AIX Version 7.1

Security

AIX Version 7.1

Security

หมายเหตุ

ก่อนที่คุณจะใช้ข้อมูลนี้และผลิตภัณฑ์ที่สนับสนุน โปรดอ่านข้อมูลใน “คำประกาศ” ในหน้า 565

เอดิชันนี้ใช้กับ AIX เวอร์ชัน 7.1 และรีลีสในลำดับต่อมาและการแก้ไขทั้งหมด จนกว่าจะมีการบ่งชี้เป็นอย่างอื่นในเอดิชันใหม่

สารบัญ

เกี่ยวกับเอกสารนี้

คอลเล็กชันหัวข้อนี้จัดเตรียมข้อมูลที่สมบูรณ์ให้กับผู้ดูแลระบบ บนไฟล์, ระบบ, และความปลอดภัยของเครือข่าย คอลเล็กชันหัวข้อนี้ มีข้อมูลเกี่ยวกับวิธีการดำเนินการ เช่น งานที่ทำให้ระบบ เข้มแข็ง, การเปลี่ยนสิทธิ์, การตั้งค่าเมธอดการพิสูจน์ตัวตน, และการกำหนดคอนฟิกคุณลักษณะ Common Criteria Security Evaluation คอลเล็กชันหัวข้อนี้ยังมีอยู่บน CD เอกสารคู่มือที่จัดส่งมาพร้อมกับ ระบบปฏิบัติการ

การเน้น

หลักการไฮไลต์ต่อไปนี้จะถูกใช้ในเอกสารนี้:

ตัวหนา	ระบุคำสั่ง รูทีนย่อย คีย์เวิร์ด ไฟล์ โครงสร้าง ไดเร็กทอรี และรายการอื่นๆ ที่มีชื่อ ถูกกำหนดไว้แล้วโดยระบบ รวมทั้งระบุอ็อบเจ็กต์กราฟิก เช่น ปุ่ม เลเบล และไอคอนที่ผู้ใช้เลือก
ตัวเอียง	ระบุพารามิเตอร์ที่ชื่อแท้จริง หรือค่าจะถูกกำหนดโดยผู้ใช้
โมโนสเปซ	ระบุตัวอย่างค่าข้อมูลที่ระบุ ตัวอย่างข้อความที่คล้ายกับที่คุณจะเห็นเมื่อถูกแสดง ตัวอย่าง ของส่วนของโค้ดโปรแกรมที่คล้ายกับที่คุณอาจเขียนในฐานะที่เป็นโปรแกรมเมอร์ ข้อความจากระบบ หรือข้อมูลที่คุณควรพิมพ์

การคำนึงถึงขนาดตัวพิมพ์ใน AIX

ทุกอย่างในระบบปฏิบัติการ AIX® นั้นต้องคำนึงถึงขนาดตัวพิมพ์ หมายความว่าระบบจะถือว่าอักษรตัวพิมพ์ใหญ่ และตัวพิมพ์เล็กแตกต่างกัน ตัวอย่าง คุณสามารถใช้คำสั่ง **ls** เพื่อแสดงรายการไฟล์ ถ้าคุณพิมพ์ LS ระบบจะตอบกลับ ว่า ไม่พบ คำสั่งนั้น เช่นเดียวกับ **FILEA**, **FiLea** และ **filea** ถือเป็นชื่อไฟล์ที่ต่างกัน แม้ว่าจะอยู่ ในไดเร็กทอรีเดียวกัน เพื่อหลีกเลี่ยงสาเหตุการเกิดการดำเนินการที่ต้องการ ให้กระทำ ทำให้แน่ใจเสมอว่าคุณใช้ขนาดตัวพิมพ์ถูกต้อง

ISO 9000

ระบบรับรองคุณภาพที่ลงทะเบียน ISO 9000 ใช้ในการพัฒนาและการผลิตผลิตภัณฑ์นี้

ความปลอดภัย

ระบบปฏิบัติ AIX อนุญาตให้คุณดำเนินการกับงาน เช่น ระบบที่มีความเข้มแข็ง, การเปลี่ยนสิทธิ์, การตั้งค่าเมธอดการพิสูจน์ตัวตน, และการกำหนดคอนฟิกคุณลักษณะ Common Criteria Security Evaluation คอลเล็กชันหัวข้อนี้ ยังมีอยู่บน CD เอกสารคู่มือที่จัดส่งมาพร้อมกับ ระบบปฏิบัติการ

ข้อมูลที่เกี่ยวข้อง:

Computer Emergency Response Team ที่ Carnegie Mellon University (CERT)

Forum of Incident Response and Security Teams (FIRST)

Center for Education and Research in Information Assurance and Security (CERIAS)

สิ่งใหม่ใน Security

อ่านเกี่ยวกับสิ่งใหม่หรือข้อมูลที่ถูกเปลี่ยนแปลงสำหรับ ชุดของหัวข้อของ Security

วิธี ดูสิงที่มีใหม่หรือที่เปลี่ยนแปลง

ในไฟล์ PDF นี้ คุณอาจเห็นแถบการแก้ไข (|) ในขอบด้านซ้าย เพื่อระบุข้อมูลใหม่ และที่เปลี่ยนแปลง

ตุลาคม 2014

ข้อมูลต่อไปนี้ คือข้อสรุปของการอัพเดตที่ถูกสร้างขึ้นในคอลเล็กชันหัวข้อนี้:

- เพิ่มข้อจำกัดที่ Domain RBAC ไม่สามารถใช้ได้ในไฟล์ ชั่วคราวใน "โดเมน RBAC" ในหน้า 132

มิถุนายน 2014

ข้อมูลต่อไปนี้ คือข้อสรุปของการอัพเดตที่ถูกสร้างขึ้นในคอลเล็กชันหัวข้อนี้:

- จัดเรียงส่วนอีกครั้งใน "การตั้งค่า manual tunnels" ในหน้า 272
- เพิ่มข้อจำกัดที่ Domain RBAC ไม่สามารถใช้ได้ภายในพาร์ติชัน เวิร์กโหลดใน "โดเมน RBAC" ในหน้า 132
- ข้อมูลที่เก่าเกินไปถูกลบหรือเปลี่ยนในหัวข้อที่หลากหลาย

พฤศจิกายน 2013

ข้อมูลต่อไปนี้ คือข้อสรุปของการอัพเดตที่ถูกสร้างขึ้นในคอลเล็กชันหัวข้อนี้:

- ข้อมูลที่เพิ่มขึ้นเกี่ยวกับคุณลักษณะของกลุ่มที่ไม่มีโดเมนใน "กลุ่มที่ไม่มีโดเมน" ในหน้า 70
- อัพเดตโพรซีเดอร์เพื่อจัดหาอิมเมจ OpenSSH ใน "อิมเมจ OpenSSH" ในหน้า 221
- อัพเดตโพรซีเดอร์ใน "การสร้าง IKE tunnels ที่ใช้ใบรับรองดิจิตัล" ในหน้า 267
- ข้อมูลที่ย้ายเกี่ยวกับการติดตั้งและการโอนย้าย Kerberos จากรีลีสโน้ตไปยัง "ภาพรวมคำสั่งรีโมตที่ปลอดภัย" ในหน้า 317

- หมายเหตุ คุณลักษณะบางอย่างใน IBM® Systems Director Console สำหรับ AIX ทำงานอย่างไม่ถูกต้องเมื่อระบบปฏิบัติการ AIX รันอยู่ในโหมด ความปลอดภัยตามค่าดีฟอลต์ใน “การรักษาความปลอดภัยค่าดีฟอลต์” ในหน้า 392

มีนาคม 2013

ข้อมูลต่อไปนี้ คือข้อสรุปของการอัพเดตที่ถูกสร้างขึ้นในคอลเล็กชันหัวข้อนี้:

- เนื้อหาที่เพิ่มเติมไปยังคำอธิบายแอ็ตทริบิวต์ rlogin และ rcmds ใน “การควบคุมบัญชีผู้ใช้” ในหน้า 60
- การทำให้เกิดความกระจ่างมากขึ้นและขั้นตอนของโพรซีเดอร์ใน “การเปิดใช้งานการตีบัก PAM” ในหน้า 220
- คำอธิบายที่เปลี่ยนแปลงสำหรับปุ่มแอ็คชันที่ชื่อ **ลบจุดออกจากพาธ root** และ **ลบจุดออกจากพาธที่ไม่ใช่ root** ใน “กลุ่ม AIX Security Expert Miscellaneous” ในหน้า 434
- ข้อมูลที่ชัดเจนเกี่ยวกับการกำหนดขนาดใน “การจำกัดรีซอร์ส” ในหน้า 30

พฤศจิกายน 2012

ข้อมูลต่อไปนี้ คือข้อสรุปของการอัพเดตที่ถูกสร้างขึ้นในคอลเล็กชันหัวข้อนี้:

- อัพเดตข้อมูลที่ใช้กับการติดตั้ง, การกำหนดคอนฟิก, และการใช้ IBM Tivoli® Directory Server:
  - “การตั้งค่าเซิร์ฟเวอร์ข้อมูลความปลอดภัย IBM Tivoli Directory Server” ในหน้า 166
  - “เซิร์ฟเวอร์ LDAP” ในหน้า 35
  - “LDAP” ในหน้า 361
  - “ยูทิลิตีการบันทึกการทำงาน” ในหน้า 373

ตุลาคม 2012

ข้อมูลต่อไปนี้เป็นการสรุป ของอัพเดตที่มีในชุดของหัวข้อนี้:

- เพิ่มในรายการบทบาทที่กำหนดไว้ล่วงหน้าที่ใช้ใน role-based access control (RBAC) ในหัวข้อ “บทบาทที่กำหนดไว้แล้ว” ในหน้า 101
- เพิ่มคำสั่ง **rbacqry** ในรายการของคำสั่ง ในหัวข้อ “คำสั่งที่เกี่ยวข้องกับ RBAC” ในหน้า 120
- เพิ่มแอ็ตทริบิวต์ในหัวข้อ “แอ็ตทริบิวต์ผู้ใช้และกลุ่มที่สนับสนุนโดย Authentication Load Modules” ในหน้า 81 และในหัวข้อ “การตั้งค่าอ็อพชันรหัสผ่านที่แนะนำ” ในหน้า 75
- อัพเดตข้อมูลในหัวข้อ “Public Key Cryptography Standards #11” ในหน้า 195

ตุลาคม 2011

ข้อมูลต่อไปนี้เป็นการสรุป ของอัพเดตที่มีในชุดของหัวข้อนี้:

- อัพเดตคุณลักษณะที่สนับสนุนคุณลักษณะ Internet Key Exchange
- เพิ่มคุณลักษณะการตรวจสอบไลบรารีการสนับสนุนสำหรับ Trusted Signature Database

---

การรักษาความปลอดภัยระบบปฏิบัติการฐาน

การรักษาความปลอดภัยระบบปฏิบัติการจัดเตรียมข้อมูล เกี่ยวกับวิธีการป้องกันระบบโดยไม่พิจารณาถึงภาวะเชื่อมต่อเครือข่าย

ส่วนนี้อธิบายวิธีติดตั้งระบบของคุณโดยเปิดใช้ อ็อพชันการรักษาความปลอดภัย และวิธีรักษาความปลอดภัย AIX จากผู้ใช้ที่ไม่มีสิทธิพิเศษที่มีการเข้าถึงระบบ

การติดตั้งและการตั้งค่าระบบอย่างปลอดภัย

ปัจจัยหลายอย่างเกี่ยวข้องกับการติดตั้งและการตั้งค่า AIX อย่างปลอดภัย

Trusted Computing Base

ผู้ดูแลระบบต้องพิจารณา trust ที่สามารถกำหนด ให้กับโปรแกรม การกำหนดนี้รวมถึงการพิจารณาค่าของ รีซอร์สข้อมูลบนระบบในการตัดสินจำนวน trust ที่จำเป็นสำหรับ โปรแกรมที่จะถูกติดตั้งพร้อมกับ privilege

Trusted Computing Base (TCB) เป็นส่วนหนึ่งของระบบที่รับผิดชอบ ในการบังคับนโยบายการรักษาความปลอดภัยข้อมูลทั่วทั้งระบบ โดยการติดตั้งและการใช้ TCB คุณสามารถกำหนดการเข้าถึงผู้ใช้กับพาธการสื่อสารที่ไว้วางใจ ซึ่งอนุญาตการสื่อสารที่ปลอดภัยระหว่างผู้ใช้และ TCB คุณลักษณะ TCB สามารถถูกเปิดใช้งานเมื่อระบบปฏิบัติการถูกติดตั้ง เมื่อต้องการติดตั้ง TCB บนเครื่องที่ติดตั้งแล้ว คุณจำเป็นต้องทำการติดตั้ง Preservation การเปิดใช้ TCB อนุญาตให้คุณเข้าถึงเชลล์ที่ไว้วางใจ กระบวนการ ที่ไว้วางใจ และ Secure Attention Key (SAK)

การติดตั้งระบบพร้อมกับ TCB:

TCB เป็นส่วนหนึ่งของระบบที่มีหน้าที่ในการบังคับใช้ นโยบายความปลอดภัยข้อมูลของระบบ ฮาร์ดแวร์ของคอมพิวเตอร์ทั้งหมด ถูกรวมไว้ใน TCB, แต่ผู้ที่ดูแลระบบควรมุ่งเป้าไปที่ คอมโพเนนต์ซอฟต์แวร์ของ TCB

ถ้าคุณติดตั้งระบบด้วยตัวเลือก Trusted Computing Base, คุณเปิดใช้งาน พาธที่ไว้วางใจ เชลล์ที่ไว้วางใจ และการตรวจสอบ system-integrity (คำสั่ง **tcbck**) คุณลักษณะเหล่านี้สามารถถูกเปิดใช้ *เฉพาะ* ระหว่างการติดตั้ง base operating system (BOS) ถ้าตัวเลือก TCB ไม่ได้ถูกเลือกระหว่างการติดตั้งเริ่มต้น คำสั่ง **tcbck** ถูกปิดใช้งาน คุณสามารถใช้คำสั่งนี้ เฉพาะโดยการติดตั้งระบบซ้ำโดยมีการเปิดใช้ตัวเลือก TCB

เมื่อต้องการตั้งค่า ตัวเลือก TCB ระหว่างการติดตั้ง BOS ให้เลือก **More Options** จาก หน้าจอ Installation and Settings ในหน้าจอ Installation Options ค่าดีฟอลต์สำหรับการเลือก **Install Trusted Computing Base** คือ **no** เมื่อต้องการเปิดใช้ TCB พิมพ์ 2 และ กด Enter

เนื่องจากทุกอุปกรณ์เป็นส่วนหนึ่งของ TCB ทุกไฟล์ใน ไดเร็กทอรี `/dev` ถูกมอนิเตอร์โดย TCB นอกจากนี้ TCB มอนิเตอร์ไฟล์เพิ่มเติมมากกว่า 600 ไฟล์ เก็บข้อมูลสำคัญ เกี่ยวกับไฟล์เหล่านี้ในไฟล์ `/etc/security/sysck.cfg` ถ้าคุณกำลัง ติดตั้ง TCB ทันทีหลังจากการติดตั้งให้สำรองข้อมูลไฟล์นี้ ไว้ที่สื่อบันทึกที่ถอดได้ เช่น เทป CD หรือดิสก์และเก็บไว้ใน ที่ปลอดภัย

การตรวจสอบ TCB:

ความปลอดภัยของระบบปฏิบัติการถูกทำให้มีอันตรายได้เมื่อไฟล์ Trusted Computing Base (TCB) ไม่ได้ถูกป้องกันอย่างถูกต้องหรือเมื่อ configuration files มีค่าไม่ปลอดภัย

คำสั่ง **tcbck** ตรวจสอบสถานความปลอดภัยของ Trusted Computing Base คำสั่ง **tcbck** ตรวจสอบข้อมูลนี้โดยการอ่านไฟล์ `/etc/security/sysck.cfg` ไฟล์นี้ รวมรายละเอียดของไฟล์ TCB , configuration files และคำสั่ง ที่ไว้วางใจทั้งหมด

ไฟล์ `/etc/security/sysck.cfg` ไม่ได้ ออฟไลน์และจึงอาจถูกแก้ไขโดยแฮกเกอร์ได้ ตรวจสอบว่าคุณได้สร้าง สำเนาอ่านอย่างเดียวแบบออฟไลน์หลังจากแต่ละการอัพเดต TCB และคัดลอกไฟล์นี้จาก สื่อบันทึกถาวรไปที่ดิสก์ก่อนทำการตรวจสอบ

โครงสร้างของไฟล์ sysck.cfg:

คำสั่ง **tcbck** อ่านไฟล์ /etc/security/sysck.cfg เพื่อกำหนดไฟล์ที่จะตรวจสอบ แต่ละโปรแกรมที่ไว้วางใจบนระบบถูกอธิบายโดย stanza ในไฟล์ /etc/security/sysck.cfg

แต่ละ stanza มีแอ็ตทริบิวต์ดังต่อไปนี้:

แอ็ตทริบิวต์	คำอธิบาย
acl	สตริงข้อความแสดงรายการค่าควบคุมการเข้าใช้สำหรับ ไฟล์ ต้องมีรูปแบบเดียวกับเอาต์พุตของคำสั่ง **aclget** ถ้าไม่ตรงกับไฟล์ ACL (access control list) จริง, คำสั่ง **sysck** จะนำค่านี้มาใช้โดยใช้คำสั่ง **aclput** **หมายเหตุ:** แอ็ตทริบิวต์ SUID, SGID และ SVTX ต้องตรงกับข้อมูลที่ระบุ สำหรับโหมด ถ้ามี
class	ชื่อของกลุ่มของไฟล์ แอ็ตทริบิวต์นี้อนุญาตให้มีการตรวจสอบหลายไฟล์ ที่ชื่อเหมือนกันพร้อมกันโดยระบุอาร์กิวเมนต์เดียวไปที่ คำสั่ง **tcbck** สามารถระบุได้มากกว่าหนึ่งคลาส โดยแต่ละคลาสแยกกันด้วยคอมมา
group	ID กลุ่มหรือชื่อของกลุ่มไฟล์ ถ้าข้อมูลนี้ไม่ตรงกับกลุ่มไฟล์ คำสั่ง **tcbck** จะเซ็ต ID กลุ่มของไฟล์ เป็นค่านี้
links	รายการที่คั่นด้วยคอมมาของชื่อพาธที่ลิงก์มาที่ไฟล์นี้ ถ้าชื่อพาธในรายการนี้ไม่ได้ถูกลิงก์ไปที่ไฟล์ คำสั่ง **tcbck** จะสร้างลิงก์ ถ้าใช้โดยไม่มีพารามิเตอร์ *tree* คำสั่ง **tcbck** พิมพ์ข้อความว่ามีลิงก์เพิ่มเติม แต่ไม่ได้ระบุชื่อ ถ้าใช้กับพารามิเตอร์ *tree* คำสั่ง **tcbck** พิมพ์ชื่อพาธเพิ่มเติมที่ ลิงก์มาที่ไฟล์นี้ด้วย
mode	รายการที่คั่นด้วยคอมมาของค่า ค่าที่ใช้ได้คือ SUID, SGID, SVTX และ TCB สิทธิของไฟล์ต้องเป็นค่าล่าสุด และสามารถถูกระบุเป็นค่าเลขฐานแปดหรือสตริง 9 อักขระ ตัวอย่างเช่น 755 หรือ rwxr-xr-x เป็น สิทธิของไฟล์ที่ใช้ได้ ถ้าข้อมูลนี้ไม่ตรงกับโหมดไฟล์จริง คำสั่ง **tcbck** จะนำค่าที่ถูกต้องมาใช้
owner	User ID หรือชื่อของเจ้าของไฟล์ ถ้าข้อมูลนี้ไม่ตรงกับเจ้าของไฟล์ คำสั่ง **tcbck** จะเซ็ต ID เจ้าของไฟล์ เป็นค่านี้
program	รายการที่คั่นด้วยคอมมาของค่า ค่าแรกคือ ชื่อพาธของโปรแกรมตรวจสอบ ค่าเพิ่มเติมถูกส่งเป็นอาร์กิวเมนต์ ไปที่โปรแกรมเมื่อโปรแกรมถูกรัน
source	**หมายเหตุ:** อาร์กิวเมนต์แรก เป็นหนึ่งในค่า *-y*, *-n*, *-p*, หรือ *-t* เสมอขึ้นกับแฟล็กที่คำสั่ง **tcbck** ใช้ ชื่อของไฟล์ที่ไฟล์ต้นฉบับนี้จะถูกคัดลอกมา ก่อนการตรวจสอบ ถ้าค่าเป็นค่าว่าง และนี่เป็น ไฟล์ ไดเร็กทอรี หรือ named pipe ปกติ เวอร์ชันว่างเปล่าค่าใหม่ของไฟล์นี้ถูกสร้างขึ้น ถ้ายังไม่มีอยู่ สำหรับไฟล์อุปกรณ์ ไฟล์พิเศษใหม่จะถูกสร้าง ให้กับอุปกรณ์ที่มีชนิดเดียวกัน
symlinks	รายการที่คั่นด้วยคอมมาของชื่อพาธที่ลิงก์แบบสัญลักษณ์มาที่ไฟล์นี้ ถ้าชื่อพาธในรายการนี้ไม่ได้ลิงก์แบบสัญลักษณ์ไปที่ไฟล์ คำสั่ง **tcbck** จะ สร้างลิงก์สัญลักษณ์ ถ้าใช้กับอาร์กิวเมนต์ *tree* คำสั่ง **tcbck** พิมพ์ชื่อพาธเพิ่มเติมที่เป็นลิงก์สัญลักษณ์ไปที่ไฟล์นี้

ถ้า stanza ในไฟล์ /etc/security/sysck.cfg ไม่ได้ ระบุแอ็ตทริบิวต์ จะไม่มีการทำการตรวจสอบที่เกี่ยวข้อง

การใช้คำสั่ง tcbck:

คำสั่ง **tcbck** ถูกใช้เพื่อประกัน การติดตั้งอย่างถูกต้องของไฟล์ที่เกี่ยวข้องกับความปลอดภัย; เพื่อให้แน่ใจว่าแผนผังไฟล์ ระบบไม่มีไฟล์ที่ละเมิดการรักษาความปลอดภัยของระบบ; และเพื่อ อัพเดต เพิ่ม หรือลบ ไฟล์ที่ไว้วางใจ

คำสั่ง **tcbck** โดยปกติใช้สำหรับงาน ดังต่อไปนี้:

- ตรวจสอบการติดตั้งที่ถูกต้องของไฟล์ที่เกี่ยวข้องกับความปลอดภัย
- ตรวจสอบว่าแผนผังระบบไฟล์ไม่มีไฟล์ที่ละเมิดการรักษาความปลอดภัย ของระบบ
- อัพเดต เพิ่ม หรือลบไฟล์ที่ไว้วางใจ

คำสั่ง **tcbck** สามารถถูกใช้ในวิธี ดังต่อไปนี้:

- ใช้งานปกติ
  - แบบไม่มีการโต้ตอบขณะเริ่มต้นระบบ
  - ใช้กับคำสั่ง **cron**

- ใช้แบบโต้ตอบ
  - การหาจุดบกพร่องของแต่ละไฟล์และคลาสของไฟล์
- ใช้แบบป้องกันไว้ก่อน
  - เก็บไฟล์ sysck.cfg ออฟไลน์และ เรียกคืนไฟล์เป็นระยะเพื่อการหาจุดบกพร่องของเครื่อง

แม้ว่าไม่ได้มีการรักษาความปลอดภัยโดยเข้ารหัส, TCB ใช้คำสั่ง **sum** เพื่อ checksums ฐานข้อมูล TCB สามารถถูกตั้งค่าด้วยตัวเองกับคำสั่ง checksum ที่ต่างกัน ตัวอย่างเช่นคำสั่ง **md5sum** ที่ถูกจัดส่งมาในแพ็กเกจ textutils RPM Package Manager กับ *AIX Toolbox for Linux Applications CD*

การตรวจสอบไฟล์ที่ไว้วางใจ:

ใช้คำสั่ง **tcbck** เพื่อตรวจสอบและแก้ไขไฟล์ ทั้งหมดในฐานข้อมูล tcbck และแก้ไขและสร้างล็อกของข้อผิดพลาดทั้งหมด

เพื่อตรวจสอบไฟล์ทั้งหมดในฐานข้อมูล tcbck และแก้ไขและรายงาน ข้อผิดพลาดทั้งหมดให้พิมพ์:

```
tcbck -y ALL
```

นี่จะ ทำให้คำสั่ง **tcbck** ตรวจสอบการติดตั้งของแต่ละ ไฟล์ในฐานข้อมูล tcbck ตามที่อธิบายโดยไฟล์ /etc/security/sysck.cfg

เพื่อดำเนินการ นี้โดยอัตโนมัติระหว่างการเตรียมข้อมูลระบบ และสร้างล็อก ของข้อมูลข้อผิดพลาด ให้เพิ่มสตริงคำสั่งก่อนหน้านี้ให้กับคำสั่ง **/etc/rc**

การตรวจสอบแผนผังระบบไฟล์:

เมื่อคุณสงสัยความสมบูรณ์ของระบบอาจถูกทำให้มีช่องโหว่ ให้รันคำสั่ง **tcbck** เพื่อตรวจสอบแผนผังระบบไฟล์

เพื่อตรวจสอบแผนผังระบบไฟล์ พิมพ์:

```
tcbck -t tree
```

เมื่อคำสั่ง **tcbck** ถูกใช้กับค่า *tree* ไฟล์ทั้งหมดบนระบบถูกตรวจสอบเพื่อแก้ไขการติดตั้ง (ซึ่งอาจใช้เวลานาน) ถ้าคำสั่ง **tcbck** พบว่ามีไฟล์ที่อาจเป็นภัยต่อความปลอดภัยของระบบ you can alter the suspected file to remove the offending attributes. นอกจากนี้ การตรวจสอบดังต่อไปนี้ทำกับไฟล์อื่นทั้งหมดในระบบไฟล์:

- ถ้าเจ้าของไฟล์คือ root และไฟล์มีการเซ็ตบิต SetUID บิต SetUID จะถูกเคลียร์
- ถ้ากลุ่มไฟล์เป็นกลุ่ม administrative เป็นไฟล์ที่รันได้ และไฟล์มีการเซ็ตบิต SetGID บิต SetGID จะถูกเคลียร์
- ถ้าไฟล์มีการเซ็ตแอ็ตทริบิวต์ **tcb** แอ็ตทริบิวต์นี้จะถูกเคลียร์
- ถ้าไฟล์เป็นอุปกรณ์ (อักขระหรือไฟล์บล็อกพิเศษ) จะถูกลบออก
- ถ้าไฟล์เป็นลิงก์เพิ่มเติมไปที่ชื่อพาธที่อธิบายในไฟล์ /etc/security/sysck.cfg ลิงก์จะถูกเอาออก
- ถ้าไฟล์เป็นลิงก์สัญลักษณ์เพิ่มเติมไปที่ชื่อพาธที่อธิบายในไฟล์ /etc/security/sysck.cfg ลิงก์สัญลักษณ์จะถูกเอาออก

  หมายเหตุ: รายการอุปกรณ์ทั้งหมดต้องถูกเพิ่มให้กับไฟล์ /etc/security/sysck.cfg ก่อนการกระทำการของคำสั่ง **tcbck** หรือไม่แล้วระบบจะใช้การไม่ได้ เมื่อต้องการเพิ่มอุปกรณ์ที่ไว้วางใจให้กับไฟล์ /etc/security/sysck.cfg ให้ใช้แฟล็ก **–l**

ข้อควรสนใจ: *อย่า* รันตัวเลือกคำสั่ง **tcbck -y tree** ตัวเลือกนี้ลบและปิดใช้งานอุปกรณ์ที่แสดงไม่ถูกต้องใน TCB และอาจปิดระบบของคุณ

การเพิ่มโปรแกรมที่ไว้วางใจ:

ใช้คำสั่ง **tcbck** เพื่อเพิ่มโปรแกรม ให้กับไฟล์ /etc/security/sysck.cfg

เมื่อต้องการเพิ่มโปรแกรมที่เจาะจงในไฟล์ /etc/security/sysck.cfg พิมพ์:

```
tcbck -a PathName [Attribute=Value]
```

เฉพาะ แอ็ตทริบิวต์ซึ่งค่าไม่ถูกลดลงจากสถานะปัจจุบันของไฟล์ ที่จำเป็นต้องถูกระบุบนบรรทัดคำสั่ง ชื่อแอ็ตทริบิวต์ทั้งหมดมีอยู่ใน ไฟล์ /etc/security/sysck.cfg

ตัวอย่างเช่น คำสั่งดังต่อไปนี้เรจิสเตอร์โปรแกรม SetUID root ใหม่ชื่อ /usr/bin/setgroups, ซึ่งมีลิงก์ชื่อ /usr/bin/getgroups:

```
tcbck -a /usr/bin/setgroups links=/usr/bin/getgroups
```

เมื่อต้องการเพิ่ม jfh และ jsl เป็นผู้ใช้ที่มีหน้าที่ดูแล และเมื่อต้องการเพิ่ม developers เป็นกลุ่ม administrative ที่จะถูกตรวจสอบระหว่างการตรวจสอบความปลอดภัยของไฟล์ /usr/bin/abc, พิมพ์:

```
tcbck -a /usr/bin/abc setuids=jfh,jsl setgids=developers
```

หลังจาก การติดตั้งโปรแกรม คุณอาจไม่ทราบว่าไฟล์ใหม่ใดที่ถูกเรจิสเตอร์ใน ไฟล์ /etc/security/sysck.cfg ไฟล์เหล่านี้สามารถถูก ค้นหาและเพิ่มด้วยคำสั่งดังต่อไปนี้:

```
tcbck -t tree
```

สตริงคำสั่ง นี้แสดงชื่อของไฟล์ที่จะถูกเรจิสเตอร์ในไฟล์ /etc/security/sysck.cfg

การลบโปรแกรมที่ไว้วางใจ:

ถ้าคุณลบไฟล์ออกจากระบบที่ถูกอธิบายไว้ในไฟล์ /etc/security/sysck.cfg, คุณต้องลบรายละเอียดของไฟล์นี้ออกจากไฟล์ /etc/security/sysck.cfg ด้วย

ตัวอย่างเช่น ถ้าคุณลบโปรแกรม /etc/cvid สตริงคำสั่งดังต่อไปนี้สร้างข้อความแสดงความผิดพลาด:

```
tcbck -t ALL
```

ข้อความแสดงความผิดพลาด เป็นดังนี้:

```
3001-020 The file /etc/cvid was not found.
```

รายละเอียด สำหรับโปรแกรมยังคงอยู่ในไฟล์ /etc/security/sysck.cfg เมื่อต้องการลบรายละเอียดของโปรแกรม ให้พิมพ์คำสั่งดังต่อไปนี้:

```
tcbck -d /etc/cvid
```

การตั้งค่าตัวเลือกที่ไว้วางใจเพิ่มเติม:

คุณสามารถตั้งค่าตัวเลือกเพิ่มเติมสำหรับ Trusted Computing Base (TCB)

การจำกัดการเข้าถึงเทอร์มินัล:

คุณสามารถตั้งค่าระบบปฏิบัติการให้จำกัด การเข้าถึงเทอร์มินัล

คำสั่ง **getty** และ **shell** เปลี่ยนเจ้าของและโหมดของเทอร์มินัลเพื่อป้องการการเข้าถึงเทอร์มินัลจาก โปรแกรมที่ไม่ไว้วางใจ ระบบปฏิบัติการจัดเตรียมวิธีในการตั้งค่า การเข้าถึงเทอร์มินัลเฉพาะ

การใช้ Secure Attention Key:

พาธการสื่อสารที่ไว้วางใจถูกสร้างขึ้นโดยการกด Secure Attention Key (SAK) ลำดับของคีย์ที่สงวนไว้ (Ctrl-X แล้ว Ctrl-R)

หมายเหตุ: โปรดใช้ความระมัดระวังเมื่อใช้ SAK เนื่องจากจะมีการหยุด กระบวนการทั้งหมดที่พยายามเข้าถึงเทอร์มินัลและลิงก์ที่มายังเทอร์มินัล (ตัวอย่างเช่น `/dev/console` สามารถถูกลิงก์ไปที่ `/dev/tty0`)

พาธ การสื่อสารที่ไว้วางใจถูกสร้างภายใต้เงื่อนไขดังต่อไปนี้:

- เมื่อล็อกอินเข้าสู่ระบบ

  หลังจากคุณกด SAK:
  - ถ้าหน้าจอล็อกอินแสดง หมายถึงคุณมีพาธที่ปลอดภัย
  - ถ้าพร้อมต์เชลล์ที่ไว้วางใจแสดง หน้าจอล็อกอินเริ่มต้นเป็นโปรแกรม ที่ไม่ได้รับอนุญาต ซึ่งอาจพยายามขโมยรหัสผ่านของคุณ ตรวจว่า ใครกำลังใช้เทอร์มินัลนี้อยู่ในขณะนี้โดยใช้คำสั่ง **who** แล้วล็อกออฟ
- เมื่อคุณต้องการให้คำสั่งที่คุณป้อนแสดงผลลัพธ์ในการรันโปรแกรมที่ไว้วางใจ ตัวอย่างบางส่วนรวมถึง:
  - รันในฐานะผู้ใช้ root รันในฐานะผู้ใช้ root เฉพาะเมื่อได้สร้างพาธการสื่อสาร ที่ไว้วางใจเท่านั้น นี่จะเป็นการประกันว่าไม่มีโปรแกรมที่ไม่ไว้วางใจรันด้วยสิทธิ root-user
  - รันคำสั่ง **su**, **passwd**, และ **newgrp** รันคำสั่งเหล่านี้เฉพาะเมื่อได้สร้างพาธการสื่อสารที่ไว้วางใจเท่านั้น

การตั้งค่า Secure Attention Key:

ตั้งค่า Secure Attention Key เพื่อสร้างพาธ การสื่อสารที่ไว้วางใจ

แต่ละเทอร์มินัลสามารถถูกตั้งค่าได้อย่างอิสระ เพื่อที่การกด Secure Attention Key (SAK) ที่เทอร์มินัลนั้นจะสร้างพาธการสื่อสารที่ไว้วางใจ ซึ่งถูกระบุโดยแอ็ตทริบิวต์ **sak_enabled** ในไฟล์ `/etc/security/login.cfg` ถ้าค่าของแอ็ตทริบิวต์นี้เป็น True, SAK จะถูกเปิดใช้งาน

ถ้าพอร์ตถูกใช้เพื่อการสื่อสาร (ตัวอย่างเช่น โดยคำสั่ง **uucp**), พอร์ตที่ใช้มีบรรทัดดังต่อไปนี้ใน stanza ของไฟล์ `/etc/security/login.cfg` ของพอร์ต:

```
sak_enabled = false
```

บรรทัดนี้ (หรือไม่มีรายการใน stanza นั้น) ปิดใช้งาน SAK สำหรับเทอร์มินัลนั้น

เมื่อต้องการเปิดใช้งาน SAK บนเทอร์มินัล ให้เพิ่มบรรทัดดังต่อไปนี้ให้กับ stanza สำหรับ เทอร์มินัลนั้น:

```
sak_enabled = true
```

Trusted Execution

Trusted Execution (TE) อ้างอิงการรวบรวมคุณลักษณะ ที่ใช้เพื่อตรวจสอบ integrity ของ ระบบและการประยุกต์ใช้นโยบายการรักษาความปลอดภัยขั้นสูง ซึ่งสามารถ ใช้ร่วมกับเพื่อปรับปรุงระดับความไว้วางใจของทั้งระบบ

แนวทางปกติสำหรับผู้ใช้ที่เป็นอันตรายที่อาจทำลายระบบได้คือหาทาง เข้าถึงระบบและติดตั้งโทรจัน rootkits หรือเปลี่ยนแปลง ไฟล์ที่สำคัญต่อการรักษาความปลอดภัยบางไฟล์ อันเป็นผลให้ระบบเกิดช่องโหว่ และถูกใช้ประโยชน์ได้ แนวคิดกลางที่อยู่เบื้องหลังชุดคุณลักษณะภายใต้ Trusted Execution คือการป้องกันกิจกรรมเหล่านั้น หรือในกรณีเลวร้ายที่สุด คือให้สามารถระบุเหตุการณ์ใดๆ ที่เกิดขึ้นกับระบบ การใช้ ฟังก์ชันที่จัดโดย Trusted Execution ผู้ดูแลระบบ สามารถตัดสินใจจากชุดของไฟล์การดำเนินการที่ได้รับอนุญาต ให้ทำงาน หรือชุดของส่วนขยายเคอร์เนลที่ได้รับอนุญาตให้โหลด รวมทั้งใช้เพื่อตรวจสอบสถานะการรักษาความปลอดภัยของระบบและ ระบุไฟล์ที่เปลี่ยนแปลง ด้วยเหตุนี้จึงมีการเพิ่มระดับของการไว้วางใจ ของระบบ และทำให้ยากสำหรับผู้ใช้ที่เป็นอันตรายที่ อาจทำลายระบบได้ ชุดของคุณลักษณะที่อยู่ภายใต้ TE สามารถจัดกลุ่ม ได้ดังนี้:

- การจัดการ Trusted Signature Database
- การตรวจสอบ integrity ของ Trusted Signature Database
- การตั้งค่านโยบายการรักษาความปลอดภัย
- Trusted Execution Path และ Trusted Library Path

หมายเหตุ: การทำงาน TCB ยังคงมีอยู่ในระบบปฏิบัติการ AIX TE คือกลไกที่ได้รับการปรับปรุงและมีประสิทธิภาพมากขึ้นที่มีการทำงานบางส่วนเหมือนกับการทำงาน TCB และ ยังมีนโยบายการรักษาความปลอดภัยขั้นสูงที่ควบคุม integrity ของระบบได้ดียิ่งขึ้น ในขณะที่ Trusted Computing Base ยังคงพร้อมใช้งาน Trusted Execution ได้แนะนำแนวคิดใหม่มีมีระดับสูงขึ้นในการตรวจสอบ และการป้องกัน system integrity

การจัดการ Trusted Signature Database:

คล้ายกับของ Trusted Computing Base (TCB) ที่มี ฐานข้อมูลซึ่งถูกใช้เก็บพารามิเตอร์การรักษาความปลอดภัยที่สำคัญของไฟล์ที่ไว้วางใจ ที่แสดงบนระบบ ฐานข้อมูลนี้ เรียกว่า Trusted Signature Database (TSD) อยู่ใน `/etc/security/tsd/tsd.dat`.

ไฟล์ที่ไว้วางใจ คือไฟล์ที่มีความสำคัญในด้านความปลอดภัย ของระบบ และถ้ามีช่องโหว่ สามารถเป็นอันตรายต่อความปลอดภัยของ ทั้งระบบ โดยปกติไฟล์ที่ตรงกับรายละเอียดนี้มีดังนี้:

- เคอร์เนล (ระบบปฏิบัติการ)
- โปรแกรม setuid root ทั้งหมด
- โปรแกรม setgid root ทั้งหมด
- โปรแกรมใดๆ ที่รันได้เฉพาะผู้ใช้ root เท่านั้นหรือโดยสมาชิกของ กลุ่มระบบ
- โปรแกรมใดๆ ที่ต้องรันโดยผู้ดูแลระบบขณะอยู่บนพาธการสื่อสาร ที่ไว้วางใจ (ตัวอย่าง คำสั่ง **ls**)
- ไฟล์คอนฟิกูเรชันที่ควบคุมการดำเนินงานของระบบ
- โปรแกรมใดๆ ที่รันด้วยสิทธิพิเศษ หรือสิทธิการเข้าถึงเพื่อเปลี่ยนแปลง ไฟล์คอนฟิกูเรชันเคอร์เนล หรือระบบ

ไฟล์ที่ไว้วางใจทุกไฟล์ควรมี stanza หรือนิยามไฟล์ที่เชื่อมโยงถึง ถูกเก็บอยู่ใน Trusted Signature Database (TSD) ไฟล์สามารถถูกทำเครื่องหมายเป็นไว้วางใจ โดยการเพิ่มนิยามของไฟล์ใน TSD โดยใช้คำสั่ง **trustchk** คำสั่ง **trustchk** สามารถใช้เพื่อเพิ่ม ลบ หรือแสดงรายการจาก TSD

Trusted Signature Database:

Trusted Signature Database คือฐานข้อมูลที่ใช้ เก็บพารามิเตอร์การรักษาความปลอดภัยที่สำคัญของไฟล์ที่ไว้วางใจที่แสดงบนระบบ ฐานข้อมูลนี้อยู่ในไดเร็กทอรี /etc/security/tsd/tsd.dat

ทุกไฟล์ที่ไว้วางใจต้องมี stanza หรือ นิยามไฟล์ที่เชื่อมโยงจัดเก็บไว้ใน Trusted Signature Database (TSD) ทุก ไฟล์ที่ไว้วางใจเชื่อมโยงกับแฮช cryptographic ที่ไม่ซ้ำกันและ เป็นลายเซ็นดิจิทัล แฮช cryptographic ของชุดดีฟอลต์ของไฟล์ที่ไว้วางใจ ถูกสร้างขึ้นโดยใช้อัลกอริทึม SHA-256 และลายเซ็นดิจิทัล ที่สร้างโดยใช้ RSA โดยสภาวะแวดล้อม AIX build และแพคเป็นส่วนหนึ่งของชุดไฟล์ติดตั้ง AIX ค่าการแฮช และลายเซ็นเหล่านี้ถูกส่งเป็นส่วนหนึ่งของอิมเมจการติดตั้ง AIX ที่เกี่ยวข้องและที่เก็บใน Trusted Software Database (/etc/security/tsd/tsd.dat) บนเครื่องปลายทาง ในรูปแบบ stanza ตัวอย่างที่ตามด้วย:

```
/usr/bin/ps:
          owner          = bin
          group          = system
          mode           = 555
          type           = FILE
          hardlinks      = /usr/sbin/ps
          symlinks       =
          size           = 1024
          cert_tag       = bbe21b795c550ab243
          signature      =
f7167eb9ba3b63478793c635fc991c7e9663365b2c238411d24c2a8a
          hash_value     = c550ab2436792256b4846a8d0dc448fc45
          minslabel      = SLSL
          maxslabel      = SLSL
          intlabel       = SHTL
          accessauths    = aix.mls.pdir, aix.mls.config
          innateprivs    = PV_LEF
          proxyprivs     = PV_DAC
          authprivs      =
aix.security.cmds:PV_DAC,aix.ras.audit:PV_AU_ADMIN
          secflags       = FSF_EPS
          t_accessauths  =
          t_innateprivs  =
          t_proxyprivs   =
          t_authprivs    =
          t_secflags     =
```

owner เจ้าของไฟล์ ค่านี้คำนวณโดยคำสั่ง **trustchk** เมื่อไฟล์กำลังเพิ่มใน TSD

group กลุ่มของไฟล์ ค่านี้ถูกคำนวณโดยคำสั่ง **trustchk**

mode รายการที่คั่นด้วยคอมมาของค่า ค่าที่เป็นได้คือ **SUID** (ชุดบิต SUID), **SGID** (ชุดบิต SGID), **SVTX** (ชุดบิต SVTX) และ **TCB** (Trusted Computing Base) สิทธิของไฟล์ต้องเป็น ค่าสุดท้าย และสามารถระบุเป็นค่าฐานแปด เช่น ไฟล์ที่ตั้งค่าด้วย **uid** และมีสิทธิ์บิตเป็น **rwxr-xr-x** ค่าโหมดคือ **SUID, 755** ค่าถูกคำนวณโดยคำสั่ง **trustchk**

type ประเภทของไฟล์ ค่านี้ถูกคำนวณโดยคำสั่ง **trustchk** ค่าที่เป็นได้คือ **FILE, DIRECTORY, MPX_DEV, CHAR_DEV, BLK_DEV** และ **FIFO**

hardlinks

ลิงก์ของ hardlinks ไปยังไฟล์ ค่านี้ไม่สามารถคำนวณโดย คำสั่ง **trustchk** ต้องกำหนดโดย ผู้ใช้เพื่อเพิ่มไฟล์ลงในฐานข้อมูล

symlinks

รายการลิงก์สัญลักษณ์ไปยังไฟล์ ค่านี้ไม่สามารถคำนวณ โดยคำสั่ง **trustchk** ต้องกำหนดโดย ผู้ใช้เพื่อเพิ่มไฟล์ลงในฐานข้อมูล

size กำหนดขนาดของไฟล์ ค่า **VOLATILE** หมายถึง ไฟล์ที่ไม่เปลี่ยนแปลงบ่อย

cert_tag

ฟิลด์นี้แมปลายเซ็นดิจิตัลของไฟล์ที่มีใบรับรองที่เชื่อมโยง ที่สามารถใช้ตรวจสอบลายเซ็นของไฟล์ ฟิลด์นี้จะจัดเก็บใบรับรอง ID และคำนวณโดยคำสั่ง **trustchk** ณ เวลาที่เพิ่มไฟล์ใน TSD ใบรับรองถูก เก็บในไดเร็กทอรี `/etc/security/certificates`

signature

ลายเซ็นดิจิตัลของไฟล์ ค่า **VOLATILE** หมายถึง ไฟล์ที่เปลี่ยนแปลงบ่อย ฟิลด์นี้ถูกคำนวณโดยคำสั่ง **trustchk**

hash_value

การแฮชแบบเข้ารหัสของไฟล์ ค่า **VOLATILE** หมายถึง ไฟล์ที่เปลี่ยนแปลงบ่อย ฟิลด์นี้ถูกคำนวณโดยคำสั่ง **trustchk**

minslabel

กำหนดเลเบลระดับความลับต่ำสุดสำหรับอ็อบเจ็กต์

maxslabel

กำหนดเลเบลระดับความลับสูงสุดสำหรับอ็อบเจ็กต์ (ใช้ได้บนระบบ Trusted AIX) แอ็ตทริบิวต์นี้ ไม่สามารถใช้ได้กับไฟล์ปกติและ fifo

intlabel กำหนดเลเบล integrity สำหรับอ็อบเจ็กต์ (ใช้ได้บนระบบ Trusted AIX)

accessauths

กำหนดการอนุญาตเข้าถึงบนอ็อบเจ็กต์ (ใช้ได้บนระบบ Trusted AIX)

innateprivs

กำหนดสิทธิพิเศษ innate สำหรับไฟล์

proxyprivs

กำหนดสิทธิพิเศษ proxy สำหรับไฟล์

authprivs

กำหนดสิทธิพิเศษที่กำหนดให้แก่ผู้ใช้หลังจาก การอนุญาตที่กำหนด

secflags

กำหนดแฟล็กการรักษาความปลอดภัยไฟล์ที่เชื่อมโยงกับอ็อบเจ็กต์

t_accessauth

กำหนด Trusted AIX เพิ่มเติมที่มี การอนุญาตเข้าถึง Multi-Level Security (MLS) เจาะจง (ใช้ได้บน ระบบ Trusted AIX)

t_innateprivs

กำหนด Trusted AIX เพิ่มเติมด้วย สิทธิ MLS-specific innate สำหรับไฟล์ (ใช้ได้กับระบบ Trusted AIX)

t_proxyprivs

กำหนด Trusted AIX เพิ่มเติมด้วย สิทธิ MLS-specific proxy สำหรับไฟล์ (ใช้ได้กับระบบ Trusted AIX)

t_authprivs

กำหนด Trusted AIX เพิ่มเติมด้วย สิทธิ MLS-specific ที่ถูกกำหนดให้กับผู้ใช้หลังจากให้สิทธิ (ใช้ได้กับระบบ Trusted AIX)

t_secflags

กำหนด Trusted AIX เพิ่มเติมด้วย แฟลกความปลอดภัยไฟล์ MLS-specific ที่เชื่อมโยงกับอ็อบเจ็กต์ (ใช้ได้กับระบบ Trusted AIX)

เมื่อคุณเพิ่มรายการใหม่ใน TSD ถ้าไฟล์ที่ไว้วางใจมีลิงค์สัญลักษณ์ หรือ hard links ชี้ไปที่รายการใหม่ ลิงค์เหล่านี้จะถูกเพิ่มลงใน TSD ได้โดยใช้แอ็ตทริบิวต์ **symlinks** และ **hardlinks** ที่ บรรทัดคำสั่ง ร่วมกับคำสั่ง **trustchk** ถ้าไฟล์กำลังเพิ่มถูกคาดว่าจะมีการเปลี่ยนแปลงบ่อย ให้ใช้คีย์เวิร์ด VOLATILE ที่บรรทัดคำสั่ง ดังนั้นคำสั่ง **trustchk** จะไม่คำนวณ ฟิลด์ **hash_value** และ **signature** เมื่อสร้าง นิยามไฟล์สำหรับการเพิ่มลงใน TSD ระหว่างการตรวจสอบ integrity ของไฟล์นี้ ฟิลด์ **hash_value** และ **signature** จะ ถูกข้าม

ระหว่างการเพิ่มนิยามไฟล์ปกติลงใน TSD จำเป็นต้อง มีไพรเวตคีย์ (รูปแบบ ASN.1/DER) ใช้แฟล็ก **-s** และใบรับรองดิจิตัลที่มีพับลิกคีย์ที่ตรงกันโดยใช้ แฟล็ก **-v** ไพรเวตคีย์ถูกใช้เพื่อสร้างลายเซ็นของไฟล์และ จากนั้นจะถูกละทิ้ง ทั้งนี้ขึ้นอยู่กับผู้ใช้ที่จะต้องเก็บคีย์นี้อย่างปลอดภัย ใบรับรอง ถูกเก็บในที่เก็บใบรับรองในไฟล์ `/etc/security/certificates` สำหรับลายเซ็นที่จะถูกตรวจสอบเมื่อใดก็ตามที่คุณร้องขอให้ทำตรวจสอบ integrity เนื่องจากการคำนวณลายเซ็นไม่สามารถทำได้สำหรับไฟล์ที่ไม่ใช่ไฟล์ปกติ อย่างไดเร็กทอรี หรือไฟล์อุปกรณ์ จึงไม่มีการบังคับให้ต้องกำหนด ไพรเวตคีย์และใบรับรองขณะเพิ่มไฟล์เหล่านั้นลงใน TSD

คุณสามารถกำหนดนิยามไฟล์ที่คำนวณไว้ล่วงหน้า ผ่านไฟล์โดยใช้ตัวเลือก **-f** ที่จะเพิ่มใน TSD ในกรณีนี้ คำสั่ง **trustchk** จะคำนวณ ค่าและจัดเก็บนิยามใน TSD โดยไม่ตรวจสอบ ผู้ใช้ มีหน้าที่รับผิดชอบต่อความถูกต้องของนิยามไฟล์ในกรณีนี้

การตรวจสอบไลบรารีสนับสนุน

เพื่อ สนับสนุนการตรวจสอบไลบรารี ไฟล์ `tsd.dat` จะถูกเพิ่มในไดเร็กทอรี `/etc/security/tsd/lib/` ชื่อฐานข้อมูลคือ `/etc/security/tsd/lib/lib.tsd.dat` ฐานข้อมูลนี้ใช้สำหรับไลบรารี ซึ่งรวมถึงไฟล์ stanzas `.o` ของไลบรารีที่ไว้วางใจที่ตรงกันด้วย ทุกไฟล์ stanza `.o` ของไลบรารีอยู่ใน ฟอร์แมตที่ระบุไว้ในตัวอย่างต่อไปนี้

สำหรับไลบรารี `libc.a` ถ้าไฟล์ `strcmp.o` เป็นไฟล์ชนิด `.o` ชนิดใดชนิดหนึ่ง ไฟล์ stanza `strcmp.o` ใน `/etc/security/tsd/lib/lib.tsd.dat` จะมีลักษณะดังในตัวอย่างต่อไปนี้

```
/usr/lib/libc.a/strcmp.o:
   Type = OBJ
   Size = 2345
   Hash value
   Signature =
   Cert_tag =
```

ฐานข้อมูลนี้มีรายการที่ตรงกับ ไฟล์ **type, size hash, cert tag** และ **signature** `.o` ค่าแฮชของไลบรารีถูกอัพเดตใน ไฟล์ `/etc/security/tsd/tsd.dat` สำหรับ stanza ที่ตรงกัน ค่าแอ็ตทริบิวต์เหล่านี้ถูกสร้างในระหว่าง build และค่าถูกย้ายไปยังฐานข้อมูล `/etc/security/tsd/lib/lib.tsd.dat` ในระหว่างการติดตั้ง

ไฟล์ `/etc/security/tsd/tsd.dat` stanzas สำหรับไลบรารีถูกแก้ไขให้แสดงแอ็ตทริบิวต์ **type** เป็น `LIB` และแอ็ตทริบิวต์ **size** และ **signature** ว่าง ค่าปัจจุบันสำหรับแอ็ตทริบิวต์ **dynamica size, hash, signature** ยังคงเป็นค่า **VOLATILE** การตรวจสอบไลบรารี จะถูกข้ามระหว่างบูตระบบ เริ่มต้นด้วยรีลีส AIX 6.1.0, the **size, hash,** และ **signature** ของ stanzas ที่ไว้วางใจถูก

คำนวณด้วยไฟล์ .o ของไลบรารี ในระหว่างการติดตั้ง ฐานข้อมูล tsd.dat จะแสดงค่าที่คำนวณและไฟล์ .o stanza ที่ตรงกัน กับไลบรารีที่ไว้วางใจที่จัดเก็บในฐานข้อมูล /etc/security/tsd/lib/lib.tsd.dat

การเข้าถึงฐานข้อมูล TE แบบรีโมต:

นโยบาย Trusted Signature Database (TSD) ที่รวมศูนย์และ นโยบาย Trusted Execution (TE) สามารถนำไปใช้ในสภาวะแวดล้อมระบบของคุณ โดยการเก็บใน LDAP

ฐานข้อมูลที่ควบคุมนโยบาย TSD และนโยบาย TE ถูกเก็บแยกไว้สำหรับ แต่ละระบบ AIX นโยบาย TSD และนโยบาย TE แบบรวมศูนย์ถูกเก็บไว้ใน LDAP เพื่อให้สามารถจัดการได้ จากศูนย์กลาง การใช้นโยบาย TSD และนโยบาย TE ที่รวมศูนย์ช่วยให้คุณตรวจสอบ ว่านโยบายใน LDAP เป็นสำเนาต้นฉบับหรือไม่ และนโยบาย สามารถอัพเดตไคลเอ็นต์เมื่อใดก็ตามที่ไคลเอ็นต์ถูกติดตั้งใหม่ อัพเดต หรือการรักษาความปลอดภัยถูกฝ่าฝืน นโยบาย TE ที่รวมศูนย์อนุญาตให้มีหนึ่งตำแหน่ง ที่จะบังคับใช้นโยบาย TE โดยไม่จำเป็นต้องอัพเดตแต่ละไคลเอ็นต์แยกกัน นโยบาย TSD ที่รวมศูนย์ช่วยให้จัดการได้ง่ายกว่านโยบาย TDS ที่ไม่มีการรวมศูนย์

AIX ยูทิลีตี สามารถใช้เอ็กซ์พอร์ตข้อมูลนโยบาย TSD และนโยบาย TE โลคัลไปที่ LDAP, กำหนดคอนฟิกไคลเอ็นต์ให้ใช้ข้อมูลนโยบาย TSD และนโยบาย TE ใน LDAP, ควบคุมการค้นหาข้อมูลนโยบาย TSD และนโยบาย TE และจัดการข้อมูล LDAP จากระบบไคลเอ็นต์ ส่วนต่อไปนี้ให้ ข้อมูลเพิ่มเติมเกี่ยวกับคุณลักษณะเหล่านี้

การเอ็กซ์พอร์ตข้อมูลนโยบาย TSD และนโยบาย TE ไปยัง LDAP:

ในการใช้ LDAP เป็นที่เก็บกลางสำหรับนโยบาย TSD และนโยบาย TE เซิร์ฟเวอร์ LDAP ต้องได้รับข้อมูล นโยบายก่อน

เซิร์ฟเวอร์ LDAP ต้องมี schema ของนโยบาย TSD และนโยบาย TE สำหรับ LDAP ที่ติดตั้ง ก่อนที่ไคลเอ็นต์ LDAP จะสามารถใช้เซิร์ฟเวอร์ สำหรับข้อมูลนโยบาย Schema ของนโยบาย TSD และนโยบาย TE สำหรับ LDAP มีอยู่บนระบบ AIX ในไฟล์ **/etc/security/ldap/sec.ldif** Schema สำหรับ เซิร์ฟเวอร์ LDAP ต้องอัพเดตด้วยไฟล์นี้โดยใช้คำสั่ง **ldapmodify**

ในการระบุเวอร์ชันของฐานข้อมูล TE บนเซิร์ฟเวอร์ LDAP และทำให้ ไคลเอ็นต์ LDAP ทราบถึงเวอร์ชันเฉพาะนั้น คุณต้องตั้งค่าแอ็ตทริบิวต์ **databasename** ในไฟล์ **/etc/nscontrol.conf** แอ็ตทริบิวต์ **databasename** ใช้ชื่อใดๆ เป็นค่าแอ็ตทริบิวต์ และใช้โดยคำสั่ง **tetoldif** ขณะจัดการรูปแบบ ldif

ใช้คำสั่ง **tetoldif** เพื่ออ่านข้อมูลใน ไฟล์นโยบาย TSD และนโยบาย TE โลคัล และเอาต์พุตนโยบาย ในรูปแบบที่สามารถใช้ได้สำหรับ LDAP เอาต์พุตที่สร้างโดยคำสั่ง **tetoldif** สามารถบันทึกลงไฟล์ในรูปแบบ ldif และใช้เพื่อ populate เซิร์ฟเวอร์ LDAP ด้วยข้อมูลในคำสั่ง **ldapadd** ฐานข้อมูลต่อไปนี้บนระบบโลคัลถูกใช้โดยคำสั่ง **tetoldif** เพื่อสร้างข้อมูลนโยบาย TSD และนโยบาย TE สำหรับ LDAP:

- /etc/security/tsd/tsd.dat
- /etc/security/tsd/tepolicies.dat

configuration ไคลเอ็นต์ LDAP สำหรับนโยบาย TSD และนโยบาย TE:

ระบบต้องถูกตั้งค่าเป็นไคลเอ็นต์ LDAP เพื่อใช้ข้อมูลนโยบาย TSD และนโยบาย TE ที่เก็บใน LDAP

ใช้คำสั่ง AIX **/usr/sbin/mksecldap** เพื่อตั้งค่าระบบเป็นไคลเอ็นต์ LDAP คำสั่ง **mksecldap** ค้นหาเซิร์ฟเวอร์ LDAP แบบไดนามิกที่ระบบใช้เพื่อพิจารณาตำแหน่ง ของข้อมูลนโยบาย TSD และนโยบาย TE และบันทึกผลลัพธ์ ลงไฟล์ **/etc/security/ldap/ldap.cfg**

หลังการตั้งค่าระบบเป็นไคลเอ็นต์ LDAP สำเร็จด้วย คำสั่ง **mksecldap** ระบบต้องถูกตั้งค่า เพื่อเปิดใช้งาน LDAP เป็นโดเมน การค้นหาข้อมูลนโยบาย TSD และนโยบาย TE โดยการตั้งค่า secorder ของไฟล์ **/etc/nscontrol.conf**

เมื่อระบบได้รับการตั้งค่าเป็นไคลเอ็นต์ LDAP และโดเมนการค้นหา ข้อมูลนโยบาย TSD และนโยบาย TE daemon ไคลเอ็นต์ **/usr/sbin/secldapclntd** จะเรียกออกมาข้อมูลนโยบาย TSD และนโยบาย TE จากเซิร์ฟเวอร์ LDAP เมื่อใดก็ตามที่ **trustchk** ใดๆ ถูกดำเนินการ บนไคลเอ็นต์ LDAP

การเปิดใช้งาน LDAP ด้วยคำสั่ง trustchk:

คำสั่งการจัดการฐานข้อมูลนโยบาย TSD และนโยบาย TE ทั้งหมด ถูกเปิดให้ใช้ฐานข้อมูลนโยบาย LDAP TSD และ นโยบาย TE

ใช้คำสั่ง **trustchk** ที่มีแฟล็ก **−R** เพื่อดำเนินการตั้งค่าเริ่มต้นของฐานข้อมูล LDAP การตั้งค่าเริ่มต้นเกี่ยวข้องกับ การเพิ่ม นโยบาย TSD นโยบาย TE, DNs ฐาน และการสร้าง ไฟล์ **/etc/security/tsd/ldap/tsd.dat** ฐานข้อมูลโลคัลและไฟล์ **/etc/security/tsd/ldap/tepolicies.dat**

ถ้าคำสั่ง **trustchk** ถูกรันโดยมีแฟล็ก **−R** โดยใช้อ็อพชัน LDAP การดำเนินการจะยึดตามข้อมูลเซิร์ฟเวอร์ LDAP ถ้าคำสั่ง **trustchk** ถูกรันโดยมีแฟล็ก **−R** โดยใช้อ็อพชันไฟล์ การดำเนินการจะยึดตามข้อมูลฐานข้อมูล โลคัล ค่าดีฟอลต์สำหรับแฟล็ก **−R** คือใช้อ็อพชันไฟล์

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mksecldap

คำสั่ง trustchk

การตรวจสอบ integrity ของ Trusted Signature Database:

คำสั่ง **trustchk** สามารถใช้เพื่อ ตรวจสอบสถานะ integrity ของนิยามไฟล์ใน Trusted Signature Database (TSD) เทียบกับไฟล์จริง

ถ้าคำสั่ง **trustchk** ระบุความผิดปกติ สามารถทำให้แก้ไขโดยอัตโนมัติ ได้ หรือพร้อมต์ผู้ใช้ก่อนพยายามทำการแก้ไข ถ้าความผิดปกติเช่น size, signature, cert_tag หรือ hash_value ไม่ตรง การแก้ไข ไม่สามารถทำได้ ในกรณีเช่นนั้น คำสั่ง **trustchk** จะกำหนดให้ไฟล์ไม่สามารถเข้าถึงได้ ด้วยเหตุนั้น rendering ไม่สามารถทำได้และ มีความเสียหายใดๆ

การดำเนินการแก้ไขต่อไปนี้จะเกิดขึ้นเมื่อมีแอ็ตทริบิวต์ไม่ตรงกัน แตกต่างกัน:

owner เจ้าของไฟล์จะถูกตั้งค่าให้เป็นค่าใน TSD

group กลุ่มของไฟล์จะถูกตั้งค่าให้เป็นค่าใน TSD

mode บิตโหมดของไฟล์จะถูกตั้งค่าให้เป็นค่าใน TSD

hardlinks
ถ้าลิงก์ชี้ไปที่ไฟล์อื่น จะถูกแก้ไขให้ชี้ ไปที่ไฟล์นี้ ถ้าไม่มีลิงก์ ลิงก์ใหม่จะถูกสร้างขึ้นเพื่อ ให้ชี้ไปที่ไฟล์นี้

symlinks
เหมือนกับ hardlinks

type ไฟล์ถูกกำหนดไม่ให้เข้าถึงได้

size ไฟล์ถูกกำหนดไม่ให้เข้าถึงได้ ยกเว้นในกรณีของไฟล์ **VOLATILE**

cert_tag
ไฟล์ถูกกำหนดไม่ให้เข้าถึงได้

signature
ไฟล์ถูกกำหนดไม่ให้เข้าถึงได้ ยกเว้นในกรณีของไฟล์ **VOLATILE**

hash_value
ไฟล์ถูกกำหนดไม่ให้เข้าถึงได้ ยกเว้นในกรณีของไฟล์ **VOLATILE**

minslabel
บนระบบ Trusted AIX ระดับความลับต่ำสุดถูกตั้งค่าเป็นค่าใน TSD

maxslabel
บนระบบ Trusted AIX ระดับความลับสูงสุดถูกตั้งค่าเป็นค่าใน TSD

intlabel บนระบบ Trusted AIX เลเบล integrity ถูกตั้งค่าเป็นค่าใน TSD

accessauths
การอนุญาตเข้าถึงถูกตั้งค่าให้เป็นค่าใน TSD บน Trusted AIX ค่า **t_accessauths** ถูกพิจารณาร่วมกับแอ็ตทริบิวต์ **accessauths**

innateprivs
สิทธิพิเศษ innate ถูกตั้งค่าให้เป็นค่าใน TSD บน Trusted AIX ค่า **t_innateprivs** ถูกพิจารณาร่วมกับแอ็ตทริบิวต์ **innateprivs**

inheritprivs
สิทธิพิเศษ inheritable ถูกตั้งค่าให้เป็นค่าใน TSD บน Trusted AIX ค่า **t_inheritprivs** ถูกพิจารณาร่วมกับแอ็ตทริบิวต์ **accessauths**

authprivs
สิทธิพิเศษ authorized ถูกตั้งค่าให้เป็นค่าใน TSD บน Trusted AIX ค่า **t_authprivs** ถูกพิจารณาร่วมกับแอ็ตทริบิวต์ **authprivs**

aecflags
แฟล็กความปลอดภัยถึงถูกตั้งค่าให้เป็นค่าใน TSD บน Trusted AIX ค่า **t_secglags** ถูกพิจารณาเป็นส่วนหนึ่งของแอ็ตทริบิวต์ **secflags**

คุณยังสามารถตรวจสอบความถูกต้องนิยามไฟล์เทียบกับฐานข้อมูลอื่น ได้โดยใช้อ็อพชัน **-F** ผู้ดูแลระบบควรเลี่ยง การเก็บ TSD บนระบบเดียวกัน และสำรองข้อมูลฐานข้อมูลไว้ที่ตำแหน่ง ที่เป็นทางเลือกอื่น file integrity นี้สามารถทำให้ตรงกับ เวอร์ชันสำเนาสำรองนี้ของ TSD ได้โดยใช้อ็อพชัน **-F**

การตั้งค่านโยบายการรักษาความปลอดภัย:

คุณลักษณะ Trusted Execution (TE) ช่วยให้คุณมีกลไก การตรวจสอบ file integrity แบบรันไทม์ การใช้กลไกนี้ ระบบสามารถถูกตั้งค่าเพื่อตรวจสอบ integrity ของไฟล์ที่ไว้วางใจ ก่อนที่การร้องขอทั้งหมดจะเข้าถึงไฟล์เหล่านั้น การอนุญาตได้อย่างมีประสิทธิภาพ ให้ไฟล์ที่ไว้วางใจที่ผ่านการตรวจสอบ integrity เท่านั้นที่สามารถเข้าถึงได้ บนระบบ

เมื่อไฟล์ถูกทำเครื่องหมายว่าไว้วางใจ (โดยการเพิ่มนิยามใน Trusted Signature Database) คุณลักษณะ TE สามารถถูกทำให้มอนิเตอร์ integrity บนทุกการเข้าถึง TE ยังสามารถมอนิเตอร์ระบบอย่างต่อเนื่องและสามารถ ตรวจหาการพยายามเปลี่ยนแปลงไฟล์ที่ไว้วางใจใดๆ (โดยผู้ใช้หรือแอ็พพลิเคชัน ที่เป็นอันตราย) ที่แสดงบนระบบขณะรันไทม์ (ตัวอย่าง ในตอน โหลด) ถ้าไฟล์ถูกพบว่าได้รับการเปลี่ยนแปลงเพื่อทำลาย TE สามารถดำเนินการแก้ไข ได้โดยยึดตามนโยบายที่ตั้งค่าไว้แล้ว เช่นการไม่อนุญาตให้มีการทำงาน, การเข้าถึงไฟล์ หรือการบันทึกข้อผิดพลาด ถ้าไฟล์กำลังถูกเปิดหรือถูกดำเนินงาน และมีรายการใน Trusted Signature Database (TSD) TE จะดำเนินการ ดังนี้:

- ก่อนการโหลดไบนารี คอมโพเนนต์ที่รับผิดชอบการโหลด ไฟล์ (system loader) ร้องขอระบบย่อย Trusted Execution และคำนวณค่าการแฮชโดยใช้อัลกอริทึม SHA-256 (ตั้งค่าได้)
- ค่าการแฮชที่คำนวณขณะรันไทม์นี้ต้องตรงกับค่าที่เก็บ ใน TSD
- ถ้าค่าตรงกัน การเปิด หรือการดำเนินงานไฟล์จะได้รับอนุญาต
- ถ้าค่าไม่ตรง อาจเป็นที่ไบนารีถูกเปลี่ยนแปลง หรือ มีเกิดช่องโหว่บางอย่าง ทั้งนี้ขึ้นอยู่กับผู้ใช้จะตัดสินใจเลือกการดำเนินการ ที่ใช้จัดการ กลไก TE มีอ็อพชันให้ผู้ใช้เลือกตั้งค่า นโยบายของตนเองสำหรับการดำเนินการที่จะใช้จัดการถ้าค่าการแฮช ไม่ตรง
- การดำเนินการเกี่ยวข้องจะถูกกระทำ โดยยึดตามนโยบายที่ตั้งค่าไว้เหล่านี้

นโยบายต่อไปนี้สามารถตั้งค่าได้

CHKEXEC
: ตรวจสอบค่าการแฮชของไฟล์ที่ดำเนินงานได้ที่ไว้วางใจเท่านั้นก่อนการโหลด เข้าสู่หน่วยความจำเพื่อใช้ดำเนินการ

