที่ระบุของไฟล์ SCCS

### ไวยากรณ์

ในการรับไฟล์ SCCS เวอร์ชันอ่านอย่างเดียว

```
get [-g] [-m] [-n] [-p] [-s] [-c Cutoff] [-i List] [-r SID] [-t] [-x List] [-w String] [-l [p]] [-L] File ...
```

ในการรับไฟล์ SCCS เวอร์ชันแก้ไขได้

```
get [-e] [-k] [-b] [-s] [-c Cutoff] [-i List] [-r SID] [-t] [-x List] [-l [p]] [-L] File ...
```

### คำอธิบาย

คำสั่ง `get` อ่าน เวอร์ชันที่ระบุของไฟล์ Source Code Control System (SCCS) และสร้างไฟล์ข้อความ ASCII ตามแฟล็กที่ระบุ จากนั้นคำสั่ง `get` จะเขียนไฟล์ข้อความแต่ละไฟล์ไปยังไฟล์ ที่มีชื่อเหมือนกับไฟล์ SCCS ต้นฉบับแต่ไม่มี `s`. นำหน้า (`g-file`)

แฟล็กและไฟล์สามารถระบุในลำดับใดๆ และแฟล็กทั้งหมดจะใช้กับไฟล์ที่ระบุชื่อทั้งหมด หากคุณระบุได้เรียกทอรีสำหรับ พารามิเตอร์ *File* คำสั่ง `get` จะดำเนินการตามที่ร้องขอกับไฟล์ทั้งหมดในไดเรกทอรีที่ขึ้นต้นด้วยค่านำหน้า *s*. หากคุณระบุ `-` (เครื่องหมายลบ) สำหรับพารามิเตอร์ *File* คำสั่ง `get` จะอ่านอินพุตมาตรฐานและแปลความหมาย แต่ละไฟล์เป็นชื่อของไฟล์ SCCS คำสั่ง `get` ยังคงอ่านอินพุตไปจนกระทั่งอ่านพบอักขระสิ้นสุดไฟล์

หากผู้ใช้ที่ใช้งานอยู่มีสิทธิ์ในการเขียน ในไดเรกทอรีที่มีไฟล์ SCCS แต่ผู้ใช้จริงไม่มี ดังนั้นมีเพียงหนึ่งไฟล์เท่านั้นที่สามารถ ระบุชื่อเมื่อใช้แฟล็ก `-e`

**หมายเหตุ:** คำสั่ง `get` สนับสนุน Multibyte Character Set (MBCS) สำหรับชื่อไฟล์และข้อมูลสตริงที่ระบุ ด้วยแฟล็ก `w`

### การรับเวอร์ชันไฟล์อ่านอย่างเดียว

คำสั่ง `get` สร้าง ไฟล์ทั้งเวอร์ชันอ่านอย่างเดียวและเวอร์ชันแก้ไขได้ ไฟล์เวอร์ชัน อ่านอย่างเดียวควรถูกใช้หากแอปพลิเคชันไม่ จำเป็นต้อง เปลี่ยนแปลงเนื้อหาไฟล์ ซอร์สโค้ดไฟล์เวอร์ชันอ่านอย่างเดียว สามารถคอมไพล์ได้ ไฟล์ข้อความสามารถถูกแสดง หรือพิมพ์จากเวอร์ชัน อ่านอย่างเดียว

ความแตกต่างระหว่างเวอร์ชันแก้ไขได้และ อ่านอย่างเดียวเป็นสิ่งสำคัญเมื่อใช้คีย์เวิร์ด *identification คีย์เวิร์ด Identification* คือสัญลักษณ์ที่ขยายไปยัง ค่าข้อความบางอย่างเมื่อคำสั่ง `get` เรียกข้อมูล ไฟล์เป็นแบบอ่านอย่างเดียว ในเวอร์ชันแก้ไขได้ คีย์ เวิร์ดจะไม่ถูกขยาย คีย์เวิร์ด *Identification* สามารถปรากฏในที่ใดก็ได้ในไฟล์ SCCS โปรดดูที่คำสั่ง `prs` สำหรับข้อมูลเพิ่มเติม เกี่ยวกับคีย์เวิร์ด *identification*

### ไฟล์ SCCS

นอกเหนือจากไฟล์ที่มีค่านำหน้า *s*. (*s-file*) แล้วคำสั่ง `get` ยังสร้างไฟล์เสริม: *g-file*, *l-file*, *p-file* และ *z-file* ไฟล์เหล่านี้ถูก ระบุโดยแท็ก ซึ่งเป็นตัวอักษรหน้าเครื่องหมายขีดคั่น โปรแกรม `get` ตั้งชื่อไฟล์เสริมโดยการแทนที่ *s*. นำหน้าใน ชื่อไฟล์ SCCS ด้วยแท็กที่เหมาะสม ยกเว้นสำหรับ *g-file* ซึ่งกำหนดชื่อด้วยการลบค่านำหน้า *s*. ออก ดังนั้น สำหรับไฟล์ชื่อ *s.sample* ชื่อไฟล์ เสริมจะเป็น *sample*, *l.sample*, *p.sample* และ *z.sample*

ไฟล์เหล่านี้ทำหน้าที่ตามวัตถุประสงค์ต่อไปนี้:

ไอเท็ม	คำอธิบาย
<i>s-file</i>	มีข้อความไฟล์ต้นฉบับและการเปลี่ยนแปลงทั้งหมด (เดลตา) ที่ทำกับไฟล์ รวมทั้งมีข้อมูลเกี่ยวกับบุคคลที่สามารถเปลี่ยนแปลง เนื้อหาไฟล์ บุคคลที่ทำการเปลี่ยนแปลง เวลาที่ทำการเปลี่ยนแปลง และลักษณะของการเปลี่ยนแปลง คุณไม่สามารถแก้ไขไฟล์นี้โดยตรง เนื่องจากเป็นแบบอ่านอย่างเดียว อย่างไรก็ตาม ไฟล์จะมีข้อมูลที่สำคัญสำหรับคำสั่ง SCCS ในการสร้าง <i>g-file</i> ที่คุณสามารถแก้ไขได้
<i>g-file</i>	ไฟล์ข้อความ ASCII ที่มีข้อความของเวอร์ชันไฟล์ SCCS ที่คุณระบุด้วยแฟล็ก <code>-r</code> (หรือ เวอร์ชัน trunk ล่าสุดเป็นค่าดีฟอลต์) คุณสามารถแก้ไขไฟล์นี้ได้โดยตรง เมื่อคุณทำการเปลี่ยนแปลงทั้งหมดแล้ว และต้องการสร้างเดลตาใหม่กับ ไฟล์ คุณสามารถรันคำสั่ง <code>delta</code> บนไฟล์ คำสั่ง <code>get</code> จะสร้าง <i>g-file</i> ขึ้นในไดเรกทอรีปัจจุบัน
	เมื่อใดที่ รันคำสั่ง <code>get</code> จะสร้าง <i>g-file</i> ยกเว้น วาระบุแฟล็ก <code>-g</code> หรือแฟล็ก <code>-p</code> ผู้ใช้ที่แท้จริงจะเป็นเจ้าของไฟล์ (ไม่ใช่ผู้ใช้ที่ใช้งานอยู่) หากคุณ ไม่ระบุแฟล็ก <code>-k</code> หรือ <code>-e</code> ไฟล์จะเป็นแบบอ่านอย่างเดียว หากระบุแฟล็ก <code>-k</code> หรือ <code>-e</code> เจ้าของจะมีสิทธิ์ในการเขียน สำหรับ <i>g-file</i> คุณต้องมีสิทธิ์ในการเขียน ในไดเรกทอรีปัจจุบันเพื่อสร้าง <i>g-file</i>

ไอเท็ม  
I-file

คำอธิบาย

คำสั่ง `get` สร้าง I-file เมื่อระบุแฟล็ก `-I` เป็นไฟล์อ่านอย่างเดียว โดยมีตารางแสดงว่าเดลตาใดถูกใช้ในการสร้าง `g-file` คุณต้องมีสิทธิ์ในการเขียนในไดเรกทอรีปัจจุบันเพื่อสร้าง I-file บรรทัดใน I-file มีการจัดรูปแบบต่อไปนี้:

- อักขระว่างหากเดลตาถูกนำใช้ มิฉะนั้นจะเป็นเครื่องหมายดอกจัน
- อักขระว่างหากเดลตาถูกนำใช้หรือไม่ถูกนำใช้และถูกข้าม เครื่องหมายดอกจันจะปรากฏหากเดลตาไม่ถูกนำใช้และไม่ถูกข้าม
- โค้ดที่ระบุเหตุผลพิเศษว่าทำไมเดลตาจึงถึงนำใช้หรือไม่ถูกนำใช้:

Blankspace

รวมหรือไม่รวมตามปกติ

I รวมโดยใช้แฟล็ก `-i`

X ไม่รวมโดยใช้แฟล็ก `-x`

C ตัดทิ้งโดยใช้แฟล็ก `-c`

- SID
- วันที่และเวลาที่ไฟล์ถูกสร้างขึ้น
- ชื่อล็อกอินของบุคคลที่สร้างเดลตา

ความคิดเห็นและ ข้อมูล Modification Requests (MR) ตามมาบนบรรทัดถัดมา โดยเยื้องหนึ่งอักขระแท็บแนวนอน บรรทัดว่างสิ้นสุดแต่ละรายการ ตัวอย่างเช่น สำหรับการตัดทิ้งเดลตาด้วย แฟล็ก `-c` รายการใน I-file จะเป็น:

```
**C 1.3 85/03/13 12:44:16 pat
```

และรายการสำหรับเดลตาเริ่มต้นจะเป็น:

```
1.1 85/02/27 15:42:20 pat
date and time created 85/02/27 15:42:20 by pat
```

p-file

คำสั่ง `get` สร้าง p-file เมื่อระบุแฟล็ก `-e` หรือ `-k` p-file ส่ง ข้อมูลผลลัพธ์จากคำสั่ง `get -e` ไปยังคำสั่ง `delta` p-file ยัง ป้องกันการเรียกใช้งานที่จะเกิดในภายหลังของคำสั่ง `get -e` สำหรับ SID เดียวกันจนกว่า `delta` จะถูกรัน หรืออักขระคีย์การแก้ไขร่วม (j) ถูกตั้งค่าในไฟล์ SCCS ตัวอักขระคีย์ j อนุญาตให้คำสั่ง `get` หลายคำสั่งถูกรันบน SID เดียวกัน p-file ถูกสร้างขึ้นในไดเรกทอรีที่มี ไฟล์ SCCS ในการสร้าง p-file ใน ไดเรกทอรี SCCS คุณต้องมีสิทธิ์ในการเขียนในไดเรกทอรีนั้น โค้ดสิทธิ์ของ p-file เป็นแบบอ่านอย่างเดียว กับทุกไฟล์ยกเว้นเจ้าของไฟล์ และเป็นเจ้าของโดยผู้ใช้ที่ใช้งานอยู่ p-file ไม่ควรถูกแก้ไขโดยตรงโดยเจ้าของ p-file ประกอบด้วย:

- Current SID
- SID ของเดลตาใหม่ที่จะสร้าง
- ชื่อผู้ใช้
- วันที่และเวลาของคำสั่ง `get`
- แฟล็ก `-i` หากแสดงอยู่
- แฟล็ก `-x` หากแสดงอยู่

z-file

p-file มี รายการที่มีข้อมูลนำหน้าสำหรับแต่ละเดลตาที่ค้างอยู่ สำหรับไฟล์ ต้องไม่มีสองบรรทัดไหนที่มี SID เดลตาใหม่เหมือนกัน z-file เป็นกลไกการล็อก การอัปเดตพร้อมกัน z-file มีหมายเลข กระบวนการฐานสองของคำสั่ง `get` ที่สร้างไฟล์ ไฟล์นี้ถูกสร้างขึ้นในไดเรกทอรีที่มีไฟล์ SCCS และมีอยู่ขณะคำสั่ง `get` กำลัง รันเท่านั้น

เมื่อคุณใช้คำสั่ง `get` คำสั่งจะแสดง SID ที่กำลังถูกเข้าถึงและจำนวนบรรทัดที่สร้าง จากไฟล์ SCCS หากคุณระบุแฟล็ก `-e` SID ของเดลตาที่จะถูกทำจะปรากฏหลังจาก SID ถูกเข้าถึง และก่อนที่บรรทัดจำนวนหนึ่งจะถูกสร้างขึ้น หากคุณระบุมากกว่าหนึ่งไฟล์ ไดเรกทอรี หรืออินพุตมาตรฐาน คำสั่ง `get` จะแสดงชื่อไฟล์ก่อนหน้าแต่ละไฟล์จะถูกประมวลผล หากคุณระบุแฟล็ก `-i` คำสั่ง `get` จะแสดงรายการเดลตาที่มีด้านล่างคำว่า Included หากคุณระบุแฟล็ก `-x` คำสั่ง `get` จะแสดงรายการเดลตาที่ไม่มีด้านล่างคำว่า Excluded

ตารางต่อไปนี้จะแสดงวิธีที่คำสั่ง `get` พิจารณาทั้ง SID ของไฟล์ที่เรียกข้อมูล และ SID ที่ค้างอยู่ คอลัมน์ SID Specified จะแสดงแนวทางต่างๆ ที่สามารถระบุ SID ด้วยแฟล็ก `-r` คอลัมน์แรกยังแสดงเงื่อนไขต่างๆ ที่สามารถมีได้ รวมทั้งว่าจะใช้แฟล็ก `-b` กับแฟล็ก `get -e` หรือไม่ คอลัมน์ SID Retrieved ระบุ SID ของไฟล์ที่ประกอบเป็น `g-file` คอลัมน์ SID of Delta to Be Created ระบุ SID ของเวอร์ชัน ที่จะถูกสร้างเมื่อใช้คำสั่ง `delta`

SID Determination

SID Specified	SID Retrieved	SID of Delta to Be Created
none <sup>1</sup> -b Used? no เงื่อนไขอื่น R ดีพอลดเป็น mR <sup>2</sup>	mR.mL	mR.(mL+1)
none <sup>1</sup> -b Used? yes เงื่อนไขอื่น R ดีพอลดเป็น mR	mR.mL	mR.mL.(mB+1).1
R -b Used? no เงื่อนไขอื่น R>mR	mR.mL	R.1 <sup>3</sup>
R -b Used? no เงื่อนไขอื่น R=mR	mR.mL	mR.(mL+1)
R -b Used? yes เงื่อนไขอื่น R>mR	mR.mL	mR.mL.(mB+1).1
R -b Used? yes เงื่อนไขอื่น R=mR	mR.mL	mR.mL.(mB+1).1
R -b Used? N/A เงื่อนไขอื่น R<mR และ R ไม่มีอยู่	hR.mL <sup>4</sup>	hR.mL.(mB+1).1

SID Determination

SID Specified	SID Retrieved	SID of Delta to Be Created
R -b Used? N/A เงื่อนไขอื่น ตัวตามหลัง Trunk ในรีลีส > R และมีอยู่	R.mL	R.mL.(mB+1).1
R.L. -b Used? no เงื่อนไขอื่น ไม่มี trunk successor	R.L.	R.(L+1)
R.L. -b Used? yes เงื่อนไขอื่น ไม่มี trunk successor	R.L.	R.L(mB+1).1
R.L. -b Used? N/A เงื่อนไขอื่น ตัวตามหลัง Trunk ในรีลีส > หรือ = R	R.L.	R.L.(mB+1).1
R.L.B. -b Used? no เงื่อนไขอื่น ไม่มี branch successor	R.L.B.mS	R.L.B.(mS+1)
R.L.B. -b Used? yes เงื่อนไขอื่น ไม่มี branch successor	R.L.B.mS	R.L.(mB+1).1
R.L.B.S. -b Used? no เงื่อนไขอื่น ไม่มี branch successor	R.L.B.S.	R.L.B.(S+1)
R.L.B.S. -b Used? yes เงื่อนไขอื่น ไม่มี branch successor	R.L.B.S.	R.L.(mB+1).1

SID Determination

SID Specified	SID Retrieved	SID of Delta to Be Created
R.L.B.S. -b Used? N/A เงื่อนไขอื่น Branch successor	R.L.B.S.	R.L.(mB+1).1

หมายเหตุ: ในตาราง SID Determination ตัวอักษร R, L, B และ S คือรหัส, ระดับ, สาขา, และลำดับคอมโพเนนต์ของ SID ตัวอักษร m แสดง ค่าสูงสุด

<sup>1</sup> ใช้ต่อเมื่อ แฟล็ก -d (SID ดีฟอลต์) ไม่มีอยู่ใน ไฟล์ (โปรดดูที่คำสั่ง admin)

<sup>2</sup> mR ระบุ รหัสที่มีอยู่สูงสุด

<sup>3</sup> บังคับให้ทำการสร้าง เดลตาแรกในรหัสใหม่

<sup>4</sup> hR เป็น รหัสที่มีอยู่สูงสุด ต่ำกว่าที่ระบุ รหัส R ไม่มีอยู่

**คีย์เวิร์ด Identification**

การระบุข้อมูลถูกแทรกใน ข้อความที่เรียกข้อมูลมาจากไฟล์ SCCS โดยการแทนค่าคีย์เวิร์ด identification ด้วยค่าของคีย์เวิร์ด ในตำแหน่งที่มีคีย์เวิร์ด คีย์เวิร์ดต่อไปนี้จะถูกใช้ในข้อความที่เก็บไฟล์ไฟล์ SCCS:

คีย์เวิร์ด	Value
%M%	ชื่อโมดูล อาจเป็นค่าของแฟล็ก m ในไฟล์ หรือหากไม่มี จะเป็นชื่อของไฟล์ SCCS ที่ s. ถูกลบออก
%I%	SCCS identification (SID) (%R%.%L% or %R%.%L%.%B%.%S%) ของข้อความที่เรียกข้อมูลมา
%R%	รหัส
%L%	ระดับ
%B%	สาขา
%S%	ลำดับ
%D%	วันที่ ปัจจุบัน จัดรูปแบบเป็น YY/MM/DD
%H%	วันที่ ปัจจุบัน จัดรูปแบบเป็น MM/DD/YY
%T%	เวลา ปัจจุบัน จัดรูปแบบเป็น HH:MM:SS
%E%	ส่วนตงใหม่สุดที่ใช้กับวันที่ที่สร้างขึ้น จัดรูปแบบเป็น YY/MM/DD
%G%	ส่วนตงใหม่สุดที่ใช้กับวันที่ที่สร้างขึ้น จัดรูปแบบเป็น MM/DD/YY
01:51:20	ส่วนตงที่ใช้กับเวลาใหม่สุดถูกสร้างขึ้น จัดรูปแบบเป็น HH:MM:SS
%Y%	ชนิดโมดูล: ค่าของแฟล็ก t ใน ไฟล์ SCCS
%F%	ชื่อไฟล์ SCCS
%P%	ชื่อพารามิเตอร์ SCCS
%Q%	ค่าของแฟล็ก -q ในไฟล์
%C%	หมายเลขบรรทัด ปัจจุบัน คีย์เวิร์ดนี้ใช้สำหรับระบุ เอาต์พุตข้อความโดยโปรแกรม เช่นข้อความแสดงความผิดพลาด this should not have happened %C% ไม่ได้ เจตนาใช้กับทุกบรรทัดเพื่อกำหนดหมายเลขลำดับ
%Z%	สตริงสื่อกระแส @(#) ที่ รู้จักโดย what
%W%	รูปแบบย่อสำหรับการสร้างสตริง what: %W% = %Z%%M%<tab>%I%
%A%	รูปแบบย่ออีกรูปแบบหนึ่งสำหรับการสร้างสตริง what: %A% = %Z%%Y% %M% %I%%Z%

**แฟล็ก**

ไอเท็ม	คำอธิบาย
-b	ระบุว่าเดลตาที่จะถูกสร้างควรมี SID ในสาขาใหม่ SID ใหม่ถูกกำหนดหมายเลขตามกฎที่กำหนด ในตาราง SID determination คุณสามารถใช้แฟล็ก -b กับแฟล็ก -e เท่านั้น ซึ่งจำเป็นต่อเมื่อ คุณต้องการแตกสาขาจากเดลตาปลาย (เดลตาที่ไม่มีเดลตาต่อไป) การพยายามสร้างเดลตาที่เดลตาที่ไม่ใช่เดลตาปลายโดยอัตโนมัติส่งผลให้เกิดสาขา แม้ว่าแฟล็กส่วนหัว b จะไม่ ถูกตั้งค่า หากคุณไม่ระบุแฟล็กส่วนหัว b ในไฟล์ SCCS คำสั่ง get จะข้ามแฟล็ก -b เนื่องจากไฟล์ไม่อนุญาตให้มีการแตกสาขา
-c Cutoff	ระบุวันที่และเวลาที่ตัดทิ้ง ในรูปของ YY[MM][DD][HH[MM][SS]]] คำสั่ง get ไม่รวมเดลตาในไฟล์ SCCS ที่สร้างหลังจากการตัดทิ้งที่ระบุใน g-file คำ ของเทอมที่ไม่ระบุใดๆ ในตัวแปร Cutoff จะมีค่าตีฟอลต์เป็นค่าที่อนุญาตสูงสุด ตั้งแต่วันที่และเวลาที่ตัดทิ้งที่ระบุเฉพาะปี (YY) จะระบุเดือน วัน ชั่วโมง นาที และวินาทีสุดท้ายของปีนั้น ตัวเลขใดๆ ของอักขระที่ไม่ใช่ ตัวเลขสามารถค้นค่าสองหลักของตัวแปร Cutoff วันที่และเวลา ค่านี้อนุญาตให้คุณระบุวันที่และเวลาในรูปแบบ ตัวเลข ดังนี้: <pre>-c85/9/2,9:00:00 -c"85/9/2 9:00:00" -c85/9/2 9:00:00"</pre>
-e	ระบุว่า g-file ที่กำลังถูกสร้างจะถูกแก้ไขโดย ผู้ใช้ที่ ใช้คำสั่ง get การเปลี่ยนแปลง จะถูกบันทึกภายหลังด้วยคำสั่ง delta คำสั่ง get -e สร้าง p-file ที่ป้องกันผู้ใช้คนอื่น ๆ มิให้เรียกใช้คำสั่ง get -e และ ป้องกันการแก้ไข g-file ที่สองที่มี SID เดียวกันก่อนที่คำสั่ง delta จะถูกรัน เจ้าของไฟล์สามารถแทนที่ข้อจำกัดนี้ได้โดยการอนุญาต ให้ทำการแก้ไขรวมใน SID เดียวกันผ่านทาง การใช้คำสั่ง admin ที่มี แฟล็ก -fj ผู้ใช้คนอื่น ๆ ที่มีสิทธิ สามารถขอสำเนาแบบอ่านอย่างเดียวโดยใช้คำสั่ง get โดยไม่มีแฟล็ก -e คำสั่ง get -e บังคับให้ การป้องกันไฟล์ SCCS ที่ระบุ ด้วย ceiling, floor และรายการผู้ใช้ที่อนุญาตในไฟล์ SCCS โปรดดูที่คำสั่ง admin
-g	หมายเหตุ: หากคุณทำหลาย g-file ที่สร้างขึ้นโดยไม่ตั้งใจโดยใช้คำสั่ง get -e คุณสามารถสร้างไฟล์ขึ้นใหม่ ด้วยคำสั่ง get -k ไม่แสดงการสร้างที่แท้จริงของ g-file ใช้แฟล็ก -g เป็นหลักในการสร้าง I-file หรือเพื่อยืนยันการมีอยู่ของ SID ที่จะจะจ อยาใช้รวมกับแฟล็ก -e
-i List	ระบุรายการของเดลตาที่จะรวมในการสร้าง g-file รูปแบบรายการ SID ประกอบด้วย การรวม SID แต่ละค่าค้นด้วยเครื่องหมายจุลภาค และช่วง SID ที่ระบุโดยสอง SIDs ค้นด้วยเครื่องหมายขีดคั่น คุณสามารถระบุ SID เหมือนกันกับบรรทัดคำสั่งบรรทัดใดต่อ ไปนี้: <pre>get -e -i1.4,1.5,1.6 s.file get -e -i1.4-1.6 s.file</pre>
-k	คุณสามารถระบุ SCCS identification ของเดลตาในรูปแบบที่แสดงในคอลัมน์ SID Specified ของตาราง ก่อนหน้า คำสั่ง get แปลความหมาย SID บางส่วน ดังแสดงในคอลัมน์ SID Retrieved
-l [p]	ไม่แสดงการแทนที่ของคีย์เวิร์ด identification ใน g-file ด้วยค่าของคีย์เวิร์ด แฟล็ก -k ถูกใช้โดยแฟล็ก -e หากคุณทำลาย g-file ที่สร้างขึ้นโดยไม่ตั้งใจโดยใช้คำสั่ง get -e คุณสามารถสร้างไฟล์ขึ้นใหม่โดยการเรียกใช้ คำสั่ง get กับแฟล็ก -k แทนแฟล็ก -e
-L	เขียนขอสรุปเดลตาไปยัง I-file หากคุณระบุ -lp ข้อมูลสรุปเดลตาจะถูกเขียนไปยังเอาต์พุตมาตรฐาน และคำสั่ง get ไม่สร้าง I-file ใช้แฟล็กนี้เพื่อกำหนดว่าเดลตาใด ที่จะถูกใช้เพื่อสร้าง g-file ที่ใช้งานในขณะนี้ โปรดดูที่ไฟล์ scsfile สำหรับ รูปแบบของ I-file โปรดดูที่แฟล็ก -L เช่นกัน
-m	เขียนขอสรุปเดลตาไปยังเอาต์พุตมาตรฐาน การระบุแฟล็ก -L จะเหมือนกับการใช้แฟล็ก -lp
-n	เขียนหน้าแต่ละบรรทัดข้อความใน g-file ด้วย SID ของเดลตาแทนบรรทัดที่อยู่บนไฟล์ SCCS รูปแบบ คือ:  บรรทัดเทียบ SID ของข้อความ เขียนค่าของคีย์เวิร์ด %M% หน้าแต่ละบรรทัดของข้อความใน g-file รูปแบบของค่า %M% ตามด้วยแท็บแนวนอน ตามด้วย บรรทัดข้อความ เมื่อทั้งแฟล็ก -m และ -n ถูกใช้ รูปแบบจะเป็น:
-p	ค่าเทียบ %M% บรรทัดเทียบ SID ของข้อความ เขียนข้อความที่สร้างจากไฟล์ SCCS ไปยังเอาต์พุตมาตรฐาน และไม่สร้าง g-file เอาต์พุตข้อมูลทั้งหมดโดยปกติจะส่ง ไปยังเอาต์พุต มาตรฐานจะถูกส่งไปยังข้อผิดพลาดมาตรฐาน ยกเว้นคุณระบุแฟล็ก -s ด้วยแฟล็ก -p ในกรณีนี้ เอาต์พุตที่ปกติถูกส่งไปยังเอาต์พุต มาตรฐานจะไม่แสดง ในที่ใดๆ
-r SID	ระบุสตริง SCCS identification (SID) ของเวอร์ชันไฟล์ SCCS ที่จะสร้าง ตาราง SID determination แสดง เวอร์ชันของไฟล์ที่สร้าง และ SID ของเดลตาที่ค้างอยู่เป็นฟังก์ชัน ของ SID ที่ระบุ
-s	ไม่แสดงเอาต์พุตทั้งหมดที่โดยปกติถูกเขียนไปยังเอาต์พุตมาตรฐาน ข้อความแสดงความผิดพลาด (เขียนไปยังเอาต์พุตข้อผิดพลาด มาตรฐาน) ยังไม่ได้รับผลใด
-t	เข้างเดลตาที่เพิ่งสร้างขึ้นล่าสุดในวิธีที่กำหนด หรือสำหรับวิธีและระดับที่กำหนด
-w String	แทนที่ค่า String สำหรับ คีย์เวิร์ด %W% ใน g-files ที่ ไม่ได้ใช้สำหรับการแก้ไข
-x List	ไม่รวมรายการของเดลตาที่ระบุในการสร้าง g-file โปรดดูที่แฟล็ก -i สำหรับรูปแบบรายการ SID

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

รายละเอียดและตัวอย่างต่อไปนี้จะแสดง ความแตกต่างระหว่างไฟล์เวอร์ชันอ่านอย่างเดียวและเวอร์ชันแก้ไขได้

1. ในการพิมพ์วันที่ปัจจุบันและ SID ใน ไฟล์ ให้ใส่สัญลักษณ์ต่อไปนี้ในไฟล์:

```
%H% %I%
```

**%H%** คือสัญลักษณ์สำหรับวันที่ปัจจุบัน และ **%I%** คือสัญลักษณ์สำหรับ SID เมื่อคำสั่ง **get** เรียกข้อมูลไฟล์ เป็นแบบแก้ไขได้ จะทิ้งสัญลักษณ์ในไฟล์ และไม่ดำเนินการ แทนที่ค่าข้อความใดๆ

2. ตัวอย่างต่อไปนี้ของคำสั่ง **get** จะสร้างเวอร์ชันที่ด้วย SID สูงสุดเนื่องจากต่อไปไม่ ระบุเวอร์ชันของไฟล์:

```
$ ls
s.test.c
$ get s.test.c
3.5
59 lines
$ ls
s.test.c test.c
```

3. ในสองตัวอย่างถัดไป แฟล็ก **-r** จะระบุว่าควรใช้เวอร์ชันใด:

```
$ get -r1.3 s.test.c
1.3
67 lines
```

```
$ get -r1.3.1.4 s.test.c
1.3.1.4
50 lines
```

4. หากคุณระบุเฉพาะหมายเลขรีลีส ของ SID คำสั่ง **get** จะค้นหาไฟล์ที่มี ระดับสูงสุดภายในหมายเลขรีลีสนั้น

```
$ get -r2 s.test.c
2.7
21 lines
```

5. หาก SID ที่ระบุมีค่ามากกว่า SID ที่มีอยู่สูงสุด คำสั่ง **get** จะรับค่า SID ที่มีอยู่สูงสุด หาก SID ที่ระบุมีค่าน้อยกว่า SID ที่มีอยู่ต่ำสุด SCCS จะเขียนข้อความแสดงความผิดพลาด ในตัวอย่างต่อไปนี้ รีลีส 7 คือรีลีสที่มีอยู่สูงสุด:

```
$ get -r9 s.test.c
7.6
400 lines
```

6. แฟล็ก **-t** รับค่า เวอร์ชันสูงสุดในรีลีสหรือระดับที่กำหนด เวอร์ชันสูงสุดคือ เดลตาที่เพิ่งสร้างล่าสุด โดยไม่ขึ้นกับตำแหน่ง ในตัวอย่าง ถัดไป เดลตาที่มีอยู่สูงสุดในรีลีส 3 คือ 3.5 ขณะที่ เดลตาที่เพิ่งสร้างล่าสุดคือ 3.2.1.5

```
$ get -t -r3 s.test.c
3.2.1.5
46 lines
```

7. ตัวอย่างก่อนหน้านี้ใช้คำสั่ง **get** เพื่อรับค่าไฟล์แบบอ่านอย่างเดียว ในการสร้างสำเนาของไฟล์ที่สามารถ ถูกแก้ไขและใช้เพื่อสร้างเดลตาใหม่ ให้ใช้คำสั่ง **get** ที่มีแฟล็ก **-e** ใน **unget** เพื่อ เลิกทำผลที่เกิดกับคำสั่ง **get -e** และไม่สนใจ การเปลี่ยนแปลงใดๆ ที่ทำกับไฟล์ก่อนหน้าที่จะสร้างเดลตา ตัวอย่าง ต่อไปนี้แสดงวิธีใช้แฟล็ก **-e**:

```
$ ls
s.test.c
$ get -e s.test.c
1.3
new delta 1.4
67 lines
$ ls
p.test.c s.test.c test.c
```

ไฟล์ที่กำลังทำงานคือ test.c หากคุณแก้ไขไฟล์นี้และบันทึกการเปลี่ยนแปลงใดๆ ด้วยคำสั่ง **delta SCCS** จะสร้างเดลตาใหม่ด้วย SID เป็น 1.4 ไฟล์ p.test.c เป็นไฟล์ชั่วคราว ที่ใช้โดย SCCS เพื่อเก็บการติดตามไฟล์เวอร์ชัน

ในตัวอย่างก่อนหน้านี้นี้ คุณสามารถใช้แฟล็ก **-r** เพื่อรับค่าเวอร์ชันที่ระบุ สมมติว่ารีลีส 1 คือ รีลีสที่มีอยู่สูงสุดและเดลตา 1.3 นั้นมีอยู่แล้ว และเป็นเดลตาสูงสุดในรีลีส การใช้คำสั่ง **get** สามวิธีต่อไปนี้จะให้ผลเหมือนกัน:

```
$ get -e s.test.c
$ get -e -r1 s.test.c
$ get -e -r1.3 s.test.c
```

- ในการเริ่มหมายเลขรีลีสใหม่ (สูงกว่าในค่า) รับค่าไฟล์ด้วยแฟล็ก **-r** และระบุหมายเลขรีลีสที่มากกว่าหมายเลขรีลีสที่มีอยู่สูงสุดในตัวอย่างถัดไป รีลีส 2 ยังไม่มีอยู่:

```
$ get -e -r2 s.test.c
1.3
new delta 2.1
67 lines
```

โปรดสังเกตว่าคำสั่ง **get** ระบุ เวอร์ชันของเดลตาใหม่ที่จะถูกสร้างหากคำสั่ง **delta** เก็บการเปลี่ยนแปลงของไฟล์ SCCS

- ในการสร้างเดลตาสาขา ให้ใช้แฟล็ก **-r** และระบุรีลีสและระดับที่สาขานั้นเกิดขึ้น ในตัวอย่างถัดไป เดลตา 1.3 และ 1.4 มีอยู่แล้ว

```
$ get -e -r1.3 s.test.c
1.3
new delta 1.3.1.1
67 lines
```

สร้างเดลตาบนสาขาโดยใช้วิธีเดียวกัน

ในการแก้ไขไฟล์ ให้รับค่าไฟล์เวอร์ชันโดยใช้คำสั่ง **get -e** และบันทึกการเปลี่ยนแปลงด้วยคำสั่ง **delta** เวอร์ชันที่แก้ไขได้ต่างๆ หลายเวอร์ชัน ของไฟล์ SCCS สามารถมีอยู่ตราบใดที่แต่ละเวอร์ชันอยู่ในไดเรกทอรีที่ต่างกัน หากคุณพยายามใส่รายการซ้ำของไฟล์เวอร์ชันที่แก้ไขได้ในไดเรกทอรี (โดยใช้คำสั่ง **get**) โดยไม่ใช้คำสั่ง **delta** SCCS จะเขียนข้อความแสดงความผิดพลาด

ในการรับค่าไฟล์เวอร์ชันที่แก้ไขได้เดียวกันมากกว่าหนึ่งครั้ง ให้ตั้งค่าแฟล็กส่วนหัว **j** ในไฟล์ SCCS ด้วย คำสั่ง **admin** ตั้งค่าอ็อพชัน **j** โดยใช้แฟล็ก **-f** จากนั้นคุณสามารถรับค่า SID เดียวกันหลายๆ ครั้งจากไดเรกทอรีที่ต่างกัน การสร้างไฟล์ต่างหากสำหรับแต่ละคำสั่ง **get** แม้ว่าไฟล์ต่างๆ จะเกิดมาจาก SID เดียว SCCS จะให้กำหนดแต่ละไฟล์ด้วย SID ใหม่ค่าเฉพาะ

- ในตัวอย่างต่อไปนี้ คำสั่ง **pwd** จะแสดงไดเรกทอรีปัจจุบัน จากนั้นอ็อพชัน **j** จะถูกตั้งค่าด้วยคำสั่ง **admin**:

**หมายเหตุ:** คุณต้องมีสิทธิ์ในการเข้าถึงเพื่อเขียนในทั้งสองไดเรกทอรีเพื่อเรียกใช้ คำสั่งในตัวอย่างนี้

```
$ pwd
/home/marty/scs
$ admin -fj s.test.c
```

11. จากนั้นใช้คำสั่ง `get` เพื่อเรียกข้อมูลเวอร์ชันล่าสุดของไฟล์:

หมายเหตุ: คุณต้องมีสิทธิ์ในการเข้าถึงเพื่อเขียนในทั้งสองไดเรกทอรีเพื่อเรียกใช้ คำสั่งในตัวอย่างนี้

```
$ get -e s.test.c
1.1
new delta 1.2
5 lines
```

12. เปลี่ยนเป็นไดเรกทอรี `/home/new` และ เรียกใช้คำสั่ง `get` อีกครั้ง

หมายเหตุ: คุณต้องมีสิทธิ์ในการเข้าถึงเพื่อเขียนในทั้งสองไดเรกทอรีเพื่อเรียกใช้ คำสั่งในตัวอย่างนี้

```
$ cd /home/new
$ get -e /home/marty/sccs/s.test.c
1.2
new delta 1.1.1.1
5 lines
```

โปรดสังเกตว่า SCCS สร้างสองเดลตา 1.2 และ 1.1.1.1 จากเวอร์ชันไฟล์ต้นฉบับเดียว ของ 1.1 ดูที่ไฟล์ `p.test.c` ไฟล์ จะแสดงรายการแยกกันสำหรับแต่ละเวอร์ชันที่ใช้งานในขณะนี้ ไฟล์ `p.test.c` ยังคงอยู่ในไดเรกทอรีจนกว่าคุณ จะจัดการ ทั้งสองไฟล์เวอร์ชันด้วยคำสั่ง `delta` หรือคำสั่ง `unget`

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/get</code>	มีคำสั่ง <code>get</code>

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `admin`

คำสั่ง `unget`

คำสั่ง `what`

ภาพรวม Source Code Control System (SCCS)

---

## คำสั่ง `getconf`

### วัตถุประสงค์

เขียนค่าตัวแปรการกำหนดค่าระบบไปยังเอาต์พุตมาตรฐาน

### ไวยากรณ์

```
getconf [ -v specification ] [ SystemwideConfiguration | PathConfiguration PathName ] [ DeviceVariable DeviceName ]
```

```
getconf -a
```

## คำอธิบาย

คำสั่ง `getconf` ที่เรียกใช้ด้วย พารามิเตอร์ `SystemwideConfiguration` จะเขียน ค่าของตัวแปร ดังที่ระบุโดยพารามิเตอร์ `SystemwideConfiguration` ไปยังเอาต์พุตมาตรฐาน

คำสั่ง `getconf` ที่เรียกใช้ด้วยพารามิเตอร์ `PathConfiguration` และ `PathName` จะเขียน ค่าของตัวแปร ดังที่ระบุโดย พารามิเตอร์ `PathConfiguration` สำหรับพารที่ระบุ โดยพารามิเตอร์ `PathName` ไปยังเอาต์พุตมาตรฐาน

คำสั่ง `getconf` ที่เรียกใช้ด้วยแฟล็ก `-a` จะเขียนค่าของตัวแปรการกำหนดค่าระบบทั้งหมดไปยัง เอาต์พุตมาตรฐาน

คำสั่ง `getconf` ที่เรียกใช้ด้วยพารามิเตอร์ `DeviceVariable` และ `DeviceName` จะเขียน ค่าของชื่อหรือตำแหน่งอุปกรณ์ดิสก์ สำหรับพารอุปกรณ์ที่ระบุ โดยพารามิเตอร์ `DeviceName` ไปยังเอาต์พุตมาตรฐาน

หากตัวแปรที่ระบุถูกกำหนดบนระบบ และค่าของตัวแปรถูกอธิบายว่าพร้อมใช้จากภูมิตนน้อย `confstr` ค่าของตัวแปรที่ระบุจะถูกเขียนในรูปแบบ ต่อไปนี้:

```
"%s\n", <value>
```

มิฉะนั้น หากตัวแปรที่ระบุถูกกำหนดบน ระบบ ค่าจะถูกเขียนในรูปแบบต่อไปนี้:

```
"%d\n", <value>
```

หากตัวแปรที่ระบุถูกต้อง แต่ไม่ถูกกำหนดบน ระบบ ค่าต่อไปนี้จะถูกเขียนไปยังเอาต์พุตมาตรฐาน:

```
"undefined\n"
```

หากชื่อตัวแปรไม่ถูกต้อง หรือมีข้อผิดพลาดเกิดขึ้น ข้อความวินิจฉัยจะถูกเขียนไปยังข้อผิดพลาดมาตรฐาน

## แฟล็ก

### ไอเท็ม

`-v specification`

`-a`

### คำอธิบาย

ระบุข้อกำหนดคุณลักษณะที่เจาะจงและเวอร์ชันที่ตัวแปร การกำหนดค่าจะถูกพิจารณา หากไม่ระบุแฟล็กนี้ ค่าที่ส่งกลับจะเป็นไปตามคำตีฟอลต์การนำไปปฏิบัติ XBS5 เป็นไปตามสภาวะแวดล้อม การคอมไพล์เขียนค่าของตัวแปรการกำหนดค่าระบบทั้งหมดไปยังเอาต์พุต มาตรฐาน

## พารามิเตอร์

### ไอเท็ม

`PathName`

`SystemwideConfiguration`

`PathConfiguration`

`DeviceName`

`DeviceVariable`

### คำอธิบาย

ระบุชื่อพารสำหรับพารามิเตอร์ `PathConfiguration`

ระบุ ตัวแปรการกำหนดค่าระบบ

ระบุ ตัวแปรการกำหนดค่าพารระบบ

ระบุชื่อพารของอุปกรณ์

ระบุตัวแปรอุปกรณ์

เมื่อสัญลักษณ์ที่แสดงรายการในคอลัมน์แรกของ ตารางต่อไปนี้ถูกใช้เป็นตัวถูกดำเนินการ `system_var getconf` จะให้ค่าเดียวกับ `confstr` เมื่อเรียกใช้ด้วย ค่าในคอลัมน์ที่สอง:

หมายเหตุ: ตัวแปร `_CS_AIX_ARCHITECTURE` และ `_CS_AIX_BOOTDEV` ใช้เป็นพารามิเตอร์แก่ `confstr` จะมีอยู่สำหรับ ผู้ใช้ `root` เท่านั้น

system_var	confstr Name Value
BOOT_DEVICE	_CS_AIX_BOOTDEV
MACHINE_ARCHITECTURE	_CS_AIX_ARCHITECTURE
MODEL_CODE	_CS_AIX_MODEL_CODE
PATH	_CS_PATH
XBS5_ILP32_OFF32_CFLAGS	_CS_XBS5_ILP32_OFF32_CFLAGS
XBS5_ILP32_OFF32_LDFLAGS	_CS_XBS5_ILP32_OFF32_LDFLAGS
XBS5_ILP32_OFF32_LIBS	_CS_XBS5_ILP32_OFF32_LIBS
XBS5_ILP32_OFF32_LINTFLAGS	_CS_XBS5_ILP32_OFF32_LINTFLAGS
XBS5_ILP32_OFFBIG_CFLAGS	_CS_XBS5_ILP32_OFFBIG_CFLAGS
XBS5_ILP32_OFFBIG_LDFLAGS	_CS_XBS5_ILP32_OFFBIG_LDFLAGS
XBS5_ILP32_OFFBIG_LIBS	_CS_XBS5_ILP32_OFFBIG_LIBS
XBS5_ILP32_OFFBIG_LINTFLAGS	_CS_XBS5_ILP32_OFFBIG_LINTFLAGS
XBS5_LP64_OFF64_CFLAGS	_CS_XBS5_LP64_OFF64_CFLAGS
XBS5_LP64_OFF64_LDFLAGS	_CS_XBS5_LP64_OFF64_LDFLAGS
XBS5_LP64_OFF64_LIBS	_CS_XBS5_LP64_OFF64_LIBS
XBS5_LP64_OFF64_LINTFLAGS	_CS_XBS5_LP64_OFF64_LINTFLAGS
XBS5_LPBIG_OFFBIG_CFLAGS	_CS_XBS5_LPBIG_OFFBIG_CFLAGS
XBS5_LPBIG_OFFBIG_LDFLAGS	_CS_XBS5_LPBIG_OFFBIG_LDFLAGS
XBS5_LPBIG_OFFBIG_LIBS	_CS_XBS5_LPBIG_OFFBIG_LIBS
XBS5_LPBIG_OFFBIG_LINTFLAGS	_CS_XBS5_LPBIG_OFFBIG_LINTFLAGS

## ตัวแปรสภาวะแวดล้อม

ตัวแปรสภาวะแวดล้อมต่อไปนี้จะมีผลต่อการเรียกใช้งานของ `getconf`:

ไอเท็ม	คำอธิบาย
LANG	ให้ค่าดีฟอลต์สำหรับตัวแปรการทำให้เป็นนานาชาติที่ไม่ถูกตั้งค่าหรือ null หาก LANG ไม่ถูกตั้งค่าหรือ null ค่าที่สอดคล้องกันจาก implementation-dependent   default locale จะถูกใช้ หากตัวแปรใดๆ ของตัวแปรที่ทำให้เป็นนานาชาติมีการตั้งค่า ไม่ถูกต้อง ยูทิลิตี้จะทำหน้าที่เสมือนตัวแปรที่ถูกกำหนด หากตั้งค่าเป็นค่าสตริงที่ไม่ว่าง แทนที่ค่าของตัวแปร ที่ทำให้เป็นนานาชาติอื่นทั้งหมด
LC_CALL	กำหนดไคลแคลสำหรับการแปลความหมายของลำดับไบต์ของ ข้อมูลแบบข้อความเป็นอักขระ (ตัวอย่างเช่นอักขระไบต์เดียว จะตรงข้ามกับอักขระหลายไบต์ในอาร์กิวเมนต์)
LC_CTYPE	กำหนดไคลแคลที่ควรใช้เพื่อให้มีผลต่อรูปแบบและเนื้อหา ของข้อความวินจลีย์ที่เขียนไปยังข้อผิดพลาดมาตรฐาน
LC_MESSAGES	กำหนดตำแหน่งของแค็ตตาล็อกข้อความสำหรับการประมวลผล LC_MESSAGES
NLS_PATH	

## ตัวแปรการกำหนดค่าทั้งระบบ

พารามิเตอร์ *Systemwide Configuration* ระบุตัวแปรการกำหนดค่าทั้งระบบที่ว่าจะใช้ได้ใน ทั้งระบบ ตัวแปรการกำหนดค่าระบบมีสองประเภท:

- ตัวแปรการกำหนดค่าทั้งระบบ
- ตัวแปรการกำหนดค่ามาตรฐานระบบ

### ตัวแปรการกำหนดค่าทั้งระบบ

ตัวแปรการกำหนดค่าทั้งระบบมีค่าเริ่มต้นที่ตรงตามทุกส่วนของระบบ รายการต่อไปนี้จะกำหนด ตัวแปรการกำหนดค่าทั้งระบบที่ใช้กับคำสั่ง `getconf`:

#### ไอเท็ม

`_CS_PATH`  
`ARG_MAX`  
`BC_BASE_MAX`  
`BC_DIM_MAX`  
`BC_SCALE_MAX`  
`BC_STRING_MAX`  
`CHARCLASS_NAME_MAX`  
`CHAR_BIT`  
`CHAR_MAX`  
`CHAR_MIN`  
`CHILD_MAX`  
`CLK_TCK`  
`COLL_WEIGHTS_MAX`  
`CS_PATH`  
`EXPR_NEST_MAX`  
`INT_MAX`  
`INT_MIN`  
`LINE_MAX`  
  
`LONG_BIT`  
`LONG_MAX`  
`LONG_MIN`  
`MB_LEN_MAX`  
`NGROUPS_MAX`  
`NL_ARGMAX`  
`NL_LANGMAX`  
`NL_MSGMAX`  
`NL_NMAX`  
`NL_SETMAX`  
`NL_TEXTMAX`  
`NZERO`  
`OPEN_MAX`  
`PATH`  
`RE_DUP_MAX`  
  
`SCHAR_MAX`  
`SCHAR_MIN`  
`SHRT_MAX`  
`SHRT_MIN`  
`SSIZE_MAX`  
`STREAM_MAX`

#### คำอธิบาย

ค่าสำหรับตัวแปรสถานะแวดล้อม `PATH` ที่ใช้ค้นหาคำสั่ง  
ความยาวสูงสุด เป็นไบนารี ของอาร์กิวเมนต์สำหรับหนึ่งในรูทีนย่อย `exec` รวมถึงข้อมูลสถานะแวดล้อม  
ค่าสูงสุดที่อนุญาตสำหรับตัวแปร `obase` ที่มีคำสั่ง `bc`  
จำนวนอิลิเมนต์สูงสุดที่อนุญาตในอาร์เรย์โดยคำสั่ง `bc`  
ค่าสูงสุดที่อนุญาตสำหรับตัวแปร `scale` ที่มีคำสั่ง `bc`  
ความยาวสูงสุดของสตริงคางค์ที่มีคำสั่ง `bc` ยอมรับ  
จำนวนไบนารีสูงสุดในชื่อคลาสอักขระ  
จำนวนบิตในชนิด `character`  
ค่าสูงสุดของชนิด `character`  
ค่าต่ำสุดของชนิด `character`  
จำนวนกระบวนการที่เกิดขึ้นพร้อมกันสูงสุดสำหรับแต่ละ ID ผู้ใช้จริง  
จำนวนสัญญาณนาฬิกาต่อวินาทีที่ส่งกลับโดยรูทีนย่อย `time`  
จำนวนน้ำหนักสูงสุดที่สามารถกำหนดให้แก่รายการใน `LC_COLLATE` locale stanza ในไฟล์นิยาม locale  
ค่าของตัวแปรสถานะแวดล้อม `PATH` ที่ใช้ ค้นหาคำสั่ง  
จำนวนนิพจน์สูงสุดที่สามารถซ่อนภายในวงเล็บ โดยคำสั่ง `expr`  
ค่าสูงสุดของชนิด `int`  
ค่าต่ำสุดของชนิด `int`  
ความยาวสูงสุดเป็นไบนารี ของบรรทัดอินพุตของคำสั่ง (อินพุต มาตรฐานหรือไฟล์อื่น) เมื่ออยู่ที่ลิท็อกอธิบาย  
เป็นประมวลผลไฟล์ข้อความ ความยาวจะรวมช่องว่างสำหรับการเพิ่มอักขระบรรทัดใหม่  
จำนวนบิตในชนิด `long int`  
ค่าสูงสุดของชนิด `long int`  
ค่าต่ำสุดของชนิด `long int`  
จำนวนไบนารีสูงสุดในอักขระสำหรับ locale ใดๆ ที่สนับสนุน  
จำนวน ID กลุ่มเสริมพร้อมกันสูงสุดสำหรับแต่ละกระบวนการ  
ค่าสูงสุดของตัวเลขในการเรียกใช้ไปยังรูทีนย่อย `printf` และ `scanf`  
จำนวนไบนารีสูงสุดในชื่อ `LANG`  
หมายเลขข้อความสูงสุด  
จำนวนไบนารีสูงสุดในการแม็พการเทียบ `N-to-1`  
หมายเลขชุดสูงสุด  
จำนวนไบนารีสูงสุดในสตริงข้อความ  
ระดับความสำคัญกระบวนการดีฟอลต์  
จำนวนไฟล์สูงสุดที่หนึ่งกระบวนการสามารถเปิดได้ในหนึ่งครั้ง  
ลำดับของค่านำหน้าพาธที่ค้นด้วยโคลอนเพื่อค้นหาคำสั่ง  
จำนวนการเกิดซ้ำสูงสุดของนิพจน์ปกติที่อนุญาต เมื่อใช้พารามิเตอร์แบบช่วงระยะ เช่นพารามิเตอร์ `m` และ `n` ที่มีคำสั่ง `ed`  
ค่าสูงสุดของชนิด `signed char`  
ค่าต่ำสุดของชนิด `signed char`  
ค่าสูงสุดของชนิด `short`  
ค่าต่ำสุดของชนิด `short`  
ค่าสูงสุดของอ็อบเจ็กต์ชนิด `ssize_t`  
จำนวนสตรีมที่หนึ่งกระบวนการสามารถเปิดได้ในหนึ่งครั้ง

ไอเอ็ม	คำอธิบาย
TMP_MAX	จำนวนชื่อพาธเฉพาะต่ำสุดที่สร้างโดยรูทีนย่อย <code>tmpnam</code> จำนวนครั้งสูงสุดที่แอฟพลิเคชันสามารถเรียกใช้รูทีนย่อย <code>tmpnam</code> ที่เชื่อถือได้
TZNAME_MAX	จำนวนไบต์สูงสุดที่สนับสนุนสำหรับเขตเวลา (ไม่ใช่ ความยาวของตัวแปรสถานะแวดล้อม TZ)
UCHAR_MAX	ค่าสูงสุดของชนิด <code>unsigned char</code>
UINT_MAX	ค่าสูงสุดของชนิด <code>unsigned int</code>
ULONG_MAX	ค่าสูงสุดของชนิด <code>unsigned long int</code>
USHRT_MAX	ค่าสูงสุดของชนิด <code>unsigned short int</code>
WORD_BIT	จำนวนบิตในหนึ่ง word หรือชนิด <code>int</code>
KERNEL_BITMODE	บิตโหมดของเคอร์เนล 32 บิต หรือ 64 บิต
REAL_MEMORY	ขนาดหน่วยความจำจริง
HARDWARE_BITMODE	บิตโหมดของฮาร์ดแวร์เครื่อง 32 บิต หรือ 64 บิต
MP_CAPABLE	ความเข้ากันได้กับ MP ของเครื่อง

## ตัวแปรการกำหนดค่ามาตรฐานระบบ

ตัวแปรการกำหนดค่ามาตรฐานระบบมีค่า *minimum* ที่มาตรฐานระบบเฉพาะต้องการ คำนำหน้า `_POSIX_`, `POSIX2_` และ `_XOPEN_` บ่งชี้ว่าตัวแปรมีค่าต่ำสุดสำหรับคุณสมบัติระบบที่มาตรฐานระบบ POSIX 1003.1, POSIX 1003.2 และ X/Open ต้องการ ตามลำดับ มาตรฐานระบบเป็นค่าขั้นต่ำ ของทั้งระบบที่ระบบต้องมีเพื่อสนับสนุนมาตรฐานระบบเฉพาะ ค่า Actual Configuration อาจเกินมาตรฐานเหล่านี้ ตัวแปรการกำหนดค่า มาตรฐานระบบสำหรับคำสั่ง `getconf` ถูก กำหนดดังนี้:

ไอเอ็ม	คำอธิบาย
<code>_POSIX_ARG_MAX</code>	ความยาวสูงสุด เป็นไบต์ ของอาร์กิวเมนต์สำหรับหนึ่งในรูทีนย่อย <code>exec</code> รวมถึงข้อมูลสถานะแวดล้อม
<code>_POSIX_CHILD_MAX</code>	จำนวนกระบวนการที่เกิดขึ้นพร้อมกันสูงสุดสำหรับแต่ละ ID ผู้ใช้จริง
<code>_POSIX_JOB_CONTROL</code>	ค่า 1 หากระบบสนับสนุนการควบคุมงาน
<code>_POSIX_LINK_MAX</code>	จำนวนลิงก์สูงสุดไปยังไฟล์เดียว
<code>_POSIX_MAX_CANON</code>	จำนวนไบต์สูงสุดในอินพุตคิวที่ยอมรับของเทอร์มินัล
<code>_POSIX_MAX_INPUT</code>	จำนวนไบต์สูงสุดที่อนุญาตในเทอร์มินัลอินพุตคิว
<code>_POSIX_NAME_MAX</code>	จำนวนไบต์สูงสุดในชื่อไฟล์ (ไม่รวม <code>null</code> ที่ท้าย)
<code>_POSIX_NGROUPS_MAX</code>	จำนวน ID กลุ่มเสริมพร้อมกันสูงสุดสำหรับแต่ละกระบวนการ
<code>_POSIX_OPEN_MAX</code>	จำนวนไฟล์สูงสุดที่หนึ่งกระบวนการสามารถเปิดได้ในหนึ่งครั้ง
<code>_POSIX_PATH_MAX</code>	จำนวนไบต์สูงสุดในชื่อพาธ
<code>_POSIX_PIPE_BUF</code>	จำนวนไบต์สูงสุดที่รับประกันว่าเป็นกลุ่มเดียวกันเมื่อเขียนไปยัง ไพพ์
<code>_POSIX_SAVED_IDS</code>	ค่า 1 แต่ละกระบวนการมี <code>set-user-ID</code> ที่บันทึกและ <code>set-group-ID</code> ที่บันทึก
<code>_POSIX_SSIZE_MAX</code>	ค่าสูงสุดที่สามารถเก็บในอ็อบเจกต์ชนิด <code>ssize_t</code>
<code>_POSIX_STREAM_MAX</code>	จำนวนสตรีมที่หนึ่งกระบวนการสามารถเปิดได้ในหนึ่งครั้ง
<code>_POSIX_TIMESTAMP_RESOLUTION</code>	ความละเอียดของไฟล์ของไฟล์ทั้งหมดเป็นนาโนวินาที
<code>_POSIX_TZNAME_MAX</code>	จำนวนไบต์สูงสุดที่สนับสนุนสำหรับเขตเวลา (ไม่ใช่ ความยาวของตัวแปรสถานะแวดล้อม TZ)
<code>_POSIX_VERSION</code>	เวอร์ชันของมาตรฐาน POSIX 1 (การโยกภาษา C) ที่ระบบปฏิบัติการเป็นไปตาม
<code>_XOPEN_CRYPT</code>	ค่า 1 หากระบบสนับสนุน X/Open Encryption Feature Group
<code>_XOPEN_ENH_I18N</code>	ค่า 1 หากระบบสนับสนุน X/Open Enhanced Internationalisation Feature Group
<code>_XOPEN_SHM</code>	ค่า 1 หากระบบสนับสนุน X/Open Shared Memory Feature Group
<code>_XOPEN_VERSION</code>	เวอร์ชันของ X/Open Portability Guide ที่ระบบปฏิบัติการ เป็นไปตาม
<code>_XOPEN_XCU_VERSION</code>	เวอร์ชันของ X/Open Commands and Utilities specification ที่ระบบปฏิบัติการเป็นไปตาม
<code>_XOPEN_XPG2</code>	ค่า 1 หากระบบสนับสนุน X/Open Portability Guide, Volume 2, January 1987, XVS System Calls and Libraries มิฉะนั้นจะไม่ถูกกำหนด
<code>_XOPEN_XPG3</code>	ค่า 1 หากระบบสนับสนุน X/Open Specification, February 1992, System Interfaces and Headers, Issue 3 มิฉะนั้นจะไม่ถูกกำหนด
<code>_XOPEN_XPG4</code>	ค่า 1 หากระบบสนับสนุน X/Open CAE Specification, July 1992, System Interfaces and Headers, Issue 4 มิฉะนั้นจะไม่ถูกกำหนด
<code>POSIX2_BC_BASE_MAX</code>	ค่าสูงสุดที่อนุญาตสำหรับตัวแปร <code>obase</code> ที่มีคำสั่ง <code>bc</code>
<code>POSIX2_BC_DIM_MAX</code>	จำนวนอิลิเมนต์สูงสุดที่อนุญาตในอาร์เรย์โดยคำสั่ง <code>bc</code>
<code>POSIX2_BC_SCALE_MAX</code>	ค่าสูงสุดที่อนุญาตสำหรับตัวแปร <code>scale</code> ที่มีคำสั่ง <code>bc</code>
<code>POSIX2_BC_STRING_MAX</code>	ความยาวสูงสุดของสตริงค่าคงที่ที่มีคำสั่ง <code>bc</code> ยอมรับ
<code>POSIX2_CHAR_TERM</code>	ค่า 1 หากระบบสนับสนุนชนิดเทอร์มินัลอย่างน้อยหนึ่งชนิด มิฉะนั้น จะมีค่า -1