CHKSHLIBS
: ตรวจสอบค่าการแฮชของไลบรารีที่แบ่งใช้ที่ไว้วางใจเท่านั้นก่อน โหลดเข้าสู่หน่วยความจำเพื่อใช้ดำเนินการ

CHKSCRIPTS
: ตรวจสอบค่าการแฮชของเชลล์สคริปต์ที่ไว้วางใจเท่านั้นก่อน โหลดเข้าสู่หน่วยความจำ

CHKKERNEXT
: ตรวจสอบค่าการแฮชของส่วนขยายเคอร์เนลเท่านั้นก่อนการโหลด เข้าสู่หน่วยความจำ

STOP_UNTRUSTD
: หยุดการโหลดไฟล์ที่ไม่ไว้วางใจ เฉพาะไฟล์ที่อยู่ใน TSD เท่านั้นที่ถูกโหลด นโยบายนี้ใช้งานได้เฉพาะเมื่อใช้ร่วมกับนโยบาย CHK* ใดๆ ที่กล่าวถึงด้านบน ตัวอย่าง ถ้า **CHKEXEC=ON** และ **STOP_UNTRUSTD=ON** ไบนารีที่ดำเนินงานได้ใดๆ ที่ไม่เป็นของ TSD จะถูกบล็อก มิให้ดำเนินการ

STOP_ON_CHKFAIL
: หยุดการโหลดไฟล์ที่ไว้วางใจที่ไม่ผ่านการตรวจสอบค่าการแฮช นโยบายนี้ ยังใช้ร่วมกับนโยบาย CHK* ได้ ตัวอย่าง ถ้า **CHKSHLIBS=ON** และ **STOP_ON_CHKFAIL=ON** ไลบรารีที่แบ่งใช้ใดๆ ที่ไม่เป็นของ TSD จะถูกบล็อกมิให้โหลดเข้าสู่ หน่วยความจำเพื่อใช้งาน

TSD_LOCK
: ล็อก TSD เพื่อไม่ให้ทำการแก้ไข

TSD_FILES_LOCK
: ล็อกไฟล์ที่ไว้วางใจ ค่านี้ไม่อนุญาตให้เกิดไฟล์ที่ไว้วางใจ ในโหมดเขียน

TE เปิดใช้งาน/ปิดใช้งานการทำงานของ Trusted Execution เฉพาะเมื่อค่านี้ เปิดใช้งานเท่านั้น ที่นโยบายที่กล่าวถึงด้านบนจะมีผลใช้ได้

ตารางต่อไปนี้ที่มีการโต้ตอบระหว่างนโยบาย CHK* และนโยบาย STOP* ที่ต่างกันเมื่อเปิดใช้งาน

นโยบาย	STOP_UNTRUSTD	STOP_ON_CHKFAIL
CHKEXEC	หยุดการโหลดไฟล์ที่ดำเนินงานได้ที่ไม่เป็น ของ TSD	หยุดการโหลดไฟล์ที่ดำเนินงานได้ที่มีค่าการแฮช ไม่ตรงกับค่า TSD
CHKSHLIBS	หยุดการโหลดไลบรารีที่แบ่งใช้ที่ไม่เป็น ของ TSD	หยุดการโหลดไลบรารีที่แบ่งใช้ที่มีค่าการแฮช ไม่ตรงกับค่า TSD
CHKSCRIPTS	หยุดการโหลดเชลล์สคริปต์ที่ไม่เป็น ของ TSD	หยุดการโหลดเชลล์สคริปต์ที่มีค่าการแฮช ไม่ตรงกับค่า TSD
CHKKERNEXT	หยุดการโหลดส่วนขยายเคอร์เนลที่ไม่เป็น ของ TSD	หยุดการโหลดส่วนขยายเคอร์เนลที่มีค่าการแฮช ไม่ตรงกับค่า TSD

หมายเหตุ: นโยบายสามารถถูกเปิดใช้งานหรือปิดใช้งานได้ตลอดเวลาจนกว่า TE ถูกเปิดใช้เพื่อให้นโยบายมีผลใช้ เมื่อนโยบายมีผลใช้ การปิดใช้งานนโยบายนั้นจะมีผลกระทบต่อวัฏจักรการเปิดเครื่องใหม่ในครั้งหน้า เท่านั้น ข้อความข้อมูลทั้งหมดจะถูกบันทึกลงใน **syslog**

ข้อมูลที่เกี่ยวข้อง:

เคอร์เนลเซอร์วิส TE_verify_reg

เคอร์เนลเซอร์วิส TE_verify_unreg

Trusted Execution Path และ Trusted Library Path:

Trusted Execution Path (TEP) กำหนดรายการไดเร็กทอรีที่ มีไฟล์การดำเนินการที่ไว้วางใจ เมื่อเปิดใช้การตรวจสอบ TEP system loader อนุญาตให้ไบนารีในพาธที่ระบุเท่านั้นที่สามารถดำเนินงานได้ Trusted Library Path (TLP) มีฟังก์ชันการทำงานแบบเดียวกัน ยกเว้นแต่จะถูกใช้เพื่อกำหนด ไดเร็กทอรีที่สามารถเก็บไดเร็กทอรีที่ไว้วางใจของระบบ

เมื่อเปิดใช้ TLP system loader อนุญาตให้ไลบรารีจากพาธนี้ เท่านั้นที่จะลิงก์กับไบนารี คำสั่ง **trustchk** สามารถใช้ เพื่อเปิดใช้งานหรือปิดใช้งาน TEP หรือ TLP รวมถึงรายการพาธที่คั่นด้วยโคลอน สำหรับทั้งคู่โดยใช้แอ็ตทริบิวต์บรรทัดคำสั่ง TEP และ TLP ของคำสั่ง **trustchk**

Trusted Shell และ Secure Attention Key:

Trusted Shell และ Secure Attention Key (SAK) ทำหน้าที่คล้ายกับ Trusted Computing Base (TCB) ยกเว้นถ้า Trusted Execution ถูกเปิดใช้งาน บนระบบแทน TCB นั้น Trusted Shell จะทำงานไฟล์ที่เป็นของ Trusted Signature Database เท่านั้น

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ TCB และ SAK ดูที่ Trusted Computing Base, Using the Secure Attention Key, and Configuring the Secure Attention Key

ฐานข้อมูลนโยบาย Trusted Execution (TE):

นโยบาย Trusted Execution (TE) ถูกเก็บในไฟล์ **/etc/security/tsd/tepolicies.dat** พาธสำหรับนโยบาย TE ถูกแสดงด้วย ไดเร็กทอรี TLP และ ไดเร็กทอรี TEP

Security Profile Evaluation Assurance Level 4+ และ Labeled AIX Security and Evaluation Assurance Level 4+

ผู้ดูแลระบบสามารถติดตั้งระบบที่มีตัวเลือก Base AIX Security (BAS) และ ระดับการประเมินการรับประกัน 4+ (EAL4+) หรือ Labeled AIX Security (LAS) และ Evaluation Assurance Level 4+ (EAL4+) ในระหว่างติดตั้งระบบปฏิบัติการพื้นฐาน (BOS) ระบบ ที่มีอ็อพชันเหล่านั้นมีการจำกัดบนซอฟต์แวร์ที่ติดตั้ง ระหว่างการติดตั้ง BOS รวมถึงการเข้าถึงเน็ตเวิร์กจะถูกจำกัดด้วย

หมายเหตุ: การประเมินผลกำลังดำเนินอยู่ในขณะนี้สำหรับ AIX เวอร์ชัน 7.1 โปรดอ้างอิงรีลีสโน้ต AIX เวอร์ชัน 7.1 สำหรับข้อมูลล่าสุด

ภาพรวมเกี่ยวกับ Security profile:

Security profile เป็นผลิตภัณฑ์ที่ระบุข้อกำหนดด้านความปลอดภัย สำหรับระบบปฏิบัติการที่มีวัตถุประสงค์ทั่วไปในสภาวะแวดล้อมแบบเครือข่าย โปรไฟล์นี้ระบุข้อกำหนดที่จำเป็น เพื่อบรรลุวัตถุประสงค์ของฟังก์ชันความปลอดภัยของ Target of evaluation (TOE) และสภาวะระบบ

Security profile ประกอบด้วยแพ็กเกจฐานและแพ็กเกจส่วนขยายจำนวนมาก ผลิตภัณฑ์ที่เกี่ยวข้องกับการสนับสนุนแพ็กเกจฐานของ Security profile คือ Identification and Authentication, Discretionary Access Control (DAC), Auditing, Cryptographic Services, Management of Security Mechanisms, และ Trusted Channel communications Security profile มีแพ็กเกจ ที่เป็นทางเลือกสำหรับ Labeled Security, Integrity Verification, Advanced Audit, General Purpose Cryptography, Advanced Management, Extended Identification and Authentication, Trusted Boot และ Virtualization

ข้อสมมติฐาน

- สภาวะแวดล้อมการใช้งานสำหรับ TOE:

  ข้อสมมติฐานทั้งหมดในส่วนนี้ อ้างอิงกับ Base AIX Security (โหมด BAS) และ Labeled AIX Security (โหมดLAS) เว้นแต่ว่า จะระบุไว้เป็นอย่างอื่น ข้อสมมติฐานทั้งหมดที่เกี่ยวข้องกับ Virtual input output server (VIOS) จะมีเครื่องหมาย VIOS แสดงชัดเจน VIOS ไม่ได้แบ่งใช้สมมติฐานร่วมกับระบบปฏิบัติการ AIXAIX หรือ Trusted AIX
- Physical:

  ระบบ IT มี TOE ที่มีความปลอดภัยทางกายภาพ ที่เหมาะสมที่เพียงพอกับมูลค่าสินทรัพย์ IT ที่ได้รับการป้องกันโดย TOE

  หมายเหตุ: VIOS เท่านั้น: ระบบการทำงาน มี TOE ที่มีความปลอดภัยทางกายภาพที่เหมาะสมซึ่ง เพียงพอกับมูลค่าสินทรัพย์ IT ที่ป้องกันโดย TOE
- การบริหาร:
  - ฟังก์ชันความปลอดภัย TOE จัดการโดยแต่ละบุคคลที่มีความเชี่ยวชาญ ผู้ดูแลระบบจะต้องรอบคอบ ไม่ละเว้นหรือไม่มุ่งร้าย และปฏิบัติตามคำแนะนำ ที่จัดหาให้ตามเอกสารคู่มือ
  - ผู้ใช้ที่มีสิทธิสามารถเข้าถึงข้อมูลบางอย่างที่จัดการโดย TOE และถูกคาดหวังให้ปฏิบัติงานอย่างร่วมมือ
  - ผู้ใช้ได้รับการฝึกฝนอย่างเพียงพอและได้รับความไว้วางใจให้ทำงาน หรือกลุ่มงานให้เสร็จสิ้นภายในระบบ IT ที่ปลอดภัย ซึ่งต้องบังคับใช้ ระบบควบคุมข้อมูลของผู้ใช้โดยสมบูรณ์
  - VIOS เท่านั้น: ฟังก์ชันความปลอดภัย TOE จะได้รับการจัดการโดย ผู้เชี่ยวชาญหนึ่งคนหรือหลายคน ผู้ดูแลระบบจะต้องรอบคอบ ไม่ละเว้นหรือไม่มุ่งร้าย และปฏิบัติตามคำแนะนำ ที่จัดหาให้ตามเอกสารคู่มือ

- VIOS เท่านั้น: ผู้ใช้ที่มีสิทธิมีสิทธิที่จำเป็น ในการเข้าถึงข้อมูลบางอย่างที่จัดการโดย TOE และถูกคาดหวังให้ปฏิบัติงานอย่างร่วมมือ
- VIOS เท่านั้น: ผู้ใช้ได้รับการฝึกฝนอย่างเพียงพอและได้รับความไว้วางใจให้ทำงาน หรือกลุ่มงานให้เสร็จสิ้นภายในระบบการทำงานที่ปลอดภัย ซึ่งต้องบังคับใช้ ระบบควบคุมข้อมูลของผู้ใช้โดยสมบูรณ์

• ขั้นตอน:
- การแก้ไขหรือความเสียหายใดๆ ของไฟล์ที่จัดการความปลอดภัยหรือเกี่ยวข้อง กับความปลอดภัยของ TOE ที่ผู้ใช้หรือระบบที่ใช้งานเป็นสาเหตุให้เกิดขึ้น โดยไม่ได้ตั้งใจหรือโดยอุบัติเหตุต้องตรวจหาด้วยผู้ใช้ที่เป็น ผู้ดูแลระบบ
- ระบบ IT ที่ไว้วางใจระยะไกลทั้งหมดที่ไว้วางใจโดย Target Security Function (TSF) เพื่อให้ข้อมูลหรือบริการ TSF แก่ TOE หรือสนับสนุน TSF ในการบังคับใช้การตัดสินใจนโยบายความปลอดภัย ถูกคาดว่า จะอยู่ภายใต้การควบคุมการจัดการเดียวกันและทำงานภายใต้ข้อจำกัดนโยบาย ความปลอดภัยที่ใช้ร่วมกับนโยบายความปลอดภัยของ TOE ได้
- ระบบ IT ที่ไว้วางใจในระยะไกลทั้งหมดที่ไว้วางใจโดย TSF ในการให้ข้อมูลหรือบริการ TSF แก่ TOE หรือในการสนับสนุน TSF ในการบังคับใช้ การติดสินใจนโยบายความปลอดภัย คาดว่าจะใช้ฟังก์ชัน ที่ถูกใช้โดย TSF อย่างถูกต้องโดยสอดคล้องกับข้อสมมติฐาน ที่กำหนดไว้ในฟังก์ชันนี้
- ความถูกต้องของข้อมูลต่อไปนี้ได้รับการรับรอง:
  - โค้ด TSF ทั้งหมดรวมถึงฟังก์ชันการตรวจสอบความถูกต้องที่ ถูกโหลดและรันก่อนเริ่มกลไกการตรวจสอบความถูกต้อง
  - ข้อมูล TSF ทั้งหมดรวมถึงข้อมูล TSF ที่ทำการตรวจสอบความถูกต้อง ที่ใช้โดยโค้ด TSF ที่โหลดและรันก่อนเริ่มกลไกการตรวจสอบ ความถูกต้อง
- VIOS เท่านั้น: การแก้ไขหรือความเสียหายใดๆ ของไฟล์ที่จัดการความปลอดภัยหรือเกี่ยวข้อง กับความปลอดภัยของ TOE ที่ผู้ใช้หรือระบบที่ใช้งานเป็นสาเหตุให้เกิดขึ้น โดยไม่ได้ตั้งใจหรือโดยอุบัติเหตุต้องตรวจหาด้วยผู้ใช้ที่เป็น ผู้ดูแลระบบ

• การเชื่อมต่อ: ทุกการเชื่อมต่อไปยังและจากระบบ IT ที่ไว้วางใจระยะไกล และระหว่างส่วนที่แยกต่างหากจากกันของ TSF ที่ไม่ได้รับการป้องกันโดย TSF จะได้รับการป้องกันทางกายภาพและทางลอจิกภายในระบบ TOE เพื่อให้มั่นใจว่าความถูกต้องและความลับของข้อมูล ถูกส่งและเพื่อให้มั่นใจว่าเมื่อมีการพิสูจน์ตัวตนของ end points การสื่อสาร

การขอรับซอฟต์แวร์

ติดต่อขอรับซอฟต์แวร์ ทำตามขั้นตอนต่อไปนี้:

1. ดาวน์โหลดผลิตภัณฑ์
2. คลิกที่ Help จากเมนู การสนับสนุนซอฟต์แวร์ Entitled ทางด้านซ้าย เกณฑ์ทั่วไปที่ประเมินการกำหนดค่า กำหนดให้ขอรับผลิตภัณฑ์และการอัพเดต จากมีเดียระบบหรือใช้ download director

สำหรับข้อมูลเกี่ยวกับการติดตั้งผลิตภัณฑ์ ใช้การติดตั้งระบบ BAS /EAL4+

การติดตั้งระบบ BAS/EAL4+:

RBAC ถูกเปิดใช้งานโดยอัตโนมัติเมื่อเลือกอ็อพชันนี้

ในการตั้งค่าตัวเลือก BAS/EAL4+ ระหว่างติดตั้ง BOS ให้ทำดังนี้:

1. ในหน้าจอ Installation and Settings เลือก **More Options**

2. ภายใต้ตัวเลือก More เลือก **Yes** สำหรับตัวเลือก BAS/EAL4+ ถ้าคุณใช้ WPAR เลือก **No** สำหรับตัวเลือก TCB ถ้าคุณใช้ไฟล์ `bosinst.data` ที่ปรับแต่งในการติดตั้งแบบไม่มีพร้อมต์ ตัวเลือก TCB จะตั้งค่าไปที่ **Yes**

ปิดใช้งานล็อกอิน root ระยะไกลในการติดตั้ง BAS เพื่อปิดใช้ล็อกอิน root ระยะไกล ให้รันคำสั่งต่อไปนี้หลังการติดตั้ง:

```
/usr/bin/chuser rlogin=false subgroups=SUADMIN root
```

เพิ่ม ผู้ใช้ที่เป็นผู้ดูแลระบบในกลุ่ม **SUADMIN** เพื่อเข้าถึง **su**

ตัวเลือก **เปิดใช้เทคโนโลยี BAS และ EAL4+** ที่เปิดใช้มีอยู่เฉพาะในเงื่อนไขต่อไปนี้เท่านั้น:

- วิธีการติดตั้งถูกตั้งเป็นการติดตั้งใหม่และเขียนทับโดยสมบูรณ์
- เลือกใช้ภาษาอังกฤษ
- เปิดใช้งานเคอร์เนล 64 บิต
- เปิดใช้งาน enhanced journaled file system (JFS2)

เมื่อตัวเลือก **เปิดใช้เทคโนโลยี BAS และ EAL4+** ตั้งค่าเป็น Yes ตัวเลือก **Trusted Computing Base** จะตั้งค่าเป็น Yes ด้วย และตัวเลือก **Desktop** เป็น NONE หรือ CDE

ถ้าคุณติดตั้งโดยไม่มีพร้อมต์โดยใช้ไฟล์ `bosinst.data` ต้องตั้งค่าฟิลด์ **INSTALL_TYPE** เป็น `CC_EVAL` และ ตั้งค่าฟิลด์ต่อไปนี้ดังนี้:

```
control_flow:
 CONSOLE = ???
 PROMPT = yes
 INSTALL_TYPE = CC_EVAL
 INSTALL_METHOD = overwrite
 TCB = yes
 DESKTOP = NONE or CDE
 ENABLE_64BIT_KERNEL = yes
 CREATE_JFS2_FS = yes
 ALL_DEVICES_KERNELS = no
 FIREFOX_BUNDLE = no
 HTTP_SERVER_BUNDLE = no
 KERBEROS_5_BUNDLE = no
 SERVER_BUNDLE = no
 ALT_DISK_INSTALL_BUNDLE = no

locale:
 CULTURAL_CONVENTION = en_US or C
 MESSAGES = en_US or C
```

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ RBAC ดูที่ Role Based Access Control (RBAC)

ระบบการจัดการการติดตั้งเครือข่ายสำหรับ BAS/EAL4+:

การติดตั้งของไคลเอ็นต์เทคโนโลยี BAS/EAL4+ สามารถดำเนินการได้โดยใช้สภาวะแวดล้อม Network Installation Management (NIM)

NIM มาสเตอร์ถูกตั้งค่าเพื่อจัดให้มีรีซอร์สที่จำเป็นสำหรับ การติดตั้งระดับ BAS/EAL4+ ที่เหมาะสมของ AIX 7.1 ไคลเอ็นต์ NIM อาจถูกติดตั้งโดยใช้รีซอร์สที่อยู่บน NIM มาสเตอร์ คุณสามารถติดตั้ง NIM ของไคลเอนต์โดยตั้งค่า ฟิลด์ต่อไปนี้ในรีซอร์ส `bosinst_data`:

```
control_flow:
 CONSOLE = ???
 PROMPT = no
 INSTALL_TYPE = CC_EVAL
 INSTALL_METHOD = overwrite
 TCB = yes
 DESKTOP = NONE or CDE
 ENABLE_64BIT_KERNEL = yes
 CREATE_JFS2_FS = yes
 ALL_DEVICES_KERNELS = no
 FIREFOX_BUNDLE = no
 HTTP_SERVER_BUNDLE = no
 KERBEROS_5_BUNDLE = no
 SERVER_BUNDLE = no
 ALT_DISK_INSTALL_BUNDLE = no

locale:
 CULTURAL_CONVENTION = en_US or C
 MESSAGES = en_US or C
```

NIM มาสเตอร์ไม่สามารถตั้งค่าเป็นระบบ BAS/EAL4+ และไม่สามารถเชื่อมต่อ กับเน็ตเวิร์กที่มีระบบ BAS/EAL4+ อื่นเหมือนกัน เมื่อเริ่มต้นกำหนดค่า การติดตั้งจาก NIM ไคลเอ็นต์ อ็อพชันเมนู **Remain NIM client after install SMIT** ต้องถูกตั้งค่าเป็น No หลังจาก ติดตั้งไคลเอ็นต์ NIM เป็นระบบ BAS/EAL4+ แล้ว ไคลเอ็นต์ NIM ต้องถูกลบออกจากเน็ตเวิร์กของ NIM มาสเตอร์ และการติดตั้งซอฟต์แวร์และการอัพเดต เพิ่มเติมไม่สามารถใช้กับ NIM มาสเตอร์ได้

สถานการณ์ตัวอย่างคือมีสภาวะแวดล้อมเน็ตเวิร์กสองแบบ เน็ตเวิร์ก แบบแรกมี NIM มาสเตอร์และระบบที่ไม่ใช่ BAS/EAL4+ เน็ตเวิร์กแบบที่สอง มีเฉพาะระบบ BAS/EAL4+ เท่านั้น ดำเนิน การติดตั้ง NIM บนไคลเอ็นต์ NIM หลังการติดตั้งเสร็จเรียบร้อย ยกเลิกการเชื่อมต่อระบบ BAS/EAL4+ ที่ติดตั้งใหม่จากเน็ตเวิร์กของ NIM มาสเตอร์ และเชื่อมต่อระบบกับเน็ตเวิร์กที่ประเมินผล

ตัวอย่างที่สองประกอบด้วยหนึ่งเน็ตเวิร์ก NIM มาสเตอร์ไม่ได้ เชื่อมต่อกับเน็ตเวิร์กเมื่อระบบอื่นกำลังดำเนินงานในการติดตั้งที่ประเมินผล และระบบ BAS/EAL4+ ไม่เชื่อมต่อกับ เน็ตเวิร์กระหว่างการติดตั้ง NIM

ซอฟต์แวร์บันเดิล BAS/EAL4+:

เมื่อเลือกตัวเลือก **BAS/EAL4+** เนื้อหาของกลุ่มติดตั้ง `/usr/sys/inst.data/sys_bundles/CC_EVAL.BOS.autoi` จะถูกติดตั้ง

คุณสามารถเลือกติดตั้งกลุ่มซอฟต์แวร์กราฟิกและ กลุ่มซอฟต์แวร์บริการเอสกสารด้วยตัวเลือก **BAS/EAL4+** ที่เลือก ถ้าคุณเลือกตัวเลือก **Graphics Software** กับตัวเลือก **BAS/EAL4+** เนื้อหากลุ่มซอฟต์แวร์ `/usr/sys/inst.data/sys_bundles/CC_EVAL.Graphics.bnd` จะถูกติดตั้ง ถ้าคุณเลือกตัวเลือกซอฟต์แวร์บริการเอกสาร กับตัวเลือก **BAS/EAL4+** เนื้อหา กลุ่มซอร์ฟแวร์ `/usr/sys/inst.data/sys_bundles/CC_EVAL.DocServices.bnd` จะถูกติดตั้ง

หลังจากติดตั้ง Licensed Program Products (LPPs) ระบบ จะเปลี่ยนการกำหนดค่าดิฟอลต์ตามข้อกำหนด BAS/EAL4+ การเปลี่ยนแปลง ต่อไปนี้จะถูกทำกับการตั้งค่าดีฟอลต์:

- ลบ /dev/echo ออกจากไฟล์ /etc/pse.conf
- เตรียมสร้าง instance อุปกรณ์แบบสตรีม
- อนุญาตให้ root เท่านั้นที่เข้าถึงสื่อบันทึกแบบถอดออกได้
- ลบรายการที่มิใช่ CC ออกจากไฟล์ inetd.conf
- เปลี่ยนแปลงสิทธิของไฟล์ต่างๆ
- เรจิสเตอร์ลิงก์สัญลักษณ์ในไฟล์ sysck.cfg
- เรจิสเตอร์อุปกรณ์ในไฟล์ sysck.cfg
- ตั้งค่าแอ็ตทริบิวต์ผู้ใช้และพอร์ต
- ตั้งค่าแอ็พพลิเคชัน doc_search สำหรับใช้เบราว์เซอร์
- ลบ httpdlite ออกจากไฟล์ inittab
- ลบ writesrv ออกจากไฟล์ inittab
- ลบ mkatmpvc ออกจากไฟล์ inittab
- ลบ atmsvcd ออกจากไฟล์ inittab
- ปิดใช้งาน snmpd ในไฟล์ /etc/rc.tcpip
- ปิดใช้งาน hostmibd ในไฟล์ /etc/rc.tcpip
- ปิดใช้งาน snmpmibd ในไฟล์ /etc/rc.tcpip
- ปิดใช้งาน aixmibd ในไฟล์ /etc/rc.tcpip
- ปิดใช้งาน muxatmd ในไฟล์ /etc/rc.tcpip
- พอร์ต NFS (2049) เป็นพอร์ตที่ใช้สิทธิพิเศษ
- เพิ่มเหตุการณ์ที่หายไปในไฟล์ /etc/security/audit/events
- ทำให้แน่ใจว่าส่วนการติดต่อลูปแบคกำลังทำงาน
- สร้างสำเนาสำหรับ /dev/console
- บังคับใช้สิทธิการเชื่อมต่อ X-server
- เปลี่ยนไดเร็กทอรี /var/docsearch เพื่อให้ไฟล์ทั้งหมด ที่ทุกคนสามารถอ่านได้
- เพิ่ม Object Data Manager (ODM) stanzas เพื่อตั้งค่าสิทธิคอนโซล
- ตั้งค่าสิทธิบน BSD-style ptys เป็น 000
- ปิดใช้งานไฟล์ .netrc
- เพิ่มการประมวลผลแพ็ตช์ไดเร็กทอรี

Graphical user interface:

ระบบที่ยึดตาม BAS/EAL4+ มี X Windows System เป็น graphical user interface

X Windows มี กลไกสำหรับการแสดงไคลเอ็นต์แบบกราฟิก เช่นนาฬิกา เครื่องคิดเลข และแอ็พพลิเคชันแบบกราฟิกอื่นๆ รวมถึงเทอร์มินัลเซสชันหลายเซสชัน โดยใช้คำสั่ง **aixterm** X Windows System เริ่มทำงานด้วยคำสั่ง **xinit** จากบรรทัดคำสั่งเริ่มต้นหลังจากผู้ใช้ล็อกอินที่คอนโซลของโฮสต์

ในการเริ่มทำงานเซสชัน X Windows ให้พิมพ์:

`xinit`

คำสั่งนี้เริ่มทำงานเซิร์ฟเวอร์ X Windows ที่มีกลไกการเข้าถึงโลคัลถูกเปิดใช้สำหรับผู้ร้องขอเท่านั้น ไคลเอ็นต์ X Windows ที่ถูกตั้งค่า UID เป็น root จะสามารถเข้าถึงเซิร์ฟเวอร์ X Windows ผ่านโดเมน UNIX โดยใช้การแทนที่ root ในการจำกัดการเข้าถึง ไคลเอ็นต์ X Windows ที่ถูกตั้งค่า UID เป็น ผู้ใช้อื่น หรือที่เริ่มทำงานโดยผู้ใช้อื่นจะไม่สามารถเข้าถึง เซิร์ฟเวอร์ X Windows การจำกัดนี้ ป้องกันผู้ใช้อื่นของโฮสต์มิให้เข้าถึงโดยไม่ได้รับอนุญาต ที่เซิร์ฟเวอร์ X Windows

การติดตั้งระบบ LAS/EAL4+:

RBAC ถูกเปิดใช้งานโดยอัตโนมัติเมื่อเลือกใช้อ็อพชันนี้

ในการตั้งค่าอ็อพชัน LAS/EAL4+ ระหว่างการติดตั้ง BOS ให้ทำดังนี้:

อ็อพชันการติดตั้งพร้อมใช้งาน โดยการพิมพ์ 3 เพื่อเปลี่ยน **Security Model** และพิมพ์ 4 เพื่อดูฟิลด์ **More Options** ในหน้าต่าง Installation and Settings อ็อพชันเหล่านี้แตกต่างกันไปขึ้นอยู่กับประเภทการติดตั้ง (การเขียนทับ, การคงแบบเดิม หรือการโอนย้าย) และอ็อพชันการรักษาความปลอดภัย สำหรับ LAS เมธอดการติดตั้งใหม่หรือเขียนทับโดยสมบูรณ์ เลือกการติดตั้ง การกำหนดค่า **LAS/EAL4+**

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ RBAC ดูที่ Role Based Access Control (RBAC)

LAS/EAL4+ การติดตั้งการกำหนดค่า (มีอยู่ใน Trusted AIXเท่านั้น):

ตัวเลือก **การติดตั้งการกำหนดค่า LAS/EAL4+** จะติดตั้ง Trusted AIX ในโหมดกำหนดค่า LAS/EAL4+ LASโหมดกำหนดค่า /EAL4+ มีระบบความปลอดภัยที่จำกัดเมื่อเทียบกับการติดตั้ง Trusted AIX

ถ้าคุณดำเนินการติดตั้งที่ไม่มีพรอมต์โดยใช้ไฟล์ `bosinst.data` ที่กำหนดเอง ฟิลด์ **INSTALL_TYPE** ต้องเป็นค่าว่าง และฟิลด์ **TRUSTED_AIX** ควร ถูกตั้งค่าเป็น `yes` และฟิลด์ต่อไปนี้ต้องถูกตั้งค่า ดังนี้:

```
control_flow:
 CONSOLE = ???
 PROMPT = yes
 INSTALL_TYPE =
 TRUSTED_AIX = yes
 INSTALL_METHOD = overwrite
 TCB = yes
 DESKTOP = NONE
 ENABLE_64BIT_KERNEL = yes
 CREATE_JFS2_FS = yes
 ALL_DEVICES_KERNELS = no
 FIREFOX_BUNDLE = no
 HTTP_SERVER_BUNDLE = no
 KERBEROS_5_BUNDLE = no
 SERVER_BUNDLE = no
 ALT_DISK_INSTALL_BUNDLE = no

locale:
 CULTURAL_CONVENTION = en_US or C
 MESSAGES = en_US or C
```

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ Trusted AIX ดูที่ Trusted AIX

สภาวะแวดล้อมการจัดการการติดตั้งเครือข่ายสำหรับ LAS/EAL4+>:

การติดตั้งของไคลเอ็นต์เทคโนโลยี LAS/EAL4+ สามารถดำเนินการได้โดยใช้สภาวะแวดล้อม Network Installation Management (NIM)

NIM มาสเตอร์ถูกตั้งค่าเพื่อจัดให้มีรีซอร์สที่จำเป็นสำหรับ การติดตั้งระดับ LAS/EAL4+ ที่เหมาะสมของ AIX 7.1 ไคลเอ็นต์ NIM อาจถูกติดตั้งโดยใช้รีซอร์สที่อยู่บน NIM มาสเตอร์ คุณสามารถติดตั้ง NIM โดยไม่มีพร้อมต์ของลูกค้าโดยตั้งค่า ฟิลด์ต่อไปนี้ในรีซอร์ส bosinst_data:

```
control_flow:
 CONSOLE = ???
 PROMPT = no
 INSTALL_TYPE =
 TRUSTED_AIX = yes
 INSTALL_METHOD = overwrite
 TCB = yes
 DESKTOP = NONE
 ENABLE_64BIT_KERNEL = yes
 CREATE_JFS2_FS = yes
 ALL_DEVICES_KERNELS = no
 FIREFOX_BUNDLE = no
 HTTP_SERVER_BUNDLE = no
 KERBEROS_5_BUNDLE = no
 SERVER_BUNDLE = no
 ALT_DISK_INSTALL_BUNDLE = no

locale:
 CULTURAL_CONVENTION = en_US or C
 MESSAGES = en_US or C
```

NIM มาสเตอร์ไม่สามารถตั้งค่าเป็นระบบ LAS/EAL4+ และไม่สามารถเชื่อมต่อ กับเน็ตเวิร์กที่มีระบบ LAS/EAL4+ อื่นเหมือนกัน เมื่อเริ่มต้นกำหนดค่า การติดตั้งจาก NIM ไคลเอ็นต์ อ็อพชันเมนู **Remain NIM client after install SMIT** ต้องถูกตั้งค่าเป็น No หลังจาก ติดตั้งไคลเอ็นต์ NIM เป็นระบบ LAS/EAL4+ แล้ว ไคลเอ็นต์ NIM ต้องถูกลบออกจากเน็ตเวิร์กของ NIM มาสเตอร์ และการติดตั้งซอฟต์แวร์และการอัพเดต เพิ่มเติมไม่สามารถใช้กับ NIM มาสเตอร์ได้

สถานการณ์ตัวอย่างคือมีสภาวะแวดล้อมเน็ตเวิร์กสองแบบ เน็ตเวิร์ก แบบแรกมี NIM มาสเตอร์และระบบที่ไม่ใช่ LAS/EAL4+ เน็ตเวิร์กแบบที่สอง มีเฉพาะระบบ LAS/EAL4+ เท่านั้น ดำเนิน การติดตั้ง NIM บนไคลเอ็นต์ NIM หลังการติดตั้งเสร็จเรียบร้อย ยกเลิกการเชื่อมต่อระบบ LAS/EAL4+ ที่ติดตั้งใหม่จากเน็ตเวิร์กของ NIM มาสเตอร์ และเชื่อมต่อระบบกับเน็ตเวิร์กที่ประเมินผล

ตัวอย่างที่สองประกอบด้วยด้วยหนึ่งเน็ตเวิร์ก NIM มาสเตอร์ไม่ได้ เชื่อมต่อกับเน็ตเวิร์กเมื่อระบบอื่นกำลังดำเนินงานในการติดตั้ง ที่ประเมินผล และระบบ LAS/EAL4+ ไม่เชื่อมต่อกับ เน็ตเวิร์กระหว่างการติดตั้ง NIM

BAS/EAL4+ และ LAS/EAL4+ สภาวะแวดล้อมฟิสิคัลระบบ:

ระบบ BAS/EAL4+ และ LAS/EAL4+ ระบุ ข้อกำหนดสภาวะแวดล้อมที่จะรัน

ข้อกำหนดมีดังนี้:

- การเข้าถึงแบบฟิสิคัลไปยังระบบต้องถูกจำกัดเพื่อให้เฉพาะ ผู้ดูแลระบบที่ได้รับอนุญาตเท่านั้นที่สามารถใช้คอนโซลระบบได้
- Service Processor ไม่เชื่อมต่อกับโมเด็ม
- การเข้าถึงแบบฟิสิคัลไปยังเทอร์มินัลต้องถูกจำกัดใช้ได้เฉพาะผู้ใช้ที่ได้รับอนุญาต
- ฟิสิคัลเน็ตเวิร์กมีความปลอดภัยต่อโปรแกรมการลอบฟังและการปลอมแปลง (ที่เรียกว่าโปรแกรมม้าโทรจัน) เมื่อทำการสื่อสารบน สายที่มีความปลอดภัย จำเป็นต้องมีมาตรการด้านความปลอดภัยเพิ่มขึ้น เช่น การเข้ารหัส
- ไม่อนุญาตให้มีการติดต่อกับระบบอื่นที่ไม่ใช่ระบบ AIX 7.1 BAS/EAL4+ หรือ LAS/EAL4+ หรือไม่อยู่ภายใต้ การควบคุมการจัดการที่เหมือนกัน
- เฉพาะ IPv4 จะถูกใช้เมื่อสื่อสารกับระบบ BAS/EAL4+ และ LAS/EAL4+ IPv6 อยู่ในการกำหนดค่าที่ประเมินแล้ว แต่จะรวมเฉพาะความสามารถในการทำงานของ IPv6 ที่ได้รับการสนับสนุนโดย IPv4 เท่านั้น
- ผู้ใช้ต้องไม่ได้รับอนุญาตให้เปลี่ยนแปลงเวลาระบบ
- ระบบในสภาวะแวดล้อม LPAR ไม่สามารถแบ่งใช้ PHBs

BAS/EAL4+ และ LAS/EAL4+ สภาวะองค์กรของระบบ:

ข้อกำหนดโพรซีเดอร์และองค์กรบางอย่างต้องตรงตาม ระบบ BAS/EAL4+ and LAS/EAL4+

ต้องตรงตามข้อกำหนดต่อไปนี้:
- ผู้ดูแลระบบต้องเป็นผู้ที่เชื่อถือได้และผ่านการอบรมอย่างดี
- เฉพาะผู้ใช้ที่ได้รับอนุญาตให้ทำงานกับข้อมูลบนระบบเท่านั้น ที่จะได้สิทธิ ID ผู้ใช้บนระบบ
- ผู้ใช้ต้องใช้รหัสผ่านที่มีคุณภาพดีเยี่ยม (ใช้แบบสุ่มเท่าที่เป็นได้ และ ไม่ควรใช้ร่วมกับผู้ใช้หรือองค์กร) สำหรับข้อมูล เกี่ยวกับการตั้งกฎรหัสผ่าน โปรดดูที่ “รหัสผ่าน” ในหน้า 72
- ผู้ใช้ต้องไม่เปิดเผยรหัสผ่านแก่ผู้อื่น
- ผู้ดูแลระบบต้องมีความรู้เพียงพอสำหรับการจัดการดูแลระบบ ที่ความปลอดภัยเป็นสิ่งสำคัญ
- ผู้ดูแลระบบต้องทำงานตามคำแนะนำที่ให้โดยเอกสารคู่มือระบบ
- ผู้ดูแลระบบต้องล็อกอินด้วย ID ส่วนบุคคลของตน และใช้คำสั่ง **su** เพื่อสลับไปเป็นโหมด superuser สำหรับการดูแลระบบ
- รหัสผ่านที่สร้างขึ้นสำหรับผู้ใช้ระบบโดยผู้ดูแลระบบต้องถูก ส่งถึงผู้รับอย่างปลอดภัย
- ผู้มีหน้าที่รับผิดชอบต่อระบบต้องสร้างและนำกระบวน ที่จำเป็นสำหรับการดำเนินงานระบบโดยปลอดภัยไปใช้
- ผู้ดูแลระบบต้องทำให้แน่ใจว่าการเข้าถึงรีซอร์สระบบที่ความปลอดภัยเป็น สิ่งสำคัญได้รับการปกป้องโดยการตั้งค่าบิต สิทธิและ ACLs ที่เหมาะสม
- ฟิสิคัลเน็ตเวิร์กต้องได้รับการอนุมัติโดยองค์กร เพื่อดำเนินการข้อมูลที่มีความอ่อนไหวมากที่สุดที่เก็บในระบบ
- กระบวนการบำรุงรักษาต้องรวมการวินิจฉัยระบบ สม่ำเสมอ
- ผู้ดูแลระบบต้องมีกระบวนการที่จะทำให้แน่ใจใน การดำเนินงานอย่างปลอดภัย และการกู้คืนหลังจากระบบล้มเหลว
- ตัวแปรสภาวะแวดล้อม *LIBPATH* ไม่ควรถูกเปลี่ยน เนื่องจากอาจส่งผลให้กระบวนการที่ไว้วางใจทำการโหลด ไลบรารีที่ไม่ไว้วางใจ
- การดักรับสัญญาณสายและติดตามซอฟต์แวร์ (tcpdump, trace) ต้องไม่ถูกใช้ บนระบบดำเนินการ
- โปรโตคอลแบบไม่ระบุชื่อเช่น HTTP ต้องใช้เฉพาะสำหรับข้อมูลที่เปิดเผยสู่สาธารณะเท่านั้น (เช่น เอกสารคู่มือออนไลน์)
- สามารถใช้ได้เฉพาะ NFS บน TCP เท่านั้น

- ต้องไม่ให้สิทธิในการเข้าถึงสื่อบันทึกแบบถอดออกได้แก่ผู้ใช้ไฟล์อุปกรณ์ ต้องได้รับการป้องกันโดยบิตสิทธิหรือ ACLs ที่เหมาะสม
- ผู้ดูแลระบบต้องไม่ใช้การแบ่งพาร์ติชันแบบไดนามิกเพื่อจัดสรรและ ยกเลิกการจัดสรรรีซอร์ส การตั้งค่าพาร์ติชันสามารถดำเนินการได้ต่อเมื่อ ไม่มีพาร์ติชันใดกำลังทำงานอยู่เท่านั้น

BAS/EAL4+ และ LAS/EAL4+ สภาวะการทำงานของระบบ:

ข้อกำหนดและโพรซีเดอร์การทำงานบางอย่างต้องตรง ตามระบบ BAS/EAL4+ และ LAS/EAL4+

ข้อกำหนดและโพรซีเดอร์ต่อไปนี้ต้องตรง:

- ถ้าใช้ Hardware Management Console (HMC) HMC อยู่ในสภาวะแวดล้อมที่ควบคุม เชิงกายภาพ
- บุคคลที่ได้รับอนุญาตเท่านั้นที่สามารถเข้าถึงสภาวะแวดล้อมการดำเนินการ และ HMC
- ถ้าใช้ HMC HMC สามารถใช้ได้ กับงานต่อไปนี้เท่านั้น:
  - เริ่มการตั้งค่าของพาร์ติชัน พาร์ติชันไม่สามารถ แอ็คทีฟระหว่างกระบวนการตั้งค่า
  - เริ่มทำงานของพาร์ติชัน ″hanging″ ต่อ
- HMC ต้อง ไม่ถูกใช้ในการดำเนินการทั้งหมดของระบบที่ตั้งค่า
- คุณลักษณะ ″call home″ ของระบบต้องถูกปิดใช้งาน
- รีโมตโมเด็มเข้าถึงระบบต้องถูกปิดใช้งาน
- ถ้า AIX รันในสภาวะแวดล้อมที่เปิดใช้งาน LPAR ผู้ดูแลระบบควรตรวจสอบกับเอกสารคู่มือ LPAR เพื่อดูข้อกำหนด เกี่ยวกับการดำเนินการ EAL4+ ของพาร์ติชันโลจิคัล
- คุณลักษณะการให้สิทธิเซอร์วิสต้องถูกปิดใช้งานบนพาร์ติชันโลจิคัล

การตั้งค่าระบบ BAS/EAL4+:

คุณสามารถตั้งค่าระบบ Base AIX Security (BAS) และ ระดับการประเมินการรับประกัน 4+ (EAL4+)

กลุ่ม **system, sys, adm, uucp, mail, security, cron, printq, audit** และ **shutdown** ถูกพิจารณาว่าเป็นกลุ่มด้านการดูแล เฉพาะผู้ใช้ที่ไว้วางใจเท่านั้นที่ควร เพิ่มในกลุ่มนี้

การดูแล:

ผู้ดูแลระบบต้องล็อกอินด้วยบัญชีผู้ใช้ส่วนบุคคลของตนเอง และใช้คำสั่ง **su** เพื่อเปลี่ยนเป็นผู้ใช้ root สำหรับการดูแลระบบ

เพื่อป้องกันการเดารหัสผ่านของบัญชีผู้ใช้ root ได้ อนุญาตให้ ผู้ดูแลระบบที่ได้รับอนุญาตเท่านั้นในสามารถใช้คำสั่ง **su** บนบัญชีผู้ใช้ root ได้ เพื่อให้แน่ใจ ให้ดำเนินการต่อไปนี้:

1. เพิ่มรายการใน **root** stanza ของไฟล์ `/etc/security/user` ดังนี้:

```
root:
  admin = true
  .
  .
  .
  sugroups = SUADMIN
```

2. กำหนดกลุ่มในไฟล์ `/etc/group` ที่มี เฉพาะ ID ผู้ใช้ของผู้ดูแลระบบที่ได้รับอนุญาตเท่านั้น ดังนี้:

```
system:!:0:root,paul
staff:!:1:invscout,julie
bin:!:2:root,bin
.
.
.
SUADMIN:!:13:paul
```

ผู้ดูแลระบบยังต้องปฏิบัติการกระบวนการต่อไปนี้:

- สร้างและประยุกต์ใช้โพรซีเดอร์เพื่อให้แน่ใจคอมโพเนนต์ ฮาร์ดแวร์, ซอฟต์แวร์ และเฟิร์มแวร์ที่ประกอบเป็นระบบแบบกระจาย จะถูกแจกจ่าย, ติดตั้ง และตั้งค่าในด้วยวิธีการที่ปลอดภัย
- ทำให้แน่ใจว่าระบบถูกตั้งค่าเพื่อที่มีเพียงผู้ดูแลระบบเท่านั้น ที่สามารถแนะนำซอฟต์แวร์ใหม่ที่ไว้วางใจได้ลงในระบบ
- นำโพรซีเดอร์ไปใช้เพื่อให้แน่ใจว่าผู้ใช้ลบหน้าจอก่อน ล็อกออฟออกจากอุปกรณ์ล็อกอินแบบซีเรียล (ตัวอย่าง เทอร์มินัล IBM 3151)

การตั้งค่าผู้ใช้และพอร์ต:

อ็อพชันการตั้งค่า AIX สำหรับผู้ใช้และพอร์ตต้องถูกตั้งค่าให้ตรงตามข้อกำหนด ของการประเมินผล ข้อกำหนดที่แท้จริงคือ TSF มี กลไกการคาดเดารหัสผ่านที่ถูกต้องที่ตรงตามคุณภาพเมตริกซ์ ความเป็นไปได้ของการคาดเดารหัสผ่านที่ถูกต้องซึ่งได้มาจาก ผู้จู่โจมในระหว่างระยะเวลาที่รหัสผ่านมีผลใช้ได้ โดยต้องไม่น้อยกว่า 2^-20

ไฟล์ /etc/security/user ที่แสดงในตัวอย่าง ต่อไปนี้ใช้รายการพจนานุกรม /usr/share/dict/words ไฟล์ /usr/share/dict/words ถูกเก็บใน ชุดไฟล์ bos.data คุณต้องติดตั้งชุดไฟล์ bos.data ก่อนทำการติดตั้งไฟล์ /etc/security/user ค่าที่แนะนำสำหรับไฟล์ /etc/security/user มีดังนี้:

```
ดีฟอลต์:
  admin = false
  login = true
  su = true
  daemon = true
  rlogin = true
  sugroups = ALL
  admgroups =
  ttys = ALL
  auth1 = SYSTEM
  auth2 = NONE
  tpath = nosak
  umask = 077
  expires = 0
  SYSTEM = "compat"
  logintimes =
  pwdwarntime = 5
  account_locked = false
  loginretries = 3
  histexpire   = 52
  histsize     = 20
  minage = 0
  maxage       = 8
  maxexpired   = 1
  minalpha = 2
  minother     = 2
```

```
  minlen       = 8
  mindiff = 4
  maxrepeats   = 2
  dictionlist = /usr/share/dict/words
  pwdchecks =
  dce_export = false

root:
  rlogin = false
  login = false
```

ค่ากำหนดดีฟอลต์ในไฟล์ /etc/security/user ไม่ควรถูกเขียนทับโดยค่ากำหนดที่เจาะจงสำหรับผู้ใช้เดียว

หมายเหตุ: การตั้งค่า login = false ใน root stanza เป็นการป้องกันการล็อกอิน root โดยตรง เฉพาะบัญชีผู้ใช้ที่มีสิทธิพิเศษ **su** สำหรับบัญชีผู้ใช้ root เท่านั้นที่จะสามารถล็อกอินเป็นบัญชีผู้ใช้ root ได้ ถ้า มีการเรียกใช้การโจมตี Denial of Service กับระบบที่ส่ง รหัสผ่านไม่ถูกต้องไปยังบัญชีผู้ใช้นั้น ระบบอาจล็อกบัญชีผู้ใช้ ทั้งหมด การโจมตีนี้อาจทำให้ผู้ใช้ (รวมถึงผู้ใช้ ที่มีหน้าที่ดูแลระบบ) ไม่สามารถล็อกอินเข้าสู่ระบบ เมื่อบัญชีผู้ใช้ของผู้ใช้ถูกล็อก ผู้ใช้จะไม่สามารถล็อกอินได้จนกว่าผู้ดูแลระบบตั้งค่าแอ็ตทริบิวต์ unsuccessful_login_count ของผู้ใช้ใหม่ ในไฟล์ /etc/security/lastlog ให้เป็นค่า ที่น้อยกว่าค่าของแอ็ตทริบิวต์ผู้ใช้ loginretries ถ้าบัญชีผู้ใช้ที่มีหน้าที่ดูแลระบบทั้งหมดถูกล็อก คุณอาจจำเป็นต้อง บูตระบบใหม่ให้เข้าสู่โหมดการบำรุงรักษา และรันคำสั่ง **chsec** สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการใช้คำสั่ง **chsec** ดูที่ “การควบคุมบัญชีผู้ใช้” ในหน้า 60

ค่าที่แนะนำสำหรับไฟล์ /etc/security/login.cfg มีดังนี้:

```
ดีฟอลต์:
  sak_enabled = false
  logintimes =
  logindisable = 4
  logininterval = 60
  loginreenable = 30
  logindelay = 5
```

รายการโปรแกรม setuid/setgid:

รายการแอ็พพลิเคชันที่ไว้วางใจถูกสร้างสำหรับระบบที่เปิดใช้ BAS-enabled AIX

บิต **suid/sgid** ถูกปิดทำงานสำหรับโปรแกรมที่ไม่ไว้วางใจที่มี root หรือกลุ่มที่ไว้วางใจเป็นเจ้าของ โปรแกรมบนระบบหลังการติดตั้ง BAS เท่านั้นที่เป็น **suid** และ เจ้าของโดย root หรือ **sgid** และเป็นเจ้าของโดยกลุ่มที่ไว้วางใจกลุ่มใดกลุ่มหนึ่งเหล่านี้ **system, sys, adm, uucp, mail, security, cron, printq, audit** และ **shutdown** เพิ่มเฉพาะ ผู้ใช้ที่ไว้วางใจเท่านั้นในกลุ่มเหล่านี้

รายการของแอ็พพลิเคชันที่ไว้วางใจสร้างโดยการพิจารณาแอ็พพลิเคชันทั้งหมด ที่จัดอยู่ในหมวดหมู่อย่างน้อยหนึ่งในหมวดหมู่ต่อไปนี้:

- บิต SUID root สำหรับแอ็พพลิเคชันที่เกี่ยวข้องถูกเปิดใช้งาน
- บิต SGID ของหนึ่งในกลุ่มที่ไว้วางใจถูกเปิดใช้งาน
- แอ็พพลิเคชันที่เข้าถึงฐานข้อมูลที่ไว้วางใจตาม เอกสารแนะนำผู้ดูแลระบบ

หมายเหตุ: บิต **setuid** สำหรับคำสั่ง **ipcs** ควร ถูกลบออกโดยผู้ดูแลระบบ ผู้ดูแลระบบควรรัน คำสั่ง **chmod u-s /usr/bin/ipcs** และ **chmod u-s /usr/bin/ipcs64**

การเปลี่ยนแปลงระบบไฟล์ตรวจสอบ:

RBAC เปิดใช้อัตโนมัติเมื่อเลือกตัวเลือกนี้

ระบบไฟล์ /audit เป็นระบบไฟล์jfs ต้องเปลี่ยนเป็นระบบไฟล์ jfs2 นอกจากนี้ ระบบ BAS ต้องมี คำสั่งเพิ่มเติม เพื่อเปลี่ยนแปลงระบบไฟล์ ให้ทำขั้นตอน ต่อไปนี้:

1. เปลี่ยนระบบไฟล์สำหรับระบบ BAS ป้อนคำสั่ง

   ```
   audit shutdown
   lsvg -l rootvg
   ```

   สำหรับระบบ LAS ไปที่ขั้นตอนที่ 3
2. ถ้าฟิลด์ TYPE มีเครื่องหมายคำถาม (?) ป้อนคำสั่ง:

   ```
   synclvodm -v rootvg
   ```
3. ลบระบบไฟล์ jfs และสร้างระบบไฟล์ jfs2 โดยป้อนคำสั่ง:

   ```
   umount/audit
   rmfs /audit
   crfs -v jfs2 -m /audit -g rootvg -A yes -p rw -a size=100M
   ```

การอัพเดต trusted signature database (TSD):

ส่วนนี้อธิบายขั้นตอนการอัพเดต TSD

การกำหนดค่า BAS/LAS จะเปลี่ยนบิตโหมดระบบ ข้อผิดพลาดความถูกต้อง TSD จะเกิดขึ้น

ในระหว่างรีบูทระบบ ให้เลือกตัวเลือก **Ignore All**

เพื่ออัพเดต TSD ป้อนคำสั่ง:

```
โหมด trustchk -u ทั้งหมด
```

การใช้ระบบ LAS:

ส่วนนี้อธิบายแนวทางการใช้ระบบ LAS

ตั้งค่าตัวเลือกรีบูตอัตโนมัติเป็น **false** หลังจาก ติดตั้งระบบเป็น isso โดยป้อน คำสั่ง:

```
chdev -l sys0 -a autorestart=false
```

ถ้า TSD ยังคงสร้างข้อผิดพลาด intlabel ต่อเนื่อง ลบข้อผิดพลาด โดยใช้ isso ที่มีสิทธิ **PV_ROOT** โดยป้อนคำสั่ง:

```
cp /etc/security/tsd/tsd.dat /etc/security/tsd/tsd.dat.org
trustchk -q /usr/sbin/format /usr/sbin/fdformat /usr/sbin/mount /usr/sbin/unmount \
/usr/sbin/umount /usr/sbin/tsm /usr/sbin/getty /usr/sbin/login /usr/sbin/mkvg \
/usr/sbin/extendvg /usr/bin/w /usr/bin/uptime >/tmp/list.dat
grep -p SLTL /tmp/list.dat |sed 's/SLTL/SHTL/' >/tmp/new.dat
trustchk -w -a -f /tmp/new.dat
trustchk -y ALL
```

ถ้าข้อความผิดปกติเชื่อมโยงกับระบบตรวจสอบ ที่แสดงในคอนโซล ด้วยสิทธิ isso รีสตาร์ทระบบตรวจสอบโดยป้อนคำสั่ง:

```
# audit shutdown
# audit start
```

หลังจากพยายามไม่สำเร็จครบสามครั้ง ล็อกอิน `isso/so` จะถูกบล็อกโดยเครือข่าย+ แต่ ผู้ดูแลระบบสามารถเข้าถึงบัญชีเหล่านี้ในคอนโซลท้องถิ่นได้

เอาต์พุตคำสั่งที่รันโดย `cron/at` ไม่ถูกส่งต่อไปยังสปูลเมลของผู้ใช้

ไดเร็กทอรีที่เขียนได้ ซึ่งมีช่วงฉลาก (เช่น: `/tmp`) ไม่มีพาร์ทิชัน เพื่อป้องกันความเป็นไปได้ของการไหลของข้อมูล ระหว่างฉลาก ผู้ดูแลระบบต้องกั้นพาร์ทิชันระหว่างไดเร็กทอรีนี้ทันที หลังกำหนดค่าครั้งแรก

อินเตอร์เฟสเครือข่าย:

ส่วนนี้อธิบายขั้นตอนการใช้อินเทอร์เฟซเครือข่าย

ใน Trusted AIX อินเทอร์เฟซเครือข่ายดีฟอลต์มีช่วงฉลาก ของ `minSL=impl_lo` and `maxSL=ts_all` สำหรับระบบ LAS/EAL4+ ไม่มีช่วงฉลาก กฎดีฟอลต์ถูกเปลี่ยนเป็น `impl_lo` โดยอัตโนมัติ เมื่อตัวเลือกติดตั้ง LAS/EAL4+ ถูกเลือก เมื่อต้องการเปลี่ยนแปลงกฎดีฟอลต์เป็น `isso` ใช้คำสั่ง **netrule**

ตัวอย่างเช่น:

```
/usr/sbin/netrule i+u default +impl_lo +impl_lo +impl_lo
```

การอัพเดต WPAR:

ส่วนนี้อธิบายขั้นตอนการสร้างพาร์ทิชันเวิร์กโหลด (WPAR) สำหรับ AIX ที่ตรงตาม EAL4+

สร้าง WPAR ในระบบ BAS และรัน คำสั่งต่อไปนี้ใน WPAR เพื่อสร้าง EAL4+ที่สอดคล้อง:

```
/usr/lib/security/CC_EVALify.sh
```

เมื่อคุณรัน `clogin` ในระบบ LAS ในครั้งแรก สคริปต์ firstboot จะรัน (ซึ่งรวม `CC_EVALify.sh`)

สคริปต์ firstboot เป็นสาเหตุให้ `clogin` รันนานกว่าปกติเมื่อ `clogin` เรียก TSM ให้ล็อกอิน แต่ WPAR ยังคงอยู่ในโหมดกำหนดค่า ล็อกอิน จึงถูกปฏิเสธ คุณต้องรอประมาณ 10 นาทีเพื่อให้ WPAR เสร็จสิ้นการกำหนดค่าก่อนพยายาม `clogin`อีกครั้ง สำหรับระบบ WPAR ที่สร้างใหม่ ตัวเลือกผู้ใช้ดีฟอลต์ต้องตั้งค่า ตามข้อกำหนดการประเมินผลที่รวม:

- `root` ในโหมด BAS
- `isso/sa/so` ในโหมด LAS

ผู้ใช้ `root` และ `isso` ไม่มี รหัสผ่านหรือใช้รหัสผ่านทั่วไป รหัสผ่านต้องอัพเดต ก่อนอนุญาตให้ผู้ใช้ที่ไม่ไว้วางใจในสภาวะโดยรวมหรือ WPAR ที่เกี่ยวข้อง

ข้อกำหนดรหัสผ่านการประเมินผล อยู่ที่ความเป็นไปได้ของการคาดเดารหัสผ่านที่ถูกต้องต้องมีอย่างน้อย ใน 1,000,000 และความเป็นไปได้ของการคาดเดารหัสผ่านอย่างถูกต้อง ในระหว่างพยายามทำซ้ำภายใน 1 นาทีต้องมีอย่างน้อยหนึ่งใน 100,000 เพื่อปฏิบัติตามข้อกำหนด พารามิเตอร์ผู้ใช้ในไฟล์ `/etc/security/user` ต้องเปลี่ยนเป็น:

```
default:
 maxage       = 8
 maxexpired   = 1
```

```
minother     = 2
minlen       = 8
maxrepeats   = 2
loginretries = 3
histexpire   = 52
histsize     = 20
```

การอัพเดต EFS:

ส่วนนี้อธิบายขั้นตอนการตั้งค่าแอ็ตทริบิวต์ความปลอดภัย ของ EFS ที่ถูกประเมินผลเป็นระบบไฟล์ cryptographic

การประเมินผลไม่รวมลักษณะของโหมด root guard เทียบกับการเข้าถึง `root` ทั้งหมด เมื่อเปิดใช้ EFS ตั้งค่าแอ็ตทริบิวต์ความปลอดภัยสำหรับคำสั่ง **efsmgr** และ **egskeymgr** โดยรันคำสั่ง:

```
setsecattr -c accessauths=ALLOW_ALL
innateprivs=PV_DEV_QUERY,PV_DEV_CONFIG,PV_AU_ADD,PV_DAC_R,PV_DAC_W,PV_DAC_X /usr/sbin/efsmgr

setsecattr -c accessauths=ALLOW_ALL
innateprivs=PV_DEV_QUERY,PV_DEV_CONFIG,PV_AU_ADD,PV_DAC_R,PV_DAC_W,PV_DAC_X /usr/sbin/efskeymgr

setkst -t cmd
```

การลบฮาร์ดดิสก์:

AIX อนุญาตให้ลบฮาร์ดดิสก์โดยใช้เซอร์วิส **Format media** ในแพ็คเกจวินิจฉัย AIX แพ็คเกวินิจฉัย มาพร้อมกับเอกสารจำนวนมากในคู่มือ *Diagnostic Information for Multiple Bus Systems* และคู่มือผู้ใช้ฮาร์ดแวร์ของคุณ

ในการลบฮาร์ดดิสก์ รันคำสั่งต่อไปนี้:

```
diag -T "format"
```

คำสั่งนี้เริ่มเซอร์วิส help **Format media** ในส่วน อินเตอร์เฟซแบบเมนู ถ้าได้รับพร้อมต์ ให้เลือกเทอร์มินัลของคุณ

แสดงรายการรีซอร์สที่เลือก เลือก อุปกรณ์ฮาร์ดดิสก์ที่คุณต้องการลบออกจากรายการนี้ และยอมรับการเปลี่ยนแปลง ของคุณตามคำแนะนำบนหน้าจอ

หลังจากยอมรับการเลือกของคุณ ให้เลือก **Erase Disk** จาก เมนู จากนั้นคุณจะถูกถามเพื่อให้ยืนยันการเลือกของคุณ เลือก **Yes**

จากนั้นคุณถูกถามว่าคุณต้องการ **Read data from drive** หรือ **Write patterns to drive** เลือก **Write patterns to drive**

จากนั้นคุณจะสามารถแก้ไขอ็อพชันการลบดิสก์ หลังจากคุณระบุอ็อพชันที่คุณต้องการแล้ว เลือก **Commit Your Changes** ดิสก์จะถูกลบ

หมายเหตุ: การลบนี้อาจ ใช้เวลานั้นกว่ากระบวนการนี้จะเสร็จสมบูรณ์

การจำกัดรีซอร์ส:

เมื่อคุณตั้งค่าข้อจำกัดรีซอร์สในไฟล์ `/etc/security/limits`, ให้แน่ใจว่า ข้อจำกัดสอดคล้องกับความต้องการของกระบวนการ บนระบบ

โดยเฉพาะอย่างยิ่ง, ไม่เคยตั้งค่าขนาด `stack` เป็น `unlimited` สแต็กไม่มีข้อจำกัดอาจแทนทับ เซ็กเมนต์อื่นของกระบวนการรัน ขนาด `stack_hard` ต้องถูกจำกัดไว้ด้วย

ระบบย่อยการตรวจสอบ:

มีหลายโพรซีเดอร์ที่ช่วยป้องกันระบบย่อยการตรวจสอบ

- ตั้งค่าระบบย่อยการตรวจสอบเพื่อบันทึกกิจกรรมที่เกี่ยวกับความปลอดภัยทั้งหมด ของผู้ใช้ เพื่อให้แน่ใจว่าพื้นที่ไฟล์ที่จำเป็นต้องใช้สำหรับการตรวจสอบมีอยู่ และไม่ถูกทำให้เสียหายโดยคอนซูมเมอร์อื่นของพื้นที่ระบบไฟล์ ให้ตั้งค่าระบบไฟล์เฉพาะสำหรับข้อมูลการตรวจสอบ
- ปกป้องเร็กคอร์ดการตรวจสอบ (เช่นหลักฐานการตรวจสอบ, ไฟล์ bin และข้อมูลอื่น ทั้งหมดที่เก็บใน `/audit`) จากผู้ใช้ที่มิใช่ root
- สำหรับระบบ BAS/EAL4+ การตรวจสอบโหมด **bin** ต้องถูกตั้งค่าเมื่อใช้ระบบย่อยการตรวจสอบ สำหรับข้อมูล เกี่ยวกับการตั้งค่าระบบย่อยการตรวจสอบ อ้างอิงที่ "การตั้งค่าการตรวจสอบ" ในหน้า 158
- ควรกำหนดพื้นที่ว่างในระบบไว้อย่างน้อย 20 เปอร์เซ็นต์ เพื่อใช้สำหรับหลักฐานการตรวจสอบ
- ถ้าเปิดใช้การตรวจสอบ พารามิเตอร์ *binmode* ใน start stanza ในไฟล์ `/etc/security/audit/config` ควรถูกตั้งค่าเป็น `panic` พารามิเตอร์ *freespace* ใน bin stanza ควรถูกตั้งค่าเป็นค่าต่ำสุดที่เท่ากับ 25 เปอร์เซ็นต์ของพื้นที่ดิสก์ที่ใช้เป็นหน่วยเก็บหลักฐานการตรวจสอบ พารามิเตอร์ *bytethreshold* และ *binsize* แต่ละค่าควรถูกตั้งค่าเป็น 65 536 ไบต์
- ทำสำเนาเร็กคอร์ดการตรวจสอบจากระบบไปยังหน่วยเก็บถาวรเพื่อเก็บถาวร

ไฟล์ที่ไม่แบ่งใช้ในระบบแบบกระจาย:

ไฟล์ต่อไปนี้ในไดเร็กทอรี `/etc/security` ไม่ถูกแบ่งใช้ในระบบแบบกระจาย แต่ยังคงเฉพาะโฮสต์:

/etc/security/failedlogin
ล็อกไฟล์สำหรับการล็อกอินที่ล้มเหลวต่อหนึ่งโฮสต์

/etc/security/lastlog
ข้อมูลต่อหนึ่งผู้ใช้เกี่ยวกับการล็อกอินที่สำเร็จและไม่สำเร็จ ล่าสุดบนโฮสต์นี้

/etc/security/login.cfg
คุณสมบัติการล็อกอินที่เฉพาะสำหรับโฮสต์ สำหรับพาธที่ไว้วางใจ เชลล์การล็อกอิน และ ข้อมูลที่เกี่ยวข้องกับการล็อกอินอื่น

/etc/security/portlog
ข้อมูลต่อพอร์ตสำหรับพอร์ตที่ถูกล็อกบนโฮสต์นี้

ไฟล์สำรองข้อมูลที่สร้างขึ้นโดยอัตโนมัติของไฟล์ที่แบ่งใช้ และไม่แบ่งใช้ไฟล์สำรองข้อมูลมีชื่อเหมือนกับไฟล์ต้นฉบับ แต่มีตัวพิมพ์เล็ก `o` นำหน้า

การใช้คุณลักษณะ DACinet สำหรับค่าควบคุมการเข้าใช้เน็ตเวิร์กที่ยึดตามผู้ใช้และตาม พอร์ต:

คุณลักษณะ DACinet สามารถใช้จำกัดการเข้าถึงพอร์ต TCP ของผู้ใช้

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ DACinet ดูที่ "ค่าควบคุมการเข้าใช้พอร์ต TCP ตามผู้ใช้โดยใช้ discretionary access control for internet ports" ในหน้า 231 ตัวอย่าง เมื่อใช้ DACinet เพื่อจำกัดการเข้าถึงพอร์ต TCP/25 ขาเข้า สำหรับ root เฉพาะกับคุณ

ลักษณะ DACinet เท่านั้น เฉพาะผู้ใช้ root จากโฮสต์ที่เป็นไปตาม BAS/EAL4+ เท่านั้นที่สามารถเข้าถึง พอร์ตนี้ สถานการณ์นี้จำกัดความเป็นไปได้ของผู้ใช้ ทั่วไปที่จะ spoofing อีเมลโดยการใช้ telnet เพื่อเชื่อมต่อพอร์ต TCP/25 ของเหยื่อ

ในการ เรียกทำงาน ACLs สำหรับการเชื่อมต่อ TCP ในตอนเปิดเครื่องใหม่ สคริปต์ /etc/rc.dacinet จะถูกรันจาก /etc/inittab สคริปต์จะอ่านนิยาม ในไฟล์ /etc/security/acl และโหลด ACLs เข้าสู่ เคอร์เนล พอร์ตที่ไม่ควรถูกป้องกันโดย ACLs จะแสดงในไฟล์ /etc/security/services ซึ่งใช้รูปแบบเดียวกับไฟล์ /etc/services

โดยการสมมติ ซับเน็ตของ 10.1.1.0/24 สำหรับระบบที่เชื่อมต่อทั้งหมด รายการ ACL เพื่อ จำกัดการเข้าถึงให้แก่ผู้ใช้ root เท่านั้นสำหรับ X (TCP/6000) ในไฟล์ /etc/security/acl จะเป็นดังนี้:

```
6000    10.1.1.0/24 u:root
```

การติดตั้งซอฟต์แวร์เพิ่มเติมบนระบบที่เป็นไปตาม BAS/EAL4+:

ผู้ดูแลระบบสามารถติดตั้งซอฟต์แวร์เพิ่มเติมบนระบบที่เป็นไปตาม BAS/EAL4+ ถ้าซอฟต์แวร์ ไม่ถูกรันโดยผู้ใช้ root หรือด้วยสิทธิพิเศษของผู้ใช้ root ค่านี้จะไม่ทำให้การยึดตาม BAS/EAL4+ เป็นโมฆะ ตัวอย่างทั่วไปประกอบด้วยออฟฟิศแอ็พพลิเคชันที่รันโดยผู้ใช้ทั่วไป เท่านั้น และไม่มีคอมโพเนนต์ SUID