ไอเอ็ม	คำอธิบาย
POSIX2_COLL_WEIGHTS_MAX	จำนวนน้ำหนักสูงสุดที่สามารถกำหนดให้แก่วิธีการของตัวแปรโลแคล LC_COLLATE ในไฟล์นิยามโลแคล
POSIX2_C_BIND	ค่า 1 หากระบบสนับสนุน C Language Binding Option จาก POSIX 2 มิฉะนั้น จะมีค่า -1
POSIX2_C_DEV	ค่า 1 หากระบบสนับสนุน C Language Development Utilities จาก POSIX 2 มิฉะนั้น จะมีค่า -1
POSIX2_C_VERSION	เวอร์ชันของมาตรฐาน POSIX 2 (การโยกย้ายภาษา C) ที่ระบบปฏิบัติการเป็นไปตาม
POSIX2_EXPR_NEST_MAX	จำนวนนิพจน์สูงสุดที่สามารถซ้อนภายในวงเล็บ โดยคำสั่ง expr
POSIX2_FORT_DEV	ค่า 1 หากระบบสนับสนุน FORTRAN Development Utilities Options จาก POSIX 2 มิฉะนั้น จะมีค่า -1
POSIX2_FORT_RUN	ค่า 1 หากระบบสนับสนุน FORTRAN Runtime Utilities Option จาก POSIX 2 มิฉะนั้น จะมีค่า -1
POSIX2_LINE_MAX	ความยาวสูงสุดเป็นไบต์ ของบรรทัดอินพุตของคำสั่ง (อินพุต มาตรฐานหรือไฟล์อื่น) เมื่อคำสั่งถูกอธิบายเป็นประมวลผลไฟล์ข้อความ ความยาวจะรวมช่องว่างสำหรับการเพิ่มอักขระบรรทัดใหม่
POSIX2_LOCALEDEF	ค่า 1 หากระบบสนับสนุนการสร้างโลแคลโดย คำสั่ง localedef มิฉะนั้น จะไม่ถูกกำหนด
POSIX2_RE_DUP_MAX	จำนวนการเกิดซ้ำสูงสุดของนิพจน์ปกติที่อนุญาต เมื่อใช้พารามิเตอร์แบบช่วงระยะ เช่น พารามิเตอร์ <i>m</i> และ <i>n</i> ที่มีคำสั่ง ed
POSIX2_SW_DEV	ค่า 1 หากระบบสนับสนุน Software Development Utilities Option มิฉะนั้น จะมีค่า -1
POSIX2_UPE	ค่า 1 หากระบบสนับสนุน User Portability Utilities Option จาก POSIX 2 มิฉะนั้น จะมีค่า -1
POSIX2_VERSION	วันที่อนุมัติของเวอร์ชันล่าสุดของมาตรฐาน POSIX 2 ที่ระบบสนับสนุน วันที่เป็นตัวเลขทศนิยมหลักที่มีสี่หลักแรก ระบุปี และสองหลักสุดท้ายระบุเดือน เวอร์ชันที่แตกต่างกัน ของมาตรฐาน POSIX 2 ถูกอนุมัติเป็นระยะโดย IEEE Standards Board และวันที่อนุมัติจะถูกใช้เพื่อแยกความแตกต่างระหว่างเวอร์ชัน

## ตัวแปรการกำหนดค่าพาธระบบ

พารามิเตอร์ *Path Configuration* ระบุตัวแปรการกำหนดค่าพาธระบบที่ซึ่งค่าของตัวแปรจะมีข้อมูล เกี่ยวกับพาธและโครงสร้างพาธในระบบ รายการต่อไปนี้กำหนด ตัวแปรเหล่านี้:

ไอเอ็ม	คำอธิบาย
_POSIX_CHOWN_RESTRICTED	รูทีนย่อย <b>chown()</b> ถูกจำกัดเฉพาะกระบวนการ ที่มีสิทธิเหมาะสม และเพื่อทำการเปลี่ยนแปลง ID กลุ่มของไฟล์เฉพาะกับ ID กลุ่มที่ใช้งานอยู่ หรือกับหนึ่งใน ID กลุ่ม เพิ่มเติม หากพารามิเตอร์ <i>PathName</i> อ้างถึงไดเรกทอรี ค่าที่ส่งกลับจะเป็นไฟล์ใดๆ ยกเว้นไดเรกทอรีที่มีอยู่ หรือสามารถถูกสร้างภายในไดเรกทอรี
_POSIX_NO_TRUNC	ชื่อพาธยาวเกินขีดจำกัดที่ระบุโดยตัวแปร <i>NAME_MAX</i> จะทำให้เกิดข้อผิดพลาด หากพารามิเตอร์ <i>PathName</i> อ้างถึงไดเรกทอรี ค่าที่ส่งกลับจะเป็นชื่อไฟล์ภายในไดเรกทอรี
_POSIX_VDISABLE	อักขระพิเศษเทอร์มินัล ที่กำหนดในไฟล์ <i>termios.h</i> สามารถถูกปิดใช้งานโดยใช้ค่า อักขระนี้
LINK_MAX	จำนวนลิงก์สูงสุดไปยังไฟล์เดียว หากพารามิเตอร์ <i>PathName</i> อ้างถึงไดเรกทอรี ค่าที่ส่งกลับจะเป็นไดเรกทอรี
MAX_CANON	จำนวนไบต์สูงสุดในบรรทัดอินพุตที่ยอมรับของเทอร์มินัล
MAX_INPUT	จำนวนไบต์สูงสุดที่ซึ่งช่องว่างมีอยู่ในเทอร์มินัล อินพุตคิว
NAME_MAX	จำนวนไบต์สูงสุดในชื่อไฟล์ (ไม่รวม null ที่ท้าย) หากพารามิเตอร์ <i>PathName</i> อ้างถึงไดเรกทอรี ค่าที่ส่งกลับจะเป็นชื่อไฟล์ภายในไดเรกทอรี
PATH_MAX	จำนวนไบต์สูงสุดในชื่อพาธ รวมอักขระ null ที่ท้าย หากพารามิเตอร์ <i>PathName</i> อ้างถึง ไดเรกทอรี ค่าที่ส่งกลับจะเป็นความยาวสูงสุดของชื่อพาธสัมพัทธ์ เมื่อไดเรกทอรีที่ระบุเป็นไดเรกทอรีการทำงาน
PIPE_BUF	จำนวนไบต์สูงสุดที่รับประกันว่าเป็นกลุ่มเดียวกันเมื่อเขียนไปยัง ไพพ์ ค่าพารามิเตอร์ <i>PathName</i> อ้างถึง FIFO หรือ ไพพ์ ค่าที่ส่งกลับจะเป็นอ็อบเจกต์ที่ถูกอ้างอิง หากพารามิเตอร์ <i>PathName</i> อ้างถึงไดเรกทอรี ค่าที่ส่งกลับจะเป็น FIFO ใดๆ ที่มีอยู่ หรือสามารถถูกสร้างภายในไดเรกทอรี
DISK_PARTITION	ขนาดฟิสิกส์พาร์ติชันของดิสก์ หมายเหตุ: สำหรับตัวแปรการกำหนดค่าพาธ <i>DISK_PARTITION</i> พารามิเตอร์ <i>PathName</i> ต้อง ระบุพาธสมบูรณ์ของดิสก์ซึ่งมีข้อมูลที่จะถูก เคียวรี

ไอเท็ม  
DISK\_SIZE

คำอธิบาย  
ขนาดดิสก์เป็นเมกะไบต์  
หมายเหตุ: สำหรับตัวแปรที่กำหนดค่าพารามิเตอร์ `DISK_SIZE` พารามิเตอร์ `PathName` ต้องระบุพารามิเตอร์ของดิสก์ที่ข้อมูลจะถูกเก็บไว้

## ตัวแปรอุปกรณ์

พารามิเตอร์ `Device Variable` ระบุว่าพารามิเตอร์ `DeviceName` คือพารามิเตอร์ของอุปกรณ์ เช่น `/dev/hdisk0` พารามิเตอร์ที่กำหนดของดิสก์คำสั่ง `getconf` จะแสดงชื่ออุปกรณ์หรือตำแหน่งของดิสก์

ไอเท็ม	คำอธิบาย
<code>DISK_DEVNAME</code>	ชื่ออุปกรณ์หรือตำแหน่งของอุปกรณ์

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย  
0 ตัวแปรที่ระบุใช้ได้ และข้อมูลเกี่ยวกับสถานะ ปัจจุบันถูกเขียนสำเร็จ  
>0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการแสดงค่าของตัวแปร `ARG_MAX` ให้ป้อน คำสั่งต่อไปนี้:

```
getconf ARG_MAX
```

2. เมื่อต้องการแสดงค่าของตัวแปร `NAME_MAX` สำหรับไดเรกทอรี `/usr` ป้อนคำสั่งต่อไปนี้:

```
getconf NAME_MAX /usr
```

3. ลำดับของคำสั่งเซลล์ต่อไปนี้จะแสดงวิธีการจัดการผลลัพธ์ที่ไม่ระบุ:

```
if value=$(getconf PATH_MAX /usr)
    ดังนั้น
        if [ "$value" = "undefined" ]
            ดังนั้น
                echo
                    The value of PATH_MAX in /usr is undefined.
            else
                echo
                    The value of PATH_MAX in /usr is $value.
            fi
        else
            echo Error in the getconf command.
        fi
```

4. หากคำสั่ง:

```
getconf _XBS5_ILP32_OFF32
```

ไม่เขียน `-1` หรือ `undefined` ไปยังเอาต์พุตมาตรฐาน ดังนั้นคำสั่งที่มีรูปแบบ:

```
getconf -v XBS5_ILP32_OFF32 ...
```

พิจารณาค่าสำหรับตัวแปรการกำหนดค่าที่สอดคล้องกับสถานะแวดล้อมการคอมไพล์ XBS5\_ILP32\_OFF32 ที่ระบุใน c89, Extended คำอธิบาย

5. หากคำสั่ง:

```
getconf _XBS5_ILP32_OFFBIG
```

ไม่เขียน -1\n หรือ undefined\n ไปยังเอาต์พุตมาตรฐาน ดังนั้นคำสั่งที่มีรูปแบบ:

```
getconf -v XBS5_ILP32_OFFBIG ...
```

พิจารณาค่าสำหรับตัวแปรการกำหนดค่าที่สอดคล้องกับสถานะแวดล้อมการคอมไพล์ XBS5\_ILP32\_OFFBIG ที่ระบุใน c89, Extended คำอธิบาย

6. หากคำสั่ง:

```
getconf _XBS5_LP64_OFF64
```

ไม่เขียน -1\n หรือ undefined\n ไปยังเอาต์พุตมาตรฐาน ดังนั้นคำสั่งที่มีรูปแบบ:

```
getconf -v XBS5_LP64_OFF64 ...
```

พิจารณาค่าสำหรับตัวแปรการกำหนดค่าที่สอดคล้องกับสถานะแวดล้อมการคอมไพล์ XBS5\_LP64\_OFF64 ที่ระบุใน c89, Extended คำอธิบาย

7. หากคำสั่ง:

```
getconf _XBS5_LPBIG_OFFBIG
```

ไม่เขียน -1\n หรือ undefined\n ไปยังเอาต์พุตมาตรฐาน ดังนั้นคำสั่งที่มีรูปแบบ:

```
getconf -v _XBS5_LPBIG_OFFBIG
```

พิจารณาค่าสำหรับตัวแปรการกำหนดค่าที่สอดคล้องกับสถานะแวดล้อมการคอมไพล์ XBS5\_LPBIG\_OFFBIG ที่ระบุใน c89, Extended คำอธิบาย

8. เมื่อต้องการกำหนดขนาดดิสก์สำหรับดิสก์ hdisk0 ในฐานะผู้ใช้ root ป้อนคำสั่งต่อไปนี้:

```
getconf DISK_SIZE /dev/hdisk0
```

9. เมื่อต้องการกำหนดขนาดหน่วยความจำจริง ให้ป้อนคำสั่งต่อไปนี้:

```
getconf REAL_MEMORY
```

10. เมื่อต้องการกำหนดว่าฮาร์ดแวร์เครื่องเป็น 32 บิตหรือ 64 บิต ให้ป้อน คำสั่งต่อไปนี้:

```
getconf HARDWARE_BITMODE
```

11. เมื่อต้องการกำหนดว่าเคอร์เนลเป็น 32 บิตหรือ 64 บิต ป้อนคำสั่ง ต่อไปนี้:

```
getconf KERNEL_BITMODE
```

12. เมื่อต้องการกำหนดชื่ออุปกรณ์ หรือตำแหน่งของดิสก์ hdisk0 ป้อนคำสั่งต่อไปนี้:

```
getconf DISK_DEVNAME hdisk0
```

## ไฟล์

ไอเท็ม  
/usr/bin/getconf  
/usr/include/limits.h  
/usr/include/unistd.h

คำอธิบาย  
มีคำสั่ง `getconf`  
กำหนดตัวแปรการกำหนดค่าระบบ  
กำหนดตัวแปรการกำหนดค่าระบบ

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `confstr`  
คำสั่ง `pathconf`  
คำสั่ง `sysconf`  
คำสั่ง `Commands`

---

## คำสั่ง `getdev`

### วัตถุประสงค์

แสดงรายการอุปกรณ์ที่ตรงกับเกณฑ์ที่ระบุ

### ไวยากรณ์

```
getdev [ -a ] [ -e ] [ Criteria ] [ DeviceList ]
```

### คำอธิบาย

แสดงรายการอุปกรณ์ที่ตรงกับเกณฑ์ที่กำหนด เกณฑ์ที่กำหนดใน รูปของนิพจน์ คำสั่ง `getdev` สามารถตรวจสอบอุปกรณ์ทั้งหมดบนระบบ หรือรายการของอุปกรณ์ที่ระบุ

### แฟล็ก

ไอเท็ม	คำอธิบาย
-a	ระบุอุปกรณ์ต้องตรงกับเกณฑ์ทั้งหมด จึงจะรวมในรายการที่สร้างโดยคำสั่งนี้ แฟล็ก -a ไม่มีผลใดหากไม่กำหนดเกณฑ์
-e	ระบุอุปกรณ์ที่จัดใหม่ใน <code>devicelist</code> จะไม่รวมในรายการที่สร้างโดยคำสั่ง <code>getdev</code> หากไม่มีแฟล็ก -e เฉพาะอุปกรณ์ใน <code>devicelist</code> เท่านั้นที่ถูกสร้าง แฟล็กนี้จะถูกข้าม หากไม่ระบุอุปกรณ์

# พารามิเตอร์

## ไอเท็ม Criteria

### คำอธิบาย

กำหนดเกณฑ์ที่อุปกรณ์ต้องตรงก่อนจึงจะถูก รวมในรายการที่สร้าง Criteria สามารถ ระบุเป็นนิพจน์หรือรายการของนิพจน์ซึ่งอุปกรณ์ต้องเป็นไปตามเงื่อนไขที่จะถูกรวมในรายการที่สร้างโดย getdev หากไม่ระบุเกณฑ์ อุปกรณ์ทั้งหมดจะถูกรวมในรายการ

อุปกรณ์ต้องเป็นไปตามอย่างน้อยหนึ่งเกณฑ์ในรายการ อย่างไรก็ตาม อ็อปชัน -a สามารถใช้เพื่อระบุว่าการดำเนินการ "ค่าและเชิงตรรกะ" ควรถูกดำเนินการ ดังนั้น เฉพาะอุปกรณ์ที่ตรงกับเกณฑ์ทั้งหมดในรายการที่จะถูกรวม

โดยมีขนิตนิพจน์ที่เป็นไปได้สี่ชนิดซึ่งเกณฑ์ที่ระบุ ในพารามิเตอร์ Criteria จะเป็นตามนั้น:

Attribute=Value

ดึงอุปกรณ์ทั้งหมดที่มีสมาชิกที่มี Attribute ที่กำหนดและเท่ากับ Value

Attribute!=Value

ดึงอุปกรณ์ทั้งหมดที่มีสมาชิกที่มี Attribute ที่กำหนดและไม่เท่ากับ Value

Attribute:\*

ดึงอุปกรณ์ทั้งหมดที่มีสมาชิกที่มี Attribute ที่กำหนด

Attribute!:\*

ดึงอุปกรณ์ทั้งหมดที่มีสมาชิกที่ไม่มี Attribute ที่กำหนด

ต่อไปนี้เป็นชื่ออุปกรณ์อุปกรณ์ที่ใช้ได้:

alias ชื่อที่อุปกรณ์เป็นที่รู้จัก

desc รายละเอียดของอุปกรณ์

type โทเค็นที่อธิบายชนิดของอุปกรณ์ ชุดของค่าที่ใช้ได้สำหรับแอตทริบิวต์ type สามารถหาได้โดยการเรียกใช้งานคำสั่งต่อไปนี้ `odmget PdDv | grep -w class | awk '{print $3}' | sed 's//g' | sort | uniq`

status สภาวะปัจจุบันของอุปกรณ์ รายการค่าที่เป็นไปได้สำหรับ สถานะได้แก่: 1. Defined 2. Available 3. Stopped 4.

Diagnose ค่าสำหรับ status ไม่คำนึงถึงขนาดตัวพิมพ์

ระบุรายการที่ค้นด้วยช่องว่างของอุปกรณ์ที่จะถูกตรวจสอบ สำหรับ Criteria

DeviceList

## สถานะออก

0 คำสั่งเสร็จสิ้นสมบูรณ์

> 1 มีความล้มเหลวเกิดขึ้น

## ตัวอย่าง

1. ในการแสดงอุปกรณ์ทั้งหมด ให้ป้อน:

```
getdev
```

2. ในการแสดงรายการอุปกรณ์ที่มีชนิดเป็น "logical\_volume" ให้ป้อน:

```
getdev type=logical_volume
```

3. ในการแสดงรายการที่ไม่มีชนิดเป็น "logical\_volume" ให้ป้อน:

```
getdev type!=logical_volume
```

4. ในการแสดงรายการอุปกรณ์ที่มีชนิดเป็น "logical\_volume" หรือ alias อุปกรณ์ เป็น "sys0" ให้ป้อน:

```
getdev type=logical_volume alias=sys0
```

เอาต์พุตจะคล้ายกับต่อไปนี้:

```
hd1
hd2
hd3
hd4
...
sys0
```

5. ในการแสดงรายการอุปกรณ์ที่มีชนิดเป็น "logical\_volume" และ alias อุปกรณ์ เป็น "lv01" ให้ป้อน:

```
getdev -a type=logical_volume alias=lv01
```

6. ในการแสดงอุปกรณ์ที่มีแอตทริบิวต์ status ถูกกำหนดให้ป้อน:

```
getdev status:*
```

7. ในการแสดงอุปกรณ์ที่มีแอตทริบิวต์ desc ไม่ถูกกำหนดให้ป้อน:

```
getdev desc!:*
```

## ไฟล์

ไอเท็ม

/usr/sbin/getdev

คำอธิบาย

มีคำสั่ง getdev

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง getdgrp”

---

## คำสั่ง getdgrp

### วัตถุประสงค์

แสดงรายการคลาสอุปกรณ์ที่ตรงกับเกณฑ์ที่ระบุ

### ไวยากรณ์

```
getdgrp [-a] [-e] [-l] [Criteria] [DeviceClassList]
```

### คำอธิบาย

แสดงรายการคลาสอุปกรณ์ที่มีอุปกรณ์ตรงกับเกณฑ์ที่กำหนด เกณฑ์ถูกกำหนดในรูปของนิพจน์

### แฟล็ก

ไอเท็ม

-a

คำอธิบาย

ระบุว่าอุปกรณ์ต้องตรงกับเกณฑ์ทั้งหมดของ คลาสอุปกรณ์จึงจะถูกรวมในรายงานที่สร้างโดยคำสั่งนี้ แฟล็ก -a ไม่มีผลใดหากไม่กำหนดเกณฑ์

-e

ระบุว่าคลาสอุปกรณ์ที่ระบุใน รายการพารามิเตอร์จะไม่รวมในรายงานที่สร้างโดยคำสั่งนี้ แฟล็ก -e ไม่มีผลใดหากไม่ระบุอุปกรณ์

-l

ระบุว่าคลาสอุปกรณ์ทั้งหมดที่เป็นไปตามอ็อปชัน -e และรายการ dgroup จะถูกแสดงรายการแม้ว่า ไม่มีสมาชิกอุปกรณ์ที่ถูกต้อง อ็อปชันนี้ไม่มีผลหากระบุ Criteria บนบรรทัดคำสั่ง

### พารามิเตอร์

## ไอเท็ม Criteria

### คำอธิบาย

กำหนดเกณฑ์ที่อุปกรณ์ต้องมีค่าตรงก่อนที่คลาสอุปกรณ์ ที่อุปกรณ์เป็นสมาชิกอยู่จะสามารถรวมในรายการที่สร้างขึ้นได้ Criteria สามารถระบุเป็นนิพจน์หรือรายการของนิพจน์ซึ่ง อุปกรณ์ต้องเป็นไปตามเงื่อนไขของคลาสที่จะถูกรวมในรายการที่สร้างโดย `getdgrp` หากไม่กำหนดเกณฑ์ คลาสอุปกรณ์ทั้งหมด จะถูกรวมในรายการ

อุปกรณ์ต้องเป็นไปตามอย่างน้อยหนึ่งเกณฑ์ในรายการ อย่างไรก็ตาม อ็อปชัน `-a` สามารถใช้เพื่อระบุว่าดำเนินการ "ค่าและเชิงตรรกะ" ควรถูกดำเนินการ ดังนั้นเฉพาะคลาส ที่มีอุปกรณ์ที่ตรงกับเกณฑ์ทั้งหมดในรายการเท่านั้นที่จะถูกรวม

โดยมีชนิดนิพจน์ที่เป็นไปได้สี่ชนิดซึ่งเกณฑ์ที่ระบุในพารามิเตอร์ *Criteria* จะเป็นตามนั้น:

*Attribute=Value*

ดึงคลาสอุปกรณ์ทั้งหมดที่มีสมาชิกที่มี *Attribute* ที่กำหนดและเท่ากับ *Value*

*Attribute!=Value*

ดึงคลาสอุปกรณ์ทั้งหมดที่มีสมาชิกที่มี *Attribute* ที่กำหนดและไม่เท่ากับ *Value*

*Attribute:\**

ดึงคลาสอุปกรณ์ทั้งหมดที่มีสมาชิกที่มี *Attribute* ที่กำหนด

*Attribute!:\**

ดึงคลาสอุปกรณ์ทั้งหมดที่มีสมาชิกที่ไม่มี *Attribute* ที่กำหนด

ต่อไปนี้เป็นชื่ออุปกรณ์อุปกรณ์ที่ใช้ได้:

**alias** ชื่อที่อุปกรณ์เป็นที่รู้จัก

**desc** รายละเอียดของอุปกรณ์

**type** โทเค็นที่อธิบายชนิดของอุปกรณ์

**status** สภาวะปัจจุบันของอุปกรณ์ รายการค่าที่เป็นไปได้สำหรับ สถานะได้แก่: 1. Defined 2. Available 3. Stopped 4. Diagnose  
ค่าสำหรับ **status** ไม่คำนึงถึงขนาดตัวพิมพ์

*DeviceClassList* ระบุชื่อคลาสอุปกรณ์ในฐานะข้อมูล Customized Device Configuration หรือในฐานะข้อมูล Predefined Device Configuration

## สถานะออก

0 คำสั่งเสร็จสิ้นสมบูรณ์

1 ไวยากรณ์คำสั่งไม่ถูกต้อง ใช้อ็อปชันไม่ถูกต้อง หรือเกิดข้อผิดพลาด ภายใน

2 อ็อบเจ็กต์คลาส Customized Devices หรืออ็อบเจ็กต์คลาส Predefined Devices ไม่สามารถเปิดเพื่ออ่าน

## ตัวอย่าง

1. ในการแสดงคลาสอุปกรณ์ทั้งหมดให้ป้อน:

```
getdgrp
```

เอาต์พุตคล้ายกับตัวอย่างต่อไปนี้:

```
adapter  
aio  
bus  
cdrom  
disk  
diskette  
gxme  
if
```

```
keyboard
lft
logical_volume
lvm
memory
mouse
planar
processor
pty
pwrmtg
rcm
sys
tape
tcpip
tty
```

2. ในการแสดงรายการคลาสอุปกรณ์ที่อุปกรณ์มีชนิดเป็น "logical\_volume" ให้ป้อน:

```
getdgrp type=logical_volume
```

เอาต์พุตคล้ายกับตัวอย่างต่อไปนี้:

```
logical_volume
```

3. ในการแสดงรายการคลาสอุปกรณ์ที่อุปกรณ์มีชนิดเป็น "logical\_volume" หรือ alias อุปกรณ์เป็น "sys0" ให้ป้อน:

```
getdgrp type=logical_volume alias=sys0
```

เอาต์พุตคล้ายกับตัวอย่างต่อไปนี้:

```
logical_volume
sys
```

4. ในการแสดงรายการคลาสอุปกรณ์ที่มีแอตทริบิวต์สถานะคลาสอุปกรณ์ถูกกำหนดให้ป้อน:

```
getdgrp status=defined
```

เอาต์พุตคล้ายกับตัวอย่างต่อไปนี้:

```
logical_volume
posix_aio
rcm
```

5. ในการแสดงคลาสอุปกรณ์สำหรับอุปกรณ์ที่มีแอตทริบิวต์ status ถูกกำหนดและอยู่ในคลาสอุปกรณ์ "processor" ให้ป้อน:

```
getdgrp status:* processor
```

เอาต์พุตคล้ายกับตัวอย่างต่อไปนี้:

```
processor
```

6. ในการแสดงคลาสอุปกรณ์สำหรับอุปกรณ์ที่มีแอตทริบิวต์ status ไม่ถูกกำหนดให้ป้อน:

```
getdgrp status!:* processor
```

## ไฟล์

ไอเท็ม  
/usr/sbin/getdgrp

คำอธิบาย  
มีคำสั่ง getdgrp

สิ่งอ้างอิงที่เกี่ยวข้อง:  
“คำสั่ง getdev” ในหน้า 709

---

## คำสั่ง getea

### วัตถุประสงค์

เรียกข้อมูลแอดทริบิวต์ส่วนเพิ่มที่ระบุชื่อจากไฟล์

### ไวยากรณ์

```
getea [-n Name] [-l] [-e RegExp] [-s] FileName
```

### คำอธิบาย

คำสั่ง `getea` จะอ่านแอดทริบิวต์ส่วนเพิ่มที่ระบุชื่อจากไฟล์ หากระบุพารามิเตอร์ `-n Name` ดังนั้นเฉพาะแอดทริบิวต์ส่วนเพิ่มที่ตรงกับ `Name` เท่านั้นที่ถูกเรียกข้อมูล

**หมายเหตุ:** ในการป้องกันการใช้ชื่อซ้ำกัน JFS2 ได้สงวนคำนำหน้า 8-อักขระ (0xf8) SYSTEM(0xf8) สำหรับแอดทริบิวต์ส่วนขยาย-ที่กำหนดโดยระบบ หลีกเลี่ยงการใช้คำนำหน้านี้สำหรับการกำหนดชื่อแอดทริบิวต์ส่วนขยาย-ที่กำหนดโดยผู้ใช้

หากระบุพารามิเตอร์ `-e RegExp` ดังนั้นเฉพาะแอดทริบิวต์ส่วนเพิ่มที่ตรงกับ `RegExp` เท่านั้นที่ถูกเรียกข้อมูล หากไม่ระบุแฟล็ก `-n` หรือ `-e` แอดทริบิวต์ส่วนเพิ่มทั้งหมดจะถูกเรียกข้อมูล

คำสั่งนี้ไม่ใช่เพื่อรับค่า ACLs คำสั่ง `aclget` จะถูกใช้เพื่อรับค่า ACLs

### แฟล็ก

ไอเท็ม  
`-e RegExp`  
`-l`  
`-n Name`  
`-s`  
`FileName`

คำอธิบาย  
ระบุนิพจน์ทั่วไปเพื่อเรียกข้อมูลแอดทริบิวต์ส่วนเพิ่ม ทั้งหมดที่ตรงกัน ค่าถูกแสดงในรูปแบบอักขระระบุเพื่อรับค่าแอดทริบิวต์ส่วนเพิ่มจากลิงก์ สัญลักษณ์แทนไฟล์ที่ชี้ไป  
ระบุชื่อของแอดทริบิวต์ส่วนเพิ่มที่เจาะจงเพื่อเรียกข้อมูล ค่าถูกแสดงในรูปแบบอักขระแสดงเฉพาะชื่อและไม่แสดงค่าสำหรับแอดทริบิวต์ส่วนเพิ่ม  
ระบุไฟล์ที่จะอ่านแอดทริบิวต์ส่วนเพิ่ม

### สถานะออก

ไอเท็ม  
0  
เลขจำนวนเต็มบวก

คำอธิบาย  
ดำเนินการเป็นผลสำเร็จ  
มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

1. ในการเรียกข้อมูลแอ็ตทริบิวต์ส่วนเพิ่มที่ระบุชื่อทั้งหมดสำหรับไฟล์ design.html พิมพ์:  

```
getea design.html
```
2. ในการเรียกข้อมูลแอ็ตทริบิวต์ส่วนเพิ่มที่ระบุชื่อ Approver สำหรับไฟล์ design.html พิมพ์:  

```
getea -n Approver design.html
```
3. ในการเรียกข้อมูลเฉพาะชื่อของแอ็ตทริบิวต์ส่วนเพิ่มที่ระบุชื่อทั้งหมดสำหรับไฟล์ design.html พิมพ์:  

```
getea -s design.html
```
4. ในการเรียกข้อมูลแอ็ตทริบิวต์ส่วนเพิ่มที่ระบุชื่อทั้งหมดสำหรับลิงก์สัญลักษณ์ design.html พิมพ์:  

```
getea -l design.html
```

## Location

/usr/sbin

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง chfs

คำสั่ง crfs

คำสั่ง setea

---

## คำสั่ง getopt

### วัตถุประสงค์

วิเคราะห์คำแฟล็กและพารามิเตอร์บรรทัดคำสั่ง

### ไวยากรณ์

**getopt** *Format Tokens*

## คำอธิบาย

คำสั่ง `getopt` จะวิเคราะห์คำรายการ ของโทเค็นโดยใช้รูปแบบที่ระบุแฟล็กและอาร์กิวเมนต์ที่ควรมีแฟล็ก คือตัวอักษร ASCII ตัวเดียว และเมื่อตามด้วย : (โคลอน) จะถูกคาดหวังต้องมีอาร์กิวเมนต์ที่อาจถูกค้นหรือไม่ถูกค้นโดยแท็บหรือช่องว่างอย่างน้อยหนึ่งค่า คุณสามารถมีอาร์กิวเมนต์หลายไบต์ในอาร์กิวเมนต์ แต่ต้องไม่เป็นตัวอักษรแฟล็ก

คำสั่ง `getopt` ดำเนินการประมวลผล เสร็จสมบูรณ์เมื่ออ่านโทเค็นทั้งหมด หรือเมื่อพบโทเค็นพิเศษ — (เครื่องหมายขีดค้นคู่) จากนั้นคำสั่ง `getopt` จะเอาต์พุต แฟล็กที่ประมวลผล — (เครื่องหมายขีดค้นคู่) และโทเค็นคงเหลือใดๆ

หากโทเค็นล้มเหลวในการจับคู่กับแฟล็ก คำสั่ง `getopt` เขียนข้อความไปยังข้อผิดพลาดมาตรฐาน

## ตัวอย่าง

คำสั่ง `getopt` สามารถใช้ในเชลล์สคริปต์โครงสร้างเพื่อวิเคราะห์อ็อปชัน ดังในตัวอย่างต่อไปนี้:

```
#!/usr/bin/bsh
# parse command line into arguments
set -- `getopt a:bc $*`
# check result of parsing
if [ $? != 0 ]
    ดั้งนั้น
        exit 1
fi
while [ $1 != -- ]
do
    case $1 in
        -a) # set up the -a flag
            AFLG=1
            AARG=$2
            shift;;
        -b) # set up the -b flag
            BFLG=1;;
        -c) # set up the -c flag
            CFLG=1;;
        esac
        shift # next flag
    done
    shift # skip --
# now do the work
.
.
.
```

**หมายเหตุ:** ใน C เชลล์ ใช้คำสั่งต่อไปนี้เพื่อรันคำสั่ง `getopt`:

```
set argv=`getopt OptionString $*`
```

ในแต่ละตัวอย่างต่อไปนี้ คำสั่ง `getopt` จะประมวลผลประมวลผลและอาร์กิวเมนต์ด้วยวิธีเดียวกัน:

- -a ARG -b -c
- -a ARG -bc
- -aARG -b -c

- -b -c -a ARG

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/getopt	มีคำสั่ง <code>getopt</code>

### ข้อมูลที่เกี่ยวข้อง:

- คำสั่ง `bsh`
- คำสั่ง `csh`
- คำสั่ง `getopt`
- คำสั่ง `Shells`

## คำสั่ง `getopts`

### วัตถุประสงค์

ประมวลผลอาร์กิวเมนต์บรรทัดคำสั่งและตรวจสอบความถูกต้อง ของอ็อปชัน

### ไวยากรณ์

`getopts OptionString Name [ Argument ...]`

### คำอธิบาย

คำสั่ง `getopts` คือคำสั่งในตัว Korn/POSIX Shell ที่เรียกข้อมูลอ็อปชันและอาร์กิวเมนต์อ็อปชันจากรายการพารามิเตอร์ อ็อปชันจะขึ้นต้นด้วย + (เครื่องหมายบวก) หรือ - (เครื่องหมายลบ) ตามด้วยอักขระ อ็อปชัน ที่ไม่ได้ขึ้นต้นด้วย + หรือ - จะลงท้ายด้วย *OptionString* แต่ละครั้ง ที่คำสั่ง `getopts` ถูกเรียกใช้ จะวางค่าของอ็อปชันถัดไปใน *Name* และดัชนีของอาร์กิวเมนต์ที่จะถูกประมวลผลในตัวแปรเซลล์ `OPTIND` เมื่อใดที่เรียกใช้เซลล์ `OPTIND` จะถูกเตรียมข้อมูลเบื้องต้นเป็น 1 เมื่ออ็อปชันขึ้นต้นด้วย + แล้ว + จะถูกนำหน้าค่าใน *Name*

หากอักขระใน *OptionString* ตามด้วย : (โคลอน) อ็อปชันนั้นจะถูกคาดหวังว่ามีอาร์กิวเมนต์ เมื่ออ็อปชันจำเป็นต้องมีอาร์กิวเมนต์อ็อปชัน คำสั่ง `getopts` จะวางไว้ในตัวแปร `OPTARG`

เมื่อพบอักขระอ็อปชันที่ไม่มีใน *OptionString* หรืออ็อปชันที่พบไม่มีอาร์กิวเมนต์อ็อปชัน ที่จำเป็น:

- หาก *OptionString* ไม่ขึ้นต้นด้วย : (โคลอน)
  - *Name* จะถูกตั้งค่า เป็นอักขระ ? (เครื่องหมายคำถาม)
  - `OPTARG` จะถูกยกเลิกการตั้งค่า และ
  - ข้อความวินิจฉัยถูกเขียนไปยังข้อผิดพลาด มาตรฐาน

เงื่อนไขนี้ถูกพิจารณาว่าเป็นข้อผิดพลาดที่ตรวจพบ ระหว่างที่อาร์กิวเมนต์ถูกนำเสนอในแอสพลีเคชันการเรียกใช้ แต่ไม่ใช่ข้อผิดพลาด ในการประมวลผลของคำสั่ง `getopts` ข้อความวินิจฉัย จะถูกเขียนตามที่ระบุ แต่สถานะการออกจะเป็น ศูนย์

- หาก *OptionString* ขึ้นต้น ด้วย : (โคลอน)

- *Name* จะถูกตั้งค่า เป็นอักขระ ? (เครื่องหมายคำถาม) สำหรับอ็อปชันที่ไม่รู้จัก หรือเป็นอักขระ : (โคลอน) สำหรับอ็อปชันที่จำเป็นแต่ไม่มีอยู่
- **OPTARG** จะถูกตั้งค่า เป็นอักขระอ็อปชันที่พบ และ
- ไม่มีเอาต์พุตถูกเขียนไปยังข้อผิดพลาดมาตรฐาน

ค่าใดๆ ต่อไปนี้จะระบุการสิ้นสุดอ็อปชัน: อ็อปชันพิเศษ -- การค้นหาอาร์กิวเมนต์ที่ไม่ได้ขึ้นต้นด้วย - หรือ + หรือการพบข้อผิดพลาด

เมื่อพบจุดสิ้นสุดอ็อปชัน:

- คำสั่ง `getopts` จะออกจากการทำงานด้วยค่าส่งคืนที่มากกว่าศูนย์
- **OPTARG** จะถูกตั้งค่า เป็นดัชนีของค่าที่ไม่ใช่อาร์กิวเมนต์อ็อปชัน โดยที่อาร์กิวเมนต์ -- ถูกพิจารณาว่า เป็นอาร์กิวเมนต์อ็อปชัน หากไม่มีค่าที่ไม่ใช่อาร์กิวเมนต์อ็อปชันอื่นๆ ปรากฏ ก่อนหน้า หรือค่า `##+1` หากไม่มีค่าที่ไม่ใช่อาร์กิวเมนต์อ็อปชัน
- *Name* จะถูกตั้งค่า เป็นอักขระ ? (เครื่องหมายคำถาม)

## พารามิเตอร์

ไอเท็ม

*OptionString*

คำอธิบาย

มีสตริงของอักขระอ็อปชันที่คำสั่ง `getopts` รู้จัก หากอักขระตามด้วยโคลอน อ็อปชันนั้นจะ ถูกคาดว่า มีอาร์กิวเมนต์ ซึ่งควรระบุเป็นอาร์กิวเมนต์แยกต่างหาก อ็อปชันสามารถแยกจากอาร์กิวเมนต์ด้วยช่องว่าง อักขระตัวแรก ใน *OptionString* เป็นตัวกำหนดวิธีที่คำสั่ง `getopts` จะดำเนินงานหากไม่รู้จักอักขระอ็อปชัน หรืออาร์กิวเมนต์อ็อปชัน ไม่มีอยู่

**หมายเหตุ:** อักขระเครื่องหมายคำถามและโคลอน ต้องไม่ถูกแอสพลีเคชันใช้เป็นอักขระอ็อปชัน การใช้อักขระอื่นๆ ที่ไม่ใช่ตัวอักษรผสมตัวเลขจะก่อให้เกิดผลลัพธ์ที่ไม่สามารถระบุได้

ชื่อ

*Argument ...*

ตั้งค่าโดยคำสั่ง `getopts` เป็นอักขระอ็อปชัน ที่พบ

สตริงอย่างน้อยหนึ่งสตริงที่คั่นด้วย white space ซึ่งตรวจสอบโดยคำสั่ง `getopts` เพื่อดูความถูกต้องของอ็อปชัน หาก

*Argument* ถูกข้าม จะใช้พารามิเตอร์ตามตำแหน่งแทน โปรดดูที่ การแทนที่พารามิเตอร์ในคอร์นเชลล์ หรือ POSIX เชลล์

ในคอร์นเชลล์ เพื่อดูข้อมูลเพิ่มเติมเกี่ยวกับพารามิเตอร์ตามตำแหน่ง

**หมายเหตุ:** โดยทั่วไป คุณจะไม่ระบุ *Argument* เป็นส่วนหนึ่งของคำสั่ง `getopts` แต่อาจเป็นประโยชน์เมื่อทำการดีบักสคริปต์ของคุณ

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย

0 อ็อปชันที่ระบุ หรือไม่ระบุโดย *OptionString* ถูกพบ

>0 พบจุดสิ้นสุดอ็อปชัน หรือมีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. คำสั่ง `getopts` ต่อไปนี้ ระบุว่า a, b และ c เป็นอ็อปชันที่ใช้ได้ และอ็อปชัน a และ c มีอาร์กิวเมนต์:

```
getopts a:bc: OPT
```

2. คำสั่ง `getopts` ต่อไปนี้ ระบุว่า a, b และ c เป็นอ็อปชันที่ใช้ได้ และอ็อปชัน a และ b มีอาร์กิวเมนต์และ `getopts` ตั้งค่าของ OPT เป็น ? เมื่อพบอ็อปชันที่ไม่ได้กำหนดบนบรรทัดคำสั่ง:

```
getopts :a:b:c OPT
```

3. สคริปต์ต่อไปนี้ จะวิเคราะห์ค่าและแสดง อาร์กิวเมนต์:

```

aflag=
bflag=

while getopts ab: name
do
    case $name in
        a)    aflag=1;;
        b)    bflag=1
                bval="$OPTARG";;
        ?)    printf "Usage: %s: [-a] [-b value] args\n" $0
                exit 2;;
    esac
done

if [ ! -z "$aflag" ]; then
    printf "Option -a specified\n"
fi

if [ ! -z "$bflag" ]; then
    printf 'Option -b "%s" specified\n' "$bval"
fi

shift $((OPTIND -1))
printf "Remaining arguments are: %s\n" "$*"

```

**ข้อมูลที่เกี่ยวข้อง:**

คำสั่ง Korn เชลล์หรือ POSIX เชลล์

## คำสั่ง getrunmode

### วัตถุประสงค์

แสดงโหมดที่ระบบกำลังรันอยู่

### ไวยากรณ์

**getrunmode**

### คำอธิบาย

คำสั่ง **getrunmode** แสดงโหมดที่ระบบ กำลังรันอยู่ รันโหมดจะเป็นโหมด CONFIGURATION หรือโหมด OPERATIONAL

### ตัวอย่าง

ในการเรียกข้อมูลรันโหมดให้ป้อน:

```
getrunmode
```

### ไฟล์

ไอเท็ม	คำอธิบาย
/usr/sbin/getrunmode	มีคำสั่ง getrunmode

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง setrunmode

Trusted AIX

---

## คำสั่ง getsecconf

### วัตถุประสงค์

แสดงแฟล็กการรักษาความปลอดภัยระบบ

### ไวยากรณ์

```
getsecconf { -c | -o }
```

### คำอธิบาย

คำสั่ง `getsecconf` แสดงแฟล็กการรักษาความปลอดภัย ระบบ เมื่อเรียกใช้โดยไม่มีอ็อปชันใดๆ คำสั่ง `getsecconf` จะแสดงแฟล็กการรักษาความปลอดภัยที่เป็นไปตามโหมดของระบบที่กำลัง รันอยู่

### แฟล็ก

ไอเท็ม	คำอธิบาย
-c	ระบุโหมด CONFIGURATION
-o	ระบุโหมด OPERATIONAL

### สถานะออก

คำสั่ง `getsecconf` ส่งคืนค่า `exit` ต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	การเรียกใช้งานสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

### ตัวอย่าง

- ในการแสดงแฟล็กการรักษาความปลอดภัยระบบในโหมด CONFIGURATION ให้ป้อน:  

```
getsecconf -c
```
- ในการแสดงแฟล็กการรักษาความปลอดภัยระบบในโหมด OPERATIONAL ให้ป้อน:  

```
getsecconf -o
```

### ไฟล์

ไอเท็ม  
/usr/sbin/getseconf

คำอธิบาย  
มีคำสั่ง getseconf

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง setseconf

Trusted AIX

---

## คำสั่ง getsyslab

### วัตถุประสงค์

แสดงเลเบลต่ำสุดและสูงสุดของระบบ

### ไวยากรณ์

getsyslab

### คำอธิบาย

คำสั่ง `getsyslab` ใช้เพื่อแสดง sensitivity label (SL) ระบบต่ำสุดและสูงสุด และ integrity label (TL) ระบบต่ำสุด และสูงสุด

### ความปลอดภัย

คำสั่ง `getsyslab` เป็นคำสั่งที่ต้องมีสิทธิ์ การรัน คำสั่งนี้ได้สำเร็จจำเป็นต้องใช้การพิสูจน์ตัวตนต่อไปนี้:

ไอเท็ม  
aix.mls.system.label.read

คำอธิบาย  
จำเป็นสำหรับการแสดงรายการเลเบลระบบ

### ไฟล์ที่เข้าถึง:

ไอเท็ม  
Mode  
r

คำอธิบาย  
ไฟล์  
/etc/security/enc/LabelEncodings

### ตัวอย่าง

ในการแสดงเลเบลระบบให้ป้อน:

getsyslab

### ไฟล์

ไอเท็ม	คำอธิบาย
/usr/sbin/getsyslab	มีคำสั่ง getsyslab
/etc/security/enc/LabelEncodings	ไฟล์การเข้ารหัสเลเบลดีฟอลต์ของระบบ

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง getsyslab

Trusted AIX

---

## คำสั่ง `gettable`

### วัตถุประสงค์

รับค่าตารางโฮสต์รูปแบบ Network Information Center (NIC) จากโฮสต์

### ไวยากรณ์

```
/usr/sbin/gettable [ -v ] Host [ OutFile ]
```

### คำอธิบาย

คำสั่ง `/usr/sbin/gettable` ถูกใช้เพื่อจัดหารายชื่อโฮสต์มาตรฐาน NIC จากเซิร์ฟเวอร์ที่ระบุโดย พารามิเตอร์ `Host` ตาราง หากถูกเรียกข้อมูล จะถูกวางไว้ในไฟล์ที่ระบุโดยพารามิเตอร์ `OutFile`

คำสั่ง `gettable` เปิดการเชื่อมต่อ Transmission Control Protocol (TCP) กับพอร์ตที่ระบุใน ข้อกำหนดคุณลักษณะเซอร์วิส สำหรับพารามิเตอร์ `Host` จากนั้น ทำการร้องขอชื่อทั้งหมด และข้อมูลผลลัพธ์จะอยู่ใน เอาต์พุตไฟล์

คำสั่ง `gettable` ทำงานดีที่สุด เมื่อใช้ร่วมกับคำสั่ง `htable` ซึ่งแปลงรูปแบบไฟล์มาตรฐาน NIC เป็นที่ใช้โดยรูทีนการค้นหาเน็ตเวิร์กไลบรารี

### แฟล็ก

ไอเท็ม	คำอธิบาย
-v	รับเฉพาะหมายเลขเวอร์ชันแทนการรับตารางค่าตารางทั้งหมดและวางเอาต์พุตใน <code>OutFile</code> หรือโดยค่าดีฟอลต์ในไฟล์ชื่อ <code>hosts.ver</code>

### พารามิเตอร์

ไอเท็ม	คำอธิบาย
Host	ระบุเซิร์ฟเวอร์ที่ให้ข้อมูลตารางโฮสต์
OutFile	ระบุไฟล์ที่คุณต้องการเก็บข้อมูลตารางโฮสต์ หากคุณใช้คำสั่ง <code>gettable</code> โดยไม่มี แฟล็ก <code>-v</code> ชื่อไฟล์ดีฟอลต์จะเป็น <code>hosts.txt</code>

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `htable`” ในหน้า 818

### ข้อมูลที่เกี่ยวข้อง:

Transmission Control Protocol (TCP)

โปรโตคอล TCP/IP

---

## คำสั่ง `gettrc`

### วัตถุประสงค์

จัดการการรวบรวมไฟล์การติดตาม

### ไวยากรณ์

```
gettrc [ -c ] [ -C dirname ] [ -m ] [ -M dirname ] [ -s ] [ -S filename ]
```

### คำอธิบาย

คำสั่ง `gettrc` คือสคริปต์ที่ใช้ร่วมกับ คำสั่ง `snap` โดยจะจัดการการรวบรวมไฟล์การติดตามระบบ ไฟล์ `lightweight memory trace (LMT)` และไฟล์ `component trace (CT)`

### แฟล็ก

แฟล็ก

`-c`

`-C dirname`

`-m`

`-M dirname`

`-s`

`-S filename`

คำอธิบาย

รวบรวมไฟล์การติดตามคอมพิวเตอร์

รวบรวมไฟล์การติดตามคอมพิวเตอร์จากไดเรกทอรี ที่ระบุโดย *dirname*

รวบรวมไฟล์การติดตามหน่วยความจำ

รวบรวมไฟล์การติดตามหน่วยความจำขนาดเล็กจาก ไดเรกทอรีที่ระบุโดย *dirname*

รวบรวมไฟล์การติดตามระบบ

รวบรวมไฟล์การติดตามจากไดเรกทอรี ที่ระบุโดย *filename*

### สถานะออก

ไอเท็ม

0

>0

คำอธิบาย

คำสั่งดำเนินการเสร็จด้วยผลสำเร็จ

มีข้อผิดพลาดเกิดขึ้น

### ตัวอย่าง

1. ในการใช้ `gettrc` ร่วมกับคำสั่ง `snap` เพื่อเรียกข้อมูลประเภทที่แตกต่างกันของไฟล์การติดตามให้ป้อน:

```
snap "gettrc -c -C dirname -m -M dirname -s -S filename"
```

คำสั่งนี้จะส่งคืนไฟล์การติดตามระบบ, ไฟล์ `LMT` และไฟล์ `CT` รวมถึงไฟล์ ที่แสดงรายการในไดเรกทอรีที่ระบุโดย *dirname*

### Location

`/usr/lib/ras/snapscripts/gettrc`

# ไฟล์

/usr/lib/ras/cpufmt

/etc/trcfmt

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง snap

---

## คำสั่ง getty

### วัตถุประสงค์

ตั้งค่าคุณสมบัติของพอร์ต

### ไวยากรณ์

```
getty [[-r|-u|-U][ -d][ -H HeraldString ][ -M motdFile ][ -N ]] PortName
```

### คำอธิบาย

คำสั่ง `getty` ตั้งค่า และจัดการสายเชื่อมต่อเทอร์มินัลและพอร์ต คำสั่ง `getty` ถูกรันโดยคำสั่ง `init` คำสั่ง `getty` ถูกลิงก์ไปยังโปรแกรม Terminal State Manager โปรแกรม Terminal State Manager จัดให้มีการควบคุมเทอร์มินัลและล็อกอินฟังก์ชันร่วมกัน

คุณสามารถกำหนดค่าคำสั่ง `getty` เพื่อสร้าง โสมไดเร็กทอรีของคุณในตอนล็อกอิน หากคุณยังไม่มีโสมไดเร็กทอรี คำสั่ง `getty` เรียกใช้คำสั่ง `mkuser.sys` เพื่อสร้างโสมไดเร็กทอรี และกำหนดแอคเคาต์เอง ในการเปิดใช้ความสามารถนี้ให้ตั้งค่าแอตทริบิวต์ `mkhomeatlogin` ของ `usw stanza` ในไฟล์ `/etc/security/login.cfg` เป็น `true`

**หมายเหตุ:** คำสั่ง `getty` จะไม่ถูกป้อนบนบรรทัดคำสั่ง

เมื่อถูกเรียกใช้ เป็นคำสั่ง `getty` โปรแกรม Terminal State Manager จะจัดให้มีฟังก์ชันการจัดการพอร์ต ปกติที่ประกอบด้วย:

#### ไอเท็ม

Bidirectional use

Line speed

Parity

Delays

Character set mapping

Logger Program

Character and line erase

Echoing mode

#### คำอธิบาย

อนุญาตให้สายเชื่อมต่อเทอร์มินัลใช้เตรียมข้อมูลเบื้องต้นและยอมรับการเชื่อมต่อ

ตั้งค่าอัตรา baud สำหรับการส่งและการรับ

ตั้งค่าพาริตีเป็นคู่ คี่ หรือไม่มี

ตั้งกำหนดเวลาสำหรับการปิดแควร์ บรรทัดใหม่ และ ป้อนกระดาษ

ตั้งค่าชุดอักขระชุดอักขระสำหรับขนาดตัวพิมพ์ แท็บ และรหัสควบคุม การแสดงผล

ระบุโปรแกรมที่ใช้เพื่อบันทึกการทำงานที่ผู้ใช้เข้าสู่ระบบ หากแอตทริบิวต์ถูกตั้งค่า การ

ประมวลผล Secure Attention Key (SAK) จะถูกปิดใช้งาน หากแอตทริบิวต์ไม่ถูกตั้งค่า

จะใช้ค่าดีฟอลต์เป็น `/usr/sbin/login` แอตทริบิวต์ `logger` มีอยู่ภายใน ฐานข้อมูล Object

Data Manager (ODM)

ตั้งค่าการเคาะคีย์บอร์ดที่ใช้สำหรับการลบอักขระและบรรทัด

ตั้งค่า `echo` ไปยังโลคัลเทอร์มินัล

เมื่อคำสั่ง `getty` ถูกเรียกใช้ จะมีขั้นตอนต่อไปนี้จะเกิดขึ้น:

1. การปกป้องพอร์ตถูกตั้งค่าตาม แอ็ททริบิวต์ **owner** และ **protection** ในฐานข้อมูล ODM หากแอ็ททริบิวต์เหล่านี้ไม่ถูกระบบจะใช้ค่าดีฟอลต์ เป็น **root** และ **622**
2. พอร์ตที่ระบุโดยพารามิเตอร์ **PortName** ถูกเปิดขึ้น หากการตรวจหาคลื่นพาหะมีอยู่บนพอร์ต การเปิดพอร์ตจะยังไม่สำเร็จจนกว่าจะมีคลื่นพาหะแสดงอยู่ หรืออีกกระบวนการหนึ่งสูญเสียคลื่นพาหะไปยังพอร์ต
3. พอร์ตที่ระบุจากลูกล็อก หาก คำสั่ง **getty** รันด้วยแฟล็ก **-u** หรือ **-r** คำสั่งจะพยายาม ล็อกพอร์ต หากพอร์ตถูกล็อกอยู่แล้ว คำสั่งจะรอจนกว่าพอร์ตจะพร้อมใช้งาน จากนั้นออกจากโปรแกรม หากระบุแฟล็ก **-r** คำสั่ง **getty** จะรอจนได้รับ ข้อมูลเป็นจำนวนไบต์บนพอร์ตก่อนดำเนินการต่อ
4. เทอร์มินัลแอ็ททริบิวต์ถูกตั้งค่าตาม ข้อมูลการกำหนดค่าสำหรับพอร์ตที่ระบุ การประมวลผล Secure Attention Key สามารถถูกเปิดใช้งานที่จุดนี้โดยขึ้นอยู่กับข้อกำหนดค่าระบบ
5. ข้อความประกาศจะถูกเขียนไปยังพอร์ต ที่ระบุ
6. ชื่อล็อกอินถูกอ่านจากพอร์ต ที่ระบุ หากเกิดข้อผิดพลาดการกำหนดกรอบ หรือมีการหยุดชะงักเกิดขึ้น คำสั่ง **getty** จะทำซ้ำขั้นตอนที่สี่และห้ากับกลุ่มถัดไปของเทอร์มินัลแอ็ททริบิวต์ ที่กำหนดค่า คำสั่งนี้ใช้มากเพื่อใช้ซ้ำอัตรา baud สำหรับโมเด็ม แต่ฟิลต์ ODM ใดๆ (ยกเว้น **logmodes** และ **runmodes**) อาจถูกนำใช้ใหม่โดยการป้อนรายการของค่าที่ค้นด้วยเครื่องหมายจุลภาคในฐานข้อมูล ODM
7. เทอร์มินัลโหมดถูกรีเซ็ตตาม พารามิเตอร์ **runmodes** และชื่อ ล็อกอิน หากชื่อล็อกอินสิ้นสุดด้วยบรรทัดใหม่ คำสั่ง **getty** จะเปิดใช้การแก้ไขบรรทัดใหม่ หากอักขระแบบตัวอักษร เป็นตัวพิมพ์ใหญ่ ผู้ใช้ได้รับพร้อมท์เพื่อล็อกอินโดยใช้อักขระตัวพิมพ์เล็กหากเป็นไปได้ และการแก้ไขจากตัวพิมพ์เล็กเป็นตัวพิมพ์ใหญ่ จะถูกเปิดใช้งาน
8. หากโปรแกรมถูกระบุโดยพารามิเตอร์ **logger** โปรแกรมจะถูกเรียกใช้งานและการประมวลผล Secure Attention Key ถูกเปิดใช้งาน มิฉะนั้น โปรแกรม Terminal State Manager จะดำเนินการล็อกอินระบบ แบบมาตรฐาน

หมายเหตุ: ถ้าลำดับ Secure Attention Key ถูกพิมพ์ระหว่างล็อกอินของผู้ใช้ ผู้ใช้จะล็อกอินเข้าสู่เซลล์ที่ไว้วางใจ (ถ้าระบบถูกกำหนดคอนฟิกโดยที่พอร์ตได้รับความไว้วางใจและผู้ใช้ได้รับอนุญาตบนพาหะที่ไว้วางใจ)

## แฟล็ก

ไอเท็ม	คำอธิบาย
<b>-d</b>	จัดใหม่ข้อมูลการดีบั๊ก
<b>-H HeraldString</b>	ระบุข้อความประกาศอื่นเพื่อเขียนบนพอร์ต เพื่อพร้อมรับชื่อล็อกอิน สตริงข้อความต้องเป็นหนึ่งคำและไม่สามารถมีช่องว่างใดๆ สตริงนี้จะมาก่อนข้อความ ประกาศที่กำหนดในไฟล์ <b>/etc/security/login.cfg</b> หากไม่ระบุสตริงที่มีอ็อพชันนี้ หรือในไฟล์ <b>login.cfg</b> การประกาศค่าดีฟอลต์จากแค็ตตาล็อกข้อความจะถูกใช้
<b>-M motdFile</b>	ระบุพาธไปยังข้อความอื่นของไฟล์วิน หากไม่ระบุ ค่านี้จะเป็น <b>/etc/motd</b> เป็น ค่าดีฟอลต์
<b>-N</b>	ทำให้ <b>getty</b> ข้ามการตรวจสอบใดๆ กับ ID กระบวนการในไฟล์ <b>/etc/utmp</b> ค่านี้อนุญาตให้ กระบวนการที่นอกเหนือจากล็อกอินเซลล์ที่ต่ำสุดเพื่อ <b>exec getty</b>
<b>-r</b>	ทำให้พอร์ตพร้อมใช้งานสำหรับการแบ่งใช้ (สองทิศทาง) หากทำการล็อกไม่สำเร็จ คำสั่ง <b>getty</b> จะรอจนทำการล็อกได้ จากนั้นออกจากการทำงาน หากการล็อกสำเร็จ คำสั่ง <b>getty</b> จะรอไปต่อข้อมูลจำนวนหนึ่งบน พอร์ตหลังการล็อกพอร์ต
<b>-u</b>	ทำให้พอร์ตพร้อมใช้งานสำหรับการแบ่งใช้ (สองทิศทาง) หากทำการล็อกไม่สำเร็จ คำสั่ง <b>getty</b> จะรอจนทำการล็อกได้ จากนั้นออกจากการทำงาน
<b>-U</b>	เหมือนกับแฟล็ก <b>-u</b> ยกเว้น <b>getty</b> จะไม่รอให้การล็อกพร้อม แฟล็กนี้ทำให้พอร์ตพร้อมใช้ โดยไม่คำนึงถึงการล็อก

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: โปรแกรมนี้ควรถูกติดตั้ง เป็นโปรแกรมใน Trusted Computing Base ที่สามารถเรียกใช้งานโดยผู้ใช้ใดๆ และ **setuid** เป็น **root**

## ตัวอย่าง

ในการเปิดใช้งานการล็อกเข้าสู่ tty0 ให้เพิ่มบรรทัด ต่อไปนี้ในไฟล์ `/etc/inittab`:

```
tty0:2:respawn: /usr/sbin/getty /dev/tty0
```

คำสั่งนี้เตรียมข้อมูลเบื้องต้นสำหรับพอร์ต `/dev/tty0` และตั้งค่าคุณสมบัติของพอร์ต

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/sbin/getty</code>	มีคำสั่ง <code>getty</code>
<code>/etc/locks</code>	มีล็อกไฟล์ที่ป้องกันการใช้อุปกรณ์สื่อสารหลายอย่าง และการเรียกใช้ระบบโมดหลาย ๆ การเรียกใช้
<code>/usr/sbin/login</code>	คำสั่ง <code>login</code>
<code>/etc/security/login.cfg</code>	มีการกำหนดค่าพอร์ตล็อกอิน
<code>/etc/motd</code>	มีข้อความของวันที่แสดงหลังการล็อกอิน
<code>/usr/bin/setmaps</code>	คำสั่ง <code>setmaps</code>
<code>/etc/utmp</code>	มีข้อมูลเกี่ยวกับผู้ใช้ที่ล็อกเข้าสู่ระบบ

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `login`

คำสั่ง `shell`

`telinit` หรือ `init`

ภาพรวม Object Data Manager (ODM) สำหรับโปรแกรมเมอร์

---

## คำสั่ง `gprof`

### วัตถุประสงค์

แสดงข้อมูลโปรไฟล์กราฟการเรียกใช้

### ไวยากรณ์

```
/usr/ccs/bin/gprof [ -b ] [ -c [ filename ] ] [ -e Name ] [ -E Name ] [ -f Name ] [ -g filename ] [ -i filename ] [ -p filename ] [ -F Name ] [ -L PathName ] [ -s ] [ -x [ filename ] ] [ -z ] [ a.out [ gmon.out ... ] ]
```

### คำอธิบาย

คำสั่ง `gprof` จะสร้าง โปรไฟล์การเรียกใช้งานของโปรแกรม C, FORTRAN หรือ COBOL ผลของรูทีนที่ถูกเรียกใช้ จะรวมอยู่ในโปรไฟล์ของตัวเรียกแต่ละตัว คำสั่ง `gprof` มีประโยชน์ในการระบุวิธีที่โปรแกรมใช้รีซอร์สตัวประมวลผล เมื่อต้องการหาว่าฟังก์ชันใด (รูทีน) ใดในโปรแกรมกำลังใช้ตัวประมวลผล คุณสามารถสร้างตัวประมวลผลโปรแกรม โดยใช้คำสั่ง `gprof`

ข้อมูลโปรไฟล์ ถูกนำมาจากไฟล์โปรไฟล์กราฟการเรียกใช้ (`gmon.out` โดย ดีฟอลต์) ที่สร้างโดยโปรแกรมที่คอมไพล์ด้วยคำสั่ง `cc` โดยใช้อ็อปชัน `-pg` อ็อปชัน `-pg` ยังลิงก์ใน เวอร์ชันของรูทีนไลบรารีที่คอมไพล์สำหรับการสร้างโปรไฟล์ และอ่าน ตารางสัญลักษณ์ในไฟล์อ็อบเจกต์ที่ระบุชื่อ (`a.out` โดยดีฟอลต์) โดยรวมเข้ากับไฟล์โปรไฟล์กราฟการเรียกใช้ หากมีการระบุมากกว่าหนึ่งโปรไฟล์ เอาต์พุตของคำสั่ง `gprof` จะแสดง ผลรวมของข้อมูลโปรไฟล์ในไฟล์โปรไฟล์ที่ระบุ

อ็อปชัน `-pg` ทำให้คอมไพเลอร์แทรกการเรียกใช้ยูทิลิตี้ `mcount` ในโค้ดอ็อบเจกต์ที่สร้างขึ้นสำหรับแต่ละฟังก์ชันที่คอมไพเลอร์ของโปรแกรมของคุณ ระหว่างการเรียกใช้งาน โปรแกรม แต่ละครั้งที่พาเรนต์เรียกใช้ซายด์ฟังก์ชัน ซายด์จะเรียกใช้ยูทิลิตี้ `mcount` เพื่อเพิ่มค่าตัวนับสำหรับคู่พาเรนต์-ซายด์นั้น โปรแกรมที่ไม่ถูกคอมไพล์ใหม่ด้วยอ็อปชัน `-pg` จะไม่มียูทิลิตี้ `mcount` แทรก และทำให้ไม่เก็บเร็กคอร์ดของผู้ที่เรียกใช้ยูทิลิตี้

**หมายเหตุ:** สัญลักษณ์จากชื่อไฟล์อ็อบเจกต์ C++ จะถูกเปลี่ยนก่อนที่จะถูกใช้

ตัวแปรสถานะแวดล้อม `GPROF` สามารถใช้เพื่อตั้งค่าอ็อปชันต่างๆ สำหรับการทำให้โปรไฟล์ไวยากรณ์ของตัวแปรสถานะแวดล้อมนี้ถูกกำหนดดังนี้:

`GPROF = profile:<profile-type>,scale:<scaling-factor>,file:<file-type>,filename:<filename>`

โดย:

- `<profile-type>` อธิบายชนิดของการสร้างโปรไฟล์ที่ต้องดำเนินการ ซึ่งสามารถเป็นกระบวนการหรือเธรด ชนิด 'process' ระบุว่าโปรไฟล์ เป็นกลุ่มย่อยๆ ในระดับกระบวนการ ในขณะที่ 'thread' ระบุว่าโปรไฟล์เป็นกลุ่มย่อยๆ ในระดับเธรด
- `<scaling-factor>` อธิบายจำนวนของหน่วยความจำที่ต้องจัดสรร สำหรับโปรไฟล์กราฟการเรียกใช้โดยดีฟอลต์ค่าสเกลจะเป็น 2 สำหรับการสร้างโปรไฟล์ระดับกระบวนการ และเป็น 8 สำหรับการสร้างโปรไฟล์ระดับเธรด ค่าสเกลที่เป็น 2 ระบุว่ามีการจัดสรรหน่วยความจำขนาดครึ่งหนึ่งของขนาดกระบวนการสำหรับทุกกระบวนการหรือเธรด ค่าสเกลที่เป็น 8 ระบุว่ามีการจัดสรรหน่วยความจำขนาดหนึ่งในแปดของขนาดกระบวนการ สำหรับทุกกระบวนการของเธรด หน่วยความจำเป็นพื้นที่บัฟเฟอร์ที่ใช้เก็บข้อมูลกราฟ การเรียกใช้
- `<file-type>` อธิบายชนิดของไฟล์ `gmon.out` ที่ต้องการ ค่า `multi` ระบุว่าต้องการไฟล์ `gmon.out` ต่อกระบวนการ ค่าของ `multithread` ระบุว่าต้องการไฟล์ `gmon.out` หนึ่งไฟล์ ต่อเธรด หากแอ็พพลิเคชันถูกทำให้โปรไฟล์ด้วยอ็อปชัน `-pg` แอ็พพลิเคชันจะ forks จากนั้นการระบุชนิดไฟล์เป็น `multi` จะสร้างไฟล์ `gmon.out` สำหรับกระบวนการพาเรนต์และไฟล์อื่นสำหรับกระบวนการชายน์ หลักการตั้งชื่อสำหรับไฟล์ `gmon.out` ที่สร้างขึ้น มีดังต่อไปนี้:
  - สำหรับชนิดไฟล์ `multi`: `<prefix>-processname-pid.out`
  - สำหรับชนิดไฟล์ `multithread`: `<prefix>-processname-pid-Pthread<threadid>.out``<prefix>` เป็น `gmon` โดยดีฟอลต์ คุณสามารถกำหนดส่วนนำหน้าของคุณเองโดยใช้พารามิเตอร์ `filename` ของตัวแปรสถานะแวดล้อม `GPROF`
- `<filename>` อธิบายส่วนนำหน้าที่ต้องการใช้สำหรับ ไฟล์ `gmon.out` ที่สร้างขึ้น โดยดีฟอลต์ ส่วนนำหน้าคือ `gmon`

**หมายเหตุ:** การระบุ `profile:thread` จะสร้างไฟล์ `gmon.out` รูปแบบที่สามารถอ่านโดยคำสั่ง `AIX 5.3 gprof` เท่านั้น หากต้องการไฟล์ `gmon.out` รูปแบบเก่าและยังต้องการระบุ `profile:thread` ดังนั้นคุณต้องระบุ `file:multithread` โดยจะสร้างไฟล์ `gmon.out` รูปแบบเก่าต่อเธรด ดังนั้น หากแอ็พพลิเคชันของคุณ มี 2 เธรด ดังนั้นไฟล์ `gmon.out` 2 ไฟล์จะถูกสร้าง หนึ่งไฟล์ต่อเธรด โดยใช้หลักการตั้งชื่อ คุณไม่สามารถเปิดใช้งานการทำโปรไฟล์ระดับเธรด โดยการคอมไพล์แอ็พพลิเคชันด้วยแฟล็ก `-pg` ใน AIX 5.2 หรือก่อนหน้านั้นและรัน ใน AIX 5.3 เมื่อต้องการเปิดใช้งาน การทำโปรไฟล์ระดับเธรด คุณต้องคอมไพล์แอ็พพลิเคชันดังกล่าวด้วยแฟล็ก `-pg` ใน AIX 5.3 และใหม่กว่า

คำสั่ง `gprof` ก่อให้เกิดรายการสาม รายการ:

1. ชั้นแรก โปรไฟล์แบบไม่มีลำดับชั้นจะถูกสร้างขึ้นคล้ายกับโปรไฟล์ที่จัดทำโดยคำสั่ง `prof` รายการนี้จะแสดงเวลาการประมวลผลทั้งหมด และจำนวนการเรียกใช้สำหรับแต่ละฟังก์ชันในโปรแกรม ซึ่งเรียงลำดับ ตามเวลาที่ลดลง จากนั้นเวลาจะถูกกระจายไปตามขอบของกราฟการเรียกใช้ วงรอบ ถูกสำรวจและการเรียกใช้ในวงรอบถูกจัดทำเพื่อแบ่งเวลาของวงรอบ

2. รายการที่สองแสดงฟังก์ชันที่เรียงลำดับตามเวลาที่แสดง รวมถึงเวลาของการสืบทอด กราฟการเรียกใช้ ด้านล่างของรายการ ฟังก์ชันแต่ละรายการคือชายด์ของกราฟการเรียกใช้ (โดยตรง) ที่มีการระบุ เวลาที่ถูกกระจายไปยังฟังก์ชันนี้ เช่นเดียวกับที่แสดงด้านบน ฟังก์ชันจะแสดงว่าเวลาของ ฟังก์ชันและเวลาของการสืบทอดถูกกระจาย ไปยังพาเรนต์กราฟการเรียกใช้ (โดยตรง) อย่างไร
3. วงรอบยังถูกแสดงเช่นกัน โดยมีรายการสำหรับวงรอบโดยรวมและรายการ ของสมาชิกของวงรอบ และการแบ่งเวลาและจำนวนการเรียกใช้ของวงรอบ

**หมายเหตุ:** หากอินพุตสำหรับคำสั่ง `gprof` มีข้อมูลการทำโปรไฟล์ ระดับเธรด (ไฟล์ `gmon.out` รูปแบบ) ดังนั้น คำสั่ง `gprof` จะสร้างสามไอเท็มที่ระบุสำหรับทุกเธรด เริ่มต้นด้วย รายงานสะสม ตามด้วยรายงานต่อเธรด (เรียงลำดับ ID เธรดจากน้อยไปมาก)

คำสั่ง `gprof` ยังสามารถใช้เพื่อวิเคราะห์ โปรไฟล์การเรียกใช้งานของโปรแกรมบนเครื่องรีโมต ซึ่งยังสามารถทำได้โดยการรันคำสั่ง `gprof` โดยใช้อ็อปชัน `-c` บนไฟล์โปรไฟล์กราฟการเรียกใช้ (`gmon.out` โดย ดีพอลต์) เพื่อสร้างไฟล์ (`gprof.remote` โดย ดีพอลต์) ซึ่ง จากนั้นสามารถประมวลผลบนเครื่องรีโมต หากต้องการใช้ไฟล์ โปรไฟล์กราฟการเรียกใช้อื่นที่ไม่ใช่ `gmon.out` ชื่อไฟล์โปรไฟล์กราฟการเรียกใช้ ต้องระบุหลังจาก `-c Filename` และ ชื่อที่สามารถเรียกใช้ได้ `Filename` ต้องถูกระบุไว้ หากแอตทริบิวต์ `file` ของ ตัวแปรสถานะแวดล้อม `GPROF` ถูกตั้งค่าเป็น `multi` ไฟล์ `gmon.out` หลายไฟล์ ถูกสร้างขึ้น โดยมีไฟล์ `gmon.out` หนึ่งไฟล์สำหรับแต่ละ PID เมื่อรัน โปรแกรม forks อ็อปชัน `-x` สามารถใช้บนเครื่อง รีโมตเพื่อประมวลผลไฟล์ `gprof.remote` (โดยค่าดีพอลต์) เพื่อสร้างรายงานโปรไฟล์

### การทำโปรไฟล์ด้วยวิธีที่น้อย fork และ exec

การทำโปรไฟล์โดยใช้ คำสั่ง `gprof` อาจสร้างปัญหาหากโปรแกรมของคุณรันที่น้อย `fork` หรือ `exec` บนกระบวนการหลายคำสั่ง และพร้อมกัน การทำโปรไฟล์คือแอตทริบิวต์ของสถานะแวดล้อม ของแต่ละกระบวนการ ดังนั้นหากคุณกำลังทำโปรไฟล์ กระบวนการที่ forks กระบวนการใหม่ กระบวนการชายด์จะถูกโปรไฟล์เช่นกัน อย่างไรก็ตาม ทั้งสองกระบวนการจะเขียนไฟล์ `gmon.out` ในไดเรกทอรีที่คุณรันกระบวนการพาเรนต์นั้น โดยเขียนทับหนึ่งในไฟล์เหล่านั้น คำสั่ง `tprof` ถูกแนะนำให้ใช้สำหรับการทำโปรไฟล์หลายกระบวนการ คุณสามารถใช้ `file:multi` เพื่อ ป้องกันการลบไฟล์ `gmon.out` ของกระบวนการพาเรนต์ `file:multi` โดยใช้ หลังการตั้งชื่อของ AIX เพื่อสร้างไฟล์ `gmon.out` ดังนั้นกระบวนการชายน์ ไฟล์ `gmon.out` จะไม่มีชื่อเดียวกับพาเรนต์ ซึ่งป้องกันการเขียนทับ

### การทำโปรไฟล์โดยไม่มีซอร์สโค้ด

หากคุณไม่มีซอร์สสำหรับโปรแกรมของคุณ คุณสามารถสร้าง โปรไฟล์โดยใช้คำสั่ง `gprof` โดยไม่ต้องคอมไพล์ใหม่อย่างไรก็ตาม คุณต้องสามารถลิงก์โปรแกรมโมดูลของคุณกับคำสั่งคอมไพลเลอร์ที่เหมาะสมอีกครั้ง (ตัวอย่างเช่น `cc` สำหรับ C) หากคุณ ไม่คอมไพล์ใหม่ คุณจะไม่ได้รับจำนวนความถี่ในการเรียกใช้ แม้ว่าโปรไฟล์เปล่า ยังคงใช้ได้โดยไม่ต้องมีจำนวนนั้น เพื่อเป็นประโยชน์เพิ่มเติม โปรแกรมของคุณจะรัน เร็วเกือบเท่ากับที่ทำตามปกติ ต่อไปนี้อธิบายวิธีการโปรไฟล์:

```
cc -c dhry.c          # Create dhry.o without call counting code.
cc -pg dhry.o -L/lib -L/usr/lib -o dhryfast
                    # Re-link (and avoid -pg libraries).
dhryfast            # Create gmon.out without call counts.
gprof >dhryfast.out # You get an error message about no call counts
                    # -- ignore it.
```

ผลลัพธ์ของการรันโดยไม่มีจำนวนการเรียกใช้คือบางฟังก์ชันที่เรียกใช้งานอย่างรวดเร็ว (ซึ่งคุณทราบว่าจะต้องถูกเรียกใช้) ไม่ปรากฏในรายการ แม้ว่าจะไม่เป็นไปโดยธรรมชาติ ผลลัพธ์นี้เป็นค่าปกติสำหรับคำสั่ง `gprof` คำสั่ง `gprof` แสดง เฉพาะฟังก์ชันที่ถูกเรียกใช้อย่างน้อยหนึ่งครั้ง หรือถูกเรียกใช้อย่างน้อยหนึ่งสัญญาณนาฬิกา แม้ว่าจะรันอยู่ ฟังก์ชันที่เรียกใช้งาน

อย่างรวดเร็ว มักไม่ได้รับสัญญาณพิก้า เนื่องจากการนับการเรียกใช้ถูกหยุดทำงานชั่วคราว ฟังก์ชัน เล็กๆ เหล่านี้จะไม่ถูกแสดงในรายการ (คุณสามารถรับค่าจำนวนการเรียกใช้สำหรับรันใหม่ รูทีนได้โดยการไม่ระบุอ็อปชัน -L บนบรรทัดคำสั่ง cc -pg )

## การใช้หน่วยความจำจริง น้อยลง

การทำโปรไฟล์ด้วยคำสั่ง gprof สามารถทำให้โปรแกรมต้องเพจบ่อยเกินไปเนื่องจากอ็อปชัน -pg ใช้พื้นที่บัฟเฟอร์หน่วยความจำจริงที่ยืดได้ เท่ากับครึ่งหนึ่งของขนาดของข้อความ ของโปรแกรมของคุณ การเพจที่มากเกินไปจะไม่มีผลกับข้อมูล ที่สรว้าขึ้นโดยการทำโปรไฟล์ เนื่องจากโปรแกรมที่ทำโปรไฟล์จะไม่สร้าง tick เมื่อรอ I/O แต่จะสร้างเมื่อใช้ตัวประมวลผลเท่านั้น หากเกิดการหน่วงเวลา ที่เกิดจากการเพจที่มากเกินไปเกินค่าที่ยอมรับได้ แนะนำให้ใช้ คำสั่ง tprof

## แฟล็ก

ไอเท็ม	คำอธิบาย
-b	ไม่แสดงการพิมพ์รายละเอียดของแต่ละฟิลด์ในโปรไฟล์
-c <i>Filename</i>	สร้างไฟล์ที่มีข้อมูลที่จำเป็น สำหรับการประมวลผลแบบของข้อมูลการทำโปรไฟล์ อย่าใช้แฟล็ก -c รวม กับแฟล็กอื่นๆ
-E <i>Name</i>	ไม่แสดงการพิมพ์ของรายการโปรไฟล์กราฟสำหรับรูทีน <i>Name</i> และตัวที่สืบทอดมา คล้ายกับแฟล็ก -e แต่ไม่รวมเวลาที่ใช้โดยรูทีน <i>Name</i> และตัวที่สืบทอดมาจากการคำนวณเวลาทั้งหมดหรือเป็นเปอร์เซ็นต์ (-E <i>MonitorCount</i> -E <i>MonitorCleanup</i> is the default.)
-e <i>Name</i>	ไม่แสดงการพิมพ์ของรายการกราฟโปรไฟล์สำหรับรูทีน <i>Name</i> และตัวที่สืบทอดทั้งหมด (ยกเว้นว่ามีต้นกำเนิด อื่นที่ถูกแสดง) โดยสามารถกำหนดมากกว่าหนึ่งแฟล็ก -e โดยสามารถกำหนดได้เพียงหนึ่งรูทีนเท่านั้นกับและแฟล็ก -e
-F <i>Name</i>	พิมพ์รายการกราฟโปรไฟล์ของรูทีน <i>Name</i> และตัวที่สืบทอดคล้ายกับแฟล็ก -f แต่ใช้จำนวนครั้งของรูทีนที่ถูกพิมพ์เท่านั้นในการคำนวณเวลาทั้งหมดและ ที่เป็นเปอร์เซ็นต์ โดยสามารถกำหนดมากกว่าหนึ่งแฟล็ก -F โดย สามารถกำหนดได้เพียงหนึ่งรูทีนเท่านั้นกับและแฟล็ก -F แฟล็ก -F จะแทนที่แฟล็ก -E
-f <i>Name</i>	พิมพ์รายการกราฟโปรไฟล์ของรูทีนที่ระบุ <i>Name</i> และตัวที่สืบทอด โดยสามารถกำหนดมากกว่าหนึ่งแฟล็ก -f โดยสามารถกำหนดได้เพียงหนึ่งรูทีนเท่านั้นกับและแฟล็ก -f
-g <i>Filename</i>	เขียนข้อมูลกราฟการเรียกใช้ไปยังเอาต์พุต <i>filename</i> ที่ระบุ รวมทั้งไม่แสดงข้อมูลโปรไฟล์ยูเวน จะใช้แฟล็ก -p
-i <i>Filename</i>	เขียนตารางดัชนีรูทีนไปยังเอาต์พุต <i>filename</i> ที่ระบุ หากไม่ใช้แฟล็กนี้ ตารางดัชนีจะไปยัง ทายของเอาต์พุตมาตรฐาน หรือที่ด้านล่างของชื่อไฟล์ ที่ระบุด้วยแฟล็ก -p และ -g
-L <i>PathName</i>	ใช้ชื่อพาทสำหรับหารหาอ็อบเจกต์แบบแบ่งใช้
-p <i>Filename</i>	เขียนข้อมูลโปรไฟล์แบบไม่มีลำดับชั้นไปยังชื่อไฟล์ เอาต์พุตที่ระบุ รวมทั้งไม่แสดงข้อมูลกราฟการเรียกใช้ยูเวนจะใช้แฟล็ก -g
-s	สร้างโปรไฟล์ไฟล์ <i>gmon.sum</i> ซึ่งแสดงผลรวมของข้อมูลโปรไฟล์ในไฟล์โปรไฟล์ที่ระบุทั้งหมด ไฟล์โปรไฟล์สรุปนี้อาจถูกกำหนดเพื่อเรียกใช้งาน คำสั่ง gprof ในภายหลัง (โดยใช้แฟล็ก -s) เพื่อสะสม ข้อมูลโปรไฟล์จากหลายการรันของไฟล์ <i>a.out</i>
-x <i>Filename</i>	ดึงข้อมูลจาก <i>Filename</i> (ไฟล์ที่สร้างด้วย อ็อปชัน -c) เพื่อสร้างรายงานโปรไฟล์ หากไม่ระบุ <i>Filename</i> คำสั่ง gprof จะค้นหาไฟล์ <i>gprof.remote</i> ดีฟอลต์
-z	แสดงรูทีนที่มีการใช้งานเป็นศูนย์ (ตั้งที่ระบุโดยกจำนวนนับการเรียกใช้ และเวลาที่สะสม)

## ตัวอย่าง

1. เมื่อต้องการขอรับเอาต์พุตที่ทำโปรไฟล์ ให้ป้อนคำสั่งต่อไปนี้:

```
gprof
```

2. เมื่อต้องการเอาต์พุตการทำโปรไฟล์จากคำสั่งที่รันก่อนหน้านี้และอาจถูกย้ายไปแล้ว ให้ป้อนคำสั่งต่อไปนี้:

```
gprof -L/home/score/lib runfile runfile.gmon
```

ตัวอย่างนี้ ใช้ไฟล์ *runfile.gmon* สำหรับข้อมูลตัวอย่างและไฟล์ *runfile* สำหรับสัญลักษณ์โลคัล และตรวจสอบไฟล์ */u/score/lib* สำหรับ อ็อบเจกต์ที่โหลดได้

3. ในการโปรไฟล์โปรแกรมตัวอย่าง *dhry.c*:

a. คอมไพล์แอสเพลิกซ์โปรแกรมอีกครั้งด้วยคำสั่ง *cc -pg* ดังนี้:

```
cc -pg dhry.c -o dhry # Re-compile to produce gprof output.
```

- b. รันโปรแกรมที่คอมไพล์อีกครั้ง ไฟล์ที่ชื่อ gmon.out ถูกสร้างในไดเรกทอรีการทำงานปัจจุบัน (ไม่ใช่ไดเรกทอรีที่มีไฟล์ที่รันได้ ของโปรแกรม)

```
dhry # Execute program to generate ./gmon.out file.
```

- c. รันคำสั่ง **gprof** ในไดเรกทอรีที่มีไฟล์ gmon.out เพื่อสร้างกราฟการเรียกใช้และรายงานโปรไฟล์แบบไม่มีลำดับชั้น

```
gprof >gprof.out # Name the report whatever you like  
vi gprof.out # Read flat profile first.
```

- d. เมื่อต้องการสร้างรายละเอียดการทำโปรไฟล์ระดับเธรด ให้เอ็กซ์พอร์ตตัวแปรสถานะแวดล้อม GPROF ดังต่อไปนี้ และรันแอสพลีเคชันให้ป้อนคำสั่งต่อไปนี้:

```
export GPROF=profile:thread  
dhry # Execute program to generate ./gmon.out file which has thread level granularity
```

- e. เมื่อต้องการสร้างไฟล์ **gmon.out** ต่อกระบวนการที่มีส่วนนำหน้าของ mygmon ให้ป้อนคำสั่งต่อไปนี้:

```
export GPROF=file:multi,filename:mygom  
dhry # Execute program to generate ./gmon-dhry-2468.out
```

- f. เมื่อต้องการสร้างไฟล์ **gmon.out** ต่อกระบวนการ ที่มีค่าสเกล 10 ที่ชื่อไฟล์นำหน้าด้วย tgmon ให้ป้อนคำสั่งต่อไปนี้:

```
export GPROF=profile:thread,file:multithread,scale:10,filename:tgmon  
dhry # Execute program to generate ./tgmon-dhry-2468-Pthread215.out
```

- g. เมื่อต้องการดูรายงานโปรไฟล์แบบไม่มีลำดับชั้นจาก gmon-dhry-2468.out ให้ป้อนคำสั่งต่อไปนี้:

```
gprof -p fprofile.out ./dhry ./gmon-dhry-2468.out
```

- h. เมื่อต้องการดูรายงานโปรไฟล์กราฟการเรียกใช้จาก gmon-dhry-2468.out ให้ป้อนคำสั่งต่อไปนี้:

```
gprof -g callgraph.out ./dhry ./gmon-dhry-2468.out
```

#### 4. ในการใช้คุณลักษณะการประมวลผลรีโมตของคำสั่ง **gprof**:

- a. คอมไพล์แอสพลีเคชันโปรแกรมอีกครั้งด้วยคำสั่ง **cc -pg**:

```
cc -pg thread.c -o thread -lpthread
```

- b. เปิดใช้งานกลุ่มการโปรไฟล์ระดับเธรด และใช้ชื่ออื่นสำหรับ **gmon.out**:

```
export GPROF=profile:thread,filename:mygom
```

- c. รันโปรแกรมที่คอมไพล์อีกครั้ง ไฟล์ที่ชื่อ **mygmon.out** ถูกสร้างในไดเรกทอรีการทำงานปัจจุบัน (ไม่ใช่ไดเรกทอรีที่มีไฟล์ที่รันได้ ของโปรแกรม)

```
thread # Execute program to generate mygmon.out file.
```

- d. ใช้แฟล็ก **-c** เพื่อสร้างไฟล์ **my.remote** ซึ่งสามารถนำไปไว้ที่เครื่องรีโมตเพื่อทำการประมวลผล:

```
gprof -c my.remote thread mygom.out
```

- e. บนเครื่องรีโมต ใช้แฟล็ก **-x** เพื่อแยก ข้อมูลจากไฟล์ **my.remote**:

```
gprof -x my.remote
```

ตลอดทั้งรายละเอียดของคำสั่ง **gprof** นี้ ตัวอย่างส่วนใหญ่ใช้ภาษา C **dhry.c** อย่างไรก็ตาม การอภิปรายและตัวอย่างจะใช้กับโมดูล FORTRAN หรือ COBOL ได้เหมือนกันโดยการแทนที่ชื่อคอมไพเลอร์ที่เหมาะสมแทน C คอมไพเลอร์ **cc** และคำว่า *subroutine* ด้วยคำว่า *function* ตัวอย่างเช่น คำสั่งต่อไปนี้จะแสดงวิธีทำโปรไฟล์โปรแกรม FORTRAN ชื่อ **matrix.f**:

```

xlf -pg matrix.f -o matrix # FORTRAN compile of matrix.f program
matrix                    # Execute with gprof profiling,
                           # generating gmon.out file
gprof > matrix.out       # Generate profile reports in
                           # matrix.out from gmon.out
vi matrix.out             # Read flat profile first.

```

## ไฟล์

ไอเท็ม	คำอธิบาย
a.out	รายการชื่อและช่องว่างข้อความ
gmon.out	กราฟและโปรไฟล์การเรียกใช้แบบไดนามิก
gmon.sum	กราฟและโปรไฟล์การเรียกใช้แบบไดนามิกฉบับย่อ
gprof.remote	ไฟล์สำหรับการทำโปรไฟล์แบบรีโมต
/usr/ucb/gprof	มีคำสั่ง gprof
/usr/ccs/bin/gprof	มีคำสั่ง gprof

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง prof

คำสั่ง exit

การมอนิเตอร์และปรับคำสั่งและรูทีนย่อย

ภาพรวมรูทีนย่อย

## คำสั่ง grap

### วัตถุประสงค์

เรียงพิมพ์กราฟที่จะประมวลผลโดยคำสั่ง pic

### ไวยากรณ์

```
grap [-l] [-T Name] [-] [File ...]
```

### คำอธิบาย

คำสั่ง **grap** ประมวลผลอินพุตไฟล์ภาษา **grap** และสร้างอินพุตไปยังคำสั่ง **pic** ภาษา **grap** คือภาษาสำหรับการเรียงพิมพ์กราฟ บรรทัดคำสั่งปกติคือ:

```
grap File | pic | troff | Typesetter
```

กราฟถูกล้อมรอบด้วยการร้องขอคำสั่ง **.G1** และ **.G2 troff** ข้อมูลที่อยู่ภายในการร้องขอเหล่านี้ จะถูกกำหนดมาตามตราส่วน และพล็อตจุด พร้อมมีเครื่องหมายเลือกมาให้โดยอัตโนมัติ คำสั่งมี เพื่อแก้ไขเฟรม เพิ่มเลเบล แทนที่เครื่องหมายการเลือกค่า ดีฟอลต์ เปลี่ยนลักษณะการพล็อต กำหนดช่วงพิกัด และการแปลงสภาพ รวมถึงการรวมข้อมูลจากไฟล์ นอกจากนั้น คำสั่ง **grap** ยังจัดให้มีการวนซ้ำ มีเงื่อนไข และการประมวลผลมาโครเหมือนกับคำสั่ง **pic**

ไฟล์ภาษา Grap มีโปรแกรม grap โปรแกรม grap ถูกเขียนในรูป:

.G1  
grap Statement  
grap Statement  
grap Statement  
.G2

## พารามิเตอร์

ไอเท็ม คำอธิบาย  
ไฟล์ ระบุไฟล์ภาษา grap (โปรแกรม grap) ที่จะประมวลผลโดย คำสั่ง grap สำหรับเป็นอินพุตไปยังคำสั่ง pic

## สรุปคำสั่ง grap

ต่อไปนี้เป็นสรุปคำสั่ง grap ที่คุณสามารถใช้เพื่อสร้างโปรแกรม grap:

ไอเท็ม คำอธิบาย  
frame กำหนดกรอบที่ใช้ล้อมรอบกราฟ ไวยากรณ์คือ:  
frame [ht Expression] [wid Expression] [[Side] Lineคำอธิบาย]

แอตทริบิวต์ถูกกำหนดดังนี้:

- *Side*: top, bot, left, right
- *Lineคำอธิบาย*: solid, invis, dotted [Expression], dashed [Expression]

ส่วนสูงดีฟอลต์คือ 2 นิ้ว ความกว้างดีฟอลต์คือ 3 นิ้ว ด้านข้างดีฟอลต์เป็นสี่ทึบ หากไม่ระบุ side จะใช้ *linedesc* กับกรอบทั้งหมด

ไอเท็ม คำอธิบาย  
label วางเลเบลบนตำแหน่งระบุของกราฟ ไวยากรณ์คือ:  
label Side StringList ... Shift

แอตทริบิวต์ถูกกำหนดดังนี้:

- *Shift*: left, right, up, or down *expression*
- *StringList*: str ... rjust, ljust, above, below [size (+)Expression] ...
- *String*: "..."

ไอเท็ม คำอธิบาย  
coord กำหนดระบบการแทนที่รูปแบบไวยากรณ์คือ:  
coord [Name] [x Expression,Expression] [y Expression,Expression] [[log x] [log y] [log log]]  
ticks วางเครื่องหมายเลือกบนด้านหนึ่งของกรอบ ไวยากรณ์คือ:  
ticks side [[in] [out] [Expression]] [Shift] [TickLocations]

แอตทริบิวต์ถูกกำหนดดังนี้:

- *Shift*: left, right, up, down Expression
- *TickLocations*: at [Name] Expression [String], Expression [String], ... from [Name] Expression to Expression [by [Operation] Expression] String

หากไม่ระบุเครื่องหมายเลือก จะจัดให้มีโดยอัตโนมัติ ticks off ระบุการเลือกอัตโนมัติ

**ไอเท็ม** คำอธิบาย  
**grid** สร้างเส้นกริดบน (คือ ตั้งฉากกับ) ด้านที่ระบุชื่อ รูปแบบไวยากรณ์คือ:  
grid Side [Lineคำอธิบาย] [Shift] [TickLocations]

กริดถูกเลเบลด้วยวิธีเดียวกันกับเครื่องหมายเลือก

**ไอเท็ม** คำอธิบาย  
**plot** วางข้อความที่จุด รูปแบบไวยากรณ์คือ:  
StartList at Point plot Expression [Start] at Point

แอ็ททริบิวต์ถูกกำหนดดังนี้:

- *StringList*: str ... rjust, ljust, above, below [size+)Expression] ...
- *Point*: [Name] Expression Expression

**ไอเท็ม** คำอธิบาย  
**line** วาดเส้นหรือลูกศรจากจุดหนึ่งไปอีกจุดหนึ่ง ไวยากรณ์คือ:  
{line | arrow} from Point to Point [Lineคำอธิบาย]

แอ็ททริบิวต์ linedesc ถูกกำหนดดังนี้:

- *Point*: [Name] Expression Expression
- *Lineคำอธิบาย*: solid, invis, dotted [Expression], dashed Expression]

**ไอเท็ม** คำอธิบาย  
**circle** วาดวงกลม รูปแบบไวยากรณ์คือ:  
circle at Point [radius Expression]

รัศมีหน่วยเป็นนิ้ว ขนาดตีฟอลต์คือขนาดเล็ก

**ไอเท็ม** คำอธิบาย  
**draw** กำหนดลำดับของเส้น รูปแบบไวยากรณ์คือ:  
draw [Name] at Point [Lineคำอธิบาย]

**next** ดำเนินการลำดับต่อ รูปแบบไวยากรณ์คือ:  
next [Name] at Point [Lineคำอธิบาย]

**new** เริ่มต้นลำดับใหม่ รูปแบบไวยากรณ์คือ:  
new [Name] at Point [Lineคำอธิบาย]

**numberlist** สร้างเส้นจากชุดของตัวเลขที่กำหนด ตัวเลขจะถือเป็นจุด x, y1, y2 ตามลำดับ และพล็อตที่ตำแหน่ง x เดียว รูปแบบไวยากรณ์คือ:  
number x, y1, y2 ...

**for** สร้างการวนซ้ำรูปแบบไวยากรณ์คือ:  
for Variable {from | =} Expression to Expression \  
[by [arithmetic or multiplicative operator] Expression] do X Anything X

X คืออักขระเดี่ยวใดๆ ที่ไม่ปรากฏในสตริง หาก X เป็นวงเล็บปีกกาซ้าย {, สตริงอาจมีการสร้างสมดุลภายในคือตามด้วยวงเล็บปีกกาขวา } ข้อความ Anything ถูกทำซ้ำขณะที่ Variable รับค่าจาก Expression แรกไปยัง Expression ที่สอง

**if** สร้างการประเมินผลแบบมีเงื่อนไขรูปแบบไวยากรณ์คือ:  
if Expression then X Anything X [else X Anything X]

**define** จัดให้มีตัวประมวลผลมาโครเหมือนกับ Priority Interrupt Controller (PIC) ทำรูปแบบไวยากรณ์คือ:  
define MacroName X Anything X

ไอเท็ม	คำอธิบาย
copy	คัดลอกไฟล์ โดยรวมเนื้อหาของไฟล์ปัจจุบัน รูปแบบไวยากรณ์คือ: copy Filename
copy-thru	คัดลอกไฟล์ทางแมโคร copy Filename thru MacroName
	แต่ละหมายเลขหรือสตริงที่อยู่ในเครื่องหมายคำพูดจะ ถือเป็นหนึ่งอาร์กิวเมนต์ การคัดลอกยังคงทำต่อไปจนถึงจุดสิ้นสุดไฟล์ หรือ .G2 ถัดไปอนุประโยคทางเลือก until String ทำให้การคัดลอกหยุดลงเมื่อมีบรรทัดที่ฟิลด์แรกเป็น String
	คำสั่งต่อไปจะคัดลอก บรรทัดต่อๆ มาผ่านแมโคร: copy thru MacroName
	ในทุกกรณี คุณสามารถระบุแมโครได้โดยใช้อินไลน์แทน โดยใช้ชื่อ: copy thru x MacroBody x
sh	ส่งข้อความผ่านไปยังเชลล์ UNIX ไวยากรณ์คือ: sh x Anything x
pic	ตัวแปร Anything ถูกสแกนสำหรับแมโคร แมโคร pid เป็นแบบในตัว คือเป็นสตริงที่ประกอบด้วยหมายเลข identification กระบวนการ คุณสามารถใช้เพื่อสร้างชื่อไฟล์เฉพาะได้ ส่งข้อความผ่านไปยัง pic โดย pic ถูกลบออก ตัวแปรและแมโครจะไม่ถูกประเมินค่า บรรทัดที่ขึ้นต้นด้วย จุด (นั่นคือไม่ใช่ตัวเลข) จะถูกส่งผ่านในแบบตรงตามตัวอักษรนั้น ภายใน สมมติฐานว่าเป็นคำสั่ง troff
graph	กำหนดกราฟใหม่ชื่อ Picname และรีเซ็ต ระบบพิกัดทั้งหมด รูปแบบไวยากรณ์คือ: graph Picname [pic-text]
	หากใช้คำสั่ง graph ใน โปรแกรม grap โปรแกรมคำสั่งหลัง .G1 ต้อง เป็นคำสั่ง graph คุณสามารถใช้ pic-text เพื่อกำหนดตำแหน่งกราฟนี้โดยสัมพันธ์กับกราฟก่อนหน้าโดยการอ้างอิง Frames ของกราฟตั้งในตัวอย่างต่อไปนี้ graph First ... graph Second with .Frames.w at First.Frame.e + [0.1,0]
print	แมโครและนิพจน์ใน pic-text จะไม่ถูกประเมินค่า Picnames ต้องขึ้นต้นด้วยตัวอักษรตัวพิมพ์ใหญ่ตาม ไวยากรณ์ pic เขียนบน stderr ขณะที่ grap กำลังประมวลผล อินพุต คำสั่งนี้สามารถเป็นประโยชน์ในการดีบั๊ก รูปแบบไวยากรณ์คือ: print [Expression   String]

## ระเบียบการใช้ภาษา grap

ใช้ระเบียบต่อไปนี้:

- # (เครื่องหมาย ปอนด์) จะแสดงว่าเป็นข้อคิดเห็น ความคิดเห็นสิ้นสุดโดยอัตโนมัติเมื่อสิ้นสุด บรรทัด
- ประโยคคำสั่งที่ยาวต่อเนื่องมากกว่าหนึ่ง บรรทัดต้องถูกนำหน้าด้วย \ (อักขระแบ็กสแลช) ที่ตำแหน่งเริ่มต้นของบรรทัดใหม่ แต่ละบรรทัด
- หลายๆ คำสั่งที่ปรากฏบนบรรทัดเดียวกัน ต้องคั่นด้วยเซมิโคลอน
- ภาษา grap จะข้ามบรรทัดว่าง
- สตริงที่กำหนดไว้แล้วได้แก่ bullet, plus, box, star, dot, times, htick, vtick, square และ delta
- ฟังก์ชันในตัวที่มีอยู่ใน grap ได้แก่ log (ฐาน 10), exp (ฐาน 10), int, sin, cos, atan2, sqrt, min, max และ rand

## แฟล็ก

ไอเท็ม คำอธิบาย  
 -I หยุดทำงานคำสั่ง `grap` ในการค้นหาไลบรารี `/usr/lib/dwb/grap.defines` เพื่อหา macro definitions  
 -T *Name* ระบุค่าของตัวแปร *Name* เป็นอุปกรณ์เอาต์พุตคำสั่ง `grap` ค่าดีฟอลต์คือ `-Tibm3816`  
 -- (เส้นประคู่) บ่งชี้ว่าสิ้นสุดแฟล็ก

## File

ไอเท็ม	คำอธิบาย
<code>/usr/lib/dwb/grap.defines</code>	มีนียมสำหรับอักขระการพล็อตมาตรฐาน

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `pic`

## คำสั่ง `greek`

### วัตถุประสงค์

แปลงเอาต์พุตภาษาอังกฤษจากเวิร์กสเตชัน Teletype Model 37 ไปยังเอาต์พุตสำหรับเวิร์กสเตชันอื่น

### ไวยากรณ์

`greek [ -T Name ]`

### คำอธิบาย

คำสั่ง `greek` จะแปลความหมาย ชุดอักขระ Teletype Model 37 อีกครั้ง รวมถึงการเคลื่อนไหวแบบย้อนกลับ แบบครึ่งบรรทัด สำหรับแสดงบนเวิร์กสเตชันอื่น โดยเลียนแบบอักขระพิเศษ หากเป็นไปได้โดยการขีดทับ คำสั่ง `greek` อ่านอินพุตมาตรฐาน และเขียนไปยังเอาต์พุตมาตรฐาน

### แฟล็ก

ไอเท็ม	คำอธิบาย
-T <i>Name</i>	ทอเวิร์กสเตชันที่ระบุ หากคุณข้ามแฟล็ก -T คำสั่ง <code>greek</code> จะพยายามใช้ เวิร์กสเตชันที่ระบุในตัวแปรสถานะแวดล้อม <code>\$TERM</code> ค่าของตัวแปร <i>Name</i> สามารถเป็นค่าหนึ่งค่าใด ต่อไปนี้:
300	DASI 300
300-12	DASI 300 ที่มี 12-pitch
300s	DASI 300s
300s-12	DASI 300s ที่มี 12-pitch
450	DASI 450
450-12	DASI 450 ที่มี 12-pitch
2621	Hewlett-Packard 2621, 2640 และ 2645
2640	Hewlett-Packard 2621, 2640 และ 2645
2645	Hewlett-Packard 2621, 2640 และ 2645
4014	Tektronix 4014
hp	Hewlett-Packard 2621, 2640 และ 2645
tek	Tektronix 4014

## ตัวแปรสถานะแวดล้อม

ไอเท็ม คำอธิบาย  
\$TERM ระบุชื่อเวิร์กสเตชัน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง eqn” ในหน้า 419

“คำสั่ง hp” ในหน้า 804

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mm

คำสั่ง troff

---

## คำสั่ง grep

### วัตถุประสงค์

ค้นหารูปแบบในไฟล์

### ไวยากรณ์

```
grep [-E|-F][-i][-h][-H][-L][-r|-R][-s][-u][-v][-w][-x][-y][[[-b][-n]]|[-c|-l|-q]][-p[  
Separator]]{[-e PatternList ...][-f PatternFile ...]|PatternList ...}[File ...]
```

### คำอธิบาย

คำสั่ง `grep` ค้นหารูปแบบ ที่ระบุโดยพารามิเตอร์ *Pattern* และเขียน แต่ละบรรทัดที่ตรงไปยังเอาต์พุตมาตรฐาน รูปแบบคือ นิพจน์ปกติ ที่ถูกจำกัดให้อยู่ในลักษณะของคำสั่ง `ed` หรือ `egrep` คำสั่ง `grep` ใช้อัลกอริทึมแบบไม่มีการกำหนดความกระชับ

คำสั่ง `grep` แสดงชื่อของ ไฟล์ที่มีบรรทัดที่ตรงกันหากคุณระบุมากกว่าหนึ่งชื่อ ในพารามิเตอร์ *File* อักขระที่มีความหมาย พิเศษสำหรับเชลล์ (`$`, `*`, `[`, `]`, `^`, `(`, `)`, `\`) ต้องอยู่ในเครื่องหมายคำพูด เมื่อปรากฏในพารามิเตอร์ *Pattern* เมื่อพารามิเตอร์ *Pattern* ไม่ใช่สตริงปกติโดยปกติคุณต้องปิดรูปแบบทั้งหมดในเครื่องหมายคำพูด เดี่ยว ในนิพจน์ เช่น `[a-z]` - (เครื่องหมาย ลบ) `cml` ระบุช่วง ตามลำดับ การเรียงปัจจุบัน ลำดับการเรียงอาจกำหนดคลาสที่เหมือนกันเพื่อใช้ในขอบเขต อักขระ หากไม่ ระบุไฟล์ `grep` จะถือว่า ใช้อินพุตมาตรฐาน

### Notes:

1. อย่ารันคำสั่ง `grep` บนไฟล์พิเศษเนื่องจากคำสั่ง จะทำให้เกิดผลลัพธ์ที่ไม่คาดคิด บรรทัดอินพุตไม่ควรมี อักขระ `NULL`
2. อินพุตไฟล์ควรจบด้วยอักขระขึ้นบรรทัดใหม่
3. อักขระบรรทัดใหม่จะไม่ถูก จับคู่โดยนิพจน์ปกติ
4. แม้ว่าสามารถระบุ บางแฟล็กได้แบบพร้อมกัน บางแฟล็กจะแทนที่แฟล็กอื่นๆ ตัวอย่างเช่น อีอพชัน `-l` มาก่อนแฟล็กอื่น ทั้งหมด และหากคุณระบุทั้งแฟล็ก `-E` และ `-F` แฟล็กที่ระบุสุดท้ายจะเป็นแฟล็กที่มีระดับความสำคัญกว่า

### แฟล็ก

ไอเท็ม	
-b	คำอธิบาย นำหน้าแต่ละบรรทัดด้วยหมายเลขบล็อกที่พบ ใช้แฟล็กนี้เพื่อช่วยค้นหาหมายเลขบล็อกของคุณดิสก์ตามบริบท แฟล็ก -b ไม่สามารถเข้าร่วมกับอินพุตจาก stdin หรือ ไฟล์
-c	แสดงเฉพาะจำนวนบรรทัดที่ตรงกันเท่านั้น
-E	ถือว่าแต่ละรูปแบบระบุเป็น Extended Regular Expression (ERE) คำ NULL สำหรับ ERE ตรงกับทุกบรรทัด หมายเหตุ: คำสั่ง grep ที่มีแฟล็ก -E จะเหมือนกับคำสั่ง egrep ยกเว้นว่าข้อความ แสดงข้อผิดพลาดและการใช้งานแตกต่างกัน และแฟล็ก -s ทำงาน ต่างกัน
-e PatternList	ระบุรูปแบบการค้นหาอย่างน้อยหนึ่งรูปแบบ คำสั่งนี้ทำงานคล้ายกับ รูปแบบอย่างง่ายที่เป็นประโยชน์เมื่อรูปแบบขั้นต้นด้วย - (เครื่องหมายลบ) รูปแบบ ควรถูกค้นด้วยอักขระบรรทัด รูปแบบ NULL สามารถถูก ระบุโดยอักขระสองบรรทัดติดกัน หรือเครื่องหมายคำพูด ตามด้วยอักขระบรรทัดใหม่ (^n) แต่ละรูปแบบจะถูกถือเหมือน Basic Regular Expression (BRE) ยกเว้นแฟล็ก -E หรือ -F ถูกระบุเช่นกัน แฟล็ก -e และ -f สามารถยอมรับโดย grep รูปแบบที่ระบุทั้งหมดถูกใช้ เมื่อจับคู่ บรรทัด แต่ลำดับของการประเมินผลไม่ถูกระบุ
-F	ถือรูปแบบที่ระบุแต่ละรูปแบบเป็นสตริงแทนนิพจน์ ปกติ สตริง NULL ตรงกับทุกบรรทัด หมายเหตุ: คำสั่ง grep ที่มีแฟล็ก -F จะเหมือนกับคำสั่ง fgrep ยกเว้นว่าข้อความ แสดงข้อผิดพลาดและการใช้งานแตกต่างกัน และแฟล็ก -s ทำงาน ต่างกัน
-f PatternFile	ระบุไฟล์ที่มีรูปแบบการค้นหา แต่ละรูปแบบควร ค้นด้วยอักขระบรรทัดใหม่ และบรรทัดจะถูกมองว่าเป็นรูปแบบ NULL แต่ละรูปแบบจะถูกถือเหมือน Basic Regular Expression (BRE) ยกเว้นแฟล็ก -E หรือ -F ถูกระบุเช่นกัน
-h	ป้องกันชื่อของไฟล์ที่มีบรรทัดที่ตรงกัน มิให้ถูกผนวกต่อท้ายที่บรรทัดนั้น ไม่แสดงชื่อไฟล์เมื่อระบุหลาย ไฟล์
-H	หากระบุชื่อพจน์ -r หรือ -R และการลิงก์สัญลักษณ์อ้างอิงไฟล์ที่มีชนิดไดเรกทอรี ที่ถูกระบุ บรรทัดคำสั่ง grep จะ ค้นหาไฟล์ของไดเรกทอรีที่อ้างอิงโดยลิงก์สัญลักษณ์ และไฟล์ทั้งหมดในลำดับชั้นไฟล์ด้านล่าง
-i	ไม่สนใจขนาดตัวพิมพ์ (ตัวพิมพ์ใหญ่ หรือตัวพิมพ์เล็ก) ของตัวอักษรเมื่อทำ การเปรียบเทียบ

ไอเท็ม	
-I	คำอธิบาย แสดงรายการเฉพาะชื่อของไฟล์ (ครั้งเดียว) ซึ่งมีบรรทัดที่ พบ แต่ละชื่อไฟล์ถูกแยกด้วยอักขระขึ้นบรรทัดใหม่ ถ้าอินพุตมาตรฐาน ถูกค้นหา ชื่อพาธ (StandardInput) ถูก ส่งคืน แฟล็ก -I ที่มีแฟล็ก -c และ -n รวมกันใดๆ จะทำหน้าที่ เหมือนแฟล็ก -I เท่านั้น
-L	หากระบุชื่อพจน์ -r หรือ -R และลิงก์สัญลักษณ์อ้างอิงไฟล์ที่เป็นไดเรกทอรี ถูกระบุ บรรทัดคำสั่งหรือพบระหว่างการแหว่ผ่าน ลำดับชั้นไฟล์ grep จะค้นหาไฟล์ ของไดเรกทอรีที่ถูกระบุโดยลิงก์สัญลักษณ์ และไฟล์ทั้งหมด ในลำดับชั้นไฟล์ด้านล่าง หากระบุทั้ง -H และ -L ชื่อพจน์สุดท้ายที่ถูกระบุ บรรทัดคำสั่งจะมีผล
-n	นำหน้าแต่ละบรรทัดด้วยหมายเลขบรรทัดสัมพันธ์ในไฟล์ แต่ละไฟล์เริ่มต้นด้วยบรรทัด 1 และตัวนับบรรทัดจะถูกรีเซ็ตค่าเมื่อประมวลผล แต่ละไฟล์
-p[Separator]	แสดงย่อหน้าทั้งหมดที่มีบรรทัดที่ตรงกัน ย่อหน้า ถูกค้นด้วยตัวค้นย่อหน้า ตามที่ระบุโดยพารามิเตอร์ Separator ซึ่งรูปแบบอยู่ในพอร์ม เดียวกับรูปแบบการค้นหา บรรทัดที่มีตัวค้นย่อหน้า ถูกใช้เฉพาะเป็นตัวค้น; จะไม่มีการรวมไว้ในเอาต์พุต ตัวค้นย่อหน้าดีฟอลต์คือบรรทัดว่าง
-q	ยับยั้งการเขียนทั้งหมดทั้งหมดไปที่เอาต์พุตมาตรฐาน ไม่ว่าบรรทัด จะตรงกันหรือไม่ จบ ด้วยสถานะศูนย์ ถ้าบรรทัดอินพุตถูกเลือก แฟล็ก -q ที่มีการใช้แฟล็ก -c, -I และ -n ใดๆ รวมกัน จะทำหน้าที่เหมือน -q เท่านั้น
-r	ค้นหาไดเรกทอรีแบบเรียกซ้ำ โดยค่าดีฟอลต์ ลิงก์ไปยังไดเรกทอรีจะถูกตามไป
-R	ค้นหาไดเรกทอรีแบบเรียกซ้ำ โดยค่าดีฟอลต์ ลิงก์ไปยังไดเรกทอรีจะไม่ถูกตาม
-s	ไม่แสดงข้อความแสดงความผิดพลาดที่ปกติเขียนด้วยไฟล์ที่ไม่มีอยู่ หรือไฟล์ที่ไม่สามารถอ่านได้ ข้อความแสดงความผิดพลาดอื่นๆ จะถูกระงับ
-u	ทำให้เอาต์พุตถูกยกเลิกบูฟเฟอร์
-v	แสดงบรรทัดทั้งหมดที่ไม่ตรงกับรูปแบบที่ระบุ
-w	ทำการคนหาคำ
-x	แสดงบรรทัดที่ตรงกับรูปแบบที่ระบุอย่างชัดเจน โดยไม่มีอักขระเพิ่มเติม
-y	ละเว้นตัวพิมพ์ของตัวอักษร เมื่อทำการเปรียบเทียบ
PatternList ไฟล์	ระบุอย่างน้อยหนึ่งรูปแบบที่จะใช้ระหว่างการค้นหา รูปแบบถูกถือเสมือนว่าถูกระบุโดยใช้แฟล็ก -e ระบุชื่อของไฟล์ที่จะถูกคนหารูปแบบ หาก ไม่กำหนดตัวแปร File จะใช้อินพุตมาตรฐาน แทน

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย

0 พบข้อมูลตรงกัน

1 ไม่พบข้อมูลตรงกัน

>1 พบข้อผิดพลาดทางไวยากรณ์หรือไม่เข้าถึงไม่ได้ (แม้ว่าจะพบข้อมูลที่ตรงกัน)

## ตัวอย่าง

1. ในการใช้รูปแบบที่มีอักขระ การจับคู่รูปแบบ \*, ^, ?, [, ], \(\, \), \{, ให้ผ่าน \} ให้ป้อน:

```
grep "[a-zA-Z]" pgm.s
```

คำสั่งนี้ แสดงทุกบรรทัดใน pgm.s ที่มีอักขระ ตัวแรกเป็นตัวอักษร

2. ในการแสดงบรรทัดทั้งหมดที่ไม่ตรงกับ รูปแบบ ให้ป้อน:

```
grep -v "^#" pgm.s
```

คำสั่งนี้แสดง ทุกบรรทัดใน pgm.s ที่มีอักขระตัวแรก ไม่ใช่ # (เครื่องหมายสี่เหลี่ยม)

3. ในการแสดงบรรทัดทั้งหมดในไฟล์ file1 ที่ตรงกับสตริง abc หรือ xyz ให้ป้อน:

```
grep -E "abc|xyz" file1
```