นอกจากนั้น ซอฟต์แวร์ที่ติดตั้งอยู่ที่ทำงานด้วยสิทธิพิเศษของผู้ใช้ root ทำให้การยึดตาม BAS/EAL4+ เป็นโมฆะ ตัวอย่างนี้หมายความว่าไดรเวอร์สำหรับ JFS ที่เก่ากว่าไม่ควรถูกติดตั้ง ขณะกำลังทำงานในโหมด แอ็พพลิเคชันที่ได้รับสิทธิ ใดสิทธิหนึ่งหรือหลายสิทธิผ่าน /etc/security/privcmds ไม่เป็นที่ยอมรับ daemons อื่นที่รันเป็น root (ตัวอย่าง SNMP daemon) ยังทำให้การยึดตาม BAS/EAL4+ เป็นโมฆะ ระบบที่เปิดใช้งาน BAS/EAL4+ ไม่สามารถอัพเกรด (โดยทั่วไป)

ระบบที่เป็นไปตาม BAS/EAL4+ ไม่ค่อยถูกใช้ในการตั้งค่าที่ประเมินผล โดยเฉพาะในสภาวะแวดล้อมการพาณิชย์โดยทั่วไป จำเป็นต้องมีเซอร์วิสเพิ่มเติม เพื่อให้ระบบการทำงานจริงยึดตาม ระบบที่ประเมินผล แต่ไม่เป็นไปตามข้อกำหนดที่แท้จริงของระบบที่ประเมินผล

NSF v4 Access Control Lists และนโยบายเนื้อหา:

NFS v4 Access Control List (ACL) ประกอบด้วยฟิลด์ **Type, Mask** และ **Flags**

ต่อไปนี้คือรายละเอียดของฟิลด์เหล่านี้:

- ฟิลด์ **Type** มีค่าใดค่าหนึ่ง ต่อไปนี้:
  - ALLOW – ให้สิทธิประธาน ที่ระบุในฟิลด์ **Who** สิทธิที่ระบุในฟิลด์ **Mask**
  - DENY – ปฏิเสธประธาน ที่ระบุในฟิลด์ **Who** สิทธิที่ระบุในฟิลด์ **Mask**
- ฟิลด์ Mask มีอย่างน้อยหนึ่งค่าของค่าสิทธิ โดยละเอียดต่อไปนี้:
  - READ_DATA / LIST_DIRECTORY – อ่านข้อมูลจาก อ็อบเจ็กต์ที่ไม่ใช่ไดเร็กทอรี หรือแสดงรายการอ็อบเจ็กต์ในไดเร็กทอรี
  - WRITE_DATA / ADD_FILE – เขียนข้อมูลลงในอ็อบเจ็กต์ที่ไม่ใช่ ไดเร็กทอรี หรือเพิ่มอ็อบเจ็กต์ที่ไม่ใช่ไดเร็กทอรีลงในไดเร็กทอรี
  - APPEND_DATA / ADD_SUBDIRECTORY – ต่อท้ายข้อมูล ลงในอ็อบเจ็กต์ที่ไม่ใช่ไดเร็กทอรี หรือเพิ่มไดเร็กทอรีย่อยในไดเร็กทอรี
  - READ_NAMED_ATTRS – อ่านแอ็ตทริบิวต์ที่มีชื่อ ของอ็อบเจ็กต์
  - WRITE_NAMED_ATTRS – เขียนแอ็ตทริบิวต์ที่มีชื่อ ของอ็อบเจ็กต์

  - EXECUTE – ทำงานไฟล์ หรือสำรวจ/ค้นหา ไดเร็กทอรี
  - DELETE_CHILD – ลบไฟล์หรือไดเร็กทอรีภายใน ไดเร็กทอรี
  - READ_ATTRIBUTES – อ่านแอ็ตทริบิวต์ระดับต้น (ไม่ใช่ ACL) ของไฟล์
  - WRITE_ATTRIBUTES – เปลี่ยนเวลาที่เชื่อมโยง กับไฟล์หรือไดเร็กทอรี
  - DELETE – ลบไฟล์หรือไดเร็กทอรี
  - READ_ACL – อ่าน ACL
  - WRITE_ACL – เขียน ACL
  - WRITE_OWNER – เปลี่ยนเจ้าของและกลุ่ม
  - SYNCHRONIZE – ซิงโครไนซ์การเข้าถึง (มีอยู่เพื่อ ความเข้ากันได้กับไคลเอ็นต์ NFS v4 อื่น แต่ไม่มีฟังก์ชันที่ถูกนำไปใช้)
- ฟิลด์ **Flags** – ฟิลด์นี้กำหนดการสืบทอด ความสามารถของ ACLs ไดเร็กทอรีและบ่งชี้ว่าฟิลด์ **Who** มีกลุ่มหรือไม่ ฟิลด์นี้ศูนย์หรือมากกว่าศูนย์แฟล็กของแฟล็ก ต่อไปนี้:
  - **FILE_INHERIT** – ระบุว่าในไดเร็กทอรีนี้ อ็อบเจ็กต์ ที่ไม่ใช่ไดเร็กทอรีที่ถูกสร้างใหม่จะสืบทอดค่าของรายการนี้
  - **DIRECTORY_INHERIT** – ระบุว่าในไดเร็กทอรีนี้ ไดเร็กทอรีย่อยที่สร้างใหม่จะสืบทอดค่าของรายการนี้
  - **NO_PROPAGATE_INHERIT** – ระบุว่าในไดเร็กทอรีนี้ ไดเร็กทอรีย่อยที่สร้างใหม่จะสืบทอดค่าของรายการนี้ แต่ไดเร็กทอรีย่อยเหล่านี้ ไม่ส่งค่ารายการนี้ไปยังไดเร็กทอรีย่อยที่สร้างใหม่ของตน
  - **INHERIT_ONLY** – ระบุว่ารายการนี้ไม่มีผลใช้ กับไดเร็กทอรีนี้ มีเพียงอ็อบเจ็กต์ที่สร้างใหม่เท่านั้นท่จะสืบทอด รายการนี้
  - **IDENTIFIER_GROUP** – ระบุว่าฟิลด์ **Who** แทนกลุ่ม มิฉะนั้นฟิลด์ **Who** จะแทนผู้ใช้หรือค่า **Who** พิเศษ
- ฟิลด์ **Who** – ฟิลด์นี้มีค่าใดค่าหนึ่ง ต่อไปนี้:
  - User – ระบุผู้ใช้ที่รายการนี้ นำใช้
  - Group – ระบุกลุ่มที่รายการนี้ นำใช้
  - Special – แอ็ตทริบิวต์นี้สามารถเป็นค่าใดค่าหนึ่ง ต่อไปนี้:
    - OWNER@ – ระบุว่ารายการนี้นำใช้กับ เจ้าของอ็อบเจ็กต์
    - GROUP@ – ระบุว่ารายการนี้นำใช้กับ กลุ่มที่เป็นเจ้าของอ็อบเจ็กต์
    - EVERYONE@ – ระบุว่ารายการนี้นำใช้ กับผู้ใช้ทุกคนในระบบประกอบด้วยเจ้าของและกลุ่ม

ถ้า ACL ว่าง เฉพาะประธานที่มี UID ที่มีผลเป็น 0 เท่านั้นที่สามารถเข้าถึงอ็อบเจ็กต์ เจ้าของอ็อบเจ็กต์มีความหมายว่ามี ค่า mask ต่อไปนี้โดยไม่คำนึงว่า ACL อาจมีหรืออาจไม่มี:

- READ_ACL
- WRITE_ACL
- READ_ATTRIBUTES
- WRITE_ATTRIBUTES

ค่า APPEND_DATA ถูกนำไปใช้เป็น WRITE_DATA ไม่มีความแตกต่างด้านฟังก์ชันการทำงานระหว่างค่า WRITE_DATA และค่า APPEND_DATA ทั้งสองค่าต้องถูกตั้ง หรือไม่ถูกตั้งค่าใน unison

ความเป็นเจ้าของอ็อบเจ็กต์สามารถแก้ไขได้โดยการใช้ค่า WRITE_OWNER เมื่อเจ้าของหรือกลุ่มเปลี่ยนแปลง บิต **setuid** ต้องถูกปิด แฟล็กการสืบทอดมีความหมายใน ACL ของไดเร็กทอรีเท่านั้น และนำใช้กับอ็อบเจ็กต์ที่สร้างในไดเร็กทอรีหลังจาก แฟล็กการสืบทอดถูกตั้งค่าแล้วเท่านั้น (ตัวอย่าง อ็อบเจ็กต์ที่มีอยู่แล้ว ไม่ได้รับผลจากการเปลี่ยนแปลงการสืบทอดที่กระทำกับ ACL ของไดเร็กทอรี พาเรนต์) รายการใน NFS v4 ACL ขึ้นอยู่กับลำดับ ในการพิจารณา ถ้าการเข้าถึงที่ร้องขอได้รับอนุญาตหรือไม่ แต่ละรายการจะถูกประมวลผลตามลำดับ เฉพาะรายการที่มีค่าต่อไปนี้เท่านั้นที่ได้รับการพิจารณา:

- ฟิลด์ **Who** ที่ตรงกับ UID ที่มีผล
- ผู้ใช้ที่ระบุในรายการหรือ GID ที่มีผล
- กลุ่มที่ระบุในรายการของเรื่อง

แต่ละรายการถูกประมวลผลจนทุกบิตของการเข้าถึงของ ผู้ร้องขอเป็น ALLOWED หลังจากประเภทการเข้าถึงเปลี่ยนเป็น ALLOWED โดย รายการ รายการจะไม่ถูกพิจารณาในการประมวลผลรายการภายหลังอีกต่อไป ถ้ารายการ DENY ถูกพบโดยที่การเข้าถึงของผู้ร้องขอสำหรับ ค่า mask นั้นจำเป็นต้องใช้และยังไม่ได้พิจารณา การร้องขอถูกปฏิเสธ ถ้า การประเมินผลถึงจุดท้ายของ ACL การร้องขอถูกปฏิเสธ

ขนาด ACL ที่สนับสนุนสูงสุดคือ 64 KB แต่ละรายการใน ACL มี ความยาวผันแปรได้และ 64 KB เป็นข้อจำกัดของหนึ่งรายการเท่านั้น

ค่า WRITE OWNER:

นโยบาย NFS v4 จัดให้มีการควบคุมว่าผู้ใดที่จะสามารถอ่านและ เขียนแอ็ตทริบิวต์ของอ็อบเจ็กต์ได้

เรื่องที่มี UID ที่มีผลค่าเป็น 0 สามารถแทนที่นโยบาย NFS v4 ได้เสมอ เจ้าของอ็อบเจ็กต์สามารถอนุญาตให้ผู้อื่นอ่านและเขียนแอ็ตทริบิวต์ ของแอ็ตทริบิวต์ได้โดยใช้แอ็ตทริบิวต์ READ_ATTRIBUTES, WRITE_ATTRIBUTES , READ_NAMED_ATTRS และ WRITE_NAME_ATTRS ของ ACL mask เจ้าของสามารถควบคุมว่าใครที่จะสามารถอ่านและเขียน ACL โดยใช้ค่า READ_ACL และ WRITE_ACL ของ ACL mask เจ้าของอ็อบเจ็กต์มีการเข้าถึงแบบ READ_ATTRIBUTES, WRITE_ATTRIBUTES, READ_ACL และ WRITE_ACL เสมอ เจ้าของอ็อบเจ็กต์ยังสามารถอนุญาต ให้ผู้อื่นเปลี่ยนเจ้าของและกลุ่มของอ็อบเจ็กต์ได้โดยใช้แอ็ตทริบิวต์ WRITE_OWNER โดยค่าดีฟอลต์ เจ้าของอ็อบเจ็กต์ไม่สามารถเปลี่ยนเจ้าของหรือกลุ่มของอ็อบเจ็กต์ แต่เจ้าของอ็อบเจ็กต์สามารถเพิ่มรายการ WRITE_OWNER ใน ACL เพื่อระบุตนเอง หรืออ็อบเจ็กต์สามารถสืบทอดรายการ ACL ที่ระบุรายการ WRITE_OWNER ด้วยค่า **Who** ของ OWNER@ เมื่อเจ้าของหรือกลุ่มเปลี่ยนแปลง บิต **setuid** ต้องถูก ปิด

ต่อไปนี้เป็นข้อยกเว้นบางอย่างของกฎ:

- ถ้าอ็อบเจ็กต์เป็นเจ้าของโดย UID 0 มีเพียง UID 0 เท่านั้นที่สามารถเปลี่ยนเจ้าของ แต่กลุ่มยังคงถูกเปลี่ยนโดยประธานที่มีแอ็ตทริบิวต์ WRITE_OWNER
- โดยถือว่าอ็อบเจ็กต์มีแอ็ตทริบิวต์ WRITE_OWNER สำหรับอ็อบเจ็กต์ ในเวอร์ชันของ AIX 5.3 ก่อนหน้า Technology Level 5300-05 ถ้าอ็อบเจ็กต์มีเจ้าของที่มิใช่ UID 0 เจ้าของสามารถ เปลี่ยนเป็นผู้ใช้ UID 0 ที่ไม่ใช่อื่นอีก ใน AIX ที่มี 5300-05 และภายหลัง ถ้าอ็อบเจ็กต์มีเจ้าของที่ไม่ใช่ UID 0 เจ้าของสามารถเปลี่ยน EUID ของเป้าหมายที่พยายามเปลี่ยนเจ้าของ
- กลุ่มสามารถเปลี่ยนเป็นกลุ่มใดๆ ในชุดกลุ่มที่เกิดขึ้นพร้อมกัน ของเป้าหมายที่มีข้อยกเว้นว่าไม่สามารถเปลี่ยนเป็น GID 0 หรือ GID 7 (ระบบหรือการรักษาความปลอดภัย) แม้ว่าสองกลุ่มนี้จะอยู่ใน ชุดกลุ่มที่เกิดขึ้นพร้อมกันของเป้าหมาย

ฐานข้อมูลการจัดการที่ยึดตาม LDAP และยึดตามไฟล์ที่สนับสนุน:

การประเมินผลไม่สนับสนุนการใช้ฐานข้อมูลการจัดการ NFS วิธีการพิสูจน์ตัวตนเช่น DCE และ NIS ไม่ได้รับการสนับสนุน

การประเมินผลสนับสนุนวิธีการต่อไปนี้เท่านั้น:

- การพิสูจน์ตัวตนที่ยึดตามไฟล์ (ดีฟอลต์)
- การพิสูจน์ตัวตนที่อิงตาม UNIX-style LDAP (ใช้เซิร์ฟเวอร์ LDAP IBM Tivoli Directory Server v 6.0)

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการพิสูจน์ตัวตนที่ยึดตามไฟล์ ดูที่ User Authentication

การพิสูจน์ตัวตน LDAP:

I&A ที่ยึดตาม LDAP ถูกตั้งค่าในโหมดการพิสูจน์ตัวตน "UNIX-type" ในโหมดนี้ ข้อมูลการจัดการ (ได้แก่ชื่อผู้ใช้ IDs และรหัสผ่าน) ถูกเก็บใน LDAP โดยจำกัดการเข้าถึงข้อมูล ได้เฉพาะผู้ดูแลระบบ LDAP

เมื่อผู้ใช้ล็อกอินเข้าสู่ระบบ ระบบจะเชื่อมกับเซิร์ฟเวอร์ LDAP โดยใช้บัญชีผู้ใช้ผู้ดูแลระบบ LDAP บนการเชื่อมต่อ SSL เรียกข้อมูลที่จำเป็นสำหรับผู้ใช้ออกมา (รวมรหัสผ่าน) จาก LDAP จากนั้นดำเนินการพิสูจน์ตัวตนโดยใช้ข้อมูลที่เรียกออกมา จาก LDAP ระบบคงฐานข้อมูลการจัดการไว้บนเซิร์ฟเวอร์ LDAP โฮสต์ที่เหลืออิมพอร์ตข้อมูลการจัดการจาก เซิร์ฟเวอร์ LDAP เดียวกันทางกลไกเดียวกันกับที่อธิบายก่อนหน้านี้ ระบบคงฐานข้อมูลการจัดการที่สอดคล้องกันโดยทำการ เปลี่ยนแปลงการจัดการทั้งหมดบนเซิร์ฟเวอร์ LDAP ที่กำหนดไว้ ID ผู้ใช้ บนคอมพิวเตอร์เครื่องใดจะอ้างถึงข้อมูลเฉพาะเดียวกันบนคอมพิวเตอร์เครื่องอื่นทั้งหมด นอกจากนั้น การตั้งค่ารหัสผ่าน การแม็พ name-to-UID และ ข้อมูลอื่นๆ ต้องเหมือนกันบนโฮสต์ทั้งหมดในระบบแบบกระจาย

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการตั้งค่าการพิสูจน์ตัวตน LDAP ดูที่ Light Directory Access Protocol สำหรับข้อมูลเพิ่มเติมในการตั้งค่า SSL บน LDAP ดูที่ Setting up SSL on the LDAP server และ Setting up SSL on the LDAP client

เซิร์ฟเวอร์ LDAP:

คำสั่ง **mksecldap -s** ตั้งค่าระบบ AIX เป็นเซิร์ฟเวอร์ LDAP สำหรับ การพิสูจน์ความถูกต้องด้านความปลอดภัยและการจัดการข้อมูล

ดำเนินงานต่อไปนี้:

- ใช้ RFC2307AIX schema ด้วยอ็อพชัน **-S**
- ตั้งค่าเซิร์ฟเวอร์เพื่อใช้ Secure Sockets Layer (SSL) โดยใช้อ็อพชัน **-k** แอ็คชันนี้ต้องการให้ติดตั้งชุดไฟล์ **GSKit V8** และชุดไฟล์ **idsldap.clt_max_crypto32bit63.rte** สำหรับระบบแบบ 32 บิต หรือชุดไฟล์ **idsldap.clt_max_crypto64bit63.rte** สำหรับระบบ 64 บิต ใช้ยูทิลิตี ikeyman เพื่อสร้างคู่ของคีย์ สำหรับไดเร็กทอรีเซิร์ฟเวอร์

อ็อพชันผู้ใช้ LDAP ต้องถูกตั้งค่าให้ตรงกับข้อกำหนด ของการประเมินผล RFC2370AIX schema กำหนดแอ็ตทริบิวต์ผู้ใช้ใช้ค่าเดิมที่อธิบายไว้ในการตั้งค่าระบบ BAS/EAL4+ ผู้ดูแลระบบ Tivoli Directory Server ไม่ถูกบังคับใช้เพื่อเปลี่ยนรหัสผ่านเป็นช่วงเวลา (ตัวอย่างเช่น, ไม่มีค่า **MaxAge** สำหรับรหัสผ่านการดูแลระบบ) เนื่องจาก เหตุผลนี้ รหัสผ่านการจัดการ LDAP จึงต้องเปลี่ยนบ่อยเช่นเดียวกับ ผู้ใช้ AIX (**MaxAge** = 8 (สัปดาห์))

ใน Tivoli Directory Server 6.3, การจัดการความล้มเหลวในการพิสูจน์ตัวตน ไม่ใช้กับ Directory Administrator หรือกับสมาชิกของกลุ่ม การจัดการ กฎการประกอบรหัสผ่านไม่มีผลกับบัญชีผู้ใช้ การจัดการเช่นกัน กฎเหล่านี้จำเป็นต้องถูกบังคับหาก Tivoli Directory Server 6.3 ถูกใช้

ถ้าผู้ดูแลระบบไม่ได้ใช้ฐานข้อมูล LDAP ส่วนหลังทั่วไป สำหรับการจัดการกับผู้ใช้, ผู้ดูแลระบบต้องมั่นใจว่า ฐานข้อมูลที่มีหนังสือรับรองได้ถูกจัดการไว้ระหว่างส่วนของระบบ TCP Offload Engine (TOE) ที่แตกต่างกันของหนึ่งเครือข่าย ตัวอย่างมีดังต่อไปนี้:

- `/etc/group`

- /etc/passwd
- /etc/security/.ids
- /etc/security/.profile
- /etc/security/environ
- /etc/security/group
- /etc/security/limits
- /etc/security/passwd
- /etc/security/user

ข้อมูลที่เกี่ยวข้อง:

ข้อมูล IBM Tivoli Directory Server เกี่ยวกับแพ็กเกจ, ชุดไฟล์, และสิ่งที่จำเป็นต้องมี

ไคลเอ็นต์ LDAP:

คำสั่ง **mksecldap -c** ตั้งค่าระบบ AIX เป็นไคลเอ็นต์ LDAP สำหรับ การพิสูจน์ความถูกต้องด้านความปลอดภัยและการจัดการข้อมูล

ดำเนินงานต่อไปนี้:

- การใช้คำสั่ง **mksecldap -c** ระบุ **unix_auth** สำหรับ **authType** ด้วยอ็อพชัน **-A**
- ตั้งค่าไคลเอ็นต์ให้ใช้ SSL โดยใช้อ็อพชัน **-k** ในคำสั่ง **mksecldap -c** การระบุคีย์ SSL ไคลเอ็นต์ที่จำเป็นสำหรับการติดตั้งชุดไฟล์ **GSKit** และชุดไฟล์ **ldap.max_crypto_client** ใช้ยูทิลิตี **gsk7ikm** เพื่อสร้างคู่ของคีย์ สำหรับไดเร็กทอรีเซิร์ฟเวอร์

NFS v4 Client/Server และ Kerberos:

สภาวะแวดล้อม NFS v4 Client/Server ประกอบด้วย LDAP สำหรับการจัดการ ข้อมูลการพิสูจน์ตัวตนและ Kerberos สำหรับการสร้างแชนเนลที่ไว้วางใจระหว่าง ไคลเอ็นต์ NFS v4 กับเซิร์ฟเวอร์ คอนฟิกูเรชันที่มีการประเมินค่า สนับสนุน NAS v1.4 สำหรับ Kerberos และ IBM Tivoli Directory Server v6.0 (เซิร์ฟเวอร์ LDAP) สำหรับฐานข้อมูลผู้ใช้

NAS v1.4 (Kerberos Version 5 Server) ต้องถูกตั้งค่าให้ใช้ LDAP สำหรับ ฐานข้อมูลของตน ตั๋ว Kerberos ที่ให้สิทธิก่อนหน้านี้โดยเซิร์ฟเวอร์ Kerberos จะ ใช้ได้จนกว่าจะหมดอายุ

เมื่อคุณกำลังใช้การพิสูจน์ตัวตน Kerberos credential ที่ใช้ใน remote procedure calls ที่เริ่มต้นโดยผู้ใช้จะถูกเชื่อมโยงเข้ากับตั๋ว Kerberos ปัจจุบันที่ถือครองโดยผู้ใช้ และไม่ถูกควบคุมโดย UID จริง หรือที่มีผล ของกระบวนการ เมื่อคุณกำลังเข้าถึงระบบไฟล์รีโมต NFS โดยใช้การพิสูจน์ตัวตน Kerberos ขณะรันโปรแกรม **setuid** UID ที่เห็นที่เซิร์ฟเวอร์ จะยึดตาม Kerberos identity ไม่ใช่ UID ที่เป็นเจ้าของโปรแกรม **setuid** ที่กำลังรัน

การตั้งค่าที่ประเมินจะเกี่ยวกับการตั้งค่า NFS เพื่อใช้การรักษาความปลอดภัย RPCSEC-GSS สำหรับข้อมูลเพิ่มเติม ดูที่ Network File System, Configuring an NFS server และ Configuring an NFS client เมื่อทำการตั้งค่าเซิร์ฟเวอร์ เลือกการพิสูจน์ตัวตน Kerberos และเปิดใช้การรักษาความปลอดภัยที่ปรับปรุงบนเซิร์ฟเวอร์ คุณสามารถเปิดใช้ได้ทาง SMIT โดยใช้คำสั่ง **chnfs** คำสั่ง **chnfs** มีอ็อพชันที่จะเปิดใช้งานการรักษาความปลอดภัย RPCSEC_GSS เมื่อคุณตั้งค่าไคลเอ็นต์ ปฏิบัติตามคำแนะนำในการใช้ Kerberos ใน Configuring an NFS client ดูที่ Setting up a network for RPCSEC-GSS เพื่อดูคำแนะนำ ในการตั้งค่าเซิร์ฟเวอร์ข้อมูล Kerberos ด้วยการเข้ารหัส DES3 สำหรับการรักษาความปลอดภัย การตั้งค่า ที่ประเมินให้การสนับสนุนเฉพาะการเข้ารหัส des3 เท่านั้น

กฎรหัสผ่าน:

การตั้งค่าที่ประเมินควรมีค่าเหล่านี้สำหรับกฎรหัสผ่าน เมื่อคุณกำลังใช้เซิร์ฟเวอร์ Kerberos ที่มี LDAP เป็น ฐานข้อมูล

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับกฎรหัสผ่าน ดูที่ "Chapter 9. Managing Network Authentication Service passwords" ใน *IBM Network Authentication Service Version 1.4 for AIX, Linux and Solaris Administrator's and User's Guide*

รายการของค่ามีดังนี้:

mindiff 4

maxrepeats
2

minalpha
2

minother
2

minlen 8

minage 0

histsize 10

เพื่อให้ไคลเอ็นต์ AIX NFS v4 และเซิร์ฟเวอร์ AIX NFS v4 สามารถ สื่อสารได้อย่างปลอดภัยโดยใช้ DES3 enctypes เท่านั้น ให้สร้างหลักการเซิร์ฟเวอร์ "nfs/hostname" ที่มี DES3 enctype (เช่น des3-cbc-sha1) ร่วมกับรายการที่สอดคล้องในไฟล์ `keytab` (โดยใช้ส่วนการติดต่อ **kadmin**) และมี DES3 (เช่น **des3-cbc-sha1**) เป็นรายการแรกในส่วน **default_tgs_enctypes** ของ ไฟล์ `/etc/krb5/krb5.conf` บนเครื่องไคลเอ็นต์ NFS v4

Virtual I/O Server:

Virtual I/O Server (VIOS) อยู่ในพาร์ติชัน LPAR ต่างหาก และจัดให้มีค่าควบคุมการเข้าใช้เบื้องต้นระหว่างไดรเวอร์อุปกรณ์ VIOS SCSI ที่ทำหน้าที่แทนพาร์ติชัน LPAR และวอลุ่มโลจิคัลบน SCSI และ ฟิสิคัลวอลุ่มผ่านการแม็พ

พาร์ติชัน LPAR (ผ่านไดรเวอร์อุปกรณ์ VIOS SCSI) อาจถูกแม็พกับ 0 หรือมากกว่า 0 วอลุ่มโลจิคัลและฟิสิคัล แต่หนึ่งวอลุ่มสามารถแม็พได้กับ หนึ่งพาร์ติชัน LPAR เท่านั้น การแม็พนี้จำกัดพาร์ติชัน LPAR กับวอลุ่มที่กำหนด เท่านั้น VIOS ยังควบคุมการแม็พของไดรเวอร์อุปกรณ์อะแด็ปเตอร์อีเทอร์เน็ต VIOS กับไดรเวอร์อุปกรณ์อีเทอร์เน็ต VIOS ที่ทำหน้าที่แทนกลุ่มของพาร์ติชัน LPAR ที่แบ่งใช้เน็ตเวิร์กเสมือน ในการตั้งค่าที่ประเมิน อนุญาตให้ทำ การแม็พแบบหนึ่ง-ต่อ-หนึ่งสำหรับไดรเวอร์อุปกรณ์อะแด็ปเตอร์อีเทอร์เน็ตกับ ไดรเวอร์อุปกรณ์อะแด็ปเตอร์อีเทอร์เน็ตที่ทำหน้าที่แทนกลุ่มของพาร์ติชัน LPAR เท่านั้น การแม็พ หนึ่ง-ต่อ-หนึ่งถูกตั้งค่าโดยผู้ดูแลระบบและบังคับใช้โดยไดรเวอร์อุปกรณ์ อีกทั้ง แพ็กเก็ตอีเทอร์เน็ตต้องไม่ถูกแท็กด้วยแท็ก VLAN ในการตั้งค่า ที่ประเมิน กลไกนี้สามารถใช้เพื่อจำกัดว่าพาร์ติชัน LPAR ใดที่จะเห็น แพ็กเก็ตอีเทอร์เน็ตที่กำหนด

ส่วนการติดต่อ VIOS ควรได้รับการป้องกันมิให้เข้าถึงโดยผู้ใช้ที่ไม่มีสิทธิ อ็อพชันผู้ใช้ VIOS ต้อตั้งค่าให้ตรงตามข้อกำหนดของการประเมินผล ข้อกำหนดที่แท้จริงคือ TSF มี กลไกตรวจสอบว่าความลับตรงตามเกณฑ์เมตริกซ์คุณภาพต่อไปนี้: ความเป็นไปได้ที่ความลับได้มาจากผู้โจมตีในระหว่าง ที่รหัสผ่านมีผลใช้ได้ไม่น้อยกว่า 2^-20 พารามิเตอร์ต่อไปนี้ควรถูกเปลี่ยนแปลงสำหรับผู้ใช้ในไดเร็กทอรี `/etc/security/user`:

maxage
8

maxexpired
1

minother
2

minlen 8

maxrepeats
2

loginretries
3

histexpire
52

histsize 20

ในการเปลี่ยนค่าดีฟอลต์ใช้คำสั่งต่อไปนี้:

```
type oem_setup_env

chsec -f /etc/security/user -s default  -a maxage=8 -a maxexpired=1 -a minother=2
  -a minlen=8 -a maxrepeats=2 -a loginretries=3 -a histexpire=52 -a histsize=20
```

เมื่อผู้ดูแลระบบหลัก (**padmin**) สร้างใหม่ ต้องระบุแอ็ตทริบิวต์ ผู้ใช้เฉพาะสำหรับผู้ใช้นั้น ตัวอย่าง ในการสร้าง ผู้ใช้ที่มีชื่อ *davis* **padmin** จะใช้คำสั่งต่อไปนี้:

```
mkuser  maxage=8 maxexpired=1 minother=2 minlen=8 maxrepeats=2 loginretries=3
  histexpire=52 histsize=20 davis
```

padmin ควรหยุดทำงาน daemons ต่อไปนี้ จากนั้นบูตใหม่:

- ในการลบ **writesrv** และ **ctrmc** ออกจากไฟล์ /etc/inittab:

  ```
  sshd:   stopsrc -s sshd
  ```

- ในการป้องกัน daemon มิให้เริ่มทำงานตอนเปิดเครื่องใหม่ ให้ลบไฟล์ /etc/rc.d/rc2.d/Ksshd และ /etc/rc.d/rc2.d/Ssshd หลังจากการบูตใหม่ หยุดทำงาน RSCT daemons:

  ```
  stopsrc -g rsct_rm stopsrc -g rsct
  ```

ผู้ใช้ทั้งหมด ไม่ว่าจะมีบทบาทใด จะถูกพิจารณาเป็นผู้ใช้ที่ทำหน้าที่ ดูแลระบบ

ผู้ดูแลระบบสามารถรันคำสั่งทั้งหมดยกเว้นคำสั่งที่อยู่ใน รายการต่อไปนี้ที่จำกัดเฉพาะผู้ดูแลหลัก (**padmin**):

- **chdate**
- **chuser**
- **cleargcl**
- **de_access**

- **diagmenu**
- **invscout**
- **loginmsg**
- **lsfailedlogin**
- **lsgcl**
- **mirrorios**
- **mkuser**
- **motd**
- **oem_platform_level**
- **oem_setup_env**
- **redefvg**
- **rmuser**
- **shutdown**
- **unmirrorios**

การควบคุมล็อกอิน

คุณสามารถเปลี่ยนค่าดีฟอลต์ของหน้าจอล็อกอินเพื่อเหตุผลด้านความปลอดภัย ภายหลังการติดตั้งระบบ

แฮกเกอร์สามารถหาข้อมูลอันมีค่าได้จากหน้าจอล็อกอิน AIX ดีฟอลต์ เช่นชื่อโฮสต์และเวอร์ชันของระบบปฏิบัติการ ข้อมูลนี้อาจช่วยให้พวกแฮกเกอร์พิจารณาหาวิธีการใช้ประโยชน์ที่จะพยายามลองดูได้ เพื่อเหตุผลด้านความปลอดภัย คุณอาจต้องการเปลี่ยนค่าดีฟอลต์หน้าจอล็อกอิน ทันทีที่ทำได้หลังการติดตั้งระบบ

เดสก์ท็อป KDE และ GNOME มีปัญหาด้านความปลอดภัยบางอย่างเหมือนกัน สำหรับ ข้อมูลเพิ่มเติมเกี่ยวกับ KDE และ GNOME อ้างอิง *การติดตั้งและการย้าย*

สำหรับข้อมูลเกี่ยวกับผู้ใช้ กลุ่มและรหัสผ่าน ดูที่ “ผู้ใช้ กลุ่ม และรหัสผ่าน” ในหน้า 54

การตั้งค่าควบคุมล็อกอิน:

คุณสามารถตั้งค่าการควบคุมล็อกอินในไฟล์ `/etc/security/login.cfg`

ในการทำให้ระบบโดนโจมตีด้วยการเดารหัสผ่านทำได้ยากขึ้น ให้ตั้งค่าการควบคุมล็อกอินในไฟล์ `/etc/security/login.cfg` ดังนี้:

ตารางที่ 1. แอ็ตทริบิวต์และค่าที่แนะนำสำหรับการควบคุม ล็อกอิน

แอ็ตทริบิวต์	นำใช้กับ PtYs (เน็ตเวิร์ก)	นำใช้กับ TTYs	ค่าที่แนะนำ	หมายเหตุ
sak_enabled	Y	Y	false	คีย์ Secure Attention ไม่ค่อยจำเป็นต้องใช้ ดูที่ "การใช้ Secure Attention Key" ในหน้า 7
logintimes	N	Y		ระบุจำนวนครั้งการล็อกอินที่อนุญาตที่นี่
logindisable	N	Y	4	ปิดใช้งานล็อกอินบนเทอร์มินัลนี้หลังพยายามแล้วล้มเหลวติดต่อกัน 4 ครั้ง
logininterval	N	Y	60	เทอร์มินัลจะถูกปิดใช้งานเมื่อการพยายามที่ไม่สำเร็จตามที่ระบุ ครบจำนวนภายใน 60 วินาที
loginreenable	N	Y	30	เปิดใช้งานเทอร์มินัลอีกครั้งหลังถูกปิดใช้งานโดยอัตโนมัติ หลัง 30 นาที
logindelay	Y	Y	5	เวลาเป็นวินาทีระหว่างการพยายามล็อกอิน ค่านี้ จะถูกคูณด้วยจำนวนครั้งของการพยายามที่ล้มเหลว ตัวอย่าง 5,10,15,20 วินาทีเมื่อค่าเริ่มต้นคือ 5

ข้อจำกัดพอร์ตเหล่านี้ส่วนใหญ่ทำงานได้บนเทอร์มินัลอนุกรมที่ถูกรวมเข้า ไม่ใช่บนเทอร์มินัลจำลองที่ใช้โดยการล็อกอินเน็ตเวิร์ก คุณสามารถระบุเทอร์มินัลโดยชัดแจ้ง ในไฟล์นี้ ตัวอย่างเช่น:

```
/dev/tty0:
        logintimes = 0600-2200
        logindisable = 5
        logininterval = 80
        loginreenable = 20
```

การเปลี่ยนข้อความต้อนต้องบนหน้าจอล็อกอิน:

เพื่อหลีกเลี่ยงการแสดงข้อมูลเฉพาะบนหน้าจอล็อกอิน ให้แก้ไข พารามิเตอร์ *herald* ในไฟล์ /etc/security/login.cfg

herald ดีฟอลต์มีข้อความ ต้อนรับที่แสดงพร้อมกับพร้อมล็อกอินของคุณ ในการเปลี่ยนค่าพารามิเตอร์นี้ คุณ สามารถใช้คำสั่ง **chsec** หรือแก้ไขไฟล์โดยตรง

ตัวอย่าง ต่อไปนี้ใช้คำสั่ง **chsec** เพื่อเปลี่ยน พารามิเตอร์ *herald* ดีฟอลต์:

```
# chsec -f /etc/security/login.cfg -s default
-a herald="Unauthorized use of this system is prohibited.\n\nlogin:"
```

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับคำสั่ง **chsec** ดูที่ *ข้อมูลอ้างอิงคำสั่ง วอลุ่ม 1*

ในการแก้ไข ไฟล์โดยตรง ให้เปิดไฟล์ /etc/security/login.cfg และอัพเดตพารามิเตอร์ *herald* ดังนี้:

```
default:
herald ="Unauthorized use of this system is prohibited\n\nlogin:"
 sak_enable = false
 logintimes =
 logindisable = 0
 logininterval = 0
 loginreenable = 0
 logindelay = 0
```

หมายเหตุ: ในการทำให้ระบบมีความปลอดภัยมากยิ่งขึ้น ให้ตั้งค่าตัวแปร *logindisable* และ *logindelay* เป็นจำนวนที่มากกว่า 0 (# > 0)

การเปลี่ยนหน้าจอล็อกอินสำหรับสภาวะแวดล้อมเดสก์ท็อปทั่วไป:

ปัญหาด้านความปลอดภัยนี้ยังมีผลต่อผู้ใช้ Common Desktop Environment (CDE) หน้าจอล็อกอิน CDE ยังแสดงชื่อโฮสต์และเวอร์ชันระบบปฏิบัติการ อันเป็นค่าดีฟอลต์ เพื่อหลีกเลี่ยงมิให้แสดงข้อมูลนี้ ให้แก้ไขไฟล์ `/usr/dt/config/$LANG/Xresources` โดยที่ **$LANG** อ้างถึงภาษาท้องถิ่นที่ติดตั้งบนเครื่องของคุณ

ในตัวอย่างของเรา สมมติว่า **$LANG** ถูกตั้งค่าเป็น **C** ให้ทำสำเนาไฟล์นี้ไปไว้ในไดเร็กทอรี `/etc/dt/config/C/Xresources` ถัดไป เปิดไฟล์ `/usr/dt/config/C/Xresources` และแก้ไข เพื่อลบข้อความต้อนรับที่มีชื่อโฮสต์และเวอร์ชันระบบปฏิบัติการ ออก

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับปัญหาด้านความปลอดภัย CDE ดูที่ “การจัดการข้อควรพิจารณาของ X11 และ CDE” ในหน้า 45

การปิดใช้งานการแสดงชื่อผู้ใช้และการเปลี่ยนพร้อมต์รหัสผ่าน:

ในสภาวะแวดล้อมที่มีความปลอดภัย อาจจำเป็นต้องซ่อน การแสดงชื่อผู้ใช้ล็อกอินหรือเพื่อให้มีพร้อมต์รหัสผ่านแบบกำหนดเองที่แตกต่าง จากค่าดีฟอลต์

ลักษณะการทำงานของข้อความดีฟอลต์สำหรับพร้อมล็อกอินและรหัสผ่าน ถูกแสดงด้านล่าง:

```
login: foo
foo's Password:
```

ในการปิดใช้งานการแสดงชื่อผู้ใช้จาก พร้อมต์และข้อความแสดงความผิดพลาดระบบ ให้แก้ไขพารามิเตอร์ *usernameecho* ในไฟล์ `/etc/security/login.cfg` ค่าดีฟอลต์ สำหรับ *usernameecho* คือจริงซึ่งส่งผลให้ชื่อผู้ใช้ ถูกแสดง ในการเปลี่ยนพารามิเตอร์นี้ คุณสามารถใช้คำสั่ง **chsec** หรือแก้ไขไฟล์โดยตรง

ตัวอย่างต่อไปนี้ใช้คำสั่ง **chsec** เพื่อเปลี่ยนพารามิเตอร์ *usernameecho* ดีฟอลต์เป็น เท็จ:

```
# chsec -f /etc/security/login.cfg -s default -a usernameecho=false
```

สำหรับ ข้อมูลเพิ่มเติมเกี่ยวกับคำสั่ง **chsec** ดูที่ *ข้อมูลอ้างอิงคำสั่ง วอลุ่ม 1*

ในการ แก้ไขไฟล์โดยตรง ให้เปิดไฟล์ `/etc/security/login.cfg` และเพิ่มหรือแก้ไขพารามิเตอร์ *usernameecho* ดังนี้:

```
default:
 usernamecho = false
```

การตั้งค่าพารามิเตอร์ *usernameecho* เป็นเท็จจะส่งผลให้ไม่มีการแสดงชื่อผู้ใช้ที่พร้อมต์ล็อกอิน โดยชื่อผู้ใช้ถูกทำเครื่องหมายด้วยอักขระ '*' สำหรับพร้อมต์ระบบแทน และข้อความแสดงความผิดพลาดดังแสดงด้านล่าง:

```
login:
***'s Password:
```

พร้อมต์รหัสผ่านอาจถูกแก้ไขแยกต่างหาก เพื่อให้เป็นสตริงแบบกำหนดเองโดยการตั้งค่าพารามิเตอร์ *pwdprompt* ในไฟล์ `/etc/security/login.cfg` ค่าดีฟอลต์ คือสตริง "*user*'s Password: " โดยที่ *user* ถูกแทนค่าด้วยชื่อผู้ใช้ในการพิสูจน์ตัวตน

ในการเปลี่ยนพารามิเตอร์นี้ คุณสามารถใช้คำสั่ง **chsec** หรือแก้ไขโดยตรง

ตัวอย่าง ต่อไปนี้ใช้คำสั่ง **chsec** เพื่อเปลี่ยน พารามิเตอร์ *pwdprompt* ดีฟอลต์เป็น "Password: ":

```
# chsec -f /etc/security/login.cfg -s default -a pwdprompt="Password: "
```

ในการแก้ไขไฟล์โดยตรง ให้เปิดไฟล์ /etc/security/login.cfg และเพิ่มหรือแก้ไขพารามิเตอร์ *pwdprompt* ดังนี้:

```
default:
 pwdprompt = "Password: "
```

การตั้งค่าพารามิเตอร์ *pwdprompt* เป็น "Password: " จะส่งผลในพร้อมต์ที่ระบุที่กำลังแสดงตามล็อกอิน และโดยแอ็พพลิเคชันอื่นที่ใช้พร้อมต์รหัสผ่านระบบ ลักษณะการทำงาน พร้อมค่าล็อกอินเมื่อพร้อมต์ณแบบกำหนดเองถูกตั้งค่า ดังนี้:

```
login: foo
Password:
```

การตั้งค่าพารามิเตอร์การล็อกอินดีฟอลต์ระบบ:

แก้ไขไฟล์ /etc/security/login.cfg เพื่อตั้งค่า พารามิเตอร์ล็อกอินดีฟอลต์ระบบ

ในการตั้งค่าตามค่าดีฟอลต์สำหรับพารามิเตอร์ล็อกอินจำนวนมาก เช่น พารามิเตอร์ที่คุณอาจตั้งค่าสำหรับผู้ใช้ใหม่ (จำนวนการลองล็อกอินใหม่, เปิดให้ล็อกอินได้อีกครั้ง, และภายในการล็อกอิน) แก้ไขไฟล์ /etc/security/login.cfg

การรักษาความปลอดภัยเทอร์มินัลที่ไม่ได้ใส่ใจ:

การใช้คำสั่ง **lock** และ **xlock** เพื่อรักษาความปลอดภัยเทอร์มินัลของคุณ

ระบบทั้งหมดสามารถเสี่ยงได้ถ้าเทอร์มินัลถูกล็อกอินทิ้งไว้และ ไม่ได้ใส่ใจ ปัญหาสำคัญที่สุดเกิดขึ้นเมื่อผู้จัดการระบบปล่อยเทอร์มินัลทิ้งไว้โดยไม่ได้ใส่ใจโดยที่ถูกเปิดใช้งานด้วยสิทธิ root โดยทั่วไป ผู้ใช้ควรล็อกออกจากระบบเมื่อผู้ใช้จะออกห่างจากเทอร์มินัลของตน การออกห่างจากเทอร์มินัลระบบ อาจก่อให้เกิดอันตรายด้านความปลอดภัยที่ไม่มีการรักษาความปลอดภัย เมื่อต้องการล็อกเทอร์มินัลของคุณ ใช้คำสั่ง **lock** ถ้าอินเตอร์เฟสของคุณเป็น AIXwindows ใช้คำสั่ง **xlock**

การเปิดใช้ล็อกออฟอัตโนมัติ:

เปิดใช้การล็อกออฟอัตโนมัติเพื่อป้องกันผู้บุกรุกมิให้เจาะเข้าสู่ช่องโหว่ ด้านความปลอดภัยของระบบ

ข้อกังวลที่เกี่ยวกับความปลอดภัยอีกประการหนึ่งเป็นผลจากผู้ใช้ล็อกอิน ด้วยบัญชีผู้ใช้ของตนทิ้งไว้โดยไม่ได้สนใจเป็นระยะเวลาหนึ่ง สถานการณ์นี้ทำให้ ผู้บุกรุกสามารถควบคุมเทอร์มินัลของผู้ใช้ ซึ่งเป็นการเจาะช่องโหว่ ด้านความปลอดภัยของระบบที่อาจเกิดขึ้นได้

เพื่อหลีกเลี่ยงความเสี่ยงด้านความปลอดภัย ที่อาจเกิดขึ้นประเภทนี้ คุณสามารถเปิดใช้การล็อกออฟอัตโนมัติบนระบบ โดยตั้งค่าตัวแปรระบบ TMOUT และ TIMEOUT เป็นจำนวนวินาทีที่ไม่ทำงาน หลังจากผ่านเวลา ที่ไม่ทำงาน คุณจะถูกล็อกออฟอัตโนมัติ ดังแสดงในตารางต่อไปนี้:

```
TMOUT=600; TIMEOUT=600; export TMOUT TIMEOUT
```

ในตัวอย่างข้างต้น จำนวน 600 เป็นวินาที ซึ่งเท่ากับ 10 นาที เมธอดนี้ใช้ได้กับแอ็พพลิเคชัน shell เท่านั้น ตัวแปรสามารถป้องกันการเขียนทับโดยไม่ตั้งใจหลายครั้ง โดยทำให้อ่านได้ดังนี้:

```
TMOUT TIMEOUT แบบสุ่ม
```

ตัวแปรระบบ TMOUT และ TIMEOUT ถูกกำหนดไว้ในไฟล์ .profile ของผู้ใช้หรือในไฟล์ /etc/security/.profile ซึ่งอนุญาตให้เพิ่มไฟล์ในไฟล์ .profile ของ ผู้ใช้เมื่อสร้างผู้ใช้

การปกป้อง Stack Execution Disable

การป้องกันให้ระบบคอมพิวเตอร์มีความปลอดภัยก่อให้เกิดแนวทางสำคัญของ ธุรกิจตามต้องการ (On Demand) ในโลกของสภาวะแวดล้อมแบบเน็ตเวิร์กอย่างมากทุกวันนี้ ถือเป็นความท้าทายอย่างสูงที่จะป้องกันการโจมตีจากแหล่งที่มาที่แตกต่างมากมาย

มีความเป็นไปได้เพิ่มสูงขึ้นที่ระบบคอมพิวเตอร์จะต้องเป็นเหยื่อของการโจมตี ที่ซับซ้อน ส่งผลให้เกิดการขัดขวางการทำงานประจำวันของธุรกิจและ หน่วยงานรัฐ ขณะที่ยังไม่มีมาตรการในการวัดความปลอดภัยที่สามารถจัดให้มีการป้องกันมิให้เกิดอันตราย จากการโจมตี คุณควรใช้กลไกการรักษาความปลอดภัยหลายๆ วิธีเพื่อยับยั้ง การโจมตีด้านความปลอดภัย ส่วนนี้ครอบคลุมกลไกการรักษาความปลอดภัยที่ใช้กับ AIX เพื่อยับยั้งการโจมตีเนื่องจากการทำงาน ที่ก่อให้เกิดบัฟเฟอร์โอเวอร์โฟลว์

ช่องโหว่ด้านความปลอดภัยอาจเกิดขึ้นในหลายรูปแบบ แต่หนึ่งในวิธีที่นิยม สูงสุดคือการมอนิเตอร์เครื่องมือการดูแลจัดการที่ระบบจัดให้มี ค้นหา และหาประโยชน์จากบัฟเฟอร์โอเวอร์โฟลว์ การโจมตีโดยบัฟเฟอร์โอเวอร์โฟลว์เกิดขึ้นเมื่อ บัฟเฟอร์โปรแกรมภายในถูกเขียนทับ เนื่องจากข้อมูลไม่ถูกตรวจสอบความถูกต้อง อย่างเหมาะสม (เช่นบรรทัดคำสั่ง ตัวแปรสภาวะแวดล้อม ดิสก์ หรือ I/O เทอร์มินัล) โค้ด การโจมตีถูกแทนในกระบวนการที่กำลังทำงานผ่านบัฟเฟอร์โอเวอร์โฟลว์ การเปลี่ยนพาธการทำงานของกระบวนการที่กำลังทำงาน รีเทิร์นแอดเดรสถูกเขียนทับ และเปลี่ยนเส้นทางไปยังตำแหน่งโค้ดที่แทรก สาเหตุทั่วไปของช่องโหว่ได้แก่ การตรวจสอบข้อจำกัดที่ไม่เหมาะสม หรือไม่มีอยู่ หรือการสันนิษฐาน ไม่ถูกต้องเกี่ยวกับความถูกต้องของแหล่งข้อมูล ตัวอย่าง บัฟเฟอร์โอเวอร์โฟลว์สามารถเกิดขึ้นเมื่อ อ็อบเจ็กต์ข้อมูลมีขนาดใหญ่พอที่จะเก็บข้อมูล 1 KB แต่โปรแกรมไม่ ได้ตรวจสอบข้อจำกัดของอินพุต และทำให้สามารถคัดลอกขนาดมากกว่า 1 KB ลงในอ็อบเจ็กต์ข้อมูล

เป้าหมายของผู้บุกรุกคือโจมตีคำสั่งและ/หรือเครื่องมือที่ให้สิทธิ์พิเศษ root แก่ผู้ใช้ปกติ ควบคุมโปรแกรมที่ได้รับอนุญาตให้มีสิทธิ์พิเศษ ทั้งหมดถูกเปิดใช้งาน การอนุญาตให้เกิดการโอเวอร์โฟลว์ของบัฟเฟอร์ ผู้โจมตีโดยปกติ เน้นที่ชุด UID ที่ root เป็นเจ้าของ หรือโปรแกรมที่นำไปสู่การทำงาน ของเชลล์ เพื่อให้ได้การเข้าถึงเชลล์ของระบบจากระดับ root

คุณสามารถป้องกันการโจมตีเหล่านี้ได้โดยการบล็อกการทำงานของโค้ดการโจมตี ที่ทำผ่านบัฟเฟอร์โอเวอร์โฟลว์ปิดใช้งานการเรียกทำงานบนพื้นที่หน่วยความจำของกระบวนการ ที่โดยทั่วไปไม่มีการทำงานเกิดขึ้น (พื้นที่หน่วยความจำสแต็กและฮีป)

กลไกการปกป้องบัฟเฟอร์โอเวอร์โฟลว์ SED:

AIX ได้ เปิดใช้งานกลไก stack execution disable (SED) เพื่อปิดใช้งาน การทำงานของโค้ดบนสแต็กและพื้นที่ข้อมูลที่เลือกของกระบวนการ

โดยการปิดใช้งานการทำงานและจากนั้นจบการทำงานโปรแกรม ที่ละเมิด ผู้โจมตีถูกป้องกันมิให้ได้รับสิทธิ์พิเศษผู้ใช้ root ผ่านการโจมตีบัฟเฟอร์โอเวอร์โฟลว์ แม้คุณลักษณะนี้จะไม่หยุดทำงาน บัฟเฟอร์โอเวอร์โฟลว์ แต่จะมีการป้องกันโดยการปิดใช้การทำงาน การโจมตีบนบัฟเฟอร์ที่เกิดโอเวอร์โฟลว์

เริ่มตั้งแต่ตัวประมวลผลกลุ่ม POWER4 คุณสามารถใช้คุณลักษณะการเปิดใช้งานและ/หรือปิดใช้งานการทำงานระดับหน้าสำหรับหน่วยความจำ กลไก AIX SED ใช้การสนับสนุนฮาร์ดแวร์ที่สำคัญนี้สำหรับการใช้ คุณลักษณะไม่ให้มีการทำงานบนพื้นที่หน่วยความจำที่เลือก เมื่อคุณลักษณะนี้ ถูกเปิดใช้งาน ระบบปฏิบัติการจะตรวจสอบและแฟล็กไฟล์ต่างๆ ระหว่าง โปรแกรมเรียกทำงาน จากน้นแจ้งเตือนตัวจัดการหน่วยความจำระบบปฏิบัติการ และตัวจัดการกระบวนการว่า SED ถูกเปิดใช้งาน

สำหรับกระบวนการ ที่ถูกสร้างขึ้น พื้นที่หน่วยความจำที่เลือกถูกทำเครื่องหมายสำหรับไม่ให้มีการทำงาน ถ้ามีการทำงานใดๆ เกิดขึ้นบนพื้นที่ที่ทำเครื่องหมายเหล่านี้ ฮาร์ดแวร์จะสร้าง แฟล็กข้อยกเว้นและระบบปฏิบัติการหยุดทำงานกระบวนการ ที่สัมพันธ์กัน รายละเอียดข้อยกเว้นและการจบการทำงานแอ็พพลิเคชันถูกดักจับ ผ่านทางเหตุการณ์บันทึกข้อผิดพลาด AIX

SED ถูกประยุกต์ใช้เป็นหลักผ่านคำสั่ง **sedmgr** คำสั่ง **sedmgr** อนุญาตการควบคุมโหมด SED ของการดำเนินการทั้งระบบรวมถึงการตั้งค่าไฟล์เรียกทำงานตาม แฟล็ก SED

โหมดและการมอนิเตอร์ SED:

กลไก stack execution disable (SED) ใน AIX ถูกนำใช้ ทางแฟล็กโหมดทั้งระบบ รวมถึงแฟล็กส่วนหัวตามส่วนหัว แต่ละค่า

ขณะที่แฟล็กทั้งระบบควบคุมการดำเนินการทั้งระบบของ SED แฟล็กระดับไฟล์ระบุวิธีที่ไฟล์ถูกปฏิบัติใน SED กลไกการปกป้องบัฟเฟอร์โอเวอร์โฟลว์ (BOP) มีโหมดการดำเนินการ ทั้งระบบสี่โหมด:

off กลไก SED ถูกปิดทำงาน และไม่มีกระบวนการถูกทำเครื่องหมายสำหรับการปกป้อง SED

select เฉพาะชุดของไฟล์ที่เลือกถูกเปิดใช้งานและมอนิเตอร์สำหรับการปกป้อง SED ชุดของไฟล์ที่เลือกถูกเลือกโดยการตรวจทานแฟล็กที่เกี่ยวกับ SED ในส่วนหัวของไบนารีโปรแกรมเรียกทำงาน ส่วนหัวโปรแกรมเรียกทำงาน เปิดใช้งานแฟล็กที่เกี่ยวกับ SED เพื่อร้องขอให้รวมในโหมด **select**

setidfiles
อนุญาตให้คุณเปิดใช้งาน SED ไม่เฉพาะสำหรับไฟล์การร้องขอ กลไก แต่รวมถึงไฟล์ระบบ **setuid** และ **setgid** ที่สำคัญ ทั้งหมด ในโหมดนี้ ระบบปฏิบัติการไม่เพียงจัดให้มี SED สำหรับ ไฟล์ที่มีแฟล็ก **request** SED ตั้งค่า แต่ยังเปิดใช้งาน SED สำหรับไฟล์เรียกทำงานที่มีคุณสมบัติต่อไปนี้ (ยกเว้น ไฟล์ที่ทำเครื่องหมาย *exempt* ในส่วนหัวไฟล์):

- ไฟล์ SETUID ที่ root เป็นเจ้าของ
- ไฟล์ SETGID ที่มีกลุ่มหลักเป็น **system** หรือ **security**

all โปรแกรมเรียกทำงานทั้งหมดที่โหลดบนระบบได้รับการป้องกันด้วย SED ยกเว้นไฟล์ที่ร้องขอการยกเว้นจากโหมด SED แฟล็ก ที่เกี่ยวกับการยกเว้นเป็นส่วนหนึ่งของส่วนหัวโปรแกรมเรียกทำงาน

คุณลักษณะ SED บน AIX ยัง มีความสามารถในการมอนิเตอร์แทนการหยุดทำงานกระบวนการเมื่อ เกิด exception การควบคุมทั้งระบบนี้อนุญาตให้ผู้ดูแลระบบ ตรวจสอบหาจุดที่เกิดความเสียหายและปัญหาในสภาวะแวดล้อมระบบโดยการมอนิเตอร์ ก่อนที่ SED ถูกนำไปพัฒนาใช้ในระบบการทำงานจริง

คำสั่ง **sedmgr** จัดให้มีอ็อพชันที่อนุญาตให้คุณเปิดใช้งาน SED เพื่อมอนิเตอร์ไฟล์ แทนการหยุดทำงานกระบวนการเมื่อเกิด exceptions ผู้ดูแลระบบ สามารถวิเคราะห์ว่าโปรแกรมเรียกทำงานกำลังทำ การเรียกทำงานสแต็กใดๆ ที่ถูกต้องหรือไม่ การตั้งค่านี้ทำงานร่วมกับ ชุดโหมดทั้งระบบโดยใช้อ็อพชัน `-c` เมื่อโหมด **monitor** ถูกเปิดทำงาน ระบบอนุญาต ให้กระบวนการดำเนินการต่อแม้จะเกิด exception ที่เกี่ยวกับ SED แทนการหยุดทำงานกระบวนการ ระบบปฏิบัติการจะบันทึก exception ไว้ในบันทึกข้อผิดพลาด AIX ถ้า การมอนิเตอร์ SED ปิดทำงาน ระบบปฏิบัติการจะหยุดทำงานกระบวนการใดๆ ที่ละเมิด และก่อให้เกิด exception ต่อโปรแกรมอำนวยความสะดวก SED

การเปลี่ยนแปลงใดๆ กับแฟล็กทั้งระบบของโหมด SED จำเป็นที่คุณ ต้องรีสตาร์ทระบบเพื่อให้การเปลี่ยนแปลงมีผล ประเภทของเหตุการณ์ทั้งหมด เหล่านี้จะถูกตรวจสอบ

แฟล็ก SED สำหรับไฟล์เรียกทำงาน:

ใน AIX คุณสามารถ ใช้คำสั่ง **sedmgr** เพื่อแฟล็กไฟล์เรียกทำงานจากกลไก SE

ตัวเชื่อมโยงได้ถูกปรับปรุงเพื่อสนับสนุนแฟล็ก 2 แบบที่เกี่ยวข้องกับ SED ใหม่ เพื่อเปิดใช้งานอ็อพชัน `select` และ `exempt` ในส่วนหัวของไฟล์ที่เรียกใช้งาน แฟล็ก **select** อนุญาต ให้ไฟล์ที่รันได้สามารถร้องขอและเป็นส่วนหนึ่งของการป้องกัน SED ระหว่างโหมด `select` ของการดำเนินงาน SED ทั้งระบบ โดยที่แฟล็ก **exempt** อนุญาตให้ไฟล์ที่รันได้ สามารถร้องขอการยกเว้นจากกลไก SED ไฟล์ที่รันได้เหล่านี้ไม่ถูกเปิดใช้งานสำหรับพื้นที่หน่วยความจำประมวลผลใดๆ

แฟล็กการยกเว้นอนุญาตให้ผู้ดูแลระบบมอนิเตอร์กลไก SED และวิเคราะห์สถานการณ์ ผู้ดูแลระบบสามารถเปิดใช้งานการทำงาน บนพื้นที่สแต็กและพื้นที่ข้อมูลที่จำเป็นสำหรับแอ็พพลิเคชัน โดยมีความเข้าใจใน ความเสี่ยงที่สัมพันธ์

ตารางต่อไปนี้แสดงวิธีการตั้งค่าทั้งระบบและการตั้งค่าไฟล์ ที่มีผลต่อโหมด SED ของการดำเนินการ:

ตารางที่ 2. การตั้งค่าทั้งระบบและการตั้งค่าไฟล์ที่มีผล ต่อโหมด SED

โหมด SED ระบบ	แฟล็ก SED ไฟล์เรียกทำงาน			ไฟล์ Setuid-root หรือ setgid-system/security
	ร้องขอ	ยกเว้น	ระบบ	
ปิด	–	–	–	–
เลือก	เปิดใช้งาน	–	–	–
setgidfiles	เปิดใช้งาน	–	–	เปิดใช้งาน
ทั้งหมด	เปิดใช้งาน	–	เปิดใช้งาน	เปิดใช้งาน

ประเด็นและข้อควรพิจารณา SED:

โดยดีฟอลต์ AIX SED มาพร้อมกับโหมด **select** โปรแกรม **setuid** และ **setgid** จำนวนหนึ่ง เป็นแบบเปิดใช้งาน **select** สำหรับ SED และทำงานในโหมดที่ป้องกันโดย ดีฟอลต์

การเปิดใช้งาน SED อาจทำให้ไบนารีไฟล์เก่าเสียหายถ้าไม่สามารถ จัดการคุณลักษณะ no-execution บนพื้นที่สแต็ก ฮีป แอ็พพลิเคชันเหล่านี้ต้องรันบนพื้นที่สแต็กข้อมูล ผู้ดูแลระบบ สามารถวิเคราะห์สถานการณ์และแฟล็กเพื่อการยกเว้น โดยใช้คำสั่ง **bopmgr** AIX Java™ 1.3.1 และ AIX Java 1.4.2 มีคอมไพลเลอร์ Just-In-Time (JIT) ที่สร้างและรันอ็อบเจ็กต์โค้ดเดิมแบบไดนามิกขณะรันแอ็พพลิเคชัน Java (Java Virtual Machine เลือกว่าโค้ดใดที่จะคอมไพล์ ตามโปรไฟล์การทำงานของแอ็พพลิเคชัน) อ็อบเจ็กต์โค้ดนี้ ถูกเก็บในบัฟเฟอร์ข้อมูลที่จัดสรรโดย JIT ดังนั้น ถ้า AIX ถูกตั้งค่าเพื่อรัน ในโหมด SED **ALL** ต้องกำหนดคอนฟิกแฟล็กการยกเว้นของไบนารีไฟล์ Java

เมื่อแฟล็กที่เกี่ยวกับ SED ในไฟล์เรียกทำงานถูกเปลี่ยนแปลง แฟล็ก จะใช้กับการโหลดล่วงหน้าและการทำงานของไฟล์เท่านั้น การเปลี่ยนแปลงนี้ ไม่ใช้กับกระบวนการที่กำลังดำเนินการอยู่ในขณะนี้โดยยึดตามไฟล์นี้ โปรแกรมอำนวยความสะดวก SED ควบคุมและมอนิเตอร์โปรแกรมเรียกทำงานทั้งแบบ 32 บิตและ 64 บิต สำหรับการตั้งค่าทั้งระบบและระดับไฟล์ โปรแกรมอำนวยความสะดวก SED มีอยู่เฉพาะเมื่อระบบปฏิบัติการ AIX ใช้กับเคอร์เนล 64 บิตเท่านั้น

ข้อมูลที่เกี่ยวข้อง

คำสั่ง **sedmgr**

AIX **Error-Logging Facility**

การจัดการข้อควรพิจารณาของ X11 และ CDE

มีความเปราะบางของการรักษาความปลอดภัยที่เกี่ยวข้อง กับ X11 X server และ Common Desktop Environment (CDE)

การลบไฟล์ /etc/rc.dt:

การลบไฟล์ /etc/rc.dt บนระบบ ที่ต้องมีการรักษาความปลอดภัยระดับสูง

แม้การทำงานอินเตอร์เฟส CDE จะสะดวกสำหรับผู้ใช้ แต่ก็ เกี่ยวข้องกับประเด็นเรื่องการรักษาความปลอดภัย ด้วยเหตุผลนี้ อย่ารัน CDE บนเซิร์ฟเวอร์ ที่ต้องมีการรักษาความปลอดภัยระดับสูง วิธีแก้ปัญหาที่ดีที่สุดคือการหลีกเลี่ยงการติดตั้ง ชุดไฟล์ CDE (dt) ถ้าคุณได้ติดตั้งชุดไฟล์เหล่านี้บนระบบของคุณ ขอให้ถอนการติดตั้งออก โดยเฉพาะอย่างยิ่งสคริปต์ /etc/rc.dt ซึ่งจะเริ่มทำงาน CDE

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ CDE ดูที่ *การจัดการระบบปฏิบัติการและอุปกรณ์*

การป้องกันการมอนิเตอร์รีโมต X server ที่ไม่ได้รับอนุญาต:

ปัญหาด้านความปลอดภัยที่สำคัญที่เชื่อมโยงกับ X11 server ถูกทำการมอนิเตอร์ แบบไม่มีการโต้ตอบโดยไม่ได้รับอนุญาต ของรีโมตเซิร์ฟเวอร์

คำสั่ง **xwd** และ **xwud** สามารถใช้มอนิเตอร์กิจกรรม X server ได้เนื่องจากมีความสามารถในการ ดักจับการเคาะคีย์บอร์ด ซึ่ง สามารถเปิดเผยให้ทราบรหัสผ่านและข้อมูลที่มีความอ่อนไหวอื่นๆ ในการ แก้ปัญหานี้ ให้ลบไฟล์เรียกทำงานเหล่านี้ออก เว้น แต่จะจำเป็น ต้องใช้ภายใต้การตั้งค่าของคุณ หรือที่เป็นทางเลือก คือเปลี่ยนการเข้าถึงคำสั่งเหล่านี้ ให้เป็น root เท่านั้น

คำสั่ง **xwd** และ **xwud** อยู่ในชุดไฟล์ X11.apps.clients

ถ้า คุณจำเป็นต้องเก็บคำสั่ง **xwd** และ **xwud** ให้พิจาณาการใช้ OpenSSH และ MIT Magic Cookies แอ็พพลิเคชันของบุคคลที่ สามเหล่านี้ ช่วยป้องกันความเสี่ยงที่เกิดขึ้นโดยกาการรันคำสั่ง **xwd** และ **xwud**

สำหรับ ข้อมูลเพิ่มเติมเกี่ยวกับ OpenSSH และ MIT Magic Cookies อ้างอิงที่เอกสารคู่มือของแต่ละแอ็พพลิเคชัน ตามลำดับ

การเปิดใช้งานและปิดใช้งานค่าควบคุมการเข้าใช้:

เซิร์ฟเวอร์ X อนุญาตให้โฮสต์รีโมตใช้คำสั่ง **xhost +** เพื่อเชื่อมต่อระบบของคุณ

ทำให้แน่ใจว่าคุณระบุชื่อโฮสต์ด้วยคำสั่ง **xhost +** เนื่องจากจะปิดใช้งานค่าควบคุมการเข้าใช้สำหรับเซิร์ฟเวอร์ X คำสั่งนี้ อนุญาตให้คุณให้สิทธิ การเข้าถึงแก่โฮสต์ที่เจาะจง ซึ่งช่วยให้ง่ายต่อการมอนิเตอร์การโจมตีที่อาจเกิดขึ้นได้ กับเซิร์ฟเวอร์ X ในการให้สิทธิการเข้าถึงแก่โฮสต์ที่เจาะจง ให้รันคำสั่ง **xhost** ดังนี้:

```
# xhost + hostname
```

ถ้าคุณ ไม่ได้ระบุชื่อโฮสต์ จะให้สิทธิการเข้าถึงแก่โฮสต์ทั้งหมด

สำหรับ ข้อมูลเพิ่มเติมเกี่ยวกับคำสั่ง **xhost** ดูที่ *การอ้างอิงคำสั่ง*

การปิดใช้สิทธิผู้ใช้ที่จะรันคำสั่ง xhost:

คุณสามารถป้องกันการทำงานที่มิได้รับอนุญาตของคำสั่ง **xhost** ได้โดยใช้คำสั่ง **chmod**

อีกวิธีหนึ่งเพื่อให้แน่ใจว่าคำสั่ง **xhost** กำลังถูกใช้อย่างเหมะสมเพื่อจำกัดการทำงานของคำสั่งนี้ให้เฉพาะสิทธิผู้ใช้ root เท่านั้น ในการทำนี้ ใช้คำสั่ง **chmod** เพื่อเปลี่ยน สิทธิของ /usr/bin/X11/xhost เป็น 744 ดังนี้:

```
chmod 744/usr/bin/X11/xhost
```

รายการโปรแกรม setuid/setgid

มีโปรแกรม setuid/setgid ที่แตกต่างกันบนระบบ AIX คุณสามารถลบสิทธิพิเศษ เหล่านี้บนคำสั่งที่ไม่จำเป็นต้องมีสำหรับผู้ใช้ทั่วไป

โปรแกรมต่อไปนี้รวมอยู่ในการติดตั้ง AIX ปกติ ในระบบ AIX ที่ตั้งค่า CC รายการนี้ถูกตัดออกและรวม โปรแกรมน้อยลง