4. ในการค้นหา \$ (เครื่องหมาย ดอลลาร์) ในไฟล์ชื่อ test2 ให้ป้อน:

```
grep \$ test2
```

อักขระ \\$ (แบ็กสแลช) จำเป็นต้องใช้ เพื่อบังคับให้เซลล์ส่ง \\$ (แบ็กสแลช เดียว, เครื่องหมายดอลลาร์) ไปยังคำสั่ง grep อักขระ \ (แบ็กสแลชเดียว) บอกให้ คำสั่ง grep ถู้อักขระต่อไปนี้ (ในตัวอย่างนี้ \$) เป็นอักขระตามตัวอักษร มากกว่าอักขระนิพจน์ ใช้คำสั่ง fgrep เพื่อหลีกเลี่ยงความจำเป็นในการใช้ escape characters เช่นแบ็กสแลช

5. ในการค้นหาแบบไม่ระบุชื่อผ่านทาง /tmp เพื่อค้นหา ไฟล์ที่มีคำว่า IBM โดยไม่มีการเรียกซ้ำทาง ลิงก์ไปยังไดเรกทอรี ให้ป้อน:

```
grep -R IBM /tmp
```

OR

```
grep -r -H IBM /tmp
```

6. ในการค้นหาแบบเรียกซ้ำผ่าน /tmp เพื่อค้นหา ไฟล์ที่มีคำว่า IBM และทำซ้ำผ่านลิงก์ ด้วย ให้พิมพ์:

```
grep -r IBM /tmp
```

OR

```
grep -R -L IBM /tmp
```

## ไฟล์

ไอเท็ม  
/usr/bin/grep

คำอธิบาย  
มีคำสั่ง grep

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง egrep” ในหน้า 357

“คำสั่ง fgrep” ในหน้า 546

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง sed

การเปลี่ยนทิศทางอินพุตและเอาต์พุต

---

## กลุ่ม groups

### วัตถุประสงค์

แสดงความเป็นสมาชิกกลุ่ม

### ไวยากรณ์

```
groups [ User... ]
```

### คำอธิบาย

โดยดีฟอลต์ คำสั่ง `groups` จะเขียนข้อมูลความสัมพันธ์กลุ่มของกระบวนการปัจจุบันไปยัง เอาต์พุตมาตรฐาน หากระบุผู้ใช้หลายรายเป็นพารามิเตอร์คำสั่ง ความเป็นสมาชิกกลุ่มสำหรับผู้ใช้แต่ละรายจะถูกแสดงจากฐานข้อมูล

คำสั่ง `groups` จะยังคงทำการดำเนินการกับผู้ใช้ถัดไปในรายการพารามิเตอร์ต่อไปหลังจากแสดงข้อความเตือน หากไม่พบผู้ใช้ที่กำหนดในฐานข้อมูลค้นหา

### ความปลอดภัย

ค่าควบคุมการเข้าถึง: โปรแกรมนี้ควรติดตั้งเป็น โปรแกรมผู้ใช้ปกติใน Trusted Computing Base

### ตัวอย่าง

ในการแสดงความเป็นสมาชิกกลุ่มของผู้ใช้ที่แสดงรายการใน รายการพารามิเตอร์ให้ป้อน:

```
$ groups sys root lp adm
sys : sys
root : system bin sys security cron audit lp
lp : lp printq
adm : adm
```

### ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/groups	มีคำสั่ง groups
/usr/ucb/groups	ลิงก์สัญลักษณ์ไปยังคำสั่ง groups
/etc/group	กลุ่มไฟล์ ที่มี ID กลุ่ม
/etc/ogroup	เวอร์ชันก่อนหน้าของไฟล์กลุ่ม
/etc/passwd	ไฟล์รหัสผ่าน มี ID ผู้ใช้
/etc/opasswd	เวอร์ชันก่อนหน้าของไฟล์รหัสผ่าน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง getty” ในหน้า 724

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง login

คำสั่ง setgroups

## คำสั่ง grpck

### วัตถุประสงค์

ตรวจสอบความถูกต้องของนิยามกลุ่ม เอกสารนี้อธิบายทั้งคำสั่ง AIX `grpck` และคำสั่ง System V `grpck`

### ไวยากรณ์

```
grpck { -nl|-pl|-tl|-y } { ALL|Group ... }
```

### คำอธิบาย

คำสั่ง `grpck` จะตรวจสอบความถูกต้องของนิยามกลุ่มในไฟล์ฐานข้อมูลผู้ใช้ โดยการตรวจสอบนิยามสำหรับกลุ่มทั้งหมดหรือสำหรับกลุ่มที่ระบุโดยพารามิเตอร์ *Group* หากระบุมากกว่าหนึ่งกลุ่ม ต้องมีช่องว่างระหว่างกลุ่ม

หมายเหตุ: คำสั่งนี้ เขียนข้อความไปยัง `stderr`

คุณต้องเลือกแพลตฟอร์มเพื่อระบุระบบ ต้องพยายามแก้ไขข้อผิดพลาดของแอ็ททริบิวต์ แอ็ททริบิวต์ต่อไปนี้จะถูกตรวจสอบ:

ไอเท็ม	คำอธิบาย
name	ตรวจสอบความเป็นค่าเฉพาะ และการประกอบขึ้นเป็นชื่อกลุ่ม ชื่อกลุ่ม ต้องเป็นสตริงที่ไม่ซ้ำกันที่มีขนาด 8 หรือน้อยกว่า ไม่สามารถขึ้นต้นด้วย + (เครื่องหมายบวก), : (โคลอน), - (เครื่องหมายลบ) หรือ ~ (tilde) ซึ่งไม่สามารถมี : (โคลอน) ในสตริงและไม่สามารถเป็นคีย์เวิร์ด ALL หรือ default ไม่สามารถทำการแก้ไขระบบ
groupID	ตรวจสอบความเป็นค่าเฉพาะ และการประกอบขึ้นเป็น ID กลุ่ม ID ต้องไม่เป็น null และต้องประกอบด้วยตัวเลขฐานสิบเท่านั้น ไม่สามารถทำการแก้ไขระบบ
ผู้ใช้	ตรวจสอบการมีอยู่ของผู้ใช้ที่แสดงรายการในไฟล์ฐานข้อมูลกลุ่ม หากคุณระบุระบบต้องแก้ไขข้อผิดพลาด ระบบจะลบ ผู้ใช้ทั้งหมดที่ไม่พบในไฟล์ฐานข้อมูลผู้ใช้
adms	ตรวจสอบการมีอยู่ของผู้ใช้ที่แสดงรายการเป็นผู้ดูแลระบบกลุ่ม ในไฟล์ฐานข้อมูลกลุ่ม หากคุณระบุ ระบบต้องแก้ไขข้อผิดพลาด ระบบจะลบ ผู้ดูแลระบบทั้งหมดที่ไม่พบในไฟล์ฐานข้อมูลผู้ใช้
admin	ตรวจสอบแอ็ททริบิวต์ admin ที่ถูกต้องสำหรับแต่ละกลุ่มในไฟล์ /etc/security/group ไม่มีทำการแก้ไขระบบ

โดยทั่วไป คำสั่ง `sysck` เรียกใช้คำสั่ง `grpck` เป็นส่วนหนึ่งของการตรวจสอบ ของการติดตั้งที่ระบุไว้ข้างใจ นอกจากนี้ ผู้ใช้ root หรือสมาชิก ของกลุ่มความปลอดภัยสามารถป้อนคำสั่ง

คำสั่ง `grpck` ตรวจสอบเพื่อดูว่าไฟล์การรักษาความปลอดภัยการจัดการฐานข้อมูล (/etc/passwd.nm.idx, /etc/passwd.id.idx, /etc/security/passwd.idx และ /etc/security/lastlog.idx) มีข้อมูลล่าสุด หรือใหม่กว่าไฟล์การรักษาความปลอดภัยระบบที่เกี่ยวข้อง สามารถยอมรับได้ที่ /etc/security/lastlog.idx จะไม่ใหม่กว่า /etc/security/lastlog หากไฟล์การรักษาความปลอดภัย การจัดการฐานข้อมูลล่าสุดมีข้อความเตือนจะปรากฏเพื่อระบุว่า ผู้ใช้ root ต้องรันคำสั่ง `mkpasswd`

## แฟล็ก

ไอเท็ม	คำอธิบาย
-n	รายงานข้อผิดพลาดแต่ไม่แก้ไขข้อผิดพลาดเหล่านั้น
-p	แก้ไขข้อผิดพลาดแต่ไม่รายงานข้อผิดพลาดเหล่านั้น
-t	รายงานข้อผิดพลาดและถามว่าควรแก้ไขหรือไม่
-y	แก้ไขข้อผิดพลาดและรายงานข้อผิดพลาด

## ความปลอดภัย

การควบคุมการเข้าถึง: คำสั่งนี้ต้องกำหนดสิทธิ์การเรียกใช้งาน (x) ให้กับผู้ใช้ root และสมาชิก ของกลุ่มความปลอดภัย คำสั่ง `setuid` สำหรับผู้ใช้ root ต้องมีแอตทริบิวต์ `trusted computing base`

ไฟล์ที่เข้าถึง:

โหมด	File
r	/etc/passwd
r	/etc/security/user
rw	/etc/security/group
rw	/etc/group

การตรวจสอบเหตุการณ์:

เหตุการณ์	ข้อมูล
GROUP_User	ผู้ใช้, กลุ่ม, แอตทริบิวต์   ข้อผิดพลาด, สถานะ
GROUP_Adms	ผู้ใช้, กลุ่ม, แอตทริบิวต์   ข้อผิดพลาด, สถานะ

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

## ตัวอย่าง

- เมื่อต้องการตรวจสอบว่าสมาชิกกลุ่มและผู้ดูแลระบบทั้งหมดมีอยู่ในฐานข้อมูลผู้ใช้ และเพื่อรายงานข้อผิดพลาดทั้งหมด แต่ไม่ทำการแก้ไขให้ป้อนคำสั่งต่อไปนี้:  

```
grpck -n ALL
```
- เมื่อต้องการตรวจสอบว่าสมาชิกกลุ่มและผู้ดูแลระบบมีอยู่ในฐานข้อมูลผู้ใช้ และเพื่อแก้ไขข้อผิดพลาดทั้งหมดแต่ไม่รายงานข้อผิดพลาด ให้ป้อนคำสั่งต่อไปนี้:  

```
grpck -p ALL
```
- เมื่อต้องการตรวจสอบความไม่ซ้ำกันของชื่อกลุ่มและ ID กลุ่มที่กำหนดไว้สำหรับกลุ่ม `install` ให้ป้อนคำสั่งต่อไปนี้:

```
grpck -n install
```

หรือ,

```
grpck -t install
```

หรือ,

```
grpck -y install
```

คำสั่ง **grpck** ไม่แก้ไขชื่อกลุ่มและ ID ดังนั้น แฟล็ก **-n**, **-t** และ **-y** จะรายงานปัญหาเกี่ยวกับชื่อกลุ่มและ ID กลุ่ม แต่ไม่ได้แก้ไข

## ไฟล์

### ไอเท็ม

```
/usr/sbin/grpck  
/etc/passwd  
/etc/security/user  
/etc/group  
/etc/security/group
```

### คำอธิบาย

มีคำสั่ง **grpck**  
มีแอตทริบิวต์ระดับต้นของผู้ใช้  
มีแอตทริบิวต์ส่วนขยายของผู้ใช้  
มีแอตทริบิวต์ระดับต้นของกลุ่ม  
มีแอตทริบิวต์ที่ขยายเพิ่มของกลุ่ม

## คำสั่ง System V grpck

## ไวยากรณ์

```
/usr/sysv/bin/grpck
```

## คำอธิบาย

คำสั่ง **/usr/sysv/bin/grpck** จะตรวจสอบความถูกต้องของนิยามกลุ่มในไฟล์ฐานข้อมูลผู้ใช้ โดยการตรวจสอบนิยามสำหรับกลุ่มทั้งหมด คำสั่ง **/usr/sysv/bin/grpck** นี้ เป็นเวอร์ชัน System V ของคำสั่ง **grpck** ใน **/usr/sbin/** คำสั่งนี้เรียกใช้คำสั่ง **/usr/sbin/grpck** ที่มีแฟล็ก **-n** และอ็อปชัน ALL

## สถานะออก

- 0 ดำเนินการเป็นผลสำเร็จ
- >0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. เมื่อต้องการตรวจสอบว่ามีสมาชิกกลุ่มทั้งหมดและผู้ดูแลระบบในฐานข้อมูลผู้ใช้ และมีข้อผิดพลาดใดๆ ที่ถูกรายงาน (แต่ไม่ถูกแก้ไข) ให้ป้อนคำสั่งต่อไปนี้:

```
/usr/sysv/bin/grpck
```

## ไฟล์

```
/usr/sysv/bin/grpck
```

มีเวอร์ชัน System V ของคำสั่ง **grpck**

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **pwdck**

คำสั่ง sysck

คำสั่ง usrck

ความปลอดภัย

---

## คำสั่ง grpsvcsctrl

### วัตถุประสงค์

เริ่มทำงานระบบย่อยการให้บริการกลุ่ม

### ไวยากรณ์

```
grpsvcsctrl { -a | -s | -k | -d | -c | -u | -t | -o | -h }
```

### คำอธิบาย

คำสั่ง **grpsvcsctrl** เริ่มทำงานระบบย่อยการให้บริการกลุ่ม สคริปต์ควบคุมนี้จะควบคุมการดำเนินการของระบบย่อยที่จำเป็นสำหรับการให้บริการกลุ่ม ระบบย่อยเหล่านี้ที่อยู่ภายใต้การควบคุมของ system resource controller (SRC) และเป็นของกลุ่มระบบย่อยชื่อ **grpsvcs daemon** ถูกเชื่อมโยงกันด้วยแต่ละระบบย่อย จากมุมมองของการปฏิบัติการ กลุ่มระบบย่อยการให้บริการกลุ่มถูกจัดระบบดังต่อไปนี้:

#### ระบบย่อย

การให้บริการกลุ่ม

#### Subsystem group

**grpsvcs**

#### SRC subsystem

**grpsvcs** — สัมพันธ์กับ **hagsd daemon** ซึ่งระบบย่อยบนโหนดคือ **grpsvcs** ระบบย่อย **grpsvcs** บนแต่ละโหนดจะเชื่อมโยงกับคลัสเตอร์ที่โหนดเป็นสมาชิกอยู่

#### Daemon

**hagsd** — จัดให้มีฟังก์ชันการให้บริการกลุ่มเป็นส่วนใหญ่

สคริปต์ **grpsvcsctrl** โดยปกติไม่ได้รับจากบรรทัดคำสั่ง โดยปกติจะถูกเรียกใช้โดยคำสั่งเริ่มทำงานระหว่างการติดตั้งคลัสเตอร์

สคริปต์ **grpsvcsctrl** จัดให้มีการควบคุมที่หลากหลายสำหรับการดำเนินการ ระบบย่อยการให้บริการกลุ่ม:

- การเพิ่ม การเริ่มทำงาน การหยุดทำงาน การลบและการล้างค่าระบบย่อย
- การเปิดและปิดทำงานการติดตาม

ก่อนดำเนินการฟังก์ชันใดๆ เหล่านี้ สคริปต์จะจัดหาชื่อคลัสเตอร์ ปัจจุบัน

การเพิ่มระบบย่อย: เมื่อระบุแฟล็ก **-a** สคริปต์ควบคุมจะใช้คำสั่ง **mkssys** เพื่อเพิ่ม ระบบย่อยการให้บริการกลุ่มใน SRC สคริปต์ควบคุมจะทำงาน ดังนี้:

1. ช่วยให้แน่ใจว่าระบบย่อย **grpsvcs** หยุดทำงาน

- รับค่าหมายเลขพอร์ตสำหรับระบบย่อยระบบย่อย grpsvcs สำหรับคลัสเตอร์นี้ จาก object data manager (ODM) โกลบอลและช่วยให้แน่ใจว่าหมายเลขพอร์ต ถูกตั้งค่าในไฟล์ `/etc/services` ช่วงของหมายเลขพอร์ตที่ใช้ได้คือ 10000 ถึง 10100
- ชื่อเซิร์ฟเวอร์ที่ป้อนในไฟล์ `/etc/services` คือ `grpsvcs.cluster_name`
- ลบระบบย่อย grpsvcs ออกจาก SRC (ในกรณีที่ยังมีอยู่ในที่นั้น)
- เพิ่มระบบย่อย grpsvcs ใน SRC ชื่อคลัสเตอร์ถูก กำหนดค่าเป็นพารามิเตอร์ daemon บนคำสั่ง `mkssys`

**การเริ่มทำงานระบบย่อย:** เมื่อระบุแฟล็ก `-s` สคริปต์ควบคุมจะใช้คำสั่ง `startsrc` เพื่อเริ่มทำงาน ระบบย่อยการให้บริการกลุ่ม grpsvcs

**การหยุดทำงานระบบย่อย:** เมื่อระบุแฟล็ก `-k` สคริปต์ควบคุมจะใช้คำสั่ง `stopsrc` เพื่อหยุดทำงาน ระบบย่อยการให้บริการกลุ่ม grpsvcs

**การลบระบบย่อย:** เมื่อระบุแฟล็ก `-d` สคริปต์ควบคุมจะใช้คำสั่ง `rmssys` เพื่อลบ ระบบย่อยการให้บริการกลุ่มออกจาก SRC สคริปต์ควบคุมจะทำงาน ดังนี้:

- ช่วยให้แน่ใจว่าระบบย่อย grpsvcs หยุดทำงาน
- ลบระบบย่อย grpsvcs ออกจาก SRC โดยใช้คำสั่ง `rmssys`
- ลบหมายเลขพอร์ตออกจาก ไฟล์ `/etc/services`

**การล้างค่าระบบย่อย:** เมื่อระบุแฟล็ก `-c` สคริปต์ควบคุมจะหยุดทำงานและลบระบบย่อยการให้บริการกลุ่ม สำหรับพาร์ติชันระบบทั้งหมดออกจาก SRC สคริปต์ควบคุมจะทำงาน ดังนี้:

- หยุดทำงานอินสแตนซ์ทั้งหมดของระบบย่อยในกลุ่มระบบย่อยในทุกพาร์ติชัน โดยใช้คำสั่ง `stopsrc -g grpsvcs`
- ลบอินสแตนซ์ทั้งหมดของระบบย่อยในกลุ่มระบบย่อยในทุกพาร์ติชัน ออกจาก SRC โดยใช้คำสั่ง `rmssys`

**การเปิดทำงานการติดตาม:** เมื่อระบุแฟล็ก `-t` สคริปต์ควบคุมจะเปิดทำงานการติดตามสำหรับ `hagsd` daemon โดยใช้คำสั่ง `traceson`

**การปิดทำงานการติดตาม:** เมื่อระบุแฟล็ก `-o` สคริปต์ควบคุมจะปิดทำงานการติดตามสำหรับ `hagsd` daemon โดยใช้คำสั่ง `tracesoff`

**การบันทึกการทำงาน:** ขณะที่กำลังรัน daemons การให้บริการกลุ่ม จะให้ข้อมูลเกี่ยวกับการดำเนินการและข้อผิดพลาดโดยการเขียนรายการใน ไฟล์บันทึกการทำงานในไดเรกทอรี `/var/ha/log`

แต่ละ daemon จำกัดขนาดบันทึกการทำงานเป็นจำนวนบรรทัดที่กำหนดไว้ก่อนสร้าง ค่า ดีฟอลต์คือ 5000 บรรทัด เมื่อขึ้นขีดจำกัด daemon จะผนวกสตริง `.bak` ที่ท้ายชื่อของไฟล์บันทึกการทำงานปัจจุบัน และเริ่มต้นบันทึกการทำงานใหม่ หากเวอร์ชัน `.bak` มีอยู่แล้ว จะต้องถูกลบก่อนที่บันทึกการทำงานปัจจุบันจะถูก เปลี่ยนชื่อ

## แฟล็ก

- `-a`   เพิ่มระบบย่อย
- `-s`   เริ่มทำงานระบบย่อย
- `-k`   หยุดทำงานระบบย่อย

- d    ลบบระบบย่อย
- c    ล้างค่าระบบย่อย (นั่นคือ ลบออกจากพาร์ติชันระบบทั้งหมด)
- u    ลบบระบบย่อยการให้บริการกลุ่มออกจากพาร์ติชันทั้งหมด
- t    เปิดการติดตามสำหรับระบบย่อย
- o    ปิดการติดตามสำหรับระบบย่อย
- h    เขียนคำสั่งการใช้สคริปต์ไปที่เอาต์พุตมาตรฐาน

## ความปลอดภัย

คุณต้องรันด้วย ID ผู้ใช้ root

## สถานะออก

- 0    บ่งชี้ความสำเร็จของคำสั่ง
- 1    หมายถึงมีข้อผิดพลาดเกิดขึ้น

## ข้อจำกัด

สคริปต์นี้ใช้ได้เฉพาะในสภาวะแวดล้อม HACMP เท่านั้น

## เอาต์พุตมาตรฐาน

เมื่อระบุแฟล็ก -h ไว้ข้อความการใช้งานของคำสั่งนี้ ถูกเขียนลงในเอาต์พุตมาตรฐาน

## ข้อผิดพลาดมาตรฐาน

คำสั่งนี้เขียนข้อความแสดงความผิดพลาด (ตามความจำเป็น) ไปที่ข้อผิดพลาดมาตรฐาน

## ตัวอย่าง

1. ในการเพิ่มระบบย่อยการให้บริการกลุ่มกับ SRC ให้ป้อน:
 

```
grpsvcctrl -a
```
2. ในการเริ่มทำงานระบบย่อยการให้บริการกลุ่ม ให้ป้อน:
 

```
grpsvcctrl -s
```
3. ในการหยุดทำงานระบบย่อยการให้บริการกลุ่ม ให้ป้อน:
 

```
grpsvcctrl -k
```
4. ในการลบบระบบย่อยการให้บริการกลุ่มออกจาก SRC ให้ป้อน:
 

```
grpsvcctrl -d
```
5. ในการล้างค่าระบบย่อยการให้บริการกลุ่ม ให้ป้อน:
 

```
grpsvcctrl -c
```
6. ในการเปิดทำงานสำหรับ daemon การให้บริการกลุ่ม hagsd ให้ป้อน:
 

```
grpsvcctrl -t
```

7. ในการปิดทำงานสำหรับ daemon การให้บริการกลุ่ม **hagsd** ให้ป้อน:

```
grpsvcctrl -o
```

## Location

```
/usr/sbin/rsct/bin/grpsvcctrl
```

มีสคริปต์ `grpsvcctrl`

## ไฟล์

```
/var/ha/log/grpsvcs_nodenum_instnum.cluster_name
```

มีบันทึกการทำงานของ **hagsd** daemons บนโหนด

ชื่อไฟล์ประกอบด้วยตัวแปรต่อไปนี้:

*nodenum*

คือหมายเลขโหนดที่ daemon กำลังรัน

*instnum*

คือหมายเลขอินสแตนซ์ของ daemon

*cluster\_name*

คือชื่อของคลัสเตอร์ที่ daemon กำลังรัน

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `lssrc`

คำสั่ง `mkssys`

คำสั่ง `startsrc`

คำสั่ง `stopsrc`

---

## gssd Daemon

### วัตถุประสงค์

การร้องขอเซอร์วิสเคอร์เนลสำหรับการดำเนินการ GSS

### ไวยากรณ์

```
/usr/sbin/gssd
```

## คำอธิบาย

วิธีการรักษาความปลอดภัย NFS บางวิธีเช่น Kerberos 5 ถูกจัดให้มีอยู่ภายใต้ โกลโกที่มีลักษณะทั่วไปมากกว่าที่เรียกว่า General Security Services หรือ GSS ใน AIX เซอร์วิส GSS จัดให้มีโดยไลบรารีในชุดไฟล์ IBM Network Authentication Service (NAS) NAS มาพร้อมกับแพ็คเกจส่วนขยาย gssd daemon ช่วยให้เซอร์วิส GSS เหล่านี้ใช้ได้กับเซิร์ฟเวอร์เคอร์เนลโค้ด NFS หาก gssd daemon ไม่ได้รันอยู่ ความพยายามที่จะเข้าถึงไฟล์ ทาง NFS โดยใช้วิธีการรักษาความปลอดภัย GSS เช่น Kerberos 5 จะล้มเหลว gssd daemon รีจิสเตอร์โดยใช้โปรแกรม RPC หมายเลข 400234

gssd daemon เริ่มทำงานและระบบด้วยคำสั่ง System Resource Controller (SRC) ต่อไปนี้:

```
startsrc -s gssd
stopsrc -s gssd
```

## ไฟล์

### ไอเท็ม

/etc/nfs/hostkey

### คำอธิบาย

ระบุตำแหน่งไฟล์ keytab และ host principal ด้วยรูปแบบต่อไปนี้:

path to keytab file

host principal

/etc/nfs/princmap

ระบุการแม็พกับ host principals ด้วยรูปแบบ ต่อไปนี้:

principal1 alias1 alias2 alias3

principal2 alias1

Aliases สามารถเป็น IP แอดเดรสหรือชื่อโฮสต์ principal ต้องตรงกับโฮสต์คีย์ที่ดูแลรักษาโดย kerberos



---

## h

คำสั่ง AIX ต่อไปนี้เริ่มต้นด้วยตัวอักษร h

---

### คำสั่ง ha.vsd

#### วัตถุประสงค์

เคียวรีและควบคุมกิจกรรมของ rvsd daemon ของระบบย่อยดิสก์เสมือนที่แบ่งใช้ที่เรียกคืนได้

#### ไวยากรณ์

```
ha.vsd {adapter_recovery [on|off] | debug [off] | mksrc | query | quorum n | qsrc | refresh [noquorum] | reset |
reset_quorum | rmsrc | start | stop | trace [off]}
```

#### คำอธิบาย

ใช้คำสั่งนี้เพื่อแสดงข้อมูลเกี่ยวกับระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ เพื่อเปลี่ยนจำนวนโหนดที่จำเป็นสำหรับ quorum และเพื่อเปลี่ยน สถานะของระบบย่อย

#### แฟล็ก

- a ระบุดิสก์ที่แบ่งใช้แบบเสมือนทั้งหมด
- v vsd\_name\_list  
ระบุชื่อดิสก์ที่แบ่งใช้แบบเสมือนอย่างน้อยหนึ่งชื่อ คั่นด้วยเครื่องหมายจุลภาค
- n node\_list  
ระบุหมายเลขโหนดอย่างน้อยหนึ่งหมายเลข คั่นด้วยเครื่องหมายจุลภาค

#### พารามิเตอร์

##### adapter\_recovery [on | off]

เปิดใช้งานหรือปิดใช้งานการกู้คืนอะแดปเตอร์การสื่อสาร ค่าดีฟอลต์คือ on

ระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ต้องถูกรีสตาร์ท เพื่อให้ตัวถูกดำเนินการนี้มีผลใช้ได้

##### debug [off]

ระบุ debug เพื่อเปลี่ยนทิศทางเอาต์พุตมาตรฐานและข้อผิดพลาดมาตรฐาน ของระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ไปยังคอนโซลและทำให้ ระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ไม่ถูกสร้างขึ้นใหม่ หากออกจากการทำงานโดย มีข้อผิดพลาด (คุณสามารถใช้คำสั่ง Iscons เพื่อกำหนดคอนโซล ปัจจุบัน)

ระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ต้องถูกรีสตาร์ท เพื่อให้ตัวถูกดำเนินการนี้มีผลใช้ได้

เมื่อการดีบั๊กถูกเปิดทำงาน และ ระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ถูกรีสตาร์ท ha.vsd trace ควร ถูกเรียกใช้เพื่อเปิดทำงานการติดตาม

ใช้ตัวถูกดำเนินการนี้ภายใต้ คำแนะนำของเจ้าหน้าที่บริการ IBM ของคุณ

**หมายเหตุ:** ค่าดีพอลต์เมื่อโหนดถูกบูตคือให้จัดเส้นทางเอาต์พุตมาตรฐานและ ข้อผิดพลาดมาตรฐานไปยังคอนโซล หากการดีบั๊กถูกปิดทำงาน เอาต์พุตมาตรฐานและ ข้อผิดพลาดมาตรฐานจะถูกจัดเส้นทางไปยัง /dev/null รวมถึงข้อความ การติดตามที่จะเกิดขึ้นทั้งหมดจะสูญหายไป คุณสามารถระบุว่าการดีบั๊กถูกเปิดทำงานหรือไม่ โดยการเรียกใช้ `ha.vsd qsrc` หากการดีบั๊กถูกเปิดทำงาน ค่าส่งคืน จะเป็น:

```
action = "2"
```

**mksrc** ใช้ `mkssys` เพื่อสร้างระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้

**เคียวรี** แสดงสถานะปัจจุบันของระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้อย่างละเอียด

#### **quorum n**

ตั้งค่า `quorum` ซึ่งเป็นจำนวนโหนดทั้งหมดที่ ต้องรวมในกลุ่มก่อนที่ระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนจะถูกเรียกทำงาน โดยทั่วไป `quorum` ถูกกำหนดเป็นหลักของโหนดที่ถูกกำหนดเป็นโหนดดิสก์ที่แบ่งใช้แบบเสมือนในเพียร์โตเมน RSCT แต่คำสั่งนี้อ่อนุญาตให้คุณแทนที่ นิยามนั้นได้

ระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ต้องอยู่ในสถานะแอ็คทีฟเมื่อคุณเรียกใช้คำสั่งนี้ไม่ใช่การเปลี่ยนแปลง ถาวร

**qsrc** แสดงการกำหนดค่า System Resource Controller (SRC) ของ daemon ดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้

#### **refresh [noquorum]**

ใช้คำสั่ง `refresh` เพื่อเริ่มทำงานรีเฟรชโปรโตคอล แบบอะซิงโครนัสกับระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ที่กำลังรันทั้งหมด `quorum` จะถูกรีเซ็ตก่อนที่จะเกิดการรีเฟรช ยกเว้นว่าระบุ `noquorum` ใช้ `ha.vsd query` เพื่อตรวจสอบความสำเร็จ รายการต่อไปนี้ ถูกรีเฟรชในไดเรกทอรีอุปกรณ์

1. โหนดที่ถูกเพิ่มหรือลบออก
2. ดิสก์ที่แบ่งใช้แบบเสมือนที่ถูกเพิ่มหรือลบออก
3. แอ็ททริบิวต์ที่เปลี่ยน `size_in_MB` สำหรับดิสก์ที่แบ่งใช้แบบเสมือน

**reset** หยุดทำงานและรีเซ็ตระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้

#### **reset\_quorum**

รีเซ็ต `quorum` ดีพอลต์

**rmsrc** ใช้ `rmssys` เพื่อลบระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้

**start** เริ่มทำงานระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้

**stop** หยุดทำงานระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้

#### **trace [off]**

ร้องขอหรือหยุดทำงานการติดตามระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ ระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ต้องอยู่ในสถานะแอ็คทีฟเมื่อเรียกใช้คำสั่งนี้

ตัวถูกดำเนินการนี้จะมีความหมายหลังจากตัวถูกดำเนินการ `debug` ได้ถูกใช้เพื่อส่งเอาต์พุตมาตรฐานและข้อผิดพลาดมาตรฐาน ไปยังคอนโซล และระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ถูก รีเซ็ต

## **ความปลอดภัย**

คุณต้องมีสิทธิ์การใช้งาน `root` เพื่อรันคำสั่งนี้

## สถานะออก

0 บ่งชี้ความสำเร็จของคำสั่ง

## ไม่เป็นศูนย์

หมายถึงมีข้อผิดพลาดเกิดขึ้น

## ข้อจำกัด

คุณต้องใช้คำสั่งนี้จากโหนดที่ออนไลน์อยู่ในโดเมนแบบเพียร์ หากต้องการทำให้โดเมนเพียร์ออนไลน์ให้ใช้คำสั่ง `startprdomain` เมื่อต้องการทำให้โหนดออนไลน์ในเพียร์โดเมนที่มีอยู่ ใช้คำสั่ง `startprnode` สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการสร้างและการดูแล RSCT เพียร์โดเมน อ้างอิงถึง *คู่มือการดูแล RSCT*

## ตัวอย่าง

1. ในการหยุดทำงานระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้และรีสตาร์ทให้ป้อน:

```
ha.vsd reset
```

ระบบส่งคืนข้อความ:

```
Waiting for the rvsd subsystem to exit.  
rvsd subsystem exited successfully.  
Starting rvsd subsystem.  
rvsd subsystem started PID=xxx.
```

2. เมื่อต้องการเปลี่ยน quorum เป็นทำโหนดของเพียร์โดเมน RSCT ให้ป้อน:

```
ha.vsd quorum 5
```

ระบบส่งคืนข้อความ:

```
Quorum has been changed from 8 to 5.
```

3. เมื่อต้องการเคียวรีระบบย่อย rvsd ให้ป้อน:

```
ha.vsd query
```

ระบบแสดงข้อความที่เหมือนดังต่อไปนี้:

```
Subsystem      Group      PID      Status  
rvsd           rvsd      18320    active  
rvsd(vsd): quorum= 9/4, active=1, state=idle, isolation=member,  
NoNodes=10, lastProtocol=nodes_failing,  
adapter_recovery=on, adapter_status=up,  
RefreshProtocol has never been issued from this node,  
Running function level 4.1.0.0.
```

โดยที่:

### quorum

คือจำนวนของโหนดหรือเซิร์ฟเวอร์โหนดทั้งหมดที่ต้องเข้าร่วมในกลุ่ม ก่อนที่ระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนจะถูกเรียกทำงาน ในเอาต์พุตระบบด้านบน quorum 9/4 ระบุจำนวนโหนดทั้งหมด (9) และจำนวน เซิร์ฟเวอร์โหนด (4)

**active** ระบุสถานะการเรียกทำงานของกลุ่มที่กำลังถูกรวม:

- 0: กลุ่มไม่แอคทีฟ (quorum ไม่ตรงกัน)
- 1: กลุ่มแอคทีฟ และดิสก์ที่แบ่งใช้ถูกเรียกทำงาน

**state** ระบุว่าโปรโตคอลปัจจุบันกำลังรัน

#### isolation

ระบุสถานะความเป็นสมาชิกกลุ่ม

#### isolated:

กลุ่ม "join" ยังไม่ถูกเสนอ

#### proposed:

กลุ่ม "join" ถูกเสนอ

#### member:

เราเป็นสมาชิก (ผู้ให้บริการ) ของกลุ่ม

#### NoNodes

ระบุจำนวนโหนดที่ได้เข้าร่วมในกลุ่ม

#### lastProtocol

ระบุโปรโตคอลล่าสุดที่ถูกรันในทั้งกลุ่ม

#### adapter\_recovery

ระบุการสนับสนุนการกู้คืนอะแดปเตอร์การสื่อสาร:

**on:** การกู้คืนอะแดปเตอร์ถูกเปิดใช้งาน

**off:** การกู้คืนอะแดปเตอร์ถูกปิดใช้งาน

#### adapter\_status

ระบุสถานะอะแดปเตอร์การสื่อสาร:

**up:** อะแดปเตอร์เปิดทำงาน

**down:** อะแดปเตอร์ปิดทำงาน

#### unknown:

สถานะอะแดปเตอร์เป็นไม่รู้จัก

#### RefreshProtocol ...

ระบุว่ารีเฟรชโปรโตคอลได้ถูกเรียกใช้งานโหนดนี้ ถ้าเป็นดังนั้น วันที่และเวลาของการทำสำเร็จหรือข้อผิดพลาด จะถูกแสดง

#### Running function level

ระบุระดับฟังก์ชันที่ระบบย่อยกำลังรัน โดยแสดงเวอร์ชัน รีลีส์ การแก้ไข รูปแบบระดับการแก้ไข (vrmf) (ที่มีอยู่ร่วมกับระดับล่าง ของระบบย่อย ซึ่งอาจจำกัดให้รันในระดับฟังก์ชันที่ลดจำนวน)

## Location

/opt/rsct/vsd/bin/ha.vsd

สิ่งอ้างอิงที่เกี่ยวข้อง:

## คำสั่ง ha\_star

### วัตถุประสงค์

ประมวลผลเหตุการณ์ที่มีความพร้อมใช้สูง

### ไวยากรณ์

ha\_star [-C]

### คำอธิบาย

คำสั่ง ha\_star คือคำสั่งในการจัดการความพร้อมใช้ สูงทั่วไป คำสั่งจะถูกเรียกใช้โดยอัตโนมัติโดยระบบปฏิบัติการผ่านทาง /etc/rc.ha\_star เมื่อมีการรายงานความล้มเหลว CPU ที่อาจเกิดขึ้น โดยเฟิร์มแวร์

หาก ha\_star ถูกเรียกใช้โดยไม่มีแฟล็ก เฉพาะเหตุการณ์ใหม่ เท่านั้นที่ได้รับการจัดการ หาก ha\_star ไม่พบเหตุการณ์ใหม่ใดๆ จะออกจากการทำงาน

เมื่อกำลังรัน ha\_star จะจัดการเหตุการณ์ใหม่ทั้งหมด แม้แต่ที่เข้ามาในขณะที่ ha\_star กำลังจัดการเหตุการณ์ ที่มีอยู่แล้ว หนึ่งในอินสแตนซ์ของ ha\_star เท่านั้นที่สามารถ รันได้ในเวลาที่กำหนด อินสแตนซ์ที่สองของ ha\_star ถูกเรียกใช้ เมื่ออันที่หนึ่ง ออกจากการทำงาน

ระบบปฏิบัติการเรียกใช้ ha\_star เมื่อมีการรายงานถึง เหตุการณ์ความพร้อมใช้สูง การจัดการเหตุการณ์อาจล้มเหลว หรืออาจ ถูกยกเลิก (ตัวอย่างเช่น โดยสัญญาณ) เหตุการณ์ที่ล้มเหลวหรือถูกยกเลิกจะถูกเก็บในหน่วยความจำ ภายในเคอร์เนล เมื่อ สาเหตุของการยกเลิกได้ถูกแก้ไขแล้ว การจัดการเหตุการณ์ สามารถดำเนินการใหม่อีกครั้ง ซึ่งเกิดขึ้นเมื่อ ha\_star ถูก เรียกใช้ ด้วยตนเองโดยผู้ดูแลระบบ

คำสั่ง ha\_star สร้างรายการบันทึกข้อผิดพลาด หรือความล้มเหลว

### รายละเอียดตามชนิดเหตุการณ์

คำสั่ง ha\_star ถูกเรียกใช้โดยระบบปฏิบัติการ เพื่อยกเลิกการจัดสรร CPU เมื่อตรวจพบเหตุการณ์ความล้มเหลวที่อาจเกิดขึ้น การยกเลิกการจัดสรรนี้อาจล้มเหลวเนื่องจากบางเรดยังคงถูกโยกกับ CPU ที่กำลัง ถูกยกเลิกการจัดสรร ในบางกรณี ผู้ดูแล ระบบสามารถแก้ไขปัญหาซึ่งนำไปสู่ความล้มเหลวของการยกเลิกการจัดสรรได้ ตัวอย่างเช่น สามารถ ระบุและหยุดทำ งานแอปพลิเคชันที่มีเรดโยกอยู่กับโลจิคัล CPU สุดท้าย

แฟล็ก -C ระบุว่าเหตุการณ์ความพร้อมใช้ สูงที่จะถูกทำงานต่อคือเหตุการณ์การยกเลิกการจัดสรร CPU

### แฟล็ก

ไอเอ็ม  
-C

คำอธิบาย  
ระบุเหตุการณ์ที่จะถูกรีเซ็ตคือการยกเลิกการจัดสรร CPU

## ไฟล์

ไอเอ็ม  
/usr/sbin/ha\_star

คำอธิบาย  
มีคำสั่ง ha\_star

### ข้อมูลที่เกี่ยวข้อง:

การยกเลิกการจัดสรรตัวประมวลผลแบบไดนามิก  
การเปิดใช้งานคำสั่ง

---

## คำสั่ง ha\_vsd

### วัตถุประสงค์

เริ่มทำงานและรีเซ็ตระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ รวมถึงการกำหนดค่าดิสก์ที่แบ่งใช้แบบเสมือน และการเรียกทำงานระบบย่อย ความสามารถในการกู้คืน

### ไวยากรณ์

ha\_vsd [reset]

### คำอธิบาย

ใช้คำสั่งนี้เพื่อเริ่มทำงานซอฟต์แวร์ดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ หลังจากที่คุณได้ติดตั้ง หรือด้วยอ็อปชัน reset เพื่อหยุดทำงานและรีเซ็ตโปรแกรม

### แฟล็ก

-a ระบุดิสก์ที่แบ่งใช้แบบเสมือนทั้งหมด

-v *vsd\_name\_list*

ระบุชื่อดิสก์ที่แบ่งใช้แบบเสมือนอย่างน้อยหนึ่งชื่อ คั่นด้วยเครื่องหมายจุลภาค

-n *node\_list*

ระบุหมายเลขโหนดอย่างน้อยหนึ่งหมายเลข คั่นด้วยเครื่องหมายจุลภาค

### พารามิเตอร์

reset หยุดทำงานและรีเซ็ตระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้

### ความปลอดภัย

คุณต้องมีสิทธิ์การใช้งาน root เพื่อรันคำสั่งนี้

## สถานะออก

- 0 บ่งชี้ความสำเร็จของคำสั่ง
- 1 หมายถึงมีข้อผิดพลาดเกิดขึ้น

## ข้อจำกัด

คุณต้องใช้คำสั่งนี้จากโหนดที่ออนไลน์อยู่ในโดเมนแบบเพียร์ หากต้องการทำให้โดเมนเพียร์ออนไลน์ให้ใช้คำสั่ง `starttrpdomain` เมื่อต้องการทำให้โหนดออนไลน์ในเพียร์โดเมนที่มีอยู่ ใช้คำสั่ง `starttrpnode` สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการสร้างและการดูแล R SCT เพียร์โดเมน อ้างอิงถึง *คู่มือการดูแล R SCT*

## ตัวอย่าง

ในการหยุดทำงานระบบย่อยดิสก์ที่แบ่งใช้แบบเสมือนที่สามารถกู้คืนได้ และรีเซ็ตาร์ทให้ป้อน:

```
ha_vsd reset
```

## Location

```
/opt/rsct/vsd/bin/ha_vsd
```

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง ha.vsd” ในหน้า 749

---

## haemd Daemon

### วัตถุประสงค์

เฝ้าดูอินสแตนซ์ตัวแปรรีซอร์สที่ถูกอัปเดตโดยรีซอร์สมอนิเตอร์ และสร้างและรายงานเหตุการณ์ไปยังไคลเอ็นต์โปรแกรม

### ไวยากรณ์

```
haemd
```

### คำอธิบาย

**haemd** (ตัวจัดการเหตุการณ์) daemon เฝ้าดูอินสแตนซ์ตัวแปรรีซอร์ส ที่ถูกอัปเดตโดยรีซอร์สมอนิเตอร์ และสร้างและรายงานเหตุการณ์ไปยังไคลเอ็นต์โปรแกรม

อินสแตนซ์หนึ่งของ **haemd** daemon เรียกใช้งานบนทุกโหนดของคลัสเตอร์ **haemd** daemon อยู่ภายใต้การควบคุมของ system resource controller (SRC)

เนื่องจาก daemon อยู่ภายใต้การควบคุม SRC ซึ่งไม่สามารถเริ่มทำงานจากบรรทัดคำสั่ง โดยตรง โดยปกติจะเริ่มทำงานโดยคำสั่ง `emsvcsctrl` หากคุณต้องเริ่มทำงานหรือหยุดทำงาน daemon โดยตรง ให้ใช้คำสั่ง `emsvcsctrl`

เมื่อ SRC สร้าง **haemd** daemon โปรแกรมที่แท้จริงที่เริ่มทำงาน คือ **haemd\_HACMP** โปรแกรม **haemd\_HACMP** หลังจากที่ได้รับรวบรวม ข้อมูลที่จำเป็นโดย daemon แล้ว จะรันโปรแกรม **haemd** ใน อีกนัยหนึ่ง โปรแกรม **haemd\_HACMP** จะถูกแทนที่โดยโปรแกรม **haemd** ในที่สร้างขึ้นโดย SRC

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ daemon ตัวจัดการเหตุการณ์ โปรดดูที่คำสั่ง **emsvcsctrl**

## ข้อมูลจำเพาะการนำไปใช้งาน

Daemon นี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT) สำหรับ AIX

### Location

`/usr/sbin/rsct/bin/haemd`

ตำแหน่งของ **haemd** daemon

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง **emsvcsctrl**” ในหน้า 379

“คำสั่ง **haemd\_HACMP**”

---

## คำสั่ง **haemd\_HACMP**

### วัตถุประสงค์

โปรแกรมเริ่มทำงานสำหรับ daemon ตัวจัดการเหตุการณ์

### ไวยากรณ์

**haemd\_HACMP** [`-d trace_arg`]

### คำอธิบาย

คำสั่ง **haemd\_HACMP** คือโปรแกรมเริ่มทำงานสำหรับ **haemd** daemon เมื่อระบบย่อยการจัดการเหตุการณ์ถูกกำหนดคอนฟิก ใน system resource controller (SRC) โดยคำสั่ง **emsvcsctrl** นั้น **haemd\_HACMP** จะถูกระบุ เป็นโปรแกรมที่จะเริ่มทำงาน

โปรแกรมนี้สามารถถูกเรียกใช้โดย SRC เท่านั้น ในการเริ่มทำงานระบบย่อยการจัดการ เหตุการณ์ ให้ใช้คำสั่ง **emsvcsctrl**

### แฟล็ก

`-d trace_arg`

ควรใช้ภายใต้คำแนะนำของ IBM Support Center เท่านั้น อาร์กิวเมนต์การติดตามที่เป็นไปได้จะเหมือนกับคำสั่ง **haemtrcon** ยกเว้นสำหรับ **reg** และ **dinsts** ในการใช้แฟล็กนี้ นิยามระบบย่อย **emsvcs** ใน SRC ต้องถูกเปลี่ยนโดยใช้คำสั่ง **chssys** ที่มีแฟล็ก `-a` จากนั้น daemon ต้อง ถูกหยุดทำงานและรีสตาร์ท

### ข้อจำกัด

คำสั่งนี้ใช้ได้เฉพาะในสถานะแวดล้อม HACMP เท่านั้น

## ข้อมูลจำเพาะการนำไปใช้งาน

สคริปต์นี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

### Location

/usr/sbin/rsct/bin/haemd\_HACMP

ตำแหน่งของโปรแกรม haemd\_HACMP

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง emsvcsctrl” ในหน้า 379

“haemd Daemon” ในหน้า 755

“คำสั่ง haemtrcon” ในหน้า 764

---

## คำสั่ง haemqvar

### วัตถุประสงค์

เคียวรีตัวแปรรีชอร์ส

### ไวยากรณ์

```
haemqvar [ -H domain | -S domain ] [ -c | -d | -i ] [ -f file ] [ -h ] [ class var rsrcID [ " ] ]
```

### คำอธิบาย

คำสั่ง haemqvar จะเคียวรีระบบย่อย Event Management เพื่อดูข้อมูล เกี่ยวกับตัวแปรรีชอร์ส โดยค่าดีฟอลต์ คำสั่งจะเขียนนิยามสำหรับตัวแปรรีชอร์สทั้งหมด ในโดเมน SP ปัจจุบัน ไปยังเอาต์พุตมาตรฐาน นั่นคือ พาร์ติชันระบบ SP ปัจจุบันที่กำหนดโดยตัวแปรสถานะแวดล้อม SP\_NAME หากไม่ ตั้งค่า SP\_NAME จะใช้พาร์ติชันระบบดีฟอลต์แทน แฟล็ก -S สามารถใช้เพื่อระบุโดเมน SP (พาร์ติชันระบบ) อื่น ในการเคียวรีตัวแปรในโดเมน HACMP ให้ใช้แฟล็ก -H สำหรับโดเมน SP โดเมนแฟล็กอาร์กิวเมนต์คือชื่อ พาร์ติชันระบบ สำหรับโดเมน HACMP โดเมนแฟล็กอาร์กิวเมนต์คือชื่อคลัสเตอร์ HACMP เมื่อระบุแฟล็ก -H คำสั่งต้องถูกเรียกใช้งานบนโหนดใดโหนดหนึ่ง ในคลัสเตอร์ HACMP/ES

โดยจะรายงานข้อมูลต่อไปนี้สำหรับแต่ละนิยามตัวแปรรีชอร์ส:

- Variable Name
- Value Type
- Data Type
- SBS Format (หากชนิดข้อมูลเป็น Structured Byte String)
- Initial Value
- Class
- Locator
- Variable คำอธิบาย
- Resource ID และรายละเอียด

- Default Expression (หากกำหนด) และรายละเอียด

เนื่องจากลักษณะการทำงานดีฟอลต์ของคำสั่งนี้อาจสร้างเอาต์พุตจำนวนมาก ดังนั้นเอาต์พุตมาตรฐานควรถูกเปลี่ยนทิศทางไปยังไฟล์

หากระบุแฟล็ก -d เฉพาะชื่อตัวแปรรีซอร์สและรายละเอียด อย่างย่อเท่านั้นที่ถูกเขียนไปยังเอาต์พุตมาตรฐาน หนึ่งชื่อและหนึ่งรายละเอียดต่อหนึ่งบรรทัด

หากระบุแฟล็ก -c ค่าปัจจุบันของอินสแตนซ์ตัวแปรรีซอร์ส ทั้งหมดจะถูกเขียนไปยังเอาต์พุตมาตรฐาน หนึ่งอินสแตนซ์ต่อหนึ่งบรรทัด บรรทัดของเอาต์พุต จะมีตำแหน่งของอินสแตนซ์ตัวแปรรีซอร์ส (หมายเลขโหนด) ชื่อ ตัวแปรรีซอร์ส ID รีซอร์สของอินสแตนซ์ และค่าอินสแตนซ์ ตัวแปรรีซอร์ส หากตัวแปรรีซอร์สมีชนิดข้อมูลเป็น Structured Byte String (SBS) ดังนั้นค่าของแต่ละฟิลด์ SBS จะถูกรายงาน

แฟล็ก -i รายงานตัวแปรเดียวกันกับแฟล็ก -c ยกเว้นว่า ค่าของอินสแตนซ์ตัวแปรจะเป็นค่าล่าสุดที่ทราบ มากกว่าค่า ปัจจุบัน แฟล็ก -i เป็นประโยชน์ต่อการพิจารณาว่ามีอินสแตนซ์ตัวแปรรีซอร์สใดอยู่

สำหรับทั้งแฟล็ก -c และ -i หากพบข้อผิดพลาดในการจัดหา ข้อมูลเกี่ยวกับอินสแตนซ์ตัวแปรรีซอร์ส บรรทัดเอาต์พุต ประกอบด้วย ข้อความแสดงความผิดพลาด โค้ดระบุความผิดพลาดสัญลักษณ์ ตำแหน่งที่เกิดข้อผิดพลาด (หากสามารถระบุได้) ชื่อตัวแปรรีซอร์ส และ ID รีซอร์ส

ในการส่งกลับข้อมูลเกี่ยวกับตัวแปรรีซอร์สที่เจาะจง ให้ระบุคลาส ตัวถูกดำเนินการ var และ rsrcID ตัวถูกดำเนินการเหล่านี้สามารถใช้ซ้ำเพื่อระบุตัวแปร รีซอร์สเพิ่มเติม นอกจากนี้ ตัวถูกดำเนินการ var และ rsrcID ยังสามารถเป็น wildcard เพื่อจับคู่หมายเลขของตัวแปรรีซอร์ส โปรดทราบว่าตัวถูกดำเนินการสตริง null หรือ เครื่องหมายดอกจันต้องอยู่ในเครื่องหมายคำพูดในเชลล์

หากคลาสไม่เป็นสตริง null ตัวแปรทั้งหมดในคลาสที่ระบุ ที่ถูกจำกัดโดยอาร์กิวเมนต์ var และ rsrcID จะเป็นเป้าหมายของเคียวรี หากคลาสเป็นสตริง null ตัวแปรของคลาสทั้งหมด ที่ถูกจำกัดโดยอาร์กิวเมนต์ var และ rsrcID จะเป็นเป้าหมายของเคียวรี อาร์กิวเมนต์ var สามารถใช้ wildcard ด้วยวิธีใดวิธีหนึ่งในสองวิธีนี้:

1. ระบุชื่อตัวแปรเป็นสตริง null
2. ตัดปลายชื่อหลังคอมโพเนนต์ใดๆ

เมื่อชื่อตัวแปรรีซอร์สเป็น wildcard ในรูปแบบแรก ดังนั้น ตัวแปรรีซอร์สทั้งหมด ที่จำกัดโดยอาร์กิวเมนต์ class และ rsrcID จะเป็นเป้าหมายของเคียวรี เมื่อตัวแปรรีซอร์สเป็น wildcard ใน รูปแบบที่สอง ตัวแปรรีซอร์สทั้งหมดที่มีคอมโพเนนต์ลำดับสูง (ซ้ายสุด) ตรงกับอาร์กิวเมนต์ var ที่จำกัดโดยอาร์กิวเมนต์ class และ rsrcID เป็นเป้าหมายของเคียวรี

อินสแตนซ์ตัวแปรรีซอร์ส หรือนิยามทั้งหมด หากไม่ระบุแฟล็ก -c หรือ -i ของตัวแปรที่ระบุโดยอาร์กิวเมนต์ class และ var ที่ตรงกับอาร์กิวเมนต์ rsrcID คือเป้าหมายของเคียวรี

หากไม่ระบุ -c หรือ -i อาร์กิวเมนต์ rsrcID จะเป็น รายการที่ค้นด้วยเคมีโคลนของชื่ออิลีเมนต์ ID รีซอร์ส หากระบุแฟล็ก -c หรือ -i อาร์กิวเมนต์ rsrcID คือรายการที่ค้นด้วยเคมีโคลน ของชื่อ/ค่า คู่ชื่อ/ค่าประกอบด้วยชื่ออิลีเมนต์ ID รีซอร์สตามด้วยเครื่องหมายเท่ากับตามด้วยค่าของอิลีเมนต์ ID รีซอร์ส ค่าอิลีเมนต์อาจประกอบด้วยค่าเดียว ช่วงของค่า รายการที่ค้นด้วยเครื่องหมายจุลภาค ค่าเดียว หรือรายการที่ค้นด้วยเครื่องหมายจุลภาคของช่วง ช่วงจะอยู่ในรูป a-b และใช้ได้สำหรับอิลีเมนต์ ID รีซอร์สไม่ว่าชนิดใดเท่านั้น (ข้อมูล ชนิดสามารถหาได้จากนิยามตัวแปร โดยต้องไม่มี ช่องว่างใน ID รีซอร์ส

อิลีเมนต์ ID รีซอร์สเป็น wildcard โดยการระบุค่าเป็นอักขระ เครื่องหมายดอกจัน เฉพาะตัวแปรที่ถูกกำหนดที่จะมีอิลีเมนต์ และเฉพาะ อิลีเมนต์ที่ระบุในอาร์กิวเมนต์ rsrcID เป็นเป้าหมายของเคียวรี หาก อิลีเมนต์ใดๆ ของ ID รีซอร์สประกอบด้วย อักขระเครื่องหมายดอกจัน แทนค่าชื่อ/ค่า (หรือเฉพาะชื่อหากเป็นการเคียวรีนิยาม) ตัวแปรทั้งหมด ที่ถูกกำหนดเพื่อให้มี อย่างน้อยอิลีเมนต์ที่ระบุที่คงเหลืออยู่ที่ เป็นเป้าหมายของเคียวรี ID รีซอร์สทั้งหมดมี wildcard หากมี เฉพาะเครื่องหมายดอกจันเท่านั้น อินสแตนซ์ทั้งหมดของตัวแปรรีซอร์สทั้งหมด ที่จำกัดโดยอาร์กิวเมนต์ class และ var เป็นเป้าหมายของเคียวรี

โปรดทราบว่าอาร์กิวเมนต์ rsrcID ต้องอยู่ในเครื่องหมายคำพูดในเซลล์หากมี เซมิโคลอนหรือเครื่องหมายดอกจัน

ตัวถูกดำเนินการ class, var และ rsrcID สามารถอยู่ในไฟล์ โดยมีหนึ่งชุดของ ตัวถูกดำเนินการต่อหนึ่งบรรทัด แทนการถูกระบุ เป็นอาร์กิวเมนต์คำสั่ง ใช้แฟล็ก -f เพื่อระบุชื่อของไฟล์สำหรับคำสั่ง หากใช้แฟล็ก -f ตัวถูกดำเนินการใดๆ ในคำสั่งจะถูกข้าม ภายในไฟล์ สตริง null จะถูกระบุเป็นเครื่องหมายคำพูดคู่ติดกันสองตัว ID รีซอร์สที่ wildcard โดยสมบูรณ์ สามารถเป็นเครื่องหมายดอกจัน (\*) เดียว หรือเครื่องหมายดอกจันใน เครื่องหมายคำพูด ("\*") อาร์กิวเมนต์ต้องถูกคั่นด้วยช่องว่าง หรือแท็บ บน แต่ละบรรทัด

ตัวอย่างบางตัวอย่างของการใช้ wildcards ในอาร์กิวเมนต์ rsrcID ตาม สำหรับ ตัวอย่างเหล่านี้ จะถือว่าอาร์กิวเมนต์ class และ var เป็นสตริง null หากอาร์กิวเมนต์ class หรือ var หรือทั้งคู่ไม่ได้เป็นสตริง null เป้าหมายสำหรับเคียวรี จะถูกจำกัดให้สอดคล้องกัน ในสามตัวอย่างแรก ตัวแปรทั้งหมดที่มี ID รีซอร์สถูกกำหนดเพื่อให้มี อิลีเมนต์ NodeNum, VG และ LV และ อิลีเมนต์ เหล่านี้แทนที่ที่มีค่าตรง

1. ในตัวอย่างนี้ มีเพียงหนึ่งอินสแตนซ์เท่านั้นที่ตรง:

```
NodeNum=5;VG=rootvg;LV=hd4
```

2. ในตัวอย่างนี้ หนึ่งอินสแตนซ์จากแต่ละโหนดที่ตรง:

```
NodeNum=*;VG=rootvg;LV=hd4
```