- /opt/IBMinvscout/bin/invscoutClient_VPD_Survey
- /opt/IBMinvscout/bin/invscoutClient_PartitionID
- /usr/lpp/diagnostics/bin/diagsetrto
- /usr/lpp/diagnostics/bin/Dctrl
- /usr/lpp/diagnostics/bin/diagela
- /usr/lpp/diagnostics/bin/diagela_exec
- /usr/lpp/diagnostics/bin/diagrpt
- /usr/lpp/diagnostics/bin/diagrto
- /usr/lpp/diagnostics/bin/diaggetrto
- /usr/lpp/diagnostics/bin/update_manage_flash
- /usr/lpp/diagnostics/bin/utape
- /usr/lpp/diagnostics/bin/uspchrp
- /usr/lpp/diagnostics/bin/update_flash
- /usr/lpp/diagnostics/bin/uesensor
- /usr/lpp/diagnostics/bin/usysident
- /usr/lpp/diagnostics/bin/usysfault
- /usr/lpp/X11/bin/xlock
- /usr/lpp/X11/bin/aixterm
- /usr/lpp/X11/bin/xterm
- /usr/lpp/X11/bin/msmitpasswd
- /usr/lib/boot/tftp
- /usr/lib/lpd/digest
- /usr/lib/lpd/rembak
- /usr/lib/lpd/pio/etc/piodmgrsu
- /usr/lib/lpd/pio/etc/piomkpq
- /usr/lib/lpd/pio/etc/pioout
- /usr/lib/mh/slocal
- /usr/lib/perf/libperfstat_updt_dictionary
- /usr/lib/sa/sadc
- /usr/lib/semutil

- /usr/lib/trcload
- /usr/sbin/allocp
- /usr/sbin/audit
- /usr/sbin/auditbin
- /usr/sbin/auditcat
- /usr/sbin/auditconv
- /usr/sbin/auditmerge
- /usr/sbin/auditpr
- /usr/sbin/auditselect
- /usr/sbin/auditstream
- /usr/sbin/backbyinode
- /usr/sbin/cfgmgr
- /usr/sbin/chcod
- /usr/sbin/chcons
- /usr/sbin/chdev
- /usr/sbin/chpath
- /usr/sbin/chtcb
- /usr/sbin/cron
- /usr/sbin/acct/accton
- /usr/sbin/arp64
- /usr/sbin/arp
- /usr/sbin/devinstall
- /usr/sbin/diag_exec
- /usr/sbin/entstat
- /usr/sbin/entstat.ethchan
- /usr/sbin/entstat.scent
- /usr/sbin/diskusg
- /usr/sbin/exec_shutdown
- /usr/sbin/fdformat
- /usr/sbin/format
- /usr/sbin/fuser
- /usr/sbin/fuser64
- /usr/sbin/getlvcb
- /usr/sbin/getlvname
- /usr/sbin/getvgname
- /usr/sbin/grpck

- /usr/sbin/getty
- /usr/sbin/extendvg
- /usr/sbin/fastboot
- /usr/sbin/frcactrl64
- /usr/sbin/frcactrl
- /usr/sbin/inetd
- /usr/sbin/invscout
- /usr/sbin/invscoutd
- /usr/sbin/ipl_varyon
- /usr/sbin/keyenvoy
- /usr/sbin/krlogind
- /usr/sbin/krshd
- /usr/sbin/lchangelv
- /usr/sbin/lchangepv
- /usr/sbin/lchangevg
- /usr/sbin/lchlvcopy
- /usr/sbin/lcreatelv
- /usr/sbin/ldeletelv
- /usr/sbin/ldeletepv
- /usr/sbin/lextendlv
- /usr/sbin/lmigratelv
- /usr/sbin/lmigratepp
- /usr/sbin/lparsetres
- /usr/sbin/lpd
- /usr/sbin/lquerylv
- /usr/sbin/lquerypv
- /usr/sbin/lqueryvg
- /usr/sbin/lqueryvgs
- /usr/sbin/lreducelv
- /usr/sbin/lresynclp
- /usr/sbin/lresynclv
- /usr/sbin/lsaudit
- /usr/sbin/lscfg
- /usr/sbin/lscons
- /usr/sbin/lslv
- /usr/sbin/lspath

- /usr/sbin/lspv
- /usr/sbin/lsresource
- /usr/sbin/lsrset
- /usr/sbin/lsslot
- /usr/sbin/lsuser
- /usr/sbin/lsvg
- /usr/sbin/lsvgfs
- /usr/sbin/login
- /usr/sbin/lvaryoffvg
- /usr/sbin/lvaryonvg
- /usr/sbin/lvgenmajor
- /usr/sbin/lvgenminor
- /usr/sbin/lvrelmajor
- /usr/sbin/lvrelminor
- /usr/sbin/lsmcode
- /usr/sbin/mailq
- /usr/sbin/mkdev
- /usr/sbin/mklvcopy
- /usr/sbin/mknod
- /usr/sbin/mkpasswd
- /usr/sbin/mkpath
- /usr/sbin/mkvg
- /usr/sbin/mount
- /usr/sbin/netstat64
- /usr/sbin/mtrace
- /usr/sbin/ndp
- /usr/sbin/newaliases
- /usr/sbin/named9
- /usr/sbin/named8
- /usr/sbin/netstat
- /usr/sbin/nfsstat
- /usr/sbin/pdelay
- /usr/sbin/pdisable
- /usr/sbin/penable
- /usr/sbin/perf/diag_tool/getschedparms
- /usr/sbin/perf/diag_tool/getvmparms

- /usr/sbin/phold
- /usr/sbin/portmir
- /usr/sbin/pshare
- /usr/sbin/pstart
- /usr/sbin/putlvcb
- /usr/sbin/putlvodm
- /usr/sbin/qdaemon
- /usr/sbin/quota
- /usr/sbin/reboot
- /usr/sbin/redefinevg
- /usr/sbin/repquota
- /usr/sbin/restbyinode
- /usr/sbin/rmdev
- /usr/sbin/ping
- /usr/sbin/rmgroup
- /usr/sbin/rmpath
- /usr/sbin/rmrole
- /usr/sbin/rmuser
- /usr/sbin/rsct/bin/ctstrtcasd
- /usr/sbin/srcd
- /usr/sbin/srcmstr
- /usr/sbin/rmsock64
- /usr/sbin/sendmail_ssl
- /usr/sbin/sendmail_nonssl
- /usr/sbin/rmsock
- /usr/sbin/sliplogin
- /usr/sbin/sendmail
- /usr/sbin/rwhod
- /usr/sbin/route
- /usr/sbin/snappd
- /usr/sbin/swap
- /usr/sbin/swapoff
- /usr/sbin/swapon
- /usr/sbin/swcons
- /usr/sbin/switch.prt
- /usr/sbin/synclvodm

- /usr/sbin/tsm
- /usr/sbin/umount
- /usr/sbin/umountall
- /usr/sbin/unmount
- /usr/sbin/varyonvg
- /usr/sbin/watch
- /usr/sbin/talkd
- /usr/sbin/timedc
- /usr/sbin/uucpd
- /usr/bin/bellmail
- /usr/bin/at
- /usr/bin/capture
- /usr/bin/chcore
- /usr/bin/acctras
- /usr/bin/acctctl
- /usr/bin/chgroup
- /usr/bin/chkey
- /usr/bin/chque
- /usr/bin/chquedev
- /usr/bin/chrole
- /usr/bin/chsec
- /usr/bin/chuser
- /usr/bin/confsrc
- /usr/bin/crontab
- /usr/bin/enq
- /usr/bin/filemon
- /usr/bin/errpt
- /usr/bin/fileplace
- /usr/bin/fileplacej2
- /usr/bin/fileplacej2_64
- /usr/bin/ftp
- /usr/bin/getconf
- /usr/bin/ipcs
- /usr/bin/ipcs64
- /usr/bin/iostat
- /usr/bin/logout

- /usr/bin/lscore
- /usr/bin/lssec
- /usr/bin/mesg
- /usr/bin/mkgroup
- /usr/bin/mkque
- /usr/bin/mkquedev
- /usr/bin/mkrole
- /usr/bin/mkuser
- /usr/bin/netpmon
- /usr/bin/newgrp
- /usr/bin/pagdel
- /usr/bin/paginit
- /usr/bin/paglist
- /usr/bin/passwd
- /usr/bin/pwck
- /usr/bin/pwdadm
- /usr/bin/pwdck
- /usr/bin/rm_mlcache_file
- /usr/bin/rdist
- /usr/bin/remsh
- /usr/bin/rlogin
- /usr/bin/rexec
- /usr/bin/rcp
- /usr/bin/rmque
- /usr/bin/rmquedev
- /usr/bin/rsh
- /usr/bin/ruptime
- /usr/bin/rwho
- /usr/bin/script
- /usr/bin/setgroups
- /usr/bin/setsenv
- /usr/bin/shell
- /usr/bin/su
- /usr/bin/sysck
- /usr/bin/tcbck
- /usr/bin/sysck_r

- /usr/bin/telnet
- /usr/bin/tftp
- /usr/bin/traceroute
- /usr/bin/tn
- /usr/bin/tn3270
- /usr/bin/usrck
- /usr/bin/utftp
- /usr/bin/vmstat
- /usr/bin/vmstat64
- /usr/bin/yppasswd
- /sbin/helpers/jfs2/backbyinode
- /sbin/helpers/jfs2/diskusg
- /sbin/helpers/jfs2/restbyinode

ผู้ใช้ กลุ่ม และรหัสผ่าน

คุณสามารถจัดการ AIX ผู้ใช้ และกลุ่ม

การสร้างโฮมไดเร็กทอรีโดยอัตโนมัติเมื่อล็อกอิน

ระบบปฏิบัติการ AIX สามารถสร้างโฮมไดเร็กทอรีเมื่อผู้ใช้ล็อกอินโดยอัตโนมัติ

คุณลักษณะนี้เป็นประโยชน์สำหรับผู้ใช้ที่กำหนดแบบรีโมต (ตัวอย่างเช่น ผู้ใช้ที่กำหนดในเซิร์ฟเวอร์ LDAP) ผู้อาจไม่มีโฮมไดเร็กทอรี ในระบบโลคัล ระบบปฏิบัติการ AIX จัดเตรียมสองกลไกเพื่อสร้างโฮมเพจแบบอัตโนมัติ เมื่อผู้ใช้ล็อกอิน: กลไก AIX และกลไก PAM วิธีเหล่านี้สามารถเปิดใช้พร้อมกันได้

วิธี AIX วิธี AIX ครอบคลุม ถึงการล็อกอินผ่านคำสั่งต่อไปนี้: **getty**, **login**, **rlogin**, **rsh**, **telnet** และ **tsm** กลไก AIX สนับสนุนการพิสูจน์ตัวตน STD_AUTH และการพิสูจน์ตัวตน PAM_AUTH โดยใช้โมดูล pam_aix เปิดใช้งานกลไก AIX ในไฟล์ **/etc/security/login.cfg** โดยการตั้งค่าแอ็ตทริบิวต์ mkhomeatlogin ของ usw stanza เป็น `true` (ดูที่ไฟล์ `/etc/security/login.cfg` สำหรับข้อมูลเพิ่มเติมเกี่ยวกับไฟล์) ใช้คำสั่ง **chsec** เพื่อเปิดหรือปิดใช้งานคุณลักษณะ automatic-home-directory-creation-at-login ตัวอย่าง ในการเปิดใช้คุณลักษณะ ให้รันคำสั่งต่อไปนี้:

```
# chsec -f /etc/security/login.cfg -s usw -a mkhomeatlogin=true
```

เมื่อ ถูกเปิดใช้งาน กระบวนการล็อกอินจะตรวจหาโฮมไดเร็กทอรีของผู้ใช้หลัง จากการพิสูจน์ตัวตนสำเร็จ ถ้าไม่มีโฮมไดเร็กทอรีของผู้ใช้อยู่ จะสร้างโฮมไดเร็กทอรีขึ้น

หมายเหตุ: แอ็ตทริบิวต์ `mkhomeatlogin` สนับสนุนบน AIX เวอร์ชัน 6.1 ที่มีระดับเทคโนโลยี 6100-02 หรือใหม่กว่าเท่านั้น

วิธี PAM

AIX ยังจัดให้มีโมดูล pam_mkuserhome สำหรับการสร้างโฮมไดเร็กทอรีสำหรับวิธี PAM โมดูล pam_mkuserhome

สามารถถูกสแต็กร่วมกับโมดูลเซสชันอื่นสำหรับเซอร์วิสการล็อกอิน ในการเปิดใช้โมดูล PAM นี้สำหรับเซอร์วิส ต้องเพิ่มรายการใน เซอร์วิสนั้น ตัวอย่าง ในการเปิดใช้การสร้างโฮมไดเร็กทอรีผ่าน คำสั่ง **telnet** โดยใช้ PAM ให้เพิ่มรายการต่อไปนี้ในไฟล์ **/etc/pam.cfg**:

```
telnet session optional pam_mkuserhome
```

ID บัญชีผู้ใช้

บัญชีผู้ใช้แต่ละบัญชีจะมี ID ตัวเลขซึ่งระบุถึงบัญชีผู้ใช้นั้น โดยเฉพาะ ระบบปฏิบัติการ AIX ให้สิทธิ์ตาม ID แอคเคาต์

เป็นสิ่งสำคัญคือต้องเข้าใจว่าบัญชีผู้ใช้ที่มี ID เหมือนกัน นั้นแท้จริงแล้วเป็นบัญชีผู้ใช้เดียวกัน เมื่อสร้างผู้ใช้และกลุ่ม คำสั่ง AIX **mkuser** และ **mkgroup** จะตรวจสอบหา รีจิสทรีปลายทางเสมอเพื่อให้แน่ใจว่าบัญชีผู้ใช้ที่ถูกสร้าง ไม่มี ID ชนกับบัญชีผู้ใช้ที่มีอยู่แล้ว

ทั้งยังสามารถตั้งค่าระบบให้ตรวจสอบรีจิสทรีของผู้ใช้ (กลุ่ม) ทั้งหมด ระหว่างการสร้างบัญชีผู้ใช้ได้โดยใช้แอ็ตทริบิวต์ระบบ **dist_uniqid** แอ็ตทริบิวต์ **dist_uniqid** ของ usw stanza ในไฟล์ `/etc/security/login.cfg` สามารถจัดการได้โดยใช้คำสั่ง **chsec** ในการตั้งค่า ระบบเพื่อให้ตรวจหาการชนกันของ id เสมอกับรีจิสทรีทั้งหมด ให้รัน:

```
# chsec -f /etc/security/login.cfg -s usw -a dist_uniqid=always
```

มีค่าที่ใช้ได้สามค่าสำหรับแอ็ตทริบิวต์ **dist_uniqid** :

never ค่านี้ไม่ตรวจสอบการชนกันของ ID กับรีจิสทรีที่ไม่ได้เป็น ค่าปลายทาง (ดีฟอลต์)

always ค่านี้ตรวจหาการชนกันของ ID กับรีจิสทรีอื่นทั้งหมด ถ้าพบการชนกันระหว่างรีจิสทรีปลายทางกับรีจิสทรีอื่นใด คำสั่ง **mkuser** (**mkgroup**) จะเลือก ID เฉพาะซึ่งไม่ถูกใช้โดยรีจิสทรีใด โดยจะล้มเหลวต่อเมื่อ ค่า ID ถูกระบุจากบรรทัดคำสั่ง (ตัวอย่าง `mkuser id=234 foo` และ ID 234 ถูกใช้งานอยู่แล้วโดยผู้ใช้หนึ่งใน ค่ารีจิสทรีใดๆ)

uniqbyname
ค่านี้ตรวจหาการชนกันของ ID กับรีจิสทรีอื่นทั้งหมด การชนกันระหว่างรีจิสทรีจะมีได้เฉพาะเมื่อบัญชีผู้ใช้ที่จะถูกสร้าง ขึ้นนั้นมีชื่อเดียวกับบัญชีผู้ใช้ที่มีอยู่แล้วสำหรับประเภทคำสั่ง `mkuser id=123 foo` ถ้า ID ไม่ถูกระบุจากบรรทัดคำสั่ง บัญชีผู้ใช้ใหม่อาจไม่มีค่า ID เหมือนกับ บัญชีผู้ใช้ที่มีอยู่แล้วที่มีชื่อเหมือนกันในอีกรีจิสทรีหนึ่ง ตัวอย่าง *acct1* ที่มี ID 234 เป็นบัญชีผู้ใช้โลคัล เมื่อสร้าง บัญชีผู้ใช้ LDAP *acct1* ขึ้น `mkuser -R LDAP acct1` อาจ เลือก ID เฉพาะของ 235 สำหรับบัญชีผู้ใช้ LDAP ผลลัพธ์คือ *acct1* ที่มี ID 234 บนโลคัล และ *acct1* ที่มี 235 บน LDAP

หมายเหตุ: การตรวจพบการชนกันของ ID ในรีจิสทรีปลายทางต้องถูกบังคับการเปลี่ยนเสมอ โดยไม่คำนึงถึงแอ็ตทริบิวต์ **dist_uniqid**

ค่า **uniqbyname** ทำงานได้ดีกับรีจิสทรีสองค่า ด้วยรีจิสทรีมากกว่าสองค่า และเมื่อมีการชนกันของ ID อยู่แล้ว ระหว่างสองรีจิสทรี ลักษณะการทำงานของ **mkuser** (**mkgroup**) จะไม่ถูกระบุเมื่อสร้างบัญชีผู้ใช้ใหม่ในรีจิสทรีที่สามโดยใช้ ค่า ID ที่มีการชนกันนั้น การสร้างบัญชีผู้ใช้ใหม่อาจสำเร็จหรือ ล้มเหลวขึ้นอยู่กับลำดับของรีจิสทรีที่ถูกตรวจสอบ

ตัวอย่าง: สมมติระบบถูกตั้งค่าด้วยรีจิสทรีสามค่า: local, LDAP และ DCE บัญชีผู้ใช้ *acct1* มีอยู่แล้วใน LDAP และบัญชีผู้ใช้ *acct2* ใน DCE ทั้งคู่มี ID 234 เมื่อผู้ดูแลระบบรันคำสั่ง `mkuser -R files id=234 acct1` (`mkgroup -R files id=234 acct1`) เพื่อสร้างบัญชีผู้ใช้โลคัลที่มีค่า **uniqbyname** คำสั่ง **mkuser** (**mkgroup**) จะตรวจสอบกับรีจิสทรี LDAP เป็นอันดับแรก และพบว่า ID 234 ถูกนำไปใช้โดยบัญชีผู้ใช้ LDAP *acct1* เนื่องจากบัญชีผู้ใช้ที่สร้างมี ชื่อบัญชีผู้ใช้เหมือนกัน คำสั่ง **mkuser** (**mkgroup**) จึงสร้างบัญชีผู้ใช้โลคัล *acct1* ที่มี ID 234 ได้สำเร็จ ถ้าตรวจสอบรีจิสทรี DCE เป็นอันดับแรก คำสั่ง **mkuser** (**mkgroup**) จะพบว่า ID 234 ถูกนำไปใช้โดยบัญชีผู้ใช้ DCE *acct2* และ การสร้างบัญชีผู้ใช้โลคัล *acct1* จะล้มเหลว การตรวจ

หาการชนกันของ ID บังคับให้ต้องมีการใช้ ID ค่าเฉพาะระหว่างรีจิสทรีโลคัลและรีจิสทรีรีโมต หรือระหว่างรีจิสทรีรีโมตด้วยกัน ไม่มีการรับประกันสำหรับการใช้ค่าเฉพาะ ID ระหว่างบัญชีผู้ใช้ที่สร้างขึ้นใหม่บนรีจิสทรีรีโมต และผู้ใช้โลคัล ที่มีอยู่แล้วบนระบบอื่น ซึ่งใช้รีจิสทรีรีโมตเดียวกัน คำสั่ง **mkuser** (**mkgroup**) ข้ามรีจิสทรีรีโมตถ้ารีจิสทรีนั้นไม่สามารถเข้าถึงได้ในตอนที่รันคำสั่ง

บัญชีผู้ใช้ Root

บัญชีผู้ใช้ root มีการเข้าถึงโปรแกรม ไฟล์ และรีซอร์สทั้งหมดบนระบบ แบบไม่จำกัดโดยแท้จริง

บัญชีผู้ใช้ root เป็นผู้ใช้พิเศษในไฟล์ `/etc/passwd` ที่มี ID ผู้ใช้(UID) เป็น `0` และโดยทั่วไป ถูกกำหนดชื่อผู้ใช้เป็น *root* ไม่ใช่ชื่อผู้ใช้ที่ทำให้บัญชีผู้ใช้ root มีความพิเศษ แต่เป็นค่า UID ที่เป็น `0` นี้หมายความว่า ผู้ใช้ใดที่มี UID เป็น `0` ก็มีสิทธิ์พิเศษเหมือนกับ ผู้ใช้ root เช่นกัน รวมทั้งบัญชีผู้ใช้ root ถูกพิสูจน์ตัวตนเสมอด้วยวิธีของ ไฟล์การรักษาความปลอดภัยโลคัล

บัญชีผู้ใช้ root ควรมีรหัสผ่านเสมอ ซึ่งไม่ควรแบ่งใช้ร่วมกัน บัญชีผู้ใช้ root ควรถูกกำหนดรหัสผ่านในทันที่หลังจากติดตั้งระบบ เฉพาะผู้ดูแลระบบเท่านั้นที่ควรทราบรหัสผ่าน root ผู้ดูแลระบบ ควรทำหน้าที่เป็นผู้ใช้ root เท่านั้นเพื่อดำเนินการฟังก์ชันการดูแลจัดการ ระบบที่จำเป็นต้องมีสิทธิ์พิเศษ root สำหรับการดำเนินการอื่นทั้งหมด ผู้ดูแลควร กลับไปใช้บัญชีผู้ใช้ปกติของตน

ข้อควรสนใจ: การดำเนินการที่ทำเป็นประจำในฐานะผู้ใช้ root สามารถส่งผล ให้เกิดความเสียหายต่อระบบเนื่องจากบัญชีผู้ใช้ root แทนที่การป้องกันหลายๆ อย่างใน ระบบ

การปิดใช้งานล็อกอิน root โดยตรง:

วิธีการโจมตีทั่วไปของแฮกเกอร์ที่อาจมีคือการหารหัสผ่าน root

เพื่อหลีกเลี่ยงการโจมตีประเภทนี้ คุณสามารถปิดใช้งานการเข้าถึงโดยตรง ไปยัง root ID ของคุณและผู้ดูแลระบบของคุณต้องหาสิทธิพิเศษ root โดยใช้คำสั่ง **su –** นอกเหนือจากการอนุญาตให้คุณ ลบผู้ใช้ root ที่เป็นจุดของการโจมตี การจำกัดการเข้าถึงเป็น root โดยตรง อนุญาตให้คุณมอนิเตอร์ว่าผู้ใช้รายใดที่ได้รับสิทธิการเข้าถึงแบบ root รวมถึง เวลาของการดำเนิน คุณสามารถทำได้โดยการดูไฟล์ `/var/adm/sulog` อีกทางหนึ่งคือเปิดใช้การตรวจสอบระบบ ซึ่งจะรายงานให้ทราบถึงกิจกรรมประเภทนี้

ในการปิดใช้งานการเข้าถึงล็อกอินรีโมตสำหรับผู้ใช้ root ของคุณ แก้ไข ไฟล์ `/etc/security/user` ระบุ False เป็นค่า rlogin value บนรายการสำหรับ root

ก่อนที่คุณจะปิดใช้งานล็อกอิน root รีโมต ตรวจสอบและวางแผนสำหรับสถานการณ์ที่จะกันมิให้ผู้ดูแลระบบ ล็อกอินโดยใช้ ID ผู้ใช้ของผู้ที่มิใช่ root ตัวอย่าง ถ้าระบบไฟล์โฮมของ ผู้ใช้เต็ม ผู้ใช้จะไม่สามารถล็อกอิน ถ้าการล็อกอิน root แบบรีโมตถูกปิดใช้งาน และผู้ใช้ที่สามารถใช้คำสั่ง **su –** เพื่อเปลี่ยนเป็น root มีผู้ใช้ไฟล์โฮมเต็ม root ไม่เคยเข้าควบคุม ระบบ ปัญหานี้สามารถเลี่ยงได้โดยผู้ดูแลระบบ สร้างระบบไฟล์โฮมของตนเองที่มีขนาดใหญ่กว่าระบบไฟล์ของ ผู้ใช้โดยเฉลี่ย

บัญชีผู้ใช้

มีงานการดูแลความปลอดภัยหลายงาน สำหรับบัญชีผู้ใช้

แอ็ตทริบิวต์ผู้ใช้ที่แนะนำ:

การดูและผู้ใช้ประกอบด้วยการสร้างผู้ใช้และกลุ่มและการกำหนด แอ็ตทริบิวต์

แอ็ตทริบิวต์หลักของผู้ใช้คือวิธีที่ผู้ใช้ถูกตรวจสอบตัวตน ผู้ใช้ คือเอเจนต์หลักบนระบบ แอ็ตทริบิวต์ควบคุมสิทธิการเข้าถึง สภาวะแวดล้อม วิธีพิสูจน์ตัวตน ของผู้ใช้เช่นเดียวกับ วิธี เวลา และ สถานที่ที่บัญชีผู้ใช้ของผู้ใช้สามารถถูกเข้าถึง

กลุ่มเป็นคอลเล็กชันของผู้ใช้ซึ่งสามารถแบ่งใช้สิทธิการเข้าถึงเหมือนกันสำหรับรีซอร์สที่มีการป้องกัน กลุ่มมี ID และประกอบด้วยสมาชิกและผู้ดูแลระบบ ผู้สร้างกลุ่ม โดยปกติคือผู้ดูแลระบบคนแรก

แอ็ตทริบิวต์จำนวนมากสามารถถูกเซ็ต สำหรับแต่ละบัญชีผู้ใช้ รวมถึงแอ็ตทริบิวต์รหัสผ่านและล็อกอิน สำหรับรายการของแอ็ตทริบิวต์ที่กำหนดค่าได้ อ้างถึง “ภาพรวมระบบโควต้าดิสก์” ในหน้า 85 แนะนำแอ็ตทริบิวต์ดังต่อไปนี้:

- แต่ละผู้ใช้ควรมี ID ผู้ใช้ ที่ไม่ถูกแบ่งใช้กับผู้ใช้อื่น เครื่องมือป้องกันการปลอดภัยและความสามารถของผู้ใช้ ทำงานเฉพาะถ้าแต่ละ ผู้ใช้มี ID เฉพาะ
- กำหนดชื่อผู้ใช้ที่มีความหมายแก่ผู้ใช้บนระบบ ชื่อจริง ดีที่สุด เนื่องจากระบบจดหมายอิเล็กทรอนิกใช้ ID ผู้ใช้เพื่อเลเบลเมล์ขาเข้า
- เพิ่ม, เปลี่ยน, และลบผู้ใช้โดยใช้อินเตอร์เฟส SMIT แม้ว่า คุณสามารถดำเนินการกับงานเหล่านี้ทั้งหมดได้จากบรรทัดรับคำสั่ง, อินเตอร์เฟส SMIT ช่วยลดข้อผิดพลาดเล็กๆ น้อยๆ
- อย่าให้รหัสผ่านเริ่มต้นกับผู้ใช้จนกว่าผู้ใช้จะพร้อม ล็อกอินเข้าสู่ระบบ ถ้าฟิลด์รหัสผ่านถูกกำหนดเป็น * (เครื่องหมายดอกจัน) ในไฟล์ `/etc/passwd` ข้อมูลบัญชีผู้ใช้ถูกเก็บไว้ แต่ไม่สามารถล็อกอินกับบัญชีผู้ใช้นั้นได้
- อย่างเปลี่ยน ID ผู้ใช้ที่ระบบกำหนดซึ่งจำเป็นต่อการทำงานอย่างถูกต้อง ของระบบ ID ผู้ใช้ที่ระบบกำหนดแสดงในไฟล์ `/etc/passwd`
- โดยทั่วไป อย่างเซ็ตพารามิเตอร์ *admin* เป็น `true` สำหรับ ID ผู้ใช้ เฉพาะผู้ใช้ root สามารถเปลี่ยนแอ็ตทริบิวต์สำหรับผู้ใช้ที่มี `admin=true` เซ็ต ไว้ในไฟล์ `/etc/security/user`

ระบบปฏิบัติการสนับสนุนแอ็ตทริบิวต์ผู้ใช้มาตรฐาน ที่พบในไฟล์ `/etc/passwd` และ `/etc/system/group` เช่น:

ข้อมูลการพิสูจน์ตัวตน
ระบุรหัสผ่าน

Credentials
ระบุ identifier กลุ่มหลัก และ ID กลุ่มเสริมของผู้ใช้

สภาวะแวดล้อม
ระบุสภาวะแวดล้อม home หรือ shell

การจำกัดความยาวชื่อผู้ใช้และชื่อกลุ่ม:

คุณสามารถตั้งค่าและเรียกข้อมูลการจำกัดความยาวชื่อผู้ใช้และกลุ่ม

ค่าดีฟอลต์พารามิเตอร์การจำกัดความยาวชื่อผู้ใช้และชื่อกลุ่มคือ 9 อักขระ สำหรับ AIX 5.3 และ รุ่นสูงกว่า คุณสามารถเพิ่มการจำกัดความยาวชื่อผู้ใช้และชื่อกลุ่มจาก 9 อักขระ เป็น 256 อักขระ เนื่องจากพารามิเตอร์การจำกัดความยาวชื่อผู้ใช้และชื่อกลุ่ม รวมอักขระ NULL ที่ปิดท้าย ความยาวชื่อที่ถูกต้องจริงคือ จาก 8 อักขระถึง 255 อักขระ

การจำกัดความยาวชื่อผู้ใช้และชื่อกลุ่ม ถูกระบุด้วยพารามิเตอร์คอนฟิกูเรชันของระบบ `v_max_logname` สำหรับอุปกรณ์ `sys0` คุณสามารถเปลี่ยนหรือเรียกข้อมูล ค่าพารามิเตอร์ `v_max_logname` จากฐานข้อมูล kernel หรือ ODM ค่าพารามิเตอร์ในเคอร์เนลเป็นค่าที่ระบบใช้ ขณะรัน ค่าพารามิเตอร์ในฐานข้อมูล ODM คือค่าที่ระบบใช้ หลังจากการรีสตาร์ทครั้งต่อไป

หมายเหตุ: การทำงานที่ไม่คาดคิดอาจเกิดขึ้น ถ้า คุณลดการจำกัดความยาวชื่อผู้ใช้และชื่อกลุ่ม หลังจากทำการเพิ่ม ชื่อ ผู้ใช้และกลุ่มที่คุณสร้างด้วยการจำกัดที่ค่ามากกว่าอาจยังคงมี อยู่ในระบบ

การเรียกข้อมูลค่าจำกัดความยาวชื่อผู้ใช้และกลุ่มจากฐานข้อมูล ODM:

คุณสามารถใช้คำสั่งหรือรูทีนย่อยเพื่อเรียกข้อมูลพารามิเตอร์ **v_max_logname**

คุณสามารถใช้คำสั่ง **lsattr** เพื่อเรียกข้อมูล พารามิเตอร์ **v_max_logname** ในฐานข้อมูล ODM คำสั่ง **lsattr** จะแสดงพารามิเตอร์ **v_max_logname** เป็นแอ็ตทริบิวต์ max_logname

สำหรับข้อมูลเพิ่มเติม ดูที่คำสั่ง **lsattr** ใน *ข้อมูลอ้างอิงคำสั่ง วอลุ่ม 3*

ตัวอย่างต่อไปนี้แสดงวิธีใช้คำสั่ง **lsattr** เพื่อเรียกข้อมูลแอ็ตทริบิวต์ max_logname:

```
$ lsattr -El sys0
SW_dist_intr    false           Enable SW distribution of interrupts             True
autorestart     true            Automatically REBOOT system after a crash        True
boottype        disk            N/A                                              False
capacity_inc    1.00            Processor capacity increment                     False
capped          true            Partition is capped                              False
conslogin       enable          System Console Login                             False
cpuguard        enable          CPU Guard                                        True
dedicated       true            Partition is dedicated                           False
ent_capacity    4.00            Entitled processor capacity                      False
frequency       93750000        System Bus Frequency                             False
fullcore        false           Enable full CORE dump                            True
fwversion       IBM,SPH01316    Firmware version and revision levels             False
iostat          false           Continuously maintain DISK I/O history           True
keylock         normal          State of system keylock at boot time             False
max_capacity    4.00            Maximum potential processor capacity             False
max_logname     20              Maximum login name length at boot time           True
maxbuf          20              Maximum number of pages in block I/O BUFFER CACHE True
maxmbuf         0               Maximum Kbytes of real memory allowed for MBUFS  True
maxpout         0               HIGH water mark for pending write I/Os per file  True
maxuproc        128             Maximum number of PROCESSES allowed per user     True
min_capacity    1.00            Minimum potential processor capacity             False
minpout         0               LOW water mark for pending write I/Os per file   True
modelname       IBM,7044-270    Machine name                                     False
ncargs          6               ARG/ENV list size in 4K byte blocks              True
pre430core      false           Use pre-430 style CORE dump                      True
pre520tune      disable         Pre-520 tuning compatibility mode                True
realmem         3145728         Amount of usable physical memory in Kbytes       False
rtasversion     1               Open Firmware RTAS version                       False
sec_flags       0               Security Flags                                   True
sed_config      select          Stack Execution Disable (SED) Mode               True
systemid        IBM,0110B5F5F   Hardware system identifier                       False
variable_weight 0               Variable processor capacity weight               False
$
```

การเรียกข้อมูลค่าจำกัดความยาวชื่อผู้ใช้และกลุ่มจากเคอร์เนล:

คุณสามารถใช้คำสั่งและรูทีนย่อยเพื่อเรียกข้อมูลพารามิเตอร์ **v_max_logname** จากเคอร์เนล

การใช้คำสั่ง getconf

คุณ สามารถใช้คำสั่ง **getconf** ที่มีพารามิเตอร์ LOGIN_NAME_MAX เพื่อเรียกข้อมูลค่าจำกัดความยาวชื่อผู้ใช้และกลุ่มในเคอร์เนล เอาต์พุตคำสั่ง **getconf** มีอักขระ NULL ปิดท้าย

ตัวอย่าง ต่อไปนี้แสดงวิธีใช้คำสั่ง **getconf** เพื่อเรียกข้อมูล ค่าจำกัดชื่อผู้ใช้และกลุ่มจากเคอร์เนล:

```
$ getconf LOGIN_NAME_MAX
20
$
```

การใช้รูทีนย่อย sysconf

คุณ สามารถใช้รูทีนย่อย sysconf ที่มีพารามิเตอร์ _SC_LOGIN_NAME_MAX เพื่อเรียกข้อมูลค่าจำกัดความยาวชื่อผู้ใช้และกลุ่มในเคอร์เนล

ตัวอย่าง ต่อไปนี้แสดงวิธีใช้รูทีนย่อย sysconf เพื่อเรียกข้อมูลค่าจำกัดความยาวชื่อผู้ใช้และกลุ่มจากเคอร์เนล:

```
#include <unistd.h>
main()
{
       long len;

       len = sysconf(_SC_LOGIN_NAME_MAX);

       printf("The name length limit is %d\n", len);
}
```

การใช้รูทีนย่อย sys_parm

คุณ สามารถใช้รูทีนย่อย sys_parm ที่มีพารามิเตอร์ SYSP_V_MAX_LOGNAME เพื่อเรียกข้อมูลค่าจำกัดความยาวชื่อผู้ใช้ปัจจุบันในเคอร์เนล

ตัวอย่าง ต่อไปนี้แสดงวิธีใช้รูทีนย่อย sys_parm เพื่อเรียกข้อมูลค่าจำกัดความยาวชื่อผู้ใช้จากเคอร์เนล:

```
#include <sys/types.h>
#include <sys/var.h>
#include <errno.h>
main()
{
   int rc;
   struct vario myvar;

   rc = sys_parm (SYSP_GET, SYSP_V_MAX_LOGNAME, &myvar);

   if (!rc)
      printf("Max_login_name = %d\n", myvar.v.v_max_logname.value);
   else
      printf("sys_parm() failed rc = %d, errno = %d\n", rc, errno);
}
```

การเปลี่ยนกลุ่มผู้ใช้และความยาวชื่อที่จำกัดในฐานข้อมูล ODM:

คุณสามารถตั้งค่าจำกัดความยาวชื่อผู้ใช้และกลุ่มใน เคอร์เนลได้เฉพาะระหว่างช่วงบูตระบบใหม่ คุณสามารถเปลี่ยนค่าในฐานข้อมูล ODM ได้โดยใช้คำสั่ง **chdev** การเปลี่ยนแปลง มีผลหลังจากระบบเริ่มทำต่อในครั้งต่อไป

ตัวอย่างต่อไปนี้แสดงวิธีใช้คำสั่ง **chdev** เพื่อเปลี่ยนพารามิเตอร์ `v_max_logname` ในฐานข้อมูล ODM:

```
$ chdev -l sys0 -a max_logname=30
sys0 changed
$
```

การควบคุมบัญชีผู้ใช้:

บัญชีผู้ใช้มีแอ็ตทริบิวต์ที่สามารถถูกเปลี่ยนแปลงได้

แต่ผลบัญชีผู้ใช้มีชุดของแอ็ตทริบิวต์ที่เชื่อมโยงกัน แอ็ตทริบิวต์เหล่านี้ ถูกสร้างจากค่าดีฟอลต์เมื่อผู้ใช้ถูกสร้างโดยใช้คำสั่ง **mkuser** แอ็ตทริบิวต์ สามารถถูกแก้ไขโดยใช้คำสั่ง **chuser** ต่อไปนี้เป็นแอ็ตทริบิวต์ผู้ใช้ที่ควบคุมล็อกอินและไม่เกี่ยวกับคุณภาพของรหัสผ่าน:

account_locked
ถ้าบัญชีผู้ใช้ต้องถูกล็อก แอ็ตทริบิวต์นี้สามารถถูกตั้งให้เป็น True; ค่าดีฟอลต์คือ False

admin ถ้าตั้งค่าเป็น True ผู้ใช้นี้ไม่สามารถเปลี่ยนรหัสผ่านได้ เฉพาะ ผู้ดูแลระบบเท่านั้นที่สามารถเปลี่ยนได้

admgroups
แสดงกลุ่มซึ่งผู้ใช้นี้มีสิทธิ การดูแลระบบ กลุ่มดังกล่าวนี้ ผู้ใช้สามารถเพิ่มหรือลบสมาชิกได้

auth1 วิธีการพิสูจน์ตัวตนที่ถูกใช้เพื่อให้สิทธิการเข้าถึงแก่ผู้ใช้ โดยปกติ จะถูกตั้งค่าเป็น `SYSTEM` ซึ่งจะใช้ เมธอดที่ใหม่กว่า

หมายเหตุ: แอ็ตทริบิวต์ **auth1** ไม่ได้รับการยอมรับ และไม่ควรใช้

auth2 เมธอดที่รันหลังจากผู้ใช้ได้รับการพิสูจน์ตัวตนตามข้อมูล ที่ระบุใน **auth1** ไม่สามารถบล็อกการเข้าถึงระบบ โดยปกติถูกตั้งค่าเป็น `NONE`

หมายเหตุ: แอ็ตทริบิวต์ **auth2** ไม่ได้รับการยอมรับ และไม่ควรใช้

daemon
พารามิเตอร์บูลีนนี้ระบุว่าผู้ใช้ได้รับอนุญาตให้ สตาร์ท daemons หรือ subsystems ด้วยคำสั่ง **startsrc** หรือไม่ และยังจำกัดการใช้ความสามารถ cron และ at

login ระบุว่าผู้ใช้นี้ได้รับอนุญาตให้ล็อกอินหรือไม่ การล็อกอิน สำเร็จรีเซ็ตแอ็ตทริบิวต์ **unsuccessful_login_count** ให้มีค่าเป็น `0` (จากรูทีนย่อย **loginsuccess**)

logintimes
จำกัดเวลาที่ผู้ใช้สามารถล็อกอินได้ ตัวอย่างเช่น ผู้ใช้อาจถูกจำกัด ให้เข้าถึงระบบเฉพาะระหว่างเวลา ทำงานปกติเท่านั้น

registry
ระบุรีจิสทรีผู้ใช้ สามารถถูกใช้เพื่อบอกระบบเกี่ยวกับ รีจิสทรีทางเลือกสำหรับข้อมูลผู้ใช้ เช่น NIS, LDAP, หรือ Kerberos

rlogin ระบุผู้ใช้ที่ระบุเฉพาะสามารถล็อกอินโดยใช้คำสั่ง **rlogin** หรือ **telnet** แอ็ตทริบิวต์ rlogin ควบคุมล็อกอินแบบรีโมตเท่านั้น สำหรับข้อมูลเกี่ยวกับการควบคุมความสามารถ ในการรันคำสั่งแบบรีโมต, โปรดดู rcmds

su ระบุว่าผู้ใช้อื่นสามารถสลับ ID นี้กับคำสั่ง **su**

sugroups
ระบุกลุ่มซึ่งได้รับอนุญาตให้สลับ ID ผู้ใช้นี้

ttys จำกัดจำนวนบัญชีผู้ใช้ต่อพื้นที่รักษาความปลอดภัยแบบฟิสิคัล

expires จัดการบัญชีผู้ใช้ student หรือ guest; และยังสามารถถูกใช้เพื่อ ปิดบัญชีผู้ใช้ชั่วคราว

loginretries
ระบุจำนวนสูงสุดของความพยายามล็อกอินที่ล้มเหลวอย่างต่อเนื่อง ก่อนที่ ID ผู้ใช้ถูกล็อกโดยระบบ ความพยายามที่ล้มเหลว ถูกบันทึกในไฟล์ /etc/security/lastlog

umask ระบุ **umask** เริ่มต้นสำหรับผู้ใช้

rcmds ระบผู้ใช้ที่ระบุเฉพาะสามารถรันคำสั่งโดยใช้คำสั่ง **rsh** หรือคำสั่ง **rexec** ค่า allow ระบุว่าคุณสามารถรันคำสั่ง โดยใช้คำสั่ง **rsh** และ **rexec** ค่า deny บ่งชี้ว่าคุณไม่สามารถรันคำสั่งแบบรีโมตได้ ค่า hostlogincontrol บ่งชี้ว่าการรันคำสั่งแบบรีโมตถูกควบคุมโดยแอ็ตทริบิวต์ **hostallowedlogin** และ **hostsdeniedlogin** สำหรับข้อมูลเกี่ยวกับการควบคุมล็อกอินแบบรีโมต, โปรดดูแอ็ตทริบิวต์ rlogin

hostallowedlogin
ระบุโฮสต์ซึ่งอนุญาตให้ผู้ใช้ ล็อกอิน แอ็ตทริบิวต์นี้มีเป้าหมายเพื่อใช้ในสภาวะแวดล้อมที่เป็นเน็ตเวิร์ก ซึ่งแอ็ตทริบิวต์ผู้ใช้ถูกแบ่งใช้โดยหลายโฮสต์

hostsdeniedlogin
ระบุโฮสต์ซึ่งไม่อนุญาตให้ ผู้ใช้ล็อกอิน แอ็ตทริบิวต์นี้มีเป้าหมายเพื่อใช้ในสภาวะแวดล้อมที่เป็นเน็ตเวิร์ก ซึ่งแอ็ตทริบิวต์ผู้ใช้ถูกแบ่งใช้โดยหลายโฮสต์

maxulogs
ระบุจำนวนสูงสุดของล็อกอินต่อผู้ใช้ ถ้าผู้ใช้มีการล็อกอินถึงค่าสูงสุดของการล็อกอินที่อนุญาต การล็อกอิน จะถูกปฏิเสธ

ชุดสมบูรณ์ของแอ็ตทริบิวต์ผู้ใช้ถูกกำหนดในไฟล์ /etc/security/user, /etc/security/limits, /etc/security/audit/config และ /etc/security/lastlog ค่าดีฟอลต์สำหรับการสร้างผู้ใช้ด้วยคำสั่ง **mkuser** ถูกระบุในไฟล์ /usr/lib/security/mkuser.default เฉพาะตัวเลือกที่แทนที่ค่าดีฟอลต์ในดีฟอลต์ stanzas ของไฟล์ /etc/security/user และ /etc/security/limits เช่นเดียวกับคลาสการตรวจสอบ ต้องถูกระบุในไฟล์ mkuser.default บางส่วนของแอ็ตทริบิวต์เหล่านี้ควบคุมวิธีที่ผู้ใช้สามารถล็อกอิน และสามารถ ถูกกำหนดค่าให้ล็อกบัญชีผู้ใช้ (ป้องกันไม่ให้ล็อกอิน) อัตโนมัติภายใต้เงื่อนไขที่กำหนด

หลังจากบัญชีผู้ใช้ถูกล็อกโดยระบบเนื่องจาก จำนวนของการล็อกอินไม่สำเร็จ ผู้ใช้จะไม่สามารถล็อกอิน จนกว่าผู้ดูแลระบบจะรีเซ็ตแอ็ตทริบิวต์ผู้ใช้ **unsuccessful_login_count** ในไฟล์ /etc/security/lastlog ให้มีค่าน้อยกว่า ค่าการล็อกอินซ้ำ ซึ่งทำได้โดยใช้คำสั่ง **chsec** ดังนี้:

```
chsec -f /etc/security/lastlog -s username -a
unsuccessful_login_count=0
```

ค่าดีฟอลต์สามารถถูกเปลี่ยนโดยใช้คำสั่ง **chsec** เพื่อแก้ไขดีฟอลต์ stanza ในไฟล์ความปลอดภัยที่เหมาะสม เช่นไฟล์ /etc/security/user หรือ /etc/security/limits ค่าดีฟอลต์เหล่านี้จำนวนมากถูกกำหนดให้เป็นการทำงานมาตรฐาน เมื่อต้องการ ระบุแอ็ตทริบิวต์ที่ถูกเซ็ตทุกครั้งที่ผู้ใช้ใหม่ถูกสร้าง ให้เปลี่ยนรายการ *user* ใน /usr/lib/security/mkuser.default

สำหรับข้อมูลเกี่ยวกับแอ็ตทริบิวต์รหัสผ่านผู้ใช้เพิ่มเติม อ้างถึง “รหัสผ่าน” ในหน้า 72

คำสั่งที่เกี่ยวกับการล็อกอินที่ได้รับผลจากแอ็ตทริบิวต์ user

ตารางดังต่อไปนี้ แสดงแอ็ตทริบิวต์ที่ควบคุมล็อกอินและคำสั่ง ที่มีผล

แอ็ตทริบิวต์ผู้ใช้	คำสั่ง
account_locked	**rexec, rsh, rcp, ssh, scp, rlogin, telnet, ftp, login**
login	มีผลเฉพาะจากคอนโซล ค่า ของแอ็ตทริบิวต์ **login** ไม่มีผลกับคำสั่งรีโมตล็อกอิน คำสั่งรีโมตเชลล์ หรือคำสั่งทำสำเนาแบบรีโมต **rexec, rsh, rcp, ssh, scp, rlogin, telnet**, และ **ftp**)
logintimes	**rexec, rsh, rcp, ssh, scp, rlogin, telnet, ftp, login**
rlogin	มีผลเฉพาะคำสั่งรีโมตล็อกอิน บาง คำสั่งรีโมตเชลล์ และบางคำสั่งการทำสำเนาแบบรีโมต (**ssh, scp, rlogin,** และ **telnet**)
loginretries	**rexec, rsh, rcp, ssh, scp, rlogin, telnet, ftp, login**
/etc/nologin	**rexec, rsh, rcp, ssh, scp, rlogin, telnet, ftp, login**
rcmds=deny	**rexec, rsh, rcp, ssh, scp**
rcmds=hostlogincontrol and hostsdeniedlogin=<target_hosts>	**rexec, rsh, rcp, ssh, scp, rlogin, telnet, ftp, login**
ttys = !REXEC, !RSH	**rexec, rsh, rcp, ssh, scp, rlogin, telnet, ftp, login**
ttys = !REXEC, !RSH, /dev/pts	**rexec, rsh**
ttys = !REXEC, !RSH, ALL	**rexec, rsh**
expires	**rexec, rsh, rcp, ssh, scp, rlogin, telnet, ftp, login**

หมายเหตุ: **rsh** ไม่อนุญาตให้ใช้คำสั่งรีโมต เท่านั้น รีโมตล็อกอินสามารถใช้ได้

ข้อมูลที่เกี่ยวข้อง:

loginsuccess subroutine

คำสั่ง **rexec**

คำสั่ง **rsh**

คำสั่ง **startsrc**

คำสั่ง **su**

ล็อกอิน ID ผู้ใช้:

ระบบปฏิบัติการระบุผู้ใช้โดยล็อกอิน ID ผู้ใช้

ล็อกอิน ID ผู้ใช้ทำให้ระบบสามารถติดตามการดำเนินการของผู้ใช้ทั้งหมด กับซอร์ส หลังจากผู้ใช้ล็อกอินเข้าสู่ระบบ แต่ก่อนรัน โปรแกรมผู้ใช้แรก ระบบเซ็ต login ID ของกระบวนการให้กับ ID ผู้ใช้ ที่พบในฐานข้อมูลผู้ใช้ กระบวนการที่ตามมาทั้งหมดระหว่างเซสชันล็อกอิน ถูกแท็ก (tag) ด้วย ID นี้ แท็กเหล่านี้จัดเตรียมการติดตามกิจกรรมทั้งหมดที่ ดำเนินการโดยล็อกอิน ID ผู้ใช้ ผู้ใช้สามารถรีเซ็ต ID ผู้ใช้ที่มีผล ID ผู้ใช้จริง กลุ่ม ID ที่มีผล กลุ่ม ID จริงและ กลุ่ม ID เสริมระหว่าง เซสชัน แต่ไม่สามารถเปลี่ยนล็อกอิน ID ผู้ใช้ได้

เพิ่มประสิทธิภาพการรักษาความปลอดภัยผู้ใช้ด้วย Access Control Lists:

เพื่อบรรลุระดับความปลอดภัยที่เหมาะสมในระบบของคุณ พัฒนา นโยบายความปลอดภัยที่สอดคล้องกันในการจัดการบัญชีผู้ใช้ กลไกความปลอดภัยที่ใช้โดยทั่วไปมากที่สุดคือ access control list (ACL)

สำหรับข้อมูลเกี่ยวกับ ACL และการพัฒนานโยบาย ความปลอดภัย ดูที่ “Access Control Lists” ในหน้า 136

ตัวแปรสภาวะแวดล้อม PATH:

ตัวแปรสภาวะแวดล้อม **PATH** เป็นตัวควบคุมความปลอดภัย ที่สำคัญ ซึ่งระบุไดเร็กทอรีที่จะถูกค้นหาเพื่อหาคำสั่ง

ค่า systemwide **PATH** ดีฟอลต์ถูกระบุในไฟล์ `/etc/profile` และแต่ละผู้ใช้โดยปกติมีค่า **PATH** ในไฟล์ `$HOME/.profile` ของผู้ใช้ ค่า **PATH** ในไฟล์ `.profile` แทนที่ ค่า systemwide **PATH** หรือเพิ่มไดเร็กทอรี เข้าไป

การเปลี่ยนแปลงตัวแปรสภาวะแวดล้อม **PATH** ที่ไม่ได้รับอนุญาตสามารถ ทำให้ผู้ใช้บนระบบ ″spoof″ ผู้ใช้อื่น (รวมทั้งผู้ใช้ root) โปรแกรม *Spoofing* (เรียกอีกอย่างว่าโปรแกรม *ม้าโทรจัน*) แทนที่คำสั่งระบบแล้วจับข้อมูลทีสำคัญสำหรับ คำสั่งนั้น เช่นรหัสผ่านของผู้ใช้

ตัวอย่างเช่น สมมุติว่าผู้ใช้เปลี่ยนค่า **PATH** เพื่อให้ ระบบค้นหาไดเร็กทอรี `/tmp` ก่อน เมื่อคำสั่งถูกรัน จากนั้นผู้ใช้นำโปรแกรมไปไว้ในไดเร็กทอรี `/tmp` โปรแกรมชื่อ **su** ซึ่งถามรหัสผ่านเหมือนกับคำสั่ง **su** จากนั้นโปรแกรม **/tmp/su** ส่งเมลรหัสผ่านของ root ไปที่ผู้ใช้ แล้วเรียกคำสั่ง **su** จริงก่อนจบการทำงาน ใน สถานการณ์นี้ ผู้ใช้ root ซึ่งใช้คำสั่ง **su** จะเปิดเผยรหัสผ่านของ root และผู้ใช้จะไม่ทราบว่ามีการเปิดเผยข้อมูล

เพื่อป้องกันปัญหากับตัวแปรสภาวะแวดล้อม **PATH** สำหรับผู้ดูแลระบบและผู้ใช้ให้ทำดังต่อไปนี้:

- เมื่อมีข้อสงสัย ให้ระบุชื่อพาธแบบเต็ม ถ้ามีการระบุชื่อพาธแบบเต็ม ตัวแปรสภาวะแวดล้อม **PATH** จะถูกข้าม
- อย่าระบุไดเร็กทอรีปัจจุบัน (ระบุโดย . (จุด)) ในค่า **PATH** ที่ระบุสำหรับผู้ใช้ root อย่าอนุญาต ให้ระบุไดเร็กทอรีปัจจุบันใน `/etc/profile`
- ผู้ใช้ root ควรมีค่ากำหนด **PATH** ของตัวเอง ในไฟล์ `.profile` ส่วนตัว โดยปกติ ค่ากำหนด ใน `/etc/profile` แสดงมาตรฐาน ขั้นต่ำสำหรับผู้ใช้ทั้งหมด ซึ่งผู้ใช้ root อาจจำเป็นต้องใช้ไดเร็กทอรี มากขึ้นหรือน้อยลงกว่าค่าดีฟอลต์
- เตือนผู้ใช้อื่นไม่ให้เปลี่ยนไฟล์ `.profile` ของพวกเขา โดยไม่ปรึกษาผู้ดูแลระบบ หรือมิฉะนั้น ผู้ใช้ปกติอาจ ทำการเปลี่ยนแปลงให้มีการอนุญาตการเข้าถึงที่ไม่ตั้งใจ ไฟล์ `.profile` ของผู้ใช้ ควรมีสิทธิกำหนดไว้ที่ 740
- ผู้ดูแลระบบไม่ควรใช้คำสั่ง **su** เพื่อรับ root privilege จากเซสชันผู้ใช้ เนื่องจากค่า **PATH** ของผู้ใช้ ที่รบะในไฟล์ `.profile` มีผล ผู้ใช้สามารถตั้งค่าไฟล์ `.profile` ของพวกเขาเอง ผู้ดูแลระบบ ควรล็อกอินที่เครื่องของผู้ใช้ในฐานะ root หรือ ใช้ ID ของพวกเขาเองแล้วใช้คำสั่งดังต่อไปนี้:

  ```
  /usr/bin/su - root
  ```

  ซึ่ง ประกันว่าสภาวะแวดล้อมของ root ถูกใช้ระหว่างเซสชัน ถ้า ผู้ดูแลระบบไม่ได้ดำเนินการในฐานะ root ในเซสชันผู้ใช้อื่น ผู้ดูแลระบบควรระบุชื่อพาธแบบเต็มผ่าน เซสชัน

- ป้องกันตัวแปรสภาวะแวดล้อม input field separator (**IFS**) จากการถูกเปลี่ยนแปลงในไฟล์ `/etc/profile` ตัวแปรสภาวะแวดล้อม **IFS** ในไฟล์ `.profile` สามารถถูกใช้เพื่อเปลี่ยนค่า **PATH**

การใช้ secldapclntd daemon:

secldapclntd daemon จัดการการเชื่อมต่อกับ เซิร์ฟเวอร์ LDAP แบบไดนามิก

เมื่อเริ่มทำงาน **secldapclntd** daemon เชื่อมต่อกับเซิร์ฟเวอร์ที่กำหนด ในไฟล์ **/etc/security/ldap/ldap.cfg** (หนึ่งการเชื่อมต่อต่อเซิร์ฟเวอร์ LDAP) ต่อมา ถ้า **secldapclntd** daemon พบว่าการเชื่อมต่อ LDAP จำกัดการร้องขอการประมวลผล LDAP, daemon จะสร้างการเชื่อมต่อ อื่นโดยอัตโนมัติกับเซิร์ฟเวอร์ LDAP ปัจจุบัน กระบวนการนี้ดำเนินไปจนกว่า จะถึงจำนวนครั้งสูงสุดของการเชื่อมต่อที่กำหนดไว้ หลังจากถึงจำนวนการเชื่อมต่อสูงสุด จะไม่มีการเพิ่มการเชื่อมต่อใหม่

secldapclntd daemon จะตรวจสอบการเชื่อมต่อกับ เซิร์ฟเวอร์ LDAP ปัจจุบันเป็นระยะ ถ้ามีการเชื่อมต่ออื่นที่ไม่ใช่การเชื่อมต่อแรก ว่างลงถึงเวลาที่กำหนดไว้ daemon จะปิดการเชื่อมต่อนั้น

ตัวแปร connectionsperserver ในไฟล์ **/etc/security/ldap/ldap.cfg** ถูกใช้เป็ตจำนวนการเชื่อมต่อสูงสุด อย่างไรก็ตาม ถ้าตัวแปร connectionsperserver มากกว่าตัวแปร numberofthread, **secldapclntd** daemon เซ็ตค่า connectionsperserver เป็นค่า numberofthread ค่าที่ใช้ได้สำหรับ ตัวแปร connectionsperserver คือ 1 ถึง 100 ค่าดีฟอลต์คือ 10 (`connectionsperserver: 10`)

ตัวแปร connectionmissratio ในไฟล์ **/etc/security/ldap/ldap.cfg** เซ็ตเกณฑ์สำหรับสร้างการเชื่อมต่อ LDAP ใหม่ ตัวแปร connectionmissratio เป็นเปอร์เซ็นต์ของการดำเนินการที่ล้มเหลวในการรับการเชื่อมต่อ LDAP (handle-miss) ระหว่างการดำเนินการครั้งแรก ถ้าจำนวนความพยายามที่ไม่สำเร็จ มากกว่าตัวแปร connectionmissratio, **secldapclntd** daemon เพิ่มเคียวรี LDAP โดยสร้างการเชื่อมต่อ LDAP ใหม่ (ไม่เกิน จำนวนการเชื่อมต่อที่กำหนดในตัวแปร connectionsperserver) ค่า ที่ใช้ได้สำหรับตัวแปร connectionmissratio คือ 10 ถึง 90 ค่าดีฟอลต์ คือ 50 (`connectionmissratio: 50`)

ตัวแปร connectiontimeout ในไฟล์ **/etc/security/ldap/ldap.cfg** ถูกใช้เป็นระยะเวลาที่การเชื่อมต่อยังคงว่างอยู่ ก่อนที่จะถูกปิด โดย **secldapclntd** daemon ค่าที่ถูกต้องสำหรับตัวแปร connectiontimeout คือ 5 วินาทีหรือมากกว่านั้น (ไม่มีการจำกัดค่าสูงสุด) ค่าดีฟอลต์คือ 300 วินาที (`connectiontimeout: 300`)

การตั้งค่า FTP แบบไม่ระบุชื่อด้วยบัญชีผู้ใช้ที่ปลอดภัย

คุณสามารถตั้งค่า FTP แบบไม่ระบุชื่อด้วยบัญชีผู้ใช้ที่ปลอดภัย

สถานการณ์นี้ตั้งค่า FTP แบบไม่ระบุชื่อด้วยบัญชีผู้ใช้ ที่ปลอดภัย โดยใช้ส่วนการติดต่อบรรทัดคำสั่งและสคริปต์

1. ตรวจสอบว่าชุดไฟล์ `bos.net.tcp.client` ถูกติดตั้งบนระบบของคุณ โดยการพิมพ์คำสั่งต่อไปนี้:

   ```
   lslpp -L | grep bos.net.tcp.client
   ```

   ถ้าคุณไม่ได้รับเอาต์พุต แสดงว่ายังไม่ได้ติดตั้งชุดไฟล์ สำหรับ คำแนะนำวิธีติดตั้ง ดูที่ *การติดตั้งและการย้าย*
2. ด้วยสิทธิ root เปลี่ยนเป็นไดเร็กทอรี `/usr/samples/tcpip` ตัวอย่าง:

   ```
   cd /usr/samples/tcpip
   ```
3. ในการตั้งค่าบัญชีผู้ใช้ รันสคริปต์ต่อไปนี้:

   ```
   ./anon.ftp
   ```

4. เมื่อได้รับพร้อมต์ Are you sure you want to modify /home/ftp? พิมพ์ yes เอาต์พุตที่คล้าย กับที่แสดงต่อไปนี้จะแสดงให้เห็น:

```
Added user anonymous.
Made /home/ftp/bin directory.
Made /home/ftp/etc directory.
Made /home/ftp/pub directory.
Made /home/ftp/lib directory.
Made /home/ftp/dev/null entry.
Made /home/ftp/usr/lpp/msg/en_US directory.
```

5. เปลี่ยนเป็นไดเร็กทอรี /home/ftp ตัวอย่าง:

```
cd /home/ftp
```

6. สร้างไดเร็กทอรีย่อย home โดยการพิมพ์:

```
mkdir home
```

7. เปลี่ยนสิทธิของไดเร็กทอรี /home/ftp/home เป็น drwxr-xr-x โดยการพิมพ์:

```
chmod 755 home
```

8. เปลี่ยนเป็นไดเร็กทอรี /home/ftp/etc โดยการพิมพ์:

```
cd /home/ftp/etc
```

9. สร้างไดเร็กทอรีย่อย objrepos โดยการพิมพ์:

```
mkdir objrepos
```

10. เปลี่ยนสิทธิของไดเร็กทอรี /home/ftp/etc/objrepos เป็น drwxrwxr-x โดยการพิมพ์:

```
chmod 775 objrepos
```

11. เปลี่ยนเจ้าของและกลุ่มของไดเร็กทอรี /home/ftp/etc/objrepos เป็นผู้ใช้ root และกลุ่มระบบ โดยการพิมพ์:

```
chown root:system objrepos
```

12. สร้างไดเร็กทอรีย่อย security โดยการพิมพ์

```
mkdir security
```

13. เปลี่ยนสิทธิของไดเร็กทอรี /home/ftp/etc/security เป็น drwxr-x--- โดยการพิมพ์:

```
chmod 750 security
```

14. เปลี่ยนเจ้าของและกลุ่มของไดเร็กทอรี /home/ftp/etc/security เป็นผู้ใช้ root และกลุ่ม security โดยการพิมพ์:

```
chown root:security security
```

15. เปลี่ยนเป็นไดเร็กทอรี /home/ftp/etc/security โดยการพิมพ์:

```
cd security
```

16. เพิ่มผู้ใช้โดยการพิมพ์พาธด่วน SMIT ต่อไปนี้:

```
smit mkuser
```

ในสถานการณ์นี้ เราจะเพิ่มผู้ใช้ชื่อ test

17. ในฟิลด์ SMIT ป้อนค่าต่อไปนี้:

```
User NAME                                       [test]
ADMINISTRATIVE USER?                             true
Primary GROUP                                   [staff]
```

```
Group SET                                       [staff]
Another user can SU TO USER?                     true
HOME directory                                  [/home/test]
```

หลังจากคุณป้อนการเปลี่ยนแปลงของคุณ กด Enter เพื่อสร้างผู้ใช้ หลังจากกระบวนการ SMIT เสร็จสมบูรณ์ ให้ออกจาก SMIT

18. สร้างรหัสผ่านสำหรับผู้ใช้นี้ด้วย คำสั่งต่อไปนี้:

    ```
    passwd test
    ```

    เมื่อได้รับพร้อมต์ ให้ป้อนรหัสผ่านที่ต้องการ คุณต้องป้อน รหัสผ่านใหม่ครั้งที่สองเพื่อยืนยัน
19. เปลี่ยนเป็นไดเร็กทอรี /home/ftp/etc โดยการพิมพ์

    ```
    cd /home/ftp/etc
    ```
20. ทำสำเนาไฟล์ /etc/passwd ไปยังไฟล์ /home/ftp/etc/passwd โดยใช้คำสั่งต่อไปนี้:

    ```
    cp /etc/passwd /home/ftp/etc/passwd
    ```
21. โดยใช้เอดิเตอร์โปรดของคุณ ให้แก้ไขไฟล์ /home/ftp/etc/passwd ตัวอย่าง:

    ```
    vi passwd
    ```
22. ลบทุกบรรทัดออกจากเนื้อหาที่ทำสำเนายกเว้นบรรทัดที่ใช้สำหรับ ผู้ใช้ root, ftp และ test หลังจากการแก้ไขของคุณ เนื้อหาควรคล้าย กับเนื้อหาต่อไปนี้:

    ```
    root:!:0:0::/:/bin/ksh
    ftp:*:226:1::/home/ftp:/usr/bin/ksh
    test:!:228:1::/home/test:/usr/bin/ksh
    ```
23. บันทึกการเปลี่ยนแปลงของคุณและออกจากเอดิเตอร์
24. เปลี่ยนสิทธิของไฟล์ /home/ftp/etc/passwd เป็น -rw-r--r-- โดยการพิมพ์:

    ```
    chmod 644 passwd
    ```
25. เปลี่ยนเจ้าของและกลุ่มของไฟล์ /home/ftp/etc/passwd เป็นผู้ใช้ root และกลุ่ม security โดยการพิมพ์:

    ```
    chown root:security passwd
    ```
26. ทำสำเนาเนื้อหาของไฟล์ /etc/security/passwd ไปยังไฟล์ /home/ftp/etc/security/passwd โดยใช้ คำสั่งต่อไปนี้:

    ```
    cp /etc/security/passwd /home/ftp/etc/security/passwd
    ```
27. โดยใช้เอดิเตอร์โปรดของคุณ ให้แก้ไขไฟล์ /home/ftp/etc/security/passwd ตัวอย่าง:

    ```
    vi ./security/passwd
    ```
28. ลบ stanzas ทั้งหมดออกจากเนื้อหาที่ทำสำเนายกเว้น stanza สำหรับผู้ใช้ test
29. ลบบรรทัด flags = ADMCHG ออกจาก stanza ผู้ใช้ test หลังจากการแก้ไขของคุณ เนื้อหาควรคล้าย กับเนื้อหาต่อไปนี้:

    ```
    test:
            password = 2HaAYgpDZX3Tw
            lastupdate = 990633278
    ```
30. บันทึกการเปลี่ยนแปลงของคุณและออกจากเอดิเตอร์
31. เปลี่ยนสิทธิของไฟล์ /home/ftp/etc/security/passwd เป็น -rw------- โดยการพิมพ์:

    ```
    chmod 600 ./security/passwd
    ```
32. เปลี่ยนเจ้าของและกลุ่มของไฟล์ /home/ftp/etc/security/passwd เป็นผู้ใช้ root และกลุ่ม security โดยการพิมพ์:

```
chown root:security ./security/passwd
```

33. ใช้เอดิเตอร์ที่คุณชอบ สร้างและแก้ไข ไฟล์ /home/ftp/etc/group ตัวอย่าง:

```
vi group
```

34. เพิ่มบรรทัดต่อไปนี้ในไฟล์:

```
system:*:0:
staff:*:1:test
```

35. บันทึกการเปลี่ยนแปลงของคุณและออกจากเอดิเตอร์
36. เปลี่ยนสิทธิของไฟล์ /home/ftp/etc/group เป็น -rw-r--r-- โดยการพิมพ์:

```
chmod 644 group
```

37. เปลี่ยนเจ้าของและกลุ่มของไฟล์ /home/ftp/etc/group เป็นผู้ใช้ root และกลุ่ม security โดยการพิมพ์:

```
chown root:security group
```

38. ใช้เอดิเตอร์ที่คุณชอบ สร้างและแก้ไข ไฟล์ /home/ftp/etc/security/group ตัวอย่าง:

```
vi ./security/group
```

39. เพิ่มบรรทัดต่อไปนี้ในไฟล์:

```
system:
 admin = true
staff
 admin = false
```

40. บันทึกการเปลี่ยนแปลงของคุณและออกจากเอดิเตอร์ ในการทำ นี้ ดำเนินขั้นตอนต่อไปนี้:
    a. ทำสำเนาไฟล์ /etc/security/user ไปยังไดเร็กทอรี /home/ftp/etc/security โดยการพิมพ์:

    ```
    cp /etc/security/user /home/ftp/etc/security
    cd /home/ftp/etc/
    ```

    b. ลบ stanzas ทั้งหมดออกจากเนื้อหาที่ทำสำเนา ยกเว้น stanza สำหรับผู้ใช้ test โดยใช้เอดิเตอร์โดยการพิมพ์:

    ```
    vi ./security/user
    ```

    c. บันทึกและออกจากเอดิเตอร์
41. เปลี่ยนสิทธิของไฟล์ /home/ftp/etc/security/group เป็น -rw-r----- โดยการพิมพ์:

```
chmod 640 ./security/group
```

42. เปลี่ยนเจ้าของและกลุ่มของไฟล์ /home/ftp/etc/security/group เป็นผู้ใช้ root และการรักษาความปลอดภัย โดยการพิมพ์:

```
chown root:security ./security/group
```

43. ใช้คำสั่งต่อไปนี้เพื่อทำสำเนาเนื้อหาที่เหมาะสม ไปไว้ในไดเร็กทอรี /home/ftp/etc/objrepos:

```
cp /etc/objrepos/CuAt ./objrepos
cp /etc/objrepos/CuAt.vc ./objrepos
cp /etc/objrepos/CuDep ./objrepos
cp /etc/objrepos/CuDv ./objrepos
cp /etc/objrepos/CuDvDr ./objrepos
cp /etc/objrepos/CuVPD ./objrepos
cp /etc/objrepos/Pd* ./objrepos
```

44. เปลี่ยนเป็นไดเร็กทอรี /home/ftp/home โดยการพิมพ์:

```
cd ../home
```

45. สร้างโฮมไดเร็กทอรีใหม่สำหรับผู้ใช้ของคุณ โดยการพิมพ์:

    ```
    mkdir test
    ```

    นี้จะเป็นโฮมไดเร็กทอรี สำหรับผู้ใช้ ftp ใหม่
46. เปลี่ยนเจ้าของและกลุ่มของไดเร็กทอรี /home/ftp/home/test เป็นผู้ใช้ test และกลุ่ม staff โดยการพิมพ์:

    ```
    chown test:staff test
    ```
47. เปลี่ยนสิทธิของไฟล์ /home/ftp/home/test เป็น -rwx------ โดยการพิมพ์:

    ```
    chmod 700 test
    ```
48. ปิดใช้งานรีโมตล็อกอินและคอนโซลล็อกอินสำหรับ ผู้ใช้ทดสอบ โดยพิมพ์:

    ```
    chuser login=false rlogin=false test
    ```

ณ จุดนี้ คุณได้ตั้งค่าล็อกอินย่อย ftp บนเครื่องของคุณแล้ว คุณสามารถทดสอบการตั้งค่านี้โดยใช้โพรซีเดอร์ต่อไปนี้:

1. โดยใช้ ftp เชื่อมต่อกับโฮสต์ซึ่งคุณสร้างผู้ใช้ ทดสอบ ตัวอย่างเช่น:

    ```
    ftp MyHost
    ```
2. ล็อกอินเป็น แบบนิรนาม เมื่อได้รับการพร้อมต์ให้ป้อนรหัสผ่าน ให้กด Enter
3. สลับไปยังผู้ใช้ ทดสอบ ที่สร้างขึ้นใหม่โดยใช้ คำสั่งต่อไปนี้:

    ```
    user test
    ```

    เมื่อได้รับการพร้อมต์ ให้ป้อนรหัสผ่าน ให้ใช้รหัสผ่านที่คุณสร้างขึ้นในขั้นตอน 18 ในหน้า 66
4. ใช้คำสั่ง **pwd** เพื่อตรวจสอบว่ามีไดเร็กทอรีโฮมของผู้ใช้ อยู่ ตัวอย่างเช่น:

    ```
    ftp> pwd
         /home/test
    ```

    เอาต์พุตแสดง /home/test เป็น ไดเร็กทอรีย่อย ftp ชื่อพาธแบบเต็มบน โฮสต์โดยแท้จริงคือ /home/ftp/home/test

หมายเหตุ:

- คุณสามารถสลับผู้ใช้ด้วยผู้ใช้ย่อย ftp เท่านั้น ตัวอย่างเช่น test เป็น ผู้ใช้ย่อย ftp
- เมื่อคุณสร้างผู้ใช้ ftp แบบนิรนาม ด้วย สคริปต์ anon.users.ftp คุณสามารถกำหนดชื่อใดๆ ให้แก่ผู้ใช้ก็ได้ โดยการแทนที่ *username* ในสคริปต์
- สำหรับผู้ใช้ แบบไม่ระบุชื่อ เนื่องจากเซิร์ฟเวอร์ดำเนินการ คำสั่ง **chroot** ในโฮมไดเร็กทอรีของ บัญชีผู้ใช้ ไฟล์ที่เกี่ยวกับการตั้งค่าใดๆ เช่น *fileftpaccess.ctl* ควรอยู่ในโฮมไดเร็กทอรี เช่น ~/etc/ ของผู้ใช้แบบไม่ระบุชื่อตามลำดับ 'ข้อจำกัด 'Writeonly,' 'readonly' และ 'readwrite' ในไฟล์ /etc/ftpaccess.ctl ต้องมีพาธสัมพันธ์กับพาธ chrooted

สำหรับข้อมูลเพิ่มเติม:

- "ความปลอดภัย TCP/IP" ใน *การรักษาความปลอดภัย*
- "**คำสั่ง ftp**" ใน *การอ้างอิงคำสั่ง*

บัญชีผู้ใช้พิเศษของระบบ

AIX จัดให้มี ชุดของบัญชีผู้ใช้พิเศษของระบบค่าดีฟอลต์ที่ป้องกันมิให้บัญชีผู้ใช้ root และบัญชีผู้ใช้ระบบเป็นเจ้าของไฟล์ระบบปฏิบัติการ และ ระบบไฟล์ทั้งหมด

ข้อควรสนใจ: ควรใช้ความระมัดระวังเมื่อลบบัญชีผู้ใช้พิเศษ ของระบบออก คุณสามารถปิดใช้งานบัญชีผู้ใช้ที่ระบุโดยการแทรกเครื่องหมายดอกจัน (*) ที่ต้นบรรทัดที่เกี่ยวข้องของไฟล์ /etc/security/passwd อย่างไรก็ตาม ขอให้ระมัดระวังในการปิดใช้งานบัญชีผู้ใช้ root ถ้าคุณลบ บัญชีผู้ใช้พิเศษของระบบออก หรือปิดใช้งานบัญชีผู้ใช้ root ระบบปฏิบัติการ จะไม่ทำงาน

บัญชีผู้ใช้ต่อไปนี้ถูกกำหนดไว้แล้วในระบบปฏิบัติการ:

adm บัญชีผู้ใช้ adm เป็นเจ้าของฟังก์ชันระบบระดับต้นต่อไปนี้:

- การวินิจฉัย เครื่องมือซึ่งถูกเก็บในไดเร็กทอรี /usr/sbin/perf/diag_tool
- การจัดการบัญชีผู้ใช้ เครื่องมือซึ่งถูกเก็บในไดเร็กทอรีต่อไปนี้:
  - /usr/sbin/acct
  - /usr/lib/acct
  - /var/adm
  - /var/adm/acct/fiscal
  - /var/adm/acct/nite
  - /var/adm/acct/sum

bin บัญชีผู้ใช้ bin โดยทั่วไปเป็นเจ้าของไฟล์เรียกทำงานสำหรับคำสั่ง ผู้ใช้ส่วนใหญ่ วัตถุประสงค์หลักของบัญชีผู้ใช้นี้คือช่วยในการแจกจ่าย ความเป็นเจ้าของของไดเร็กทอรีและไฟล์ระบบที่สำคัญ เพื่อมิให้บัญชีผู้ใช้ root และ sys เป็นเจ้าของทั้งหมดแต่เพียงผู้เดียว

daemon
บัญชีผู้ใช้ daemon มีอยู่เฉพาะเพื่อเป็นเจ้าของและรันกระบวนการเซิร์ฟเวอร์ ระบบและไฟล์ที่เชื่อมโยงเท่านั้น บัญชีผู้ใช้นี้รับประกันว่า กระบวนการจะรันด้วยสิทธิการเข้าถึงไฟล์ที่เหมาะสม

nobody บัญชีผู้ใช้ nobody ถูกใช้โดย Network File System (NFS) เพื่อเปิดใช้งานการพิมพ์รีโมต บัญชีผู้ใช้นี้มีอยู่เพื่อให้โปรแกรมสามารถ อนุญาตการเข้าถึง root ชั่วคราวให้แก่ผู้ใช้ root ตัวอย่าง ก่อนการเปิดใช้งาน Secure RPC หรือ Secure NFS ตรวจสอบคีย์ /etc/public บนเซิร์ฟเวอร์ NIS หลักเพื่อค้นหาผู้ใช้ที่ยังไม่ถูกกำหนด พับลิกคีย์และคีย์ลับ ในฐานะผู้ใช้ root คุณสามารถสร้างรายการ ขึ้นในฐานข้อมูลสำหรับแต่ละผู้ใช้ที่ไม่ถูกกำหนดโดยการป้อน:

```
newkey -u username
```

หรือ คุณสามารถสร้างรายการในฐานข้อมูลสำหรับบัญชีผู้ใช้ nobody และจากนั้นผู้ใช้ใดก็ตามสามารถรันโปรแกรม **chkey** เพื่อสร้างรายการ ของตนในฐานข้อมูลโดยไม่ต้องล็อกอินเป็น root

root บัญชีผู้ใช้ root คือ UID 0 ซึ่งคุณสามารถดำเนินงานการบำรุงรักษา ระบบและแก้ไขปัญหาของระบบ

sys ผู้ใช้ sys เป็นเจ้าของจุดการเมาท์ดีฟอลต์สำหรับแคช Distributed File Service (DFS) ซึ่ง ต้องมีอยู่ก่อนคุณจึงจะสามารถติดตั้งหรือตั้งค่า DFS บนไคลเอ็นต์ ไดเร็กทอรี /usr/sys ยังสามารถเก็บอิมเมจการติดตั้ง

system กลุ่ม System คือกลุ่มที่ระบบกำหนดสำหรับผู้ดูแลระบบ ผู้ใช้ของกลุ่ม system มีสิทธิ์พิเศษในการดำเนินงานการบำรุงรักษาระบบ บางอย่างโดยไม่ต้องร้องขอสิทธิ root

การลบบัญชีผู้ใช้ดีฟอลต์ที่ไม่จำเป็นออก:

ระหว่างการติดตั้งระบบปฏิบัติการ มี ID ผู้ใช้ และ ID กลุ่มจำนวนหนึ่งถูกสร้างขึ้น ทั้งนี้ขึ้นอยู่กับแอ็พพลิเคชัน ที่คุณกำลังทำงานบนระบบของคุณ และตำแหน่งที่ระบบของคุณตั้งอยู่ ในเน็ตเวิร์ก บาง ID ผู้ใช้และกลุ่มเหล่านี้สามารถเป็นจุดอ่อนด้าน

ความปลอดภัย เสี่ยงต่อการใช้งานในทางที่ไม่ถูกต้อง ถ้า ID ผู้ใช้และกลุ่มเหล่านี้ ไม่จำเป็น คุณสามารถลบออกเพื่อลดความเสี่ยงต่อความปลอดภัยที่เกี่ยวข้องกับ ID เหล่านี้ให้เหลือน้อยสุด

ตารางต่อไปนี้แสดงรายการ ID ผู้ใช้ดีฟอลต์ส่วนใหญ่ ที่คุณอาจสามารถลบออกได้:

ตารางที่ 3. ID ผู้ใช้ดีฟอลต์ ทั่วไปที่คุณอาจสามารถลบออก

ID ผู้ใช้	คำอธิบาย
uucp, nuucp	เจ้าของไฟล์ซ่อนที่ใช้โดยโปรโตคอล uucp บัญชีผู้ใช้ uucp ใช้สำหรับ UNIX-to-UNIX Copy Program ซึ่ง เป็นกลุ่มของคำสั่ง โปรแกรม และไฟล์ ที่มีอยู่บนระบบ AIX ส่วนใหญ่ ที่อนุญาตให้ผู้ใช้สื่อสารกับระบบ AIX อื่น ผ่านสายสื่อสารเฉพาะงานหรือสายโทรศัพท์
lpd	เจ้าของไฟล์ที่ใช้โดยระบบย่อยการพิมพ์
guest	อนุญาตการเข้าถึงแก่ผู้ใช้ที่ไม่มีการเข้าถึง บัญชีผู้ใช้

ตารางต่อไปนี้แสดง ID กลุ่มทั่วไป ที่อาจไม่จำเป็นต้องใช้:

ตารางที่ 4. ID กลุ่มทั่วไปที่อาจไม่ จำเป็น

ID กลุ่ม	คำอธิบาย
uucp	กลุ่มซึ่งผู้ใช้ uucp และ nuucp อยู่
printq	กลุ่มที่ผู้ใช้ lpd อยู่

วิเคราะห์ระบบของคุณเพื่อพิจารณาว่า IDs ใดไม่จำเป็น จริงๆ ทั้งยังอาจมี ID ผู้ใช้หรือกลุ่มเพิ่มขึ้นที่ คุณอาจไม่จำเป็นต้องใช้ ก่อนที่ระบบของคุณจะเริ่มทำงานจริง ให้ดำเนิน การโดยการประเมิน IDs ที่มีอยู่

บัญชีผู้ใช้ที่สร้างโดยคอมโพเนนต์ความปลอดภัย:

เมื่อคอมโพเนนต์ความปลอดภัยเช่น LDAP และ OpenSSH ถูกติดตั้ง หรือตั้งค่า บัญชีผู้ใช้หรือบัญชีกลุ่มจะถูกสร้างขึ้น

บัญชีผู้ใช้และบัญชีกลุ่มที่สร้างประกอบด้วย:

- **Internet Protocol (IP) Security:** IP Security เพิ่มผู้ใช้ *ipsec* และ กลุ่ม *ipsec* ระหว่างการติดตั้ง ID เหล่านี้ถูกใช้โดยเซอร์วิสการจัดการคีย์ โปรดทราบว่า ID กลุ่มใน `/usr/lpp/group.id.keymgt` ไม่สามารถ กำหนดเองก่อนการติดตั้ง
- **Kerberos and Public Key Infrastructure (PKI):** คอมโพเนนต์เหล่านี้ ไม่สร้างผู้ใช้หรือแอคเคาต์กลุ่มใดๆ
- **LDAP:** เมื่อไคลเอ็นต์หรือเซิร์ฟเวอร์ LDAP ถูกติดตั้ง แอคเคาต์ผู้ใช้ *ldap* จะถูกสร้างขึ้น ID ผู้ใช้ของ *ldap* ไม่ถูกกำหนดคงที่ เมื่อติดตั้งเซิร์ฟเวอร์ LDAP, เซิร์ฟเวอร์จะติดตั้งฐานข้อมูล DB2® โดยอัตโนมัติ การติดตั้ง DB2 จะสร้างแอคเคาต์ผู้ใช้กลุ่ม *dbsysadm* ID กลุ่มดีฟอลต์ของ *dbsysadm* คือ 400 ระหว่างการกำหนดคอนฟิกของเซิร์ฟเวอร์ LDAP คำสั่ง **mksecldap** จะสร้างแอคเคาต์ผู้ใช้ *ldapdb2*
- **OpenSSH:** ระหว่างการติดตั้ง OpenSSH ผู้ใช้ *sshd* และ กลุ่ม *sshd* ถูกเพิ่มในระบบ ID ผู้ใช้และกลุ่ม ที่สอดคล้องต้องไม่ถูกเปลี่ยนแปลง คุณลักษณะการจัดแบ่งสิทธิ ใน SSH จำเป็นต้องใช้ ID

กลุ่มที่ไม่มีโดเมน

คุณลักษณะกลุ่มที่ไม่มีโดเมนอนุญาตให้คุณกำหนดให้กับผู้ใช้ ที่ถูกนิยามในหนึ่งโดเมนเพื่อจัดกลุ่มที่ถูกนิยามไว้ใน โดเมนอื่น คุณลักษณะนี้สนับสนุนเฉพาะ Lightweight Database Access Protocol (LDAP) และโดเมนโลคัล

คุณสามารถสร้างผู้ใช้และกลุ่มบนเซิร์ฟเวอร์ LDAP โดยใช้ LDAP Authentication Load Module (โมดูล LDAP) คุณยังสามารถสร้างผู้ใช้ และกลุ่มบนระบบโลคัลโดยใช้ Local Authentication Load Module (โมดูลโลคัล) เมื่อเปิดใช้งานคุณลักษณะ **domainlessgroups**, ผู้ใช้และกลุ่มผู้ใช้ที่ถูกสร้างบน LDAP หรือระบบโลคัลไม่สามารถกำหนดให้กับกลุ่มภายนอกโดเมนโหลดที่ถูกสร้าง ตัวอย่างเช่น, ผู้ใช้ที่ถูกสร้างขึ้นใน โดเมน LDAP ไม่สามารถกำหนดให้กับกลุ่มที่เชื่อมโยงกับ โดเมนโลคัล

คุณสามารถได้รับผลลัพธ์ของข้อจำกัดนี้และกำหนดผู้ใช้ทั้ง LDAP และกลุ่มโลคัลโดยเปิดใช้งานคุณสมบัติระบบ **domainlessgroups** คุณสมบัติ **domainlessgroups** ถูกนิยามในไฟล์ `/etc/secvars.cfg` ซึ่ง สนับสนุนเฉพาะ LDAP และโมดูลโลคัล ค่าที่อาจเป็นไปได้ สำหรับคุณสมบัตินี้มีดังต่อไปนี้:

false (ค่าดีฟอลต์)
แอ็ตทริบิวต์กลุ่มถูกผสานจากโมดูล LDAP และโมดูลโลคัล

true แอ็ตทริบิวต์กลุ่มถูกผสานจาก LDAP และโมดูลโลคัล ตัวอย่างเช่น, ผู้ใช้ LDAP สามารถกำหนดให้กับกลุ่มโลคัล

เมื่อต้องการดูค่าของๆ คุณสมบัติ **domainlessgroups**, รันคำสั่งต่อไปนี้:

```
lssec -f /etc/secvars.cfg -s groups -a domainlessgroups
```

เมื่อต้องการตั้งค่าคุณสมบัติ **domainlessgroups** ให้เป็นจริง, ให้รันคำสั่งต่อไปนี้:

```
chsec -f /etc/secvars.cfg -s groups -a domainlessgroups=true
```

ตารางต่อไปนี้อธิบายถึงผลลัพธ์ที่แตกต่างกันของคำสั่งผู้ใช้ และกลุ่ม, ขึ้นอยู่กับค่าติดตั้งของคุณสมบัติ **domainlessgroups**

ตารางที่ 5. ผลลัพธ์ของคำสั่งที่เลือกไว้ ซึ่งมีผลต่อคุณสมบัติ **domainlessgroups**

คำสั่ง	**ส่งผลเมื่อคุณสมบัติ domainlessgroups ถูกตั้งค่าเป็นจริง**
`chgroup -R ldap|files`	อัพเดตกลุ่มในโดเมนที่ระบุ คุณสามารถ เพิ่มผู้ใช้ให้กับ LDAP หรือกลุ่มโลคัล
`chuser -R ldap|files`	เปลี่ยนค่าติดตั้งสำหรับผู้ใช้ใน โดเมนที่ระบุไว้ ถ้ากลุ่มที่ถูกนิยามในโดเมนอื่นถูกระบุไว้, กลุ่มเหล่านั้นยังถูกอัพเดตด้วยข้อมูลผู้ใช้
`login` *username* หรือ `su`	ดึงแอ็ตทริบิวต์ผู้ใช้จากรีจิสทรี ผู้ใช้, ยกเว้นแอ็ตทริบิวต์ ID กลุ่ม แอ็ตทริบิวต์ผู้ใช้สำหรับ ID กลุ่มถูกผสานจากทั้งโดเมน LDAP และโดเมนโลคัล
`lsgroup -R ldap|files`	แสดงรายการแอ็ตทริบิวต์กลุ่มทั้งหมดสำหรับ โดเมนที่ระบุไว้ ถ้าไม่ได้ค้นหากลุ่มที่ระบุไว้ในโดเมนที่ระบุไว้, คำสั่งจะล้มเหลว
`lsuser -R ldap|files`	แสดงรายการแอ็ตทริบิวต์ของผู้ใช้หลังจากข้อมูล ถูกผสานจากกลุ่มทั้งหมดในโดเมนที่ผู้ใช้นิยามไว้ และโดเมนอื่น ถ้ากลุ่มหลักของผู้ใช้ไม่ได้ถูกนิยามไว้ในโดเมนซึ่งนิยามผู้ใช้ไว้, ซึ่งถูกแก้ไขจาก โดเมนอื่น
`mkgroup -R ldap|files`	สร้างกลุ่มในโดเมนโดเมนที่ระบุเฉพาะ หลังจาก ที่คุณสร้างกลุ่ม, คุณกำหนดผู้ใช้ (LDAP หรือโลคัล) ให้กับกลุ่มในฐานข้อมูลสำหรับโดเมนนั้น คุณสามารถเพิ่มผู้ใช้ ให้กับ LDAP หรือกลุ่มโลคัล
`mkuser -R ldap|files`	สร้างผู้ใช้ในโดเมนที่ระบุไว้ ถ้ากลุ่ม ที่ถูกนิยามไว้ในโดเมนอื่นที่ระบุไว้, กลุ่มเหล่านั้น ยังถูกอัพเดตกับข้อมูลผู้ใช้
`rmgroup -R ldap|files`	ลบกลุ่มที่ระบุไว้จาก โดเมนที่ระบุไว้ ถ้ากลุ่มถูกกำหนดไว้เป็นกลุ่มหลักสำหรับผู้ใช้ใดๆ ที่ถูกนิยามไว้ในโดเมนใดๆ, คำสั่งจะล้มเหลว

ตารางที่ 5. ผลลัพธ์ของคำสั่งที่เลือกไว้ ซึ่งมีผลต่อคุณสมบัติ **domainlessgroups** *(ต่อ)*

คำสั่ง	ส่งผลเมื่อคุณสมบัติ domainlessgroups ถูกตั้งค่าเป็นจริง
`rmuser -R ldap|files`	ลบผู้ใช้ที่ระบุไว้จาก โดเมนที่ระบุไว้ ซึ่งยังลบผู้ใช้ออกจากกลุ่มใดๆ ที่ถูกนิยามไว้ใน โดเมนอื่นๆ และมีผู้ใช้นี้เป็นสมาชิก

หลักการที่เกี่ยวข้อง:

"โหลดโมดูลการพิสูจน์ตัวตน LDAP" ในหน้า 165

การใช้ประโยชน์ LDAP ของระบบย่อยการรักษาความปลอดภัยที่ถูกนำไปใช้ เป็นโหลดโมดูลการพิสูจน์ตัวตน LDAP โดยความคิดแล้วเหมือนกับ โหลดโมดูลอื่นๆ เช่น NIS, DCE และ KRB5 โหลดโมดูลถูกกำหนดใน ไฟล์ `/usr/lib/security/methods.cfg`

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง chgroup

คำสั่ง chuser

คำสั่ง login

คำสั่ง lsgroup

คำสั่ง lsuser

คำสั่ง mkgroup

คำสั่ง mkuser

คำสั่ง rmgroup

คำสั่ง rmuser

คำสั่ง su

รหัสผ่าน

การเดารหัสผ่านเป็นวิธีการโจมตีรูปแบบหนึ่งที่นิยมมากสุดที่ ระบบต้องประสบ ดังนั้น จึงจำเป็นที่จะต้องมีการควบคุมและการมอนิเตอร์นโยบาย การจำกัดรหัสผ่านของคุณ

AIX มีกลไกเพื่อ ช่วยคุณบังคับใช้นโยบายรหัสผ่านที่มีความรัดกุมมากขึ้น เช่นการสร้างค่าสำหรับ สิ่งต่อไปนี้:

- จำนวนสัปดาห์ต่ำสุดและสูงสุดที่สามารถผ่านไปก่อนและหลัง ที่จะสามารถให้เปลี่ยนรหัสผ่านได้
- ความยาวต่ำสุดของรหัสผ่าน
- จำนวนของอักขระแบบตัวอักษรต่ำสุดที่สามารถใช้เมื่อเลือก รหัสผ่าน

การสร้างรหัสผ่านที่ดี:

รหัสผ่านที่ดีเป็นด่านแรกที่มีประสิทธิผลในการป้องกัน การเข้าสู่ระบบที่ไม่ได้รับอนุญาต

รหัสผ่านจะมีประสิทธิผลถ้า:

- มีการผสมระหว่างตัวอักษรทั้งตัวพิมพ์ใหญ่และตัวพิมพ์เล็ก
- การใช้ร่วมกันของตัวอักษร ตัวเลข หรือเครื่องหมายวรรคตอน รวมทั้งรหัสผ่านยังสามารถมีอักขระพิเศษเช่น `~!@#$%^&*()-_=+[]{}|\;:'",.<>?/<space>`
- ไม่ถูกเขียนลงในที่อื่นใด

- ต้องมีความยาวอย่างน้อย 7 ตัวอักษรจนถึงค่าสูงสุด PW_PASSLEN อักขระ ถ้าใช้ไฟล์ /etc/security/passwd (การนำใช้การพิสูจน์ตัวตนที่ใช้รีจิสทรี เช่น LDAP สามารถมีรหัสผ่านที่ยาวเกินความยาวสูงสุดนี้)
- ไม่ใช่คำศัพท์ที่สามารถพบได้ในพจนานุกรมใดๆ
- ไม่มีรูปแบบของตัวอักษรเรียงตามที่อยู่บนคีย์บอร์ด เช่น *qwerty*
- ไม่ใช่คำศัพท์หรือรูปแบบที่รู้จักที่ถูกสะกดแบบย้อนกลับ
- ไม่มีข้อมูลส่วนบุคคลใดๆ เกี่ยวกับตัวคุณ ครอบครัว หรือเพื่อนๆ
- อย่าใช้รูปแบบเดิมเหมือนกับที่ใช้กับรหัสผ่านก่อนหน้า
- สามารถพิมพ์ได้อย่างรวดเร็วต่อเนื่องเพื่อที่บุคคลอื่นที่อยู่ในบริเวณใกล้เคียงไม่สามารถ จดจำรหัสผ่านของคุณได้

นอกเหนือจากวิธีการเหล่านี้แล้ว คุณยังสามารถบังคับใช้ กฎที่มีความเข้มงวดมากขึ้นได้โดยการจำกัดรหัสผ่านเพื่อให้ไม่สามารถมี คำ UNIX มาตรฐานซึ่ง สามารถถูกคาดเดาได้ คุณลักษณะนี้ใช้ dictionlist ซึ่งจำเป็นที่ อันดับแรกคุณต้องมีชุดไฟล์ bos.data และ bos.txt ถูกติดตั้ง

ในการนำใช้ dictionlist ที่กำหนดก่อนหน้านี้ ให้แก้ไขที่บรรทัดต่อไปนี้ในไฟล์ /etc/security/users:

```
dictionlist = /usr/share/dict/words
```

ไฟล์ /usr/share/dict/words ใช้ dictionlist เพื่อป้องกันคำ UNIX มาตรฐานมิให้ถูกนำมาใช้เป็นรหัสผ่าน

การใช้ไฟล์ /etc/passwd:

โดยทั่วไป ไฟล์ /etc/passwd ถูกใช้เพื่อเก็บค่าการติดตามผู้ใช้ที่ลงทะเบียนทุกคนที่มีการเข้าถึง ระบบ

ไฟล์ /etc/passwd คือไฟล์ที่คั่นด้วย โคลอนที่มีข้อมูลต่อไปนี้:

- ชื่อผู้ใช้
- รหัสผ่านที่เข้ารหัส
- หมายเลข ID ผู้ใช้ (UID)
- หมายเลข ID กลุ่มของผู้ใช้ (GID)
- ชื่อเต็มของผู้ใช้ (GECOS)
- โฮมไดเร็กทอรีของผู้ใช้
- ล็อกอินเชลล์

ต่อไปนี้คือตัวอย่างของไฟล์ /etc/passwd:

```
root:!:0:0::/:/usr/bin/ksh
daemon:!:1:1::/etc:
bin:!:2:2::/bin:
sys:!:3:3::/usr/sys:
adm:!:4:4::/var/adm:
uucp:!:5:5::/usr/lib/uucp:
guest:!:100:100::/home/guest:
nobody:!:4294967294:4294967294::/:
lpd:!:9:4294967294::/:
lp:*:11:11::/var/spool/lp:/bin/false
```

```
invscout:*:200:1::/var/adm/invscout:/usr/bin/ksh
nuucp:*:6:5:uucp login user:/var/spool/uucppublic:/usr/sbin/uucp/uucico
paul:!:201:1::/home/paul:/usr/bin/ksh
jdoe:*:202:1:John Doe:/home/jdoe:/usr/bin/ksh
```

AIX ไม่เก็บรหัสผ่านที่เข้ารหัส ในไฟล์ /etc/password ตามแบบที่ใช้ในระบบ UNIX แต่เก็บไว้ที่ไฟล์ /etc/security/password [1] ตามค่าดีฟอลต์ ซึ่งสามารถอ่านได้เฉพาะผู้ใช้ root เท่านั้น รหัสผ่านที่อยู่ในไฟล์ /etc/passwd ถูก ใช้โดย AIX เพื่อใช้แสดงว่า มีรหัสผ่านหรือบัญชีผู้ใช้ถูกบล็อกหรือไม่

ไฟล์ /etc/passwd เป็นเจ้าของโดยผู้ใช้ root และต้องสามารถอ่านได้โดยผู้ใช้ทุกคน แต่ มีเพียงผู้ใช้ root เท่านั้นที่มีสิทธิสามารถเขียนได้ ซึ่งถูกแสดงเป็น -rw-r--r-- ถ้า ID ผู้ใช้ มีรหัสผ่าน ดังนั้นฟิลด์รหัสผ่านจะมีเครื่องหมาย ! (เครื่องหมายตกใจ) ถ้า ID ผู้ใช้ ไม่มีรหัสผ่าน ดังนั้นฟิลด์รหัสผ่าน จะมี * (เครื่องหมายดอกจัน) รหัสผ่านที่เข้ารหัส ถูกเก็บในไฟล์ /etc/security/passwd ตัวอย่างต่อไปนี้มีรายการสี่รายการล่าสุดในไฟล์ /etc/security/passwd โดยยึดตามรายการจากไฟล์ /etc/passwd ที่แสดงก่อนหน้านี้

```
guest:
        password = *

nobody:
        password = *

lpd:
        password = *

paul:
        password = eacVScDKri4s6
        lastupdate = 1026394230
        flags = ADMCHG
```

ID ผู้ใช้ jdoe ไม่มีรายการในไฟล์ /etc/security/passwd เนื่องจากไม่มีรหัสผ่านถูกตั้งค่าในไฟล์ /etc/passwd

ความสอดคล้องกัน ของไฟล์ /etc/passwd สามารถตรวจสอบได้โดยใช้คำสั่ง **pwdck** คำสั่ง **pwdck** ตรวจสอบความถูกต้องของ ข้อมูลรหัสผ่านในไฟล์ฐานข้อมูลผู้ใช้โดยการตรวจสอบนิยาม สำหรับผู้ใช้ทุกคน หรือเฉพาะผู้ใช้ที่ระบุ

การใช้ไฟล์ /etc/passwd และสภาวะแวดล้อมเน็ตเวิร์ก:

ในสภาวะแวดล้อมที่เป็นเน็ตเวิร์กโดยทั่วไป ผู้ใช้ต้องมีบัญชีผู้ใช้ บนแต่ละระบบเพื่อสามารถเข้าถึงระบบนั้น

โดยปกติจะหมายความว่าผู้ใช้จะมีรายการในแต่ละไฟล์ของไฟล์ /etc/passwd บนแต่ละระบบ อย่างไรก็ตาม ใน สภาวะแวดล้อมแบบมีการกระจาย ไม่มีวิธีการที่ง่ายในการทำให้มั่นใจว่าทุกระบบ มีไฟล์ /etc/passwd เดียวกัน เมื่อต้องการแก้ไขปัญหานี้ หลายๆ วิธีสร้างข้อมูลในไฟล์ /etc/passwd ที่พร้อมใช้งานผ่านเครือข่าย รวมถึง Network Information System (NIS)

การซ่อนชื่อผู้ใช้และรหัสผ่าน:

เพื่อให้มีการการรักษาความปลอดภัยในระดับสูงยิ่งขึ้น โปรดตรวจสอบว่า ID ผู้ใช้ และรหัสผ่านไม่สามารถเห็นได้ในระบบ

1. /etc/security/password

ไฟล์ .netrc มี ID ผู้ใช้ และรหัสผ่าน ไฟล์นี้ไม่ได้รับการป้องกันโดยการเข้ารหัสหรือแปลงให้เป็นรหัส ดังนั้นเนื้อหาของไฟล์จะถูกแสดงเป็นข้อความธรรมดาอย่างชัดเจน ในการค้นหาไฟล์เหล่านี้ ให้รันคำสั่งต่อไปนี้:

```
# find `awk -F: '{print $6}' /etc/passwd` -name .netrc -ls
```

หลังจากคุณค้นหาไฟล์เหล่านี้พบแล้ว ให้ลบทิ้ง วิธีการที่มีประสิทธิผลมากยิ่งขึ้นในการ บันทึกรหัสผ่านคือโดยการติดตั้ง Kerberos สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ Kerberos ดูที่ "Kerberos" ในหน้า 317

การตั้งค่าอ็อพชันรหัสผ่านที่แนะนำ:

การจัดการรหัสผ่านอย่างเหมาะสมสามารถกระทำได้โดยการให้ความรู้แก่ ผู้ใช้เท่านั้น ในการจัดให้มีการรักษาความปลอดภัยเพิ่มขึ้นบางอย่าง ระบบปฏิบัติการ จะมีข้อจำกัดรหัสผ่านที่สามารถตั้งค่าได้ ซึ่งอนุญาตให้ ผู้ดูแลระบบสามารถจำกัดการเลือกใช้รหัสผ่านโดยผู้ใช้ และบังคับให้ มีการเปลี่ยนรหัสผ่านเป็นประจำ

อ็อพชันรหัสผ่านและแอ็ตทริบิวต์ผู้ใช้ที่ขยายเพิ่มจะอยู่ ในไฟล์ /etc/security/user ซึ่งเป็นไฟล์ ASCII ที่มี stanzas แอ็ตทริบิวต์สำหรับผู้ใช้ ข้อจำกัดเหล่านี้ ถูกบังคับใช้เมื่อมีการกำหนดรหัสผ่านใหม่สำหรับผู้ใช้ ข้อจำกัดรหัสผ่าน ทั้งหมดถูกกำหนดให้แก่แต่ละผู้ใช้ โดยการเก็บรักษาข้อจำกัดไว้ใน stanza ดีฟอลต์ของไฟล์ /etc/security/user ทำให้ข้อจำกัดเดียวกันถูกบังคับใช้กับผู้ใช้ทั้งหมด ในการเก็บรักษาความปลอดภัย รหัสผ่าน รหัสผ่านทั้งหมดต้องได้รับการป้องกันแบบเดียวกัน

ผู้ดูแลระบบ ยังสามารถขยายข้อจำกัดรหัสผ่านเพิ่มได้ โดยใช้แอ็ตทริบิวต์ **pwdchecks** ของไฟล์ /etc/security/user ผู้ดูแลระบบ สามารถเพิ่มรูทีนย่อยใหม่ (หรือที่เรียกว่า *วิธีการ*) ในโค้ด ข้อจำกัดรหัสผ่าน ดังนั้น นโยบายของโลคัลไซต์สามารถถูกเพิ่ม และบังคับใช้ โดยระบบปฏิบัติการ สำหรับข้อมูลเพิ่มเติม ดูที่ "การขยายข้อจำกัดรหัสผ่าน" ในหน้า 79

ใช้ ข้อจำกัดรหัสผ่านอย่างสมเหตุสมผล การพยายามสร้างข้อจำกัดมากเกินไป เช่น การจำกัดขอบเขตของรหัสผ่าน จะทำให้เดารหัสผ่าน ได้ง่ายขึ้น หรือการบังคับให้ผู้ใช้เลือกรหัสผ่านที่จดจำได้ ยาก อาจทำให้ต้องเขียนเก็บไว้ ก็สามารถทำให้เสี่ยงต่อการรักษาความปลอดภัย รหัสผ่าน ในท้ายที่สุดแล้ว การรักษาความปลอดภัยรหัสผ่านจะต้องขึ้นอยู่กับผู้ใช้ ข้อจำกัด รหัสผ่านอย่างง่าย ร่วมกับแนวทางที่มีเหตุผลและการตรวจสอบ เป็นครั้งคราวเพื่อยืนยันว่ารหัสผ่านปัจจุบันเป็นค่าเฉพาะ ถือเป็นนโยบายที่ดีที่สุด

ตารางต่อไปนี้แสดงรายการค่าที่แนะนำสำหรับแอ็ตทริบิวต์การรักษาความปลอดภัยบางค่า ที่เกี่ยวข้องกับรหัสผ่านผู้ใช้ในไฟล์ /etc/security/user

ตารางที่ 6. ค่าแอ็ตทริบิวต์การรักษาความปลอดภัยที่แนะนำสำหรับรหัสผ่านผู้ใช้

แอ็ตทริบิวต์	คำอธิบาย	ค่าที่แนะนำ	ค่าดีฟอลต์	ค่าสูงสุด
dictionlist	ตรวจสอบว่ารหัสผ่านไม่มีคำ UNIX มาตรฐาน	**/usr/share/dict/words**	ไม่ระบุ	ไม่ระบุ
histexpire	จำนวนสัปดาห์ก่อนที่รหัสผ่านจะสามารถใช้ใหม่ได้	26	0	260*
histsize	จำนวนการวนซ้ำรหัสผ่านที่อนุญาต	20	0	50
maxage	จำนวนสัปดาห์สูงสุดก่อนที่จะต้องเปลี่ยน รหัสผ่าน	8	0	52

ตารางที่ 6. ค่าแอ็ตทริบิวต์การรักษาความปลอดภัยที่แนะนำสำหรับรหัสผ่านผู้ใช้ (ต่อ)

แอ็ตทริบิวต์	คำอธิบาย	ค่าที่แนะนำ	ค่าดีฟอลต์	ค่าสูงสุด
maxexpired	จำนวนสัปดาห์สูงสุดหลังจากเลย *maxage* ที่ รหัสผ่านที่หมดอายุสามารถเปลี่ยนได้โดยผู้ใช้ (ยกเว้น Root)	2	-1	52
maxrepeats	จำนวนอักขระสูงสุดที่สามารถซ้ำได้ในรหัสผ่าน	2	8	8
minage	จำนวนสัปดาห์ต่ำสุดก่อนที่รหัสผ่านจะสามารถ ถูกเปลี่ยน ค่านี้ไม่ควรถูกตั้งค่าเป็นค่าที่ไม่ใช่ศูนย์ยกเว้น ผู้ดูแลระบบ เข้าถึงเพื่อตั้งค่ารหัสผ่านที่ รั่วไหลโดยบังเอิญที่ เพิ่งถูกเปลี่ยนเมื่อเร็วๆ นี้ได้ง่าย	0	0	52
minalpha	จำนวนอักขระแบบตัวอักษรต่ำสุดที่จำเป็น ต้องมีในรหัสผ่าน	2	0	PW_PASSLEN**
mindiff	จำนวนอักขระเฉพาะต่ำสุดที่รหัสผ่าน ต้องมี	4	0	PW_PASSLEN**
minlen	ความยาวต่ำสุดของรหัสผ่าน	6 (8 สำหรับผู้ใช้ root)	0	PW_PASSLEN**
minother	จำนวนอักขระที่มิใช้ตัวอักษรต่ำสุด ที่จำเป็นต้องมีในรหัสผ่าน	2	0	PW_PASSLEN**
pwdwarntime	จำนวนวันก่อนที่ระบบจะออกคำเตือน ที่จำเป็นต้องให้มีการเปลี่ยนรหัสผ่าน	5	ไม่ระบุ	ไม่ระบุ
pwdchecks	รายการนี้สามารถใช้เพื่อเพิ่มคำสั่ง **passwd** ที่มีโค้ดแบบกำหนดเองที่จะตรวจสอบคุณภาพของรหัสผ่าน	สำหรับข้อมูลเพิ่มเติม ดูที่ "การขยายข้อจำกัดรหัสผ่าน" ในหน้า 79	ไม่ระบุ	ไม่ระบุ

* รหัสผ่านสูงสุด 50 รหัสผ่านจะถูกเก็บไว้

** PW_PASSLEN ถูกกำหนดในไฟล์ `userpw.h`

ถ้า มีการติดตั้งการประมวลผลข้อความบนระบบ ผู้ดูแลระบบสามารถ ใช้ไฟล์ `/usr/share/dict/words` เป็นไฟล์พจนานุกรม **dictionlist** ในกรณีเช่นนี้ ผู้ดูแลระบบสามารถตั้งค่าแอ็ตทริบิวต์ **minother** เป็น 0 เนื่องจากคำส่วนใหญ่ในไฟล์พจนานุกรมไม่มี อักขระใดๆ ที่อยู่ในหมวดหมู่แอ็ตทริบิวต์ **minother** การตั้งค่าแอ็ตทริบิวต์ **minother** เป็น 1 หรือ มากกว่าเพื่อขจัดความจำเป็นในการใช้คำจำนวนมากมายในไฟล์พจนานุกรม นี้

ความยาวต่ำสุดของรหัสผ่าน บนระบบถูกตั้งค่าโดยค่าของแอ็ตทริบิวต์ **minlen** หรือค่าของแอ็ตทริบิวต์ **minalpha** ที่เพิ่มในค่าของแอ็ตทริบิวต์ **minother** ขึ้นอยู่กับว่าค่าใดมากกว่ากัน

ความยาวสูงสุดของรหัสผ่านคือจำนวน อักขระที่ระบุโดยแอ็ตทริบิวต์ **PW_PASSLEN** จำนวน อักขระที่ใช้เมื่อสร้างค่ารหัสผ่านที่เก็บจะขึ้นกับ อัลกอริทึมรหัสผ่านที่ใช้บนระบบ อัลกอริทึม รหัสผ่านถูกกำหนดในไฟล์ /etc/security/pwdalg.cfg และอัลกอริทึมรหัสผ่านดีฟอลต์ที่จะใช้สามารถกำหนดค่า ผ่านแอ็ตทริบิวต์ **pwd_algorithm** ในไฟล์ /etc/security/login.cfg ค่าของแอ็ตทริบิวต์ **minalpha** ที่เพิ่มในค่า ของแอ็ตทริบิวต์ **minother** ต้องไม่มากกว่าแอ็ตทริบิวต์ **PW_PASSLEN** ถ้าค่าของแอ็ตทริบิวต์ **minalpha** ที่เพิ่มในค่าแอ็ตทริบิวต์ **minother** สูงกว่า แอ็ตทริบิวต์ **PW_PASSLEN** ค่าของแอ็ตทริบิวต์ **minother** จะถูกลดเป็นค่าของแอ็ตทริบิวต์ **PW_PASSLEN** ที่น้อยกว่า ค่าของแอ็ตทริบิวต์ **minalpha**

ถ้าค่าของทั้ง แอ็ตทริบิวต์ **histexpire** และแอ็ตทริบิวต์ **histsize** ถูกตั้งค่า ระบบจะเก็บรหัสผ่านตามจำนวนที่ต้องการเพื่อให้เป็นไปตาม เงื่อนไขทั้งสอง โดยขีดจำกัดระบบสูงสุดคือ 50 รหัสผ่านต่อหนึ่งผู้ใช้ รหัสผ่านว่างไม่สามารถเก็บได้

คุณสามารถแก้ไขไฟล์ /etc/security/user เพื่อรวมค่าดีฟอลต์ที่คุณต้องการใช้จัดการรหัสผ่านผู้ใช้ อีกทางหนึ่ง คุณสามารถเปลี่ยนค่าแอ็ตทริบิวต์โดยใช้คำสั่ง **chuser**

คำสั่ง อื่นที่สามารถใช้กับไฟล์นี้ได้คือคำสั่ง **mkuser, lsuser** และ **rmuser** คำสั่ง **mkuser** สร้างรายการสำหรับผู้ใช้ใหม่ แต่ละรายในไฟล์ /etc/security/user และกำหนดค่าเริ่มต้น สำหรับแอ็ตทริบิวต์ด้วยคุณสมบัติที่กำหนดในไฟล์ /usr/lib/security/mkuser.default ในการแสดงแอ็ตทริบิวต์และค่า ให้ใช้คำสั่ง **lsuser** ในการลบผู้ใช้ออก ให้ใช้คำสั่ง **rmuser**

การสนับสนุนรหัสผ่านที่มีความยาวมากกว่า 8 อักขระ และ Loadable Password Algorithm:

การพัฒนาในฮาร์ดแวร์คอมพิวเตอร์เมื่อเร็วๆ นี้ทำให้การเข้ารหัสผ่าน UNIX แบบดั้งเดิมเสี่ยงต่อการโจมตีรหัสผ่านแบบ brute-force อัลกอริทึม ที่มีจุดอ่อนด้านการเข้ารหัสอาจนำไปสู่การกู้คืนแม้แต่รหัสผ่านที่คาดเดายาก AIX สนับสนุน Loadable Password Algorithm (LPA), ซึ่งจัดเตรียมกลไกการแฮชรหัสผ่านที่มีความปลอดภัย

ฟังก์ชันรหัสผ่าน crypt แบบเดิม:

กลไกการพิสูจน์ตัวตน AIX มาตรฐาน ใช้ฟังก์ชันแฮชทางเดียว เรียกว่า **crypt** เพื่อการพิสูจน์ตัวตน ผู้ใช้ ฟังก์ชัน **crypt** เป็นอัลกอริทึม DES ที่ถูกดัดแปลง โดยดำเนิน การเข้ารหัสทางเดียวของอาร์เรย์ข้อมูลคงที่ กับรหัสผ่านที่กำหนดและ Salt

ฟังก์ชัน **crypt** ใช้เฉพาะอักขระแปดตัวแรกจากสตริงรหัสผ่าน รหัสผ่านของผู้ใช้ถูกตัดท้ายให้เหลือแปดอักขระ ถ้า รหัสผ่านมีน้อยกว่าแปดอักขระ จะถูกเสริมด้วยบิตศูนย์ ทางด้านขวา คีย์ 56-bit DES ที่สืบทอดโดยใช้ 7 บิตจากแต่ละ อักขระ

Salt คือสตริงสองอักขระ (12 บิตของ Salt ถูกใช้เพื่อ ทำให้อัลกอริทึม DES มีความซับซ้อน) เลือกจากชุดอักขระ "**A-Z**", "**a-z**","**0-9**","."(เครื่องหมายจุด) และ "**/**" Salt ถูกใช้เพื่อสร้างความหลากหลายให้กับอัลกอริทึมการแฮช ดังนั้นรหัสผ่านข้อความชัดเจน เดียวกันสามารถสร้างการเข้ารหัสรหัสผ่านที่เป็นไปได้ 4,096 แบบ การดัดแปลง อัลกอริทึม DES, การสลับบิต i และ i+24 ในเอาต์พุต DES E-Box เมื่อ บิต i ถูกเซ็ตใน Salt, และยังทำให้ฮาร์ดแวร์การเข้ารหัส DES ไร้ประโยชน์สำหรับการเดารหัสผ่าน

บล็อก 64-bit all-bits-zero ถูกเข้ารหัส 25 ครั้งด้วย คีย์ DES เอาต์พุตสุดท้ายคือ 12-บิต salt ต่อด้วยค่า 64-บิตที่เข้ารหัส ค่า 76-บิตผลลัพธ์ถูกจัดโค้ดใหม่เป็นอักขระ ASCII 13 ตัวที่พิมพ์ได้ในฟอร์มของ base64

อัลกอริทึมการแฮชรหัสผ่าน:

อัลกอริทึมการแฮชเช่น MD5 เจาะได้ยากกว่าฟังก์ชัน **crypt** อัลกอริทึมนี้จัดให้มีกลไกที่ยากต่อการโจมตีที่ใช้การเดารหัสผ่านโดยใช้คำที่ใช้ทั่วไป เนื่องจากรหัสผ่านทั้งคำถูกใช้เพื่อสร้างการแฮช จึงไม่มีข้อจำกัดด้านความยาว เมื่อใช้อัลกอริทึมการแฮชรหัสผ่านเพื่อเข้ารหัส รหัสผ่าน

Loadable Password Algorithm:

AIX 6.1 และ ใหม่กว่าใช้กลไก Loadable Password Algorithm (LPA) ที่ สามารถนำอัลกอริทึมการเข้ารหัสรหัสผ่านใหม่ไปใช้ได้ง่าย

อัลกอริทึมการเข้ารหัสรหัสผ่านที่สนับสนุนแต่ละวิธีถูกนำไปใช้เป็น โหลดโมดูล LPA ที่ถูกโหลดในตอนรันไทม์เมื่อจำเป็นต้องใช้ อัลกอริทึม LPAs ที่สนับสนุนรวมถึงแอ็ตทริบิวต์จะถูกกำหนดไว้ในไฟล์คอนฟิกูเรชันระบบ `/etc/security/pwdalg.cfg`

ผู้ดูแลระบบสามารถตั้งค่ากลไกการเข้ารหัสรหัสผ่านของทั้งระบบ ที่ใช้ LPA ที่เจาะจงเพื่อเข้ารหัสรหัสผ่าน หลังจากกลไกรหัสผ่าน ของทั้งระบบถูกเปลี่ยนแปลง รหัสผ่านที่ถูกเข้ารหัสโดยใช้กลไก การเข้ารหัสรหัสผ่านที่เลือกก่อนหน้านี้ (เช่นฟังก์ชัน **crypt**) ยังคงได้รับการสนับสนุน

การสนับสนุนรหัสผ่านที่ยาวกว่าแปดอักขระ:

LPAs ทั้งหมดที่ถูกนำไปใช้สำหรับ AIX 6.1 และใหม่กว่าจะสนับสนุนรหัสผ่านที่ยาวเกินแปดอักขระ ขีดจำกัดด้านความยาวของรหัสผ่านเปลี่ยนแปลงเมื่อ LPAs ต่างกัน ความยาวรหัสผ่านสูงสุด ที่สนับสนุนคือ 255 อักขระ

ไฟล์คอนฟิกูเรชัน LPA:

ไฟล์คอนฟิกูเรชัน LPA คือ `/etc/security/pwdalg.cfg` ไฟล์นี้เป็นไฟล์ stanza ที่กำหนดแอ็ตทริบิวต์ของ LPAs ที่สนับสนุน

แอ็ตทริบิวต์ LPA ต่อไปนี้ถูกกำหนดให้ไฟล์คอนฟิก:

- พาธไปยังโมดูล LPA
- แฟล็กทางเลือกที่ถูกส่งไปยังโมดูล LPA ตอนรันไทม์

แอ็ตทริบิวต์ LPA ที่กำหนดในไฟล์คอนฟิกูเรชันสามารถเข้าถึงด้วย อินเตอร์เฟส **getconfattr** และ **setconfattr**

stanza ตัวอย่างใน `/etc/security/pwdalg.cfg` กำหนด LPA ชื่อ **ssha256**:

```
ssha256:
    lpa_module = /usr/lib/security/ssha
    lpa_options = algorithm=sha256
```

อัลกอริทึมรหัสผ่านระบบ:

ผู้ดูแลระบบสามารถตั้งค่าอัลกอริทึมรหัสผ่านทั้งระบบ โดยการเลือก LPA เป็นอัลกอริทึมการแฮชรหัสผ่าน โดยสามารถมีได้เพียงหนึ่งอัลกอริทึมรหัสผ่านระบบที่ใช้งานในแต่ละครั้งเท่านั้น อัลกอริทึมรหัสผ่าน ระบบถูกกำหนดโดยแอ็ตทริบิวต์ระบบ **pwd_algorithm** ใน **usw** stanza ในไฟล์ `/etc/security/login.cfg`

ค่าที่ถูกต้องสำหรับแอ็ตทริบิวต์ **pwd_algorithm** ในไฟล์ /etc/security/login.cfg คือชื่อ LPA stanza ที่ถูกกำหนดในไฟล์ /etc/security/pwdalg.cfg อีกค่าที่ถูกต้องสำหรับแอ็ตทริบิวต์ **pwd_algorithm** คือ **crypt** ซึ่งอ้างอิงการเข้ารหัส **crypt** แบบดั้งเดิม ถ้าแอ็ตทริบิวต์ **pwd_algorithm** ถูกละเว้นในคอนฟิกไฟล์ **crypt** ถูกใช้เป็น ค่าดีฟอลต์

ตัวอย่างต่อไปนี้ของไฟล์ /etc/security/login.cfg ใช้ **ssha256** LPA เป็นอัลกอริทึมการเข้ารหัสรหัสผ่านทั้งระบบ

```
 ... ...
usw:
   shells = /bin/sh,/bin/bsh,/bin/csh,/bin/ksh,/bin/tsh,/bin/ksh93
   maxlogins = 32767
   logintimeout = 60
   maxroles = 8
   auth_type = STD_AUTH
   pwd_algorithm = ssha256
 ... ...
```

อัลกอริทึมรหัสผ่านระบบมีผลสำหรับรหัสผ่านที่เพิ่งสร้าง ใหม่และรหัสผ่านที่ถูกเปลี่ยนเท่านั้น หลังการโอนย้าย รหัสผ่านใหม่และการเปลี่ยน รหัสผ่านให้ภายหลังจะใช้อัลกอริทึมรหัสผ่านระบบ รหัสผ่านที่มีอยู่แล้วก่อนเลือกใช้อัลกอริทึมรหัสผ่านระบบที่สร้างขึ้นโดยฟังก์ชัน **crypt** มาตรฐานหรือ โดยโมดูล LPA ที่สนับสนุนอื่นๆ จะยังคงใช้ได้บนระบบ ดังนั้น รหัสผ่านที่ผสมกันที่ถูกสร้างขึ้นโดยใช้ LPAs ต่างกันสามารถมีอยู่ร่วมกัน บนระบบ

การตั้งค่าอัลกอริทึมรหัสผ่านระบบ:

ผู้ดูแลระบบสามารถใช้คำสั่ง **chsec** เพื่อตั้งค่าอัลกอริทึมรหัสผ่านระบบ หรือใช้เอดิเตอร์เช่น **vi** เพื่อ แก้ไขแอ็ตทริบิวต์ **pwd_algorithm** ด้วยตนเองในไฟล์ /etc/security/login.cfg

แนะนำให้คุณใช้คำสั่ง **chsec** เพื่อ ตั้งค่าอัลกอริทึมรหัสผ่านระบบ เนื่องจากคำสั่ง **chsec** จะตรวจสอบนิยามของ LPA ที่ระบุโดยอัตโนมัติ

การใช้คำสั่ง chsec

รันคำสั่งต่อไปนี้ เพื่อตั้งค่า **smd5** LPA เป็นโมดูลการเข้ารหัสรหัสผ่านทั้งระบบ:

```
chsec -f /etc/security/login.cfg -s usw -a pwd_algorithm=smd5
```

เมื่อคุณใช้คำสั่ง **chsec** เพื่อแก้ไขแอ็ตทริบิวต์ **pwd_algorithm** คำสั่ง **chsec** จะตรวจสอบไฟล์ /etc/security/pwdalg.cfg เพื่อยืนยัน LPA ที่ระบุ คำสั่ง **chsec** จะล้มเหลวถ้า การตรวจสอบนี้ล้มเหลว

การใช้เอดิเตอร์

ถ้าคุณใช้เอดิเตอร์เพื่อเปลี่ยนค่า แอ็ตทริบิวต์ **pwd_algorithm** ในไฟล์ /etc/security/login.cfg ด้วยตนเอง ตรวจสอบให้แน่ใจว่าค่าที่ระบุคือชื่อของของ stanza ที่ถูกกำหนดใน ไฟล์ /etc/security/pwdalg.cfg

การขยายข้อจำกัดรหัสผ่าน:

กฎที่ใช้โดยโปรแกรมรหัสผ่านเพื่อยอมรับหรือปฏิเสธรหัสผ่าน (ข้อจำกัดของการประกอบขึ้นเป็นรหัสผ่าน) สามารถถูกขยายได้โดยผู้ดูแลระบบ เพื่อให้มีข้อจำกัดที่เจาะจงสำหรับไซต์

ข้อจำกัดถูกขยายโดยการเพิ่มวิธี ซึ่งจะถูกเรียกใช้ ระหว่างการเปลี่ยนรหัสผ่าน แอ็ตทริบิวต์ **pwdchecks** ในไฟล์ /etc/security/user จะระบุวิธีที่จะเรียกใช้

ตั้งแต่ *AIX เวอร์ชัน 6.1 ข้อมูลอ้างอิงด้านเทคนิค* จะมีคำอธิบายของ **pwdrestrict_method** อินเตอร์เฟสรูทีนย่อย ที่วิธีการจำกัดรหัสผ่านที่ระบุต้องทำตาม ในการ ขยายข้อจำกัดของการประกอบขึ้นเป็นรหัสผ่านอย่างถูกต้อง ผู้ดูแลระบบ ต้องตั้งโปรแกรมอินเตอร์เฟสนี้เมื่อทำการกำหนดวิธีจำกัดรหัสผ่าน ใช้ ความระมัดระวังในการขยายข้อจำกัดการประกอบขึ้นเป็นรหัสผ่าน ข้อกำหนดการขยายเหล่านี้ มีผลโดยตรงกับคำสั่ง **login**, คำสั่ง **passwd**, คำสั่ง **su** และโปรแกรมอื่นๆ ความปลอดภัยของระบบสามารถถูกทำลายได้อย่างง่ายดายโดยโค้ดที่เป็นอันตรายหรือมีข้อบกพร่อง

การพิสูจน์ตัวตนผู้ใช้

Identification และการพิสูจน์ตัวตนถูกใช้เพื่อสร้าง การระบุผู้ใช้

ผู้ใช้แต่ละคนจำเป็นต้องล็อกอินเข้าสู่ระบบ ผู้ใช้ระบุชื่อผู้ใช้ของ บัญชีผู้ใช้และรหัสผ่าน ถ้ามีหนึ่งบัญชีผู้ใช้ (ในระบบที่มีความปลอดภัย บัญชีผู้ใช้ทั้งหมดต้องมีรหัสผ่านหรือไม่แล้วจะเป็นบัญชีผู้ใช้ที่ไม่ถูกต้อง) ถ้ารหัสผ่าน ถูกต้อง ผู้ใช้ล็อกอินเข้าสู่บัญชีผู้ใช้นั้น ผู้ใช้ได้รับสิทธิการเข้าถึง และ privileges ของบัญชีผู้ใช้นั้น ไฟล์ /etc/passwd และ /etc/security/passwd เก็บรักษารหัสผ่านผู้ใช้

โดยดีฟอลต์ผู้ใช้ถูกกำหนดในรีจิสทรี Files หมายความว่าบัญชีผู้ใช้ และข้อมูลกลุ่มถูกเก็บในไฟล์ flat-ASCII ด้วยการนำ ปลั๊กอินโหลดโมดูลมาใช้ ผู้ใช้สามารถถูกกำหนดในรีจิสทรี อื่นได้เช่นกัน ตัวอย่างเช่น เมื่อปลั๊กอินโมดูล LDAP ถูกใช้สำหรับการดูแลผู้ใช้ ข้อกำหนดผู้ใช้จะถูกเก็บในที่เก็บ LDAP ในกรณีนี้ จะไม่มีรายการสำหรับผู้ใช้ในไฟล์ /etc/security/user (มีข้อยกเว้นสำหรับแอ็ตทริบิวต์ผู้ใช้ **SYSTEM** และ **registry**) เมื่อชุดโหลดโมดูล (ตัวอย่าง โหลดโมดูลที่มีการพิสูจน์ตัวตนและ ส่วนของฐานข้อมูล) ถูกใช้สำหรับการดูแลผู้ใช้ ส่วนฐานข้อมูลตรวจสอบ ว่า AIX ข้อมูลบัญชีผู้ใช้ ถูกดูแลอย่างไร และส่วนการพิสูจน์ตัวตนอธิบายการพิสูจน์ตัวตน และรหัสผ่านที่เกี่ยวกับการดูแล ส่วนการพิสูจน์ตัวตนอาจอธิบาย แอ็ตทริบิวต์การดูแลบัญชีผู้ใช้แบบ authentication-specific เช่นกันโดยนำ อินเตอร์เฟสโหลดโมดูลมาใช้ (newuser, getentry, putentry และอื่นๆ)

เมธอดการพิสูจน์ตัวตนถูกพิสูจน์โดย **SYSTEM** และ ริจิสทรี แอ๊ตทริบิวต์ซึ่งถูกกำหนดใน /etc/security/user file. ผู้ดูแลระบบสามารถกำหนดแอ็ตทริบิวต์ authcontroldomain ของไฟล์ /etc/security/login.cfg เพื่อบังคับ **SYSTEM** และแอ็ตทริบิวต์รีจิสทรีที่จะถูกเรียกข้อมูลจาก authcontroldomain เช่น authcontroldomain=LDAP บังคับระบบ ให้ค้นหา **SYSTEM** ของผู้ใช้ และรีจิสทรีจาก LDAP เพื่อกำหนด เมธอดการพิสูจน์ตัวตนที่ถูกใช้สำหรับผู้ใช้ มี ข้อยกเว้นสำหรับผู้ใช้ที่กำหนดแบบโลคัลที่การตั้งค่า authcontroldomain ถูกละเว้น และ **SYSTEM** และรีจิสทรีจะถูกเรียกข้อมูล จากไฟล์ /etc/security/user

โทเค็นอื่นที่ยอมรับสำหรับแอ๊ตทริบิวต์ authcontroldomain คือ ไฟล์หรือชื่อ stanza จากไฟล์ /usr/lib/security/methods.cfg

ค่าของแอ็ตทริบิวต์ **SYSTEM** ถูกกำหนดผ่านไวยากรณ์โดยการใช้ไวยากรณ์นี้ ผู้ดูแลระบบสามารถรวมหนึ่งเมธอดหรือมากกว่านั้น เพื่อพิสูจน์ตัวตนผู้ใช้กับระบบ โทเค็นเมธอดที่รู้จักกันดีคือ compat, DCE, files และ NONE

ค่าดีฟอลต์ของระบบคือ compat ค่าดีฟอลต์ SYSTEM=compat แจ้ง แก่ระบบให้ใช้ฐานข้อมูลโลคัลสำหรับการพิสูจน์ตัวตน และถ้าไมพบข้อมูล จะใช้ฐานข้อมูล Network Information Services (NIS) โทเค็น files ระบุว่าเฉพาะไฟล์โลคัลเท่านั้นที่จะถูกใช้ระหว่างการพิสูจน์ตัวตน โดยที่ SYSTEM=DCE มีผลใน ลำดับการพิสูจน์ตัวตน DCE

โทเค็น NONE ปิดเมธอดการพิสูจน์ตัวตน เมื่อต้องการปิด การพิสูจน์ตัวตนทั้งหมดโทเค็น NONE ต้องมีอยู่ในบรรทัด SYSTEM และ auth1 ของ stanza ของผู้ใช้

คุณสามารถระบุสองเมธอดหรือมากกว่านั้นและรวมกับโลจิคัล constructors AND และ OR ตัวอย่าง SYSTEM=DCE OR compat ชี้ว่า ผู้ใช้ได้รับอนุญาตให้ล็อกอินถ้า DCE หรือการพิสูจน์ตัวตน โลคัล (crypt()) สำหรับในลำดับที่กำหนดนี้

ในรูปแบบเดียวกันผู้ดูแลระบบสามารถใช้ชื่อโหลดโมดูลการพิสูจน์ตัวตน สำหรับแอ็ตทริบิวต์ **SYSTEM** ตัวอย่าง เมื่อแอ็ตทริบิวต์ **SYSTEM** ถูกเซ็ตเป็น SYSTEM=KRB5files OR compat, AIX โฮสต์จะใช้ลำดับ Kerberos สำหรับการพิสูจน์ตัวตนและถ้าล้มเหลว จะใช้การพิสูจน์ตัวตน AIX มาตรฐาน

แอ็ตทริบิวต์ **SYSTEM** และ **registry** ถูกเก็บอยู่บนระบบไฟล์โลคัลเสมอ ในไฟล์ /etc/security/user ถ้าผู้ใช้ AIX ถูกกำหนดใน LDAP และ แอ็ตทริบิวต์ **SYSTEM** และ **registry** ถูกเซ็ตตามลำดับ ผู้ใช้จะมีรายการในไฟล์ /etc/security/user

แอ็ตทริบิวต์ **SYSTEM** และ **registry** ของผู้ใช้สามารถถูกเปลี่ยน โดยใช้คำสั่ง **chuser**

โทเค็นที่ยอมรับได้สำหรับแอ็ตทริบิวต์ **SYSTEM** สามารถถูกกำหนดในไฟล์ /usr/lib/security/methods.cfg

หมายเหตุ: ผู้ใช้ root ถูกพิสูจน์ตัวตนเสมอตามวิธีของไฟล์ความปลอดภัย ของระบบโลคัล รายการแอ็ตทริบิวต์ **SYSTEM** สำหรับผู้ใช้ root ถูกเซ็ตเป็นพิเศษเป็น SYSTEM=compat ในไฟล์/etc/security/user

เมธอดทางเลือกของการพิสูจน์ตัวตนถูกรวมไว้ในระบบตามวิธี ของแอ็ตทริบิวต์ **SYSTEM** ที่แสดงใน /etc/security/user ตัวอย่าง Distributed Computing Environment (DCE) ต้องการการพิสูจน์รหัสผ่าน แต่ตรวจสอบรหัสผ่านเหล่านี้ในวิธีที่ต่างจากโมเดล การเข้ารหัสที่ใช้ใน etc/passwd และ /etc/security/passwd ผู้ใช้ซึ่งพิสูจน์ตามวิธี DCE มี stanza ใน /etc/security/user เซ็ต เป็น SYSTEM=DCE ได้

ค่าแอ็ตทริบิวต์ **SYSTEM** อื่นคือ **compat, files** และ **NONE** โทเค็น compat ถูกใช้เมื่อการค้นหาชื่อ (และการพิสูจน์ตัวตน ที่ตามมา) ดำเนินในฐานข้อมูลโลคัล และถ้าไม่พบ จะค้นหาในฐานข้อมูล Network Information Services (NIS) โทเค็น files ระบุ ว่าเฉพาะไฟล์โลคัลเท่านั้นที่ถูกใช้ระหว่างการพิสูจน์ตัวตน สุดท้ายโทเค็น NONE ปิดเมธอดการพิสูจน์ตัวตน เมื่อต้องการปิด การพิสูจน์ตัวตนทั้งหมดโทเค็น NONE ต้องมีอยู่ในบรรทัด **SYSTEM** และ **auth1** ของ stanza ของผู้ใช้

โทเค็นอื่นที่ยอมรับได้สำหรับแอ็ตทริบิวต์ **SYSTEM** สามารถ ถูกกำหนดใน /usr/lib/security/methods.cfg

หมายเหตุ: ผู้ใช้ root ถูกพิสูจน์ตัวตนเสมอตามวิธีของไฟล์ความปลอดภัย ของระบบโลคัล รายการแอ็ตทริบิวต์ **SYSTEM** สำหรับผู้ใช้ root ถูกเซ็ตเป็นพิเศษเป็น SYSTEM=compat ใน /etc/security/user

ดูที่ *การจัดการระบบปฏิบัติการและอุปกรณ์* สำหรับข้อมูลเพิ่มเติม เกี่ยวกับการป้องกันรหัสผ่าน

ล็อกอิน ID ผู้ใช้

เหตุการณ์การตรวจสอบทั้งหมดที่บันทึกสำหรับ ผู้ใช้นี้ถูกเลเบลด้วย ID นี้และสามารถถูกตรวจสอบเมื่อคุณสร้างบันทึก การตรวจสอบ ดูที่ *การจัดการระบบปฏิบัติการและอุปกรณ์* สำหรับข้อมูลเพิ่มเติม เกี่ยวกับล็อกอิน ID ผู้ใช้

แอ็ตทริบิวต์ผู้ใช้และกลุ่มที่สนับสนุนโดย Authentication Load Modules

ชุดของแอ็ตทริบิวต์ สัมพันธ์กับผู้ใช้และสัมพันธ์กับกลุ่ม ถูก ใช้ใน identification และการพิสูจน์ตัวตนใน AIX

ตารางดังต่อไปนี้แสดงแอ็ตทริบิวต์ ผู้ใช้และกลุ่มส่วนใหญ่ ตามที่แสดงและยังแสดงการสนับสนุนจากโหลดโมดูลต่างๆ สำหรับแอ็ตทริบิวต์เหล่านี้ แต่ละแถวของตารางตรงกับแอ็ตทริบิวต์ และแต่ละคอลัมน์แสดงโหลดโมดูล แอ็ตทริบิวต์ที่สนับสนุนโดย โหลดโมดูลถูกแสดงด้วย ใช่ ในคอลัมน์โหลดโมดูล

หมายเหตุ: PKI และ Kerberos เป็นโมดูลสำหรับการพิสูจน์ตัวตนเท่านั้น และต้องถูกรวมกับโมดูลฐานข้อมูล (เช่น LOCAL หรือ LDAP) ซึ่งสนับสนุนแอ็ตทริบิวต์เพิ่มเติม (ส่วนขยาย) นอกจากที่จัดเตรียมโดย LOCAL หรือ LDAP เครื่องหมายถูกแสดงเฉพาะกับแอ็ตทริบิวต์ส่วนขยายเหล่านี้ สำหรับโมดูลเหล่านี้ แม้ว่าแอ็ตทริบิวต์อื่นได้รับได้โดยใช้ LOCAL หรือ LDAP

ตารางที่ 7. แอ็ตทริบิวต์ผู้ใช้และการสนับสนุน Authentication Load Module

แอ็ตทริบิวต์ผู้ใช้	Local	NIS	LDAP	PKI	Kerberos
account_locked	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
admgroups	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
admin	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
auditclasses	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
auth_cert	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่	ไม่ใช่
auth_domain	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
auth_name	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
auth1 **หมายเหตุ:** แอ็ตทริบิวต์ **auth1** ไม่ได้รับการยอมรับ และไม่ควรใช้	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
auth2 **หมายเหตุ:** แอ็ตทริบิวต์ **auth2** ไม่ได้รับการยอมรับ และไม่ควรใช้	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
capabilities	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
core	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
core_compress	ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่
core_hard	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
core_naming	ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่
core_path	ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่
core_pathname	ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่
cpu	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
daemon	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
ข้อมูล	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
data_hard	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
dce_export	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
dictionlist	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
expires	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ใช่
แฟล็ก	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ใช่
fsize	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
fsize_hard	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่