3. ในตัวอย่างนี้ อินสแตนซ์ทั้งหมดของตัวแปรรีซอร์สที่จับคู่ที่ ตรง:

```
NodeNum=*;VG=*;LV=*
```

4. ในตัวอย่างนี้ ตัวแปรทั้งหมดที่มี ID รีซอร์สถูกกำหนดเพื่อให้มี อิลีเมนต์ NodeNum เท่านั้นที่ตรง อินสแตนซ์ที่ตรงจะถูก เชื่อมโยง กับโหนด 9:

```
NodeNum=9
```

5. ในตัวอย่างนี้ ตัวแปรชุดเดียวกันจะตรง แต่อินสแตนซ์ทั้งหมด ของแต่ละตัวแปรจะตรง:

```
NodeNum=*
```

6. ในตัวอย่างนี้ ตัวแปรทั้งหมดที่มี ID รีซอร์สถูกกำหนดเพื่อให้มี อิลีเมนต์ NodeNum และ VG รวมถึงอิลีเมนต์เพิ่มเติม จะตรง อินสแตนซ์ที่ตรงจะถูกเชื่อมโยง กับโหนด 9:

```
NodeNum=9;VG=*;*
```

7. ในตัวอย่างนี้ ตัวแปรทั้งหมดที่มี ID รีซอร์สถูกกำหนดเพื่อให้มี อิลีเมนต์ NodeNum รวมถึงอิลีเมนต์เพิ่มเติมที่ตรง อินสแตนซ์ทั้งหมดของตัวแปรที่ตรง:

```
NodeNum=*;*
```

ช่วยให้มีความยืดหยุ่นในการระบุตัวแปรรีซอร์สสำหรับการเคียวรี โดยเป็นไปได้ที่อาจไม่มีอินสแตนซ์ตัวแปรรีซอร์สหรือนิยามตัวแปรรีซอร์สที่จะจับคู่ หากไม่มีการจับคู่ ข้อมูลข้อผิดพลาดที่เหมาะสมจะถูกรายงาน อาจในรูปที่อธิบายด้านบน หรือดังต่อไปนี้

หากข้อกำหนดคุณลักษณะของอาร์กิวเมนต์ class, var หรือ rsrcID มีข้อผิดพลาด บรรทัดเอาต์พุตจะมีข้อความแสดงข้อผิดพลาด โค้ดระบุความผิดพลาดสัญลักษณ์ และชื่อ คลาสชื่อตัวแปรรีซอร์ส และ ID. รีซอร์สที่ระบุ

## แฟล็ก

### -H domain

เคียวรีตัวแปรรีชอร์สในโดเมน HACMP ที่ระบุโดย *domain*

### -S domain

เคียวรีตัวแปรรีชอร์สในโดเมน SP ที่ระบุโดย *domain*

-c เคียวรีค่าตัวแปรรีชอร์สปัจจุบัน

-d เคียวรีนิยามตัวแปรรีชอร์สแต่สร้างเอาต์พุตในรูปแบบย่อ

-i อินสแตนซ์เคียวรีของตัวแปรรีชอร์ส

-f file เคียวรีตัวแปรรีชอร์สที่ระบุใน *file*

-h แสดงคำสั่งการใช้งาน

## พารามิเตอร์

*class* ระบุชื่อของคลาสตัวแปรรีชอร์ส หรือสตริง null

*var* ระบุชื่อของตัวแปรรีชอร์ส หรือสตริง null

*rsrCID* ระบุ ID รีชอร์สหรือเครื่องหมายดอกจัน

## ความปลอดภัย

คุณต้องมีสิทธิ์ *root* และการเข้าถึงเพื่อเขียนใน SDR เพื่อรันคำสั่งนี้

คุณควรกำลังรันอยู่บนเวิร์กสเตชันการควบคุม ก่อนรันคำสั่งนี้ คุณต้องตั้งค่าตัวแปรสภาวะแวดล้อม *SP\_NAME* ให้เป็นชื่อพาร์ติชันระบบที่เหมาะสม

## สถานะออก

0 บ่งชี้ความสำเร็จของคำสั่ง

1 หมายถึงมีข้อผิดพลาดเกิดขึ้น โดยมีข้อความแสดงความผิดพลาดอย่างน้อยหนึ่ง ข้อความที่ระบุถึงสาเหตุของข้อผิดพลาด

## ข้อจำกัด

คำสั่งนี้ใช้ได้เฉพาะในสภาวะแวดล้อม PSSP เท่านั้น

## เอาต์พุตมาตรฐาน

เมื่อคำสั่งดำเนินงานสำเร็จเรียบร้อย จะเขียนข้อความแสดงข้อมูล ต่อไปนี้:

การอ่านข้อมูล การจัดการเหตุการณ์ สำหรับพาร์ติชัน *syspar\_name*

*CDB=new\_EMDCB\_file\_name* *Version=EMDCB\_version\_string*

## ข้อผิดพลาดมาตรฐาน

คำสั่งนี้เขียนข้อความแสดงความผิดพลาด (ตามความจำเป็น) ไปที่ข้อผิดพลาดมาตรฐาน

### ตัวอย่าง

1. ในการขอรับนิยามสำหรับตัวแปรรีชอร์สทั้งหมดในคลัสเตอร์ปัจจุบัน และใส่เอาต์พุตในไฟล์ให้ป้อน:  

```
haemqvar -H HAcluster > vardefs.out
```
2. ในการจัดการรายการในรูปแบบย่อของตัวแปรรีชอร์สทั้งหมดที่มี ID รีชอร์ส มีอิลีเมนต์ VG ในคลัสเตอร์ HACMP ชื่อ HAcluster ให้ป้อน:  

```
haemqvar -H HAcluster -d "" "" "VG;*"
```
3. ในการจัดการตัวแปรรีชอร์สที่ ID รีชอร์สมีเฉพาะอิลีเมนต์ VG และ NodeNum เท่านั้น ให้ป้อน:  

```
haemqvar -H HAcluster -d "" "" "VG;NodeNum"
```

### Location

/usr/sbin/rsct/bin/haemqvar

ตำแหน่งของคำสั่ง **haemqvar**

### ไฟล์

/usr/sbin/rsct/install/config/haemloadlist

มีข้อมูลการกำหนดค่าดีฟอลต์สำหรับระบบย่อย Event Management

### ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

---

## คำสั่ง haemtrcoff

### วัตถุประสงค์

ปิดทำการการติดตาม Event Manager daemon

### ไวยากรณ์

```
haemtrcoff -s subsys_name -a trace_list
```

### คำอธิบาย

คำสั่ง **haemtrcoff** ใช้เพื่อปิดทำงานกิจกรรมที่ระบุ ของ Event Manager daemon เอาต์พุตการติดตามจะอยู่ในบันทึกการติดตาม Event Management สำหรับพาร์ติชันระบบ

### แฟล็ก

-s *subsys\_name*

ระบุชื่อของระบบย่อย Event Management บนโหนดนี้ คือ emsvcs อาร์กิวเมนต์นี้ต้องถูกระบุ

-a trace\_list

ระบุรายการของอาร์กิวเมนต์การติดตาม แต่ละอาร์กิวเมนต์จะระบุชนิดของกิจกรรมที่การติดตามจะถูกปิดทำงาน โดยต้องระบุอย่างน้อยหนึ่งอาร์กิวเมนต์ หากระบุมากกว่าหนึ่งอาร์กิวเมนต์ อาร์กิวเมนต์ต้องถูก คั่นด้วยเครื่องหมายจุลภาค รายการจะต้องไม่มีช่องว่าง

## พารามิเตอร์

อาร์กิวเมนต์การติดตามต่อไปนี้สามารถระบุได้:

init	หยุดการติดตามการกำหนดค่าเริ่มต้นของ Event Manager daemon
config	หยุดการดั้มพ์ข้อมูลจากไฟล์คอนฟิกูเรชัน
insts	หยุดการติดตามอินสแตนซ์ตัวแปรรีซอร์สที่จัดการโดย daemon
rmctrl	หยุดการติดตามตัวควบคุมรีซอร์สมอนิเตอร์
cci	หยุดการติดตามอินเตอร์เฟซการสื่อสารไคลเอ็นต์ (ภายใน)
emp	หยุดการติดตามโปรโตคอลตัวจัดการเหตุการณ์
obsv	หยุดการติดตามการสังเกตตัวแปรรีซอร์ส
evgn	หยุดการติดตามการสร้างและการแจ้งเตือนเหตุการณ์
reg	หยุดการติดตามการลงทะเบียนและการถอนการลงทะเบียนเหตุการณ์
pci	หยุดการติดตามอินเตอร์เฟซการสื่อสารเพียร์ (ภายใน)
msgs	หยุดการติดตามข้อความทั้งหมดที่เข้ามายัง และถูกเรียกใช้จาก daemon
เคียววี	หยุดการติดตามเคียววีที่ถูกจัดการโดย daemon
gsi	หยุดการติดตามอินเตอร์เฟซการให้บริการกลุ่ม (ภายใน)
eval	หยุดการติดตามการประเมินค่านิพจน์
rdi	หยุดการติดตามอินเตอร์เฟซ daemon ที่เชื่อถือได้ (ภายใน)
sched	หยุดการติดตามตัวกำหนดตารางเวลาภายใน
shm	หยุดการติดตามกิจกรรมการจัดการหน่วยความจำที่แบ่งใช้
ทั้งหมด	หยุดการติดตามกิจกรรมทั้งหมด

all\_but\_msgs

หยุดการติดตามกิจกรรมทั้งหมดยกเว้นข้อความ กิจกรรมข้อความ ถูกกำหนดโดยอาร์กิวเมนต์ msgs

## ความปลอดภัย

คุณต้องมีสิทธิ์ root และการเข้าถึงเพื่อเขียนใน SDR เพื่อรันคำสั่งนี้

คุณควรกำลังรันอยู่บนเวิร์กสเตชันการควบคุม ก่อนรันคำสั่งนี้ คุณต้องตั้งค่าตัวแปรสภาวะแวดล้อม SP\_NAME ให้เป็นชื่อพาร์ติชันระบบที่เหมาะสม

## สถานะออก

0 บ่งชี้ความสำเร็จของคำสั่ง

1 หมายถึงมีข้อผิดพลาดเกิดขึ้น โดยมีข้อความแสดงความผิดพลาดอย่างน้อยหนึ่ง ข้อความที่ระบุถึงสาเหตุของข้อผิดพลาด

## ข้อจำกัด

อย่างไร้คำสั่งนี้ระหว่างการดำเนินการปกติ ใช้คำสั่งนี้ภายใต้ คำแนะนำของ IBM Support Center เท่านั้น ซึ่งจะให้ข้อมูล เพื่อวัตถุประสงค์ในการดีบั๊ก และอาจลดผลการทำงานของระบบย่อยการจัดการ เหตุการณ์ลง หรืออาจมีสิ่งอื่นที่กำลังรันในพาร์ติชันระบบ

## เอาต์พุตมาตรฐาน

เมื่อคำสั่งดำเนินงานสำเร็จเรียบร้อย จะเขียนข้อความแสดงข้อมูล ต่อไปนี้:

การอ่านข้อมูล การจัดการเหตุการณ์ สำหรับพาร์ติชัน *syspar\_name*

```
CDB=new_EMADB_file_name Version=EMADB_version_string
```

## ข้อผิดพลาดมาตรฐาน

คำสั่งนี้เขียนข้อความแสดงความผิดพลาด (ตามความจำเป็น) ไปที่ข้อผิดพลาดมาตรฐาน

## ตัวอย่าง

- ในการปิดทำงานการติดตามทั้งหมดสำหรับระบบย่อย Event Management บนหนึ่งใน คลัสเตอร์โหนด ให้ล็อกอินเข้าสู่ โหนดและป้อน:  

```
haemtrcoff -s emsvcs -a all
```
- ในการปิดทำงานการติดตามการกำหนดค่าเริ่มต้นและการกำหนดค่าสำหรับระบบย่อย Event Management บนคลัสเตอร์ โหนด ให้ล็อกอินเข้าสู่โหนดและป้อน:  

```
haemtrcoff -s emsvcs -a init,config
```

## Location

```
/usr/sbin/rsct/bin/haemtrcoff
```

ตำแหน่งของคำสั่ง **haemtrcoff**

## ไฟล์

```
/var/ha/log/em.trace.cluster_name
```

มีบันทึกการติดตามของ **haemd** daemon บนคลัสเตอร์ชื่อ *cluster\_name*

```
/var/ha/log/em.msgtrace.cluster_name
```

มีเอาต์พุตการติดตามข้อความจาก Event Manager daemon บนคลัสเตอร์ ชื่อ *cluster\_name*

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง haemtrcon”

“haemd Daemon” ในหน้า 755

“คำสั่ง emsvcsctrl” ในหน้า 379

---

## คำสั่ง haemtrcon

### วัตถุประสงค์

เปิดทำการการติดตามสำหรับ daemon ตัวจัดการเหตุการณ์

### ไวยากรณ์

```
haemtrcon -s subsys_name -a trace_list
```

### คำอธิบาย

คำสั่ง **haemtrcon** ใช้เพื่อเปิดทำงานการติดตามกิจกรรม ที่ระบุของ daemon ตัวจัดการเหตุการณ์ เอาต์พุตการติดตามจะอยู่ในบันทึกการติดตาม การจัดการเหตุการณ์สำหรับพาร์ติชันระบบ เมื่อใช้งาน พารามิเตอร์ **regs**, **dinsts**, **iolists** และ **olists** จะดำเนินการติดตาม ครั้งเดียว ข้อมูลที่ระบุจะอยู่ในบันทึกการติดตาม แต่ไม่มีการดำเนินการติดตาม เพิ่มเติม

### แฟล็ก

**-s** *cluster\_name*

ระบุชื่อของระบบย่อยการจัดการเหตุการณ์ บนโหนด *cluster\_name* คือ **emsvcs** แฟล็กและพารามิเตอร์นี้ต้องถูกระบุ

**-a** *trace\_list*

ระบุรายการของพารามิเตอร์การติดตาม แต่ละพารามิเตอร์จะระบุชนิดของกิจกรรมที่การติดตามจะถูกเปิดทำงาน ต้องมีการระบุอย่างน้อย หนึ่งพารามิเตอร์ หากระบุมากกว่าหนึ่งพารามิเตอร์ พารามิเตอร์ต้องถูก คั่นด้วยเครื่องหมายจุลภาค รายการจะต้องไม่มีช่องว่าง

### พารามิเตอร์

พารามิเตอร์การติดตามต่อไปนี้สามารถระบุได้:

**init** ติดตามการกำหนดค่าเริ่มต้นของ daemon ตัวจัดการเหตุการณ์

**config** ดัมพ์ข้อมูลจากไฟล์คอนฟิกูเรชัน

**insts** ติดตามอินสแตนซ์ตัวแปรรีซอร์สที่จัดการโดย daemon

**rmctrl** ติดตามการควบคุมรีซอร์สมอนิเตอร์

**cci** ติดตามอินเตอร์เฟซการสื่อสารไคลเอ็นต์ (ภายใน)

<b>emp</b>	ติดตามโปรโตคอลตัวจัดการเหตุการณ์
<b>obsv</b>	ติดตามการสังเกตตัวแปรรีซอร์ส
<b>evgn</b>	ติดตามการสร้างและการแจ้งเตือนเหตุการณ์
<b>reg</b>	ติดตามการลงทะเบียนและการถอนการลงทะเบียนเหตุการณ์
<b>pci</b>	ติดตามอินเตอร์เฟซการสื่อสารทางเพียร์ (ภายใน)
<b>msgs</b>	ติดตามข้อความทั้งหมดที่เข้ามายัง และถูกเรียกใช้จาก daemon
<b>เคียวรี</b>	ติดตามเคียวรีที่ถูกจัดการโดย daemon
<b>gsi</b>	ติดตามอินเตอร์เฟซการให้บริการกลุ่ม (ภายใน)
<b>eval</b>	ติดตามการประเมินค่านิพจน์
<b>rdi</b>	ติดตามอินเตอร์เฟซ daemon ที่เชื่อถือได้ (ภายใน)
<b>sched</b>	ติดตามตัวกำหนดตารางเวลาภายใน
<b>shm</b>	ติดตามกิจกรรมการจัดการหน่วยความจำที่แบ่งใช้
<b>ทั้งหมด</b>	ติดตามกิจกรรมทั้งหมด

#### **all\_but\_msgs**

หยุดการติดตามกิจกรรมทั้งหมดยกเว้นข้อความ กิจกรรมข้อความ ถูกกำหนดโดยอาร์กิวเมนต์ msgs

<b>regs</b>	ติดตามเหตุการณ์ที่ลงทะเบียนในขณะนี้
<b>dinsts</b>	ติดตามอินสแตนซ์ตัวแปรรีซอร์สทั้งหมดที่ daemon รู้จัก
<b>iolists</b>	ติดตามรายการการสังเกตในทันที
<b>olists</b>	ติดตามรายการการสังเกต

#### **ข้อจำกัด**

อย่างใช้คำสั่งนี้ระหว่างดำเนินการปกติ ใช้คำสั่งนี้ภายใต้คำแนะนำของ IBM Support Center เท่านั้น ซึ่งจะให้ข้อมูล เพื่อวัตถุประสงค์ในการดีบั๊ก และอาจลดผลการทำงานของระบบย่อยการจัดการ เหตุการณ์ลง หรืออาจมีสิ่งอื่นที่กำลังรันในพาร์ติชันระบบ

#### **ข้อมูลจำเพาะการนำไปใช้งาน**

คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ Reliable Scalable Cluster Technology (RSCT)

#### **ตัวอย่าง**

1. ในการเปิดทำงานการติดตามระบบย่อยการจัดการเหตุการณ์บนหนึ่งใน คลัสเตอร์โหนด ให้ล็อกอินเข้าสู่โหนดและป้อน:  

```
haemtrcon -s emsvcs -a all
```
2. ในการเปิดทำงานการติดตามการกำหนดค่าเริ่มต้นและการกำหนดค่าสำหรับระบบย่อยการจัดการ เหตุการณ์บนคลัสเตอร์โหนด ให้ล็อกอินเข้าสู่โหนดและป้อน:

```
haemtrcon -s emsvcs -a init,config
```

## Location

/usr/sbin/rsct/bin/haemtrcon

ตำแหน่งของคำสั่ง **haemtrcon**

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง haemtrcoff” ในหน้า 761

“haemd Daemon” ในหน้า 755

“คำสั่ง emsvcsctrl” ในหน้า 379

---

## คำสั่ง haemunlkrm

### วัตถุประสงค์

ปลดล็อกและเริ่มทำงานรีซอร์สมอนิเตอร์

### ไวยากรณ์

```
haemunlkrm -s subsys_name -a resmon_name
```

### คำอธิบาย

หาก daemon การจัดการเหตุการณ์ไม่สามารถเริ่มทำงานรีซอร์สมอนิเตอร์ได้สำเร็จ หลังจากพยายามสามครั้งภายในช่วงเวลาสองชั่วโมง หรือหาก daemon ทำการเชื่อมต่อกับอินสแตนซ์ของรีซอร์สมอนิเตอร์ได้สำเร็จ  $n$  ครั้งภายในช่วงเวลาสองชั่วโมง รีซอร์สมอนิเตอร์จะถูก “ล็อก” และไม่มี ความพยายามใดที่จะเริ่มทำงาน หรือเชื่อมต่อกับอินสแตนซ์ใดๆ  $n$  คือ 3 ในคลัสเตอร์ HACMP/ES เมื่อสาเหตุของความล้มเหลวได้รับการนำไปวิเคราะห์ และแก้ไขปัญหาลแล้ว สามารถใช้คำสั่ง **haemunlkrm** เพื่อปลดล็อกรีซอร์สมอนิเตอร์และพยายามเริ่มทำงาน หรือเชื่อมต่อกับอินสแตนซ์ของรีซอร์สมอนิเตอร์

สถานะของ daemon ตัวจัดการเหตุการณ์ ดังที่แสดงโดยคำสั่ง `lssrc` ระบุว่ารีซอร์สมอนิเตอร์ถูกล็อกหรือไม่

### แฟล็ก

`-s subsys_name`

ระบุชื่อของระบบย่อยการจัดการเหตุการณ์ บนโหนด *subsys\_name* คือ **emsvcs** แฟล็กและพารามิเตอร์นี้ต้องถูกระบุ

`-a resmon_name`

ระบุชื่อของรีซอร์สมอนิเตอร์เพื่อปลดล็อกและเริ่มทำงาน

### พารามิเตอร์

พารามิเตอร์การติดตามต่อไปนี้อาจระบุได้:

**init** ติดตามการกำหนดค่าเริ่มต้นของ daemon ตัวจัดการเหตุการณ์

**config** ดัมพ์ข้อมูลจากไฟล์คอนฟิกูเรชัน

**insts** ติดตามอินสแตนซ์ตัวแปรรีซอร์สที่จัดการโดย daemon  
**rmctrl** ติดตามการควบคุมรีซอร์สมอนิเตอร์  
**cci** ติดตามอินเตอร์เฟซการสื่อสารไคลเอ็นต์ (ภายใน)  
**emp** ติดตามโปรโตคอลตัวจัดการเหตุการณ์  
**obsv** ติดตามการสังเกตตัวแปรรีซอร์ส  
**evgn** ติดตามการสร้างและการแจ้งเตือนเหตุการณ์  
**reg** ติดตามการลงทะเบียนและการถอนการลงทะเบียนเหตุการณ์  
**pci** ติดตามอินเตอร์เฟซการสื่อสารทางเพียร์ (ภายใน)  
**msgs** ติดตามข้อความทั้งหมดที่เข้ามา และถูกเรียกใช้จาก daemon  
**เคียวรี** ติดตามเคียวรีที่ถูกจัดการโดย daemon  
**gsi** ติดตามอินเตอร์เฟซการให้บริการกลุ่ม (ภายใน)  
**eval** ติดตามการประเมินค่านิพจน์  
**rdi** ติดตามอินเตอร์เฟซ daemon ที่เชื่อถือได้ (ภายใน)  
**sched** ติดตามตัวกำหนดตารางเวลาภายใน  
**shm** ติดตามกิจกรรมการจัดการหน่วยความจำที่แบ่งใช้

### ทั้งหมด

ติดตามกิจกรรมทั้งหมด

### all\_but\_msgs

หยุดการติดตามกิจกรรมทั้งหมดยกเว้นข้อความ กิจกรรมข้อความ ถูกกำหนดโดยอาร์กิวเมนต์ msgs

**regs** ติดตามเหตุการณ์ที่ลงทะเบียนในขณะนี้  
**dinsts** ติดตามอินสแตนซ์ตัวแปรรีซอร์สทั้งหมดที่ daemon รู้จัก  
**iolists** ติดตามรายการการสังเกตในทันที  
**olists** ติดตามรายการการสังเกต

## ความปลอดภัย

คุณต้องมีสิทธิ์ root และการเข้าถึงเพื่อเขียนใน SDR เพื่อรันคำสั่งนี้

คุณควรกำลังรันอยู่บนเวิร์กสเตชันการควบคุม ก่อนรันคำสั่งนี้ คุณต้องตั้งค่าตัวแปรสถานะแวดล้อม SP\_NAME ให้เป็นชื่อพาร์ติชันระบบที่เหมาะสม

## สถานะออก

- 0 บ่งชี้ความสำเร็จของคำสั่ง
- 1 หมายถึงมีข้อผิดพลาดเกิดขึ้น โดยมีข้อความแสดงความผิดพลาดอย่างน้อยหนึ่ง ข้อความที่ระบุถึงสาเหตุของข้อผิดพลาด

## ข้อจำกัด

อย่างไร้คำสั่งนี้ระหว่างการดำเนินการปกติใช้คำสั่งนี้ภายใต้คำแนะนำของ IBM Support Center เท่านั้น ซึ่งจะให้ข้อมูลเพื่อวัตถุประสงค์ในการดีบั๊ก และอาจลดผลการทำงานของระบบย่อยการจัดการ เหตุการณ์ลง หรืออาจมีสิ่งอื่นที่กำลังรันในพาร์ติชันระบบ

## เอาต์พุตมาตรฐาน

เมื่อคำสั่งดำเนินงานสำเร็จเรียบร้อย จะเขียนข้อความแสดงข้อมูล ต่อไปนี้:

การอ่านข้อมูล การจัดการเหตุการณ์ สำหรับพาร์ติชัน *syspar\_name*

*CDB=new\_EMADB\_file\_name* *Version=EMADB\_version\_string*

## ข้อผิดพลาดมาตรฐาน

คำสั่งนี้เขียนข้อความแสดงความผิดพลาด (ตามความจำเป็น) ไปที่ข้อผิดพลาดมาตรฐาน

## ตัวอย่าง

1. ตัวอย่างนี้ใช้กับการปลดล็อกรีซอร์สมอนิเตอร์บนไหนด

หากเอาต์พุตของคำสั่ง *lssrc* ระบุว่าโปรแกรมรีซอร์สมอนิเตอร์ **IBM.PSSP.harmpd** ถูกบล็อก ให้แก้ไขสถานะที่ทำให้รีซอร์สมอนิเตอร์ไม่สามารถเริ่มทำงานได้ และป้อน:

```
haemunlkrm -s emsvcs -a IBM.PSSP.harmpd
```

## Location

*/usr/sbin/rsct/bin/haemunlkrm*

ตำแหน่งของคำสั่ง **haemunlkrm**

## ไฟล์

*/var/ha/log/em.trace.cluster\_name*

มีบันทึกการติดตามของ haemd daemon บนคลัสเตอร์ชื่อ *cluster\_name*

*/var/ha/log/em.msgtrace.cluster\_name*

มีเอาต์พุตการติดตามข้อความจาก daemon ตัวจัดการเหตุการณ์บนคลัสเตอร์ชื่อ *cluster\_name*

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง *haemtrcoff*” ในหน้า 761

“haemd Daemon” ในหน้า 755

“คำสั่ง *emsvcsctrl*” ในหน้า 379

---

## hagsd Daemon

### วัตถุประสงค์

เฝ้าดูอินสแตนซ์ตัวแปรรีซอร์สที่ถูกอัปเดตโดยรีซอร์สมอนิเตอร์ และสร้างและรายงานเหตุการณ์ไปยังไคลเอ็นต์โปรแกรม

## ไวยากรณ์

`hagsd [-a] [-s] [-k] [-d] [-c] [-u] [-t] [-o] [-r] [-h] daemon_name`

## คำอธิบาย

`hagsd daemon` เป็นส่วนหนึ่งของระบบย่อยการให้บริการกลุ่ม ซึ่งจัดให้มีโปรแกรมอำนวยความสะดวกประสงค์สำหรับการประสานและการมอนิเตอร์การเปลี่ยนแปลง ในสถานะของแอปพลิเคชันที่กำลังรันบนโหนดของคลัสเตอร์ `daemon` นี้มีบริการส่วนใหญ่ของระบบย่อย `daemon_name` ระบุชื่อที่ใช้โดย `daemon` เพื่อตั้งชื่อไฟล์บันทึกการทำงานและระบุข้อความในบันทึกข้อผิดพลาด AIX

อินสแตนซ์หนึ่งของ `hagsd daemon` เรียกใช้งานบนแต่ละคลัสเตอร์โหนด `hagsd daemon` อยู่ภายใต้การควบคุมของ system resource controller (SRC)

เนื่องจาก `daemon` อยู่ภายใต้การควบคุมของ SRC จึงไม่ควรเริ่มทำงานจากบรรทัดคำสั่ง โดยตรง โดยปกติถูกเรียกโดยคำสั่ง `grpsvcctrl` ซึ่งในทางกลับกันถูกเรียกโดยกระบวนการเริ่มทำงานคลัสเตอร์ หากคุณต้องเริ่มทำงาน หรือหยุดทำงาน `daemon` โดยตรง ให้ใช้คำสั่ง `startsrc` หรือ `stopsrc`

## แฟล็ก

- a เพิ่มระบบย่อย
- s เริ่มทำงานระบบย่อย
- k หยุดทำงานระบบย่อย
- d ลบบระบบย่อย
- c ล้างค่าระบบย่อย นั่นคือ ลบบระบบย่อยออกจากพาร์ติชันระบบทั้งหมด
- u ไม่กำหนดค่าระบบย่อยจากพาร์ติชันระบบทั้งหมด
- t เปิดการติดตามสำหรับระบบย่อย
- o ปิดการติดตามสำหรับระบบย่อย
- r รีเฟรชระบบย่อย
- h แสดงข้อมูลการใช้งาน

## พารามิเตอร์

`daemon_name`

ระบุชื่อที่ใช้โดย `daemon` เพื่อตั้งค่าไฟล์บันทึกการทำงานและระบุข้อความในบันทึกข้อผิดพลาด AIX

## ความปลอดภัย

คุณต้องมีสิทธิ์ `root` ในการรันสคริปต์นี้

## สถานะออก

- 0 บ่งชี้ความสำเร็จของคำสั่ง

1 หมายถึงมีข้อผิดพลาดเกิดขึ้น

## ข้อจำกัด

คำสั่งนี้ใช้ได้เฉพาะสถานะแวดล้อม PSSP เท่านั้น

## เอาต์พุตมาตรฐาน

เมื่อระบุแฟล็ก -h ไว้ข้อความการใช้งานของคำสั่งนี้ ถูกเขียนลงในเอาต์พุตมาตรฐาน

## ข้อผิดพลาดมาตรฐาน

คำสั่งนี้เขียนข้อความแสดงความผิดพลาด (ตามความจำเป็น) ไปที่ข้อผิดพลาดมาตรฐาน

## ตัวอย่าง

1. ในการเพิ่มระบบย่อยการให้บริการกลุ่มกับ SRC ในพาร์ติชันระบบ ปัจจุบันให้ตั้งค่าตัวแปรสถานะแวดล้อม SP\_NAME เป็นชื่อพาร์ติชันระบบที่เหมาะสมให้ป้อน:

```
hagsctrl -a
```

2. ในการเริ่มทำงานระบบย่อยการให้บริการกลุ่มในพาร์ติชันระบบ ปัจจุบันให้ตั้งค่าตัวแปรสถานะแวดล้อม SP\_NAME เป็นชื่อพาร์ติชันระบบที่เหมาะสมให้ป้อน:

```
hagsctrl -s
```

3. ในการหยุดทำงานระบบย่อยการให้บริการกลุ่มในพาร์ติชันระบบ ปัจจุบันให้ตั้งค่าตัวแปรสถานะแวดล้อม SP\_NAME เป็นชื่อพาร์ติชันระบบที่เหมาะสมให้ป้อน:

```
hagsctrl -k
```

4. ในการลบระบบย่อยการให้บริการกลุ่มออกจาก SRC ในพาร์ติชันระบบ ปัจจุบันให้ตั้งค่าตัวแปรสถานะแวดล้อม SP\_NAME เป็นชื่อพาร์ติชันระบบที่เหมาะสมให้ป้อน:

```
hagsctrl -d
```

5. ในการล้างค่าระบบย่อยการให้บริการกลุ่มบนพาร์ติชันระบบทั้งหมด ให้ป้อน:

```
hagsctrl -c
```

6. ในการไม่กำหนดค่าระบบย่อยการให้บริการกลุ่มในพาร์ติชันระบบทั้งหมด บนเวิร์กสเตชันการควบคุมให้ป้อน:

```
hagsctrl -u
```

7. ในการเปิด daemon การให้บริการกลุ่มในพาร์ติชันระบบ ปัจจุบันให้ตั้งค่าตัวแปรสถานะแวดล้อม SP\_NAME เป็นชื่อพาร์ติชันระบบที่เหมาะสมให้ป้อน:

```
hagsctrl -t
```

8. ในการปิด daemon การให้บริการกลุ่มในพาร์ติชันระบบ ปัจจุบันให้ตั้งค่าตัวแปรสถานะแวดล้อม SP\_NAME เป็นชื่อพาร์ติชันระบบที่เหมาะสมให้ป้อน:

```
hagsctrl -o
```

## Location

/usr/sbin/rsct/bin/hagsd

มี hagsd daemon

## ไฟล์

`/var/ha/log/hags_nodenum_instnum.syspar_name`

มีบันทึกการทำงานของ **hagsd** daemons บนโหนด

`/var/ha/log/hags.syspar_name_nodenum_instnum.syspar_name`

มีบันทึกการทำงานสำหรับแต่ละ **hagsd** daemon บนเวิร์กสเตชันการควบคุม

ชื่อไฟล์ประกอบด้วยตัวแปรต่อไปนี้:

- *nodenum* คือหมายเลขโหนดที่ daemon กำลังรัน
- *instnum* คือหมายเลขอินสแตนซ์ของ daemon
- *syspar\_name* คือชื่อของพาร์ติชันระบบที่ daemon กำลังรัน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `grpsvcctrl`” ในหน้า 743

---

## คำสั่ง **hagsns**

### วัตถุประสงค์

รับข้อมูลเซิร์ฟเวอร์ชื่อเซอวิริส กลุ่ม

### ไวยากรณ์

`hagsns [-h host] [-c] -g group_name`

`hagsns [-h host] [-c] -s subsystem_name`

`hagsns [-h host] [-c] -p subsystem_pid`

### คำอธิบาย

ใช้คำสั่ง **hagsns** เพื่อเคียวิริสสถานะของ nameserver การให้บริการกลุ่ม

### แฟล็ก

`-c` บังคับให้เอาต์พุตเป็น "English\_only" ถ้าแฟล็ก `-c` ไม่ถูกระบุ โลแคลของ daemon จะถูกใช้ สำหรับเอาต์พุต

`-g group_name`

ระบุกลุ่มของระบบย่อยเพื่อขอรับสถานะ คำสั่งจะไม่สำเร็จ หากตัวแปร *group\_name* ไม่มีอยู่ในอ็อบเจกต์คลาสระบบย่อย

`-h host` ระบุโฮสต์ที่จะรับค่าสถานะเซิร์ฟเวอร์ชื่อ

`-p subsystem_pid`

ระบุอินสแตนซ์เฉพาะของ *subsystem\_pid* เพื่อรับ ค่าสถานะเซิร์ฟเวอร์ชื่อ

`-s subsystem_name`

ระบบย่อยเพื่อขอรับสถานะ ตัวแปร `subsystem_name` สามารถเป็นชื่อระบบย่อยจริง หรือชื่อพ้องสำหรับระบบย่อย คำสั่ง จะทำไม่สำเร็จหากตัวแปร `subsystem_name` ไม่มีอยู่ใน อ็อบเจกต์คลาสระบบย่อย

## พารามิเตอร์

`daemon_name`

ระบุชื่อที่ใช้โดย daemon เพื่อตั้งค่าไฟล์บันทึกการทำงานและระบุข้อความในบันทึกข้อผิดพลาด AIX

## ความปลอดภัย

คุณต้องมีสิทธิ์การใช้งาน `root` เพื่อรันคำสั่งนี้

## สถานะออก

0 บ่งชี้ว่า คำสั่งได้ดำเนินการเป็นผลสำเร็จแล้ว

ค่าไม่เป็นศูนย์

หมายถึงมีข้อผิดพลาดเกิดขึ้น

## ข้อจำกัด

คำสั่งนี้ใช้ได้เฉพาะในสภาวะแวดล้อม PSSP เท่านั้น

## เอาต์พุตมาตรฐาน

เมื่อระบุแฟล็ก `-h` ไว้ข้อความการใช้งานของคำสั่งนี้ ถูกเขียนลงในเอาต์พุตมาตรฐาน

## ข้อผิดพลาดมาตรฐาน

คำสั่งนี้เขียนข้อความแสดงความผิดพลาด ตามความจำเป็น ไปที่ข้อผิดพลาดมาตรฐาน

## ตัวอย่าง

ในการรับข้อมูลจากระบบย่อยการให้บริการกลุ่มให้ป้อน:

```
hagsns -c -s cthags
```

หรือ

```
hagsns -s cthags
```

เอาต์พุตจะมีลักษณะดังนี้:

```
HA GS NameServer Status
NodeID=1.16, pid=14460, domainID=6.14, NS established, CodeLevel=GSLevel(DRL=8)
NS state=kCertain, protocolInProgress=kNoProtocol, outstandingBroadcast=KNoBcast
Process started on Jun 19 18:34:20, (10d 20:19:22) ago, HB connection took (19:14:9).
Initial NS certainty on Jun 20 13:48:45, (10d 1:4:57) ago, taking (0:0:15).
Our current epoch of Jun 23 13:05:19 started on (7d 1:48:23), ago.
Number of UP nodes: 12
List of UP nodes: 0 1 5 6 7 8 9 11 17 19 23 26
```

ในตัวอย่างนี้ domainID=6.14 หมายความว่าไหนด 6 เป็นไหนดเซิร์ฟเวอร์ชื่อ (NS) ID โดเมนประกอบด้วยหมายเลขไหนด และหมายเลขการเกิด หมายเลขการเกิดเป็นเลขจำนวนเต็ม ที่ถูกเพิ่มค่าเมื่อเริ่มทำงาน daemon การให้บริการกลุ่ม NS established หมายความว่าเซิร์ฟเวอร์ชื่อ ถูกสร้างขึ้น

## Location

/usr/sbin/rsct/bin/hagsns

มีคำสั่ง hagsns

## ไฟล์

/var/ha/log/hags\_nodenum\_instnum.syspar\_name

มีบันทึกการทำงานของ hagsd daemons บนไหนด

/var/ha/log/hags.syspar\_name\_nodenum\_instnum.syspar\_name

มีบันทึกการทำงานสำหรับแต่ละ hagsd daemon บนเวิร์กสเตชันการควบคุม

ชื่อไฟล์ประกอบด้วยตัวแปรต่อไปนี้:

- *nodenum* คือหมายเลขไหนดที่ daemon กำลังรัน
- *instnum* คือหมายเลขอินสแตนซ์ของ daemon
- *syspar\_name* คือชื่อของพาร์ติชันระบบที่ daemon กำลังรัน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง hagsvote”

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง lssrc

คำสั่ง nlssrc

---

## คำสั่ง hagsvote

### วัตถุประสงค์

รับข้อมูลโหวตสำหรับ กลุ่มเซอร์วิส

### ไวยากรณ์

**hagsvote** [-h *host*] [-l] [-a *argument*] [-c] -g *group\_name*

**hagsvote** [-h *host*] [-l] [-a *argument*] [-c] -s *subsystem\_name*

**hagsvote** [-h *host*] [-l] [-a *argument*] [-c] -p *subsystem\_pid*

### คำอธิบาย

ใช้คำสั่ง hagsvote เพื่อเคียวรีสถานะของโปรโตคอลการโหวต สำหรับการให้บริการกลุ่ม

## แฟล็ก

- a ระบุชื่อกลุ่มการให้บริการกลุ่ม ชื่อกลุ่มนี้ต่าง จากชื่อของแฟล็ก -g ในกรณีนี้ กลุ่มถูกสร้างจากการเรียกใช้ครั้งแรกของไคลเอ็นต์เพื่อรวมโปรโตคอล
- c ร้องขอเอาต์พุตแท้จริงของข้อมูลการไหลต่อการให้บริการกลุ่ม เอาต์พุตถูกแสดงเป็นภาษาอังกฤษไม่ว่าไคลเอ็นต์ภาษาที่ติดตั้งจะเป็นภาษาใด หากไม่ระบุ -c ไคลเอ็นต์ของ daemon จะถูกใช้สำหรับ เอาต์พุต
- g *group\_name*  
ระบุกลุ่มของระบบย่อยเพื่อขอรับสถานะ คำสั่งจะไม่สำเร็จ หากตัวแปร *group\_name* ไม่มีอยู่ในอ็อบเจกต์คลาสระบบย่อย
- h *host* ระบุชื่อโฮสต์ที่จะรับค่าสถานะ
- l ร้องขอเอาต์พุตรายละเอียดในรูปแบบ “ยาว”
- p *subsystem\_pid*  
ระบุอินสแตนซ์เฉพาะของตัวแปร *subsystem\_pid* เพื่อรับค่าการไหล
- s *subsystem\_name*  
ระบุระบบย่อยเพื่อไหล ตัวแปร *subsystem\_name* สามารถเป็นชื่อระบบย่อยจริง หรือชื่อพ้องสำหรับระบบย่อย คำสั่งจะไม่สำเร็จหากตัวแปร *subsystem\_name* ไม่มีอยู่ใน อ็อบเจกต์คลาสระบบย่อย

## พารามิเตอร์

*daemon\_name*

ระบุชื่อที่ใช้โดย daemon เพื่อตั้งค่าไฟล์บันทึกการทำงานและระบุข้อความในบันทึกข้อผิดพลาด AIX

## ความปลอดภัย

คุณต้องมีสิทธิ์ **root** ในการรันคำสั่งนี้

### สถานะออก

0 บ่งชี้ความสำเร็จของคำสั่ง

### ไม่เป็นศูนย์

หมายถึงมีข้อผิดพลาดเกิดขึ้น

## ข้อจำกัด

คำสั่งนี้ใช้ได้เฉพาะโหมด PSSP เท่านั้น

## เอาต์พุตมาตรฐาน

คำสั่งนี้เขียนข้อความแสดงความคิดเห็น (ตามความจำเป็น) ไปที่ข้อผิดพลาดมาตรฐาน

## ข้อผิดพลาดมาตรฐาน

คำสั่งนี้เขียนข้อความแสดงความคิดเห็น ตามความจำเป็น ไปที่ข้อผิดพลาดมาตรฐาน

## ตัวอย่าง

1. ในการดูข้อมูลเกี่ยวกับสถานะของโปรโตคอลการโหวตสำหรับกลุ่ม `theSourceGroup` ในแบบยาว ให้ป้อน:

```
hagsvote -ls cthags -a theSourceGroup (locale-dependent)
```

เอาต์พุตจะมีลักษณะดังนี้:

```
Number of groups: 4
Group name [theSourceGroup] GL node [26] voting data:
GL in phase [1] of n-phase protocol of type [Join].
Local voting data:
Number of providers: 1
Number of providers not yet voted: 1 (vote not submitted).
Given vote: [No vote value] Default vote: [No vote value]
ProviderID Voted? Failed? Conditional?
[101/26] No No Yes
Global voting data:
Number providers not yet voted: 1
Given vote: [No vote value] Default vote: [No vote value]
Nodes that have voted: []
Nodes that have not voted: [26]
```

The first line of the output means that the total number of groups is 4. The second line provides the group name and the group leader node (in this case 26). The remaining lines give the voting data:

- The group leader is in phase 1 of a n-phase protocol.
- The protocol is the Join protocol.
- For the local node, it has 1 provider, the number of providers which have not voted yet is 1.
- No default vote value is given and no vote value is given.
- Under the line "ProviderID Voted? Failed? Conditional?," "[101/16] No No Yes," means that the provider ID is 101/26, not voted yet, not failed, but wait for the vote (so it is conditional).

The output then shows the global voting status:

- The number of providers that have not voted yet is 1.
- No vote value given yet, no default vote value.
- The nodes that have voted is none.
- The nodes that have not voted is node 26.

2. In the following example, the meaning of each line of output is the same as in the first example except that node 26 is the group leader node.

```
hagsvote -ls cthags -a theSourceGroup -c (canonical form)
```

เอาต์พุตจะมีลักษณะดังนี้:

```
Number of groups: 4
Group Name: theSourceGroup
GL Node: 26 (I am GL)
Current phase number of an n-phase protocol: 1
Protocol name: [Join]
Local voting data:
Number of local providers: 1
Number of local providers not yet voted: 1 (vote not submitted)
Given vote: [No vote value] Default vote: [No vote value]Global voting data:
```

Number of nodes in group: 1  
Number of global providers not yet voted: 1  
Given vote: [No vote value] Default vote: [No vote value]  
Nodes that have voted: []  
Nodes that have not voted: [26]

## Location

/usr/sbin/rsct/bin/hagsvote

มีคำสั่ง **hagsvote**

## ไฟล์

/var/ha/log/hags\_nodenum\_instnum.syspar\_name

มีบันทึกการทำงานของ **hagsd** daemons บนโหนด

/var/ha/log/hags.syspar\_name\_nodenum\_instnum.syspar\_name

มีบันทึกการทำงานสำหรับแต่ละ **hagsd** daemon บนเวิร์กสเตชันการควบคุม

ชื่อไฟล์ประกอบด้วยตัวแปรต่อไปนี้:

- *nodenum* คือหมายเลขโหนดที่ daemon กำลังรัน
- *instnum* คือหมายเลขอินสแตนซ์ของ daemon
- *syspar\_name* คือชื่อของพาร์ติชันระบบที่ daemon กำลังรัน

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง **hagsns**” ในหน้า 771

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **lssrc**

คำสั่ง **nlssrc**

---

## คำสั่ง **halt** หรือ **fasthalt**

### วัตถุประสงค์

หยุดทำงานตัวประมวลผล

### ไวยากรณ์

```
{ halt | fasthalt } [-l] [-n] [-p] [-q] [-y]
```

### คำอธิบาย

คำสั่ง **halt** เขียนข้อมูล ไปยังดิสก์ จากนั้นหยุดทำงานตัวประมวลผล เครื่องไม่รีสตาร์ท เฉพาะผู้ใช้ **root** เท่านั้นที่สามารถรันคำสั่งนี้ ห้ามใช้คำสั่งนี้หากผู้ใช้อื่นล็อกอินอยู่ในระบบ หากไม่มีผู้ใช้อื่น ๆ ล็อกอิน คำสั่ง **halt** สามารถใช้ได้ ใช้คำสั่ง **halt** หากคุณไม่ได้รีสตาร์ทเครื่องในทันที เมื่อข้อความ **....Halt completed....** จะถูกแสดง คุณสามารถปิดเครื่อง

คำสั่ง **halt** จะบันทึกการปิดระบบโดยใช้คำสั่ง **syslogd** และวางเร็กคอร์ดของการปิดระบบในไฟล์แอดเคาต์การล็อกอิน `/var/adm/wtmp` ระบบยังเขียนรายการลงใน บันทึกข้อผิดพลาดที่แสดงว่าระบบปิดการทำงาน

คำสั่ง **fasthalt** หยุดทำงาน ระบบโดยการเรียกใช้คำสั่ง **halt** คำสั่ง **fasthalt** จัดให้มีความเข้ากันได้กับ BSD

## แฟล็ก

ไอเท็ม คำอธิบาย

- l อย่านบันทึกการทำงานการหยุดชะงักในไฟล์แอดเคาต์ แฟล็ก -l ไม่ได้หยุดทำงานการอัปเดตไฟล์แอดเคาต์ แฟล็ก -n และ -q แสดงถึงแฟล็ก -l flag.
- n ป้องกัน **sync** ก่อนที่จะหยุดการทำงาน
- p หยุดชะงักระบบโดยไม่ปิดเครื่อง

หมายเหตุ: แฟล็ก -p ไม่มีผลใดๆ หากใช้ร่วมกับแฟล็กที่ไม่ต้องการการหยุด แบบถาวร Power ยัง ปิดอยู่หากโอเปอร์เรตติ้งอื่นร้องขอให้หน่วงเวลา power-on และรีสตาร์ท

- q ทำให้เกิดการหยุดชะงักอย่างรวดเร็ว

Notes:

- การรันคำสั่ง **halt** โดยมีแฟล็ก -q จะไม่เรียกใช้ **sync** ดังนั้นระบบจะหยุดในทันที
  - หากคุณรันคำสั่ง **halt** พร้อมกับแฟล็ก -q ในเวิร์กโพลิตาร์ตชัน (WPAR) คำสั่ง **halt** สามารถหยุดการทำงานของ WPAR และทำให้อยู่ใน สภาวะ D (defined) WPAR อาจไม่หยุดการทำงานอย่างสมบูรณ์ และทำให้ WPAR อยู่ในสภาวะ T (transitional) เนื่องจากเงื่อนไขการหมดเวลาหรือความล่าช้าที่เกิดขึ้นขณะการยกเลิกการเม้าท์ ระบบไฟล์
- y หยุดระบบจากการดำเนินการเรียกเลขหมาย

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

1. เมื่อต้องการหยุดการทำงานระบบโดยไม่มีกรบันทึกการหยุดในไฟล์แอดเคาต์ให้ป้อนคำสั่งต่อไปนี้:

```
halt -l
```

2. เมื่อต้องการหยุดการทำงานระบบอย่างรวดเร็วให้ป้อนคำสั่งต่อไปนี้:

```
halt -q
```

3. เมื่อต้องการหยุดระบบจากการดำเนินการ dial-up ให้ป้อนคำสั่งต่อไปนี้:

```
halt -y
```

## ไฟล์

ไอเท็ม  
/etc/rc  
/var/adm/wtmp

คำอธิบาย  
ระบุสคริปต์การเริ่มต้นทำงานกับระบบ  
ระบุไฟล์แอดเคาต์การล็อกอิน

## ข้อมูลที่เกี่ยวข้อง:

คำสั่ง fastboot

คำสั่ง shutdown

คำสั่ง sync

คำสั่ง syslogd

---

## คำสั่ง hangman

### วัตถุประสงค์

เริ่มทำงานเกมการเดาคำศัพท์แฉงค์แมน (hangman)

### ไวยากรณ์

hangman [ *File* ]

### คำอธิบาย

คำสั่ง **hangman** เลือก คำที่มีอย่างน้อยเจ็ดตัวอักษรจากพจนานุกรมมาตรฐาน พารามิเตอร์ *File* จะระบุพจนานุกรมทางเลือก คุณเดาคำ โดยการเดาตัวอักษรครั้งละหนึ่งตัว คุณได้รับอนุญาตให้เดาผิดได้เจ็ดครั้ง

เมื่อคุณเริ่มเล่นเกมแฉงค์แมน เกมจะแสดง:

```
guesses: word: ..... errors: 0/7
```

```
guess:
```

guesses แสดง ตัวอักษรที่คุณใช้เป็นคำที่เดา ทุกๆ ตัวอักษรที่คุณเดาจะแสดงรายการหลัง guesses word: ..... แสดง จำนวนตัวอักษรในคำปริศนา ในกรณีนี้จะมี . (จุด) เจ็ดจนเพื่อให้แทนตัวอักษรเจ็ดตัว ในคำ เมื่อคุณเดาตัวอักษรได้ถูกต้อง เกมจะแทนที่ . ที่เหมาะสม ด้วยตัวอักษรที่ถูกต้อง errors: 0/7 แสดงจำนวนที่เดาไม่ถูก คุณป้อน ตัวอักษรที่คุณทายที่พร้อมท์ guess: ตัวอย่างเช่น:

```
guesses: word: ..... errors: 0/7
```

```
guess: q
```

```
guesses: q word: ..... errors: 1/7
```

```
guess: a
```

```
guesses: aq word: .a....a... errors: 1/7
```

```
guess: b
```

```
guesses: abq word: .a....a... errors 2/7
```

```
guess: j
```

```
guesses: abjq word: .a....a... errors: 3/7
```

```
guess: s
```

```
guesses: abjqs word: .a....a..s errors: 3/7
```

```
guess: z
```

```
guesses: abjqsz word: .a....a..s errors: 4/7
```

```
guess: y
```

```
guesses: abjqsyz word: .a....a..s errors: 5/7
guess: k
guesses: abjkqsyz word: .a....a..s errors: 6/7
guess: x
the answer was calculates, you blew it
```

เมื่อต้องการออกจากเกม ให้กดลำดับคีย์ Interrupt (Ctrl-C) หรือ End Of File (Ctrl-D)

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/games	ตำแหน่งของเกมของระบบ

### ข้อมูลที่เกี่ยวข้อง:

คำสั่งทางคณิตศาสตร์

คำสั่ง number

คำสั่ง quiz

คำสั่ง turnon

---

## คำสั่ง hash

### วัตถุประสงค์

จดจำหรือรายงานชื่อพาราคำสั่ง

### ไวยากรณ์

ในการเพิ่มพาราคำสั่งให้แก่รายการชื่อพาร:

```
hash[ Command... ]
```

ในการล้างค่ารายการชื่อพาร:

```
hash -r
```

### คำอธิบาย

คำสั่ง `hash` มีผลต่อ วิธีที่เชลล์ปัจจุบันจนวนชื่อพารของคำสั่ง โดยการเพิ่ม ชื่อพารในรายการหรือลบเนื้อหาออกจากรายการ

เมื่อไม่ระบุพารามิเตอร์หรือแฟล็ก คำสั่ง `hash` รายงานเนื้อหาของรายการชื่อพารไปยังเอาต์พุต มาตรฐาน รายงานประกอบด้วยชื่อพารของคำสั่งในสถานะแวดล้อมเชลล์ปัจจุบัน ที่พบโดยการเรียกทำงานคำสั่ง `hash` หน้าจออาจมีคำสั่งเหล่านี้ถูกเรียกใช้และถูกพบจาก กระบวนการค้นหาคำสั่งปกติ

หมายเหตุ: คำสั่งในตัวของเชลล์ไม่ถูกรายงานโดยคำสั่ง `hash`

คุณสามารถใช้แฟล็ก `-r` เพื่อล้างค่า เนื้อหาของรายการชื่อพารคำสั่ง ชื่อพารยังสามารถถูกลบออกจาก รายการได้โดยการรีเซ็ตค่าของตัวแปรสถานะแวดล้อม `PATH` ในรูปแบบง่ายที่สุด จะทำได้โดยการป้อน:

```
PATH="$PATH"
```

หากใช้พารามิเตอร์ *Command* คำสั่ง **hash** ค้นหาชื่อพารของ คำสั่งที่ระบุและเพิ่มพารนี้ในรายการ อย่าใช้ / (สแลช) เมื่อคุณระบุคำสั่ง

เนื่องจากคำสั่ง **hash** ส่งผล ต่อสถานะแวดล้อมเซลล์ปัจจุบัน คำสั่งจะถูกจัดเตรียมเป็นคำสั่งในตัวปกติของคอร์นเซลล์หรือ POSIX เซลล์ หากคำสั่ง **hash** ถูกเรียกใช้ในสถานะแวดล้อมการเรียกใช้งานคำสั่งต่างหาก อย่างในตัวอย่างต่อไปนี้จะไม่มีผลต่อกระบวนการค้นหาคำสั่งของสถานะแวดล้อมของผู้เรียก:

```
nohup hash -r  
find . -type f | xargs hash
```

การใช้คำสั่ง **hash** เทียบเท่ากับการใช้คำสั่ง **alias -t**

## แฟล็ก

ไอเท็ม คำอธิบาย  
-r ล่างค่าเนื้อหาของรายการชื่อพาร

## พารามิเตอร์

ไอเท็ม คำอธิบาย  
คำสั่ง ระบุ *Command* ที่จะเพิ่มใน รายการชื่อพาร

## สถานะออก

ค่าออกต่อไปนี้จะถูกส่งคืน:

ไอเท็ม คำอธิบาย  
0 ดำเนินการเป็นผลสำเร็จ  
>0 มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. ในการค้นหาชื่อพารของคำสั่ง **wc** และเพิ่มในรายการชื่อพารให้ป้อน:

```
hash wc
```

2. ในการล้างค่าเนื้อหาของรายการชื่อพารให้ป้อน:

```
hash -r
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/ksh	มีคำสั่งในตัว hash ของคอร์นเชลล์
/usr/bin/hash	มีคำสั่ง hash

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง alias

คำสั่ง bsh

คำสั่ง ksh

---

## คำสั่ง hatsoptions

### วัตถุประสงค์

ควบคุมอ็อพชันทอพอโลยีเซอร์วิส บนโหนด หรือเวิร์กสแตชันควบคุม

### ไวยากรณ์

**hatsoptions** [-s] [-d]

### คำอธิบาย

ก่อนที่จะเรียกใช้งานคำสั่งนี้ได้ ตัวแปรสถานะแวดล้อม HB\_SERVER\_SOCKET ต้องถูกตั้งค่าเป็น ตำแหน่งของซ็อกเก็ตโตเมน UNIX ที่ใช้โดยระบบย่อยทอพอโลยีเซอร์วิส โดยสามารถใช้คำสั่งด้านล่างได้:

```
export HB_SERVER_SOCKET=/var/ha/soc/hats/server_socket.partition name
```

หรือทางเลือก variable HA\_SYSPAR\_NAME สามารถถูกตั้งค่าเป็นชื่อ พาร์ติชัน

daemon เซอร์วิสการจัดเรียงต้องกำลังรันเพื่อให้คำสั่งนี้ ทำงานได้สำเร็จ

**hatsoptions** สามารถใช้เพื่อควบคุมจำนวนอ็อพชันในทอพอโลยี เซอร์วิส อ็อพชัน **-s** สั่งให้ daemon ทอพอโลยีเซอร์วิสปฏิเสธข้อความที่มีการหน่วงเวลาอย่างชัดเจน ค่านี้สามารถใช้ในการกำหนดค่าระบบขนาดใหญ่มาก ที่บางครั้งข้อความถูกหน่วงเวลาในเน็ตเวิร์กหรือ ในโหนดของผู้ส่งและผู้รับ ใช้อ็อพชันนี้ต่อเมื่อนาฬิกาบอกเวลา Time-Of-Day ถูกซิงโครไนซ์กับทุกโหนด และเวิร์กสแตชันควบคุม มิฉะนั้นข้อความอาจถูกละเว้นอย่างไม่ถูกต้องเมื่อนาฬิกาบอกเวลา Time-Of-Day ของผู้ส่งสูงกว่าของผู้รับ

อ็อพชัน **-d** สั่งให้ daemon ทอพอโลยีเซอร์วิสไม่ปฏิเสธข้อความที่มีการหน่วงเวลาอย่างชัดเจน นี่คือ ค่าดีฟอลต์

### แฟล็ก

**-s**      สั่งให้ daemon ทอพอโลยีเซอร์วิสปฏิเสธข้อความที่ถูกหน่วงเวลา อย่างชัดเจน

**-d**      สั่งให้ daemon ทอพอโลยีเซอร์วิสไม่ปฏิเสธข้อความที่ถูกหน่วงเวลา อย่างชัดเจน

### ความปลอดภัย

คุณต้องมีสิทธิ์ **root** ในการรันคำสั่งนี้

## สถานะออก

- 0 บ่งชี้ความสำเร็จของคำสั่ง
- 1 บ่งชี้ว่าคำสั่งทำงานไม่สำเร็จ

## ตัวแปรสถานะแวดล้อม

### HB\_SERVER\_SOCKET

ตัวแปรสถานะแวดล้อมนี้ควรถูกตั้งค่าก่อนที่จะสามารถเรียกใช้งานคำสั่งนี้ โดยต้องถูกตั้งค่าเป็นตำแหน่งของซ็อกเก็ต UNIX ที่ใช้โดยทอพอโลยีเซอร์วิส โคลเอ็นต์ เพื่อเชื่อมต่อกับ daemon ทอพอโลยีเซอร์วิส ตัวแปรสถานะแวดล้อมนี้ต้องถูกตั้งค่าเป็น `/var/ha/soc/hats/server_socket.partition name`

### HA\_SYSPAR\_NAME

หากไม่ตั้งค่า HB\_SERVER\_SOCKET ต้องตั้งค่า HA\_SYSPAR\_NAME เป็น ชื่อพาร์ติชันแทน

## ข้อจำกัด

คำสั่งนี้ใช้ได้เฉพาะโรดเมนเท่านั้น

## เอาต์พุตมาตรฐาน

เมื่อระบุแฟล็ก `-h` ไว้ข้อความการใช้งานของคำสั่งนี้ ถูกเขียนลงในเอาต์พุตมาตรฐาน ข้อความ verbose ทั้งหมดถูกเขียนลงในเอาต์พุตมาตรฐาน

## ข้อผิดพลาดมาตรฐาน

คำสั่งนี้เขียนข้อความแสดงความผิดพลาด (ตามความจำเป็น) ไปที่ข้อผิดพลาดมาตรฐาน

## ตัวอย่าง

ในการสั่งให้ daemon ทอพอโลยีเซอร์วิสบนโหนดเริ่มการละเว้น ข้อความที่ถูกหน่วงเวลาอย่างชัดเจน ให้ป้อน:

```
export HA_SYSPAR_NAME=partition1
```

```
/usr/sbin/rsct/bin/hatsoptions -s
```

## Location

```
/usr/sbin/rsct/bin/hatsoptions
```

มีคำสั่ง `hatsoptions`

## ไฟล์

```
/var/ha/soc/hats/server_socket.partition name
```

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `lssrc`

คำสั่ง `startsrc`

คำสั่ง `stopsrc`

---

## คำสั่ง head

### วัตถุประสงค์

แสดงสองสามบรรทัดแรกของไฟล์

### ไวยากรณ์

```
head[ -Count | -c Number | -n Number ][ File ... ]
```

### คำอธิบาย

คำสั่ง **head** จะเขียนบรรทัดหรือไบต์ตามจำนวนที่ระบุของแต่ละไฟล์ที่ระบุ หรือของอินพุตมาตรฐานไปยัง เอาต์พุตมาตรฐาน หากไม่ระบุแฟล็กกับคำสั่ง **head** 10 บรรทัดแรกจะถูกแสดงโดยค่าดีฟอลต์ พารามิเตอร์ *File* ระบุชื่อของอินพุตไฟล์ อินพุตไฟล์ ต้องเป็นไฟล์ข้อความ เมื่อระบุมากกว่าหนึ่งไฟล์ ตำแหน่งเริ่มต้นของแต่ละไฟล์ จะมีลักษณะต่อไปนี้:

```
==> filename <==
```

ในการแสดงชุดของไฟล์สั้น โดยการระบุแต่ละไฟล์ให้ป้อน:

```
example% head -9999 filename1 filename2...
```

### แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-Count</code>	ระบุจำนวนบรรทัดตั้งแต่เริ่มต้นของแต่ละไฟล์ ที่ระบุที่จะถูกแสดง ตัวแปร <i>Count</i> ต้องเป็น เลขจำนวนเต็มบวกฐานสิบ แฟล็กนี้เทียบเท่ากับแฟล็ก <code>-n Number</code> แต่ไม่ควรให้หากำหนดถึง ความสามารถในการเคลื่อนย้าย
<code>-c Number</code>	ระบุจำนวนไบต์ที่จะแสดง ตัวเลข <i>Number</i> ต้องเป็นเลขจำนวนเต็มบวกฐานสิบ
<code>-n Number</code>	ระบุจำนวนบรรทัดตั้งแต่เริ่มต้นของแต่ละไฟล์ ที่ระบุที่จะถูกแสดง ตัวแปร <i>number</i> ต้องเป็น เลขจำนวนเต็มบวกฐานสิบ แฟล็กนี้เทียบเท่ากับแฟล็ก <code>-Count</code>

### สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

### ตัวอย่าง

ในการแสดงห้าบรรทัดแรกของไฟล์ `Test` ให้ป้อน:

```
head -5 Test
```

หรือ

```
head -n 5 Test
```

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง tail

คำสั่ง Files

การเปลี่ยนทิศทางอินพุตและเอาต์พุต

---

## คำสั่ง help

### วัตถุประสงค์

ให้ข้อมูลสำหรับผู้ใช้ใหม่

### ไวยากรณ์

help

### คำอธิบาย

คำสั่ง **help** แสดงการแสดงผลข้อมูล หนึ่งหน้าสำหรับผู้ใช้ใหม่ ข้อมูลจะมีหัวข้อต่อไปนี้:

- การต่อ หรือการแสดงผลไฟล์
- การแก้ไขบรรทัดแบบโต้ตอบ
- การส่งและการรับเมล
- การอ่านข้อความระบบ
- การเปลี่ยนข้อมูลไฟล์รหัสผ่าน
- การระบุผู้ใช้ปัจจุบันของระบบ
- การส่งข้อความไปยังผู้ใช้อื่นบนระบบ
- การแสดงเนื้อหาของไดเร็กทอรี
- การดูข้อมูลบน Source Code Control System
- การตั้งค่าเทอร์มินัลโหมด

### ตัวอย่าง

ในการขอรับวิธีใช้ให้พิมพ์ **help** ที่บรรทัดคำสั่ง

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง ls

คำสั่ง mail

คำสั่ง sccshelp

คำสั่ง who

# คำสั่ง **hfistat**

## วัตถุประสงค์

แสดงสถิติผลการทำงานของ host fabric interface

## ไวยากรณ์

**hfistat** [-O options] [ interval [ count ] ]

**hfistat** [-h]

## คำอธิบาย

คำสั่ง **hfistat** แสดงสถิติผลการทำงาน ที่เกี่ยวข้องกับ host fabric interface

ข้อมูลต่อไปนี้เป็นคำอธิบายของส่วนหัวของคอลัมน์ ในตารางเอาต์พุต

ประเภท	ชื่อคอลัมน์พร้อมด้วยคำอธิบาย
HFI	<p><b>HFI</b> ตัวระบุ Host fabric interface (0, 1, ...) ระบุ อินเตอร์เฟซที่มีการแสดงสถิติ</p> <p><b>packets - sent</b> ผลรวมของการนับแพ็กเก็ตที่ส่ง (ตัวนับ 56-บิต) คือจำนวนของแพ็กเก็ตที่ส่งไปยังเน็ตเวิร์กคลัสเตอร์โดยไม่คำนึงถึงหน้าต่างที่ส่ง</p> <p><b>packets - imm_send</b> ผลรวมของการนับแพ็กเก็ตที่ส่งในทันที (ตัวนับ 56-บิต) คือจำนวนของแพ็กเก็ตที่ต้องถูกส่งออกโดยทันทีโดยไม่พิจารณาถึงหน้าต่างที่ส่ง</p> <p><b>packets - receive</b> ผลรวมของการนับแพ็กเก็ตที่ได้รับ (ตัวนับ 56-บิต) คือจำนวนของแพ็กเก็ตที่ได้รับเป็นผลสำเร็จจากเน็ตเวิร์กคลัสเตอร์โดยไม่พิจารณาถึงหน้าต่างที่รับ ตัวนับ <b>packets - receive</b> จะเพิ่มขึ้นเมื่อใดก็ตามที่ตัวนับแพ็กเก็ตที่ได้รับ ของหน้าต่างใดๆ เพิ่มขึ้น</p> <p><b>sent_packets - fullRDMA</b> ผลรวมของการนับแพ็กเก็ตแบบ full-RDMA แบบเต็มที่ส่ง (ตัวนับ 56-บิต) คือจำนวนของแพ็กเก็ตแบบ full-RDMA แบบเต็มที่ส่ง โดยไม่พิจารณาถึงหน้าต่างที่ส่ง</p> <p><b>sent_packets - halfRDMA</b> ผลรวมของการนับแพ็กเก็ตแบบ half-RDMA ที่ส่ง (ตัวนับ 56-บิต) คือจำนวนของแพ็กเก็ตแบบ half-RDMA ที่ส่ง โดยไม่พิจารณาถึงหน้าต่างที่ส่ง ตัวนับ <b>sent_packets - halfRDMA</b> ไม่ได้เพิ่มขึ้นสำหรับการแจ้งให้ทราบ</p>

ประเภท	ชื่อคอลัมน์พร้อมด้วยคำอธิบาย
	<p><b>sent_packets – smallRDMA</b>            ผลรวมของการนับแพ็กเก็ตแบบ small-RDMA ที่ส่ง (ตัวนับ 56-บิต) คือจำนวนของแพ็กเก็ตแบบ small-RDMA ที่ส่ง โดยไม่พิจารณาถึงหน้าต่าง ที่ส่ง ตัวนับ sent_packets – smallRDMA ไม่ได้เพิ่มขึ้นสำหรับการแจ้งให้ทราบ</p> <p><b>sent_packets – ip</b>            ผลรวมของการนับแพ็กเก็ต IP ที่ส่ง (ตัวนับ 56-บิต) คือจำนวนของแพ็กเก็ต IP ที่ส่ง โดยไม่พิจารณาถึงหน้าต่างที่ส่ง</p> <p><b>sent_packets – cau</b>            ผลรวมของการนับแพ็กเก็ต CAU (ตัวนับ 56-บิต) คือจำนวนของแพ็กเก็ต CAU ที่ส่ง โดยไม่พิจารณาถึงหน้าต่างที่ส่ง</p> <p><b>sent_packets – gups</b>            ผลรวมของการนับแพ็กเก็ต GUPS ที่ส่ง (ตัวนับ 56-บิต) ระบุจำนวนของแพ็กเก็ต GUPS ที่ส่ง โดยไม่พิจารณาถึงหน้าต่างที่ส่ง</p> <p><b>dropped_packets – sending</b>            ผลรวมของแพ็กเก็ตที่ปล่อยจากการนับการส่ง (ตัวนับ 56-บิต) ระบุจำนวนของแพ็กเก็ตที่ปล่อยและไม่ได้ส่งโดย FIFO ซึ่งหมายความว่าส่งแพ็กเก็ต โดยไม่พิจารณาถึงหน้าต่าง</p> <p><b>dropped_packets – receiving</b>            ผลรวมของแพ็กเก็ตที่ปล่อยจากการนับที่ได้รับ (ตัวนับ 56-บิต) ระบุจำนวนของแพ็กเก็ตจาก ISR ที่ปล่อยและไม่ได้รับ โดยไม่พิจารณาถึงหน้าต่างที่ได้รับ</p> <p><b>xlat – wait</b>            แสดงการนับ xlat ที่รอ (ตัวนับ 56-บิต) ระบุจำนวนของ การแปลที่หายไปและค้างอยู่ รีจิสเตอร์ xlat – wait จะรีเซ็ตเมื่อคุณเขียนสิ่งใดๆ ไปยังรีจิสเตอร์นี้</p>
ISR	<p><b>HFI</b> ตัวระบุ Host fabric interface (0, 1, ...) ระบุ อินเทอร์เน็ตเฟสที่มีการแสดงสถิติ</p> <p><b>cycBlocked – sending</b>            วงรอบจะบล็อกจากการส่งตัวนับ (64-บิต) ที่เพิ่มขึ้นทุกๆ 2 - 3 GHz วงรอบของชิปเมื่อการรอไม่สามารถส่งผ่านลิงก์ได้</p> <p><b>flits – sent</b>            flits ส่ง (ตัวนับ 64-บิต) คล้ายกับวงรอบที่บล็อกตัวนับ ยกเว้นว่า ตัวนับนี้เพิ่มขึ้นในแต่ละครั้งที่ส่วนหัว flit ส่งผ่านอินเทอร์เน็ตเฟส ISR ที่สอดคล้องกัน</p> <p><b>flits – dropped</b>            flits ที่ปล่อย (ตัวนับ 40-บิต) คือจำนวนของ flits ที่ถูกปล่อยและนับเมื่อใดก็ตามที่เหตุการณ์เกิดขึ้น เช่น ไอเท็มต่อไปนี้:</p> <ol style="list-style-type: none"> <li>1. บิตสถานะลิงก์ของพอร์ตคือปิด</li> <li>2. ISR ID ไม่ถูกต้อง</li> </ol> <p><b>link – retries</b>            ตัวนับระดับความพยายามของลิงก์ (ตัวนับ 24-บิต) จะเพิ่มขึ้นในแต่ละครั้งที่ flit ถูกถอนออกจากลิงก์ที่เล่นบัฟเฟอร์และส่งผ่านลิงก์ อีกครั้งเนื่องจากเกิดข้อผิดพลาด</p>

ประเภท	ชื่อคอลัมน์พร้อมด้วยคำอธิบาย
NMMU	<p><b>dyn_prot_cache – hits</b>            แคชการปกป้อง Nest memory management unit แบบไดนามิกที่ฮิต</p> <p><b>dyn_prot_cache – misses</b>            แคชการปกป้อง Nest memory management unit แบบไดนามิกที่หายไป</p> <p><b>ATLB – hits</b>            บัฟเฟอร์การแปลแอดเดรส Nest memory management unit ที่ฮิต</p> <p><b>ATLB – misses</b>            บัฟเฟอร์การแปลแอดเดรส Nest memory management unit ที่หายไป</p>
CAU	<p><b>cycles – waiting</b>            วงรอบที่รอดตามเครดิต (ตัวนับที่ไม่ใช่ดัชนี)</p>
Window-based	<p><b>HFI</b>    ตัวระบุ Host fabric interface (0, 1, ...) ระบุ อินเทอร์เน็ตที่มีการแสดงสถิติ</p> <p><b>Win</b>    หมายเลขหน้าต่างต่าง (0, 1, 2 ...) ระบุหน้าต่างที่มีการแสดงสถิติ</p> <p><b>packet_indicated – send</b>            การนับที่ส่งแพ็กเก็ตที่ระบุไว้ (ตัวนับ 56-บิต) เพิ่มขึ้นเมื่อใดก็ตามที่ส่งแพ็กเก็ตที่มีบิต Packet-Indicated-Count ในชุดส่วนหัว</p> <p><b>packet_indicated – receive</b>            การนับที่ได้รับแพ็กเก็ตที่ระบุไว้ (ตัวนับ 56-บิต) เพิ่มขึ้นเมื่อใดก็ตามที่ได้รับแพ็กเก็ตที่มีบิต packet-indicated-count ในชุดส่วนหัว แต่ก่อนที่แพ็กเก็ตจะถูกเขียนลงในหน่วยความจำ</p> <p><b>packets – sent</b>            การนับแพ็กเก็ตที่ส่ง (ตัวนับ 56-บิต) คือจำนวนของแพ็กเก็ตที่ส่งไปยังเน็ตเวิร์กคลัสเตอร์เป็นผลสำเร็จและรวมแพ็กเก็ตที่ส่งไปยัง HFI เดียวกันโดยส่งไปยังเน็ตเวิร์กคลัสเตอร์ที่ห่อกลับไว้</p> <p><b>packets – received</b>            การนับแพ็กเก็ตที่ได้รับ (ตัวนับ 56-บิต) คือจำนวนของ แพ็กเก็ตที่ได้รับเป็นผลสำเร็จจากเน็ตเวิร์กคลัสเตอร์ และ รวมแพ็กเก็ตที่ได้รับจาก HFI เดียวกัน ซึ่งจะถูกรับห่อกลับไว้โดย เน็ตเวิร์กคลัสเตอร์</p> <p><b>packet_dropped – sending</b>            แพ็กเก็ตที่ปล่อยจากการส่งการนับ (ตัวนับ 40-บิต) คือจำนวนของแพ็กเก็ตจากการส่ง FIFO ที่ถูกลบออกและไม่ได้ส่ง</p> <p><b>packet_dropped – receiving</b>            แพ็กเก็ตที่ปล่อยจากการรับการนับ (ตัวนับ 40-บิต) คือจำนวนของแพ็กเก็ตจาก ISR ที่ถูกลบออกและไม่ได้รับ</p> <p><b>immediate – send_pkts</b>            การนับแพ็กเก็ตที่ส่งทันที (ตัวนับ 56-บิต)</p>

## แฟล็ก

### คำอธิบาย

ระบุเนื้อหาและการนำเสนอ ของรายงาน ใช้พารามิเตอร์ Options กับแฟล็ก -O

-O option1=value1,option2=value2,option3="value3 value4 value5"

รายการของอ็อปชันต้องเป็นรายการของค่าที่คั่นด้วยเครื่องหมายคอมมาและรายการของค่า ต้องอยู่ในเครื่องหมายอัฒภาคเปิดและปิด (" ") และคั่นด้วยช่องว่าง

ต่อไปนี้เป็นอ็อปชันที่สนับสนุนค่าเหล่านั้น:

- `type = [window nonwindow hfi isr nmmu cau all ]`

อ็อปชัน `type` ระบุว่า คำสั่ง `hfistat` แสดงเฉพาะประเภทของค่าที่ระบุซึ่งต้องถูกแสดง

Default value: hfi

คุณสามารถระบุค่าต่อไปนี้กับอ็อปชัน `type`:

**หน้าต่าง** แสดงสถิติผลการทำงานที่อ้างอิงหน้าต่าง

#### nonwindow

แสดงสถิติผลการทำงานที่ไม่ได้อ้างอิงหน้าต่าง-

**hfi** แสดงค่าตัวนับผลการทำงานสำหรับ HFI

**isr** แสดงค่าการลงทะเบียน integrated switch router (ISR)

**nmmu** แสดงค่าการลงทะเบียน Nest Memory Management Unit (NMMU)

**cau** แสดงค่าการลงทะเบียน Collectives Acceleration Unit (CAU)

**ทั้งหมด** แสดงค่าการลงทะเบียนทั้งหมด

- `display = [raw|delta]`

อ็อปชัน `display` ถูกใช้เพื่อตีพิมพ์ค่าการลงทะเบียน

Default value: none

คุณสามารถระบุค่าต่อไปนี้ด้วยอ็อปชัน `display`:

**ดิบ** ตีพิมพ์ค่าการลงทะเบียนดิบที่รวบรวมไว้

**delta** ตีพิมพ์ค่าการเปลี่ยนแปลงของการลงทะเบียน

#### หมายเหตุ:

1. เครื่องมือ `hfistat` แสดงเอาต์พุตการจัดรูปแบบ เมื่อไม่ได้จัดเตรียมอ็อปชันการแสดงผลไว้
2. ค่าอ็อปชันดิบและที่เปลี่ยนแปลงเป็นค่าเฉพาะ

- `hfi = [ 0 1 ... ]`

อ็อปชัน `hfi` ระบุรายการของ host fabric interfaces ที่รายงาน ค่าการลงทะเบียน

Default value: All available HFIs in the system

#### หมายเหตุ:

1. คุณสามารถระบุช่วงของ host fabric interfaces ด้วยหนึ่งในวิธีต่อไปนี้:

`hfistat -O hfi="0 1 2 3"`

`hfistat -O hfi=0-3`

2. ส่วนท้ายที่ว่างของรายการสามารถใช้เพื่อแสดง host fabric interface ที่พร้อมใช้งานล่าสุด ตัวอย่างของอ็อปชัน `hfi` นี้บ่งชี้ช่วงตั้งแต่ 1 ถึง host fabric interface ที่พร้อมใช้งานล่าสุด

`hfistat -O hfi=1-`

## ไอเท็ม

### คำอธิบาย

- `window = [ 0 1 2... ]`

อ็อปชัน `window` ระบุรายการของจำนวนหน้าต่าง HFI ที่ค่าการลงทะเบียน ถูกรายงาน

*Default value: All available HFI windows for the specified HFIs*

### หมายเหตุ:

1. คุณสามารถระบุช่วงของจำนวนหน้าต่าง HFI ด้วยหนึ่งในวิธีต่อไปนี้:

```
hfistat -0 window="210 211 212 213 214 215 216"
```

```
hfistat -0 window=210-216
```

2. ส่วนท้ายที่ว่างของรายการสามารถใช้เพื่อแสดงจำนวนหน้าต่าง HFI ที่พร้อมใช้งานล่าสุด ตัวอย่างนี้ระบุอ็อปชัน `window` ที่บ่งชี้ช่วงจาก 0 ถึงหน้าต่าง HFI ที่พร้อมใช้งานล่าสุด

```
hfistat -0 window=0-
```

- `output = <filename>`

อ็อปชัน `output` ระบุไฟล์เอาต์พุตที่ต้องถูกใช้แทน `stdout`

*ค่าดีฟอลต์: None*

**หมายเหตุ:** ชื่อไฟล์เป็นไฟล์ดิงค์

ระบุช่วงเวลาในหน่วยวินาทีสำหรับคำสั่ง `hfistat` เพื่อรวบรวมและพิมพ์สถิติ หากไม่ระบุพารามิเตอร์

`interval` ไว้ คำสั่ง `hfistat` จะรันด้วยช่วงเวลา 2- วินาที

ระบุจำนวนการทำซ้ำสำหรับคำสั่ง `hfistat` เพื่อรวบรวมและพิมพ์สถิติ ใช้พารามิเตอร์ `Count` พร้อม

กับอ็อปชัน `interval` ของช่วง หากไม่ระบุทั้งพารามิเตอร์ `Count` และ `interval` ไว้ คำสั่ง `hfistat` รัน 10 ครั้ง หาก

ระบุพารามิเตอร์ `interval` และไม่ระบุพารามิเตอร์ `Count` คำสั่ง `hfistat` ให้รันอย่างไม่มีการกำหนด

`interval`

`Count`

## ตัวอย่าง

1. หากต้องการแสดง HFI- ที่อ้างอิงสถิติผลการทำงานสำหรับ HFIs ที่พร้อมใช้งานทั้งหมดโดยใช้ช่วงเวลา 2-วินาที สำหรับการวนซ้ำ 10 ครั้ง ให้รันคำสั่งต่อไปนี้:

```
# hfistat
```

2. หากต้องการแสดงสถิติผลการทำงานทั้งหมดสำหรับ HFIs ที่พร้อมใช้งานซึ่งมีสถิติผลการทำงานแบบอิงหน้าต่างของหน้าต่างทั้งหมดในเอาต์พุตที่จัดรูปแบบแล้ว ให้รันคำสั่งต่อไปนี้:

```
# hfistat -0 type=all 2 5
```

3. หากต้องการแสดงสถิติผลการทำงานแบบอิงหน้าต่างสำหรับ หน้าต่างไม่กี่บาน (0-15) ในเอาต์พุตที่จัดรูปแบบแล้ว ให้รันคำสั่งต่อไปนี้:

```
# hfistat -0 type=window,window=0-15
```

4. หากต้องการแสดงค่าการลงทะเบียน CAU สำหรับ HFI-1 ในเอาต์พุตการจัดรูปแบบ สำหรับ 10 ตัวอย่าง (ค่าดีฟอลต์) และช่วงเวลา 2 วินาที (ค่าดีฟอลต์) ให้รันคำสั่งต่อไปนี้:

```
# hfistat -0 type=cau,hfi=1
```

5. หากต้องการแสดงทั้ง CAU และค่าการลงทะเบียน nest memory management unit (NMMU) ในเอาต์พุตที่จัดรูปแบบแล้ว ให้รันคำสั่งต่อไปนี้:

```
# hfistat -0 type="cau nmmu"
```

6. หากต้องการพิมพ์ค่าการลงทะเบียนดิบของ HFIs ทั้งหมดซึ่งมีสถิติผลการทำงานที่อ้างอิงหน้าต่าง สำหรับหน้าต่าง 0 ด้วยช่วงเวลา 2 วินาที และการวนซ้ำ 5 ครั้ง ให้รันคำสั่งต่อไปนี้:

```
# hfistat -0 display=raw,type=all>window=0 2 5
```

7. หากต้องการพิมพ์ค่าการเปลี่ยนแปลงสำหรับตัวนับผลการทำงาน ISR เท่านั้นด้วยช่วงเวลา 2-วินาทีและการวนซ้ำ 5 ครั้ง ให้รันคำสั่งต่อไปนี้:

```
# hfistat -0 display=delta,type=isr 2 5
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/hfistat	มีคำสั่ง hfistat

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง perfstat\_hfistat

Perfstat API

---

## คำสั่ง hmcauth

### วัตถุประสงค์

คำสั่ง **hmcauth** ใช้เพื่อพิสูจน์ตัวตนกับ Hardware Management Console (HMC) และขอรับโทเค็นเพื่อใช้เซอรัลิส HMC สำหรับการดำเนินการ AIX Live Update นอกจากนี้ยังสามารถใช้เพื่อทำให้โทเค็นใช้ไม่ได้ด้วย

### ไวยากรณ์

เมื่อต้องการพิสูจน์ตัวตนกับ HMC และรับโทเค็น ให้ใช้ไวยากรณ์ต่อไปนี้:

```
hmcauth [ -u user_name ] [ -p password ] [ -a hmc_uri ] [ -P port ]
```

เมื่อต้องการทำให้ใช้ไม่ได้และลบโทเค็นที่สร้างขึ้นก่อนหน้านี้ ให้ใช้ไวยากรณ์ต่อไปนี้:

```
hmcauth -r [ -a hmc_uri ] [ -u user_name ]
```

เมื่อต้องการแสดงรายการโทเค็นการพิสูจน์ตัวตน HMC ที่รู้จักทั้งหมด ให้ใช้ไวยากรณ์ต่อไปนี้:

```
hmcauth -l
```

เมื่อต้องการแสดงข้อความสั่งการใช้งานคำสั่ง ให้ใช้ไวยากรณ์ต่อไปนี้:

```
hmcauth -h
```

### คำอธิบาย

คุณสามารถใช้คำสั่ง **hmcauth** ถ้าคุณมีสิทธิ์เข้าถึงอ็อบเจกต์ทั้งหมด และ สิทธิควบคุมดูแล HMC ที่เหมาะสม คำสั่ง **hmcauth** จะสร้างโทเค็นที่ ผู้ดูแลระบบพาร์ติชัน AIX สามารถใช้เพื่อทำการดำเนินการ Live Update ถ้าคำสั่งสำเร็จ โทเค็น จะมีการจัดเก็บไว้ในแคอร์เนลเพื่อให้อินเทอร์เฟซ **geninstall** สามารถทำการดำเนินการ Live Update

ในการใช้คำสั่งนี้ คุณต้องมีสิทธิ์ทำภารกิจต่อไปนี้:

- เปิดพาร์ติชันที่ถูกจัดการ
- ปิดพาร์ติชันที่ถูกจัดการ

- ลบพาร์ติชันที่ถูกจัดการ (โหนดอัตโนมัติเท่านั้น)
- สร้างพาร์ติชันที่ถูกจัดการโดยใช้ข้อมูลโปรไฟล์ปัจจุบัน (โหนดอัตโนมัติเท่านั้น)
- ตั้งค่าอุปกรณ์บูตของพาร์ติชันที่ถูกจัดการ
- จัดการอะแดปเตอร์อีเทอร์เน็ตเสมือน

บทบาท `hmcclientliveupdate` HMC มีสิทธิ์ทั้งหมดที่จำเป็นสำหรับการดำเนินการ Live Update หากผู้ใช้ถูกกำหนด บน HMC ด้วยบทบาทนี้ การพิสูจน์ตัวตนสามารถทำได้โดยผู้ใช้นี้แทนที่จะใช้ผู้ใช้ `hscroot`

คำสั่ง `hmcauth` ยังสามารถใช้โดยไม่มีแฟล็กได้ด้วย ถ้าคุณไม่ได้ระบุแฟล็กใดๆ คำสั่ง `hmcauth` จะพร้อมตัวให้ป้อนข้อมูลที่จำเป็นทั้งหมด เช่น `user_name`, `hmc_uri` และ `password`

**หมายเหตุ:** หาก LPAR รีสตาร์ท โทเค็นการพิสูจน์ตัวตน HMC จะไม่ถูกเก็บไว้ ดังนั้น คุณต้อง พิสูจน์ตัวตนกับ HMC อีกครั้ง ก่อนที่จะพยายามดำเนินการ Live Update

## พารามิเตอร์

Item	คำอธิบาย
<code>user_name</code> รหัสผ่าน	สตริงของอักขระสูงสุด 64 อักขระที่ระบุผู้ใช้ HMC
<code>hmc_uri</code>	สตริงของอักขระสูงสุด 64 อักขระที่ระบุรหัสผ่าน
<code>port</code>	สตริงของอักขระสูงสุด 64 อักขระที่ระบุ uniform resource identifier (URI) ซึ่งมีชื่อโฮสต์ หรือ IP แอดเดรสของ HMC เพื่อใช้พิสูจน์ตัวตน
<code>port</code>	สตริงของอักขระสูงสุด 16 อักขระที่ระบุหมายเลขพอร์ตที่จะติดต่อ HMC

## แฟล็ก

Item	คำอธิบาย
<code>-a hmc_uri</code>	ระบุ URI ซึ่งมีชื่อโฮสต์ หรือ IP แอดเดรสของ HMC เพื่อใช้พิสูจน์ตัวตน
<code>-h</code>	ถ้าไม่ได้ระบุตัวแปร <code>hmc_uri</code> คำสั่งจะพร้อมตัวให้ป้อน
<code>-r</code>	เขียนขอความถี่การใช้งานคำสั่งไปยังเอาต์พุตมาตรฐาน
<code>-P port</code>	ลบโทเค็นที่สร้างขึ้นโดย HMC
<code>-p password</code>	ระบุหมายเลขพอร์ตที่จะใช้เพื่อติดต่อ HMC
<code>-u user_name</code>	แฟล็ก <code>-P</code> เป็น ทางเลือก ดังนั้น ถ้าไม่ได้ระบุหมายเลขพอร์ต ค่าดีฟอลต์ของหมายเลขพอร์ต คือ 12443
	ระบุรหัสผ่านสำหรับการพิสูจน์ตัวตน ถ้าไม่ได้ระบุรหัสผ่านบนบรรทัดรับ คำสั่ง คุณจะได้รับพร้อมตัวให้ป้อนรหัสผ่าน
	ระบุชื่อผู้ใช้ HMC เพื่อใช้พิสูจน์ตัวตน คุณต้องมีสิทธิ์เข้าถึงอ็อบเจกต์ทั้งหมด และ สิทธิทำภารกิจที่เหมาะสมบน HMC

## ตัวอย่าง

1. เมื่อต้องการพิสูจน์ตัวตนกับ HMC ที่ชื่อ `apollo` บนพอร์ต 12499 ให้ป้อนคำสั่ง ต่อไปนี้:
 

```
# hmcauth -a apollo -u hscroot -p T2x6z42p -P 12499
```
2. เมื่อต้องการพิสูจน์ตัวตนกับ HMC ที่ IP 5.5.55.121 ด้วยพร้อมตัวรหัสผ่าน ให้ป้อนคำสั่ง ต่อไปนี้:
 

```
# hmcauth -a 5.5.55.121 -u hscroot
Enter password for hscroot:
```
3. เมื่อต้องการโมฆะการพิสูจน์ตัวตนก่อนหน้านี้กับ HMC ที่ IP 5.5.55.121 ให้ป้อนคำสั่ง ต่อไปนี้:
 

```
# hmcauth -r -a 5.5.55.121
```

---

## คำสั่ง host

### วัตถุประสงค์

แปลงชื่อโฮสต์ เป็น Internet Protocol (IP) แอดเดรสหรือ IP แอดเดรส เป็นชื่อโฮสต์

### ไวยากรณ์

```
host [-n [-a] [-c Class] [-d] [-r] [-t Type] [-v] [-w]] Hostname | Address [ Server ]
```

```
hostnew [-a] [-c Class] [-d] [-r] [-t Type] [-v] [-w] Hostname | Address [ Server ]
```

### คำอธิบาย

คำสั่ง `/usr/bin/host` จะส่งคืน IP แอดเดรสของเครื่องโฮสต์เมื่อระบุพารามิเตอร์ `HostName` และชื่อของโฮสต์เมื่อระบุพารามิเตอร์ `Address` ขึ้นอยู่กับการกำหนดคอนฟิกเซอรัวส์การแปลงชื่อ คำสั่ง `host` ยังอาจแสดงนามแฝงใดๆ ที่เชื่อมโยงกับพารามิเตอร์ `HostName` ตัวอย่างของเซอรัวส์ การกำหนดชื่อได้แก่ `local`, `nis` และ `bind`

หาก โคลด์โฮสต์กำลังใช้ Domain Name Protocol ฐานข้อมูลเซิร์ฟเวอร์ชื่อแบบโลคัลหรือรีโมต จะถูกเคียวรีก่อนที่จะค้นหาโลคัลไฟล์ `/etc/hosts`

### แฟล็ก

ไอเท็ม	คำอธิบาย
-a	เทียบเท่ากับการใช้ "-v -t *"
-c Class	ระบุคลาสที่จะค้นหาเมื่อค้นหาข้อมูลที่ไม่ใช่ข้อมูลอินเทอร์เน็ต คลาสที่ใช้ได้แก่:  IN อินเทอร์เน็ตคลาส  CHAOS คลาส Chaos  HESIOD คลาส MIT Athena Hesiod  ANY Wildcard (ค่าใดค่าหนึ่งด้านบน)
-d	เปิดทำงานโหมดการตีบัก
-n	เทียบเท่ากับการเรียกใช้คำสั่ง <code>/usr/bin/hostnew</code> คำสั่ง <code>hostnew</code> ดำเนินการเซอรัวส์การกำหนด <code>bind</code>
-r	ปิดใช้งานการประมวลผลที่เรียกซ้ำ

ไอเท็ม	คำอธิบาย
-t Type	ระบุชนิดของเร็กคอร์ดที่จะเคียวรี ชนิดที่ใช้ได้แก่:
	A IP แอดเดรสของโฮสต์
	CNAME ชื่อเสมือนสำหรับ alias
	HINFO ชนิดตัวประมวลผลและระบบปฏิบัติการของโฮสต์
	KEY เร็กคอร์ดคีย์การรักษาความปลอดภัย
	MINFO ข้อมูลเมลบ็อกซ์หรือรายการเมล
	MX Mail exchanger
	NS Nameserver สำหรับโซนที่ระบุชื่อ
	PTR ชื่อโฮสต์หากเคียวรีเป็น IP แอดเดรสไม่ฉะนั้น เป็นตัวชี้ไปยังข้อมูลอื่น
	SIG เร็กคอร์ดลายเซ็น
	SOA ข้อมูล "start-of-authority" ของโดเมน
	TXT ข้อมูลข้อความ
	UINFO ข้อมูลผู้ใช้
	WKS เซอร์วิสที่เป็นที่ระบุที่ได้รับการสนับสนุน
-v	โหมตรงายละเอียด
-w	รอไปตลอดจนกว่าจะได้รับการตอบกลับจากเซิร์ฟเวอร์ DNS

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
Address	ระบุ IP แอดเดรสของเครื่องโฮสต์เพื่อใช้ในการแปลงชื่อโฮสต์ พารามิเตอร์ Address ต้องเป็น IP แอดเดรสที่ถูกตั้งอยู่ในรูปเลขฐานสิบที่คั่นด้วยจุด
HostName	ระบุชื่อของเครื่องโฮสต์เพื่อใช้ในการแปลง IP แอดเดรส พารามิเตอร์ HostName สามารถเป็นชื่อโฮสต์เฉพาะ หรือชื่อโฮสต์ที่รู้จัก (เช่น nameserver, printserver หรือ timeserver หากมีชื่อเหล่านี้แล้ว)
Server	ระบุ nameserver เพื่อเคียวรี

## ตัวอย่าง

- ในการแสดงแอดเดรสของเครื่องโฮสต์ชื่อ mephisto ป้อนคำสั่งต่อไปนี้:

```
host mephisto
```

เอาต์พุต จะคล้ายกับข้อมูลต่อไปนี้:

```
mephisto is 192.100.13.5, Aliases: engr, sarah
```
- เมื่อต้องการแสดงโฮสต์ที่แอดเดรสคือ 192.100.13.1 ให้ป้อนคำสั่งต่อไปนี้:

```
host 192.100.13.1
```

เอาต์พุต จะคล้ายกับข้อมูลต่อไปนี้:

```
mercutio is 192.100.13.1
```
- เมื่อต้องการแสดงเร็กคอร์ด MX สำหรับโดเมนที่ชื่อ test.ibm.com ให้ป้อน:

```
host -n -t mx test.ibm.com
```

หรือ

```
hostnew -t mx test.ibm.com
```

เอาต์พุต จะคล้ายกับข้อมูลต่อไปนี้:

```
test.ibm.com mail is handled (pri=10) by test1.tt.ibm.com
test.ibm.com mail is handled (pri=10) by test2.aix.ibm.com
```

## ไฟล์

ไอเท็ม คำอธิบาย  
/etc/hosts มีชื่อ Internet Protocol (IP) และแอดเดรสของ โฮสต์บนโลคัลเน็ตเวิร์ก

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง hostname” ในหน้า 802

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง named

การสื่อสารและเน็ตเวิร์ก

---

## คำสั่ง host9

### วัตถุประสงค์

ดำเนินการค้นหา DNS

### ไวยากรณ์

```
host9 [-aCdIrsTwv] [-c class] [-N ndots] [-R number] [-t type] [-W wait] [-m flag] [-4] [-6] name [server]
```

### คำอธิบาย

คำสั่ง `host9` เป็นยูทิลิตี้ที่ง่ายสำหรับ ดำเนินการค้นหา DNS คุณสามารถใช้คำสั่งนี้เพื่อแปลงชื่อ เป็น IP แอดเดรสและแปลงกลับ เมื่อคุณไม่ระบุอาร์กิวเมนต์หรืออ็อปชัน คำสั่ง `host9` จะพิมพ์ข้อมูลสรุปอย่างย่อ ของอาร์กิวเมนต์และอ็อปชันบรรทัดคำสั่ง

### แฟล็ก

ไอเท็ม	คำอธิบาย
-a	เทียบเท่ากับการใช้แฟล็กของ <code>-v -t *</code>
-c class	สิ่งในสิ่งเดียวรี DNS ของคลาส ที่ระบุ คุณสามารถใช้แฟล็กนี้เพื่อค้นหาคลาสรีซอร์สเร็กคอร์ด Hesiod หรือ Chaosnet ดีพอลต์คลาสคือ IN (อินเทอร์เน็ต)
-C	พยายามแสดงเร็กคอร์ด SOA สำหรับ ชื่อโซนจากเซิร์ฟเวอร์ชื่อที่ผ่านการอนุญาตที่แสดงรายการทั้งหมด สำหรับ โซนนั้น เร็กคอร์ด NS ที่พบสำหรับโซนจะกำหนดรายการเซิร์ฟเวอร์ชื่อ
-d	สร้างเอาต์พุตรายละเอียด แฟล็กนี้เทียบเท่ากับ แฟล็ก <code>-v</code>
-I	ระบุโหมดรายการ คำนี้ทำให้คำสั่ง <code>host9</code> ดำเนินการถ่ายโอนโซนสำหรับชื่อ โซน ถ่ายโอนโซนจะพิมพ์เร็กคอร์ด NS, PTR และแอดเดรส (A/AAAA) หากคุณใช้แฟล็ก <code>-I</code> กับ แฟล็ก <code>-a</code> คำสั่ง <code>host9</code> จะพิมพ์เร็กคอร์ดทั้งหมด
-m flag	ตั้งค่าเร็กคอร์ดแฟล็กการตีบักการใช้งานหน่วยความจำ การใช้งาน และการติดตาม

## ไอเท็ม

-N *ndots*

### คำอธิบาย

ตั้งค่าจำนวนจุดที่ต้องมีอยู่ในชื่อซึ่งจะพิจารณาว่าเป็นค่าสัมบูรณ์ ค่าตีพอลต์คือค่าที่กำหนดโดยใช้คำสั่ง *ndots* ในไฟล์ `/etc/resolv.conf` หรือ 1 ถ้าไม่มีคำสั่ง *ndots* อยู่ชื่อที่มีจุดน้อยกว่า ถูกแปลงเป็นชื่อสัมพัทธ์และถูกค้นหาในโดเมนที่แสดงในไดเรกทีฟการค้นหาหรือโดเมนในไฟล์ `/etc/resolv.conf`

-r

เปิดให้คำสั่ง *host9* ลองเปลี่ยนลักษณะการทำงานของเซิร์ฟเวอร์ชื่อโดยการทำเคียวรีแบบไม่มีการเรียกซ้ำ และรอรับคำตอบของเคียวรีนั้นที่โดยปกติแล้ว เป็นการอ้างอิงไปยังเซิร์ฟเวอร์ชื่ออื่น

-R *number*

เปลี่ยนจำนวนการลองค้นหา UDP ใหม่ ค่า *number* ระบุจำนวนครั้งที่คำสั่ง *host9* ทำซ้ำเคียวรีที่ยังไม่ได้รับคำตอบ จำนวนการลองใหม่ค่าตีพอลต์คือ 1 หากจำนวน เป็นค่าลบหรือศูนย์ จำนวนการลองครั้งเปลี่ยนเป็นค่าตีพอลต์ 1

-s

แจ้งให้คำสั่ง *host9* ไม่ต้องส่งเคียวรีไปยังเซิร์ฟเวอร์ชื่อถัดไปหากมีเซิร์ฟเวอร์ใดตอบกลับ โดยมีการตอบกลับ `SERVFAIL`

-t *type*

เลือกชนิดเคียวรี ชนิดสามารถเป็นชนิดเคียวรีใดๆ ที่รู้จัก: `CNAME`, `NS`, `SOA` และอื่นๆ เมื่อไม่ระบุชนิดเคียวรี คำสั่ง *host9* จะเลือกชนิดเคียวรีที่เหมาะสมโดยอัตโนมัติ โดยค่าตีพอลต์ จะค้นหาเร็กคอร์ด `A` แต่หากคุณระบุแฟล็ก `-C` เคียวรีจะถูก ดำเนินการสำหรับเร็กคอร์ด `SOA` และหากชื่อเป็น IPv4 address ที่มีจุดทศนิยม หรือ IPv6 address ที่คั่นด้วยโคลอน คำสั่ง *host9* จะเคียวรีเร็กคอร์ด `PTR` หากเลือกชนิดเคียวรี `IXFR` คุณสามารถ ระบุเลขลำดับเริ่มต้นโดยการผนวกท้ายเครื่องหมายเท่ากับ ตามด้วย หมายเลขลำดับเริ่มต้น (ตัวอย่างเช่น `-t IXFR=12345678`)

-T

ใช้การเชื่อมต่อ TCP เมื่อเคียวรีเซิร์ฟเวอร์ชื่อ TCP จะถูกเลือกใช้โดยอัตโนมัติสำหรับเคียวรีที่จำเป็นต้องใช้ เช่นการร้องขอเพื่อถ่ายโอนโซน (`AXFR`)

-v

สร้างเอาต์พุตรายละเอียด แฟล็กนี้เท่ากับ แฟล็ก `-d`

-w

รอการตอบกลับไปตลอด เวลาที่จะรอ การตอบกลับจะถูกตั้งค่าเป็นจำนวนวินาทีที่กำหนดโดยค่าสูงสุดสำหรับจำนวน เลขจำนวนเต็มของฮาร์ดแวร์

-W *wait*

รอเป็นเวลา *wait* วินาที หากค่า *wait* น้อยกว่าหนึ่ง ช่วงเวลา การรอจะถูกตั้งค่าเป็น 1 วินาที

-4

บังคับให้คำสั่ง *host9* ใช้การถ่ายโอนเคียวรี IPv4 เท่านั้น

-6

บังคับให้คำสั่ง *host9* ใช้การถ่ายโอนเคียวรี IPv6 เท่านั้น

*name*

ระบุโดเมนเนมที่จะถูก ค้นหา โดยสามารถเป็น IPv4 address แบบจุดทศนิยมหรือ IPv6 address ที่คั่นด้วยโคลอน ไม่ว่าเป็นแบบใด คำสั่ง *host9* จะดำเนินการค้นหาแบบย้อนกลับสำหรับแอดเดรสนั้น

เซิร์ฟเวอร์

ระบุอาร์กิวเมนต์ทางเลือก ซึ่งอาจเป็น ชื่อหรือ IP แอดเดรสของเซิร์ฟเวอร์ชื่อที่คำสั่ง *host9* เคียวรีแทนเซิร์ฟเวอร์ที่แสดงรายการในไฟล์ `/etc/resolv.conf`

## การสนับสนุน IDN

ถ้าคำสั่ง *host9* ได้ถูกสร้างโดย มีการสนับสนุน internationalized domain name (IDN) คำสั่งสามารถรับและแสดง โดเมนเนม non-ASCII ได้ คำสั่ง *host9* แปลงการเข้ารหัสอักขระ ของโดเมนเนมอย่างเหมาะสม ก่อนการส่งการร้องขอไปที่เซิร์ฟเวอร์ DNS หรือแสดงการตอบกลับจากเซิร์ฟเวอร์ ถ้าคุณต้องการ ปิดการสนับสนุน IDN ให้กำหนดตัวแปรสถานะแวดล้อม `IDN_DISABLE` การสนับสนุน IDN ถูกปิดใช้งาน ถ้าตัวแปร ถูกเซตเมื่อคำสั่ง *host9* รัน

## ไฟล์

ไอเท็ม

คำอธิบาย

`/etc/resolv.conf`

## ตัวอย่าง

1. ในการแสดงแอดเดรสของเครื่องโฮสต์ชื่อ *mephisto* ป้อนคำสั่งต่อไปนี้:

```
host9 mephisto
```

คำสั่งนี้ แสดงข้อมูลคล้ายกับต่อไปนี้:

```
mephisto is 192.100.13.5, Aliases: engr, sarah
```

2. ในการแสดงเครื่องโฮสต์ที่มีแอดเดรส 192.100.13.1 ให้ป้อน คำสั่งต่อไปนี้:

```
host9 192.100.13.1
```

คำสั่งนี้ แสดงข้อมูลคล้ายกับต่อไปนี้:

```
mercutio is 192.100.13.1
```

3. ในการแสดงเร็กคอร์ด MX สำหรับโดเมนชื่อ test.ibm.com ให้ป้อนคำสั่งต่อไปนี้:

```
host9 -n -t mx test.ibm.com
```

คำสั่งนี้ แสดงข้อมูลคล้ายกับต่อไปนี้:

```
test.ibm.com mail is handled (pri=10) by test1.tt.ibm.com  
test.ibm.com mail is handled (pri=10) by test2.aix.ibm.com
```

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง named9

คำสั่ง nsupdate9

คำสั่ง rndc-confgen

คำสั่ง rndc.conf

---

## คำสั่ง hostent

### วัตถุประสงค์

จัดการรายการการแม็พแอดเดรสโดยตรงในฐานข้อมูลการกำหนดค่าระบบ

### ไวยากรณ์

ในการเพิ่มการแม็พแอดเดรส-กับ-ชื่อโฮสต์

```
hostent -a IPAddress -h "HostName..."
```

ในการลบการแม็พแอดเดรส-กับ-ชื่อโฮสต์

```
hostent -d IPAddress
```

ในการลบการแม็พแอดเดรส-กับ-ชื่อโฮสต์ทั้งหมด

```
hostent -X
```

ในการเปลี่ยนการแม็พแอดเดรส-กับ-ชื่อโฮสต์

```
hostent -c IPAddress -h "HostName..." [-i NewIPAddress ]
```

ในการแสดงแอดเดรสหรือชื่อโฮสต์ในรูปแบบโคลอน

```
hostent -s { IPAddress | "HostName" } [-Z ]
```

ในการแสดงการแม็พแอดเดรส-กับ-ชื่อโฮสต์ทั้งหมดในรูปแบบโคลอน

hostent -S [-Z]

## คำอธิบาย

คำสั่งระดับต่ำ **hostent** จะเพิ่ม ลบ หรือเปลี่ยนรายการการแม็พแอดเดรสในฐานข้อมูลการกำหนดค่า ระบบ รายการในฐานข้อมูลถูกใช้เพื่อแม็พ Internet Protocol (IP) address (โลคัลหรือรีโมต) กับชื่อโฮสต์ที่เทียบเท่ากัน

คำสั่ง **hostent** สามารถแสดง รายการการแม็พแอดเดรส-กับ-ชื่อโฮสต์หนึ่งหรือหลายรายการในไฟล์ `/etc/hosts` Internet Protocol (IP) address ของ โฮสต์โลคัลหรือรีโมตที่กำหนดอาจสัมพันธ์กับอย่างน้อยหนึ่ง ชื่อโฮสต์ โดยแสดง IP แอดเดรสในรูปแบบทศนิยมที่มีจุด แสดงชื่อโฮสต์เป็นสตริงที่มีความยาว สูงสุด 255 อักขระ และไม่ใช้อักขระช่องว่าง แต่ละรายการต้องอยู่ในหนึ่งบรรทัด โดยสามารถระบุหลาย *HostNames* (หรือ *aliases*)

**หมายเหตุ:** ชื่อโฮสต์หรือชื่อโฮสต์ alias ที่ถูกต้องต้องมี อักขระแบบตัวอักษรอย่างน้อยหนึ่งตัว หากคุณเลือกที่จะระบุชื่อโฮสต์หรือ alias ที่ขึ้นต้นด้วย x ตามด้วยเลขฐานสิบหกใดๆ (0-f) ชื่อโฮสต์หรือ alias ต้อง มีอักขระเพิ่มอย่างน้อยหนึ่งตัวที่ไม่สามารถถูกแสดงเป็นเลข ฐานสิบหกได้ ระบบจะแปลความหมาย x นำหน้าทีตามด้วย เลขฐานสิบหกว่าเป็นการแสดงค่าฐาน 16 ของแอดเดรสยกเว้น จะมีอย่างน้อยหนึ่งอักขระในชื่อโฮสต์ หรือ alias ที่ไม่ใช่ เลขฐานสิบหก ดังนั้น xdeer จะเป็นชื่อโฮสต์ที่ถูกต้อง ในขณะที่ xdee ไม่ใช่

คุณสามารถใช้พาดว่น **smit hostent** ของ System Management Interface Tool (SMIT) เพื่อรันคำสั่งนี้

## แฟล็ก

**หมายเหตุ:** แฟล็ก `-a`, `-d`, `-c` และ `-s` ไม่สามารถใช้รวมกัน

ไอเท็ม	คำอธิบาย
<code>-a IPAddress</code>	เพิ่มรายการการแม็พ IP address-to-host name สำหรับ Internet Protocol address ในฐานข้อมูล ระบุชื่อโฮสต์ด้วยแฟล็ก <code>-h</code>
<code>-c IPAddress</code>	เปลี่ยนแปลงรายการการแม็พ IP address-to-host name ในฐานข้อมูล ที่สอดคล้องกับแอดเดรสที่ระบุโดยตัวแปร <i>IPAddress</i> ระบุชื่อโฮสต์ที่เปลี่ยนแปลงด้วยแฟล็ก <code>-h</code> หากคุณต้องการเปลี่ยน IP แอดเดรสปัจจุบัน เป็นแอดเดรสใหม่ ( <i>IPAddress</i> ) ให้ใช้แฟล็ก <code>-i</code>
<code>-d IPAddress</code>	ลบรายการการแม็พ IP address-to-host name ในฐานข้อมูล ที่สอดคล้องกับแอดเดรสที่ระบุโดยตัวแปร <i>IPAddress</i>
<code>-h "HostName..."</code>	ระบุรายการชื่อโฮสต์ รายการในลิสต์จะถูก แบ่งด้วยช่องว่าง แฟล็ก <code>-h "HostName..."</code> ควรถูกใช้กับแฟล็ก <code>-a</code> แฟล็ก <code>-c</code> อาจต้องการ <code>-h "HostName..."</code>
<code>-i NewIPAddress</code>	ระบุ IP แอดเดรสใหม่ แฟล็กนี้จำเป็นต้องมีโดยแฟล็ก <code>-c</code> หาก IP แอดเดรสที่มีอยู่จะถูกแทนที่ด้วย ตัวแปร <i>NewIPAddress</i>
<code>-S</code>	แสดงรายการทั้งหมดในฐานข้อมูล
<code>-s "HostName"</code>	แสดงรายการการแม็พ IP address-กับ-ชื่อโฮสต์ที่ตรงกับ ชื่อโฮสต์ที่ระบุโดยตัวแปร <i>"HostName"</i>
<code>-s IPAddress</code>	แสดงรายการการแม็พ IP address-กับ-ชื่อโฮสต์ที่ตรงกับรายการ ที่ระบุโดยตัวแปร <i>IPAddress</i>
<code>-X</code>	ลบรายการการแม็พ IP address-กับ-ชื่อโฮสต์ทั้งหมดในฐานข้อมูล
<code>-Z</code>	สร้างรายงานของเคียวรีในรูปแบบโคโลน แฟล็กนี้ ใช้เมื่อคำสั่ง <b>hostent</b> เริ่มทำงานจากอินเตอร์เฟซ SMIT usability

**หมายเหตุ:** คำสั่ง **hostent** ไม่รู้จัก แอดเดรสต่อไปนี้: .08, .008, .09 และ .009 แอดเดรสที่มีศูนย์นำหน้า จะถูกแปลเป็นเลขฐานแปด และตัวเลขฐานแปดไม่สามารถมีเลข 8 หรือ 9

## ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the **lssecattr** command or the **getcmdattr** subcommand.

## ตัวอย่าง

1. ในการเพิ่มรายการในฐานข้อมูลที่เชื่อมโยง แอดเดรสกับชุดของชื่อโฮสต์ให้ป้อนคำสั่งในรูปแบบต่อไปนี้:

```
hostent -a 192.100.201.7 -h "alpha bravo charlie"
```

ในตัวอย่าง 1, IP แอดเดรส 192.100.201.7 ถูกระบุเป็นแอดเดรสของโฮสต์ที่มีชื่อโฮสต์หลัก เป็น alpha ที่มีความหมายเหมือน bravo และ charlie

2. ในการแสดงรายการในฐานข้อมูลที่ตรงกับชื่อโฮสต์ให้ป้อนคำสั่งในรูปแบบต่อไปนี้:

```
hostent -s alpha
```

ในตัวอย่าง 2 รายการที่แสดงจะตรงกับชื่อโฮสต์ alpha

3. ในการเปลี่ยน IP แอดเดรสของรายการเป็น IP แอดเดรสใหม่ให้ป้อนคำสั่งในรูปแบบต่อไปนี้:

```
hostent -c 192.100.201.7 -i 192.100.201.8
```

ในตัวอย่าง 3, IP แอดเดรสเก่าคือ 192.100.201.7 และแอดเดรสใหม่คือ 192.100.201.8

## ไฟล์

ไอเท็ม	คำอธิบาย
/etc/hosts	มีชื่อโฮสต์และแอดเดรสสำหรับเน็ตเวิร์ก

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง hostname” ในหน้า 802

ข้อมูลที่เกี่ยวข้อง:

การระบุชื่อ TCP/IP

---

## คำสั่ง hostid

### วัตถุประสงค์

ตั้งค่าหรือแสดง identifier ของโลคัลโฮสต์ปัจจุบัน

### ไวยากรณ์

```
/usr/sbin/hostid [ HexNumber | InternetAddress | HostName ]
```

## คำอธิบาย

คำสั่ง `/usr/sbin/hostid` แสดง identifier (อาจเป็นชื่อโฮสต์เฉพาะหรืออาร์กิวเมนต์ตัวเลข) ของโลคัลโฮสต์ปัจจุบันเป็นเลขฐานสิบหก ค่าตัวเลขนี้จะต้องเป็น ค่าเฉพาะในโฮสต์ทั้งหมดและถูกตั้งค่าร่วมกันเป็นแอดเดรสของโฮสต์ ที่ระบุโดยพารามิเตอร์ `InternetAddress` หรือ `HostName` ผู้ใช้ `root` สามารถตั้งค่าคำสั่ง `hostid` โดยการระบุเลขฐานสิบหกสำหรับพารามิเตอร์ `HexNumber`, `InternetAddress` หรือ `HostName` identifier โฮสต์ถูกตั้งค่าเป็นชื่อโฮสต์โดยไฟล์ `/etc/rc.net`

## พารามิเตอร์

ไอเท็ม	คำอธิบาย
<code>HexNumber</code>	ระบุเลขฐานสิบหกเฉพาะที่แสดงโลคัลโฮสต์ปัจจุบัน
<code>InternetAddress</code>	ระบุอินเทอร์เน็ตแอดเดรสที่แสดงโลคัลโฮสต์ปัจจุบัน
<code>HostName</code>	ระบุชื่อสัญลักษณ์ที่แม่กับโฮสต์เฉพาะ

## ตัวอย่าง

- ในการตั้งค่า identifier ของโลคัลโฮสต์เป็น โลคัลอินเทอร์เน็ตแอดเดรสด้วยคำสั่ง `hostid` ให้ป้อน คำสั่งในรูปแบบต่อไปนี้:

```
hostid 192.9.200.3
0xc009c803
```

คำสั่ง `hostid` แปลงอินเทอร์เน็ต แอดเดรส 192.9.200.3 เป็นการแสดงในรูปเลขฐานสิบหก 0xc009c803 จากนั้นตั้งค่าโลคัลโฮสต์ (เวิร์กสเตชันของคุณ ที่เชื่อมต่อกับเน็ตเวิร์ก) กับแอดเดรสนี้

- ในการแสดง identifier ของโลคัลโฮสต์ให้ป้อน:

```
hostid
0xc009c803
```

คำสั่ง `hostid` แสดง identifier ของโฮสต์เป็นตัวเลขฐานสิบหก

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `hostname`” ในหน้า 802

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `gethostid`

คำสั่ง `sethostid`

การกำหนดแอดเดรส TCP/IP

---

## hostmibd Daemon

### วัตถุประสงค์

เริ่มทำงาน `hostmibd dpi2 sub-agent daemon` เป็นแบบการประมวลผลส่วนหลัง

### ไวยากรณ์

```
hostmibd [-f File] [-d [Level]] [-h Hostname] [-c Community]
```

## คำอธิบาย

คำสั่ง `hostmibd` เริ่มทำงานเอเจนต์ย่อย `hostmibd dpi2` คำสั่งนี้เรียกใช้ได้เพียงผู้ใช้ที่มี `root privileges` หรือเป็นสมาชิกของกลุ่มระบบ

`hostmibd` daemon เป็นไปตามมาตรฐาน Simple Network Management Protocol Distributed Protocol Interface เวอร์ชัน 2.0 ที่กำหนดโดย RFC 1592 โดยทำหน้าที่เป็นเอเจนต์ย่อย `dpi2` เพื่อสื่อสารกับเอเจนต์ `dpi2` ผ่าน `dpiPortForTCP.0 (1.3.6.1.4.1.2.2.1.1.1.0)` ซึ่งกำหนดใน RFC1592 section 3.1

Management Information Base (MIB) ถูกกำหนดโดย RFC 1155 ตัวแปร MIB ที่ระบุที่ `hostmibd` กำหลังจัดการถูกกำหนดโดย RFC 2790 ตัวแปร MIB จริงที่จัดการโดย `hostmibd` คือหกทรีย่อยต่อไปนี้:

- `hrSystem (1.3.6.1.2.1.25.1)`
- `hrStorage (1.3.6.1.2.1.25.2)`
- `hrDevice (1.3.6.1.2.1.25.3)`
- `hrSWRun (1.3.6.1.2.1.25.4)`
- `hrSWRunPerf (1.3.6.1.2.1.25.5)`
- `hrSWInstalled (1.3.6.1.2.1.25.6)`

โดยปกติ `hostmibd` daemon ถูกเรียกใช้งาน ระหว่างที่เริ่มทำงานระบบเมื่อเซลล์สคริปต์ `/etc/rc.tcpip` ถูกเรียกใช้

`hostmibd` daemon ควรถูกควบคุม โดยใช้ System Resource Controller (SRC) ไม่แนะนำให้ป้อน `hostmibd` ที่บรรทัดคำสั่ง ใช้คำสั่ง SRC ต่อไปนี้เพื่อดำเนินการกับ `hostmibd` daemon:

### startsrc

เริ่มต้นระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย

**stopsrc** หยุดระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย

**refresh** ทำให้ระบบย่อยหรือกลุ่มของระบบย่อยอ่าน ไฟล์คอนฟิกูเรชันอีกครั้ง

**lssrc** รับข้อมูลสถานะของระบบย่อย กลุ่มของระบบย่อย หรือเซิร์ฟเวอร์ย่อย หากผู้ใช้ที่เรียกใช้รูปแบบสถานะแบบยาวของคำสั่ง `lssrc` ไม่ใช่ผู้ใช้ `root` จะไม่แสดงข้อมูลชื่อ `community`

## แฟล็ก

### ไอเท็ม

`-c Community`

### คำอธิบาย

ใช้ชื่อ `community` ที่ระบุ หากไม่ระบุแฟล็ก `-c` ชื่อ `community` ดีฟอลต์คือ `'public'`

ไอเท็ม -d Level	คำอธิบาย ระบุระดับการติดตาม/ดีบั๊ก ระดับได้แก่: <ul style="list-style-type: none"> <li>• 0 = Least level</li> <li>• 8 = DPI level 1</li> <li>• 16 = DPI level 2</li> <li>• 32 = Internal level 1</li> <li>• 64 = Internal level 2</li> <li>• 128 = Internal level 3</li> </ul> <p>เพิ่มตัวเลขสำหรับระดับการติดตามหลายระดับ ระดับดีฟอลต์ คือ 56 หากระบุแฟล็ก -d แต่ไม่ระบุ Level หากไม่ระบุแฟล็ก -d ระดับดีฟอลต์เป็น 0</p>
-f File	ระบุไฟล์คอนฟิกูเรชันไม่มีใช้ค่าดีฟอลต์ หากไม่ระบุแฟล็ก -f ไฟล์คอนฟิกูเรชัน ดีฟอลต์คือ /etc/hostmibd.conf โปรดดูที่ไฟล์ /etc/hostmibd.conf เพื่อดูข้อมูลเกี่ยวกับรูปแบบไฟล์นี้
-h Host	ส่งการร้องขอไปยังโฮสต์ที่ระบุค่า Host สามารถเป็น IPv4 address, IPv6 address หรือชื่อโฮสต์ หากไม่ระบุแฟล็ก -h โฮสต์ปลายทางดีฟอลต์ จะเป็น 'loopback' (127.0.0.1)

## ตัวอย่าง

- เมื่อต้องการเริ่มทำงาน **hostmibd** daemon ให้ป้อนคำสั่ง เหมือนดังต่อไปนี้:

```
startsrc -s hostmibd -a "-f /tmp/hostmibd.conf"
```

คำสั่งนี้ เริ่มทำงาน **hostmibd** daemon และอ่าน ไฟล์คอนฟิกูเรชันจาก **/tmp/hostmibd.conf**

- เมื่อต้องการหยุดทำงาน **hostmibd** daemon โดยปกติให้ป้อน:

```
stopsrc -s hostmibd
```

คำสั่งนี้ หยุดทำงาน **hostmibd** daemon แฟล็ก -s ระบุระบบย่อยที่ติดตามเพื่อให้ หยุดทำงาน

- เมื่อต้องการดูสถานะแบบสั้นๆ จาก **hostmibd** ให้ป้อน:

```
lssrc -s hostmibd
```

คำสั่งนี้ ส่งกลับชื่อของ daemon ID กระบวนการของ daemon และสถานะของ daemon (แอ็คทีฟหรือไม่แอ็คทีฟ)

- เมื่อต้องการรับค่าสถานะแบบยาวจาก **hostmibd** daemon ให้ป้อน:

```
lssrc -ls hostmibd
```

หากคุณเป็นผู้ใช้ root รูปแบบยาว ของรายงานสถานะนี้จะแสดงรายการพารามิเตอร์การกำหนดค่าใน **/etc/hostmibd.conf**

## ไฟล์

ไอเท็ม /etc/hostmibd.conf /etc/mib.defs	คำอธิบาย กำหนดพารามิเตอร์การกำหนดค่าสำหรับคำสั่ง <b>hostmibd</b> กำหนดตัวแปร Management Information Base (MIB) ที่เอเจนต์และตัวจัดการ SNMP ควรรู้จักและจัดการ
---	---

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง snmpdv3

คำสั่ง snmpmibd

---

## คำสั่ง hostname

### วัตถุประสงค์

ตัวค่าหรือแสดงชื่อของระบบโฮสต์ปัจจุบัน

### ไวยากรณ์

```
/usr/bin/hostname [ HostName ] [ -s ]
```

### คำอธิบาย

คำสั่ง `/usr/bin/hostname` แสดงชื่อของระบบโฮสต์ปัจจุบัน เฉพาะผู้ใช้ที่มีสิทธิ์ผู้ใช้ `root` เท่านั้น ที่สามารถตั้งค่าชื่อโฮสต์ คำสั่ง `mkdev` และคำสั่ง `chdev` ยังตั้งค่าชื่อโฮสต์ถาวร ใช้คำสั่ง `mkdev` เมื่อคุณกำลังกำหนดอินสแตนซ์ TCP/IP เป็น ครั้งแรก

คุณสามารถใช้พารามิเตอร์ `smit mkhostname` ของ System Management Interface Tool (SMIT) เพื่อรันคำสั่งนี้

### แฟล็ก

ไอเท็ม คำอธิบาย  
-s ตัดส่วนข้อมูลโดเมนใดๆ ออกจากชื่อที่พิมพ์

### พารามิเตอร์

ไอเท็ม คำอธิบาย  
*HostName* ตั้งค่าชื่อหลักของโฮสต์

หมายเหตุ: คุณต้องมีสิทธิ์ผู้ใช้ `root` เพื่อใช้พารามิเตอร์ *HostName*

### ความปลอดภัย

**Attention RBAC users and Trusted AIX users:** This command can perform privileged operations. Only privileged users can run privileged operations. For more information about authorizations and privileges, see Privileged Command Database in *Security*. For a list of privileges and the authorizations associated with this command, see the `lssecattr` command or the `getcmdattr` subcommand.

#### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `chdev`

คำสั่ง `mkdev`

คำสั่ง `gethostname`

การระบุชื่อ TCP/IP

---

## คำสั่ง hosts2ldif

### วัตถุประสงค์

สร้างไฟล์ LDAP Data Interchange Format (LDIF) จากโฮสต์ไฟล์

### ไวยากรณ์

```
hosts2ldif [ -i InputFile ] [ -o OutputFile ] [ -s SearchBase ]
```

### คำอธิบาย

คำสั่ง `/usr/sbin/hosts2ldif` สร้างไฟล์ LDAP Data Interchange Format (LDIF) จาก `/etc/hosts` หรือไฟล์อื่นที่คล้ายกับ `/etc/hosts` เมื่อไม่มีแฟล็ก `/etc/hosts` จะถูกใช้เพื่อสร้างไฟล์ `/tmp/hosts.ldif` LDIF โดยใช้ `cn=hosts` เป็น baseDN

ไฟล์ LDIF ที่สร้างโดยคำสั่งนี้จะเข้ากันได้กับ SecureWay Directory Schema และใช้สำหรับการตั้งค่ากลไก `ldap` กลไก `ldap` ได้รับการสนับสนุน แต่แนะนำให้ใช้กลไก `nis_ldap` มากกว่ากลไก `ldap`

### แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-i InputFile</code>	ระบุโฮสต์ไฟล์ที่ใช้เป็นอินพุต
<code>-o OutputFile</code>	ระบุไฟล์ LDIF ที่ใช้เป็นเอาต์พุต
<code>-s SearchBase</code>	ระบุ baseDN ของตารางโฮสต์บนเซิร์ฟเวอร์ LDAP

### ตัวอย่าง

- ในการสร้าง `/home/ldifhosts` จากไฟล์ `/etc/hosts` ให้พิมพ์:  

```
hosts2ldif -o /home/ldifhosts
```
- ในการสร้าง `/tmp/hosts.ldif` จากไฟล์ `/home/hosts.bak` ให้พิมพ์:  

```
hosts2ldif -i /home/hosts.bak
```
- ในการสร้าง `/home/ldifhosts` จากไฟล์ `/etc/hosts` โดยใช้ `cn=hoststab` เป็น baseDN ให้พิมพ์:  

```
hosts2ldif -o /home/ldifhosts -s cn=hoststab
```

### ไฟล์

ไอเท็ม	คำอธิบาย
<code>/etc/hosts</code>	มีชื่อ Internet Protocol (IP) และแอดเดรสของโฮสต์บนโลคอลเน็ตเวิร์ก

### ข้อมูลที่เกี่ยวข้อง:

การระบุชื่อ TCP/IP

---

## คำสั่ง hp

### วัตถุประสงค์

จัดการฟังก์ชันพิเศษสำหรับเทอร์มินัลรุ่น HP2640 และ HP2621

### ไวยากรณ์

hp [-e] [-m...]

### คำอธิบาย

คำสั่ง **hp** อ่านอินพุตมาตรฐาน (โดยปกติเป็นเอาต์พุตจากคำสั่ง **nroff**) และเขียนไปยังเอาต์พุตมาตรฐาน ซึ่งโดยปกติเป็นหน้าจอเทอร์มินัล Hewlett-Packard รุ่น 2640 และ 2621

หากเทอร์มินัลของคุณมีคุณลักษณะการเพิ่มประสิทธิภาพการแสดงผล คุณสามารถแสดงอักขระตัวห้อยและอักขระตัวยกด้วยคุณลักษณะ สัญลักษณ์ทางคณิตศาสตร์ คุณสามารถแสดงอักขระภาษากรีกและอักขระพิเศษอื่นๆ ได้โดยมีข้อยกเว้นสองข้อ คำสั่ง **hp** จะแสดงให้เห็นตัวดำเนินการเชิงตรรกะ NOT ด้วยลูกศรขวาและแสดงเฉพาะครึ่งบนของ เครื่องหมายอินทิกรัล

ขีดทับอักขระคืออักขระที่ตามด้วย อักขระถอยกลับและอักขระอื่น โดยแสดงเป็นถูกขีดเส้นใต้หรือที่แสดงกลับด้าน (ขึ้นกับการเพิ่มประสิทธิภาพเทอร์มินัล) หากอักขระที่ถูกเขียนทับหรือ อักขระที่พิมพ์หลังการกดแป้นถอยกลับ (backspace) เป็น อักขระ underscore

หมายเหตุ: ลำดับอักขระควบคุมบางลำดับ (reverse line-feeds และ backspaces) สามารถทำให้ข้อมูลไม่แสดง ตารางที่มีเส้นแนวดิ่งที่สร้างโดยคำสั่ง **tbl** อาจเป็นบรรทัดของข้อความที่มี ส่วนล่างของเส้นแนวดิ่ง คุณอาจสามารถหลีกเลี่ยงปัญหาได้โดยอันดับแรกการไฟฟ์อินพุตผ่านคำสั่ง **col** จากนั้นผ่านคำสั่ง **hp**

### แฟล็ก

ไอเท็ม คำอธิบาย

- e แสดงอักขระที่ขีดทับที่ขีดเส้นใต้ ด้วยก ใน half-bright และอักขระตัวห้อยใน half-bright ที่ขีดเส้นใต้ มิฉะนั้น อักขระที่ขีดทับทั้งหมด อักขระตัวห้อย และอักขระตัวยก จะปรากฏในลักษณะกลับด้านสี (dark-on-light) ใช้แฟล็กนี้ต่อเมื่อจอแสดงผลของคุณ มีคุณลักษณะการเพิ่มประสิทธิภาพการแสดงผล
- m สร้างบรรทัดว่างหนึ่งบรรทัดเท่านั้นสำหรับบรรทัดว่างที่ตามมาหลายๆ บรรทัด ในข้อความ

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง eqn” ในหน้า 419

“คำสั่ง greek” ในหน้า 735

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง col

คำสั่ง nroff

คำสั่ง tbl

---

## คำสั่ง hplj

### วัตถุประสงค์

ประมวลผลเอาต์พุตคำสั่ง **troff** สำหรับเครื่องพิมพ์ HP LaserJet Series ภายหลัง

### ไวยากรณ์

```
hplj [-F Directory] [-quietly] [-landscape] [File ...]
```

### คำอธิบาย

คำสั่ง **hplj** ประมวลผล เอาต์พุตของคำสั่ง **troff** สำหรับเอาต์พุตไปยังเครื่องพิมพ์ Hewlett-Packard LaserJet Series

หากกำหนดหนึ่งหรือหลายไฟล์เป็นอ็อปชัน คำสั่ง **hplj** จะประมวลผลไฟล์เหล่านั้น หากไม่ระบุไฟล์ จะเป็น ตัวกรองการแปล อินพุตมาตรฐาน พารามิเตอร์ *File* ระบุไฟล์ที่คำสั่ง **hplj** ประมวลผล เอาต์พุตของเครื่องพิมพ์ HP LaserJet Series

**หมายเหตุ:** คำสั่ง **hplj** สามารถใช้คาร์ทริดจ์ K หรือคาร์ทริดจ์ Text-Equations ถ้าติดตั้ง ในเครื่องพิมพ์ (คาร์ทริดจ์ข้อความ-สมการ หมายเลขชิ้นส่วน HP C2053A #C07 แทนที่คาร์ทริดจ์ K) ดีฟอลต์ฟอนต์ไฟล์ถือว่าเป็นหนึ่งในคาร์ทริดจ์ เมื่อถูกติดตั้ง หากคุณไม่มีคาร์ทริดจ์ K ให้ใช้ฟอนต์แบบบิตแมป ที่ดาวน์โหลดแทน ในการคำสั่งนี้ให้รับคำสั่ง **no\_cart** ในฟอนต์ได้เรียกทอริ สำหรับเครื่องพิมพ์ HP (`/usr/lib/font/devhplj`)

เอาต์พุตที่ไม่ถูกต้องสามารถเกิดขึ้นได้หากฟอนต์ไฟล์ซึ่งคุณถือว่า คาร์ทริดจ์ถูกเมาท์เมื่อจริงๆ แล้วไม่ใช่เอาต์พุตที่ไม่ถูกต้อง ยังอาจเกิดขึ้น หากคาร์ทริดจ์ หรือซอฟต์แวร์ฟอนต์อื่นๆ ถูกติดตั้ง นอกเหนือจากคาร์ทริดจ์ K หรือคาร์ทริดจ์ข้อความ-สมการ

คำสั่ง **hplj** จะขึ้นอยู่กับ ไฟล์ที่มีชื่อลงท้ายด้วย **.out** ในไฟล์ `/usr/lib/font/devhplj` คำสั่งนี้ไม่สร้างเอาต์พุตที่เหมาะสมยกเว้น ไฟล์ได้รับการตั้งค่าอย่างเหมาะสม โปรดดูที่เอกสารฟอนต์ไฟล์ **troff** เพื่อข้อมูลเพิ่มเติม

### แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-FDirectory</code>	ระบุไดเรกทอรีที่ระบุเป็นตำแหน่งในการค้นหา ฟอนต์ไฟล์ โดยค่าดีฟอลต์ คำสั่ง <b>hplj</b> จะค้นหาฟอนต์ไฟล์ ในไดเรกทอรี <code>/usr/lib/font/devhplj</code>
<code>-quietly</code>	ไม่แสดงข้อความแสดงความผิดพลาดที่ไม่ร้ายแรงทั้งหมด

ไอเท็ม	คำอธิบาย
<code>-landscape</code>	พิมพ์ไฟล์ที่ระบุในรูปแบบแนวนอน หน้าแนวนอน ถูกจัดแนวสำหรับการอ่านปกติ ความกว้างของหน้าใหญ่กว่า ความยาว โดยค่าดีฟอลต์ คำสั่ง <b>hplj</b> จะพิมพ์ในการวาง แนวตั้ง
	<b>หมายเหตุ:</b> แนวนอนจะใช้ได้เฉพาะ ในฟอนต์ Courier บนเครื่องพิมพ์ Hewlett-Packard Jet II เท่านั้น ดังนั้น เอกสาร <b>troff</b> ต้องได้รับการจัดรูปแบบด้วยฟอนต์ Courier ในการ ทำให้แทรกบรรทัดต่อไปนี้อยู่ที่ตอนต้นของอินพุตไฟล์ <b>troff</b> :

```
.fp 1 C
.fp 2 C
.fp 3 CB
```

ฟอนต์ Courier ถูกโหลดขึ้นมาไว้บนตำแหน่งฟอนต์ #1 & #2 และ Courier-Bold บนตำแหน่ง #3

## ตัวอย่าง

1. ในการพิมพ์ไฟล์ troff ชื่อ foo บนเครื่องพิมพ์ hp โดยใช้คำสั่ง lp ให้ป้อน:

```
troff -mm -Thplj foo | hplj | lp -dhp -o -dp
```

2. ในการพิมพ์ไฟล์ troff ชื่อ boo บนเครื่องพิมพ์ hp โดยใช้คำสั่ง qprt ให้ป้อน:

```
troff -mm -Thplj boo | hplj | qprt -dp -Php
```

หมายเหตุ: แฟล็ก **-dp** ในตัวอย่างทั้งสอง ส่งข้อมูลเครื่องพิมพ์ไปยังอุปกรณ์เครื่องพิมพ์ในโหมด pass-through (ไม่แก้ไข)

## File

ไอเท็ม

/usr/lib/font/devhpl/\*.out

คำอธิบาย

มีไฟล์ฟอนต์

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง troff

ฟอนต์ troff

---

## คำสั่ง hpmcount

### วัตถุประสงค์

วัดผลการทำงานแอปพลิเคชัน

### ไวยากรณ์

```
hpmcount [ -a ] [ -b time_base ] [ -d ] [ -D metrics ] [ -g event_groups ] [ -H ] [ -k ] [ -m metrics_groups ] [ -o file ] [ -s set ] [ -x ] command
```

```
hpmcount [-h]
```

### คำอธิบาย

คำสั่ง **hpmcount** จัดให้มีเวลาในการเรียกใช้งาน ข้อมูลตัวนับผลการทำงานฮาร์ดแวร์ เมตริกฮาร์ดแวร์ที่สืบทอด และสถิติการใช้ประโยชน์รีซอร์ส (ได้จากการเรียกใช้ระบบ `getrusage()`) สำหรับแอปพลิเคชันที่ระบุชื่อโดย *command*

ชนิดเหตุการณ์ที่จะถูกมอนิเตอร์และตัวนับผลการทำงานฮาร์ดแวร์ ที่สัมพันธ์กันจะถูกระบุโดยการตั้งค่าอ็อปชัน **-s** โดยการระบุชื่อกลุ่มเหตุการณ์ หมายเลข หรือรายการที่ค้นด้วยเครื่องหมายจุลภาค ของหมายเลขชุดในตัวแปรสภาวะแวดล้อม **HPM\_EVENT\_SET** หรือโดยการระบุตัวนับ/เหตุการณ์ POWER3 / PowerPC 604 RISC Microprocessor) หรือชื่อกลุ่มเหตุการณ์ (POWER4 และใหม่กว่า) ในอินพุตไฟล์ `libHPM_events` (มาก่อน **HPM\_EVENT\_SET**) แต่ละชุดสามารถถูกทำให้เหมาะสมโดยโหมดการนับ หมายเลขหรือชื่อกลุ่มเหตุการณ์ สามารถระบุโดยการตั้งค่าอ็อปชัน **-g** หรือการระบุรายการที่ค้นด้วยเครื่องหมายจุลภาคของกลุ่มเหตุการณ์ใน ตัวแปรสภาวะแวดล้อม **HPM\_EVENT\_GROUP** ในลักษณะเดียวกัน แต่ละกลุ่มเหตุการณ์สามารถถูกทำให้เหมาะสมโดยโหมดการนับ

หมายเลขชุดเหตุการณ์ที่ใช้ได้จะมีค่าตั้งแต่ 1 ถึงขีดจำกัดบน ที่ขึ้นอยู่กับชนิดตัวประมวลผล ซึ่งสามารถแสดงรายการได้โดยใช้คำสั่ง `pmlist` รายการที่คั่นด้วยเครื่องหมายจุลภาคของชุดเหตุการณ์สามารถระบุแทน หมายเลขชุด ซึ่งแต่ละกรณีโหมดการ multiplex ตัวนับจะถูกเลือก ในการเลือกชุดเหตุการณ์ทั้งหมด ให้ตั้งค่าหมายเลขเป็น 0

รายการที่คั่นด้วยเครื่องหมายจุลภาคของเมตริกที่สืบทอดสามารถระบุโดยการตั้งค่า อ็อพชัน `-D` แต่ละเมตริกที่สืบทอดสามารถถูกทำให้เหมาะสม โดยโหมดการนับ

รายการของกลุ่มเมตริกที่สืบทอดสามารถถูกระบุโดยการตั้งค่าอ็อพชัน `-m` หรือโดยการระบุรายการที่คั่นด้วยเครื่องหมายจุลภาค ของกลุ่มเมตริกที่สืบทอดในตัวแปรสถานะแวดล้อม `HPM_PMD_GROUP` ค่านี้อนุญาตให้เลือกเมตริกที่สืบทอดทั้งหมด ที่เป็นของกลุ่มที่ระบุ แต่ละกลุ่มเมตริกสามารถถูกทำให้เหมาะสม ได้โดยโหมดการนับ

กิจกรรมระบบและไฮเปอร์ไวเซอร์ (สำหรับตัวประมวลผลที่สนับสนุนโหมดไฮเปอร์ไวเซอร์) สามารถถูกรวมในการนับโดยการระบุอ็อพชัน `-k` และ `-H`

เมื่อทำการนับในโหมด multiplex ผลลัพธ์ต้องถูกทำให้เป็นมาตรฐาน เพื่อทำไปใช้ฐานดีฟอลต์ที่ใช้สำหรับการทำข้อมูลเป็นมาตรฐานคือ ฐานเวลา อ็อพชัน `-b` อนุญาตสำหรับการใช้เวลา PURR หรือเวลา SPURR (เมื่อตัวประมวลผลสนับสนุน) สำหรับการทำให้ข้อมูลเป็นมาตรฐาน ฐานสำหรับการทำฐานให้เป็นมาตรฐาน ยังสามารถกำหนดโดยใช้ตัวแปรสถานะแวดล้อม `HPM_NORMALIZE`

ผลลัพธ์สามารถเอาต์พุตในรูปแบบ XML โดยใช้อ็อพชัน `-x`

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-a</code>	รวมค่าตัวนับเมื่อ POE รัน
<code>-b time_base</code>	เลือกฐานสำหรับการทำข้อมูลให้เป็นมาตรฐาน ฐานที่มีอยู่ มีดังนี้:  เวลา      ฐานเวลา  <code>purr</code> เวลา PURR (เมื่อมีอยู่)  <code>spurr</code> เวลา SPURR (เมื่อมีอยู่)
<code>-d</code>	ค่าดีฟอลต์คือ <code>time</code>
<code>-D metrics</code>	เพิ่มการนับชุดรายละเอียดสำหรับโหมดการ multiplex ตัวนับ เลือกรายการของเมตริกที่สืบทอดที่จะถูกวิเคราะห์ค่า แต่ละเมตริก ที่สืบทอดสามารถถูกทำให้เหมาะสมได้โดยโหมดการนับดังนี้:  <code>metric:counting_modes</code>
<code>-g event_groups</code>	(โปรดดูที่ อ็อพชัน <code>-m</code> สำหรับโหมดการนับที่มีใช้ได้) แสดงรายการกลุ่มของเหตุการณ์ที่กำหนดไว้แล้วหรือรายการที่คั่นด้วยเครื่องหมายจุลภาค ของชื่อหรือหมายเลขกลุ่มเหตุการณ์ เมื่อใช้รายการที่คั่นด้วยเครื่องหมายจุลภาคของ กลุ่ม โหมดการ multiplex ตัวนับจะถูกเลือก แต่ละกลุ่มเหตุการณ์ สามารถถูกทำให้เหมาะสมได้โดยโหมดการนับ ดังนี้:  <code>event_group:counting_modes</code>
<code>-H</code>	(โปรดดูที่ อ็อพชัน <code>-m</code> สำหรับโหมดการนับที่มีใช้ได้)
<code>-h</code>	เพิ่มกิจกรรมไฮเปอร์ไวเซอร์ในนามของกระบวนการ
<code>-k</code>	แสดงข้อความวิธีใช้
	เพิ่มกิจกรรมระบบในนามของกระบวนการ

## ไอเท็ม

`-m metrics_groups`

### คำอธิบาย

เลือกรายการของกลุ่มเมตริกที่สืบทอดที่จะถูกวิเคราะห์ค่า กลุ่ม เมตริกที่สืบทอดอ้างอิงเมตริกที่สืบทอดทั้งหมดที่ไม่ได้เป็นของ กลุ่มเมตริกที่สืบทอดที่ระบุ แต่ละกลุ่มเมตริกสามารถถูกทำให้เหมาะสม ได้โดยโหมดการนับ ดังนี้:

`metric_group_name:counting_modes`

โหมดการนับที่มี ดังนี้:

**u** โหมดผู้ใช้

**k** โหมดเคอร์เนล

**h** โหมดไฮเปอร์ไวเซอร์

**r** โหมดrunlatch

**n** โหมดnointerrupt

ชื่อเอาต์พุตไฟล์

`-o file`

`-s set`

แสดงรายการชุดที่กำหนดไว้แล้วของเหตุการณ์หรือรายการที่ค้นด้วยเครื่องหมายจุดภาค ของชุด (1 ถึง N หรือ 0 เพื่อเลือก ทั้งหมด ดูที่คำสั่ง `pmlist`) เมื่อใช้รายการที่ค้นด้วยเครื่องหมายจุดภาคของคุณ โหมดการ multiplex ตัวนับจะถูกเลือก แต่ละชุดสามารถถูกทำให้เหมาะสมได้โดยโหมดการนับ ดังนี้:

`event_set:counting_modes`

(โปรดดูที่ อ็อพชัน `-m` สำหรับโหมดการนับที่มีใช้ได้)

`-x`

แสดงผลลัพธ์ในรูปแบบ XML

## พารามิเตอร์

### ไอเท็ม

`command`

### คำอธิบาย

ระบุโปรแกรมที่เรียกใช้งานที่จะทำการวัด ผลการทำงาน

## ตัวแปรสถานะแวดล้อม

ตัวแปรสถานะแวดล้อมต่อไปนี้มีผลโดยตรงต่อการเรียกใช้งาน ของคำสั่ง `hpmcount` (มีตัวแปรสถานะแวดล้อม `MP_*` เพิ่มเติมที่มีผลต่อการเรียกใช้งานของโปรแกรมขนาน)

### ไอเท็ม

`HPM_EVENT_SET`

### คำอธิบาย

เลือกหนึ่งในชุดเหตุการณ์ ค่าสามารถ เป็นเลขจำนวนเต็มตั้งแต่ 1 ถึง 6 บนระบบ POWER3 เป็น 1 ถึง 4 บนระบบ PowerPC 604 RISC Microprocessor หรือเป็น 1 ถึง ซัดจำกัดบนตามตัวประมวลผลบนระบบ POWER4 และใหม่กว่า ตัวแปรสถานะแวดล้อมนี้ยังสามารถใช้เพื่อเลือก ชื่อกลุ่มเหตุการณ์บนระบบ POWER4 และใหม่กว่า รายการที่ค้นด้วย เครื่องหมายจุดภาคของชุดเหตุการณ์สามารถถูกระบุ ในกรณีนี้ โหมดการ multiplex ตัวนับจะถูกเลือก แต่ละชุดเหตุการณ์สามารถถูกทำให้เหมาะสมได้โดยโหมดการนับ ดังนี้:

`event_set_number:counting_modes`

`HPM_EVENT_GROUP`

อ็อพชัน `-g` หรือ `-s` มาก่อน ตัวแปรนี้ ตัวแปรสถานะแวดล้อม `HPM_EVENT_GROUP` มาก่อนตัวแปรนี้

เลือกกลุ่มเหตุการณ์ รายการที่ค้นด้วยเครื่องหมายจุดภาคของกลุ่มเหตุการณ์ สามารถถูกระบุ ในกรณีนี้ โหมดการ multiplex ตัวนับจะถูกเลือก แต่ละกลุ่มเหตุการณ์ สามารถถูกทำให้เหมาะสมได้โดยโหมดการนับ ดังนี้:

`event_group_number:counting_modes`

อ็อพชัน `-g` หรือ `-s` มาก่อน ตัวแปรนี้ ตัวแปรสถานะแวดล้อม `HPM_EVENT_GROUP` มาก่อนตัวแปร `HPM_EVENT_SET`

ไอเท็ม	คำอธิบาย
HPM_NORMALIZE	จัดให้มีฐานที่ใช้สำหรับการทำข้อมูลให้เป็นมาตรฐาน อีพซัน -b มาก่อนตัวแปรนี้
HPM_PMD_GROUP	ระบุรายการที่ค้นด้วยเครื่องหมายจุลภาคของกลุ่มเมตริกที่สืบทอด แต่ละกลุ่มเมตริกสามารถถูกทำให้เหมาะสมได้โดยโหมดการนับ อีพซัน -m มาก่อนตัวแปรนี้
HPM_PMD_METRIC	ระบุรายการที่ค้นด้วยเครื่องหมายจุลภาคของเมตริกที่สืบทอด แต่ละเมตริกที่สืบทอดสามารถถูกทำให้เหมาะสมโดยโหมดการนับ อีพซัน -D มาก่อนตัวแปรนี้
HPM_DIV_WEIGHT	กำหนดน้ำหนัก (เลขจำนวนเต็มที่มากกว่า 1) เพื่อใช้คำนวณการพลิกที่มีค่าน้ำหนักบนระบบ
MP_CHILD	POWER4 ใช้ในสภาวะแวดล้อมแบบสภาวะแวดล้อมเมื่อการนับ สะสมถูกระบุเพื่อเสิร์ชไฟล์ผลลัพธ์เอาต์พุตให้สมบูรณ์ ( <i>myID</i> ), ซิงโครไนซ์การตรวจเทียบผลลัพธ์ และระบุข้อความวินิจฉัยรายละเอียด/ดีบั๊ก ได้ใกล้เคียงมากยิ่งขึ้น
MP_PROCS	จำนวนงานโปรแกรม
HPM_AGGREGATE_OUTPUT	รวมการนับบนแอฟพลิเคชัน POE (บังคับ ใช้อาร์กิวเมนต์บรรทัดคำสั่ง -a) โดยใช้แฟล็กนี้ ไฟล์ผลการทำงาน ไฟล์เดียวจะถูกสร้างขึ้นสำหรับงานทั้งหมด คำนี้อาจได้กับ POE หรือ Load Leveller เท่านั้น
HPM_LOG_DIR	และจำเป็นต้องมีความพร้อมใช้ของระบบไฟล์ แบบขนาน (เช่น GPFS) บนระบบ เมื่อตั้งค่าแฟล็กนี้ <code>hpmcount</code> จะเขียน ไฟล์ <code>hpm_log.id</code> ที่มีข้อมูลผลการทำงานลงใน ไดเรกทอรีที่จัดเตรียมให้ นี้เป็นเอาต์พุตนอกเหนือจากเอาต์พุตปกติ
MP_PARTITION	บนแอฟพลิเคชัน POE <i>id</i> คือ POE ID ที่จัดเตรียมให้โดย <code>MP_PARTITION</code> มิฉะนั้น จะเป็น <i>pid</i> รวมทั้งชื่อการล็อกภายในและไฟล์ข้อมูล
HPM_MX_DURATION	เมื่อการนับอยู่ในโหมดการ multiplex ตัวนับ แฟล็กนี้จะระบุช่วงเวลาของแต่ละส่วนเวลา โดยแสดงเป็น ms และต้องอยู่ในช่วง 10 ms - 30 s เมื่อไม่ตั้งค่า แฟล็กนี้จะใช้ค่าดีฟอลต์สำหรับระยะเวลาแบ่งเวลา คือ 100 ms

นอกจากนั้น ตัวแปรสภาวะแวดล้อมต่อไปนี้ที่กำหนดโดย ผู้ใช้ จะระบุการประมาณการหน่วยความจำ แคช และเวลาแฝง TLB miss สำหรับการคำนวณเมตริกที่สืบทอด ตัวแปรสภาวะแวดล้อมเหล่านี้ ไม่มีการมาก่อนการประมาณเดียวกันที่จัดให้มีในไฟล์ `HPM_flags.env` หากมีแสดงอยู่

- HPM\_MEM\_LATENCY
- HPM\_L3\_LATENCY
- HPM\_L35\_LATENCY
- HPM\_AVG\_L3\_LATENCY
- HPM\_AVG\_L2\_LATENCY
- HPM\_L2\_LATENCY
- HPM\_L25\_LATENCY
- HPM\_L275\_LATENCY
- HPM\_L1\_LATENCY
- HPM\_TLB\_LATENCY

## สถานะออก

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. ในการรันคำสั่ง `ls` และเขียนข้อมูลที่เกี่ยวข้องกับ เหตุการณ์ในชุด 5 จากตัวนับฮาร์ดแวร์ ให้อ่าน:

```
hpmcount -s 5 ls
```

- ในการรันคำสั่ง `ls` และเขียนข้อมูลเกี่ยวกับ เหตุการณ์ในชุด 5, 2 และ 9 จากตัวนับฮาร์ดแวร์โดยใช้โหมด การ multiplex ตัวนับ ให้ป้อน:

```
hpmcount -s 5,2,9 ls
```

- ในการรันคำสั่ง `ls` และรายงานเมตริกที่สืบทอดที่เกี่ยวข้องกับกลุ่มเมตริกดีพอลต์และ `cpi_breakdown` ที่ถูกนับตามลำดับ ในโหมด `kernel+user+hypervisor` และโหมดผู้ใช้ให้ป้อน:

```
hpmcount -m ตีพอลต์:kuh,cpi_breakdown:u ls
```

## ข้อมูลจำเพาะการนำไปใช้งาน

คำสั่ง `hpmcount` ใช้ API ระดับเธรด PMAPI

พารามิเตอร์ `hpmcount command` ไม่ถูกวิเคราะห์ค่า เป็นบรรทัดคำสั่งสำหรับชื่อแอ็พพลิเคชันที่มีอ้อพชัน แต่ ต้องสร้าง เซลล์สคริปต์ที่มีบรรทัดคำสั่งแทน

### Location

`/usr/bin/perf/pmapi/hpmcount`

### อินพุตมาตรฐาน

ไม่ใช่

### เอาต์พุตมาตรฐาน

ผลลัพธ์การมอนิเตอร์ผลการทำงานถูกเขียนไปยัง `stdout` ยกเว้นอ้อพชัน `-o file` ถูกระบุบนบรรทัดคำสั่ง

### ข้อผิดพลาดมาตรฐาน

ใช้สำหรับข้อความวินิจฉัยเท่านั้น

## ไฟล์

อินพุตไฟล์ต่อไปนี้จะถูกใช้หากมีอยู่

ไอเท็ม

libHPM\_events

คำอธิบาย

ชุดไฟล์เหตุการณ์ที่ผู้ใช้ระบุ ไฟล์นี้ไม่มีความสำคัญเหนือบรรทัดคำสั่งที่ระบุด้วยอ็อปชัน `-s` รูปแบบสำหรับคู่ POWER3/PowerPC 604 RISC

Microprocessor counter/event คือ *counternumber eventname* ตัวอย่างเช่น:

```
0 PM_LD_MISS_L2HIT
1 PM_TAG_BURSTRD_L2MISS
2 PM_TAG_ST_MISS_L2
3 PM_FPUO_DENORM
4 PM_LSU_IDLE
5 PM_LQ_FULL
6 PM_FPU_FMA
7 PM_FPU_IDLE
```

รายการ ที่ค้นด้วยเครื่องหมายจุดของเหตุการณ์สามารถถูกระบได้เช่นกัน  
ค่านี้เปิดทำงานโหมดการ multiplex ตัวนับ:

```
0 PM_CYC, PM_FPU_FIN, PM_IC_MISS
1 PM_LD_CMPL, PM_INST_CMPL, PM_DC_MISS
2 PM_INST_CMPL, PM_FPU_WT, PM_INST_CMPL
3 PM_LD_MISS_DC_XU, PM_CYC, PM_CYC
```

สำหรับชื่อกลุ่มเหตุการณ์ POWER4 รูปแบบคือ *event\_group\_name* ตัวอย่างเช่น:

pm\_hpmcount1

รายการ ที่ค้นด้วยเครื่องหมายจุดของเหตุการณ์สามารถถูกระบได้เช่นกัน  
ค่านี้เปิดทำงานโหมดการ multiplex ตัวนับ:

pm\_hpmcount1, pm\_hpmcount2, pm\_basic

ไฟล์ที่มีคู่ตัวแปรสภาวะแวดล้อม/ค่าที่ใช้สำหรับการคำนวณเมตริกที่สืบทอด  
ตัวอย่างเช่น:

```
HPM_L2_LATENCY 12
HPM_EVENT_SET 5
```

ล๊อคไฟล์ ไฟล์นี้ถูกสงวนไว้สำหรับการใช้ภายในของคำสั่ง **hpmcount**

ไฟล์ผลลัพธ์สะสม ไฟล์นี้ถูกสงวนไว้สำหรับการใช้ภายในของคำสั่ง **hpmcount**

HPM\_flags.env

`./hpm_lockfile_mp_partition`

`./hpm_datafile_mp_partition`

เอาต์พุตไฟล์ต่อไปนี้จะถูกใช้

ไอเท็ม

file\_myID.pid

HPM\_LOG\_DIR/hpm\_log.MP\_PARTITION or HPM\_LOG\_DIR/

hpm\_log.pid

`./hpm_lockfile_mp_partition`

`./hpm_datafile_mp_partition`

คำอธิบาย

ไฟล์ที่ระบุด้วยอ็อปชัน `-o` สำหรับผลลัพธ์เอาต์พุต **hpmcount** โดยที่ *myID*

ถูกนำมาจากตัวแปรสภาวะแวดล้อม **MP\_CHILD** ที่มีค่าดีฟอลต์ 0000

ล๊อคไฟล์ที่ระบุสำหรับตัวนับค่าสะสมเมื่อ POE รัน

ล๊อคไฟล์ ไฟล์นี้ถูกสงวนไว้สำหรับการใช้ภายในของคำสั่ง **hpmcount**

ไฟล์ผลลัพธ์สะสม ไฟล์นี้ถูกสงวนไว้สำหรับการใช้ภายในของคำสั่ง **hpmcount**

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `pmlist`

คำสั่ง `getrusage`

คำสั่ง pm\_initialize

Performance Monitor API Programming

---

## คำสั่ง hpmstat

### วัตถุประสงค์

แสดงข้อมูลตัวนับผลการทำงานฮาร์ดแวร์ระดับระบบ

### ไวยากรณ์

```
hpmstat [-b time_base] [-d] [-D metrics] [-g event_groups] [-H] [-k] [-m metrics_groups] [-o file] [-r] [-s set] [-T] [-U] [-u] [-x] [-@ ALL | WparName] interval count
```

**hpmstat [-h]**

### คำอธิบาย

คำสั่ง **hpmstat** จัดให้มีเวลาพักการแสดงผล การเรียกใช้งาน ข้อมูลตัวนับผลการทำงานฮาร์ดแวร์ และเมตริกฮาร์ดแวร์ ที่สืบทอดใช้ได้จากผู้ใช้ที่มีสิทธิ์ root เท่านั้น

เมื่อระบุโดยไม่มีอ็อปชันบรรทัดคำสั่ง **hpmstat** จะนับ การวนซ้ำ 1 ดีฟอลต์ของกิจกรรมของผู้ใช้ เคอร์เนลและ ไฮเปอร์ไวเซอร์ (สำหรับตัวประมวลผลที่สนับสนุนโหมดไฮเปอร์ไวเซอร์) เป็นเวลา 1 วินาที สำหรับชุดดีฟอลต์ 1 ของเหตุการณ์ จากนั้นเขียนค่าตัวนับที่ได้และเมตริกที่สืบทอดไปยังเอาต์พุตมาตรฐาน โดยค่าดีฟอลต์ **runlatch** ถูกปิดใช้งานเพื่อให้การนับสามารถดำเนินการได้ขณะเรียกใช้งานในวงรอบที่ไม่ได้ทำงาน

เมื่อ อ็อปชัน **-U** ระบุ *interval* เป็น ไมโครวินาที *count* การทำซ้ำจะ ไม่มีสิ้นสุด และเมตริกที่สืบทอดไม่ถูกคำนวณและเขียนไปยังเอาต์พุต มาตรฐาน อ็อปชันนี้ ถูกข้ามหากระบุโหมดการ multiplex ตัวนับ

เมื่อระบุอ็อปชัน **-T** ข้อมูลเอาต์พุต จะถูกนำหน้าด้วยการประทับเวลา (วินาทีและไมโครวินาที) และข้อมูล เวลาจะถูกเขียนเป็นการประทับเวลาแทนเวลาเป็นวินาที

ชนิดเหตุการณ์ที่จะมอนิเตอร์ และตัวนับผลการทำงานฮาร์ดแวร์ ที่สัมพันธ์กันจะถูกระบุโดยใช้อ็อปชันชุด **-s** หรือ โดยการระบุชื่อกลุ่มเหตุการณ์ หรือหมายเลขชุดในตัวแปรสถานะแวดล้อม **HPM\_EVENT\_SET** หนึ่ง อีกทางหนึ่ง ระบุคู่ตัวนับ/เหตุการณ์ (POWER3 / PowerPC 604 RISC Microprocessor) หรือชื่อกลุ่มเหตุการณ์ (POWER4 และใหม่กว่า) ในอินพุตไฟล์ **libHPM\_events** (มาก่อน **HPM\_EVENT\_SET**) แต่ละชุดสามารถถูกทำให้เหมาะสมโดยโหมดการนับ หมายเลขหรือชื่อกลุ่มเหตุการณ์ สามารถระบุโดยการตั้งค่าอ็อปชัน **-g** หรือการระบุรายการที่คั่นด้วยเครื่องหมายจุลภาคของกลุ่มเหตุการณ์ในตัวแปรสถานะแวดล้อม **HPM\_EVENT\_GROUP** ในลักษณะเดียวกัน แต่ละกลุ่มเหตุการณ์สามารถถูกทำให้เหมาะสมโดยโหมดการนับ

รายการที่คั่นด้วยเครื่องหมายจุลภาคของชุดเหตุการณ์สามารถระบุแทน หมายเลขชุด ซึ่งแต่ละกรณีโหมดการ multiplex ตัวนับจะถูกเลือก ในการเลือกชุดเหตุการณ์ทั้งหมด ให้ตั้งค่าหมายเลขชุดเป็น 0

หมายเลขชุดเหตุการณ์ที่ใช้ได้จะมีค่าตั้งแต่ 1 ถึงขีดจำกัดบน ที่ขึ้นอยู่กับชนิดตัวประมวลผล ซึ่งสามารถแสดงรายการได้โดยใช้คำสั่ง **pmlist**

รายการที่ค้นด้วยเครื่องหมายจุลภาคของเมตริกที่สืบทอดสามารถระบุโดยการตั้งค่า อ็อพชัน **-D** แต่ละเมตริกที่สืบทอดสามารถถูกทำให้เหมาะสม โดยโหมดการนับ

รายการของกลุ่มเมตริกที่สืบทอดสามารถถูกระบุโดยการตั้งค่าอ็อพชัน **-m** หรือโดยการระบุรายการที่ค้นด้วยเครื่องหมายจุลภาค ของกลุ่มเมตริกที่สืบทอดในตัวแปรสภาวะแวดล้อม **HPM\_PMD\_GROUP** คำนี้อนุญาตให้เลือกเมตริกที่สืบทอดทั้งหมด ที่เป็นของกลุ่มที่ระบุ แต่ละกลุ่มเมตริกสามารถถูกทำให้เหมาะสม ได้โดยโหมดการนับ

เมื่อทำการนับในโหมด multiplex ผลลัพธ์ต้องถูกทำให้เป็นมาตรฐาน เพื่อทำไปใช้ฐานดีฟอลต์ที่ใช้สำหรับการทำให้ข้อมูลเป็นมาตรฐานคือ ฐานเวลา อ็อพชัน **-b** อนุญาตสำหรับการใช้ เวลา PURR หรือเวลา SPURR (เมื่อตัวประมวลผลสนับสนุน) สำหรับการทำให้ข้อมูลเป็นมาตรฐาน ฐานสำหรับการทำฐานให้เป็นมาตรฐาน ยังสามารถกำหนดโดยใช้ตัวแปรสภาวะแวดล้อม **HPM\_NORMALIZE**

เมื่อคุณรันคำสั่ง **hpmstat** จาก global workload partition (WPAR) จะสามารถมอนิเตอร์ WPAR ที่เจาะจงโดยใช้อ็อพชัน **-@ WparName** คุณสามารถใช้อ็อพชัน **-@ ALL** เพื่อมอนิเตอร์ WPARs ที่แฉีกที่พทั้งหมดในระบบและเพื่อเรียกข้อมูลของ WPAR

ผลลัพธ์สามารถเอาต์พุตในรูปแบบ XML โดยใช้อ็อพชัน **-x**

## แฟล็ก

### ไอเท็ม

**-@ ALL | WparName**

### คำอธิบาย

เลือก WPAR ปลายทางที่กิจกรรม จะถูกวัดค่า คำ ALL หมายความว่า คำสั่ง **hpmstat** วัดค่า WPARs แฉีกที่พทั้งหมด ในระบบและรายงานกิจกรรมของแต่ละ WPAR อ็อพชันนี้ที่ใช้ต่อเมื่อคุณรันคำสั่ง **hpmstat** จาก WPAR โกลบอล มิฉะนั้นจะถูกข้ามเลือกฐานสำหรับการทำข้อมูลให้เป็นมาตรฐาน ฐานที่มีอยู่ มีดังนี้:

**-b time\_base**

เวลา ฐานเวลา

**purrr** เวลา PURR (เมื่อมีอยู่)

**spurr** เวลา SPURR (เมื่อมีอยู่)

**-d**

ค่าดีฟอลต์คือ **time**

**-D metrics**

เพิ่มการนับชุดรายละเอียดสำหรับโหมดการ multiplex ตัวนับ

เลือกรายการของเมตริกที่สืบทอดที่จะถูกวิเคราะห์ค่า แต่ละเมตริก ที่สืบทอดสามารถถูกทำให้เหมาะสมได้โดยโหมดการนับดังนี้:

**metric:counting\_modes**

**-g event\_groups**

(โปรดดูที่ อ็อพชัน **-m** สำหรับโหมดการนับที่มีใช้ได้)

แสดงรายการกลุ่มของเหตุการณ์ที่กำหนดไว้แล้วหรือรายการที่ค้นด้วยเครื่องหมายจุลภาค ของชื่อหรือหมายเลขกลุ่มเหตุการณ์ เมื่อใช้รายการที่ค้นด้วยเครื่องหมายจุลภาคของ กลุ่ม โหมดการ multiplex ตัวนับจะถูกเลือก แต่ละกลุ่มเหตุการณ์ สามารถถูกทำให้เหมาะสมได้โดยโหมดการนับ ดังนี้:

**event\_group:counting\_modes**

**-H**

(โปรดดูที่ อ็อพชัน **-m** สำหรับโหมดการนับที่มีใช้ได้)

**-h**

นับกิจกรรมไฮเปอร์ไวเซอร์เท่านั้น

**-k**

แสดงข้อความวิธีใช้

นับกิจกรรมระบบเท่านั้น

## ไอเท็ม

`-m metrics_groups`

## คำอธิบาย

เลือกรายการของกลุ่มเมตริกที่สืบทอดที่จะถูกวิเคราะห์ค่า กลุ่ม เมตริกที่สืบทอดดีฟอลต์ อังอิงเมตริกที่สืบทอดทั้งหมดที่ไม่ได้เป็นของ กลุ่มเมตริกที่สืบทอดที่ระบุ แต่ละกลุ่มเมตริกสามารถถูกทำให้เหมาะสมได้โดยโหมดการนับ ดังนี้:

`metric_group_name:counting_modes`

โหมด การนับที่มีที่ดังนี้:

**u** โหมดผู้ใช้

**k** โหมดเคอร์เนล

**h** โหมดไฮเปอร์ไวเซอร์

**r** โหมด runlatch

**n** โหมด nointerrupt

ชื่อเอาต์พุตไฟล์

`-o file`

`-r`

`-s set`

เปิดใช้งาน `runlatch` และปิดใช้งานการนับ ขณะเรียกใช้งานในวงรอบที่ไม่ได้ทำงาน แสดงรายการชุดที่กำหนดไว้แล้วของเหตุการณ์หรือรายการที่ค้นด้วยเครื่องหมายจุลภาคของชุด (1 ถึง  $N$  หรือ 0 เพื่อเลือก ทั้งหมด ดูที่คำสั่ง `pmlist`) เมื่อใช้รายการที่ค้นด้วยเครื่องหมายจุลภาคของคุณ โหมดการ multiplex ตัวนับจะถูกเลือก แต่ละชุดสามารถถูกทำให้เหมาะสมได้โดยโหมดการนับ ดังนี้:

`event_set:counting_modes`

(โปรดดูที่ อ็อพชัน `-m` สำหรับโหมดการนับที่มีใช้ได้)

`-T`

`-U`

`-u`

`-x`

เขียนการประทับเวลาแทนเวลาเป็นวินาที

ใส่ช่วงเวลาการนับเป็นไมโครวินาที อ็อพชันนี้ถูกข้ามหากระบุโหมดการ multiplex ตัวนับกิจกรรมผู้ใช้เท่านั้น

แสดงผลลัพธ์ในรูปแบบ VPA XML

## พารามิเตอร์

### ไอเท็ม

`interval`

`count`

### คำอธิบาย

แสดงช่วงเวลาการนับเป็นวินาที หรือไมโครวินาที ที่มีค่าดีฟอลต์เป็น 1 แสดงจำนวนการวนซ้ำเพื่อนับ ค่า ดีฟอลต์คือ 1 ที่มีช่วงเวลาเป็นวินาที และไม่สิ้นสุด เมื่อระบุอ็อพชัน `-U`

## ตัวแปรสถานะแวดล้อม

ตัวแปรสถานะแวดล้อมต่อไปนี้จะมีผลโดยตรงต่อการเรียกใช้งาน ของคำสั่ง `hpmstat` (มีตัวแปรสถานะแวดล้อม `MP_*` เพิ่มเติม ที่มีผลต่อการเรียกใช้งานของโปรแกรมขนาน)

### ไอเท็ม

`HPM_EVENT_SET`

### คำอธิบาย

เลือกหนึ่งในชุดเหตุการณ์ ค่าสามารถ เป็นเลขจำนวนเต็มตั้งแต่ 1 ถึง 6 บนระบบ POWER3 เป็น 1 ถึง 4 บนระบบ PowerPC 604 RISC Microprocessor หรือเป็น 1 ถึง ซัดจำกัดบนตามตัวประมวลผลบนระบบ POWER4 และใหม่กว่า ตัวแปรสถานะแวดล้อมนี้ยังสามารถใช้เพื่อเลือก ชื่อกลุ่มเหตุการณ์บนระบบ POWER4 และใหม่กว่า แต่ละชุดเหตุการณ์สามารถถูกทำให้เหมาะสมได้โดยโหมดการนับ ดังนี้:

`event_set_number:counting_modes`

อ็อพชัน `-g` หรือ `-s` มาก่อน ตัวแปรนี้ ตัวแปรสถานะแวดล้อม `HPM_EVENT_GROUP` มาก่อน ตัวแปรนี้

## ไอเท็ม

HPM\_EVENT\_GROUP

## คำอธิบาย

เลือกกลุ่มเหตุการณ์ รายการที่คั่นด้วยเครื่องหมายจุลภาคของกลุ่มเหตุการณ์ สามารถถูกระบุ ในกรณีนี้ โหมดการ multiplex ตัวนับจะถูกเลือก แต่ละกลุ่มเหตุการณ์ สามารถถูกทำให้เหมาะสมได้ โดยโหมดการนับ ดังนี้:

event\_group\_number:counting\_modes

HPM\_NORMALIZE

HPM\_PMD\_GROUP

HPM\_PMD\_METRIC

HPM\_DIV\_WEIGHT

HPM\_MX\_DURATION

อ็อปชัน -g หรือ -s มาก่อน ตัวแปรนี้ ตัวแปรสภาวะแวดล้อม HPM\_EVENT\_GROUP มาก่อน

ตัวแปร HPM\_EVENT\_SET

จัดให้มีฐานที่จะใช้สำหรับการทำข้อมูลให้เป็นมาตรฐาน อ็อปชัน -b มาก่อนตัวแปรนี้ ระบุรายการที่คั่นด้วยเครื่องหมายจุลภาคของกลุ่มเมตริกที่สืบทอด แต่ละกลุ่มเมตริกสามารถถูกทำให้เหมาะสมได้โดยโหมดการนับ อ็อปชัน -m มาก่อนตัวแปรนี้

ระบุรายการที่คั่นด้วยเครื่องหมายจุลภาคของเมตริกที่สืบทอด แต่ละเมตริกที่สืบทอดสามารถถูกทำให้เหมาะสมโดยโหมดการนับ อ็อปชัน -D มาก่อนตัวแปรนี้

กำหนดน้ำหนัก (เลขจำนวนเต็มที่มากกว่า 1) เพื่อใช้คำนวณการพลิกที่มีค่าน้ำหนักบนระบบ

POWER4

เมื่อการนับอยู่ในโหมดการ multiplex ตัวนับ แฟล็กนี้จะระบุช่วงเวลาของแต่ละส่วนเวลา โดยแสดงเป็น ms และต้องอยู่ในช่วง 10 ms - 30 s เมื่อไม่ตั้งค่า แฟล็กนี้จะใช้ค่าดีฟอลต์สำหรับระยะเวลาแบ่งเวลา คือ 100 ms

นอกจากนั้น ตัวแปรสภาวะแวดล้อมต่อไปนี้ ที่กำหนดโดยผู้ใช้ จะระบุการประมาณการหน่วยความจำ แคช และเวลาแฝง TLB miss สำหรับการคำนวณเมตริกที่สืบทอด ตัวแปรสภาวะแวดล้อมเหล่านี้ ไม่มีการมาก่อนการประมาณเดียวกันที่จัดให้มีในไฟล์ HPM\_flags.env หากมีแสดงอยู่

- HPM\_MEM\_LATENCY
- HPM\_L3\_LATENCY
- HPM\_L35\_LATENCY
- HPM\_AVG\_L3\_LATENCY
- HPM\_AVG\_L2\_LATENCY
- HPM\_L2\_LATENCY
- HPM\_L25\_LATENCY
- HPM\_L275\_LATENCY
- HPM\_L1\_LATENCY
- HPM\_TLB\_LATENCY

## สถานะออก

### ไอเท็ม

0

>0

### คำอธิบาย

ดำเนินการเป็นผลสำเร็จ

มีข้อผิดพลาดเกิดขึ้น

## ตัวอย่าง

1. ในการเขียนข้อมูลสำหรับกิจกรรมระบบ ผู้ใช้ และไฮเปอร์ไวเซอร์ ในช่วงเวลา 1 วินาทีเกี่ยวกับเหตุการณ์ในชุด 2 จากตัวนับ ฮาร์ดแวร์ ป้อนคำสั่งต่อไปนี้:

```
hpmstat -s 2
```

2. ในการเขียนข้อมูลกิจกรรมผู้ใช้เกี่ยวกับเหตุการณ์ของ กลุ่ม 0 และกิจกรรมระบบเกี่ยวกับเหตุการณ์ของกลุ่ม 1 สำหรับ wpar1 WPAR ในช่วงเวลาหาวินาที ป้อนคำสั่งต่อไปนี้:

```
hpmstat -@ wpar1 -g 0:u,1:k 5
```

## Location

/usr/bin/perf/pmapi/hpmstat

## อินพุตมาตรฐาน

ไม่ใช่

## เอาต์พุตมาตรฐาน

ผลลัพธ์การมอนิเตอร์ผลการทำงานถูกเขียนไปยัง `stdout` ยกเว้นอ็อปชัน `-o file` ถูกระบุบนบรรทัดคำสั่ง