ตารางที่ 7. แอ็ตทริบิวต์ผู้ใช้และการสนับสนุน Authentication Load Module (ต่อ)

แอ็ตทริบิวต์ผู้ใช้	Local	NIS	LDAP	PKI	Kerberos
funcmode	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
gecos	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
groups	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
groupsids	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
histexpire	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
home	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
host_last_login	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
host_last_unsuccessful_login	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
hostsallowedlogin	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
hostsdeniedlogin	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
id	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
krb5_attributes	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่
krb5_kvno	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่
krb5_last_pwd_change	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่
krb5_max_renewable_life	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่
krb5_mknvo	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่
krb5_mod_date	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่
krb5_mod_name	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่
krb5_names	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่
krb5_principal	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่
krb5_principal_name	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่
krb5_realm	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ใช่
lastupdate	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
login	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
loginretries	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
logintimes	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
maxage	ใช่	ใช่	ใช่	ไม่ใช่	ใช่
maxexpired	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
maxrepeats	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
maxulogs	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
minage	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่

ตารางที่ 7. แอ็ตทริบิวต์ผู้ใช้และการสนับสนุน Authentication Load Module (ต่อ)

แอ็ตทริบิวต์ผู้ใช้	Local	NIS	LDAP	PKI	Kerberos
minalpha	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
mindiff	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
mindigit	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
minlen	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
minloweralpha	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
minother	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
minspecialchar	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
minupperalpha	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
nofiles	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
nofiles_hard	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
รหัสผ่าน	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
pgid	ใช่	ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่
pgrp	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
projects	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
pwdchecks	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
pwdwarntime	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
rcmds	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
registry	ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่
rlogin	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
roles	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
rss	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
rss_hard	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
หน้าจอ	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
shell	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
spassword	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
stack	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
stack_hard	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
su	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
sugroups	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
sysenv	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
SYSTEM	ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่	ไม่ใช่

ตารางที่ 7. แอ็ตทริบิวต์ผู้ใช้และการสนับสนุน Authentication Load Module (ต่อ)

แอ็ตทริบิวต์ผู้ใช้	Local	NIS	LDAP	PKI	Kerberos
time_last_login	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
time_last_unsuccessful_login	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
tpath	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
tty_last_login	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
tty_last_unsuccessful_login	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
ttys	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
umask	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
unsuccessful_login_count	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
unsuccessful_login_times	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
usrenv	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่

ตารางที่ 8. แอ็ตทริบิวต์กลุ่มและการสนับสนุน Authentication Load Module

แอ็ตทริบิวต์ผู้ใช้	Local	NIS	LDAP	PKI	Kerberos
admin	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
adms	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
dce_export	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
id	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่
primary	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
projects	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
หน้าจอ	ใช่	ไม่ใช่	ใช่	ไม่ใช่	ไม่ใช่
users	ใช่	ใช่	ใช่	ไม่ใช่	ไม่ใช่

ภาพรวมระบบโควต้าดิสก์

ระบบโควต้าดิสก์อนุญาตให้ผู้ดูแลระบบควบคุม จำนวนไฟล์และบล็อกข้อมูลที่สามารถจัดสรรให้แก่ผู้ใช้และกลุ่ม

แนวความคิดเกี่ยวกับระบบโควต้าดิสก์:

ระบบโควต้าดิสก์ซึ่งยึดตาม Berkeley Disk Quota System จัดให้มีวิธีที่มีประสิทธิภาพในการควบคุมการใช้พื้นที่ดิสก์ ระบบโควต้า สามารถกำหนดให้แก่ผู้ใช้หรือกลุ่มแต่ละราย และคงไว้สำหรับแต่ละ journaled file system (JFS และ JFS2)

ระบบโควต้าดิสก์สร้างการจำกัดขึ้นตามพารามิเตอร์ต่อไปนี้ ที่สามารถเปลี่ยนแปลงด้วยคำสั่ง **edquota** สำหรับไฟล์ JFS และคำสั่ง **j2edlimit** สำหรับระบบไฟล์ JFS2:

- soft limits ของผู้ใช้หรือของกลุ่ม
- hard limits ของผู้ใช้หรือของกลุ่ม
- ช่วงเวลาผ่อนผันโควต้า

soft limit กำหนดจำนวนบล็อกของดิสก์หรือไฟล์ 1 KB ที่ผู้ใช้หรือกลุ่มจะได้รับอนุญาตให้ใช้ระหว่างการดำเนินการปกติ *hard limit* กำหนดจำนวนบล็อกดิสก์หรือไฟล์สูงสุดที่ ผู้ใช้สามารถสะสมได้ภายใต้โควต้าดิสก์ที่สร้างขึ้น *ช่วงเวลาผ่อนผัน โควต้า* อนุญาตให้ผู้ใช้ใช้เกิน soft limit ได้ในช่วงเวลาสั้นๆ (ค่าดีฟอลต์คือหนึ่งสัปดาห์) ถ้าผู้ใช้ไม่สามารถลดจำนวนการใช้งานให้ต่ำกว่า soft limit ได้ระหว่างช่วงเวลาที่ระบุ ระบบจะตีความว่า soft limit เป็นค่าสูงสุดของการจัดสรรที่อนุญาต และไม่มีการจัดสรรหน่วยเก็บเพิ่มเติม ให้แก่ผู้ใช้ ผู้ใช้สามารถตั้งค่าเงื่อนไขนี้ใหม่โดยการลบไฟล์เพื่อลด การใช้งานให้อยู่ต่ำกว่า soft limit

ระบบโควต้าดิสก์บันทึกการติดตามโควต้าของผู้ใช้และกลุ่มในไฟล์ `quota.user` และ `quota.group` ที่อยู่ในไดเร็กทอรี root ของระบบไฟล์ที่เปิดใช้งานโดยมีโควต้า ไฟล์ เหล่านี้ถูกสร้างด้วยคำสั่ง **quotacheck** และ **edquota** และอ่านได้โดยใช้คำสั่ง โควต้า

การกู้คืนจากสภาวะใช้เกินโควต้า:

คุณสามารถกู้คืนจากสภาวะใช้เกินโควต้าโดยการลดการใช้งาน ระบบไฟล์

ในการลดการใช้งานระบบไฟล์เมื่อคุณใช้เกินขีดจำกัดโควต้า คุณสามารถใช้วิธีต่อไปนี้:

- หยุดการทำงานกระบวนการปัจจุบันที่เป็นสาเหตุให้ระบบ ถึงขีดจำกัด ลบไฟล์ที่เกินเพื่อให้โควต้าต่ำกว่าขีดจำกัด และลองใช้โปรแกรมที่ล้มเหลวอีกครั้ง
- ถ้าคุณกำลังทำงานเอดิเตอร์เช่น vi ให้ใช้ escape sequence เชลล์ เพื่อตรวจสอบพื้นที่ไฟล์ของคุณ ลบไฟล์ที่เกินออก และกลับโดยไม่สูญเสีย ไฟล์ที่แก้ไขของคุณ อีกทางหนึ่ง ถ้าคุณกำลังใช้ C หรือ Korn เชลล์ คุณ สามารถหยุดทำงานเอดิเตอร์ชั่วคราวด้วยการกดคีย์ Ctrl-Z ออกคำสั่ง ระบบไฟล์ จากนั้นกลับด้วยคำสั่ง **fg** (foreground)
- เขียนไฟล์ลงในระบบไฟล์ชั่วคราวที่ยังไม่เกินขีดจำกัดโควต้า ลบไฟล์ที่เกิน และส่งกลับไฟล์ไปยังระบบไฟล์ ที่ถูกต้อง

การตั้งค่าระบบโควต้าดิสก์:

โดยทั่วไป เฉพาะระบบไฟล์ที่มีโฮมไดเร็กทอรีและไฟล์ผู้ใช้ เท่านั้นที่จำเป็นต้องมีโควต้าดิสก์

พิจารณาการนำระบบโควต้าดิสก์ไปใช้ภายใต้เงื่อนไข ต่อไปนี้:

- ระบบของคุณมีพื้นที่ดิสก์จำกัด
- คุณต้องการมีการรักษาความปลอดภัยระบบไฟล์มากขึ้น
- ระดับการใช้ดิสก์ของคุณมีขนาดใหญ่ เช่นในหลายๆ มหาวิทยาลัย

ถ้าเงื่อนไขเหล่านี้ไม่เข้ากับสภาวะแวดล้อมของคุณ คุณอาจ ไม่ต้องการสร้างขีดจำกัดการใช้งานดิสก์โดยการใช้ระบบโควต้าดิสก์

ระบบ โควต้าดิสก์สามารถใช้กับระบบไฟล์ที่มีการทำบันทึกประจำวันเท่านั้น

หมายเหตุ: อย่า สร้างโควต้าดิสก์สำหรับระบบไฟล์ `/tmp`

ในการ ตั้งค่าระบบโควต้าดิสก์ ใช้ขั้นตอนต่อไปนี้:

1. ล็อกอินด้วยสิทธิ root
2. พิจารณาว่าระบบไฟล์ใดที่จำเป็นต้องใช้โควต้า

   หมายเหตุ: เนื่องจากมีหลายเอดิเตอร์ และยูทิลิตีระบบที่สร้างไฟล์ชั่วคราวในระบบไฟล์ `/tmp` จึงจำเป็นต้องไม่มีการทำโควต้า

3. ใช้คำสั่ง **chfs** เพื่อรวมแอ็ตทริบิวต์การตั้งค่าโควต้า **userquota** และ **groupquota** ในไฟล์ /etc/filesystems ตัวอย่างต่อไปนี้ใช้คำสั่ง **chfs** เพื่อเปิดใช้งานโควต้า ผู้ใช้บนระบบไฟล์ /home:

   ```
   chfs -a "quota = userquota" /home
   ```

   ในการ เปิดใช้ทั้งโควต้าผู้ใช้และกลุ่มบนระบบไฟล์ /home พิมพ์:

   ```
   chfs -a "quota = userquota,groupquota" /home
   ```

   รายการ ที่สัมพันธ์กันในไฟล์ /etc/filesystems ถูกแสดง ดังนี้:

   ```
   /home:
   dev        = /dev/hd1
   vfs        = jfs
   log        = /dev/hd8
   mount      = true
   check      = true
   quota      = userquota,groupquota
   options    = rw
   ```

4. เป็นทางเลือกระบุชื่อไฟล์โควต้าดิสก์ทางเลือก ชื่อไฟล์ quota.user และ quota.group เป็นชื่อดีฟอลต์ที่อยู่ที่ไดเร็กทอรี root ของระบบไฟล์ ที่เปิดใช้งานด้วยโควต้า คุณสามารถระบุชื่อหรือไดเร็กทอรีอื่นสำหรับไฟล์โควต้าเหล่านี้ ด้วยแอ็ตทริบิวต์ **userquota** และ **groupquota** ใน ไฟล์ /etc/filesystems

   ตัวอย่างต่อไปนี้ ใช้คำสั่ง **chfs** เพื่อสร้างโควต้าผู้ใช้และกลุ่ม สำหรับระบบไฟล์ /home และตั้งชื่อไฟล์โควต้า myquota.user และ myquota.group :

   ```
   chfs -a "userquota = /home/myquota.user" -a "groupquota = /home
                  /myquota.group" /home
   ```

   รายการที่สัมพันธ์กันในไฟล์ /etc/filesystems ถูกแสดงดังนี้:

   ```
   /home:
   dev        = /dev/hd1
   vfs        = jfs
   log        = /dev/hd8
   mount      = true
   check      = true
   quota      = userquota,groupquota
   userquota  = /home/myquota.user
   groupquota = /home/myquota.group
   options    = rw
   ```

5. ถ้าก่อนหน้านี้ไม่ถูกเมาท์ให้เมาท์ระบบไฟล์ที่ระบุ
6. ตั้งค่าขีดจำกัดโควต้าที่ต้องการสำหรับแต่ละผู้ใช้หรือแต่ละกลุ่ม ใช้คำสั่ง **edquota** เพื่อสร้างขีดจำกัดแบบ soft และ hard ของแต่ละผู้ใช้และกลุ่มสำหรับพื้นที่ดิสก์ที่อนุญาต และจำนวนไฟล์สูงสุด

   รายการตัวอย่างต่อไปนี้แสดงขีดจำกัดโควต้า สำหรับผู้ใช้ *davec*:

   ```
   Quotas for user davec:
   /home: blocks in use: 30, limits (soft = 100, hard = 150)
                  inodes in use: 73, limits (soft = 200, hard = 250)
   ```

   ผู้ใช้นี้ใช้ไป 30 KB จากพื้นที่ดิสก์สูงสุด 100 KB ด้วยไฟล์สูงสุด 200 ไฟล์ *davec* สร้างไป 73 ผู้ใช้นี้มีบัฟเฟอร์ขนาด 50 KB ของ พื้นที่ดิสก์และ 50 ที่สามารถถูกจัดสรรเป็นหน่วยเก็บชั่วคราว

   เมื่อ สร้างโควต้าดิสก์สำหรับหลายผู้ใช้ให้ใช้แฟล็ก **–p** กับคำสั่ง **edquota** เพื่อทำสำเนาโควต้าของผู้ใช้ไปยังผู้ใช้รายอื่น

ในการทำสำเนาโควต้า ที่สร้างขึ้นสำหรับผู้ใช้ *davec* ให้แก่ผู้ใช้ *nanc* พิมพ์:

```
edquota -p davec nanc
```

7. เปิดใช้งานระบบโควต้าด้วยคำสั่ง **quotaon** คำสั่ง **quotaon** จะเปิดใช้งานโควต้าสำหรับระบบไฟล์ที่ระบุ หรือสำหรับระบบไฟล์ทั้งหมดที่มีโควต้า (ดังระบุในไฟล์ /etc/filesystems) เมื่อใช้ กับแฟล็ก **-a**
8. ใช้คำสั่ง **quotacheck** เพื่อตรวจสอบความสอดคล้องกัน ของไฟล์โควต้ากับการใช้งานดิสก์จริง

หมายเหตุ: ทำขั้นตอนนี้แต่ละครั้งที่คุณ เปิดใช้งานโควต้าครั้งแรกบนระบบไฟล์และหลังจากที่คุณรีบูตระบบ คำสั่ง **quotacheck** ใช้เวลาในการรันบนระบบไฟล์ JFS นานกว่าบนระบบไฟล์ JFS2 ที่มี ขนาดเท่ากัน ถ้าโควต้าถูกเปิดใช้งานตลอดเวลาก่อนที่จะรีบูต ไม่จำเป็นต้อง รันคำสั่ง **quotacheck** บนระบบไฟล์ระหว่าง การรีบูต

ในการเปิดใช้งานการตรวจสอบนี้เพื่อเปิดใช้โควต้าระหว่างเริ่มทำงานระบบ ให้เพิ่มบรรทัดต่อไปนี้ที่ท้ายของไฟล์ /etc/rc:

```
echo " Enabling filesystem quotas "
/usr/sbin/quotacheck -a
/usr/sbin/quotaon -a
```

จำนวนกลุ่มที่อนุญาต

คุณสามารถตั้งค่าและเรียกค้นค่าจำนวนกลุ่มที่อนุญาต สำหรับ AIX 7.1. ซึ่งกำหนดจำนวนกลุ่มที่ผู้ใช้สามารถเป็นสมาชิกได้

ค่าดีฟอลต์ของจำนวนกลุ่มที่อนุญาตคือ 128 ซึ่งสามารถ ปรับได้เป็น 128 ถึง 2048 จำนวนกลุ่มที่อนุญาต จะถูกระบุด้วยพารามิเตอร์กำหนดค่าระบบ v_ngroups_allowed สำหรับอุปกรณ์ sys0 คุณสามารถเปลี่ยนแปลงหรือ เรียกค้นค่าพารามิเตอร์ v_ngroups_allowed จาก kernel หรือฐานข้อมูล ODM ค่าพารามิเตอร์ใน kernel ถูกใช้โดยระบบขณะรัน ค่าพารามิเตอร์ในฐานข้อมูล ODM จะใช้ได้หลังจากรีสตาร์ทระบบแล้ว

การเรียกค้นจำนวนกลุ่มที่อนุญาตจากฐานข้อมูล ODM: คุณต้องใช้คำสั่งหรือคำสั่งย่อยเพื่อเรียกค้นพารามิเตอร์ v_ngroups_allowed คุณต้องใช้คำสั่ง **lsattr** เพื่อเรียกค้นพารามิเตอรี v_ngroups_allowed ในฐานข้อมูล ODM นี้

คำสั่ง **lsattr** แสดงพารามิเตอร์ v_ngroups_allowed เป็นแอ็ตทริบิวต์ ngroups_allowed ตัวอย่างต่อไปนี้ แสดงวิธีการใช้คำสั่ง **lsattr** เพื่อเรียกค้นแอ็ตทริบิวต์ ngroups_allowed:

```
$lsattr -El sys0
SW_dist_intr    false           Enable SW distribution of interrupts             True
autorestart     true            Automatically REBOOT system after a crash        True
boottype        disk            N/A                                              False
capacity_inc    1.00            Processor capacity increment                     False
capped          true            Partition is capped                              False
conslogin       enable          System Console Login                             False
cpuguard        enable          CPU Guard                                        True
dedicated       true            Partition is dedicated                           False
ent_capacity    4.00            Entitled processor capacity                      False
frequency       93750000        System Bus Frequency                             False
fullcore        false           Enable full CORE dump                            True
fwversion       IBM,SPH01316    Firmware version and revision levels             False
iostat          false           Continuously maintain DISK I/O history           True
keylock         normal          State of system keylock at boot time             False
max_capacity    4.00            Maximum potential processor capacity             False
max_logname     20              Maximum login name length at boot time           True
maxbuf          20              Maximum number of pages in block I/O BUFFER CACHE True
maxmbuf         0               Maximum Kbytes of real memory allowed for MBUFS  True
```

```
maxpout         0               HIGH water mark for pending write I/Os per file   True
maxuproc        128             Maximum number of PROCESSES allowed per user      True
min_capacity    1.00            Minimum potential processor capacity              False
minpout         0               LOW water mark for pending write I/Os per file    True
modelname       IBM,7044-270    Machine name                                      False
ncargs          6               ARG/ENV list size in 4K byte blocks               True
pre430core      false           Use pre-430 style CORE dump                       True
pre520tune      disable         Pre-520 tuning compatibility mode                 True
realmem         3145728         Amount of usable physical memory in Kbytes        False
rtasversion     1               Open Firmware RTAS version                        False
sec_flags       0               Security Flags                                    True
sed_config      select          Stack Execution Disable (SED) Mode                True
systemid        IBM,0110B5F5F   Hardware system identifier                        False
variable_weight 0               Variable processor capacity weight                False
ngroups_allowed 128             Number of Groups Allowed at boot time             True
$
```

การเรียกค้นจำนวนกลุ่มที่อนุญาตจาก kernel: คุณต้องใช้คำสั่ง sys_param รูทีนย่อยเพื่อเรียกค้นพารามิเตอร์ v_ngroups_allowed จากฐานข้อมูล kernel นี้

```
#include<sys/types.h>
#include<sys/var.h>
#include<errno.h>
main()
{
 int rc;
 struct vario myvar;

 rc = sys_parm (SYSP_GET, SYSP_V_NGROUPS_ALLOWED, &myvar);

 if (!rc)
 printf("Number of Groups Allowed = %d\n",
 myvar.v.v_ngroups_allowed.value);
 else
 printf("sys_parm() failed rc = %d, errno = %d\n", rc, errno);
 }
```

การเปลี่ยนจำนวนกลุ่มที่อนุญาตในฐานข้อมูล ODM: คุณต้องกำหนดค่าจำนวนกลุ่มที่อนุญาตใน kernel ในระหว่างเฟสบูตระบบ ใช้คำสั่ง **chdev** เพื่อเปลี่ยนค่าในฐานข้อมูล ODM การเปลี่ยนแปลงนี้ ส่งผลกระทบต่อการรีสตาร์ทระบบ

เพื่อเปลี่ยนแปลงพารามิเตอร์ **v_ngroups_allowed** ในฐานข้อมูล ODM โดยใช้คำสั่ง **chdev** พิมพ์:

```
$ chdev -l sys0 -a ngroups_allowed=2048
sys0 changed
$
```

การควบคุมการเข้าถึงตามบทบาท

การดูแลระบบเป็นงานสำคัญของการดำเนินการประจำวัน และการรักษาความปลอดภัยเป็นส่วนที่สืบทอดของฟังก์ชันการดูแลระบบส่วนใหญ่ รวมทั้ง นอกเหนือจากการรักษาความปลอดภัยสภาวะแวดล้อมการดำเนินงานแล้ว ยังจำเป็น ต้องมอนิเตอร์กิจกรรมของระบบรายวัน

สภาวะแวดล้อมส่วนใหญ่ต้องการผู้ใช้ที่แตกต่างกันในการจัดการหน้าที่การดูแล ระบบที่ต่างกัน จำเป็นต้องคงการแยกหน้าที่เหล่านี้ เพื่อมิให้มีผู้ใช้ที่มีหน้าที่การจัดการระบบคนเดียวสามารถข้ามการรักษาความปลอดภัยระบบ ไม่ว่าจะโดยบังเอิญ หรือโดยไม่ประสงค์ดี ขณะการดูแลระบบ UNIX แบบเดิม ไม่สามารถบรรลุเป้าหมายเหล่านี้ได้ แต่ role-based access control (RBAC) สามารถทำได้

ข้อจำกัดการดูแลจัดการ UNIX ดั้งเดิม

RBAC แก้ปัญหาการดูแลจัดการระบบ UNIX ดั้งเดิมบางประเด็น ประเด็นเหล่านี้ประกอบด้วยหัวข้อต่อไปนี้:

บัญชีผู้ใช้การดูแลจัดการ root

โดยปกติแล้ว ระบบปฏิบัติการ AIX และ UNIX อื่นๆ ได้กำหนดบัญชีผู้ใช้ผู้ดูแลระบบเดียวชื่อ **root** (โดยปกติ กำหนดด้วย UID เป็น 0) ซึ่งสามารถดำเนินงานการดูแลจัดการระบบ สิทธิ์พิเศษทั้งหมดบนระบบ ความเชื่อถือต่อผู้ใช้คนเดียวสำหรับ งานการดูแลจัดการระบบทั้งหมดเป็นปัญหาในการแบ่งแยก หน้าที่ แม้การมีบัญชีผู้ใช้ดูแลจัดการบัญชีผู้ใช้เดียวจะเป็นสิ่งที่ยอมรับได้ในบางสภาวะแวดล้อม แต่ในหลายๆ สภาวะแวดล้อมจำเป็นต้องมีผู้ดูแลระบบหลายคน ซึ่งแต่ละคนมีความรับผิดชอบในงานการดูแลจัดการระบบ แตกต่างกัน

เพื่อแบ่งหน้าที่ความรับผิดชอบการดูแลจัดการ ให้แก่ผู้ใช้ของระบบหลายคน แนวปฏิบัติที่เคยทำมาคือ การแบ่งใช้รหัสผ่านของผู้ใช้ root หรือสร้างอีกผู้ใช้ ที่มี UID เดียวกันกับผู้ใช้ root วิธีนี้เป็นการแบ่งหน้าที่การดูแลจัดการ ระบบที่ก่อให้เกิดประเด็นด้านความปลอดภัย เนื่องจากผู้ดูแลระบบแต่ละราย มีการควบคุมทั้งระบบ และไม่มีวิธีจำกัดการดำเนินการ ที่ผู้ดูแลระบบจะสามารถดำเนินการได้ เนื่องจากผู้ใช้ root เป็นผู้ใช้ ที่มีสิทธิ์พิเศษสูงสุด ผู้ใช้ root สามารถทำการดำเนินการที่ไม่ได้รับอนุญาต และ ยังสามารถลบการตรวจสอบกิจกรรมใดๆ เหล่านี้ออกได้ ทำให้ไม่สามารถ ติดตามการดำเนินการดูแลจัดการเหล่านี้ได้

การเพิ่มสิทธิ์พิเศษผ่าน SUID

การควบคุม การเข้าถึงในระบบปฏิบัติการ UNIX แต่เดิมนั้นดำเนินการโดยใช้ UID ที่เชื่อมโยง กับกระบวนการเพื่อพิจารณาการเข้าถึง อย่างไรก็ตาม root UID เป็น 0 แต่เดิมได้รับอนุญาตให้ข้ามการตรวจสอบสิทธิการใช้งาน ดังนั้น กระบวนการที่กำลังทำงานในฐานะผู้ใช้ root สามารถผ่านการตรวจสอบการเข้าถึงใดๆ และทำการดำเนินการใดๆ ก็ได้ นี้เป็นประเด็นปัญหาด้านความปลอดภัยสำหรับแนวคิด UNIX ของแอ็พพลิเคชัน **setuid**

แนวคิด **setuid** อนุญาตให้คำสั่งรันภายใต้ identity ที่อื่นได้ นอกเหนือจากผู้ใช้ที่ เรียกใช้คำสั่ง นี้เป็นสิ่งจำเป็นเมื่อผู้ใช้ปกติจำเป็นต้อง ดำเนินงานสิทธิ์พิเศษให้เสร็จ ตัวอย่างในเรื่องนี้คือคำสั่ง AIX **passwd** เนื่องจากผู้ใช้ปกติไม่มีการเข้าถึงไฟล์ที่เก็บรหัสผ่านผู้ใช้ จำเป็นต้องมีสิทธิ์พิเศษเพิ่มเพื่อเปลี่ยนรหัสผ่านของผู้ใช้ ดังนั้นคำสั่ง **passwd** ใช้ **setuid** เป็น ผู้ใช้ root เมื่อผู้ใช้ปกติรันคำสั่ง **passwd** จะแสดงต่อระบบปฏิบัติการว่าผู้ใช้ root กำลังเข้าถึง ไฟล์ และการเข้าถึงได้รับอนุญาต

แม้แนวคิดนี้ ช่วยให้มีฟังก์ชันการทำงานที่พอใจ แต่ก็เกิดความเสี่ยงที่ตามมา เนื่องจากโปรแกรม **setuid** กำลังทำงานใน root context อย่างมีประสิทธิภาพ ถ้าผู้โจมตีสามารถเข้าควบคุมโปรแกรม ก่อนที่จะออกจากการทำงาน ผู้โจมตีจะมีอำนาจทั้งหมดของ root และ ยังสามารถข้ามการตรวจสอบการเข้าถึงระบบปฏิบัติการ และทำการดำเนินการ ทั้งหมดได้ วิธีแก้ปัญหาที่ดีกว่านี้คือการกำหนดเฉพาะเซ็ตย่อยของสิทธิ์พิเศษผู้ใช้ root ให้แก่โปรแกรมเท่านั้นเพื่อให้ “กฎสิทธิพิเศษน้อยที่สุด” ในหน้า 92 ได้รับการปฏิบัติตาม และช่วยลดการคุกคาม

องค์ประกอบของ RBAC

RBAC อนุญาตการสร้างบทบาทสำหรับการดูแลระบบ และ การมอบหมายงานด้านการดูแลแก่ชุดของผู้ใช้ระบบที่ไว้วางใจ ใน AIX RBAC จัดให้มี กลไกที่ โดยทั่วไปฟังก์ชันการดูแลจัดการถูกสงวนสำหรับผู้ใช้ root สามารถ ถูกกำหนดให้แก่ผู้ใช้ระบบปกติ

RBAC ทำให้สำเร็จโดยการกำหนดฟังก์ชันงาน (บทบาท) ภายในองค์กร และการกำหนดบทบาทเหล่านี้แก่ผู้ใช้ที่เจาะจง RBAC คือเฟรมเวิร์กที่จำเป็น ที่อนุญาตให้มีการดูแลจัดการระบบผ่านการใช้บทบาท บทบาทโดยทั่วไป ถูกกำหนดด้วยขอบเขตของการจัดการลักษณะการดูแลหนึ่งหรือหลายอย่าง ของสภาวะแวดล้อม การกำหนดบทบาทให้แก่ผู้ใช้อย่างมีประสิทธิภาพเป็นการมอบชุดของ สิทธิ หรือสิทธิพิเศษและอำนาจให้แก่ผู้ใช้ ตัวอย่าง บทบาทการจัดการหนึ่ง อาจใช้เพื่อจัดการระบบไฟล์ ขณะที่อีกบทบาทหนึ่งอาจสามารถ ทำการสร้างบัญชีผู้ใช้

การจัดการ RBAC มีข้อดีต่อไปนี้เมื่อเปรียบเทียบกับการจัดการ UNIX แบบดั้งเดิม:

- การดูแลระบบสามารถดำเนินงานโดยหลายผู้ใช้โดยไม่ต้องแบ่งใช้ การเข้าถึงบัญชีผู้ใช้
- การแยกการรักษาความปลอดภัยผ่านการจัดการกลุ่มย่อยเนื่องจากผู้ดูแลระบบแต่ละคน ไม่จำเป็นต้องได้รับสิทธิให้มีอำนาจมากเกินกว่าที่จำเป็น
- อนุญาตให้มีการบังคับใช้โมเดลการรักษาความปลอดภัยที่ให้สิทธิพิเศษน้อยสุด ผู้ใช้และแอ็พพลิเคชัน ได้รับสิทธิพิเศษเท่าที่จำเป็นเท่านั้นเมื่อจำเป็น เป็นการลดผลกระทบ ที่อาจเกิดจากผู้โจมตีระบบ
- อนุญาตให้มีการนำไปใช้และการบังคับใช้นโยบายการรักษาความปลอดภัยระดับบริษัทในเรื่อง การจัดการระบบและการควบคุมการเข้าถึงอย่างเท่าเทียมกัน
- ข้อกำหนดบทบาทสามารถสร้างขึ้นเพียงครั้งเดียว และจากนั้นนำไปกำหนดให้แก่ผู้ใช้ หรือเอาออก เท่าที่จำเป็นเมื่อผู้ใช้เปลี่ยนหน้าที่งาน

เฟรมเวิร์ก RBAC ถูกรวมศูนย์อยู่ในแนวคิดหลักสามข้อต่อไปนี้:

- การอนุญาต
- บทบาท
- สิทธิพิเศษ

รวมทั้ง แนวคิดเหล่านี้อนุญาตให้ระบบ RBAC สามารถบังคับใช้กฎการให้สิทธิพิเศษ น้อยสุด

การอนุญาต:

การอนุญาตคือสตริงข้อความที่สัมพันธ์กับฟังก์ชันหรือคำสั่งที่เกี่ยวกับ การรักษาความปลอดภัยหรือ การอนุญาตจัดให้มีกลไกในการให้สิทธิ แก่ผู้ใช้สำหรับการดำเนินการที่ต้องมีสิทธิพิเศษ และจัดให้มีระดับฟังก์ชันการทำงาน ที่แตกต่างกันสำหรับคลาสผู้ใช้ที่ต่างกัน

เมื่อคำสั่งที่ควบคุมโดยการอนุญาตกำลังทำงาน การเข้าถึงได้รับอนุญาตต่อเมื่อ ผู้ใช้ที่ร้องขอมีการอนุญาตที่จำเป็น การอนุญาตสามารถ ถือเป็นคีย์ที่สามารถปลดล็อกการเข้าถึงคำสั่งอย่างน้อยหนึ่ง คำสั่ง การอนุญาตไม่ถูกกำหนดให้แก่ผู้ใช้โดยตรง ผู้ใช้ได้รับการ กำหนดบทบาท ซึ่งเป็นที่รวมของการอนุญาต

บทบาท:

บทบาทอนุญาตให้ชุดของฟังก์ชันการจัดการในระบบถูกจัดกลุ่ม เข้าด้วยกัน การใช้ความคล้ายคลึงที่การอนุญาตเปรียบเป็นกุญแจ บทบาทสามารถ ใช้เป็นพวงกุญแจที่สามารถเก็บการอนุญาตหลายๆ การอนุญาต การอนุญาต อาจถูกกำหนดให้แก่บทบาทโดยตรง หรือกำหนดโดยอ้อมผ่านบทบาทย่อย บทบาทย่อยเป็นอีกบทบาทแบบง่ายที่บทบาทที่กำหนดสืบทอดการอนุญาต มา

บทบาทเองไม่ได้ให้ผู้ใช้อำนาจเพิ่มใดๆ แต่ทำหน้าที่เป็น กลไกการรวบรวมการอนุญาตแทน และเป็นส่วนอำนวยความสะดวกสำหรับ การกำหนดการอนุญาตให้แก่ผู้ใช้ การสร้างนิยามบทบาทและการกำหนดบทบาทแก่ผู้ใช้ จะพิจารณางานการดูแลจัด

การระบบที่สามารถดำเนินการโดยผู้ใช้ หลังจากบทบาทถูกกำหนด ผู้ดูแลบทบาทจะสามารถกำหนดบทบาท ให้แก่ผู้ใช้หนึ่งหรือหลายคนเพื่อจัดการการดำเนินการสิทธิ์พิเศษที่ถูกแสดง โดยบทบาท นอกจากนั้น ผู้ใช้สามารถถูกกำหนดให้มีหลายบทบาท เมื่อบทบาทหนึ่ง ถูกกำหนดให้แก่ผู้ใช้ ผู้ใช้สามารถใช้การอนุญาตที่กำหนด ให้แก่บทบาทเพื่อปลดล็อกการเข้าถึงคำสั่งการดูแลจัดการบน ระบบ

นโยบายและขั้นตอนที่เกี่ยวกับองค์กรจะพิจารณาวิธีจัดสรรบทบาท แก่ผู้ใช้ อย่ากำหนดการอนุญาตมากเกินไปให้แก่บทบาทหรือกำหนดบทบาท ให้แก่ผู้ใช้มากเกินไป บทบาทส่วนใหญ่ควรกำหนดได้เฉพาะสมาชิกของทีมงาน การดูแลจัดการเท่านั้น เนื่องจากอำนาจของ root โดยประวัติแล้วถูกกำหนดให้แก่ผู้ใช้ ที่ไว้วางใจ บทบาทควรถูกกำหนดให้แก่ผู้ใช้ที่ไว้วางใจเท่านั้น ให้บทบาทเฉพาะ ผู้ใช้ที่มีความจำเป็นที่ถูกต้องเท่านั้น และเฉพาะช่วงเวลาที่จำเป็นเท่านั้น แนวปฏิบัตินี้ ช่วยลดโอกาสที่ผู้ใช้ที่ไม่ได้รับอนุญาตจะสามารถได้รับหรือนำการอนุญาตไปใช้ในทางที่ผิด

สิทธิ์พิเศษ:

สิทธิ์พิเศษคือแอ็ตทริบิวต์กระบวนการที่อนุญาตให้กระบวนการข้าม ข้อจำกัด และการจำกัดของระบบที่เจาะจง

กลไกสิทธิ์พิเศษจัดให้แอ็พพลิเคชันที่ไว้วางใจมีความสามารถ ในการทำงานที่แอ็พพลิเคชันที่ไม่ได้รับการไว้วางใจไม่ได้รับอนุญาต ตัวอย่าง สิทธิ์พิเศษ สามารถใช้เพื่อแทนที่ข้อจำกัดการรักษาความปลอดภัยเพื่ออนุญาตการใช้งานรีซอร์ส ระบบที่เจาะจงเพิ่มมากขึ้น เช่นหน่วยความจำและพื้นที่ดิสก์ และปรับเปลี่ยน ผลการทำงานและระดับความสำคัญของกระบวนการ สิทธิ์พิเศษสามารถเปรียบได้กับ ความสามารถที่อนุญาตให้กระบวนการมีชัยเหนือข้อจำกัดการรักษาความปลอดภัยที่เจาะจงในระบบ

การอนุญาตและบทบาทเป็นเครื่องมือระดับผู้ใช้ที่ตั้งค่าความสามารถของผู้ใช้ ในการเข้าถึงการดำเนินการสิทธิ์พิเศษ หรืออีกนัยหนึ่ง สิทธิ์พิเศษคือกลไก การจำกัดที่ใช้ในเคอร์เนลเพื่อพิจารณาว่ากระบวนการได้รับอนุญาตให้ดำเนิน การเป็นพิเศษหรือไม่

สิทธิ์พิเศษเชื่อมโยงกับกระบวนการและโดยปกติถูกจัดเตรียมผ่าน การร้องขอของคำสั่งสิทธิ์พิเศษ เนื่องจากสิทธิ์พิเศษที่เชื่อมโยงเหล่านี้ กระบวนการมีคุณสมบัติในการดำเนินการสิทธิ์พิเศษที่เกี่ยวข้องได้ ตัวอย่าง ถ้าผู้ใช้ใช้บทบาทที่มีการอนุญาตเพื่อรันคำสั่ง ชุดของสิทธิ์พิเศษ ถูกกำหนดให้แก่กระบวนการเมื่อคำสั่งถูกรัน

กฎสิทธิพิเศษน้อยที่สุด:

ในระบบปฏิบัติการ การดำเนินการบางอย่างต้องมีสิทธิพิเศษ และสิทธิ ในการดำเนินการเหล่านี้ถูกจำกัดเฉพาะผู้ใช้ที่ได้รับอนุญาต การดำเนินการสิทธิพิเศษ เหล่านี้โดยส่วนใหญ่จะมีงาน เช่น การรีบูตระบบ การเพิ่มและ การแก้ไขระบบไฟล์ การเพิ่มและการลบผู้ใช้ และการแก้ไขวันที่และเวลา ของระบบ

ในระบบ UNIX ดั้งเดิม กระบวนการ หรือผู้ใช้สามารถอยู่ในโหมดปกติ หรือโหมดสิทธิพิเศษ (หรือเรียกว่า superuser หรือ root) กระบวนการที่กำลังทำงานเป็น root สามารถเรียกใช้งานคำสั่งใดๆ และมี การดำเนินการระบบ ขณะที่ผู้ใช้ปกติไม่สามารถดำเนินงานที่ต้องใช้สิทธิพิเศษ ระบบ UNIX ดั้งเดิมมี แนวคิดทั้งหมดหรือไม่มีอะไรเลยแบบหยาบของสิทธิพิเศษและเผชิญหน้าการคุกคามด้านความปลอดภัย ของผู้ดูแลระบบที่มีสิทธิพิเศษมากเกินไป

แนวการดำเนินการ UNIX ดั้งเดิม ที่มีโมหสิทธิพิเศษเดียวที่ให้สิทธิการเข้าถึงทั้งหมดแก่ระบบเป็นสิ่งที่หยาบเกินไป ที่จะเป็นไปตามข้อกำหนดของระบบที่ต้องการการรักษาความปลอดภัยอย่างสูง ระบบที่ออกแบบ ให้มีความปลอดภัยจำเป็นที่แต่ละกระบวนการต้องได้รับอนุญาตให้มีชุดของสิสิทธิพิเศษ ที่มีการจำกัดให้มากที่สุดเฉพาะที่จำเป็นต้องใช้ในการทำงาน สิทธิพิเศษช่วยให้เกิดประโยชน์ ที่มีเพียงกระบวนการที่จำเป็นต้องได้รับสิทธิพิเศษเท่านั้นที่จะได้รับอนุญาตให้มีสิทธิพิเศษเหล่านี้

ข้อจำกัดของสิทธิพิเศษนี้เป็นที่รู้จักในชื่อกฎของการให้สิทธิพิเศษน้อยที่สุด และเป็นประโยชน์ในการช่วยจำกัดความเสียหายที่เกิดกับระบบเนื่องจากการผู้ดูแลระบบและผู้ควบคุมเครื่อง ที่ไม่ระมัดระวัง หรือไม่ประสงค์ดี

ตัวอย่าง การเปลี่ยนรหัสผ่านจำเป็นต้องมีสิทธิพิเศษเฉพาะเพื่อเข้าถึงไฟล์ ที่โดยปกติไม่สามารถเข้าถึงได้โดยผู้ใช้ปกติ ถ้าผู้ใช้มีสิทธิพิเศษเหล่านี้เสมอ ผู้ใช้จะสามารถดำเนินการอื่นๆ ที่ไม่เป็นที่ชื่นชอบ จากจุดยืนด้านความปลอดภัย ดังนั้น ควรอนุญาตให้มีสิทธิพิเศษที่จำเป็น เท่านั้นสำหรับคำสั่ง **passwd** และไม่ต้องให้แก่ผู้ใช้ทั้งหมด

ในสภาวะแวดล้อม RBAC ผู้ใช้เองไม่มีสิทธิพิเศษที่สืบทอดใดๆ ผู้ใช้ได้รับอนุญาตเพียงให้รันคำสั่งเฉพาะซึ่งจะให้สิทธิพิเศษ ถ้าผู้ใช้มีสิทธิพิเศษที่ที่อนุญาตโดยตรงแทน ผู้ใช้จะสามารถใช้สิทธิพิเศษ ได้ตลอดเวลา และในทุกๆ แนวทางที่ต้องการ การจำกัดสิทธิพิเศษแก่แต่ละคำสั่ง อนุญาตให้คอนเท็กซ์ที่สิทธิพิเศษถูกนำใช้ได้รับการจำกัด สิ่งนี้นำไปสู่การมีความปลอดภัยที่เพิ่มประสิทธิภาพมากขึ้นเนื่องจากถ้ามีแอ็พพลิเคชันที่ไว้วางใจใดๆ ถูกบุกรุก โดยผู้โจมตี ผู้โจมตีจะมีชุดของสิทธิพิเศษที่จำกัดแทนที่ จะได้รับอำนาจของ root ที่มีสิทธิพิเศษทั้งหมดแทน

แอ็พพลิเคชันที่ไว้ใจได้ต้องได้รับการตรวจสอบอย่างระมัดระวังก่อนที่ จะได้รับสิทธิ์พิเศษ นอกจากนี้การกำหนดสิทธิ์พิเศษสำหรับแอ็พพลิเคชัน ควรทำเมื่อ จำเป็น แอ็พพลิเคชันที่ไว้วางใจจะเหมือนโปรแกรมอื่นๆ แตกต่างกันเฉพาะที่แอ็พพลิเคชันที่ไว้วางใจได้รับอนุญาต ให้ดำเนินการที่จะถูกปฏิเสธสำหรับแอ็พพลิเคชันที่ไม่ไว้วางใจ

AIX RBAC

AIX จัดเตรียมการใช้ RBAC ที่จำกัดก่อนหน้า AIX 6.1

เริ่มตั้งแต่ AIX 6.1 การนำใช้ RBAC ใหม่จัดให้มีกลไกการทำกลุ่มย่อยขนาดเล็ก สำหรับงานการดูแลจัดการระบบเซ็กเมนต์ เนื่องจากการนำใช้ RBAC สองวิธีนี้ แตกต่างกันอย่างมากในด้านฟังก์ชันการทำงาน เทอมต่อไปนี้ถูกนำไปใช้:

โหมด RBAC แบบเก่า
: ลักษณะการทำงานเชิงประวัติของกฎ AIX ที่ใช้กับเวอร์ชัน ก่อน AIX 6.1

โหมด RBAC แบบปรับปรุง
: การนำใช้ใหม่เริ่มมีใน AIX 6.1

โดยได้รับการสนับสนุนการดำเนินการทั้งสองโหมด อย่างไรก็ตาม โหมด RBAC แบบปรับปรุง คือค่าดีฟอลต์บนระบบ AIX 6.1 ที่ติดตั้งใหม่ ส่วน ต่อไปนี้มีการอธิบายย่อๆ ของทั้งสองโหมด และความแตกต่างกัน และข้อมูลเกี่ยวกับการตั้งค่าระบบเพื่อดำเนินงานในโหมด RBAC ที่ต้องการ

โหมด RBAC แบบเก่า:

ก่อนหน้า AIX 6.1, AIX ได้จัดเตรียมการทำงาน RBAC ที่จำกัดซึ่งอนุญาตให้ผู้ใช้ที่ไม่ใช่ผู้ใช้ root ดำเนินการกับงาน การดูแลระบบ

ในการนำใช้ RBAC นี้ เมื่อคำสั่งการดูแลที่กำหนดถูกร้องขอ โดยผู้ใช้ที่มิใช่ root โค้ดในคำสั่งจะพิจารณาว่า ผู้ใช้ถูกกำหนดบทบาทที่มีการอนุญาตที่จำเป็นหรือไม่ ถ้าพบที่ตรงกัน การเรียกทำงานคำสั่งจะดำเนินต่อไป ถ้าไม่พบ คำสั่งจะล้มเหลวพร้อมมีข้อผิดพลาด โดยส่วนใหญ่มักจำเป็นที่คำสั่งถูกควบคุมโดยการอนุญาต เป็น **setuid** แก่ผู้ใช้ root สำหรับผู้ร้องขอที่ได้รับอนุญาตเพื่อ ให้มีสิทธิพิเศษที่จำเป็นในการการดำเนินการได้สำเร็จ

การนำใช้ RBAC นี้ยังเพิ่มชุดการอนุญาตที่กำหนดไว้แล้วแต่ผู้ใช้ สามารถขยายเพิ่มได้ที่สามารถนำไปใช้พิจารณาการเข้าถึงคำสั่ง การดูแลจัดการ นอกจากนั้น ยังมีเฟรมเวิร์กของคำสั่งด้านการดูแลจัดการ และอินเตอร์เฟสเพื่อสร้างบทบาท กำหนดการอนุญาตแก่บทบาท และกำหนด บทบาทแก่ผู้ใช้

ขณะที่การนำไปใช้จัดให้มีความสามารถในการความรับผิดชอบการดูแลจัดการระบบ เซ็กเมนต์แบบแบ่งส่วน ซึ่งทำหน้าที่โดยมีข้อจำกัดต่อไปนี้:

1. เฟรมเวิร์กจำเป็นต้องเปลี่ยนแปลงคำสั่งและแอ็พพลิเคชันเพื่อให้เปิดใช้งาน RBAC
2. การอนุญาตที่กำหนดไว้แล้วไม่เป็นแบบกลุ่มย่อย และกลไกที่จะสร้าง การอนุญาตไม่เสถียร
3. จำเป็นต้องมีความเป็นสมาชิกในกลุ่มที่แน่นอน รวมทั้งมีบทบาท ที่มีการอนุญาตที่กำหนดเพื่อใช้รันคำสั่ง
4. การแบ่งแยกหน้าที่เป็นเรื่องยากต่อการนำไปปฏิบัติใช้ ถ้าผู้ใช้ถูกกำหนด ให้มีหลายบทบาท ไม่มีวิธีที่จะทำหน้าที่ภายใต้บทบาทเดียว ผู้ใช้มัก มีการอนุญาตทั้งหมดสำหรับบทบาททั้งหมดของตนเสมอ
5. กฎสิทธิพิเศษน้อยที่สุดไม่ถูกนำมาปรับใช้ในระบบปฏิบัติการ โดยทั่วไปคำสั่งต้องเป็น SUID แก่ผู้ใช้ root

โหมด RBAC แบบเก่าได้รับการสนับสนุนเพื่อความเข้ากันได้ แต่โหมด RBAC แบบปรับปรุง เป็นโหมด RBAC ดีฟอลต์ โหมด RBAC แบบปรับปรุงเป็นที่ต้องการใช้บน AIX

โหมด RBAC แบบปรับปรุง:

การนำ RBAC ไปใช้ที่มีประสิทธิผลมากขึ้นมีมาให้พร้อมกับ AIX 6.1 แอ็พพลิเคชันที่จำเป็นต้องใช้สิทธิพิเศษการดูแลจัดการสำหรับการดำเนินการที่เฉพาะบางอย่าง มีอ็อพชันการรวมใหม่ที่มีโครงสร้างพื้นฐาน AIX RBAC แบบปรับปรุง

อ็อพชันการรวมเข้าเหล่านี้รวมศูนย์อยู่ที่การใช้สิทธิพิเศษแบบกลุ่มย่อยและ การอนุญาต และความสามารถในการตั้งค่าคำสั่งใดๆ บนระบบเป็น คำสั่งสิทธิพิเศษ คุณลักษณะของโหมด RBAC แบบปรับปรุงจะถูกติดตั้ง และเปิดใช้งานเป็นค่าดีฟอลต์ในการติดตั้ง AIX ทั้งหมด ตั้งแต่ AIX 6.1

โหมด RBAC แบบปรับปรุงจัดให้มีชุดที่ตั้งค่าได้ของการอนุญาต, บทบาท, คำสั่งสิทธิพิเศษ, อุปกรณ์และไฟล์ผ่านฐานข้อมูล RBAC ต่อไปนี้ ที่แสดงรายการด้านล่าง ด้วยการใช้ RBAC แบบปรับปรุง ฐานข้อมูลสามารถอยู่ใน ระบบไฟล์โลคัล หรือได้รับการจัดการแบบรีโมตผ่าน LDAP

- ฐานข้อมูลการอนุญาต
- ฐานข้อมูลบทบาท
- ฐานข้อมูลคำสั่งสิทธิพิเศษ
- ฐานข้อมูลอุปกรณ์สิทธิพิเศษ
- ฐานข้อมูลไฟล์สิทธิพิเศษ

โหมด RBAC แบบปรับปรุงมีข้อกำหนดการตั้งชื่อใหม่สำหรับการอนุญาต ที่อนุญาตให้สร้างสร้างของการอนุญาต AIX จัดให้มีชุดกลุ่มย่อยของการอนุญาตที่ระบบกำหนดและผู้ดูแลระบบมีอิสระ ในการสร้างการอนุญาตที่ผู้ใช้กำหนดเองได้ตามความจำเป็น

ลักษณะการทำงานของบทบาทได้รับการปรับปรุงให้มีการแบ่งแยกหน้าที่การทำงาน RBAC แบบปรับปรุงเพิ่มแนวคิดของเซสชันบทบาท เซสชันบทบาทคือ กระบวนการที่มีบทบาทที่เชื่อมโยงอยู่อย่างน้อยหนึ่งบทบาท ผู้ใช้สามารถสร้างเซสชันบทบาทสำหรับบทบาทใดๆ ที่ได้ถูกกำหนดไว้โดยการเรียกทำงานบทบาทเดียว หรือหลายบทบาทที่เลือกในครั้งหนึ่ง โดยดีฟอลต์ กระบวนการของระบบใหม่ ไม่มีบทบาทใดที่เชื่อมโยง บทบาทได้ถูกปรับปรุงเพิ่มเติมเพื่อสนับสนุน ข้อกำหนดที่ผู้ใช้ต้องพิสูจน์ตัวตนก่อนสามารถเรียกทำงานบทบาท เพื่อป้องกันผู้โจมตีเข้าใช้งานเซสชันผู้ใช้ เนื่องจากผู้โจมตี จำเป็นต้องพิสูจน์ตัวตนเพื่อเรียกทำงานบทบาทของผู้ใช้

การเพิ่มของฐานข้อมูลคำสั่งสิทธิพิเศษนำใช้กฎสิทธิพิเศษ ที่น้อยที่สุด กลุ่มย่อยของสิทธิพิเศษระบบได้ถูกเพิ่ม และสามารถให้สิทธิพิเศษที่ระบุชัดเจนแก่คำสั่ง และการทำงานของ คำสั่งสามารถถูกควบคุมโดยการอนุญาต นี้ให้ฟังก์ชันการทำงาน เพื่อบังคับใช้การตรวจสอบการอนุญาตสำหรับการเรียกทำงานคำสั่งโดยไม่จำเป็น ต้องเปลี่ยนแปลงโค้ดในคำสั่ง การใช้ฐานข้อมูลคำสั่งสิทธิพิเศษ กำจัดความต้องการใช้แอ็พพลิเคชัน SUID และ SGID เนื่องจากมีความสามารถ ในการกำหนดสิทธิพิเศษที่จำเป็นเท่านั้น

ฐานข้อมูลอุปกรณ์สิทธิพิเศษอนุญาตให้การเข้าถึงอุปกรณ์ถูกควบคุม โดยสิทธิพิเศษ ขณะที่ฐานข้อมูลไฟล์สิทธิพิเศษอนุญาตให้ผู้ใช้ที่ไม่มีสิทธิพิเศษ เข้าถึงไฟล์ที่จำกัดสิทธิได้โดยขึ้นอยู่กับการอนุญาต ฐานข้อมูลเหล่านี้เพิ่ม กลุ่มย่อยของงานการดูแลจัดการระบบที่สามารถกำหนดให้แก่ผู้ใช้ อื่นที่ไม่มีสิทธิพิเศษ

ข้อมูลในฐานข้อมูล RBAC ถูกรวมและยืนยัน จากนั้นถูกส่ง ไปที่พื้นที่ของเคอร์เนลที่กำหนดให้เป็น Kernel Security Tables (KST) สิ่งสำคัญที่ต้องทราบว่าสถานะของข้อมูลใน KST ใช้พิจารณา นโยบายการรักษาความปลอดภัยสำหรับระบบ รายการถูกแก้ไขในฐานข้อมูล RBAC ระดับผู้ใช้ไม่ถูกใช้ในการตัดสินใจด้านความปลอดภัยจนกว่าข้อมูลนี้ ถูกส่งไปที่ KST ด้วยคำสั่ง **setkst**

การตั้งค่าโหมด RBAC:

โหมด RBAC ถูกควบคุมโดยตัวแปรการตั้งค่าระบบ ในเคอร์เนล ตัวแปรนี้ระบุว่า Enhanced RBAC Mode ถูกเปิดใช้งานหรือปิดใช้งาน

โหมด RBAC ที่ปรับปรุงจะถูกเปิดใช้งานโดยดีฟอลต์บน AIX 6.1 หรือใหม่กว่า คุณสามารถรันคำสั่ง **chdev** บนอุปกรณ์ **sys0** และระบุค่า **false** สำหรับแอ็ตทริบิวต์ **enhanced_RBAC** เพื่อปิดใช้งานโหมด RBAC แบบปรับปรุง และกลับไปใช้งานโหมด RBAC แบบเก่า คุณต้องรีบูต ระบบเพื่อให้การเปลี่ยนค่าแอ็ตทริบิวต์ **enhanced_RBAC** มีผล ในการเปิดใช้งานโหมด RBAC แบบปรับปรุง แอ็ตทริบิวต์ **enhanced_RBAC** ควรตั้งค่า เป็น **true** โดยหลักการแล้ว โหมดยังสามารถถูกตั้งค่า หรือเคียวรี ผ่านการเรียกใช้ระบบ **sys_parm( )**

รันคำสั่งต่อไปนี้บนระบบเพื่อเรียกข้อมูลโหมด RBAC ปัจจุบัน:

```
lsattr -E -l sys0 -a enhanced_RBAC
```

คุณสามารถปิดใช้งานโหมด RBAC แบบปรับปรุงได้โดยการรันคำสั่งต่อไปนี้ จากนั้นรีบูตระบบ:

```
chdev -l sys0 -a enhanced_RBAC=false
```

ในสภาวะแวดล้อม WPAR โหมด RBAC สามารถตั้งค่าได้จากระบบโกลบอลเท่านั้น และจะมีผลต่อโกลบอลอย่างเท่าเทียมกัน รวมถึงทั้งหมดของ WPARs บน ระบบ

การเปรียบเทียบโหมด RBAC แบบเก่าและโหมด RBAC แบบปรับปรุง:

อินเตอร์เฟสที่มีอยู่และอินเตอร์เฟสใหม่ได้ถูกแก้ไขเพื่อตรวจสอบการตั้งค่า ระบบ และรันโค้ดใหม่ หรือทำตามการทำงานแบบเก่า

ในโหมด RBAC แบบเก่า การอนุญาตที่ถูกตรวจสอบภายในโค้ดของ คำสั่งเท่านั้นที่จะถูกบังคับใช้ Kernel Security Tables (KST) ไม่มี ผลใดๆ ต่อการทำงานคำสั่ง หรือการตรวจสอบการอนุญาต การพิจารณา ว่าผู้ใช้มีการอนุญาตตามลักษณะการทำงานโหมด RBAC แบบเก่า ของการเรียกข้อมูลการอนุญาตของผู้ใช้ทั้งหมด และการตรวจสอบเพื่อหารายการที่ตรงกัน คุณลักษณะ ใหม่ เช่นคำสั่ง **swrole** และแอ็ตทริบิวต์ **default_roles** และ **auth_mode** ไม่มีในโหมด RBAC แบบเก่า อย่างไรก็ตาม สิทธิพิเศษ การอนุญาต และคำสั่งการจัดการสำหรับการอนุญาตใหม่ได้รับการสนับสนุนในโหมด RBAC แบบเก่า

ตารางต่อไปนี้แสดงรายการความแตกต่างบางอย่างระหว่าง โหมด RBAC แบบเก่าและแบบปรับปรุง

ตารางที่ 9. ความแตกต่างระหว่างโหมด RBAC แบบเก่าและแบบปรับปรุง

คุณลักษณะ	**RBAC แบบเก่า**	**RBAC แบบปรับปรุง**
การเรียกทำงานบทบาท	บทบาทของผู้ใช้ทั้งหมดแอ็คทีฟเสมอ	โดยดีฟอลต์ บทบาทไม่แอ็คทีฟจนกว่าจะถูกระบุอย่างชัดเจน ผ่านคำสั่ง **swrole**
แอ็ตทริบิวต์ **default_roles**	ไม่มี	สนับสนุน
คำสั่ง **swrole**	ไม่มี	สนับสนุน
คำสั่งการจัดการบทบาท	สนับสนุน	สนับสนุน
คำสั่งการจัดการการอนุญาต	สนับสนุน	สนับสนุน
ลำดับชั้นการอนุญาต	แต่ละการอนุญาตเป็นอิสระต่อกัน ไม่มีฟังก์ชันการทำงานลำดับชั้น	สนับสนุนแนวคิดของลำดับชั้นการอนุญาตที่การอนุญาต สามารถเป็นพาเรนต์ของการอนุญาตอื่นๆ
การตรวจสอบการอนุญาต	ถูกบังคับใช้ต่อเมื่อตรวจสอบคำสั่งสำหรับการอนุญาต	ถูกบังคับใช้ผ่าน Privileged Command Database และ/หรือ โดยใช้คำสั่งเอง
สิทธิพิเศษกลุ่มย่อย	สนับสนุน	สนับสนุน
คำสั่ง **pvi**	ไม่มี	สนับสนุน
Kernel Security Tables	ไม่มี	สนับสนุน
ตำแหน่งฐานข้อมูล RBAC	โลคัลไฟล์	โลคัลไฟล์หรือ LDAP

การใช้ RBAC แบบปรับปรุง

ผู้ดูแลระบบควรมีความรู้ในหัวข้อต่อไปนี้ เพื่อใช้RBAC แบบปรับปรุงอย่างมีประสิทธิภาพ

การอนุญาต RBAC:

การอนุญาตเป็นส่วนสำคัญของ Role Based Access Control (RBAC) ระบบปฏิบัติการใช้สตริงการอนุญาตเพื่อพิจารณาการมีสิทธิ์ ก่อนที่จะทำการดำเนินการสิทธิ์พิเศษ การตรวจสอบที่เกี่ยวข้องสามารถถูกดำเนินการ จากภายในโค้ด หรือสามารถทำผ่านโหลดเดอร์เมื่อรัน ไฟล์เรียกทำงานสิทธิ์พิเศษที่ได้รับการป้องกัน

การตั้งค่าสตริงการอนุญาตบ่งชี้ว่าเป็นการดำเนินการสิทธิ์พิเศษ ที่สตริงแสดงและควบคุม ข้อกำหนดการตั้งชื่อ AIX สำหรับการอนุญาตสนับสนุนโครงสร้างแบบลำดับชั้นที่ถูกแสดงความหมาย โดยชื่อที่เป็นข้อความของการอนุญาต สตริงการอนุญาต AIX ใช้รูปแบบการแสดงแบบจุดเพื่ออธิบายลำดับชั้นการอนุญาต ตัวอย่าง การอนุญาตที่สร้างระบบไฟล์ใหม่คือ **aix.fs.manage.create** ถ้าการอนุญาตนี้ถูกรวมในบทบาท ผู้ใช้ที่ได้รับการกำหนดบทบาทนี้ จะสามารถสร้างระบบไฟล์ AIX ถ้าการอนุญาตพาเรนต์ **aix.fs.manage** ถูกรวมในบทบาท ดังนั้น ผู้ใช้ที่ได้รับการกำหนดบทบาทนี้จะสามารถดำเนินการงานการจัดการ ระบบไฟล์อื่นๆ ได้เช่นเดียวกับการสร้างระบบไฟล์

AIX RBAC แยกความแตกต่างระหว่าง การอนุญาตที่ระบบจัดให้มี (การอนุญาตที่ระบบกำหนด) และการอนุญาต ที่สร้างขึ้นหลังการการติดตั้ง (การอนุญาตที่ผู้ใช้กำหนดเอง)

การอนุญาตที่ระบบกำหนด:

AIX จัดให้มีชุดของการอนุญาต ที่กำหนดไว้แล้วและไม่สามารถแก้ไขได้ เหล่านี้คือการอนุญาตที่ ระบบกำหนด การอนุญาต เหล่านี้เกี่ยวข้องกับการดำเนินการ AIX สิทธิ์พิเศษที่แตกต่างกัน ความเกี่ยวข้องถูกระบุใน Privileged Command Database

ที่ชั้นบนสุดของลำดับชั้นการอนุญาตที่ระบบกำหนดคือการอนุญาต **aix** การอนุญาตนี้เป็นพาเรนต์ของการอนุญาตที่ระบบกำหนดอื่นๆ ทั้งหมด การให้การอนุญาตนี้แก่บทบาทเป็นการให้การอนุญาตที่ระบบกำหนดทั้งหมด แก่บทบาท ในการแสดงชุดที่สมบูรณ์ของการอนุญาตที่ระบบกำหนด AIX และคำอธิบายอย่างย่อของแต่ละการอนุญาต ให้รันคำสั่ง ต่อไปนี้:

```
lsauth -f -a description ALL_SYS
```

เอาต์พุตของคำสั่งด้านบนแสดงให้เห็นว่ารายการของการอนุญาตที่ระบบกำหนด เป็นลำดับชั้นแบบหลายระดับ ตัวอย่าง การอนุญาต **aix** มี ชายด์ที่ติดกันหลายชายด์ แต่ละชายด์เหล่านั้นจะเป็นพาเรนต์ของอีก ลำดับชั้น การอนุญาต **aix.fs** มีหลายการอนุญาตชายด์ รวมถึง **aix.fs.manage** ซึ่งในทางกลับกันก็มีหลายการอนุญาต เช่น **aix.fs.manage.change** และ **aix.fs.manage.create**

การอนุญาตที่ผู้ใช้กำหนดเอง:

นอกเหนือจากการอนุญาตที่ระบบกำหนดแล้ว AIX RBAC ยังอนุญาตให้ผู้ดูแลระบบ กำหนดการอนุญาตแบบกำหนดเองของตนในฐานข้อมูลการอนุญาต (/etc/security/authorizations) เหล่านี้คือการอนุญาตที่ผู้ใช้กำหนดเอง

ผู้ดูแลระบบสามารถเพิ่ม แก้ไข หรือลบการอนุญาต ที่ผู้ใช้กำหนดเอง ตัวอย่าง ผู้ดูแลระบบสามารถอนุญาตให้ผู้ใช้ บางคนรันคำสั่งสิทธิ์พิเศษโดยการสร้างการอนุญาตที่ผู้ใช้กำหนดเอง จากนั้นเชื่อมโยงการอนุญาตนี้กับคำสั่งและให้สิทธิ์ การอนุญาตแก่บทบาทที่ถูกกำหนดไปยังผู้ใช้เหล่านี้

การอนุญาตที่ผู้ใช้กำหนดเองสนับสนุนแนวคิดลำดับชั้น เหมือนกับการอนุญาตที่ระบบกำหนด อย่างไรก็ตาม มีข้อจำกัด ในการตั้งชื่อของการอนุญาตที่ผู้ใช้กำหนดเอง AIX

- การอนุญาตที่ผู้ใช้กำหนดเองต้องถูกกำหนดภายใต้พาเรนต์ระดับบนสุด ใหม่ หรืออีกนัยหนึ่ง การอนุญาตที่ผู้ใช้กำหนดเองไม่สามารถเป็นชายด์ ของการอนุญาตที่ระบบกำหนด (**aix**)
- ชื่อการอนุญาตสามารถมีอักขระที่พิมพ์ได้สูงสุด 63 อักขระ
- ลำดับชั้นพาเรนต์ของการอนุญาตสามารถมีได้สูงสุดแปด ระดับ
- การอนุญาตสามารถมีจำนวนชายด์ที่ติดกันเท่าใดก็ได้ แต่ สามารถมีพาเรนต์ติดกันได้หนึ่งพาเรนต์เท่านั้น สองการอนุญาตที่เป็นอิสระต่อกัน ไม่สามารถมีชายด์ติดกันที่เป็นค่าเดียวกันได้

เนื่องจากลำดับชั้นไม่อนุญาตให้องค์ประกอบมีพาเรนต์โดยตรง หลายพาเรนต์ คุณไม่สามารถสร้างการอนุญาตที่ผู้ใช้กำหนดเองที่ เป็นพาเรนต์ของการอนุญาตที่ระบบกำหนดที่มีอยู่แล้ว ดังนั้น ความพยายามที่จะสร้างการอนุญาตชื่อ **aix.custom** จะล้มเหลวและการสร้างการอนุญาตชื่อ **custom.aix** จะ ส่งผลให้มีการอนุญาตใหม่ และไม่ได้ทำหน้าที่เป็นพาเรนต์ ของการอนุญาตที่ระบบกำหนด **aix**

ไวยากรณ์ต่อไปนี้ได้รับการแนะนำเมื่อสร้างการอนุญาตที่ผู้ใช้กำหนดเอง เพื่อหลีกเลี่ยงความขัดแย้งระหว่างชื่อการอนุญาตในหลายๆ ซอฟต์แวร์ คอมโพเนนต์:

vendor_name.product_name.function.function1.function2...

vendor_name

ระบุชื่อของผู้จำหน่ายซอฟต์แวร์โมดูล

product_name

ชื่อผลิตภัณฑ์ระดับสูงของผลิตภัณฑ์ที่ถูกจัดการด้วย RBAC

function, *function1*, *function2* ...

สตริงเหล่านี้แทนฟังก์ชันที่ถูกจัดการด้วย RBAC สตริงเหล่านี้ยังจัดให้มีการแสดงแบบลำดับชั้น ที่ฟังก์ชันเหล่านี้ได้รับการจัดการ

ตัวอย่าง **ibm.db2.manage** สามารถแทน รูปแบบการจัดการของชุดฐานข้อมูล IBM DB2 ดังที่กล่าวถึงก่อนหน้านี้ สตริง *vendor_name* **aix** ถูกสงวนไว้สำหรับ AIX ใช้และไม่อนุญาตให้ใช้สำหรับ การอนุญาตที่ผู้ใช้กำหนดเอง

มีคำสั่งการจัดการการอนุญาตหลายคำสั่งที่ผู้ดูแลระบบ สามารถใช้เพื่อแสดงรายการ สร้าง แก้ไข และลบการอนุญาต ที่ผู้ใช้กำหนดเอง การอนุญาตที่ผู้ใช้กำหนดเองสามารถสร้างด้วยคำสั่ง **mkauth**, แก้ไข้ด้วยคำสั่ง **chauth**, ลบด้วยคำสั่ง **rmauth** และแสดงด้วยคำสั่ง **lsauth** ในการแสดงการอนุญาตที่ผู้ใช้และระบบกำหนดทั้งหมด พร้อมคำอธิบายอย่างย่อ ของแต่ละรายการ ให้รันคำสั่งต่อไปนี้:

```
lsauth -f -a description ALL_USR
```

ก่อนสร้างการอนุญาตที่ผู้ใช้กำหนดเอง ให้พิจารณาปัญหา ต่อไปนี้:

- เป็นการเหมาะสมที่จะใช้การอนุญาตที่ระบบกำหนดที่มีอยู่แล้ว แทนการสร้างการอนุญาตที่ผู้ใช้กำหนดเองขึ้นใหม่หรือไม่?
- การอนุญาตใหม่อยู่ภายใต้ลำดับชั้นการอนุญาตที่ผู้ใช้กำหนดเอง ที่มีอยู่แล้ว หรือเป็นการอนุญาตระดับแรกของลำดับชั้นใหม่?
- ถ้ามีลำดับชั้นใหม่ โครงสร้างเป็นอย่างไร?
- คำอธิบายเป็นข้อความของการอนุญาตเป็นอย่างไร?
- จำเป็นต้องมีคำแปลภาษาของคำอธิบายการอนุญาตหรือไม่?
- มีเหตุผลใดที่จะระบุ ID การอนุญาตที่เจาะจงเมื่อ สร้างการอนุญาตหรือไม่? แนะนำให้ใช้คำสั่ง **mkauth** เพื่อสร้าง ID การอนุญาต

หลังจากพิจารณาปัญหาเหล่านี้แล้ว ให้ดำเนินขั้นตอนต่อไปนี้เพื่อ สร้างการอนุญาต:

1. ถ้าจำเป็นต้องมีการแปลภาษา ให้สร้างหรือเพิ่มคำอธิบายเป็น แค็ตตาล็อกข้อความ
2. ใช้คำสั่ง **mkauth** เพื่อสร้างการอนุญาต พาเรนต์ทั้งหมดในลำดับชั้นถ้ายังไม่มี
3. ใช้คำสั่ง **mkauth** เพื่อสร้างการอนุญาต ที่ต้องการ ระบุแอ็ตทริบิวต์ **id** ด้วยคำสั่งถ้า จำเป็นต้องใช้ค่าที่ระบุ

การโอนย้ายการอนุญาตแบบเก่า:

ก่อนหน้า AIX เวอร์ชัน 6.1 ระบบปฏิบัติการมีชุดของการอนุญาตที่กำหนดไว้แล้วจำนวนจำกัดที่ ระบบปฏิบัติการจะรู้จัก การอนุญาตเหล่านี้ ไม่ถูกกำหนดในไฟล์ใดๆ บนระบบ แต่สามารถกำหนดให้แก่บทบาท ได้ในทันที เมื่อต้องการสนับสนุนการให้สิทธิ์แบบดั้งเดิมเหล่านี้ภายใน AIX เวอร์ชัน 6.1 ใหม่และ กรอบงาน RBAC ที่ใหม่กว่า การกำหนดสิทธิ์ดั้งเดิมเหล่านี้จะถูกกำหนดเป็นสิทธิ์ที่ผู้ใช้กำหนด และถูกจัดเตรียมในฐานข้อมูลการกำหนดสิทธิ์โดยดีฟอลต์

ตั้งแต่ระบบปฏิบัติ AIX กำลังย้ายไปเป็นข้อกำหนดการตั้งชื่อสิทธิ์, การตรวจสอบใดๆ สำหรับสิทธิ์เก่ามีชื่ออยู่ในระบบปฏิบัติการ AIX ได้ถูกปรับเปลี่ยนเพื่อตรวจสอบสิทธิ์ใหม่ที่สอดคล้องกัน และอนุญาตให้เข้าถึงหากสิทธิ์ใดๆ มีอยู่สำหรับกระบวนการ ตาราง ต่อไปนี้แสดงการอนุญาตที่กำหนดไว้แล้วแบบเก่าและ การอนุญาตที่ระบบกำหนดไว้แล้วใหม่ที่สอดคล้องกัน

การอนุญาต AIX ที่มี	**การอนุญาตใหม่ที่สอดคล้อง**
Backup	aix.fs.manage.backup
Diagnostics	aix.system.config.diag
DiskQuotaAdmin	aix.fs.manage.quota
GroupAdmin	aix.security.group
ListAuditClasses	aix.security.audit.list
PasswdAdmin	aix.security.passwd
PasswdManage	aix.security.passwd.normal
UserAdmin	aix.security.user
UserAudit	aix.security.user.change
RoleAdmin	aix.security.role
Restore	aix.fs.manage.restore

บทบาท RBAC:

บทบาทคือกลไกที่ใช้กำหนดการอนุญาตให้แก่ผู้ใช้ และจัดกลุ่มชุดของงานการดูแลจัดการระบบไว้ด้วยกัน บทบาท AIX คือคอนเทนเนอร์หลักสำหรับการรวบรวมการอนุญาต

AIX สนับสนุนการมอบหมาย การอนุญาตโดยตรงให้แก่บทบาท หรือการมอบหมายการอนุญาตโดยอ้อมผ่านทาง บทบาทย่อย บทบาทย่อยสามารถถูกระบุให้แก่บทบาทในแอ็ตทริบิวต์ **rolelist** ของบทบาท การตั้งค่าบทบาทเพื่อให้มีบทบาทย่อยที่กำหนดอย่างมีประสิทธิผลจะกำหนด การอนุญาตทั้งหมดในบทบาทย่อยแก่บทบาท

การกำหนดบทบาทให้ก่ผู้ใช้เพื่ออนุญาตให้ผู้ใช้เข้าถึงบทบาทและใช้ การอนุญาตที่มีอยู่ในบทบาท ผู้ดูแลระบบสามารถกำหนดบทบาทให้หลายผู้ใช้และสามารถกำหนดหลายบทบาทให้แก่หนึ่งผู้ใช้ ผู้ใช้ ที่ได้รับการกำหนดหลายบทบาทสามารถเรียกทำงานมากกว่าหนึ่งบทบาท (ได้สูงสุดแปดบทบาท) พร้อมกันถ้าจำเป็นต้องดำเนินฟังก์ชัน การจัดการระบบ

AIX จัดให้มีชุดของบทบาทที่ กำหนดไว้แล้วสำหรับการจัดการระบบ อย่างไรก็ตาม คาดว่าลูกค้าจะต้องการ สร้างบทบาทแบบกำหนดเองของตนหรือแก้ไขบทบาทที่กำหนดไว้แล้วที่ที่มีอยู่ คำสั่งการจัดการบทบาทหลายคำสั่งมีอยู่เพื่อแสดงรายการ สร้าง แก้ไข และ ลบบทบาท AIX บทบาทสามารถ สร้างด้วยคำสั่ง **mkrole** แก้ไขด้วยคำสั่ง **chrole** ลบออกด้วยคำสั่ง **rmrole** และแสดงด้วยคำสั่ง the **lsrole**

เมื่อสร้างบทบาท AIX ใหม่ ให้พิจารณาปัญหาต่อไปนี้:

- ชื่อของบทบาทจะเป็นชื่ออะไร?
- ชื่อบทบาทเป็นสตริงข้อความ แต่ควรมีแสดงให้เห็นความสามารถของบทบาท บางส่วน ชื่อบทบาทสามารถมีอักขระที่พิมพ์ได้สูงสุด 63 อักขระ

- การอนุญาตใดที่จำเป็นสำหรับบทบาท? พิจารณาว่าการอนุญาต ควรถูกกำหนดโดยตรงให้แก่บทบาท หรือกำหนดโดยอ้อมsให้แก่บทบาท ผ่านบทบาทย่อย
- ผู้ใช้ควรต้องมีการพิสูจน์ตัวตนเมื่อเรียกทำงานบทบาทหรือไม่?

การเรียกทำงานบทบาท:

โดยดีฟอลต์ใน AIX เวอร์ชัน 6.1 และ ใหม่กว่าที่มี RBAC ที่ปรับปรุง เมื่อผู้ใช้พิสูจน์ตัวตนกับระบบ เซสชันของผู้ใช้จะไม่มีบทบาทหรือสิทธิ์ใดๆ ที่เชื่อมโยง เพื่อเชื่อมโยง บทบาทกับเซสชัน ผู้ใช้ต้องเรียกใช้คำสั่งการพิสูจน์ตัวตนต่างหาก (คำสั่ง **swrole**) เพื่อสลับไปที่บทบาท

ผู้ใช้สามารถเรียกทำงานบทบาทที่ได้ถูกกำหนดให้แก่ผู้ใช้ไว้ก่อนหน้าแล้ว เท่านั้น โดยดีฟอลต์ ผู้ใช้จำเป็นต้องพิสูจน์ตัวตนเป็นตนเองเมื่อ เข้าสู่เซสชันบทบาท หรือเมื่อเพิ่มบทบาทในเซสชันของตน เป็นทางเลือกที่บทบาท สามารถถูกกำหนดให้ไม่จำเป็นต้องทำการพิสูจน์ตัวตนกับแอ็ตทริบิวต์บทบาท **auth_mode**

การสลับไปยังเซสชันบทบาทใหม่จะสร้างเชลล์ใหม่ (เซสชัน) โดยไม่มีการสืบทอด บทบาทจากเซสชันก่อนหน้า ซึ่งกระทำได้โดยการสร้างเชลล์กระบวนการใหม่ สำหรับบทบาทและกำหนด ID บทบาท (RID) ใหม่ให้แก่กระบวนการ การสร้าง เซสชันใหม่จะคล้ายกับการใช้คำสั่ง **su** ยกเว้น ในกรณีเท่านั้นที่ ID บทบาทของกระบวนการถูกเปลี่ยนแปลง และไม่เปลี่ยนคุณสมบัติ เช่น UID หรือ GID คำสั่ง **swrole** อนุญาตให้ผู้ใช้สร้าง เซสชันบทบาทที่ประกอบด้วยบทบาทเดียว หรือหลายบทบาท ไม่มีข้อจำกัด ในการป้องกันมิให้ผู้ใช้สลับจากเซสชันบทบาทเก่าไปยังเซสชันบทบาท ใหม่ เนื่องจากเซสชันใหม่คือกระบวนการใหม่ เซสชันใหม่จะไม่สืบทอด บทบาทใดๆ จากเซสชันก่อนหน้า เพื่อเรียกคืนเซสชันก่อนหน้า ผู้ใช้ต้องออกจากเซสชันบทบาทปัจจุบัน บทบาทที่สมมติใน เซสชัน (ชุดบทบาทที่แอ็คทีฟ) สามารถแสดงรายการโดยการรันคำสั่ง **rolelist** ในเซสชัน ผู้ดูแลระบบยังสามารใช้คำสั่ง **rolelist** เพื่อแสดงรายการชุดบทบาทที่แอ็คทีฟสำหรับกระบวนการระบบที่กำหนด

ผู้ใช้สามารถถูกกำหนดชุดของบทบาทดีฟอลต์ด้วยแอ็ตทริบิวต์บทบาท **default_roles** ใหม่ซึ่งเป็นทางเลือก แอ็ตทริบิวต์นี้มุ่งใช้สำหรับสถานการณ์ที่กระบวนการ ที่ถูกสร้างขึ้นในนามของผู้ใช้จำเป็นต้องเชื่อมโยงกับชุดของบทบาท ที่กำหนดเสมอ ตัวอย่างเช่น คำสั่ง **cron** โปรแกรมอำนวยความสะดวก cron รันในแบบเบื้องหลังและรันคำสั่งในฐานะผู้ใช้ที่กำหนด เป็นไปได้ที่บางคำสั่งที่รันอาจต้องการการอนุญาต ซึ่งจำเป็น ต้องใช้ความสามารถในการกำหนดชุดของบทบาทให้แอ็คทีฟเสมอสำหรับ ID ผู้ใช้ เนื่องจากไม่มีกลไกสำหรับคำสั่ง **cron** ในการจัดการ บทบาทเหล่านี้ในภายหลัง แอ็ตทริบิวต์ **default_roles** สามารถถูกตั้งค่าเพื่อรวม ชื่อบทบาทได้สูงสุดแปดชื่อ หรือค่าพิเศษ **ALL** การตั้งค่า **default_roles=ALL** กำหนด ให้แก่บทบาทของผู้ใช้ทั้งหมดให้แก่เซสชัน ถ้าผู้ใช้ได้ถูกกำหนดไว้มากกว่าแปด บทบาท จะมีเพียงแปดบทบาทแรกเท่านั้นที่ถูกเปิดใช้งานสำหรับ เซสชัน

จำนวนบทบาทสูงสุดต่อหนึ่งเซสชัน:

ใน RBAC แบบปรับปรุง ผู้ดูแลระบบสามารถตั้งค่าทั้งระบบสำหรับจำนวน บทบาทสูงสุดที่ผู้ใช้สามารถเรียกทำงานในเซสชันบทบาท ที่กำหนดได้ โดยดีฟอลต์ ผู้ใช้สามารถเรียกทำงานได้สูงสุดแปดบทบาทต่อเซสชัน

บางสภาวะแวดล้อมอาจจำเป็นต้องมีการแบ่งหน้าที่มากขึ้นซึ่ง ผู้ใช้สามารถเรียกทำงานได้ครั้งละหนึ่งบทบาทเท่านั้น ในสภาวะแวดล้อมเหล่านี้ แอ็ตทริบิวต์ **maxroles** ของ **usw** stanza ในไฟล์ `/etc/security/login.cfg` สามารถแก้ไขเพื่อจำกัดจำนวนสูงสุดที่อนุญาตของบทบาทต่อเซสชัน แอ็ตทริบิวต์ **maxroles** สามารถตั้งค่าให้อยู่ในช่วง 1 ถึง 8 เพื่อระบุจำนวนบทบาทที่อนุญาตสูงสุดต่อหนึ่งเซสชัน

ในการแสดงค่าปัจจุบันของข้อจำกัดของจำนวนบทบาท ต่อหนึ่งเซสชัน รันคำสั่งต่อไปนี้:

```
lssec -f /etc/security/login.cfg -s usw -a maxroles
```

ในการแก้ไขระบบให้อนุญาตให้ผู้ใช้เรียกทำงานได้ครั้งละบทบาทเดียวเท่านั้น รันคำสั่งต่อไปนี้:

```
chsec -f /etc/security/login.cfg -s usw -a maxroles=1
```

การแก้ไขค่าของแอ็ตทริบิวต์ **maxroles** จะมีผลในทันที สำหรับเซสชันบทบาทใหม่ที่ถูกสร้างขึ้น และไม่จำเป็นต้องรีบูตระบบ เซสชันบทบาทที่มีอยู่แล้วก่อนการแก้ไขค่าจะไม่ได้รับผล กระทบจากการเปลี่ยนแปลง การบังคับใช้จำนวนบทบาทสูงสุดต่อหนึ่งเซสชันถูกดำเนินการในตอนเริ่มต้นเซสชัน

บทบาทที่กำหนดไว้แล้ว:

ชุดของบทบาทที่กำหนดไว้ล่วงหน้าที่ถูกกำหนดในฐานข้อมูล บทบาทแบบโลคัล (`/etc/security/roles`) บนการติดตั้ง AIX เวอร์ชัน 6.1 ใหม่และหลังจากนั้น ชุดของบทบาทนี้ต้องการจัดกลุ่มความรับผิดชอบในการดูแลจัดการ

ชุดของบทบาทนี้ทำหน้าที่เป็นวิธีการที่แนะนำของการแบ่งหน้าที่ การดูแลจัดการ ผู้ดูแลจัดการบทบาทสามารถแก้ไขหรือลบบทบาทเหล่านี้ หรือสร้างบทบาทใหม่ เท่าที่จำเป็นสำหรับสภาวะแวดล้อมของตน ต่อไปนี้แสดงรายการบทบาทที่มีให้ และคำอธิบายอย่างย่อของความสามารถของแต่ละบทบาท

ชื่อบทบาท	**คำอธิบายบทบาท**
auditadm	ผู้ดูแลระบบการตรวจสอบ บทบาท auditadm มีหน้าที่ในการ กำหนดค่านโยบายการตรวจสอบ และการล็อกของระบบ รวมถึง แอ็ตทริบิวต์ของทั้งระบบ ผู้ใช้เดียว และบทบาทเดียว บทบาทนี้มี การเข้าถึงเพื่อดูหลักฐานการตรวจสอบ
fsadm	ผู้ดูแลระบบไฟล์ บทบาท fsadm สร้างระบบไฟล์ และทำให้พร้อมใช้งานสำหรับผู้ใช้บนระบบ ความรับผิดชอบบางส่วน ของบทบาท fsadm ได้แก่: • การระบุนโยบายการเมาท์ • การแบ่งใช้นโยบาย • การกำหนดโควต้า • การพิจารณาระดับการบีบอัด • การสร้างรูปแบบระบบไฟล์ • การดำเนินการกิจกรรมการสำรองและเรียกคืนข้อมูล
isso	Information System Security Officer ISSO รับผิดชอบการสร้างและการกำหนดบทบาท ดังนั้นจึงเป็นบทบาทที่มีอำนาจมากสุดบนระบบ ความรับผิดชอบของ ISSO บางอย่างได้แก่: • การสร้างและการดูแลรักษานโยบายการรักษาความปลอดภัย • การตั้งค่ารหัสผ่านสำหรับผู้ใช้ • การตั้งค่าเน็ตเวิร์ก • การดูแลจัดการอุปกรณ์
pkgadm	ผู้ดูแลระบบซอฟต์แวร์แพ็กเกจ บทบาท pkgadm รับผิดชอบ เกี่ยวกับซอฟต์แวร์ที่ติดตั้งบนระบบ และมีสิทธิ ดีฟอลต์ในการติดตั้ง อัพเดต และลบซอฟต์แวร์ระบบ

ชื่อบทบาท	คำอธิบายบทบาท
sa	ผู้ดูแลระบบ บทบาท SA มีฟังก์ชันการทำงานสำหรับการดูแลจัดการรายวัน และรับผิดชอบในส่วน: • การดูแลจัดการผู้ใช้ (ยกเว้นการตั้งค่ารหัสผ่าน) • การดูแลจัดการระบบไฟล์ • การอัพเดตการติดตั้งซอฟต์แวร์ • การจัดการ daemon เน็ตเวิร์ก • การจัดสรรอุปกรณ์
secadm	ผู้ดูแลระบบการรักษาความปลอดภัย บทบาท secadm ดูแลรักษาการตั้งค่า ความปลอดภัยบนระบบ secadm กำหนดแอ็ตทริบิวต์เช่น ความเป็นสมาชิก ในกลุ่ม บทบาท การอนุญาต และการล้างค่าให้แก่ผู้ใช้ และกำหนด บทบาทที่ยังไม่ถูกระบุให้กับบทบาท บทบาท secadm ยังกำหนดแอ็ตทริบิวต์การรักษาความปลอดภัยแก่อ็อบเจ็กต์ระบบ รวมถึง การตั้งค่า RBAC รายการควบคุมการเข้าถึง ความเป็นเจ้าของ และความเป็นสมาชิก ความรับผิดชอบ บางอย่างของบทบาท secadm มีดังต่อไปนี้: • การกำหนดรหัสผ่านสำหรับแอคเคาต์ผู้ใช้ใหม่ • การปลดล็อกแอคเคาต์ที่ถูกล็อก
so	ผู้ควบคุมระบบ บทบาท SO มีฟังก์ชันการทำงานแบบวันต่อวัน และรับผิดชอบในส่วน: • การปิดระบบและรีบูต • การสำรองข้อมูล เรียกคืน และจัดโควต้าระบบไฟล์ • การบันทึกข้อผิดพลาดระบบ การติดตามและสถิติ • การดูแลจัดการเวิร์กโหลด
svcadm	ผู้ดูแลระบบเซอร์วิส บทบาท svcadm เปิดใช้งาน กำหนดค่า และปิดใช้งานเซอร์วิสระบบ บทบาทนี้อนุญาตให้การกำหนดค่าของ แอ็ตทริบิวต์ระบบเครือข่าย เช่น IP addresses, เส้นทาง, ชื่อโฮสต์ และนโยบายไฟร์วอลล์
sysop	ผู้ควบคุมระบบ บทบาท sysop ดูแลรักษาระบบโดยรวม ด้วยสิทธิที่รวมถึงการรันการวิเคราะห์ระบบ และการดำเนินการ ดูแลรักษาระบบรูทีน งานบางอย่างที่ sysop รับผิดชอบ ได้แก่: • การลบล็อกไฟล์ และคิวการพิมพ์ • การหยุดทำงาน และการรีสตาร์ทระบบ
useradm	ผู้ดูแลระบบผู้ใช้ บทบาท useradm รับผิดชอบงาน ในระดับสูงขึ้นไปที่เกี่ยวข้องกับการดูแลรักษาผู้ใช้โดยมีการจัดการรหัสผ่าน useradm สร้าง แก้ไข และลบแอคเคาต์ผู้ใช้ตามที่กำหนด โดยการตั้งค่าการรักษาความปลอดภัยดีฟอลต์ บทบาทนี้ยังสร้างบทบาทและกลุ่มเพิ่มเติมด้วยการตั้งค่าการรักษาความปลอดภัยดีฟอลต์

การโอนย้ายบทบาท:

ถ้าระบบ AIX ก่อนหน้า AIX เวอร์ชัน 6.1 กำลังถูกอัพเดต เป็น AIX ที่มีระดับ RBAC แบบปรับปรุง ผ่านการติดตั้งการโอนย้ายระบบ การโอนย้ายของไฟล์ /etc/security/roles จะพยายามอัพเดตไฟล์เพื่อให้มีฟังก์ชันการทำงานใหม่โดยยังคงรักษาความสามารถของบทบาทปัจจุบัน

นิยามบทบาทในไฟล์ถูกสงวนและแก้ไขเพื่อรวม ID บทบาทเฉพาะเพื่ออนุญาตให้บทบาททำงานได้อย่างถูกต้องในเฟรมเวิร์กใหม่ การอนุญาตใดๆ ในไฟล์ /etc/security/roles ที่ เป็นการอนุญาตที่ไม่รู้จักว่าเป็นการกำหนดไว้แล้วจะถูกพิจารณาเป็นการอนุญาตที่ผู้ใช้กำหนดเอง ระหว่างการโอนย้าย ชื่อการอนุญาตเหล่านี้ถูกเพิ่มเป็นรายการในฐานข้อมูลการอนุญาต /etc/security/authorizations โลคัล นอกเหนือจากการโอนย้ายนิยามบทบาทเก่าแล้ว บทบาทที่กำหนดไว้แล้วใหม่ จะถูกต่อท้ายไฟล์ หลังการโอนย้าย ผู้ดูแลระบบ ต้องตรวจสอบว่าการอนุญาตและบทบาทถูกกำหนดเท่าที่จำเป็นสำหรับ สภาวะแวดล้อมนั้น

สิทธิ์พิเศษ RBAC:

เฟรมเวิร์ก RBAC แบบปรับปรุงขึ้นอยู่กับสิทธิ์พิเศษระบบเป็นอย่างยิ่ง เพื่ออนุญาตให้ผู้ใช้ที่ไม่มีสิทธิ์พิเศษสามารถดำเนินงานสิทธิ์พิเศษ สิทธิ์พิเศษคือ กลไกที่ใช้เพื่อให้สิทธิ์กระบวนการสามารถทำฟังก์ชันการทำงานที่เพิ่มขึ้นในการเรียกใช้ระบบ

แนวคิดของสิทธิ์พิเศษโดยเริ่มแรกเป็นการสร้างระดับเคอร์เนลเนื่องจาก นิยามและการตรวจสอบส่วนใหญ่เกิดขึ้นในเคอร์เนล อย่างไรก็ตาม อินเตอร์เฟสระดับ ผู้ใช้ถูกจัดให้มีเพื่อจัดการการมอบหมายสิทธิ์พิเศษ ให้แก่คำสั่ง อุปกรณ์ หรือกระบวนการ

สิ่งสำคัญคือต้องทราบความแตกต่างระหว่างสิทธิ์พิเศษและการอนุญาต ทั้งสิทธิ์พิเศษและการอนุญาตถูกใช้เพื่อควบคุมข้อยกเว้นที่สามารถอนุญาตได้ สำหรับนโยบายความปลอดภัยระบบ ความแตกต่างของการกำหนดระหว่างสิทธิ์พิเศษและ การอนุญาตคือสิทธิ์พิเศษถูกเชื่อมโยงกับกระบวนการที่เจาะจง ขณะที่การอนุญาตถูกเชื่อมโยงกับผู้ใช้ผ่านบทบาท การอนุญาตอยู่คู่กับบทบาทและผู้ใช้ที่มีบทบาท แต่ไม่ขึ้นกับ โปรแกรมที่กำลังถูกรัน สิทธิ์พิเศษอยู่คู่กับโปรแกรมและจัดให้มี กลไกที่ปรับแต่งนโยบายการรักษาความปลอดภัยระบบให้ดีที่สุด เนื่องจากสิทธิ์พิเศษที่เชื่อมโยงเหล่านี้ กระบวนการมีคุณสมบัติในการดำเนินการสิทธิ์พิเศษที่เกี่ยวข้องได้

สิทธิ์พิเศษถูกกำหนดในเคอร์เนล AIX เป็นบิตเฉพาะของ bit-mask ซึ่งบังคับใช้การควบคุมการเข้าถึงผ่านการดำเนินการ สิทธิ์พิเศษ โดยจัดให้มีมากกว่า 100 สิทธิ์พิเศษใน AIX เป็นการจัดให้มีการควบคุมการดำเนินการสิทธิ์พิเศษเป็นกลุ่มย่อยที่แยกย่อยมาก เมื่อ พิจารณาการเข้าถึงในการเรียกใช้ระบบ เคอร์เนลจะพิจารณาว่ากระบวนการ มีบิตสิทธิ์พิเศษที่เชื่อมโยงที่จำเป็นหรือไม่ จากนั้นให้สิทธิ์หรือปฏิเสธการร้องขอ

สิทธิ์พิเศษถูกกำหนดให้แก่การร้องขอคำสั่งผ่านฐานข้อมูลคำสั่งสิทธิ์พิเศษ และสิทธิ์พิเศษถูกใช้เพื่อควบคุมการเข้าถึงอุปกรณ์ผ่าน ฐานข้อมูลอุปกรณ์สิทธิ์พิเศษ

การตั้งชื่อและลำดับชั้นสิทธิ์พิเศษ:

สิทธิ์พิเศษ AIX ไม่สามารถถูกสร้าง แก้ไข หรือลบโดยผู้ดูแลระบบ

รายการของสิทธิ์พิเศษที่มีอยู่และคำอธิบายอย่างย่อของสิทธิ์พิเศษ สามารถแสดงบนระบบโดยการรันคำสั่งต่อไปนี้:

```
lspriv -v
```

สิทธิ์พิเศษที่จัดให้มีบน AIX ถูก แสดงในสิทธิ์พิเศษ AIX สิทธิ์พิเศษAIX ทั้งหมดมีการแทน ด้วยข้อความของบิตสิทธิ์พิเศษที่ขึ้นต้นด้วย **PV_** ข้อกำหนดการตั้งชื่อที่ใช้หลังคำนำหน้า **PV_** หมายถึงความสัมพันธ์ เชิงลำดับชั้นระหว่างสิทธิ์พิเศษ ตัวอย่าง

สิทธิ์พิเศษการตรวจสอบ **PV_AU_** เป็น พาเรนต์ของสิทธิ์พิเศษ **PV_AU_ADD, PV_AU_ADMIN, PV_AU_READ, PV_AU_WRITE** และ **PV_AU_PROC** เมื่อตรวจสอบสิทธิ์พิเศษ อันดับแรกระบบจะพิจารณาว่ากระบวนการ มีสิทธิ์พิเศษต่ำสุดที่จำเป็นหรือไม่ จากนั้นดำเนินการต่อในลำดับชั้นสูงขึ้นไป เพื่อตรวจหาการมีอยู่ของสิทธิ์พิเศษที่มีอำนาจมากกว่า สิทธิ์พิเศษ **PV_ROOT** คือสิทธิ์พิเศษที่พิเศษซึ่งแทนพาเรนต์ของสิทธิ์พิเศษทั้งหมด ยกเว้น **PV_SU_** กระบวนการที่ถูกกำหนดสิทธิ์พิเศษ **PV_ROOT** มีการทำงานเหมือนได้รับการกำหนดสิทธิ์พิเศษทุกอย่างบนระบบยกเว้น **PV_SU_**

ชุดสิทธิ์พิเศษของกระบวนการ:

หลายชุดของสิทธิ์พิเศษถูกกำหนดในเคอร์เนลเพื่อให้มี การควบคุมแตกต่างกันสำหรับการดำเนินการสิทธิ์พิเศษ หลายชุดสิทธิ์พิเศษอนุญาต ให้ระบบปฏิบัติการบังคับใช้การควบคุมสิทธิ์พิเศษแบบไดนามิก และอนุญาตให้ แอ็พพลิเคชันจัดการกฎสิทธิ์พิเศษต่ำที่สุด

สิทธิ์พิเศษเชื่อมโยงกับกระบวนการผ่านชุดสิทธิ์พิเศษ ต่อไปนี้:

Limiting Privilege Set (LPS)
กำหนดขีดจำกัดแน่นอน (hard limit) ของสิทธิ์พิเศษสำหรับกระบวนการที่กำหนด ไม่มี การเพิ่มสิทธิ์พิเศษในระบบที่สามารถเพิ่มสิทธิ์พิเศษของกระบวนการได้เกินค่านี้ ซึ่ง หมายความว่ากระบวนการไม่สามารถขอสิทธิ์พิเศษเพิ่มมากกว่าค่านี้โดยใช้ อินเตอร์เฟสระบบที่กำหนดแบบใดๆ หรือกล่าวอีกอย่างคือ กระบวนการถูกจำกัด ด้วยสิทธิ์พิเศษเหล่านี้ไม่ว่าเวลาใด นี้ยังหมายความว่าส่วนที่เหลือ ของชุดสิทธิ์พิเศษจะเป็นเซ็ตย่อยของ LPS เสมอ แม้ว่า LPS ไม่สามารถ ขยายได้ ทุกกระบวนการจะมีสิทธิ์ลด LPS อย่างไรก็ตาม เมื่อ LPS ถูกลด จะไม่สามารถขยายกลับไปเป็นค่าเดิมได้ การลด ค่าของ LPS อนุญาตให้กระบวนการจำกัดขอบเขตที่เกี่ยวกับ สิทธิ์พิเศษที่เชื่อมโยง ตัวอย่าง กระบวนการอาจลด LPS เฉพาะ ก่อนการรันโปรแกรมที่ผู้ใช้จัดหามาแบบกำหนดเอง โดยดีฟอลต์ สิทธิ์พิเศษทั้งหมด ที่มีอยู่บนระบบถูกตั้งค่าใน LPS สำหรับกระบวนการ

Maximum Privilege Set (MPS)
ชุดสมบูรณ์ของสิทธิ์พิเศษที่กระบวนการได้รับอนุญาตให้ใช้ MPS สามารถรวมสิทธิ์พิเศษใดๆ ใน LPS แต่ไม่สามารถเกิน LPS MPS สามารถเปลี่ยนแปลงระหว่างอายุการทำงานของกระบวนการได้ด้วยเหตุผลต่างๆ ต่อไปนี้ คือเหตุผลบางอย่าง:

- เมื่อกระบวนการปัจจุบันทำงานคำสั่งสิทธิ์พิเศษอื่น จากนั้นได้รับสิทธิ์พิเศษที่เกี่ยวข้องเพิ่ม
- ถ้ากระบวนการมีสิทธิ์พิเศษที่ถูกต้อง จะสามารถขยาย MPS ได้โดยการโปรแกรม ในลักษณะไดนามิก

Effective Privilege Set (EPS)
รายการของสิทธิ์พิเศษซึ่งขณะนี้แอ็คทีฟสำหรับกระบวนการ EPS เป็นเซ็ตย่อยของ MPS ของกระบวนการเสมอ และถูกใช้โดยเคอร์เนลเพื่อดำเนินการ ตรวจสอบการเข้าถึงในส่วนที่เกี่ยวกับการดำเนินการสิทธิ์พิเศษ EPS สามารถถูกปรับเปลี่ยน โดยกระบวนการและสามารถเท่ากับ MPS ได้ แต่ไม่สามารถเกิน MPS การปรับเปลี่ยนแบบไดนามิก ของ EPS สามารถดำเนินการโดยกระบวนการเพื่อบังคับใช้กฎสิทธิ์พิเศษต่ำสุด ตัวอย่าง โค้ดส่วนผู้ใช้สามารถเพิ่มบิตสิทธิ์พิเศษการตรวจสอบ ใน EPS ได้โดยใช้ **priv_raise** API ก่อนการเรียกใช้ระบบที่เกี่ยวกับการตรวจอสบ หรือการเรียกใช้เคอร์เนล สิทธิ์พิเศษยังสามารถถูกลดลงด้วย **priv_lower** API เมื่อการเรียกใช้การตรวจสอบกลับมา

Inheritable Privilege Set (IPS)
สิทธิ์พิเศษที่ถูกส่งจากกระบวนการพาเรนต์ไปยัง MPS และ EPS ของกระบวนการชายด์ IPS สามารถรวมสิทธิ์พิเศษใดๆ ใน LPS แต่ไม่สามารถเกิน LPS IPS สามารถถูกตั้งค่าในกระบวนการด้วยวิธีต่อไปนี้:

- ถ้ากระบวนการมีสิทธิ์พิเศษที่เหมาะสม จะสามารถขยาย IPS ได้โดยการโปรแกรม ผ่านการเรียกใช้ระบบ **setppriv**
- เมื่อคำสั่งสิทธิ์พิเศษถูกรัน สิทธิ์พิเศษที่ระบุในแอ็ตทริบิวต์ **inheritprivs** ที่สัมพันธ์กับคำสั่งถูกกำหนดใน IPS

Used Privilege Set (UPS)

แสดงถึงสิทธิ์พิเศษที่ได้ถูกใช้สำหรับการตรวจสอบการเข้าถึงระหว่าง อายุของกระบวนการ UPS สามารถใช้เพื่อพิจารณาสิทธิ์พิเศษที่จำเป็น ต่อกระบวนการ เมื่อเคอร์เนลตรวจสอบว่ากระบวนการมีอภิสิทธิ์ที่กำหนดหรือไม่ จะเก็บการตรวจสอบที่สำเร็จใน UPS สำหรับสิทธิ์พิเศษ

Workload Partition Privilege Set (WPS)

WPAR ระบบสามารถถูกจำกัดไม่ให้อนุญาตการดำเนินการสิทธิ์พิเศษทั้งหมด ที่ได้รับอนุญาตให้ WPAR โกลบอล การดำเนินการสิทธิ์พิเศษที่อนุญาตใน WPAR ระบบบาสมารถควบคุมผ่าน WPS root โกลบอลสามารถกำหนด ชุดสิทธิ์พิเศษที่จำกัดให้แก่ WPAR โดยใช้ WPS WPS สามารถถูกระบุ ในไฟล์คอนฟิกูเรชัน `/etc/wpar/secattrs` ระหว่างการเริ่มทำงาน WPAR โดยใช้คำสั่ง **/usr/sbin/startwpar** กระบวนการทั้งหมด ที่กำลังรันใน WPAR มี LPS เท่ากับ WPS ของตน

ผู้ดูแลระบบสามารถใช้คำสั่งการดูแลจัดการเพื่อแสดงรายการและแก้ไข ชุดสิทธิ์พิเศษของกระบวนการที่แตกต่างกัน คำสั่ง **lssecattr** สามารถใช้แสดง LPS, MPS, EPS, IPS และ UPS คำสั่ง **setsecattr** สามารถ ใช้เพื่อแก้ไข LPS, MPS, EPS และ IPS UPS ไม่สามารถถูกแก้ไขด้วย คำสั่ง **setsecattr** เนื่องจาก UPS เป็นแอ็ตทริบิวต์แบบอ่านอย่างเดียว

ฐานข้อมูลคำสั่งสิทธิพิเศษ:

การอนุญาต บทบาท และสิทธิพิเศษอนุญาตให้มีการใช้การควบคุม การรักษาความปลอดภัยแบบกลุ่มย่อย อย่างไรก็ตาม การใช้ประโยชน์ของ RBAC โดยการดำเนินการระบบที่แตกต่างกัน อนุญาตให้บังคับใช้นโยบายการรักษาความปลอดภัย RBAC ได้

ในขณะที่คำสั่ง AIX เก่าบางคำสั่ง ตรวจสอบการอนุญาตโดยตรง จำเป็นที่โค้ดที่รันได้ นั้นต้องถูกแก้ไขเพื่อให้ดำเนินการตรวจสอบเหล่านี้ โหมด RBAC แบบปรับปรุงจัดให้มี เฟรมเวิร์กที่บังคับใช้การตรวจสอบการอนุญาต และให้สิทธิพิเศษที่เกี่ยวข้องผ่านทางฐานข้อมูลคำสั่งสิทธิพิเศษโดยไม่จำเป็นต้องเปลี่ยนแปลงไฟล์ที่รันได้ ของระบบ

ฐานข้อมูลคำสั่งสิทธิพิเศษให้สิทธิการเข้าถึงและอำนาจแก่ผู้ใช้สำหรับคำสั่ง ที่ไม่สามารถรันได้ หรือสำหรับผู้ใช้ที่ไม่มีสิทธิพิเศษที่เหมาะสม สำหรับการดำเนินงาน ฐานข้อมูลบันทึกข้อมูลการอนุญาต สำหรับคำสั่งที่เจาะจงรวมถึงสิทธิพิเศษที่ได้รับอนุญาต ลงในกระบวนการถ้าการตรวจสอบการอนุญาตสำเร็จ เมื่อฐานข้อมูลเก็บ แบบโลคัล จะมีอยู่ในไฟล์ `/etc/security/privcmds` และมี stanzas ของข้อมูลในรูปของแอ็ตทริบิวต์คำสั่ง-กับ-การรักษาความปลอดภัย ต่อไปนี้คือบางส่วนของคีย์แอ็ตทริบิวต์ในฐานข้อมูลนี้ (สำหรับรายละเอียด โดยสมบูรณ์ของแอ็ตทริบิวต์ทั้งหมด ดูที่ไฟล์ `/etc/security/privcmds`)

accessauths

แสดงรายการการอนุญาตการเข้าถึงที่ป้องกันการเรียกใช้งานคำสั่ง ผู้ใช้ที่มีการอนุญาตหนึ่งในรายการการอนุญาตที่อนุญาตเพื่อรันคำสั่ง และดำเนินการสิทธิพิเศษบางส่วนหรือทั้งหมดที่ถูกรวมอยู่ใน คำสั่ง

innateprivs

สิทธิพิเศษเริ่มต้นคือสิทธิพิเศษที่กำหนดให้แก่กระบวนการถ้าผู้ร้องขอ ผ่านการตรวจสอบการอนุญาตการเข้าถึงได้สำเร็จ

authprivs

สิทธิพิเศษที่อนุญาตคือสิทธิพิเศษเพิ่มเติมที่กำหนดให้แก่กระบวนการ ถ้าผู้ใช้มีการอนุญาตที่เกี่ยวข้อง แอ็ตทริบิวต์นี้อนุญาตให้มีการควบคุมคำสั่ง เป็นกลุ่มย่อยมากขึ้นเพื่ออนุญาตให้กลุ่มผู้ใช้เฉพาะที่จำกัดสามารถดำเนินการ สิทธิพิเศษที่เพิ่มมา

inheritprivs

สิทธิพิเศษที่สืบทอดได้คือสิทธิพิเศษที่กระบวนการส่งต่อไปยังกระบวนการ ชายด์

secflags

รายการของแฟล็กการรักษาความปลอดภัย FSF_EPS คือแฟล็กซึ่งทำให้ maximum privilege set (MPS) ถูกโหลดเข้าสู่ effective privilege set (EPS) เมื่อรัน คำสั่ง

เมื่อผู้ใช้บนระบบที่ใช้โหมด RBAC แบบปรับปรุงพยายามรันคำสั่ง อันดับแรกคำสั่งจะถูกตรวจสอบในฐานข้อมูลคำสั่งสิทธิพิเศษ ถ้ามีคำสั่ง อยู่ในฐานข้อมูล การตรวจสอบถูกดำเนินการเทียบกับการอนุญาตที่เกี่ยวข้อง กับเซสชันของผู้ใช้ และค่าของแอ็ตทริบิวต์ **accessauths** สำหรับคำสั่ง ถ้าเซสชันมีการอนุญาตหนึ่งในรายการการอนุญาตที่แสดง ผู้ใช้จะสามารถรันคำสั่งได้ไม่ว่าผู้ใช้จะผ่านการตรวจสอบการทำงาน DAC สำหรับคำสั่งหรือไม่ เมื่อมีการร้องขอ กระบวนการประมวลผลคำสั่งมีสิทธิพิเศษ แสดงรายการในแอ็ตทริบิวต์ **innateprivs** ถูกกำหนดใน maximum privilege set (MPS) การตรวจสอบการอนุญาตเพิ่มเติมถูกดำเนินการด้วยคู่การอนุญาต–สิทธิพิเศษ ที่แสดงรายกาในแอ็ตทริบิวต์ **authprivs** ถ้าเซสชันมีหนึ่งในการอนุญาต ที่แสดงรายการ สิทธิพิเศษที่เกี่ยวข้องจะถูกเพิ่มใน MPS ของการประมวลผลคำสั่งเช่นกัน รายการคำสั่งในฐานข้อมูลคำสั่งสิทธิพิเศษ ที่มีค่า **FSF_EPS** ถูกตั้งค่าในแอ็ตทริบิวต์ **secflags** จะกำหนด สิทธิพิเศษทั้งหมดใน MPS ไปยัง effective privilege set (EPS) เมื่อคำสั่งถูกร้องขอ

คำสั่งถูกเรียกเป็นคำสั่งสิทธิพิเศษเมื่อถูกรวมในฐานข้อมูลคำสั่ง สิทธิพิเศษ ขณะที่โปรแกรม setuid ที่ไม่แสดงรายการในฐานข้อมูล โดยทางเทคนิคแล้วยังคงเป็นคำสั่งสิทธิพิเศษ โดยไม่ถูกอ้างอิงเป็นคำสั่ง สิทธิพิเศษเมื่อมีการอธิบายลักษณะการทำงาน RBAC ถ้าคำสั่งไม่มีรายการอยู่ ในฐานข้อมูลคำสั่งสิทธิพิเศษ ดังนั้นคำสั่งไม่เป็นคำสั่งสิทธิพิเศษและ การเข้าถึงถูกบังคับโดย DAC และตัวคำสั่งเอง นอกจากนั้น ถ้า คำสั่งถูกแสดงรายการในฐานข้อมูลคำสั่งสิทธิพิเศษ แต่เซสชันของผู้ใช้ไม่มีการอนุญาตที่อนุญาตการเรียกใช้คำสั่ง ระบบจะกลับคืนไปที่การตรวจสอบการเข้าถึง DAC และอนุญาตให้คำสั่งถูกรันได้ถ้า การตรวจสอบนี้สำเร็จ

คำสั่งการจัดการหลายๆ คำสั่งได้ถูกสร้างขึ้นเพื่อปรับเปลี่ยนและเคียวรี ฐานข้อมูลคำสั่งสิทธิพิเศษ รายการในฐานข้อมูลคำสั่งสิทธิพิเศษสามารถ ถูกสร้างหรือแก้ไขด้วยคำสั่ง **setsecattr** แสดงด้วยคำสั่ง **lssecattr** และลบออกด้วยคำสั่ง **rmsecattr**

การพิจารณาการอนุญาตที่จำเป็นสำหรับคำสั่ง:

แอ็พพลิเคชันการดูแลจัดการระบบหลายๆ แอ็พพลิเคชันจำเป็นต้องได้รับการอนุญาต เพื่อรันได้อย่างเหมาะสม แม้ว่าจะมีชุดของคำสั่งที่คำสั่งให้ไว้ในฐานข้อมูลคำสั่งสิทธิพิเศษ ผู้ดูแลระบบก็อาจจำเป็น ต้องเพิ่มรายการที่เจาะจงสำหรับสภาวะแวดล้อมของตน ฐานข้อมูลคำสั่ง สิทธิพิเศษอนุญาตให้เพิ่มรายการลงในฐานข้อมูลได้ การอนุญาต ที่เหมาะสมต้องแสดงอยู่ในแอ็ตทริบิวต์ **accessauths** เพื่อ ให้สามารถเข้าถึงคำสั่งได้

มีสองวิธีที่การพิสูจน์ตัวตนสามารถใช้และเช็กอินระบบปฏิบัติการ AIX โดยใช้กรอบงาน RBAC ที่พัฒนาแล้ว:

- Access Auths (การอนุญาตการเข้าถึง): แอ็ตทริบิวต์ที่ระบุใน ฐานข้อมูลคำสั่งสิทธิพิเศษและมีรายการของชื่อการอนุญาตที่คั่นด้วยคอมมา ผู้ใช้ที่เซสชันปัจจุบันของตนมีการอนุญาตหนึ่งในรายการการอนุญาต ที่อนุญาตให้รันคำสั่ง นี้ถูก ตรวจสอบโดยโหลดเดอร์ระบบขณะรันไฟล์ที่ทำงานได้สิทธิพิเศษ ที่ได้รับการป้องกัน
- Check Auths (checkauths()): การอนุญาตที่เจาะจง หรือรายการ การอนุญาตที่สามารถถูกตรวจสอบได้ในหลักการ โดยใช้ checkauths() API การอนุญาตที่ระบุถูกตรวจสอบเทียบกับการอนุญาตที่ แสดงในบทบาทภายในเซสชันปัจจุบัน จากผลลัพธ์ ของการตรวจสอบนี้ โปรแกรมอาจดำเนินงานสิทธิพิเศษ

ก่อนการเพิ่มคำสั่งในฐานข้อมูลคำสั่งสิทธิพิเศษ ต้องมีการพิจารณา ชุดการอนุญาตก่อนเพื่อให้แน่ใจว่าการทำงานคำสั่งได้รับอนุญาต โปรแกรมหรือแอ็พพลิเคชันอาจดำเนินการตรวจสอบการอนุญาตเพิ่มเติม เป็นการภายใน เป็นความจำเป็นที่ต้องพิจารณารายการของการอนุญาต ที่ใช้ในกระบวนการที่สามารถกำหนดได้ขณะสร้างบทบาทแบบกำหนดเอง

ต่อไปนี้คือยุทธวิธีเบื้องต้นในการพิจารณาการอนุญาตที่จำเป็น สำหรับคำสั่ง:

1. กำหนดสิทธิพิเศษ **PV_ROOT** ให้แก่เชลล์การร้องขอ หรือ สมมติบทบาทด้วยการอนุญาต *aix*

   สำคัญ: ใน WPAR โกลบอล สิทธิพิเศษ **PV_ROOT** ต้องถูกกำหนดให้แก่ ชุดสิทธิพิเศษสูงสุดและใช้งานอยู่ของกำหนดค่าเชลล์การร้องขอ ภายใน WPAR ระบบ สิทธิพิเศษนี้ยังต้องถูกเพิ่มในชุดสิทธิพิเศษที่ สืบทอดของกระบวนการ
2. รันคำสั่ง
3. บันทึกการอนุญาตที่ใช้สำหรับกระบวนการ
4. เก็บข้อมูลการอนุญาตที่บันทึกภายใต้ *Access Auths* ใน แอ็ตทริบิวต์ **accessauths** ของคำสั่งในฐานข้อมูลคำสั่ง สิทธิพิเศษการอนุญาตที่รายงานภายใต้ *Check Auths* สามารถ นำไปใช้ขณะสร้างบทบาทในระบบ

ขั้นตอนเหล่านี้ควรถูกดำเนินการในสภาวะแวดล้อมที่มีการควบคุมเนื่องจาก สิทธิพิเศษ **PV_ROOT** ถูกกำหนดในเชลล์ หรือสมมติ บทบาทด้วยการอนุญาต *aix* และเนื่องจากทั้งสองวิธีเหล่านี้ มีประสิทธิภาพอย่างยิ่ง นอกจากนั้น การรันคำสั่งอาจมีผลกระทบ ต่อระบบที่อาจส่งผลถึงผู้ใช้อื่นๆ ในทางปฏิบัติ นี้จะคล้าย วิธีลองผิดลองถูก เพื่อให้ได้รับ ชุดการอนุญาตโดยสมบูรณ์ คำสั่งอาจจำเป็นต้องรัน ซ้ำด้วยค่าแฟล็กและอ็อพชันแตกต่างกัน และอาจต้องใช้เวลา นานสำหรับแอ็พพลิเคชันที่มีการรันเป็นเวลานาน ชุดการอนุญาตที่จำเป็น ของกระบวนการสามารถรวบรวมได้โดยง่ายโดยใช้ขั้นตอนใดขั้นตอนหนึ่ง ต่อไปนี้ ซึ่งสามารถดำเนินการโดยผู้ดูแลระบบที่มีสิทธิ ที่เหมาะสม:

traceauth

ระบุอาร์กิวเมนต์ที่คำสั่งจะใช้ทำงาน คำสั่ง **traceauth** รันคำสั่งและบันทึกการอนุญาตทั้งสองประเภทที่ใช้ตลอดอายุการทำงานของกระบวนการ เมื่อคำสั่งเสร็จสิ้น คำสั่ง **traceauth** จะแสดงการอนุญาตที่ใช้บน **stdout**

lssecattr

ถ้าคำสั่งเป็นกระบวนการที่รันเป็นเวลานาน คำสั่ง **lssecattr** สามารถใช้เพื่อแสดงการอนุญาตที่ใช้โดยกระบวนการเพื่อ เปิดใช้งานการติดตามการอนุญาตในระบบ รันคำสั่ง ต่อไปนี้:

setrunmode -c; setsecconf -o traceauth=enableในการ แสดงการอนุญาตที่การอนุญาตสำหรับกระบวนการ รันคำสั่ง **lssecattr** ดังนี้ โดยแทนค่า PID ของกระบวนการที่กำลังถูกมอนิเตอร์:

lssecattr -p -A *PID*

หลังการอนุญาตที่จำเป็นได้รับการพิจารณาแล้ว ดำเนิน ขั้นตอนใน “การเพิ่มคำสั่งในฐานข้อมูลคำสั่งที่มีสิทธิพิเศษ” ในหน้า 109 เพื่อ เพิ่มคำสั่งในฐานข้อมูลคำสั่งสิทธิพิเศษ คำสั่งควร ถูกรันโดยผู้ใช้ที่ได้รับอนุญาตเพื่อให้แน่ใจว่าจะรันได้อย่างถูกต้อง

การพิจารณาสิทธิพิเศษที่จำเป็นสำหรับคำสั่ง:

หลายแอ็พพลิเคชันจำเป็นต้องมีสิทธิพิเศษที่จำเป็นเพื่อ ให้ทำงานได้อย่างถูกต้อง แม้ว่าจะมีชุดของคำสั่งที่คำสั่ง ในฐานข้อมูลคำสั่งสิทธิพิเศษ ผู้ดูแลระบบก็อาจจำเป็น ต้องเพิ่มรายการที่เจาะจงสำหรับแอ็พพลิเคชันหรือสภาวะแวดล้อมของตน ฐานข้อมูลคำสั่งสิทธิพิเศษอนุญาตให้เพิ่มรายการสำหรับคำสั่ง และสิทธิพิเศษที่เกี่ยวข้อง

ก่อนการเพิ่มคำสั่งในฐานข้อมูลคำสั่งสิทธิพิเศษ ต้องมีการพิจารณา ชุดสิทธิพิเศษที่จำเป็นขั้นต่ำก่อนเพื่อให้แน่ใจว่า การทำงานคำสั่งจะมีความปลอดภัยเท่าที่จะเป็นไปได้ สิทธิพิเศษใดๆ ที่อนุญาต ในภายหลังสิทธิที่จำเป็นเหล่านั้นเพื่อการทำงานที่ถูกต้องจะละเมิดกฎ สิทธิพิเศษอย่างน้อยที่สุด ดังนั้น ขั้นตอนสำคัญในการเพิ่มคำสั่งสิทธิพิเศษ ให้กับระบบคือการพิจารณาสิทธิพิเศษที่จำเป็นขั้นต่ำสุด

ต่อไปนี้คือกลยุทธ์เบื้องต้นในการพิจารณาสิทธิพิเศษจำเป็น ขั้นต่ำสุดสำหรับคำสั่ง:

1. Information System Security Officer (ISSO) หรือผู้ใช้ที่มี บทบาท isso สามารถกำหนดสิทธิพิเศษ **PV_ROOT** ให้ผู้ดูแลระบบ เรียกทำงานคำสั่งที่ถูกกำหนดไปยังฐานข้อมูลสิทธิพิเศษ การกำหนดค่า ของสิทธิ์ **PV_ROOT** จะเรียกใช้เชลล์จะสำเร็จ โดยใช้คำสั่ง setsecattr ตัวอย่าง:

   setsecattr -p eprivs=PV_ROOT mprivs=PV_ROOT $$
2. รันคำสั่งเพื่อรวบรวมชุดของสิทธิพิเศษ
3. บันทึกชุดสิทธิพิเศษที่ใช้สำหรับกระบวนการ
4. เก็บสิทธิพิเศษที่จำเป็นในแอ็ตทริบิวต์ **innateprivs** ของคำสั่งในฐานข้อมูลคำสั่งสิทธิพิเศษ

ขั้นตอนเหล่านี้ควรถูกดำเนินการในสภาวะแวดล้อมที่ควบคุมเนื่องจาก สิทธิพิเศษ **PV_ROOT** ถูกกำหนดในเชลล์และสิทธิพิเศษ **PV_ROOT** มีประสิทธิภาพอย่างยิ่ง นอกจากนั้น การรันคำสั่งอาจมีผลกระทบ ต่อระบบที่อาจส่งผลถึงผู้ใช้อื่นๆ ในทางปฏิบัติ นี้จะคล้าย วิธีลองผิดลองถูก เพื่อให้ได้รับ ชุดสิทธิพิเศษโดยสมบูรณ์ คำสั่งอาจจำเป็นต้องรัน ซ้ำด้วยค่าแฟล็กและอ็อพชันแตกต่างกัน และอาจต้องใช้เวลา นานสำหรับแอ็พพลิเคชันที่มีการรันเป็นเวลานาน ชุดสิทธิพิเศษที่จำเป็น ของกระบวนการสามารถรวบรวมได้โดยง่ายโดยใช้ขั้นตอนใดขั้นตอนหนึ่ง ต่อไปนี้ ซึ่งสามารถดำเนินการโดยผู้ดูแลระบบที่มีสิทธิที่เหมาะสม:

tracepriv

รับค่าอาร์กิวเมนต์ที่คำสั่งใช้เพื่อทำงาน คำสั่ง **tracepriv** รันคำสั่งและบันทึกสิทธิพิเศษที่ใช้ตลอดอายุการทำงาน ของกระบวนการ เมื่อคำสั่งเสร็จสิ้น คำสั่ง **tracepriv** จะแสดงสิทธิพิเศษที่ใช้บน **stdout**

lssecattr

ถ้าคำสั่งเป็นกระบวนการที่รันเป็นเวลานาน คำสั่ง **lssecattr** สามารถใช้เพื่อแสดงสิทธิพิเศษที่ใช้โดยกระบวนการ ในการแสดง ชุดสิทธิ์พิเศษที่ใช้สำหรับกระบวนการ รันคำสั่งดังนี้ โดยแทนค่า PID ของกระบวนการที่กำลังถูกมอนิเตอร์:

lssecattr -p -a uprivs *PID*

หลังจากกำหนดสิทธิ์ใช้งานขั้นต่ำที่จำเป็นแล้ว ให้ดำเนินการขั้นตอนใน “การเพิ่มคำสั่งในฐานข้อมูลคำสั่งที่มีสิทธิพิเศษ” ในหน้า 109 เพื่อเพิ่มคำสั่งในฐานข้อมูลคำสั่งสิทธิพิเศษ จากนั้นควรรัน คำสั่งโดยผู้ใช้ที่ได้รับอนุญาตเพื่อให้แน่ใจว่าจะรันได้อย่างถูกต้อง

การเพิ่มสิทธิพิเศษ:

เมื่อมีการสร้างกระบวนการใหม่โดยการเรียกใช้ระบบ **fork fork** จะให้ กระบวนการมีสิทธิพิเศษเหมือนกับกระบวนการพาเรนต์ (กระบวนการที่เรียก การเรียกใช้ระบบ **fork**) เมื่อกระบวนการดำเนินการเรียกใช้ระบบ **exec** บนไฟล์เรียกทำงาน **exec** คำนวณสิทธิพิเศษใหม่สำหรับไฟล์เรียกทำงาน ตามค่าสิทธิพิเศษที่ขณะนี้ **exec** ถือครอง และ สิทธิพิเศษที่ถือครองโดยไฟล์เรียกทำงาน

สิทธิพิเศษที่เพิ่มถูกคำนวณดังนี้:

1. อันดับแรก ยูเนียน (การดำเนินการ bitwise-OR) ของสิทธิพิเศษที่สืบทอดได้ถูกถือครอง โดยกระบวนการเก่า (พาเรนต์) และชุดของสิทธิ์พิเศษเริ่มต้นที่ถือครองโดย ไฟล์เรียกทำงานจะถูกคำนวณ

2. ถ้าผู้ใช้ได้รับอนุญาตอย่างเหมาะสม ยูเนียน (bitwise-OR) ของ ผลลัพธ์จากขั้นตอนก่อนหน้าและสิทธิ์พิเศษที่ได้รับอนุญาตจะถูกคำนวณ
3. ถ้ามีการจำกัดสิทธิ์พิเศษ ดังนั้น intersection ของผลลัพธ์ จากขั้นตอนก่อนหน้า และสิทธิ์พิเศษที่จำกัดจะถูกคำนวณ การจำกัด สิทธิ์พิเศษ ถ้ามี จะถูกสืบทอดผ่านการเรียกใช้ระบบ **exec**
4. ชุดของสิทธิ์พิเศษที่เป็นผลจากการยูเนียน จะเป็นชุดของสิทธิ์พิเศษ สูงสุดสำหรับกระบวนการใหม่
5. ถ้าสิทธิ์พิเศษที่สืบทอดมีอยู่ในไฟล์เรียกทำงาน จะถูกกำหนด ไปยังชุดสิทธิ์พิเศษที่สืบทอดได้ในกระบวนการใหม่ มิฉะนั้น ชุดของสิทธิ์พิเศษที่สืบทอด ได้ที่ถือครองโดยกระบวนการเก่า (พาเรนต์) ถูกส่งต่อไปในชุดของสิทธิ์พิเศษที่สืบทอดได้ของกระบวนการใหม่

ถ้าไฟล์เรียกทำงานมีแฟล็กการรักษาความปลอดภัยไฟล์ **FSF_EPS** ถูกตั้งค่า ชุด ของสิทธิ์พิเศษที่มีผลสำหรับกระบวนการใหม่จะเหมือนกับชุดของ สิทธิ์พิเศษสูงสุด มิฉะนั้น สิทธิ์พิเศษที่มีผลสำหรับกระบวนการใหม่ จะเหมือนกับสิทธิ์พิเศษที่สืบทอดได้ที่ถือครองโดยกระบวนการเก่า (พาเรนต์)

การเพิ่มคำสั่งในฐานข้อมูลคำสั่งที่มีสิทธิพิเศษ:

คุณควรพิจารณาอย่างระมัดระวังก่อนที่จะเพิ่มคำสั่ง ในฐานข้อมูลคำสั่งที่มีสิทธิพิเศษเพื่อให้แน่ใจว่ามีการกำหนดสิทธิพิเศษและ การอนุญาตที่เหมาะสม

ดูที่ไฟล์ `/etc/security/privcmds` สำหรับรายละเอียดโดยสมบูรณ์ของแอ็ตทริบิวต์ที่ใช้ได้สำหรับคำสั่ง โดยสามารถใช้คำถามต่อไปนี้เพื่อเป็นแนวทางในการพิจารณารายการ ที่จำเป็นสำหรับคำสั่ง:

1. การเข้าถึงเพื่อควบคุมการอนุญาตควรรันคำสั่งหรือไม่?

   YES ถ้าไม่มีการอนุญาต ให้สร้างขึ้นด้วยคำสั่ง **mkauth** ระบุ การอนุญาตในแอ็ตทริบิวต์ **accessauths**

   NO ถ้าผู้ใช้ทั้งหมดควรได้รับอนุญาตให้รันคำสั่ง ให้ระบุการอนุญาต **ALLOW_ALL** ในแอ็ตทริบิวต์ **accessauths**
2. เจ้าของหรือกลุ่มของคำสั่งควรได้รับอนุญาตให้รัน คำสั่งหรือไม่แม้ว่าไม่มีการอนุญาตที่ถูกต้อง?

   YES เพิ่มการอนุญาต **ALLOW_OWNER** หรือ **ALLOW_GROUP** ในรายการการอนุญาตในแอ็ตทริบิวต์ **accessauths**
3. เมื่อคำสั่งถูกเรียกใช้งาน จำเป็นต้องตั้งค่าสิทธิพิเศษ อย่างชัดเจนหรือไม่?

   YES รันคำสั่งด้วยอ็อพชันที่ต่างกันในฐานะผู้ใช้ root ด้วยคำสั่ง **tracepriv** เพื่อพิจารณา สิทธิที่จำเป็นสำหรับแอ็ตทริบิวต์ **innateprivs**
4. ผู้ใช้ที่มีการอนุญาตที่ระบุควรได้รับสิทธิเพิ่มเติม หรือไม่?

   YES ระบุคู่การอนุญาต-สิทธิเพิ่มเติมในคุณสมบัติ **authprivs**
5. คำสั่งต้องมีการทำงานคล้ายกับโปรแกรม SUID หรือ SGID หรือไม่?

   YES ระบุ EUID หรือ EGID ตามความเหมาะสม
6. สิทธิพิเศษที่กำหนดให้แก่คำสั่งจำเป็นต้องถูกส่งต่อไป ยังกระบวนการชายด์หรือไม่?

   YES ระบุสิทธิพิเศษในแอ็ตทริบิวต์ **inheritprivs**
7. ชุดสิทธิที่ใช้งานของคำสั่งควรเท่ากับชุดสิทธิ สูงสุดณ เวลาที่คำสั่งถูกร้องขอหรือไม่?

   YES ระบุแฟล็ก **FSF_EPS** สำหรับแอ็ตทริบิวต์ secflags

NO อย่าระบุแอ็ตทริบิวต์ **secflags** โค้ดคำสั่ง ถูกคาดหมายว่าจะเกิดขึ้นและลดสิทธิตามที่ต้องการเมื่อแฟล็ก **FSF_EPS** ไม่ถูกระบุ

8. คำสั่งต้องรันโดยใช้ ID ผู้ใช้จริงพิเศษเป็น 0 หรือไม่?

   YES ระบุแอ็ตทริบิวต์ RUID

9. คำสั่งที่สำคัญเป็นอย่างมากและ จำเป็นต้องถูกควบคุมและบังคับจัดการการมีบุคคลหลายคน ก่อนที่จะสามารถเรียกใช้ได้หรือไม่?