## ข้อผิดพลาดมาตรฐาน

ใช้สำหรับข้อความวินิจฉัยเท่านั้น

## ไฟล์

อินพุตไฟล์ต่อไปนี้จะถูกใช้หากมีอยู่

### ไอเท็ม

libHPM\_events

### คำอธิบาย

ชุดไฟล์เหตุการณ์ที่ผู้ใช้ระบุ ไฟล์นี้ไม่มีความสำคัญเหนือบรรทัดคำสั่งที่ระบุด้วยอ็อปชัน `-s` รูปแบบสำหรับคู่ POWER3/PowerPC 604 RISC Microprocessor counter/event คือ `counternumber eventname` ตัวอย่างเช่น:

- 0 PM\_LD\_MISS\_L2HIT
- 1 PM\_TAG\_BURSTRD\_L2MISS
- 2 PM\_TAG\_ST\_MISS\_L2
- 3 PM\_FPUO\_DENORM
- 4 PM\_LSU\_IDLE
- 5 PM\_LQ\_FULL
- 6 PM\_FPU\_FMA
- 7 PM\_FPU\_IDLE

สำหรับชื่อกลุ่มเหตุการณ์ POWER4 รูปแบบคือ `event_group_name` ตัวอย่างเช่น:

pm\_hpmcount1

ไฟล์ที่มีคู่ตัวแปรสภาวะแวดล้อม/ค่าที่ใช้สำหรับการคำนวณเมตริกที่สืบทอด ตัวอย่างเช่น:

- HPM\_L2\_LATENCY 12
- HPM\_EVENT\_SET 5

HPM\_flags.env

เอาต์พุตไฟล์ต่อไปนี้จะถูกใช้

ไอเท็ม  
file

คำอธิบาย  
ไฟล์ที่ระบุด้วยอ็อปชัน -o สำหรับผลลัพธ์เอาต์พุต hpmstat

สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง hpmcount” ในหน้า 806

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง pmlist

คำสั่ง pm\_initialize

Performance Monitor API Programming

---

## คำสั่ง hps\_dump

### วัตถุประสงค์

ดัมพ์เนื้อหาของหน่วยความจำอะแดปเตอร์ Network Terminal Accelerator (NTX) ไปยังโฮสต์ไฟล์

### ไวยากรณ์

```
hps_dump [ -f Name ] [ -d Device ]
```

### คำอธิบาย

คำสั่ง `hps_dump` ใช้ อินเทอร์เน็ตเฟสตัวโหลดเพื่ออัปโหลดหน่วยความจำทั้งหมดจากอะแดปเตอร์บอร์ดไปยังไฟล์ ซึ่งจะสร้างสำเนาของระบบสำหรับการวิเคราะห์ และการดีบั๊กในภายหลัง 1024 ไบต์แรกของไฟล์มีไอเท็ม ต่อไปนี้:

ไอเท็ม	คำอธิบาย
80	สตริง Identification รวมถึงเวอร์ชัน
80	เวลาและวันที่ของการดัมพ์หน่วยความจำจากระบบโฮสต์
80	ความคิดเห็น
268	ตารางบันทึกการทำงานจากโฮสต์อะแดปเตอร์
32	ตารางแอดเดรสระบบ
8	ช่วงของแอดเดรสเริ่มต้นและสิ้นสุดของดัมพ์หน่วยความจำ
476	การเสริมเต็มให้เป็น 1024 ไบต์เต็ม

### แฟล็ก

ไอเท็ม	คำอธิบาย
-f Name	ระบุชื่อของดัมพ์หน่วยความจำ ใช้อ็อปชันนี้เพื่อ แทนที่ชื่อไฟล์ดีฟอลต์ ./hpscore
-d Device	ระบุชื่อไฟล์อุปกรณ์ดิบบของอะแดปเตอร์ ใช้อ็อปชันนี้ เพื่อแทนที่ชื่ออุปกรณ์ดีฟอลต์ /dev/rhp0

### สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม คำอธิบาย  
0 ดำเนินการเป็นผลสำเร็จ  
>0 มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: คุณต้องมีสิทธิ์ root ในการรัน คำสั่งนี้

เหตุการณ์การตรวจสอบ: ไม่สามารถใช้ได้

## ตัวอย่าง

1. เมื่อต้องการรับดัมพ์หน่วยความจำของดีฟอลต์อะแดปเตอร์ไปยังไฟล์ `hpscore` ใน ไดรฟ์ทอริปัจจุบัน ให้ป้อน:

```
hps_dump
```

2. เมื่อต้องการรับดัมพ์หน่วยความจำของอะแดปเตอร์ดีฟอลต์ไปยังไฟล์ `hpsdebug` ใน ไดรฟ์ทอริปัจจุบันของอะแดปเตอร์ดีฟอลต์ ให้ป้อน:

```
hps_dump -f hpsdebug
```

3. เมื่อต้องการรับดัมพ์หน่วยความจำของหน่วยความจำของอะแดปเตอร์ `/dev/rhp1` ไปยังไฟล์ `hpsdebug` ในไดรฟ์ทอริปัจจุบันของดีฟอลต์อะแดปเตอร์ ให้ป้อน:

```
hps_dump -f hpsdebug -d /dev/rhp1
```

## ไฟล์

ไอเท็ม	คำอธิบาย
<code>/usr/bin/hps_dump</code>	มีคำสั่ง <code>hps_dump</code>
<code>/dev/rhp0</code>	ชื่อไฟล์อุปกรณ์ NTX raw ดีฟอลต์

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `/dev/rhp`

---

## คำสั่ง `htable`

### วัตถุประสงค์

แปลงโฮสต์ไฟล์ไปเป็นรูปแบบที่ใช้โดยเน็ตเวิร์ก โลบราจิริวทีน

### ไวยากรณ์

```
/usr/sbin/htable [ -c connected-nets ] [ -l local-nets ] input-file
```

หมายเหตุ: อย่าใส่ช่องว่าง บนฝั่งใดฝั่งหนึ่งของเครื่องหมายจุลภาค

### คำอธิบาย

คำสั่ง `htable` แปลงโฮสต์ไฟล์ในรูปแบบที่ระบุใน RFC 810 เป็นรูปแบบที่ใช้โดยเน็ตเวิร์ก โลบราจิริวทีน การแปลงจะสร้างสามไฟล์: ไฟล์ `/etc/hosts`, ไฟล์ `/etc/networks` และไฟล์ `/etc/gateways`

รูทีนย่อย `gethostbyname` ใช้ไฟล์ `hosts` เพื่อทำการแม็ปชื่อโฮสต์เป็นแอดเดรสเมื่อไม่ใช้ `named daemon` รูทีนย่อย `getnetent` ใช้ไฟล์ `networks` เพื่อทำการแม็ปชื่อเน็ตเวิร์กเป็นหมายเลข

ไฟล์ `gateways` อาจถูกใช้โดย `routed daemon` ในการระบุแพสซีฟอินเตอร์เน็ตเกตเวย์

หากมีไฟล์ `hosts`, `networks` หรือ `gateways` ใดๆ (`localhosts`, `localnetworks` หรือ `localgateways` ตามลำดับ) ในไดเรกทอรีปัจจุบัน เนื้อหาของไฟล์ถูกเพิ่มเป็นส่วนนำหน้าของไฟล์เอาต์พุตสำหรับไฟล์เหล่านี้ โปรแกรม `htable` จะแปรความหมายเฉพาะไฟล์ `gateways` การเพิ่มส่วนนำหน้าเข้ากับเนื้อหา อนุญาตให้ใช้คังกรักรายการโลคัลที่โดยทั่วไปไม่มีอยู่ในฐานข้อมูลต้นฉบับ

## แฟล็ก

### ไอเท็ม

`-c connected-nets`

### คำอธิบาย

ระบุรายการของเน็ตเวิร์กที่โฮสต์จะเชื่อมต่อโดยตรง หาก `daemon` การจัดส่งเส้นทางเน็ตเวิร์กใช้ไฟล์ `gateways` คั้นเน็ตเวิร์กด้วยเครื่องหมายจุลภาค และใช้ชื่อเน็ตเวิร์ก หรือรูปแบบจุดอินเตอร์เน็ตมาตรฐาน (ตัวอย่างเช่น `-c arpanet, 128.32, LocalEthernet`) คำสั่ง `htable` รวมเฉพาะเกตเวย์ที่เชื่อมต่อโดยตรงกับเน็ตเวิร์กหนึ่งในเน็ตเวิร์กที่ระบุ หรือที่สามารถเข้าถึงได้จากเกตเวย์อื่นบนเน็ตเวิร์กที่เชื่อมต่อกับเน็ตเวิร์กสำหรับคำสั่ง `htable` เพื่อทำหน้าที่เป็นโลคัล รับข้อมูลเกี่ยวกับโฮสต์บนโลคัลเน็ตเวิร์กจาก ไฟล์ `localhosts` เท่านั้น คั้นเน็ตเวิร์กด้วยเครื่องหมายจุลภาค และใช้ชื่อเน็ตเวิร์กหรือรูปแบบจุดอินเตอร์เน็ตมาตรฐาน (ตัวอย่างเช่น `-l 128.32, local-ether-net`) รายการสำหรับโลคัลโฮสต์ จากฐานข้อมูลหลักถูกข้ามเพื่อให้ไฟล์ `localhosts` สามารถแทนที่รายการในอินพุตไฟล์ (ไฟล์ที่คุณระบุบน บรรทัดคำสั่ง)

`-l local-nets`

## ไฟล์

### ไอเท็ม

`/CurrentDirectory/localgateways`

`/CurrentDirectory/localhosts`

`/CurrentDirectory/localnetworks`

### คำอธิบาย

มีข้อมูลโลคัลเกตเวย์

มีข้อมูลชื่อโลคัลโฮสต์

มีข้อมูลโลคัลเน็ตเวิร์ก

### สิ่งอ้างอิงที่เกี่ยวข้อง:

“คำสั่ง `gettable`” ในหน้า 722

### ข้อมูลที่เกี่ยวข้อง:

คำสั่ง `routed`

ไฟล์เครือข่าย

เกตเวย์การจัดเส้นทาง TCP/IP

---

## คำสั่ง `hty_load`

### วัตถุประสงค์

แสดงหรือดาวน์โหลดการกำหนดค่าอะแดปเตอร์ Network Terminal Accelerator (NTX)

### ไวยากรณ์

`hty_load [ -d Device ] [ -f ConfigFileName ]`

## คำอธิบาย

คำสั่ง `hty_load` แสดง หรือดาวน์โหลดการกำหนดค่าอะแดปเตอร์ หากคุณเรียกใช้คำสั่งนี้โดยไม่ระบุแฟล็กใดๆ ระบบจะแสดงการกำหนดค่าอะแดปเตอร์ปัจจุบันสำหรับไฟล์อุปกรณ์ `/dev/rhp0` เมื่อกำหนดพารามิเตอร์ `Device` คำสั่ง `hty_load` จะโหลดไฟล์คอนฟิกูเรชันเข้ามาในไดรเวอร์ `tty` ไดรเวอร์ `tty` ใช้ไฟล์เพื่อกำหนดค่าทั้ง `host presentation services (HPS)` และอะแดปเตอร์

โดยปกติ คำสั่ง `hty_load` ถูกเรียกจากไฟล์ `/etc/rc.ntx`

## ไฟล์คอนฟิกูเรชัน

คำสั่ง `hty_load` ใช้ไฟล์คอนฟิกูเรชันเพื่อกำหนดค่าอะแดปเตอร์ แต่ละรายการจะอยู่ในแต่ละบรรทัด แยกกัน รายการถูกคั่นด้วยอักขระบรรทัดใหม่ ฟิลด์ในรายการถูก คั่นด้วยแท็บหรืออักขระเว้นวรรค รายการในไฟล์คอนฟิกูเรชัน มีฟิลด์ต่อไปนี้:

*MinorNumber Cluster NumberOfPorts*

ฟิลด์เหล่านี้มีค่าต่อไปนี้:

ไอเท็ม	คำอธิบาย
<i>MinorNumber</i>	ระบุหมายเลขอุปกรณ์รองของบอร์ด
<i>Cluster</i>	ฟิลด์นี้มีค่าเป็น 1 เสมอ

ไอเท็ม	คำอธิบาย
<i>NumberOfPorts</i>	ระบุจำนวนอุปกรณ์ <code>hty</code> จำนวนจะขึ้นอยู่กับโมเดล ของอะแดปเตอร์ที่คุณกำลังใช้ จำนวนช่องสัญญาณที่ใช้ได้คือตั้งแต่ 1 ถึง 256 สำหรับบอร์ด 2MB หรือตั้งแต่ 1 ถึง 2048 สำหรับบอร์ด 8MB

ไฟล์คอนฟิกูเรชันยังสนับสนุนความคิดเห็น บรรทัด ความคิดเห็นจะขึ้นต้นด้วย `#` (เครื่องหมายสี่เหลี่ยม) ทั้งหมดที่อยู่ต้นขาของอักขระ ความคิดเห็นจะถูกข้าม บรรทัดความคิดเห็นจะลงท้ายด้วยอักขระบรรทัดใหม่

## แฟล็ก

ไอเท็ม	คำอธิบาย
<code>-d Device</code>	ระบุชื่อไฟล์อุปกรณ์ดิบของอะแดปเตอร์ ใช้ชื่อพจนานุกรมเพื่อแทนที่ชื่ออุปกรณ์ดีฟอลต์ <code>/dev/rhp0</code>
<code>-f ConfigFileName</code>	ระบุชื่อไฟล์คอนฟิกูเรชันของไดรเวอร์ ไฟล์คอนฟิกูเรชัน ดีฟอลต์คือไฟล์ <code>/etc/hty_config</code>

## สถานะออก

คำสั่งนี้จะส่งคืนค่าออกดังต่อไปนี้:

ไอเท็ม	คำอธิบาย
0	ดำเนินการเป็นผลสำเร็จ
>0	มีข้อผิดพลาดเกิดขึ้น

## ความปลอดภัย

ค่าควบคุมการเข้าถึง: คุณต้องมีสิทธิ์ `root` ในการรัน คำสั่งนี้

เหตุการณ์ของระบบตรวจสอบ: N/A

## ตัวอย่าง

เมื่อต้องการโหลดคอนฟิกูเรชันระบบ และใช้ไฟล์คอนฟิกูเรชันดีฟอลต์ของไดรเวอร์ให้ป้อน:

```
hty_load -d /dev/rhp0
```

## ไฟล์

ไอเท็ม	คำอธิบาย
/usr/bin/hty_load	มีคำสั่ง hty
/etc/rc.nix	เรียกใช้คำสั่ง hty_load
/etc/hty_config	ชื่อไฟล์คอนฟิกูเรชันดีฟอลต์ของไดรเวอร์ NTX
/dev/rhp0	ชื่อไฟล์อุปกรณ์ NTX raw ดีฟอลต์

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง /dev/rhp

---

## คำสั่ง hyphen

### วัตถุประสงค์

ค้นหาคำที่มีเครื่องหมายขีดคั่น

### ไวยากรณ์

hyphen [ *File* ... ]

### คำอธิบาย

คำสั่ง **hyphen** อ่านไฟล์ ภาษาอังกฤษอย่างน้อยหนึ่งไฟล์ ค้นหาบรรทัดทั้งหมดที่สิ้นสุดด้วยคำที่มี เครื่องหมายขีดคั่น และเขียนคำเหล่านี้ไปยังเอาต์พุตมาตรฐาน พารามิเตอร์ *File* ระบุไฟล์ภาษาอังกฤษที่จะอ่านโดยคำสั่ง **hyphen** คำดีฟอลต์คืออินพุตมาตรฐาน หากไม่ระบุไฟล์ หรือหากระบุ - (เครื่องหมายขีดคั่น) เป็นชื่อไฟล์สุดท้าย คำสั่ง **hyphen** จะอ่านอินพุตมาตรฐาน คำสั่ง **hyphen** สามารถใช้เป็นตัวกรอง

**หมายเหตุ:** คำสั่ง **hyphen** ไม่สามารถอ่านคำที่ถูกใส่เครื่องหมายขีดคั่นที่เป็นตัวเอียง หรือ ขีดเส้นใต้ คำสั่ง **hyphen** บางครั้งจะให้เอาต์พุตที่ไม่จำเป็น

## ตัวอย่าง

ในการตรวจสอบการใส่เครื่องหมายขีดคั่นที่ดำเนินการโดย โปรแกรมการจัดรูปแบบข้อความบนไฟล์ให้ป้อน:

```
mm [Flag...] [File...] | hyphen
```

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mm

คำสั่ง troff



---

## คำประกาศ

ข้อมูลนี้พัฒนาขึ้นสำหรับผลิตภัณฑ์และบริการที่มีในประเทศสหรัฐอเมริกาเท่านั้น

IBM อาจไม่นำเสนอผลิตภัณฑ์ เซอร์วิส หรือคุณลักษณะที่อธิบายในเอกสารนี้ในประเทศอื่น โปรดปรึกษาตัวแทน IBM ในท้องถิ่นของคุณสำหรับข้อมูลเกี่ยวกับผลิตภัณฑ์และเซอร์วิสที่มีอยู่ในพื้นที่ของคุณในปัจจุบัน การอ้างอิงใดๆ ถึงผลิตภัณฑ์ โปรแกรม หรือเซอร์วิสของ IBM ไม่ได้มีวัตถุประสงค์ที่จะระบุหรือตีความว่า สามารถใช้ได้เฉพาะ ผลิตภัณฑ์ โปรแกรม หรือ เซอร์วิสของ IBM เพียงอย่างเดียว เท่านั้น ผลิตภัณฑ์ โปรแกรม หรือเซอร์วิสใดๆ ที่สามารถทำงานได้เท่าเทียมกัน และไม่ละเมิดสิทธิทรัพย์สินทางปัญญาของ IBM สามารถนำมาใช้แทนได้ อย่างไรก็ตาม ถือเป็นความรับผิดชอบของผู้ใช้ที่จะประเมิน และตรวจสอบการดำเนินการของ ผลิตภัณฑ์ โปรแกรม หรือเซอร์วิสที่ไม่ใช่ของ IBM

IBM อาจมีสิทธิบัตร หรืออยู่ระหว่างการขอสิทธิบัตร ที่ครอบคลุมหัวข้อซึ่งอธิบายในเอกสารนี้ การนำเสนอเอกสารนี้ ไม่ได้เป็นการมอบใบอนุญาตในสิทธิบัตรดังกล่าวให้แก่คุณ คุณสามารถส่งคำถามเกี่ยวกับใบอนุญาตเป็นลายลักษณ์อักษรไปที่:

*IBM Director of Licensing*  
*IBM Corporation*  
*North Castle Drive, MD-NC119*  
*Armonk, NY 10504-1785*  
*US*

หากมีคำถามเกี่ยวกับข้อมูลชุดอักขระไบต์คู่ (DBCS) โปรดติดต่อแผนกทรัพย์สินทางปัญญาของ IBM ใน ประเทศของคุณ หรือส่งคำถามเป็นลายลักษณ์อักษร ไปยัง:

*Intellectual Property Licensing*  
*Legal and Intellectual Property Law*  
*IBM Japan Ltd.*  
*19-21, Nihonbashi-Hakozakicho, Chuo-ku*  
*Tokyo 103-8510, Japan*

INTERNATIONAL BUSINESS MACHINES CORPORATION นำเสนอสิ่งพิมพ์นี้ "ตามสภาพ" โดยไม่มีการรับประกันใดๆ โดยชัดแจ้งหรือ โดยนัย ซึ่งรวมถึงแต่ไม่จำกัดเฉพาะการรับประกันโดยนัยถึงการไม่ละเมิด ความสามารถขายได้ หรือความเหมาะสมสำหรับวัตถุประสงค์เฉพาะ ในบางรัฐไม่อนุญาตให้ ปฏิเสธการรับประกันทางตรงหรือทางอ้อมในธุรกรรมบางอย่าง ดังนั้น ข้อความนี้จึงอาจจะไม่ใช้กับคุณ

ข้อมูลนี้อาจมีความไม่ถูกต้องด้านเทคนิคหรือข้อผิดพลาดจากการพิมพ์ มีการดำเนินการเปลี่ยนแปลงข้อมูลในเอกสารนี้เป็น ครั้งคราว การเปลี่ยนแปลงเหล่านี้จะรวมอยู่ในสิ่งพิมพ์เอ디션ใหม่ IBM อาจปรับปรุง และ/หรือเปลี่ยนแปลงในผลิตภัณฑ์ และ/หรือโปรแกรมที่อธิบายไว้ในสิ่งพิมพ์นี้ได้ตลอดเวลา โดยไม่ต้องแจ้งให้ทราบ

การอ้างอิงใดๆ ในข้อมูลนี้ถึงเว็บไซต์ที่ไม่ใช่ของ IBM มีการจัดเตรียมเพื่อความสะดวกเท่านั้น และไม่ได้เป็นการรับรองเว็บไซต์เหล่านั้นในลักษณะใดๆ เอกสารประกอบที่เว็บไซต์ดังกล่าวไม่ได้ เป็นส่วนหนึ่งของเอกสารประกอบสำหรับผลิตภัณฑ์ IBM นี้ และการใช้เว็บไซต์ดังกล่าวถือเป็นความเสี่ยงของคุณเอง

IBM อาจใช้หรือแจกจ่ายข้อมูลใดๆ ที่คุณ ให้ในรูปแบบต่างๆ ซึ่ง IBM เชื่อว่ามีความเหมาะสมได้โดยไม่เกิดข้อผูกมัดใดๆ กับคุณ

ผู้รับใบอนุญาตของโปรแกรมนี้ที่ต้องการได้รับข้อมูลเกี่ยวกับโปรแกรมเพื่อเปิดใช้งาน: (i) การแลกเปลี่ยนข้อมูลระหว่างโปรแกรมที่สร้างขึ้นอย่างอิสระและโปรแกรมอื่นๆ (รวมถึงโปรแกรมนี้) และ (ii) การใช้ข้อมูลที่มีการแลกเปลี่ยนร่วมกัน ควรติดต่อ:

*IBM Director of Licensing*

*IBM Corporation*

*North Castle Drive, MD-NC119*

*Armonk, NY 10504-1785*

*US*

ข้อมูลดังกล่าวอาจพร้อมใช้งานภายใต้ระยะเวลาและเงื่อนไขที่เหมาะสม โดยมีการชำระค่าธรรมเนียมในบางกรณี

โปรแกรมที่ได้รับอนุญาตซึ่งอธิบายไว้ในเอกสารนี้และเอกสารประกอบที่ได้รับอนุญาตทั้งหมดที่มีอยู่ มีการนำเสนอโดย IBM ภายใต้ระยะเวลาของข้อตกลงกับลูกค้าของ IBM, ข้อตกลงไลเซนส์โปรแกรมระหว่างประเทศของ IBM หรือข้อตกลงที่เท่าเทียมกันใดๆ ระหว่างเรา

ข้อมูลประสิทธิภาพ และตัวอย่างลูกค้าที่ระบุมีการนำเสนอสำหรับวัตถุประสงค์การสาธิตเท่านั้น ผลลัพธ์ของประสิทธิภาพการทำงานจริงอาจขึ้นอยู่กับคอนฟิกูเรชันและเกณฑ์การทำงานที่ระบุเฉพาะ

ข้อมูลเกี่ยวกับผลิตภัณฑ์ที่ไม่ใช่ของ IBM ได้รับมาจากซัพพลายเออร์ของผลิตภัณฑ์เหล่านั้น คำประกาศที่เผยแพร่ หรือแหล่งข้อมูลที่พร้อมใช้งานสำหรับสาธารณะอื่น IBM ไม่ได้ทดสอบผลิตภัณฑ์ดังกล่าว และไม่สามารถยืนยันความถูกต้องของประสิทธิภาพ ความเข้ากันได้ หรือการเรียกร้องอื่นใดที่เกี่ยวข้องกับผลิตภัณฑ์ที่ไม่ใช่ของ IBM คำถามเกี่ยวกับ ความสามารถของผลิตภัณฑ์ที่ไม่ใช่ของ IBM ควรส่งไปยังซัพพลายเออร์ของผลิตภัณฑ์เหล่านั้น

ข้อความใดๆ ที่เกี่ยวข้องกับทิศทางในอนาคตและเจตจำนงค์ของ IBM อาจมีการเปลี่ยนแปลง หรือเพิกถอนได้โดยไม่ต้องแจ้งล่วงหน้า และนำเสนอเฉพาะเป้าหมาย และวัตถุประสงค์เท่านั้น

ราคา IBM ทั้งหมดที่แสดงเป็นราคาขายปลีกที่แนะนำของ IBM ซึ่งเป็นราคาปัจจุบัน และอาจเปลี่ยนแปลงได้โดยไม่ต้องแจ้งให้ทราบ ราคาของผู้แทนจำหน่ายอาจแตกต่างกันไป

ข้อมูลนี้ใช้สำหรับวัตถุประสงค์การวางแผนเท่านั้น ข้อมูลในเอกสารนี้อาจมีการเปลี่ยนแปลง ก่อนผลิตภัณฑ์ที่อธิบายจะมีวางจำหน่าย

ข้อมูลนี้มีตัวอย่างของข้อมูลและรายงานที่ใช้ในการดำเนินธุรกิจประจำวัน เพื่อสาธิตข้อมูลให้สมบูรณ์ที่สุดเท่าที่จะเป็นไปได้ ตัวอย่างจึงมีชื่อของแต่ละบุคคล บริษัท ยี่ห้อ และผลิตภัณฑ์ต่างๆ ชื่อเหล่านี้ทั้งหมดเป็นชื่อสมมติ และความคล้ายคลึงใดๆ กับบุคคล หรือองค์กรธุรกิจที่มีอยู่จริง ถือเป็นเหตุบังเอิญ

งานอันมีลิขสิทธิ์:

ข้อมูลนี้มีโปรแกรมแอปพลิเคชันตัวอย่างในภาษาต้นฉบับ ซึ่งสาธิต เทคนิคการเขียนโปรแกรมบนแพลตฟอร์มปฏิบัติการต่างๆ คุณอาจคัดลอก ปรับเปลี่ยน และแจกจ่ายโปรแกรมตัวอย่าง เหล่านี้ในรูปแบบใดๆ โดยไม่ต้องชำระเงินให้แก่ IBM เพื่อใช้สำหรับการพัฒนา การใช้งาน การตลาด หรือการแจกจ่ายโปรแกรมแอปพลิเคชัน ที่สอดคล้องกับอินเตอร์เฟซการเขียน

โปรแกรมแอปพลิเคชันของแพลตฟอร์มปฏิบัติการ ซึ่งใช้เขียน โปรแกรมตัวอย่าง ตัวอย่างเหล่านี้ยังไม่ได้ผ่านการทดสอบใน ทุกสภาพ ดังนั้น IBM จึงไม่สามารถรับประกันหรือแจ้งถึง ความน่าเชื่อถือ การให้บริการได้ หรือฟังก์ชันของโปรแกรมเหล่านี้ โปรแกรมตัวอย่างมีการนำเสนอ "ตาม สภาพ" โดยไม่มีการรับประกันประเภทใดๆ IBM ไม่ ต้องรับผิดชอบต่อความเสียหาย ใดๆ ที่เกิดขึ้นจากการใช้โปรแกรมตัวอย่างของคุณ

แต่ละสำเนาหรือส่วนใดๆ ของโปรแกรมตัวอย่างเหล่านี้ หรืองานที่สืบเนื่องใดๆ ต้องมีคำประกาศ ลิขสิทธิ์ดังนี้:

© (ชื่อบริษัทของคุณ) (ปี)

ส่วนของโค้ดนี้ได้รับมาจาก IBM Corp. Sample Programs

© Copyright IBM Corp. (C) ลิขสิทธิ์ IBM Corp. \_ป้อน ปี\_

---

## สิ่งที่ต้องพิจารณาเกี่ยวกับนโยบายความเป็นส่วนตัว

IBM Software products, including software as a service solutions, (“Software Offerings”) may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as the customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM’s Privacy Policy at <http://www.ibm.com/privacy> and IBM’s Online Privacy Statement at <http://www.ibm.com/privacy/details> the section entitled “Cookies, Web Beacons and Other Technologies” and the “IBM Software Products and Software-as-a-Service Privacy Statement” at <http://www.ibm.com/software/info/product-privacy>.

---

## เครื่องหมายการค้า

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at Copyright and trademark information at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Windows เป็นเครื่องหมายการค้าของ Microsoft Corporation ในประเทศ สหรัฐอเมริกา ประเทศอื่นๆ หรือทั้งสอง

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

---

## ดัชนี

### อักขระพิเศษ

/etc/qconfig file  
    แปลงเป็นไฟล์ /etc/qconfig.bin  
    การใช้คำสั่ง /user/lpd/digest 163  
/user/lpd/digest command 163

## A

acct/\* commands  
    dodisk 184

## C

commands  
    dd 89  
    defvsd 99  
    detachrsct 106  
    diff 150  
    disable 167  
    dosread 192  
    ed 297  
    edquota 344  
    elogevent 369  
    enscript 396  
    env 408  
    event response resource manager (ERRM)  
        elogevent 369  
        ewallevent 454  
    ewallevent 454  
    ex 456  
    extendlv 479  
    fccheck 493  
    fcclear 495  
    fcdecode 498  
    fcdispfid 500  
    fcfilter 501  
    fcinit 503  
    fclogerr 507  
    fcpushstk 515  
    fcreport 521  
    fcstkrpt 526  
    fcteststk 528  
    fencevsd 541  
    find 568  
    forcerpoffline 593  
    get 692

commands (ต่อ)  
    getconf 701  
    grpsvcctrl 743  
    ha\_vsd 754  
    ha.vsd 749  
    haemqvar 757  
    haemtrcoff 761  
    haemtrcon 764  
    haemunkrm 766  
    hagsvote 773  
    hatsoptions 781  
    hostent 796  
    red 297

## D

dadmin 3  
daemon  
    การเริ่มการบันทึกข้อผิดพลาด 429  
    ยกเลิกการบันทึกข้อผิดพลาด 444  
daemons  
    dhcprd 135  
    dhcpsd 139  
    fingerd 580  
    ftpd 643  
    gssd 746  
    haemd 755  
    hagsd 768  
dbx  
    tracehwp 69  
dcp 83  
delta files  
    การสร้าง 103  
DHCP 139  
    dhcpcd daemon 132  
    dhcprd daemon 135  
    dhcpsd daemon 139  
    dhcpsdv6 daemon 140  
diagnostics  
    ฮาร์ดแวร์ 142, 147  
dig 157  
dnssec-keygen 178  
dnssec-makekeyset 180  
dnssec-signkey 181  
dnssec-signzone 183  
DOS  
    การฟอร์แมตดิสเก็ต 190  
dpid2 Daemon 196

dtscript 264

## E

errdemon daemon 429

event response resource manager (ERRM)

commands

elogevent 369

ewallevent 454

ข้อมูลเหตุการณ์

การบันทึก 369

สคริปต์

elogevent 369

ewallevent 454

exportfs 465

## F

file

การค้นหาลูปแบบ

การใช้คำสั่ง egrep 357

การจัดคิว 386

ทำเครื่องหมายความแตกต่าง 156

แสดงจำนวนบล็อก 281

fingerd daemon 580

flush-secldapclntd 584

FORTRAN

การแบ่งออกเป็นไฟล์แยก 629

FRCA

การควบคุมและการกำหนดค่า 606

fsplit 629

ftpd daemon

การร้องขอ file transfer protocol 646

คำแนะนำ subtree 646

คำอธิบายของ 643

## G

gated daemon

การจัดการกับ SRC 667

คำอธิบายของ 667

สัญญาณ 667

getrunmode 719

getsecconf 720

getsyslab 721

gssd 746

## H

haemd daemon 755

haemd\_HACMP program 756

**828** AIX เวอร์ชัน 7.2: ข้อมูลอ้างอิงคำสั่ง วอลุ่ม 2, d - h

hagsd daemon 768

HCON

ไฟล์

การถ่ายโอนระหว่างระบบโลคัลและระบบโฮสต์ 654

hlpdhcpcd 132

hlpdhcprd 135

hlpdhcpsd 139

hlpecho 295

hlpedit 336

hlpexplore 463

hlpfactor 487

hlpfile 549

hlpfortune 597

hlpfsplit 629

hlpgprof 726

hlphangman 778

hlpreregisters 45

Host Fabric Interface 785

host9 794

hostmibd daemon 799

hyphen

การค้นหาคำที่มี 821

## I

i node map

การพิมพ์ข้อมูลบน 293

ISO 2022 110

## L

Live Update

คำสั่ง hmcauth 790

## M

MH

คำสั่ง dp 195

## P

ports

การตั้งค่าคุณสมบัติของ 724

PostScript

การแปลงเป็นรูปแบบข้อความ

การใช้คำสั่ง enscript 396

## S

### SCCS

delta files

การสร้าง 103

### superblock

การพิมพ์ข้อมูลบน 293

## T

### TCP/IP

ฐานข้อมูลคอนฟิกูเรชัน

การควบคุมรายการการแม่พแอตเดรส 796

ตารางโฮสต์ NIC

การขอรับ 722

ฟังก์ชันการจัดเส้นทางเกตเวย์

การจัดเตรียม 667

อินสแตนซ์

กำหนดเน็ตเวิร์กอินเตอร์เฟซ 94

อินสแตนซ์ inet

การกำหนด 96

โฮสต์

การตั้งค่า ID 798

การตั้งชื่อ 802

การรับค่า ID 798

การแสดงชื่อ 802

โฮสต์ไฟล์

แปลงรูปแบบเน็ตเวิร์กโลบารรี 818

### TCP/IP daemons

fingerd 580

ftpd 643

gated 667

### thdata

thread ที่ระบุเฉพาะ 61

## U

Unicode 110

## W

### WebExplorer

เปิดหน้าต่างหลัก

คำสั่ง explore 463

## ก

กระบวนการไฟล์

การแสดงรายการ 651

กราฟ

typesetting 731

กลุ่ม

การตรวจสอบนิยามของ 740

การแสดงความสัมพันธ์ของ 739

กลุ่ม groups 739

กลุ่ม วอลุ่ม

การเพิ่มฟิสิคัลวอลุ่ม 482

การเอ็กซ์พอร์ต นิยามจากชุดของฟิสิคัลวอลุ่ม 472

การกำหนดค่าอะแด็ปเตอร์

การแสดงผลและการดาวน์โหลด 819

การเขียน

และการเปลี่ยนแท็บเป็นเว้นวรรค 460

การควบคุมงาน 545

การคำนวณเลขจำนวนเต็ม 474

การใช้งาน ดิสก์ 281

การติดักโปรแกรม 9

การตัดบรรทัดสำหรับอุปกรณ์เอาต์พุต 586

การบริหารเวลา

การตั้งค่าวันที่และเวลา 5

การเปรียบเทียบ

ไฟล์ข้อความ 150

การมอนิเตอร์ผลการทำงาน

ผลการทำงานระบบไฟล์ 549

เกม

fortune 597

go fish 582

hangman 778

## ข

ข้อความ

การจัดสรรใหม่ 174

การติดตั้งเซตข้อความบันทึกข้อผิดพลาด 432

การแปลงเป็นรูปแบบ PostScript

การใช้คำสั่ง encrypt 396

การเพิ่มข้อมูลให้กับแค็ตตาล็อกการบันทึกข้อผิดพลาด 436

การเลือก 588

การส่งต่อ

คำสั่ง forw 598

การแสดงผลรายการ 588

เมล ไตรีกทอรี 591

ข้อมูลเหตุการณ์

ERRM

การเลือกข้อมูลเหตุการณ์ 369

การบันทึก 369

## ค

คณิตศาสตร์

การแยกตัวประกอบตัวเลข 487

คำสั่ง dc 81

เครื่องคิดเลขตั้งโต๊ะ 81

- คำออก
  - การส่งคืน 488
- คำสั่ง altscreen 205
- คำสั่ง dacinet 1
- คำสั่ง date 5
- คำสั่ง dbts 8
- คำสั่ง dbx
  - alias
    - การลบ 76
  - breakpoint stop
    - การตั้งค่า 72, 73
  - stops
    - การตั้งค่าตำแหน่งที่ระบุ 60
  - watchpoint stops
    - การตั้งค่า 72
  - watchpoint traces
    - การตั้งค่า 75
  - การ trace
    - การเปิด 69, 75
    - ข้อมูล, การพิมพ์ 67
  - การดีบั๊ก thread 62
  - การหยุดโปรแกรม dbx 45
  - ข้อมูล การติดตาม
    - printing 74
  - คาร์ริจิสเตอร์
    - การแสดง 45
  - คำสั่ง prompt, การเปลี่ยน 45
  - คำสั่ง เครื่อง
    - การรัน single 57
  - คำสั่ง ย่อย trace
    - การแสดง 55
  - คำอธิบายของ 9
  - เซลล์
    - การส่งคำสั่งไปที่ 54
  - โดเร็กทอรี
    - ค้นหารายการ, การตั้งค่า 77
  - ตัวแปร
    - deleting 76
    - การกำหนดค่าสำหรับ 50
  - ตัวระบุ
    - การแสดงการระบุคุณสมบัติเพิ่มเติมของ 80
  - เทอร์มินัล เสมือน, การเปิด 49
  - นิพจน์
    - การพิมพ์ค่าของ 42
  - บรรทัดซอร์ส
    - การรัน single 56
  - โปรแกรม dbx
    - การหยุด 45
  - โพรซีเดอร์
    - การรันและการพิมพ์ 42
    - รายการของรายการ 79
  - ฟังก์ชัน
    - ปัจจุบัน 77
- คำสั่ง dbx (ต่อ)
  - ฟังก์ชัน (ต่อ)
    - รายการของรายการ 79
  - สัญลักษณ์ระบบ
    - การแสดงการระบุคุณสมบัติเพิ่มเติม 80
  - หยุดคำสั่งย่อย
    - การแสดง 55
  - อ็อบเจกต์โค้ด
    - การรัน 9
  - แอปพลิเคชันโปรแกรม
    - การรันกับโพรซีเดอร์ที่ระบุ 47
    - การรันคำสั่งเครื่องถัดไป 42
    - การเริ่มต้น 49
    - การสแตทแอปพลิเคชัน 46
    - การแสดงการประกาศคอมไพเนนต์ของ 78
    - การหยุด 57
    - ดำเนินการจากจุดหยุดปัจจุบัน 54
  - แอปพลิเคชันโปรแกรม
    - การดำเนินการต่อ 9
- คำสั่ง dcp 83
- คำสั่ง dd 89
- คำสั่ง defvsd 99
- คำสั่ง deleteX11input 102
- คำสั่ง deroff 105
- คำสั่ง detachrset 106
- คำสั่ง devinstall 108
- คำสั่ง devnm 109
- คำสั่ง devrsrv 110
- คำสั่ง df 118
- คำสั่ง dfmounts 124
- คำสั่ง dfpd 126
- คำสั่ง dfsc 127
- คำสั่ง dfshares 129
- คำสั่ง dhcpaction 130
- คำสั่ง dhcpd6 134
- คำสั่ง dhcpsconf 137
- คำสั่ง diag 142
- คำสั่ง diaggetrto 145
- คำสั่ง diagrpt 147
- คำสั่ง diagsetrto 148
- คำสั่ง diction
  - คำอธิบายของ 149
- คำสั่ง diff 150
- คำสั่ง diff3 154
- คำสั่ง diffmk 156
- คำสั่ง dimame 166
- คำสั่ง disable 167
- คำสั่ง diskusg 169
- คำสั่ง dispgid 171
- คำสั่ง dispuid 172
- คำสั่ง dist 174
- คำสั่ง dmpuncompress 177
- คำสั่ง dodisk 184

คำสั่ง domainname 186  
 คำสั่ง domlist 187  
 คำสั่ง dosread 192  
 คำสั่ง dp 195  
 คำสั่ง dping 198  
 คำสั่ง drmgr 200  
 คำสั่ง drslot 201  
 คำสั่ง dscrctl 204  
 คำสั่ง dsh 208  
 คำสั่ง dshbak 206  
 คำสั่ง dslpaccept 216  
 คำสั่ง dslpaccess 217  
 คำสั่ง dslpadmin 218  
 คำสั่ง dslpdisable 222  
 คำสั่ง dslpenable 223  
 คำสั่ง dslpprotocol 224  
 คำสั่ง dslpreject 226  
 คำสั่ง dslpsearch 227  
 คำสั่ง dspcat 229  
 คำสั่ง dspmsg 230  
 คำสั่ง dtaction 232  
 คำสั่ง dtappintegrate 234  
 คำสั่ง dtlogin 236  
 คำสั่ง dtsession 265  
 คำสั่ง dtterm 274  
 คำสั่ง du 281  
 คำสั่ง dump 283  
 คำสั่ง dumpcheck 285  
 คำสั่ง dumpctrl 287  
 คำสั่ง dumpfs 293  
     inode map 293  
     superblock 293  
     ดิสก์แม็พ 293  
 คำสั่ง echo 295  
 คำสั่ง ed 297  
 คำสั่ง edit 336  
 คำสั่ง edquota 344  
 คำสั่ง efsenable 347  
 คำสั่ง efskeymgr 349  
 คำสั่ง efskstoldif 353  
 คำสั่ง efsmgr 355  
 คำสั่ง egrep 357  
 คำสั่ง eimadmin 360  
 คำสั่ง elogevent 369  
 คำสั่ง emgr 371  
 คำสั่ง emstat 377  
 คำสั่ง enable 382  
 คำสั่ง enotifyevent 384  
 คำสั่ง enq 386  
 คำสั่ง enroll 395  
 คำสั่ง encrypt 396  
 คำสั่ง enstat 403  
 คำสั่ง env 408  
 คำสั่ง epkg 410  
 คำสั่ง eqn 419  
     การลบการสร้างคำสั่งจาก 105  
 คำสั่ง errclear 421  
 คำสั่ง errctrl 423  
 คำสั่ง errdead 428  
 คำสั่ง errinstall 432  
 คำสั่ง errlogger 435  
 คำสั่ง ERRM  
     elogevent 369,454  
 คำสั่ง errmsg 436  
 คำสั่ง errpt 438  
 คำสั่ง errstop 444  
 คำสั่ง errupdate 445  
 คำสั่ง ethchan\_config 452  
 คำสั่ง ewallevent 454  
 คำสั่ง ex 456  
 คำสั่ง execerror 458  
 คำสั่ง excrset 458  
 คำสั่ง expand 460  
 คำสั่ง expfilt 462  
 คำสั่ง explain 463  
 คำสั่ง explore 463  
 คำสั่ง exportvg 472  
 คำสั่ง expr 474  
 คำสั่ง exptun 478  
 คำสั่ง extendlv 479  
 คำสั่ง extendvg 482  
 คำสั่ง f 485  
 คำสั่ง factor 487  
 คำสั่ง fastboot 489  
 คำสั่ง fasthalt 776  
 คำสั่ง fc 490  
 คำสั่ง fcstat 523  
 คำสั่ง fddistat 531  
 คำสั่ง fdformat 534  
 คำสั่ง fdpr 535  
 คำสั่ง fencevsc 541  
 คำสั่ง ff 543  
 คำสั่ง fg 545  
 คำสั่ง fgrep 546  
 คำสั่ง file 549  
 คำสั่ง filemon 551  
 คำสั่ง fileplace 566  
 คำสั่ง find 568  
 คำสั่ง finger 577  
     ตัวอย่างของ 486,577  
 คำสั่ง flcopy 583  
 คำสั่ง fmt 585  
 คำสั่ง fold 586  
 คำสั่ง folder 588  
 คำสั่ง forcercpoffline 593  
 คำสั่ง format 595

คำสั่ง fortune 597  
 คำสั่ง forw 598  
 คำสั่ง fpm 602  
 คำสั่ง frcactrl 606  
 คำสั่ง from 609  
 คำสั่ง fsck 610  
 คำสั่ง fsck\_cacheifs 615  
 คำสั่ง fsdb 615  
 คำสั่ง ftp 630  
 คำสั่ง fuser 651  
 คำสั่ง fwtmp 653  
 คำสั่ง fxfer 654  
 คำสั่ง gdc 670  
 คำสั่ง gencat 673  
 คำสั่ง gencopy 674  
 คำสั่ง gencore 675  
 คำสั่ง genfilt  
   การเพิ่มกฎตัวกรอง 676  
 คำสั่ง geninstall 678  
 คำสั่ง genkex 682  
 คำสั่ง genkld  
   รายการอ็อบเจกต์ที่แบ่งใช้ 683  
 คำสั่ง genld  
   รายการอ็อบเจกต์ที่โหลด 684  
 คำสั่ง gennames 685  
 คำสั่ง gensyms 686  
 คำสั่ง gentun 688  
 คำสั่ง genxlt 690  
 คำสั่ง get 692  
 คำสั่ง getconf 701  
 คำสั่ง getdev 709  
 คำสั่ง getdgrp 711  
 คำสั่ง getea 714  
 คำสั่ง getopt 715  
 คำสั่ง getopts 717  
 คำสั่ง gettable 722  
 คำสั่ง gettrc 723  
 คำสั่ง getty 724  
 คำสั่ง gprof 726  
 คำสั่ง grap 731  
 คำสั่ง greek 735  
 คำสั่ง grep 736  
 คำสั่ง grpck 740  
 คำสั่ง grpsvcctrl 743  
 คำสั่ง ha\_star 753  
 คำสั่ง ha\_vsd 754  
 คำสั่ง ha.vsd 749  
 คำสั่ง haemqvar 757  
 คำสั่ง haemtrcoff 761  
 คำสั่ง haemtrcon 764  
 คำสั่ง haemunlkrm 766  
 คำสั่ง hagsns 771  
 คำสั่ง hagsvote 773

คำสั่ง halt 776  
 คำสั่ง hangman 778  
 คำสั่ง hash 779  
 คำสั่ง hatsoptions 781  
 คำสั่ง head 783  
 คำสั่ง hfistat 785  
 คำสั่ง hmcauth 790  
 คำสั่ง host 792  
 คำสั่ง hostent 796  
 คำสั่ง hostid 798  
 คำสั่ง hostname 802  
 คำสั่ง hp 803, 804  
 คำสั่ง hplj 805  
 คำสั่ง hpmcount 806  
 คำสั่ง hpmstat 812  
 คำสั่ง hps\_dump 817  
 คำสั่ง htable 818  
 คำสั่ง hty\_load 819  
 คำสั่ง hyphen 821  
 คำสั่ง NIS  
   domainname 186  
 คำสั่ง notifyevent 384  
 คำสั่ง nroff  
   การลบการสร้างคำสั่งจาก 105  
 คำสั่ง NTX  
   hps\_dump 817  
   hty\_load 819  
 คำสั่ง pic  
   การประมวลผลกราฟ 731  
 คำสั่ง reboot 489  
 คำสั่ง red 297  
 คำสั่ง SCCS  
   delta 103  
   get 692  
 คำสั่ง tbl  
   การลบการสร้างคำสั่งจาก 105  
 คำสั่ง TCP/IP  
   gettable 722  
   hostent 796  
   hostid 798  
   hostname 802  
   htable 818  
 คำสั่ง TCP/IP smit  
   คำสั่ง hostent 796  
 คำสั่ง troff  
   การลบ การสร้างคำสั่งจาก 105  
 คำสั่ง โปรแกรมอำนวยความสะดวกข้อความ  
   dspcat 229  
   dspmsg 230  
   gencat 673  
 คำสั่ง ย่อย dbx  
   call 9  
   cont 9

## คำสั่งย่อย dbx (ต่อ)

nexti 42  
print 42  
quit 45  
registers 45  
rerun 46  
sh 54  
skip 54  
source 55  
status 55  
step 56  
stepi 57  
stop 57  
stopi 60  
t 45  
tls 65  
tnext 66  
tnexti 67  
tracei 69  
tskip 70  
tstep 71  
tstepi 71  
tstop 72  
tstophwp 72  
tstopi 73  
ttrace 74  
ttracehwp 75  
ttracei 75  
unalias 76  
unset 76  
use 77  
whatis 78  
whereis 80  
which 80  
โดยที่ 79  
ตั้งค่า 50  
ติดตาม 67  
ทำงาน 77  
เชรด  
    การตีบั๊ก thread 62  
รัน 49  
ส่งคืน 47  
หน้าจอ 49  
คิวของพรีนเตอร์  
    การเปิดใช้งาน 382  
เครื่องพิมพ์ HP LaserJet รุ่น II  
    การประมวลผลเอาต์พุตคำสั่ง troff ภายหลัง 805  
แค่ตัดลือกข้อความ  
    displaying a message 230  
    การแก้ไข 673  
    การสร้าง 673  
    การแสดง 229

## ง

งานส่วนหน้า 545

## ช

ช่องสื่อสาร  
    การนำไปใช้ 395  
ชื่อพาร 779  
ชื่อพารคำสั่ง 779  
ชื่อโฮสต์  
    การเปลี่ยนเป็นอินเทอร์เน็ตแอดเดรส 792  
เซลล์สคริปต์  
    การวิเคราะห์คำอาร์กิวเมนต์บรรทัดคำสั่ง 717  
โปรแกรมรูป  
    การส่งคืน ค่าออก 488

## ด

ดัมพ์ระบบ  
    การแยกเร็กคอร์ดข้อผิดพลาดจาก 428  
ดิสก์แม็พ  
    การพิมพ์ข้อมูลบน 293  
ดิสเก็ต  
    การคัดลอก 583  
    การจัดรูปแบบ  
        คำสั่ง fdformat 534  
        คำสั่ง format 595  
ไดนามิกโฮสต์ คอนฟิกูเรชันโปรโตคอล  
    การส่งต่อแพ็กเก็ต bootp และ dhcp  
        dhcprd daemon 135  
    ข้อมูล เซิร์ฟเวอร์แอดเดรสและคอนฟิกูเรชัน  
        dhcpcd daemon 132  
        dhcpsd daemon 139  
    ส่วนติดต่อผู้ใช้แบบกราฟิก  
        คำสั่ง dhcpsconf 137  
    อัปเดตเซิร์ฟเวอร์ DNS  
        คำสั่ง dhcraction 130  
ไดเรกทอรี  
    การเปรียบเทียบสอง 164  
ไฟล์ DOS  
    การแสดงรายการ 189

## ด

ตาราง การแปล  
    การสร้างสำหรับคำสั่ง ebxa 690  
    การสร้างสำหรับคำสั่ง axeb 690

## ท

เทอร์มินัล 803,804  
เทอร์มินัลรุ่น HP2621  
การตั้งค่าฟังก์ชันพิเศษ 803,804  
เทอร์มินัลรุ่น HP2640  
การตั้งค่าฟังก์ชันพิเศษ 803,804

## ห

นิพจน์  
การค้นหาไฟล์ที่ตรงกัน  
การใช้คำสั่ง find 568  
การหาค่า 474  
นิพจน์ที่ตรงกัน  
การค้นหาไฟล์ด้วย  
การใช้คำสั่ง find 568

## บ

บรรทัดคำสั่ง  
การวิเคราะห์ค่า  
พารามิเตอร์ 715  
แฟล็ก 715  
บันทึกข้อผิดพลาด  
การประมวลผลรายงานของไฟล์บันทึก 438  
การลบรายการจาก 421  
การสร้างรายการสำหรับโอเปอเรเตอร์ 435

## ป

ประเภทไฟล์  
การพิจารณา 549  
โปรแกรม  
haemd\_HACMP 756  
โปรแกรมลูก  
การส่งคืนค่าออก 488  
โปรโตคอลกำหนดคอนฟิกไดนามิกโฮสต์  
รัน NIM และ DHCP พร้อมกัน  
คำสั่ง bootptodhcp 130  
โปรไฟล์การเรียกใช้งาน  
การสร้าง 726

## ผ

ผู้ใช้  
การให้ข้อมูลเกี่ยวกับวิธีใช้ 784  
แสดงข้อมูลของ 485,577

## พ

พจนานุกรมคำพ้อง  
การจัดให้มีแบบโต้ตอบ 463  
พารามิเตอร์  
การวิเคราะห์ค่า 715  
โพรเซสเซอร์  
การหยุดชะงัก  
การใช้คำสั่ง fasthalt 776  
การใช้คำสั่ง halt 776

## ฟ

แฟล็ก  
การวิเคราะห์ค่า 715  
ไฟล์เตอร์  
การเลือก 588  
การแสดงรายการ 588  
การแสดงรายการในเมลโตเร็กทอรี 591  
ไฟล์ 188,549  
deleting  
DOS 188  
type  
การพิจารณา 549  
การคัดลอก  
จาก DOS 192  
ไปที่ DOS 194  
การเปรียบเทียบ 164  
ข้อความ 150  
สาม 154  
การแปลงและการคัดลอก 89  
การพิมพ์ FORTRAN 605  
การโอนย้ายระหว่างโลคัลโฮสต์และรีโมตโฮสต์ 636,637,639  
ค้นหาข้อมูลนิพจน์ที่ตรงกัน  
การใช้คำสั่ง find 568  
ไฟล์ DOS  
deleting 188  
การคัดลอกไปที่ 194  
การคัดลอกไปที่ AIX 192  
โตเร็กทอรีสำหรับ  
การแสดงรายการ 189  
ไฟล์ประวัติ 490  
ไฟล์ประวัติคำสั่ง 490

## ม

เมธอด defif 94  
เมธอด definet 96  
เมธอด TCP/IP  
defif 94  
definet 96

## เมล

- การจัดรูปแบบข้อความก่อนการส่ง 585
- การพิจารณาที่มาของ 609

## ย

- ยูทิลิตี้หลายจอภาพ
- การเริ่ม 205

## จ

### ระบบ

- การรีสตาร์ท
- การใช้คำสั่ง reboot 489

### ระบบไฟล์

- การควบคุมการซ่อมแซมแบบโต้ตอบ
- การใช้คำสั่ง dfscck 127
- การดีบัก 615
- การตรวจสอบความสอดคล้องกัน
- การใช้คำสั่ง dfscck 127
- การรายงานข้อมูลพื้นที่ว่าง 118
- การแสดงรายการชื่อไฟล์ 543
- การแสดงรายการสถิติ 543
- จัดเรียงแล้ว 96

- ระบบไฟล์ที่จัดเรียงแล้ว 96

### ระบบย่อย System V print

- การพิมพ์ที่เปิดใช้กับไดเรกทอรี
- คำสั่ง dslpaccept 216, 227
- คำสั่ง dslpaccess 217
- คำสั่ง dslpadmin 218
- คำสั่ง dslpdisable 222
- คำสั่ง dslpenable 223
- คำสั่ง dslpprotocol 224
- คำสั่ง dslpreject 226

### ระบบรีโมต

- การค้นหาผู้ใช้ 486, 577
- รายการส่วนขยายเคอร์เนล 682
- เรีกคอร์ดส่วนขยายอินพุต
- deleting 102

## ว

### วิธีใช้

- การแสดงผลข้อมูล 784
- เวิร์กสเตชัน Teletype Model 37
- การแปลงเอาต์พุตจาก 735

## ส

### สคริปต์

- elogevent 369

### สคริปต์ (ต่อ)

- emsvcsctrl 379
- enotifyevent 384
- event response resource manager (ERRM)
- elogevent 369
- ewallevent 454
- ewallevent 454
- grpsvcscctrl 743
- notifyevent 384
- สคริปต์ elogevent 369
- สคริปต์ emsvcsctrl 379
- สคริปต์ enotifyevent 384
- สคริปต์ ERRM
- elogevent 369
- ewallevent 454
- สคริปต์ ewallevent 454
- สคริปต์ notifyevent 384
- สคริปต์ควบคุม
- grpsvcscctrl 743
- สถิติผลการทำงาน 785
- สภาวะแวดล้อม
- การแสดงปัจจุบัน 408

## อ

### อ็อบเจกต์ไฟล์

- การดัมพ์ส่วนที่เลือก 283
- อาร์กิวเมนต์
- การเขียนไปยังเอาต์พุตมาตรฐาน 295
- อินเทอร์เน็ตแอดเดรส
- การเปลี่ยนเป็นชื่อโฮสต์ 792
- อุปกรณ์
- การตั้งชื่อ 109
- การติดตั้งการสนับสนุนซอฟต์แวร์ 108
- เอดีเตอร์ cd
- การค้นหาข้อความ 325, 326
- การคัดลอกข้อความ 309
- การทำการเปลี่ยนแปลงโกลบอล 321
- การทำเครื่องหมายข้อความ 322
- การบันทึกข้อความ 324
- การเปลี่ยนข้อความ 308
- การเพิ่มข้อความ 304
- การยกเลิกการเปลี่ยนแปลง 330
- การย้ายข้อความ 322
- การแยกบรรทัด 319
- การรวมบรรทัด 319
- การลบข้อความ 297, 310
- การแสดงผลข้อความ 316
- ความสามารถในการ 304
- โหมดคำสั่ง 297
- โหมดอินพุตข้อความ 297

#### เอดิเตอร์ edit

- การแก้ไขไฟล์เพิ่มเติม 341
- การคัดลอกข้อความ 342
- การแทนที่ข้อความ 343
- การบันทึก 343
  - ไฟล์หลังจากระบบขัดข้อง 343
- การเปลี่ยนแปลง 340
  - ชื่อไฟล์ปัจจุบัน 340
- การเปลี่ยนแปลงโกลบอล, ทำ 342
- การเพิ่มข้อความ 339
- การยกเลิกการเปลี่ยนแปลง 336
- การย้ายข้อความ 342
- การลบข้อความ 340
- การแสดง 336, 341
- ชื่อไฟล์ 341
  - การเปลี่ยนแปลง 340
- บรรทัดปัจจุบัน 336
- สถานะไฟล์ 341
- สิ้นสุด 342
- ออก 342

#### เอาต์พุต

- การแปลงจาก Teletype Model 37 735
- เขียนไปที่พาธที่ระบุ 166

#### เอาต์พุตมาตรฐาน

- การเขียน สตริงอักขระ 295

#### แอคเคาต์กระบวนการ

- การเขียนข้อความไปยังข้อผิดพลาดมาตรฐาน 458

#### แอคเคาต์ดีสก์

- การสร้างข้อมูลโดย ID ผู้ใช้ 169

#### แอคเคาต์ระบบ

- การเปลี่ยนรูปแบบเร็กคอร์ด 653
- การเริ่มต้น 184





พิมพ์ในสหรัฐอเมริกา