   YES ระบุแอ็ตทริบิวต์ **authroles** และกำหนดค่าด้วย รายการบทบาท ผู้ใช้ของแต่ละบทบาทจะต้องได้รับการพิสูจน์ตัวตน ก่อนที่คำสั่งจะสามารถถูกเรียกใช้ได้

หลังจากตอบคำถามเหล่านี้แล้ว รันคำสั่ง **setsecattr** ด้วยพารามิเตอร์ที่เหมาะสมเพื่อเพิ่มคำสั่งในฐานข้อมูล ถ้าคำสั่งเป็นคำสั่งที่มีอยู่แล้วและเป็นคำสั่ง SUID หรือ SGID ควรมีข้อควรพิจารณาในการลบบิต **SUID** และ **SGID** จากไฟล์เพื่อให้โมเดลสิทธิต่ำสุดบังคับใช้

ฐานข้อมูลอุปกรณ์สิทธิพิเศษ:

ฐานข้อมูลอุปกรณ์สิทธิพิเศษเก็บรายการของสิทธิพิเศษที่ ได้รับอนุญาตให้อ่าน หรือเขียนลงอุปกรณ์ ฐานข้อมูลนี้มีกลไกสำหรับผู้ดูแลระบบในการควบคุมการเข้าถึงอุปกรณ์เพิ่มเติมมากกว่าที่สามารถจัดการ ผ่านการควบคุมการเข้าถึงอุปกรณ์แบบดั้งเดิม

เมื่อฐานข้อมูลนี้ถูกเก็บแบบโลคัล จะอยู่ในไฟล์ `/etc/security/privdevs` ฐานข้อมูลเก็บสิทธิพิเศษที่จำเป็นสำหรับการเข้าถึงอุปกรณ์ที่กำหนดเพื่อการดำเนินการอ่าน หรือเขียนในแอ็ตทริบิวต์ต่อไปนี้:

readprivs
แสดงรายการสิทธิพิเศษซึ่งได้รับอนุญาตให้อ่านจากอุปกรณ์

writeprivs
แสดงรายการสิทธิพิเศษซึ่งได้รับอนุญาตให้เขียนลงในอุปกรณ์

เมื่ออุปกรณ์สิทธิพิเศษถูกร้องขอเพื่อเปิดในโหมดอ่าน การเปิดจะอนุญาต ต่อเมื่อสิทธิพิเศษหนึ่งที่ระบุในแอ็ตทริบิวต์ **readprivs** นั้นมีอยู่ใน effective privilege set (EPS) สำหรับกระบวนการ ในทำนองเดียวกัน ถ้า อุปกรณ์ถูกเปิดสำหรับโหมดเขียน สิทธิพิเศษในแอ็ตทริบิวต์ **writeprivs** ต้องมีอยู่ใน EPS

กระบวนการสำหรับการเพิ่มอุปกรณ์ในฐานข้อมูลอุปกรณ์สิทธิพิเศษไม่ใช่ การดำเนินการโดยปกติทั่วไป คำสั่ง **lssecattr** และ **setsecattr** สามารถใช้ เพื่อแสดงรายการและปรับเปลี่ยนฐานข้อมูล แต่การเพิ่มและการแก้ไขรายการ ในฐานข้อมูลจำเป็นต้องมีการตรวจสอบอย่างมาก เนื่องจากสิทธิการอ่านและเขียน สำหรับอุปกรณ์ถูกควบคุมผ่านการใช้สิทธิพิเศษ การตรวจสอบคำสั่งและแอ็พพลิเคชันทั้งหมดที่จำเป็นสำหรับการเข้าถึงอุปกรณ์ต้องถูกดำเนินการ เพื่อให้มั่นใจว่ามีการระบุสิทธิพิเศษที่เหมาะสม

ฐานข้อมูลไฟล์สิทธิ์พิเศษ:

ไฟล์คอนฟิกูเรชันระบบหลายไฟล์ในระบบ UNIX ดั้งเดิม มีผู้ใช้ root เป็นเจ้าของ และไม่สามารถแก้ไขโดยตรงโดยผู้ใช้อื่น RBAC อนุญาตให้ผู้ใช้แก้ไขไฟล์คอนฟิกูเรชันเหล่านี้โดยการเรียกทำงาน บทบาท และรันคำสั่งเพื่อให้ได้สิทธิ์พิเศษที่จำเป็นสำหรับการแก้ไขไฟล์

มีไฟล์คอนฟิกูเรชัน AIX บางไฟล์ ที่ไม่มีอินเตอร์เฟสคำสั่งที่จะอนุญาตให้ทำการแก้ไขไฟล์ในกรณีเหล่านี้ จำเป็นต้องมีเครื่องมือที่อนุญาตให้ผู้ดูแลระบบ ได้รับการอนุญาตที่เหมาะสมเพื่อแก้ไข และบันทึกไฟล์ที่ผู้อื่นไม่มีการเข้าถึง ได้โดยตรง

ฐานข้อมูลไฟล์สิทธิ์พิเศษจัดให้มีวิธีใช้การอนุญาตเพื่อ พิจารณาการเข้าถึงไฟล์คอนฟิกูเรชันระบบ เมื่อฐานข้อมูลถูกเก็บแบบโลคัล จะอยู่ในไฟล์ `/etc/security/privfiles` ฐานข้อมูลนี้ แม็พไฟล์คอนฟิกูเรชันกับการอนุญาตที่จำเป็นเพื่อดูหรือแก้ไขไฟล์เหล่านี้ การเข้าถึงไฟล์คอนฟิกูเรชันถูกควบคุมในฐานข้อมูลนี้ ด้วยแอ็ตทริบิวต์ต่อไปนี้:

readauths
แสดงรายการการอนุญาตที่อนุญาตให้อ่านจากไฟล์

writeauths
แสดงรายการการอนุญาตที่อนุญาตให้เขียนลงไฟล์ (การอนุญาตเพื่ออ่าน ถูกนำใช้ในกรณีนี้ด้วย)

รายการในฐานข้อมูลไฟล์สิทธิ์พิเศษสามารถถูกแสดงด้วยคำสั่ง **lssecattr** และสามารถสร้างหรือแก้ไขด้วยคำสั่ง **setsecattr** ไฟล์ที่กำหนดในฐานข้อมูลไฟล์สิทธิ์พิเศษสามารถเข้าถึงโดยผู้ใช้ที่ได้รับอนุญาต ด้วยคำสั่ง **/usr/bin/pvi** คำสั่ง **pvi** คือเวอร์ชันสิทธิ์พิเศษและจำกัดของเอดิเตอร์ **vi** โดยยึดตามคำสั่ง **/usr/bin/tvi** คำสั่ง **pvi** กำหนดข้อควรระวังการรักษาความปลอดภัยเดียวกันทั้งหมดเหมือนคำสั่ง **tvi** (ตัวอย่างไม่มีแฟล็ก **–r** หรือ **–t**, ไม่มี shell escapes, ไม่มีแมโครที่ ผู้ใช้กำหนด) และยังบังคับใช้ข้อจำกัดต่อไปนี้:

- ระบบต้องอยู่ในโหมด RBAC แบบปรับปรุง
- เฉพาะไฟล์ที่กำหนดในฐานข้อมูลไฟล์สิทธิ์พิเศษเท่านั้นที่สามารถถูกเปิดได้
- หนึ่งไฟล์เท่านั้นที่สามารถเปิดในแต่ละครั้ง
- การเขียนในชื่อไฟล์ที่ต่างจากชื่อที่ระบุบนบรรทัดคำสั่ง ถูกปิดใช้งาน
- ไฟล์ `/etc/security/privfiles` ไม่สามารถแก้ไข ด้วยคำสั่ง **pvi**
- ความพยายามเปิดลิงก์จะล้มเหลว เฉพาะไฟล์ปกติเท่านั้นที่สามารถแก้ไขได้

การตรวจสอบการอนุญาตถูกดำเนินการก่อนการเปิดไฟล์ ถ้า การอนุญาตตรง ชุดสิทธิ์พิเศษของกระบวนการถูกกำหนดขึ้นเพื่อรวม **PV_DAC_R** หรือ **PV_DAC_W** (ขึ้นอยู่กับ ว่าไฟล์ถูกเปิดเพื่ออ่านหรือเขียน) ถ้าการอนุญาต ไม่ตรง มีแสดงข้อความแสดงความผิดพลาดและผู้ใช้ถูกปฏิเสธการเข้าถึง ไฟล์ด้วยคำสั่ง **pvi**

Kernel security tables:

ข้อมูลที่มีอยู่การอนุญาต บทบาท คำสั่ง สิทธิพิเศษ และฐานข้อมูลอุปกรณ์สิทธิพิเศษไม่ถูกนำมาใช้สำหรับการพิจารณา การรักษาความปลอดภัยจนกว่าข้อมูลจะถูกโหลดเข้าในพื้นที่ของ เคอร์เนลที่กำหนดเป็น kernel security tables (KST) ในโหมด RBAC แบบปรับปรุง การอนุญาตและการตรวจสอบสิทธิพิเศษถูกดำเนินงานใน เคอร์เนล ดังนั้นฐานข้อมูลต้องถูกส่งไปที่เคอร์เนลก่อนจึงจะ สามารถใช้ได้

KST ประกอบด้วยตารางย่อยต่อไปนี้:

- Kernel Authorization Table (KAT)
- Kernel Role Table (KRT)
- Kernel Command Table (KCT)
- Kernel Device Table (KDT)

ตารางทั้งหมดหรือตารางที่เลือกสามารถส่งจากพื้นที่ผู้ใช้ไปที่เคอร์เนล ด้วยคำสั่ง **setkst** KRT และ KCT ขึ้นกับ KAT ดังนั้นถ้า KAT ถูกเลือก ให้ถูกอัพเดต KRT และ KCT ต้องถูกอัพเดตด้วยเพื่อให้แน่ใจว่า ตารางมีข้อมูลตรงกัน วิธีการต้องการสำหรับการเพิ่มการอัพเดตใน KST คือเพื่อสร้างหรือแก้ไขฐานข้อมูลที่จำเป็นทั้งหมดที่ระดับ ผู้ใช้ (ด้วยคำสั่งเช่น **mkauth, chauth, mkrole,** and **setsecattr**) และจากนั้นใช้คำสั่ง **setkst** เพื่อส่งตารางไปยังเคอร์เนล เมื่อตารางถูกโหลดไว้ใน เคอร์เนล คำสั่ง **lskst** จะสามารถถูกใช้เพื่อแสดงข้อมูลที่มีอยู่ในแต่ละตาราง

ตารางที่กำหนดใน KST ถูกส่งเป็นแบบตารางโดยสมบูรณ์เสมอ หรืออีก นัยหนึ่ง KST ไม่อนุญาตในการแก้ไขรายการเฉพาะทีละรายการ ต้องแทนที่ทั้งตาราง ก่อนหน้าที่จะส่งตารางไปยัง เคอร์เนล คำสั่ง **setkst** จะตรวจสอบความถูกต้องของตาราง และความสัมพันธ์ระหว่างตาราง คำสั่ง **setkst** ยังถูกรวมในไฟล์ `inittab` เพื่อให้แน่ใจว่าฐานข้อมูลถูกส่งไปยัง KST โดยเริ่มแรกในกระบวนการ บูตระบบ

ถ้ามีเหตุผลใดที่ตารางไม่สามารถสร้าง หรือไม่สามารถโหลด เข้าสู่เคอร์เนล และไม่มีตารางถูกโหลดก่อนหน้านี้ ระบบ จะทำงานเสมือนไม่มีการอนุญาตหรือบทบาท คำสั่ง APIs และการเรียกใช้ระบบสำหรับการตรวจสอบการอนุญาตและบทบาทส่งค่าความล้มเหลวกลับ ในสถานการณ์นี้เนื่องจากไม่พบรายการที่ตรงกัน การดำเนินการระบบในสถานะนี้ คล้ายกับโหมด RBAC แบบเก่า ยกเว้นตรงที่ไม่มีผู้ใช้รายใด สามารถเข้าถึงส่วนของโค้ดในคำสั่งที่บังคับใช้การอนุญาต

การปิดใช้งานผู้ใช้ root:

ในโหมด RBAC แบบปรับปรุง คุณสามารถตั้งค่าระบบ เพื่อให้ผู้ใช้ root ไม่มีอำนาจพิเศษที่สัมพันธ์ และถูกปฏิบัติ โดยระบบเป็นเหมือนผู้ใช้ทั่วไป

โดยทั่วไปแล้ว ค่า ID ของผู้ใช้ root เป็น 0 ถือเสมือนเป็น ID ที่มีสิทธิพิเศษ โดยระบบปฏิบัติการ และได้รับอนุญาตให้ข้ามการตรวจสอบการรักษาความปลอดภัยที่บังคับใช้ การปิดใช้งานผู้ใช้ root อย่างมีประสิทธิภาพจะลบการตรวจสอบในระบบปฏิบัติการ ซึ่งอนุญาตให้ ID ผู้ใช้ค่า 0 ข้ามการตรวจสอบการรักษาความปลอดภัยและจำเป็น ที่กระบวนการต้องมีสิทธิพิเศษเพื่อข้ามการตรวจสอบการรักษาความปลอดภัยแทน การปิดใช้งาน ผู้ใช้ root ลดความเสียหายที่เกิดจากผู้โจมตีได้เนื่องจากไม่มี identity ผู้ใช้ที่มีอำนาจเหนือทั้งหมด ผู้ใช้เดียวอีกต่อไปบนระบบ หลังการปิดใช้งานผู้ใช้ root การดูแลระบบต้องดำเนินการโดยผู้ใช้ที่ได้รับการกำหนดให้มีบทบาท ที่มีสิทธิพิเศษ

อำนาจของ root สามารถปิดใช้งานได้ด้วยคำสั่ง **/usr/sbin/setsecconf** รันคำสั่งต่อไปนี้ จากนั้นรีบูตระบบเพื่อปิดใช้งานอำนาจของผู้ใช้ root:

```
setsecconf -o root=disable
```

หลังการรันคำสั่งนี้ บัญชีผู้ใช้ root จะไม่สามารถเข้าถึงผ่าน การล็อกอินรีโมตหรือโลคัล หรือผ่านคำสั่ง su อย่างไรก็ตาม เนื่องจากบัญชีผู้ใช้ root ยังคงเป็นเจ้าของไฟล์บนระบบไฟล์ ถ้าได้รับ บัญชีผู้ใช้ ผู้ใช้ก็จะสามารถเข้าถึงไฟล์สิทธิพิเศษได้

บนระบบที่ root ถูกปิดใช้งาน กระบวนการที่ root เป็นเจ้าของจะ ไม่ได้รับการกำหนดให้มีอำนาจพิเศษหรือสิทธิพิเศษใดๆ อีกต่อไป ควรพิจารณาเรื่องนี้ ถ้าระบบมีแอ็พพลิเคชัน setuid ที่ root เป็นเจ้าของที่ไม่ได้ถูกเพิ่ม ในฐานข้อมูลคำสั่งสิทธิพิเศษ แอ็พพลิเคชัน setuid เหล่านี้อาจจะล้มเหลว ในสภาวะแวดล้อมที่ปิดใช้งาน root เนื่องจากกระบวนการไม่สามารถดำเนินกระบวนการ สิทธิพิเศษได้ ในระบบที่ปิดใช้งาน root คำสั่งใดๆ ที่จำเป็นต้องดำเนินกระบวนการสิทธิพิเศษ ควรถูกเพิ่มในฐานข้อมูลคำสั่งสิทธิพิเศษ และกำหนด สิทธิพิเศษที่เหมาะสมให้ ดังนั้น ควรดำเนินการวิเคราะห์ระบบและแอ็พพลิเคชัน ที่ใช้บนระบบด้วยความระมัดระวังก่อนปิดใช้งาน อำนาจของผู้ใช้ root

การสนับสนุนฐานข้อมูล RBAC รีโมต:

ในสภาวะแวดล้อมเอ็นเตอร์ไพรซ์ เป็นสิ่งที่ต้องการที่จะสามารถนำใช้ และบังคับใช้นโยบายการรักษาความปลอดภัยทั่วไปของทั้งระบบในสภาวะแวดล้อม ถ้าฐานข้อมูลที่ควบคุมนโยบายถูกเก็บแยกเป็นอิสระบนแต่ละ ระบ การจัดการนโยบายการรักษาความปลอดภัยจะมีภาระที่หนักสำหรับผู้ดูแลระบบ ที่ได้รับมอบหมาย AIX มีโหมด RBAC แบบปรับปรุงที่อนุญาตให้ฐานข้อมูล RBAC ถูกเก็บใน LDAP เพื่อให้นโยบาย การรักษาความปลอดภัยสำหรับทุกระบบในสภาวะแวดล้อมสามารถจัดการแบบรวมศูนย์ได้

การสนับสนุนถูกเพิ่มใน AIX สำหรับ ฐานข้อมูลที่เกี่ยวกับ RBAC ทั้งหมดที่ถูกเก็บใน LDAP ต่อไปนี้คือ ฐานข้อมูล RBAC ที่เกี่ยวข้อง:

- ฐานข้อมูลการอนุญาต
- ฐานข้อมูลบทบาท
- ฐานข้อมูลคำสั่งสิทธิ์พิเศษ
- ฐานข้อมูลอุปกรณ์สิทธิ์พิเศษ
- ฐานข้อมูลไฟล์สิทธิ์พิเศษ

หมายเหตุ: ฐานข้อมูลการอนุญาตที่เก็บใน LDAP มีเฉพาะการอนุญาต ที่ผู้ใช้กำหนดเองเท่านั้น การอนุญาตที่ระบบกำหนดไม่สามารถเก็บใน LDAP และ ยังคงเป็นแบบโลคัลสำหรับระบบไคลเอ็นต์แต่ละระบบ

AIX จัดให้มียูทิลิตี ที่เอ็กซ์พอร์ตข้อมูล RBAC โลคัลไปยัง LDAP ได้โดยง่าย ตั้งค่าไคลเอ็นต์เพื่อใช้ ข้อมูล RBAC ใน LDAP ควบคุมการค้นหาข้อมูล RBAC และจัดการข้อมูล LDAP จากระบบไคลเอ็นต์ ส่วนต่อไปนี้มีข้อมูลเพิ่มเติม เกี่ยวกับคุณลักษณะ LDAP ที่มีใน RBAC แบบปรับปรุง

การเอ็กซ์พอร์ตข้อมูล RBAC ไปยัง LDAP:

การเตรียมการเริ่มต้นสำหรับการใช้ LDAP เป็นที่เก็บฐานข้อมูล RBAC จำเป็นต้องมีการใส่เซิร์ฟเวอร์ LDAP ด้วยข้อมูล RBAC

เซิร์ฟเวอร์ LDAP ต้องมี RBAC schema สำหรับ LDAP ติดตั้งอยู่บนเซิร์ฟเวอร์ ก่อน ที่ไคลเอ็นต์ LDAP สามารถใช้เซิร์ฟเวอร์เกี่ยวกับข้อมูล RBAC RBAC schema สำหรับ LDAP มีอยู่บนระบบ AIX ใน ไฟล์ `/etc/security/ldap/sec.ldif` schema ของเซิร์ฟเวอร์ LDAP ควรถูกอัพเดตด้วยไฟล์นี้โดยใช้คำสั่ง **ldapmodify**

ไฟล์ **/usr/sbin/rbactoldif** สามารถใช้เพื่ออ่าน ข้อมูลในฐานข้อมูล RBAC โลคัลและเอาต์พุตข้อมูลในรูปแบบที่เหมาะสำหรับ LDAP เอาต์พุตที่สร้างโดยคำสั่ง **rbactoldif** สามารถถูกบันทึกไปยัง ไฟล์ จากนั้นใช้เพื่อใส่ข้อมูลแก่เซิร์ฟเวอร์ LDAP ด้วยข้อมูลด้วยคำสั่ง **ldapadd** ฐานข้อมูลต่อไปนี้บนระบบโลคัลถูกใช้โดยคำสั่ง **rbactoldif** เพื่อสร้างข้อมูล RBAC สำหรับ LDAP:

- `/etc/security/authorizations`
- `/etc/security/privcmds`
- `/etc/security/privdevs`
- `/etc/security/privfiles`
- `/etc/security/roles`

ตำแหน่งที่เก็บ LDAP สำหรับข้อมูล RBAC ควรมีข้อควรพิจารณาบางอย่าง ขอแนะนำให้ข้อมูล RBAC ใน LDAP เก็บไว้ภายใต้ DN พาเรนต์เดียวกับข้อมูลผู้ใช้และกลุ่ม ACLs บนข้อมูลควรถูกปรับเปลี่ยน ตามความจำเป็นสำหรับนโยบายการรักษาความปลอดภัยที่เลือก

การตั้งค่าไคลเอ็นต์ LDAP สำหรับ RBAC:

ระบบต้องถูกตั้งค่าเป็นไคลเอ็นต์ LDAP เพื่อใช้ข้อมูล RBAC ที่เก็บใน LDAP

คุณสามารถใช้คำสั่ง AIX **/usr/sbin/mksecldap** เพื่อตั้งค่าระบบเป็นไคลเอ็นต์ LDAP คำสั่ง **mksecldap** จะค้นหาแบบไดนามิกเซิร์ฟเวอร์ LDAP ที่ระบุเพื่อพิจารณาตำแหน่งของ การอนุญาต บทบาท คำสั่งสิทธิพิเศษ อุปกรณ์ และข้อมูลไฟล์ และบันทึกผลลัพธ์ในไฟล์ `/etc/security/ldap/ldap.cfg`

หลังจากตั้งค่าเสร็จเรียบร้อยระบบเป็นไคลเอ็นต์ LDAP ด้วยคำสั่ง **mksecldap** ระบบต้องถูกตั้งค่าเพิ่มเพื่อเปิดใช้งาน LDAP เป็นโดเมนการค้นหา ข้อมูล RBAC ไฟล์ `/etc/nscontrol.conf` ต้องได้รับการแก้ไข เพื่อรวม LDAP ในแอ็ตทริบิวต์ **secorder** สำหรับฐานข้อมูลที่ถูกเก็บ ใน LDAP

เมื่อระบบได้ถูกตั้งค่าเป็นทั้งไคลเอ็นต์ LDAP และเป็นโดเมนการค้นหา ข้อมูล RBAC daemon ไคลเอ็นต์ **/usr/sbin/secldapclntd** จะเรียกข้อมูล RBAC จาก LDAP และส่งข้อมูลไปที่ Kernel Security Tables (KST) ด้วยคำสั่ง **setkst** คุณสามารถตั้งค่าระยะเวลาที่ daemon ใช้เพื่อเรียกข้อมูล RBAC จาก LDAP ด้วยแอ็ตทริบิวต์ **rbacinterval** ในไฟล์ `/etc/security/ldap/ldap.cfg` ค่าดีฟอลต์ของแอ็ตทริบิวต์นี้คือ 3600 ซึ่งระบุให้เรียกข้อมูล RBAC จาก LDAP และอัพเดต KST ทุกชั่วโมง KST ยังสามารถ ถูกอัพเดตด้วยตนเองเมื่อผู้ดูแลระบบรันคำสั่ง **setkst**

ไฟล์ควบคุมบริการชื่อ:

ข้อมูล RBAC สามารถอยู่อย่างแน่นอนในโลคัลไฟล์ อย่างแน่นอนใน LDAP หรือสามารถผสมรวมในโลคัลไฟล์ และ LDAP โดยการตั้งค่าฐานข้อมูลที่กำหนด ในไฟล์ควบคุมบริการชื่อ `/etc/nscontrol.conf`

ลำดับการค้นหาสำหรับฐานข้อมูลการอนุญาต บทบาท คำสั่งสิทธิพิเศษ อุปกรณ์ และไฟล์ถูกระบุแต่ละค่าในไฟล์ `/etc/nscontrol.conf` ลำดับการค้นหาสำหรับฐานข้อมูลถูกระบุในไฟล์ด้วยแอ็ตทริบิวต์ **secorder** ซึ่งเป็นรายการโดเมนคั่นด้วยคอมมา ต่อไปนี้คือตัวอย่างของ การตั้งค่าสำหรับฐานข้อมูลการอนุญาต:

```
authorizations:
        secorder = LDAP,files
```

ตัวอย่างนี้ระบุว่าเคียวรีการอนุญาตควรค้นหาใน LDAP เป็นที่แรก จากนั้นในโลคัลไฟล์ถ้าไม่พบการอนุญาตใน LDAP คอลเล็กชันของการอนุญาตที่มีในระบบคือการผสมรวม ของการอนุญาตที่จัดให้มีโดย LDAP และที่จัดให้มีในโลคัลไฟล์ การผสมรวมไม่ใช่การรวมค่าจากสองโดเมนแบบง่าย แต่เป็นการยูเนียนค่า สำหรับการตั้งค่าข้างต้น การอนุญาต LDAP ทั้งหมด ถูกรวม จากนั้นเฉพาะการอนุญาตที่ไม่ซ้ำจากโลคัลไฟล์จะถูกเพิ่ม ในผลลัพธ์

การแก้ไข และการลบถูกดำเนินการในโดเมนแรกที่แสดงรายการและ จะถูกดำเนินการบนโดเมนถัดไปต่อเมื่อไม่พบ entity ในโดเมน แรกเท่านั้น ในกรณีนี้ LDAP ถูกดำเนินการเป็นที่แรก และพยายามดำเนินการในโลคัลไฟล์ ต่อเมื่อไม่พบการอนุญาตใน LDAP รายการใหม่จะถูกสร้างขึ้น ในโดเมนแรกที่แสดงในรายการในแอ็ตทริบิวต์ **secorder** เสมอ ใน ตัวอย่างข้างต้น การสร้างการอนุญาตใหม่จะเกิดขึ้นในฐานข้อมูล LDAP

ถ้าไม่มีรายการสำหรับฐานข้อมูลในไฟล์ /etc/nscontrol.conf หรือถ้าไม่มีไฟล์อยู่ เคียวรีและการแก้ไขบนฐานข้อมูลจะถูกดำเนินการเฉพาะในฐานข้อมูลโลคัลไฟล์เท่านั้น การตั้งค่าสำหรับฐานข้อมูล ในไฟล์สามารถตั้งค่าด้วยคำสั่ง **chsec** และถูกแสดงรายการ ผ่านคำสั่ง **lssec** ในการตั้งค่าข้อมูลการอนุญาต ที่จะถูกเรียกค้นจาก LDAP เป็นอันดับแรกจากนั้นจากโลคัลไฟล์ ให้รัน คำสั่งต่อไปนี้:

```
chsec -f /etc/nscontrol.conf -s authorizations -a secorder=LDAP,files
```

การตั้งค่าในไฟล์ /etc/nscontrol.conf ควบคุมทั้งอินเตอร์เฟสไลบรารีและอินเตอร์เฟสบรรทัดคำสั่ง แอ็พพลิเคชันสามารถเรียกข้อมูล ค่าปัจจุบันของแอ็ตทริบิวต์ **secorder** สำหรับด้วยอินเตอร์เฟส **getsecorder** ค่าของแอ็ตทริบิวต์ **secorder** สามารถถูกแทนที่สำหรับกระบวนการ ด้วยอินเตอร์เฟส **setsecorder**

การเปิดใช้คำสั่ง RBAC สำหรับ LDAP:

คำสั่งการจัดการฐานข้อมูล RBAC ทั้งหมดถูกเปิดใช้งานเพื่อใช้ การตั้งค่าในไฟล์ /etc/nscontrol.conf และ เพื่อเคียวรี แก้ไข สร้าง หรือลบ entity ในโดเมน หรือหลายๆ โดเมนที่กำหนด สำหรับฐานข้อมูลที่กำหนด

โดยดีฟอลต์โดเมนถูกประมวลผลดังที่กำหนดในแอ็ตทริบิวต์ **secorder** สำหรับฐานข้อมูล แต่สามารถแทนที่ได้โดยใช้อ็อพชัน **–R** บน บรรทัดคำสั่ง การระบุอ็อพชัน **–R** สำหรับคำสั่งเป็นการบังคับให้ การเดำเนินงานให้เกิดขึ้นบนโดเมนที่ระบุ และแทนที่การตั้งค่า ในไฟล์ /etc/nscontrol.conf คำสั่งฐานข้อมูลการจัดการต่อไปนี้ จะถูกเปิดใช้งานสำหรับการสนับสนุนโดเมนแบบรีโมต:

- **mkauth, chauth, lsauth** และ **rmauth**
- **mkrole, chrole, lsrole** และ **rmrole**
- **setsecattr, lssecattr** และ **rmsecattr**

นอกจากนั้น คำสั่ง **setkst** ถูกเปิดใช้งานเพื่อใช้ คำสั่งการตั้งค่าที่มีในไฟล์ /etc/nscontrol.conf คำสั่ง **setkst** เรียกดูสำเนาที่ถูกผสานของรายการ สำหรับฐานข้อมูลที่กำหนดดังกำหนดในไฟล์ และโหลดข้อมูลผลลัพธ์ ลงใน Kernel Security Tables

การกำหนดข้ามโดเมน:

เมื่อออกแบบสภาวะแวดล้อมที่ข้อมูล RBAC มาจากสองโดเมน เช่นโลคัลไฟล์ และ LDAP ข้อควรพิจารณาต้องมีให้สำหรับปัญหา ของการกำหนดข้ามโดเมนของ entities ตัวอย่างของการกำหนดข้ามโดเมนประกอบด้วย การกำหนดบทบาทที่กำหนด LDAP ไปยังผู้ใช้โลคัล หรือการกำหนดบทบาทที่โลคัล ไปยังผู้ใช้ LDAP

การกำหนด entity รีโมต (บทบาท LDAP) ไปยัง entity โลคัล (ผู้ใช้โลคัล) ไม่เกี่ยวพันมาก เนื่องจากไม่มีผลบนระบบอื่นในสภาวะแวดล้อม อย่างไรก็ตาม การกำหนด entity โลคัล (บทบาทโลคัล) ไปยัง entity รีโมต (ผู้ใช้ LDAP) ควรดำเนินการด้วยความระมัดระวังอย่างยิ่งเท่านั้น เนื่องจาก entity รีโมต (ผู้ใช้ LDAP) เห็นได้บนหลายไคลเอ็นต์จึงไม่มีการรับประกันว่า entity โลคัล (บทบาทโลคัล) ถูกกำหนดตามที่กำหนดไว้ หรือมีนิยามเหมือนกับ บนแต่ละระบบไคลเอ็นต์ ตัวอย่าง บทบาทอาจถูกกำหนดแบบโลคัลบนแต่ละ ไคลเอ็นต์ แต่มีการอนุญาตที่เชื่อมโยงต่างกัน ผู้ใช้รีโมตที่ ถูกกำหนดบทบาทโลคัล จะมีการอนุญาตที่แตกต่างกันบน แต่ละไคลเอ็นต์ และการทำนี้สามารถก่อให้เกิดผลตามที่ด้านความปลอดภัยที่ไม่น่าพอใจ

ในการป้องกันปัญหาความปลอดภัยที่อาจเกิดขึ้นของการกำหนด entity โลคัลไปยัง LDAP entity ขอแนะนำให้เซิร์ฟเวอร์ LDAP นำการควบคุมการเข้าถึง ฐานข้อมูล RBAC เพื่อป้องกันมิให้แต่ละไคลเอ็นต์แก้ไขรายการ เฉพาะไคลเอ็นต์ ที่เชื่อมต่อกับเซิร์ฟเวอร์ LDAP ผ่านบัญชีผู้ใช้ที่มีสิทธิพิเศษเท่านั้นที่ควรได้รับอนุญาต ให้แก้ไข LDAP RBAC entities ไคลเอ็นต์อื่นๆ ควรมีสิทธิการอ่าน ฐานข้อมูล LDAP RBAC เท่านั้น

การจำกัดขนาดใน RBAC แบบปรับปรุง:

ตารางต่อไปนี้แสดงขนาดที่แตกต่างกันสำหรับองค์ประกอบที่เกี่ยวกับ RBAC:

ตารางที่ 10. ข้อจำกัดต่างๆ สำหรับองค์ประกอบที่เกี่ยวกับ RBAC

คำอธิบาย	ขนาดสูงสุด
ชื่อบทบาท	63 อักขระที่พิมพ์ได้
บทบาทสูงสุดต่อหนึ่งเซสชัน	8
ขนาดชื่อการอนุญาตสูงสุด	63 อักขระที่พิมพ์ได้
จำนวนระดับสูงสุดในลำดับชั้นการอนุญาต	9
จำนวนสูงสุดของการอนุญาตการเข้าถึงต่อคำสั่ง	8
ชุดสิทธิพิเศษที่อนุญาตสูงสุดต่อคำสั่ง	8

การดูแล RBAC ที่ปรับปรุง:

ส่วนนี้อธิบายสถานการณ์การใช้งานบรรทัดคำสั่งทั่วไป สำหรับการดูแล RBAC ตัวอย่างเหล่านี้แสดงลักษณะการทำงานที่สำคัญ อินเตอร์เฟส SMIT ยังถูกจัดให้มีสำหรับการดูแล RBAC fastpath ไปยัง เมนู RBAC SMIT คือ `smit rbac`

การสร้างการอนุญาตที่ผู้ใช้กำหนดเอง:

คุณสามารถสร้างการอนุญาตที่ผู้ใช้กำหนดเองที่สามารถใช้ ควบคุมการทำงานของคำสั่ง

คุณสามารถใช้คำสั่ง mkauth เพื่อสร้างการอนุญาตที่ผู้ใช้กำหนดเอง การเปลี่ยนแปลงฐานข้อมูลการอนุญาตจะมีผลหลังการเปลี่ยนแปลง ถูกดาวน์โหลดไปที่เคอร์เนลด้วยคำสั่ง **setkst**

- รันคำสั่งต่อไปนี้เพื่อสร้างการอนุญาตที่ผู้ใช้กำหนดเอง:

  mkauth *auth_name*

การสร้างและการแก้ไขบทบาท:

คุณสามารถสร้างบทบาทด้วยคำสั่ง **mkrole**

บทบาทถูกสร้างด้วยคำสั่ง **mkrole** การเปลี่ยนแปลง ในฐานข้อมูลบทบาทจะมีผลหลังจากถูกดาวน์โหลดไปยังเคอร์เนล ด้วยคำสั่ง **setkst** คุณสามารถแก้ไขบทบาทด้วยคำสั่ง **chrole**

- รันคำสั่งต่อไปนี้เพื่อสร้างบทบาท:

  mkrole dflt_msg="*My Role*" *role_name*
- ในการสร้างบทบาทและสืบทอดการอนุญาตจากบทบาทที่มีอยู่ ให้รัน คำสั่งต่อไปนี้:

  mkrole rolelist=*child_role1*,*child_role2 role_name*
- ในการแก้ไขข้อกำหนดบทบาท ให้รันคำสั่งต่อไปนี้:

  chrole rolelist=*child_role3 role_name*

การกำหนดการอนุญาตให้แก่บทบาท:

คุณสามารถใช้คำสั่ง **mkrole** หรือ **chrole** เพื่อกำหนดการอนุญาตให้แก่บทบาท

- รันคำสั่ง **mkrole** เพื่อกำหนดการอนุญาต **auth_name1** และ **auth_name2** ให้แก่บทบาท **role_name**:

  ```
  mkrole authorizations=auth_name1,auth_name2 role_name
  ```

- รันคำสั่ง **chrole** เพื่อกำหนดการอนุญาต **auth_name1** และ **auth_name2** ให้แก่บทบาท **role_name** :

  ```
  chrole authorizations=auth_name1,auth_name2 role_name
  ```

การตั้งค่าโหมดการพิสูจน์ตัวตนสำหรับบทบาท:

คุณสามารถควบคุมการเปิดใช้งานบทบาทด้วยแอ็ตทริบิวต์ **auth_mode** ของบทบาท

ค่าที่ใช้ได้สำหรับแอ็ตทริบิวต์ **auth_mode** คือ:

NONE ไม่จำเป็นต้องมีการพิสูจน์ตัวตน

INVOKER
ผู้ร้องขอต้องป้อนรหัสผ่านของตน นี้เป็นค่าดีฟอลต์

ป้อนคำสั่งต่อไปนี้เพื่อบังคับให้ผู้ใช้พิสูจน์ตัวตนเป็นตนเอง เมื่อสมมติบทบาทที่กำหนด:

chrole auth_mod=INVOKER *role_name*

การกำหนดบทบาทให้แก่ผู้ใช้:

คุณสามารถใช้คำสั่ง **chuser** เพื่อกำหนดบทบาท ให้แก่ผู้ใช้

รันคำสั่งต่อไปนี้เพื่อกำหนดบทบาท **role_name1** และ **role_name2** ให้แก่ผู้ใช้ **user_name**:

```
chuser roles=role_name1,role_name2 user_name
```

การเรียกทำงานบทบาท:

โดยดีฟอลต์ ผู้ใช้ต้องเรียกทำงานบทบาทในเซสชัน เพื่อทำงานคำสั่งสิทธิพิเศษ

- ในการเรียกทำงานบทบาท **role_name1** และ **role_name2** ให้รัน คำสั่งต่อไปนี้:

  ```
  swrole role_name1,role_name2
  ```

- บางบทบาทที่กำหนดให้แก่ผู้ใช้ถูกจัดประเภทเป็นบทบาท ดีฟอลต์ บทบาทเหล่านี้ถูกเรียกทำงานโดยอัตโนมัติเมื่อผู้ใช้ล็อกอิน บทบาท เหล่านี้แอ็คทีฟอยู่ตลอดทั้งเซสชันการล็อกอิน ในการกำหนด role_name1 เป็น บทบาทดีฟอลต์สำหรับผู้ใช้ รันคำสั่งต่อไปนี้:

  ```
  chuser roles=role_name1,role_name2 default_roles=role_name1 user_name
  ```

การแสดงรายการชุดบทบาทที่แอ็คทีฟ:

คุณสามารถใช้คำสั่ง **rolelist** ที่มีอ็อพชัน **-e** เพื่อแสดงข้อมูลเกี่ยวกับชุดบทบาทที่แอ็คทีฟสำหรับหนึ่งเซสชัน

- ในการแสดงชุดบทบาทที่แอ็คทีฟที่ใช้อยู่สำหรับหนึ่งเซสชัน รันคำสั่ง ต่อไปนี้:

  ```
  rolelist -e
  ```

การแสดงรายการบทบาทสำหรับผู้ใช้:

คำสั่ง **rolelist** ให้ข้อมูลบทบาทและการอนุญาต เกี่ยวกับบทบาทปัจจุบันของผู้ใช้ หรือบทบาทที่ถูกกำหนดให้แก่ ผู้ใช้

โดยดีฟอลต์ คำสั่ง **rolelist** แสดง รายการบทบาทที่ถูกกำหนดให้แก่ผู้ใช้โดยพื้นฐานแล้วการแสดงนี้ให้ข้อมูล เหมือนที่แสดงโดยคำสั่ง `lsuser -a roles user1` ยกเว้นว่าคำสั่งนี้รวมคำอธิบายข้อความของบทบาทถ้ามีบทบาท ให้

- ในการแสดงรายการบทบาทที่กำหนดของคุณ และการอนุญาตที่สัมพันธ์ รันคำสั่ง ต่อไปนี้:

  ```
  rolelist -a
  ```

การตรวจสอบบทบาทเซสชัน:

บทบาทที่แอ็คทีฟในเซสชันล็อกอินถูกตรวจสอบพร้อมกับ แอ็ตทริบิวต์อื่นๆ เช่น UID และ GID คุณสามารถแสดงรายการบทบาทเหล่านี้ด้วย **auditpr**

ในการแสดงบทบาทจากการติดตามการตรวจสอบ รันคำสั่งต่อไปนี้:

```
auditpr -h eli -i /audit/trail
```

การกำหนดสิทธิพิเศษให้แก่กระบวนการที่กำลังทำงาน:

คุณสามารถใช้คำสั่ง **setsecattr** เพื่อแก้ไข สิทธิพิเศษของกระบวนการที่กำลังทำงาน

- ในการอัพเดตชุดสิทธิพิเศษที่ใช้งานที่สัมพันธ์กับกระบวนการ ให้รัน คำสั่งต่อไปนี้:

  setsecattr −p eprivs=*privileges pid*
- ก่อนเพิ่มสิทธิพิเศษใดๆ ในชุดสิทธิพิเศษที่ใช้งานของกระบวนการ คุณควรตรวจสอบให้แน่ใจว่ามีสิทธิพิเศษอยู่แล้วในชุดสิทธิพิเศษ สูงสุด ในการแก้ไขชุดสิทธิพิเศษสูงสุด รันคำสั่งต่อไปนี้:

  setsecattr −p mprivs=*privileges pid*

การดูแลสิทธิพิเศษ WPAR:

แต่ละ WPAR เชื่อมโยงกับชุดของสิทธิพิเศษ ที่ใช้พิจารณาขอบเขตการทำงาน นี้ถูกอ้างถึงเป็น WPAR privilege set (WPS)

กระบวนการที่กำลังทำงานภายใน WPAR ที่กำหนดสามารถใช้เฉพาะสิทธิพิเศษ ที่มีอยู่ใน WPS เท่านั้น

- ในการแก้ไข WPS จาก WPAR โกลบอล รันคำสั่งต่อไปนี้:

  chwpar −S privs+=*privileges wpar_name*

การพิจารณาสิทธิพิเศษที่จำเป็นสำหรับคำสั่ง:

คำสั่งบางคำสั่งจำเป็นต้องมีสิทธิพิเศษในการดำเนินการ สิทธิพิเศษ สิทธิพิเศษถูกใช้ในเคอร์เนลเพื่อข้ามข้อจำกัดการรักษาความปลอดภัย

คุณสามารถใช้คำสั่ง **tracepriv** เพื่อโปรไฟล์ คำสั่งเพื่อใช้พิจารณาสิทธิพิเศษที่จำเป็นสำหรับคำสั่งเพื่อให้รัน ได้สำเร็จ คำสั่ง **tracepriv** บันทึกสิทธิพิเศษ ที่ใช้โดยคำสั่งเมื่อคำสั่งถูกรัน คำสั่งควร ถูกรันด้วยสิทธิพิเศษ **PV_ROOT** เพื่อให้การพยายามใช้สิทธิพิเศษ จะสามารถทำได้สำเร็จ เมื่อคำสั่งเสร็จเรียบร้อย ชุดของสิทธิพิเศษที่ถูกใช้งาน ถูกส่งไปยัง `stdout`

- ในการโปรไฟล์คำสั่งที่กำหนด รันคำสั่งต่อไปนี้:

  tracepriv −ef *command_name*

การใช้การอนุญาตเพื่อควบคุมคำสั่ง:

การอนุญาตสามารถใช้เพื่อควบคุมการรันคำสั่ง

คุณสามารถใช้คำสั่ง **setsecattr** เพื่อเชื่อมโยง การอนุญาตกับคำสั่ง คำสั่ง **setsecattr** เพิ่ม stanza ในฐานข้อมูลคำสั่งสิทธิ์พิเศษ (`/etc/security/privcmds`) การแก้ไขในฐานข้อมูลนี้ต้องถูกดาวน์โหลดไปยังเคอร์เนลด้วยคำสั่ง **setkst**

- ในการเชื่อมโยงการอนุญาตด้วยคำสั่ง ให้รันคำสั่งต่อไปนี้:

  setsecattr -c accessauths=*auth_names* **innateprivs=***privileges* **proxyprivs=***privileges*
  authprivs=*auth_name=privileges command_name*

การควบคุมการเข้าถึงอุปกรณ์:

RBAC จัดให้มีกลไกการควบคุมต่างๆ เพิ่มเติมเพื่อควบคุมการเข้าถึงอุปกรณ์ ผู้ดูแลระบบสามารถระบุสิทธิพิเศษที่จำเป็นสำหรับการเปิดใช้ อุปกรณ์ในโหมดอ่านหรือโหมดเขียน

ตัวอย่าง การเขียนเพื่อเข้าถึง DVD ไร์เตอร์สามารถควบคุมได้โดย สิทธิพิเศษ **PV_DEV_CONFIG** เพื่อที่กระบวนการที่มีเท่านั้น ที่มีสิทธิพิเศษ หรือ DVDs

- ในการเพิ่มอุปกรณ์ลงในฐานข้อมูลอุปกรณ์ รันคำสั่งต่อไปนี้:

  setsecattr -d readprivs=*privileges* **writeprivs=***privileges device_name*

การอัพเดต RBAC Kernel Security Tables:

คำสั่ง **setkst** อ่านฐานข้อมูลการรักษาความปลอดภัย และโหลดข้อมูลจากฐานข้อมูลมาไว้ใน Kernel Security Tables (KST)

โดยดีฟอลต์ ฐานข้อมูลการรักษาความปลอดภัยทั้งหมดถูกส่งไปที่ KST อีกทางหนึ่ง ฐานข้อมูลที่เจาะจงสามารถถูกระบุด้วยอ็อพชัน **-t** อย่างไรก็ตาม การระบุว่าฐานข้อมูลการอนุญาตเท่านั้นที่ควรถูกส่งไปยัง the KST รวมถึงอัพเดตฐานข้อมูลบทบาทและคำสั่งสิทธิ์พิเศษใน KST เนื่องจากฐานข้อมูลบทบาทและคำสั่งสิทธิ์พิเศษขึ้นอยู่กับฐานข้อมูล การอนุญาต

- ในการส่งฐานข้อมูล RBAC ล่าสุดไปยังเคอร์เนล รันคำสั่ง ต่อไปนี้:

  ```
  setkst
  ```

การใช้การสลับโหมด RBAC แบบปรับปรุง:

การสลับการตั้งค่าทั้งระบบมีขึ้นเพื่อปิดใช้งานความสามารถ RBAC แบบปรับปรุง และกลับไปทำงาน RBAC แบบเก่า

ผู้ดูแลระบบสามารถปิดใช้งานโหมด RBAC แบบปรับปรุง ได้โดยการรันคำสั่ง **chdev** บนอุปกรณ์ **sys0** และการระบุแอ็ตทริบิวต์ **enhanced_RBAC** ด้วยค่า **false** และ จากนั้นรีบูตระบบ โหมดสามารถสลับกลับไปเป็นโหมด RBAC แบบปรับปรุง ได้โดยการตั้งค่าแอ็ตทริบิวต์ **enhanced_RBAC** เป็น **true** จากนั้น รีบูตระบบ

- ในการกลับไปเป็นโหมด RBAC เก่า รันคำสั่งต่อไปนี้:

  ```
  chdev -l sys0 -a enhanced_RBAC=false
  ```

- ในการแสดงรายการค่าของแอ็ตทริบิวต์ **enhanced_RBAC** รันคำสั่ง ต่อไปนี้:

  ```
  lsattr -E -l sys0 -a enhanced_RBAC
  ```

ในสภาวะแวดล้อม WPAR โหมด RBAC สามารถถูกตั้งค่าจาก ระบบโกลบอลเท่านั้น และมีผลต่อโกลบอลทั้ง WPARs

หมายเหตุ: การปิดใช้งานโหมด RBAC แบบปรับปรุงอาจลดขีดจำกัดการรักษาความปลอดภัยของระบบคุณให้ต่ำลง โดยเฉพาะอย่างยิ่ง ใน WPAR

คำสั่งที่เกี่ยวข้องกับ RBAC

ตารางต่อไปนี้แสดงรายการคำสั่งที่เกี่ยวข้องกับ RBAC ที่ มีให้ในระบบปฏิบัติการ AIX เพื่อจัดการและใช้งานเฟรมเวิร์ก RBAC

คำสั่ง	คำอธิบาย
chauth	แก้ไขแอ็ตทริบิวต์การอนุญาตที่ผู้ใช้กำหนดเอง
chrole	แก้ไขแอ็ตทริบิวต์บทบาท
ckauth	ตรวจสอบการอนุญาตของกระบวนการปัจจุบัน
lsauth	แสดงแอ็ตทริบิวต์การอนุญาตที่ผู้ใช้และระบบกำหนด
lskst	แสดงรายการใน Kernel Security Tables
lspriv	แสดงสิทธิ์พิเศษที่มีในระบบ
lsrole	แสดงแอ็ตทริบิวต์บทบาท
lssecattr	แสดงแอ็ตทริบิวต์การรักษาความปลอดภัยของคำสั่ง อุปกรณ์ กระบวนการ หรือไฟล์
mkauth	สร้างการอนุญาตที่ผู้ใช้กำหนดเองใหม่
mkrole	สร้างบทบาทใหม่
pvi	เอดิเตอร์ไฟล์สิทธิ์พิเศษ
rbacqry	เปิดใช้งาน RBAC สำหรับแอ็พพลิเคชัน
rbactoldif	เอาต์พุตฐานข้อมูลระดับผู้ใช้ RBAC ในรูปแบบที่เข้ากันได้กับ LDAP
rmauth	ลบการอนุญาตที่ผู้ใช้กำหนดเอง
rmrole	ลบบทบาท
rmsecattr	ลบนิยามของแอ็ตทริบิวต์การรักษาความปลอดภัยสำหรับคำสั่ง อุปกรณ์ หรือไฟล์
rolelist	แสดงข้อมูลบทบาทสำหรับผู้ใช้หรือกระบวนการ
setkst	ส่งรายการในฐานข้อมูลระดับผู้ใช้ RBAC ไปยัง Kernel Security Tables
setsecattr	ตั้งค่าแอ็ตทริบิวต์การรักษาความปลอดภัยของคำสั่ง อุปกรณ์ กระบวนการ หรือไฟล์
setsecconf	แก้ไขแฟล็กการรักษาความปลอดภัยเคอร์เนล
swrole	สร้างเซสชันบทบาทใหม่
tracepriv	ติดตามสิทธิ์พิเศษที่คำสั่งต้องการเพื่อให้รันได้สำเร็จ

ไฟล์ที่เกี่ยวกับ RBAC

ตารางต่อไปนี้แสดงไฟล์ที่เกี่ยวกับ RBAC ที่มีใน AIX เพื่อตั้งค่าและเก็บข้อมูล ฐานข้อมูล

ไฟล์	คำอธิบาย
/etc/nscontrol.conf	ไฟล์ควบคุมบริการชื่อสำหรับฐานข้อมูลการรักษาความปลอดภัยที่กำหนด
/etc/security/authorizations	ฐานข้อมูลการอนุญาตที่ผู้ใช้กำหนดเอง
/etc/security/privcmds	ฐานข้อมูลคำสั่งสิทธิ์พิเศษ
/etc/security/privfiles	ฐานข้อมูลไฟล์สิทธิ์พิเศษ
/etc/security/privdevs	ฐานข้อมูลอุปกรณ์สิทธิ์พิเศษ
/etc/security/roles	ฐานข้อมูลบทบาท

การใช้ RBAC แบบปรับปรุงในแอ็พพลิเคชัน

หลายๆ แอ็พพลิเคชันไม่ต้องการการแก้ไขใดๆ เพื่อให้รัน ในสภาวะแวดล้อม RBAC แบบปรับปรุงได้สำเร็จ เพียงการกำหนดการอนุญาตการเข้าถึง ของแอ็พพลิเคชันและสิทธิ์พิเศษที่เชื่อมโยง จากนั้นกำหนดแอ็พพลิเคชัน ไปยังฐานข้อมูลคำสั่งสิทธิ์พิเศษก็เป็นการเพียงพอ

อย่างไรก็ตาม แอ็พพลิเคชันสามารถใช้ RBAC แบบปรับปรุงได้โดยการเรียกใช้อินเตอร์เฟส RBAC เพื่อควบคุมการทำงานของแอ็พพลิเคชันในระดับกลุ่มย่อย และส่งผลให้ แอ็พพลิเคชันมีความปลอดภัยมากขึ้น แอ็พพลิเคชันที่อาจได้ประโยชน์จากการรวม กับ RBAC แบบปรับปรุงมีต่อไปนี้:

- แอ็พพลิเคชันที่จำกัดการใช้เฉพาะผู้ใช้ root หรือสมาชิกของกลุ่ม ที่เจาะจง โดยทั่วไปแอ็พพลิเคชันเหล่านี้ตรวจสอบ identity ผู้ใช้ที่ใช้งาน หรือกลุ่มของกลุ่มและสามารถถูกแก้ไขเพื่อตรวจสอบการอนุญาตแทน
- แอ็พพลิเคชันที่ใช้ประโยชน์บิตโหมด **setuid** หรือor **setgid** เพื่ออนุญาตที่ไม่มีสิทธิ์พิเศษให้ได้รับสิทธิ์พิเศษระหว่างการเรียกใช้คำสั่ง แอ็พพลิเคชันเหล่านี้ส่วนใหญ่มีความปลอดภัยมากขึ้นโดยใช้ privilege bracketing ทำให้มีการใช้สิทธิ์พิเศษน้อยลงในการทำงานให้สำเร็จ

การตรวจสอบการอนุญาต:

แอ็พพลิเคชันที่ขณะนี้ใช้ ID ผู้ใช้ หรือ ID กลุ่มของ ผู้ใช้ที่ร้องขอเพื่อพิจารณาความสามารถในการดำเนินงานสิทธิพิเศษ ควรถูกแก้ไขเพื่อตรวจสอบการอนุญาตแทน

ตัวอย่าง พิจารณาแอ็พพลิเคชันซึ่งดำเนินงานการตั้งค่าระบบไฟล์ และขณะนี้อนุญาตให้ผู้ใช้ root (UID = 0) กระทำการดำเนินการ สิทธิพิเศษบางอย่าง:

```
if (getuid() == 0) {
   /* allow privileged operation to continue */
}
```

ในการเปิดใช้งานแอ็พพลิเคชันนี้เพื่ออนุญาตผู้ใช้ที่มีการอนุญาตที่ระบุ (**aix.fs.config**) ให้กระทำการดำเนินการสิทธิพิเศษแทน โค้ดสามารถ ถูกเปลี่ยนเพื่อใช้ **checkauths** API ในการดำเนินงานการอนุญาต:

```
if (checkauths("aix.fs.config", CHECK_ALL)) {
  /* allow privileged operation to continue */
 }
```

checkauths API ถูกเปิดใช้งานสำหรับโหมด RBAC ทั้งแบบเก่าและแบบปรับปรุง และจะส่งค่าโค้ดแสดงการสำเร็จ **0** กลับถ้ากระบวนการการร้องขอ มีการอนุญาตที่ระบุ **checkauths** API ยังพิจารณาถ้า อำนาจผู้ใช้ root ถูกเปิดใช้งานหรือปิดใช้งาน จากนั้นอนุญาตหรือไม่อนุญาต ให้ผู้ใช้ root ข้ามการตรวจสอบการอนุญาตตามความเหมาะสม ก่อนหน้า AIX เวอร์ชัน 6.1 โดยปกติ **MatchAllAuths, MatchAnyAuths, MatchAllAuthsList** และ **MatchAnyAuthsList** APIs ถูกใช้เพื่อดำเนินการตรวจสอบ การอนุญาต แอ็พพลิเคชันที่มีใน AIX เวอร์ชัน 6.1 และ ใหม่กว่าควรใช้ **checkauths** API แทนเนื่องจากสนับสนุน ทั้งโหมด RBAC แบบเก่าและแบบปรับปรุง และการปิดใช้งาน root

ดังในตัวอย่างข้างต้น แอ็พพลิเคชันที่เรียกใช้ **getuid, getgid** หรือฟังก์ชันที่คล้ายคลึงเพื่ออนุญาตเฉพาะผู้ใช้ที่กำหนดให้ดำเนินงาน ที่ระบุเท่านั้นที่สามารถถูกแก้ไขเพื่อใช้ **checkauths** API ในการดำเนินการ ตรวจสอบการอนุญาตแทนได้ ถ้า ID ผู้ใช้หรือ ID กลุ่มที่ถูกตรวจสอบ ไม่ใช่ของผู้ใช้ root การเรียกใช้ระบบ **sys_parm** สามารถใช้เป็นอย่างแรก เพื่อเคียวรีว่า RBAC แบบปรับปรุงถูกเปิดใช้งานหรือไม่ ถ้า RBAC แบบปรับปรุงไม่ ถูกเปิดใช้งาน โค้ดสามารถดำเนินการตรวจสอบว่ามีอยู่แล้วหรือไม่ มิฉะนั้น ถ้า RBAC แบบปรับปรุงถูกเปิดใช้งาน โค้ดสามารถตรวจสอบระบบที่เกี่ยวข้องกันหรือ การอนุญาตที่ผู้ใช้กำหนดเอง

Privilege bracketing:

เมื่อแอ็พพลิเคชันถูกแก้ไขเพื่อตรวจสอบการอนุญาต แอ็พพลิเคชันสามารถถูกแก้ไขเพิ่มเติมเพื่อใช้ประโยชน์ privilege bracketing ได้สูงสุด ระหว่างการดำเนินการ

แอ็พพลิเคชันสามารถใช้ **priv_raise** API เพื่อเพิ่มสิทธิพิเศษ ที่จำเป็นในการดำเนินการ และลดสิทธิพิเศษด้วย **priv_lower** API การเพิ่มสิทธิพิเศษในทันทีก่อนที่จะมีการพยายามกระทำการดำเนินการสิทธิพิเศษ และการลดสิทธิพิเศษหลังการดำเนินการเสร็จเรียบร้อยเรียกว่า privileged bracketing และเป็นวิธีที่นิยมใช้สำหรับแอ็พพลิเคชันในการใช้สิทธิพิเศษ ในการเพิ่มสิทธิพิเศษ จำเป็นต้องสิทธิพิเศษอยู่ในชุดสิทธิพิเศษสูงสุด ของแอ็พพลิเคชันในฐานข้อมูลคำสั่งสิทธิพิเศษ การเพิ่มสิทธิพิเศษทำให้สิทธิพิเศษอยู่ใน effective privilege set (EPS) ของ กระบวนการ การลดสิทธิพิเศษจะลบสิทธิพิเศษออกจาก EPS ตัวอย่าง โค้ดต่อไปนี้แสดง privilege bracketing ของ **auditproc** API

```
priv_raise(PV_AU_ADMIN, -1); /* raise privilege when needed */
auditproc();     /* call auditing system call */
priv_lower(PV_AU_ADMIN, -1); /* lower privilege */
```

แอ็พพลิเคชันที่รู้จัก RBAC:

โดยปกติ ใน AIX และ บนระบบ RBAC แบบปรับปรุงที่เปิดใช้งาน root โปรแกรม **setuid** root หรือ root เป็นเจ้าของ (ที่มี UID=0) ที่ไม่แสดงในฐานข้อมูลคำสั่งสิทธิพิเศษได้รับอนุญาต ให้มีสิทธิพิเศษทั้งหมดในเคอร์เนล การตรวจสอบสิทธิพิเศษในเคอร์เนล จะ ส่งค่าความสำเร็จกลับเสมอแม้เมื่อสิทธิพิเศษที่ร้องขอไม่มีแสดง อยู่ใน effective privilege set (EPS) ของกระบวนการ

ลักษณะการทำงานนี้ยังคงต้องมีเพื่อสนับสนุนแอ็พพลิเคชัน **setuid** ที่มีอยู่ แต่ก็อาจมีความเสี่ยงด้านความปลอดภัยได้เนื่องจากโปรแกรม **setuid** จะมี อำนาจทั้งหมดของ root

ในการอนุญาตการควบคุมสิทธิพิเศษที่เหมาะสมในกระบวนการคือกระบวนการบนระบบ RBAC แบบปรับปรุงที่เปิดใช้งาน root บิตใหม่ในโครงสร้างกระบวนการได้รับการแนะนำ ถ้าบิตนี้ ถูกตั้งค่า กระบวนการจะเปลี่ยนกระบวนการที่รู้จัก RBAC และ

UID ที่ใช้งานมีค่า 0 ไม่มีสิทธิพิเศษใดๆ ให้ บิตนี้สามารถตั้งค่าในโปรแกรม ด้วยการเรียกใช้ระบบ **proc_rbac_op** โปรแกรม **setuid** ใดๆ ที่ ไม่แสดงในฐานข้อมูลคำสั่งสิทธิพิเศษสามารถใช้ฟังก์ชันการทำงานนี้ เพื่อลดช่องโหว่ด้านความปลอดภัยโดยการลดระดับสิทธิพิเศษที่มี โปรดทราบว่า โปรแกรมที่ถูกกำหนดในฐานข้อมูลคำสั่งสิทธิพิเศษถูกทำเครื่องหมายเป็น กระบวนการที่รู้จัก RBAC โดยอัตโนมัติ และถูกกำหนดสิทธิพิเศษที่แสดงรายการ ในฐานข้อมูลเท่านั้น

โค้ดต่อไปนี้แสดงวิธีที่แอ็พพลิเคชันสามารถทำเครื่องหมายตนเองเป็นที่รู้จัก RBAC ได้ จากนั้นดำเนินการใส่สิทธิพิเศษอย่างเหมาะสม:

```
#include <userpriv.h
#include <sys/priv.h>

privg_t effpriv;

int rbac_flags = SEC_RBACAWARE;

/* Mark the process as RBAC-aware. */
proc_rbac_op(-1, PROC_RBAC_SET, &rbac_flags);

/* Set the effective privilege set as empty. */
priv_clrall(effpriv);
setppriv(-1, &effpriv, NULL, NULL, NULL);

/* Raise privilege when required. */
priv_raise(PV_AU_ADMIN, -1);
auditproc();

/* Lower privilege when no longer needed. */
priv_lower(PV_AU_ADMIN, -1);
```

RBAC APIs:

APIs ที่เกี่ยวกับ RBAC ที่มีอยู่บนระบบจะถูกแสดงใน ตารางต่อไปนี้ โปรดดูที่ APIs เฉพาะสำหรับข้อมูลเพิ่มเติม

API	**คำอธิบาย**
checkauths	เปรียบเทียบรายการการอนุญาตที่ผ่านไป กับการอนุญาต ที่สัมพันธ์กับกระบวนการปัจจุบัน
GetUserAuths	เรียกข้อมูลชุดการอนุญาตที่กำหนดให้แก่กระบวนการปัจจุบัน
MatchAllAuths, MatchAllAuthsList, MatchAnyAuths, MatchAnyAuthsList	เปรียบเทียบการอนุญาต checkauths API ใช้มากกว่า APIs เหล่านี้
getauthattr, putauthattr	เคียวรีหรือแก้ไขการอนุญาตที่กำหนดในฐานข้อมูลการอนุญาต
getauthattrs	เรียกข้อมูลแอ็ตทริบิวต์การอนุญาตหลายแอ็ตทริบิวต์จากฐานข้อมูลการอนุญาต
putauthattrs	อัพเดตแอ็ตทริบิวต์การอนุญาตหลายแอ็ตทริบิวต์ในฐานข้อมูลแอ็ตทริบิวต์
getcmdattr, putcmdattr	เคียวรีหรือแก้ไขข้อมูลความปลอดภัยคำสั่งในฐานข้อมูลคำสั่ง ที่มีสิทธิพิเศษ

API	คำอธิบาย
getcmdattrs	เรียกข้อมูลแอ็ตทริบิวต์คำสั่งหลายแอ็ตทริบิวต์จากฐานข้อมูลคำสั่งที่มีสิทธิพิเศษ
putcmdattrs	อัพเดตแอ็ตทริบิวต์คำสั่งหลายแอ็ตทริบิวต์ในฐานข้อมูลคำสั่งที่มีสิทธิพิเศษ
getdevattr, putdevattr	เคียวรีหรือแก้ไขข้อมูลความปลอดภัยอุปกรณ์ในฐานข้อมูลอุปกรณ์ที่มีสิทธิพิเศษ
getdevattrs	เรียกข้อมูลแอ็ตทริบิวต์อุปกรณ์หลายแอ็ตทริบิวต์จากฐานข้อมูลอุปกรณ์ที่มีสิทธิพิเศษ
putdevattrs	อัพเดตแอ็ตทริบิวต์อุปกรณ์หลายแอ็ตทริบิวต์ในฐานข้อมูลอุปกรณ์ที่มีสิทธิพิเศษ
getpfileattr, putpfileattr	เคียวรีหรือแก้ไขข้อมูลความปลอดภัยในฐานข้อมูลไฟล์ ที่มีสิทธิพิเศษ
getpfileattrs	เรียกข้อมูลแอ็ตทริบิวต์ไฟล์หลายแอ็ตทริบิวต์จากฐานข้อมูลไฟล์ที่มีสิทธิพิเศษ
putpfileattrs	อัพเดตแอ็ตทริบิวต์ไฟล์หลายแอ็ตทริบิวต์ในฐานข้อมูลไฟล์ที่มีสิทธิพิเศษ
getroleattr, putroleattr	เคียวรีหรือแก้ไขบทบาทที่กำหนดในฐานข้อมูลบทบาท
getroleattrs	เรียกข้อมูลแอ็ตทริบิวต์บทบาทหลายแอ็ตทริบิวต์จากฐานข้อมูลบทบาท
putroleattrs	อัพเดตแอ็ตทริบิวต์บทบาทหลายแอ็ตทริบิวต์ในฐานข้อมูลบทบาท
getsecorder	เรียกข้อมูลการจัดลำดับโดเมนสำหรับฐานข้อมูลการรักษาความปลอดภัยที่เจาะจง
setsecorder	ตั้งค่าการจัดลำดับโดเมนสำหรับฐานข้อมูลการรักษาความปลอดภัยที่เจาะจง

สิทธิพิเศษ AIX

สิทธิพิเศษที่มีอยู่ใน AIX ถูกแสดงในตาราง ต่อไปนี้ รายละเอียดของสิทธิพิเศษแต่ละสิทธิ รวมถึงการเรียกใช้ระบบที่เกี่ยวข้องจะแสดงไว้ สิทธิพิเศษบางสิทธิจัดเป็นลำดับชั้นโดยที่สิทธิพิเศษหนึ่ง สามารถให้สิทธิทั้งหมดที่เชื่อมโยงกับอีกสิทธิหนึ่ง

เมื่อตรวจสอบสิทธิพิเศษ อันดับแรกระบบจะพิจารณาว่ากระบวนการ มีสิทธิพิเศษต่ำสุดที่จำเป็นหรือไม่ จากนั้นดำเนินต่อในลำดับชั้นสูงขึ้นไป เพื่อหาการมีอยู่ของสิทธิพิเศษที่มีอำนาจมากกว่า ตัวอย่าง กระบวนการที่มีสิทธิพิเศษ **PV_AU_** จะมี สิทธิพิเศษ **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ** และ **PV_AU_WRITE** โดยอัตโนมัติ และกระบวนการที่มีสิทธิพิเศษ **PV_ROOT** จะมีสิทธิพิเศษทั้งหมดที่แสดงรายการด้านล่าง ยกเว้นสิทธิพิเศษ **PV_SU_**

สิทธิพิเศษ	คำอธิบาย	การอ้างอิงการเรียกใช้ระบบ
PV_ROOT	ให้สิทธิกระบวนการเทียบเท่าสิทธิพิเศษทั้งหมดที่แสดงรายการ ด้านล่างยกเว้น PV_SU_ (และสิทธิพิเศษที่สิทธินั้นควบคุม)	
PV_AU_ADD	อนุญาตให้กระบวนการบันทึก/เพิ่มเร็กคอร์ดการตรวจสอบ	auditlog
PV_AU_ADMIN	อนุญาตให้กระบวนการตั้งค่าและเคียวรีระบบการตรวจสอบ	audit, auditbin, auditevents, auditobj
PV_AU_PROC	อนุญาตให้กระบวนการรับค่าหรือตั้งค่าสถานะการตรวจสอบของกระบวนการ	auditproc
PV_AU_READ	กระบวนการให้กระบวนการอ่านไฟล์ที่ทำเครื่องหมายเป็นไฟล์การตรวจสอบใน Trusted AIX	
PV_AU_WRITE	อนุญาตให้กระบวนการเขียนหรือลบไฟล์ที่ทำเครื่องหมายเป็นไฟล์การตรวจสอบ หรือทำเครื่องหมายไฟล์เป็นไฟล์การตรวจสอบใน Trusted AIX	
PV_AU_	เทียบเท่ากับสิทธิพิเศษการตรวจสอบข้างต้นทั้งหมด (PV_AU_*) รวมกัน	
PV_AZ_ADMIN	อนุญาตให้กระบวนการแก้ไขตารางการรักษาความปลอดภัยเคอร์เนล	sec_setkst
PV_AZ_READ	อนุญาตให้กระบวนการเรียกค้นตารางการรักษาความปลอดภัยเคอร์เนล	sec_getkat, sec_getkpct, sec_getkpdt, sec_getkrt อื่นๆ
PV_AZ_ROOT	ทำให้กระบวนการผ่านการตรวจสอบการอนุญาตระหว่าง exec() (ใช้สำหรับวัตถุประสงค์การสืบทอด)	
PV_AZ_CHECK	ทำให้กระบวนการผ่านการตรวจสอบการอนุญาตทั้งหมด	sec_checkauth
PV_DAC_R	อนุญาตให้กระบวนการแทนที่ข้อจำกัดการอ่าน DAC	access, creat, accessx, open, read, faccessx, mkdir, getea, rename, statx, _sched_getparam, _sched_getscheduler, statea, listea
PV_DAC_W	อนุญาตให้กระบวนการแทนที่ข้อจำกัดการเขียน DAC	หลายข้อด้านบนและ setea, write, symlink, _setpri, _sched_setparam, _sched_setscheduler, fsetea, rmdir, removeea
PV_DAC_X	อนุญาตให้กระบวนการแทนที่ข้อจำกัดการเรียกใช้งาน DAC	หลายข้อด้านบนและ execve, symlink, rmdir, chdir, fchdir, ra_execve

สิทธิพิเศษ	คำอธิบาย	การอ้างอิงการเรียกใช้ระบบ
PV_DAC_O	อนุญาตให้กระบวนการแทนที่ข้อจำกัดความเป็นเจ้าของ DAC	chmod, utimes, setacl, revoke, mprotect
PV_DAC_UID	อนุญาตให้กระบวนการเปลี่ยน ID ผู้ใช้	setuid, seteuid, setuidx, setreuid, ptrace64
PV_DAC_GID	อนุญาตให้กระบวนการตั้งค่าใหม่ หรือเปลี่ยน ID กลุ่ม	setgid, setgidx, setgroups, ptrace64
PV_DAC_RID	อนุญาตให้กระบวนการตั้งค่าใหม่หรือเปลี่ยน ID บทบาท	setroles, getroles
PV_DAC_	เทียบเท่าสิทธิพิเศษ DAC ข้างต้นทั้งหมด (PV_DAC_*) รวมกัน	
PV_FS_MOUNT	อนุญาตให้กระบวนการเมาท์และเลิกเมาท์ระบบไฟล์	vmount, umount
PV_FS_MKNOD	อนุญาตให้กระบวนการสร้างไฟล์ประเภทใดๆ หรือดำเนิน การเรียกใช้ระบบ mknod	mknod
PV_FS_CHOWN	อนุญาตให้กระบวนการเปลี่ยนความเป็นเจ้าของไฟล์	chown, chownx, fchownx, lchown
PV_FS_QUOTA	อนุญาตให้กระบวนการจัดการการดำเนินการที่เกี่ยวข้องกับโควต้าดิสก์	quotactl
PV_FS_LINKDIR	อนุญาตให้กระบวนการสร้างฮาร์ดลิงก์ไปยังไดเร็กทอรี	link, unlink, remove
PV_FS_CNTL	อนุญาตให้กระบวนการดำเนินการควบคุมต่างๆ ยกเว้น การขยาย หรือการย่อระบบไฟล์	fscntl
PV_FS_RESIZE	อนุญาตให้กระบวนการดำเนินการประเภทการขยายหรือย่อ บนระบบไฟล์	fscntl
PV_FS_CHROOT	อนุญาตให้กระบวนการเปลี่ยนไดเร็กทอรี root	chroot
PV_FS_PDMODE	อนุญาตให้กระบวนการจัดทำหรือตั้งค่าไดเร็กทอรีประเภทพาร์ติชัน	pdmkdir
PV_FS_	เทียบเท่ากับสิทธิพิเศษระบบไฟล์ข้างต้นทั้งหมด (PV_FS_*) รวมกัน	
PV_PROC_PRIV	อนุญาตให้กระบวนการแก้ไขหรือดูชุดสิทธิพิเศษที่สัมพันธ์ กับกระบวนการ	setppriv, getppriv
PV_PROC_PRIO	อนุญาตให้กระบวนการ/เธรดเปลี่ยนระดับความสำคัญ นโยบาย และ พารามิเตอร์การกำหนดการอื่นๆ	_prio_requeue, _setpri, _setpriority, _getpri, _sched_setparam, _sched_setscheduler, _thread_setsched, thread_boostceiling, thread_setmystate, thread_setstate

สิทธิพิเศษ	คำอธิบาย	การอ้างอิงการเรียกใช้ระบบ
PV_PROC_CORE	อนุญาตให้กระบวนการดัมพ์ข้อมูลคอร์	gencore
PV_PROC_RAC	อนุญาตให้กระบวนการสร้างกระบวนการมากกว่าที่จำกัด ต่อหนึ่งผู้ใช้	appsetrlimit, setrlimit64, mlock, mlockall, munlock, munlockall, plock, upfget, upfput, restart, brk, sbrk
PV_PROC_RSET	รีซอร์สให้รวมชุดรีซอร์ส (rset) กับกระบวนการหรือเธรด	bindprocessor, ra_attachrset, ra_detachrset, rs_registername, rs_setnameattr, rs_discardname, rs_setpartition, rs_getassociativity, kra_mmapv
PV_PROC_ENV	อนุญาตให้กระบวนการตั้งค่าข้อมูลผู้ใช้ในโครงสร้างผู้ใช้	ue_proc_register, ue_proc_unregister, usrinfo
PV_PROC_CKPT	อนุญาตให้กระบวนการกำหนดจุดตรวจสอบหรือรีสตาร์ทกระบวนการอื่น	setcrid, restart
PV_PROC_CRED	อนุญาตให้กระบวนการตั้งค่าแอ็ตทริบิวต์ credential	__pag_setvalue, __pag_setvalue64, __pag_genpagvalue
PV_PROC_SIG	อนุญาตให้กระบวนการส่งสัญญาณไปยังกระบวนการที่ไม่เกี่ยวข้อง	_sigqueue, kill, signohup, gencore, thread_post, thread_post_many
PV_PROC_TIMER	อนุญาตให้กระบวนการส่งและใช้ตัวจับเวลารายละเอียดย่อย	appresabs, appresinc, absinterval, incinterval, _poll, _select _timer_settime
PV_PROC_RTCLK	อนุญาตให้กระบวนการเข้าถึงนาฬิกาที่เป็นเวลาของ CPU	_clock_getres, _clock_gettime, _clock_settime, _clock_getcpuclockid
PV_PROC_VARS	อนุญาตให้กระบวนการเรียกข้อมูลและอัพเดตพารามิเตอร์ที่เปลี่ยนได้ของกระบวนการ	smttune
PV_PROC_PDMODE	อนุญาตให้กระบวนการเปลี่ยนโหมด REAL ของไดเร็กทอรีที่ทำพาร์ติชัน	setppdmode
PV_PROC_	เทียบเท่ากับสิทธิพิเศษกระบวนการข้างต้นทั้งหมด (PV_PROC_*) รวมกัน	
PV_TCB	อนุญาตให้กระบวนการแก้ไขพาธไลบรารีที่ไว้วางใจของเคอร์เนล	chpriv, fchpriv
PV_TP	บ่งชี้ว่ากระบวนการเป็นกระบวนการพาธที่ไว้วางใจ และอนุญาต ให้มีการดำเนินการที่จำกัดกับกระบวนการพาธที่ไว้วางใจ (หมายเหตุ: เหมือนกับสิทธิพิเศษ AIX BYPASS_TPATH เก่า)	
PV_WPAR_CKPT	อนุญาตให้กระบวนการดำเนินการจัดทำจุดตรวจสอบ/รีสตาร์ท ใน WPAR	smcr_proc_info, smcr_exec_info, smcr_mapinfo, smcr_net_oper, smcr_procattr, aio_suspend_io, aio_resume_io

สิทธิพิเศษ	คำอธิบาย	การอ้างอิงการเรียกใช้ระบบ
PV_KER_ACCT	อนุญาตให้กระบวนการสามารถดำเนินการที่จำกัดที่เกี่ยวกับ ระบบย่อยการจัดการบัญชีผู้ใช้	acct, _acctctl, projctl
PV_KER_DR	อนุญาตให้กระบวนการเรียกใช้การดำเนินการตั้งค่าใหม่แบบไดนามิก	_dr_register, _dr_notify, _dr_unregister, dr_reconfig
PV_KER_TIME	อนุญาตให้กระบวนการแก้ไขนาฬิการะบบและเวลาระบบ	adjtime, appsettimer, _clock_settime
PV_KER_RAC	อนุญาตให้กระบวนการใช้หน้าขนาดใหญ่ (ไม่สามารถจัดหน้า) สำหรับ เซ็กเมนต์หน่วยความจำที่แบ่งใช้	shmctl, vmgetinfo
PV_KER_WLM	อนุญาตให้กระบวนการเตรียมข้อมูลเบื้องต้นและแก้ไขการตั้งค่า WLM	_wlm_set, _wlm_tune, _wlm_assign
PV_KER_EWLM	อนุญาตให้กระบวนการเตรียมข้อมูลเบื้องต้นหรือเคียวรีสภาวะแวดล้อม eWLM	
PV_KER_VARS	อนุญาตให้กระบวนการตรวจสอบหรือตั้งค่าพารามิเตอร์ที่เปลี่ยนได้ตอนรันไทม์ของเคอร์เนล	sys_parm, getkerninfo, __pag_setname, sysconfig, kunload64
PV_KER_REBOOT	อนุญาตให้กระบวนการปิดทำงานระบบ	reboot
PV_KER_RAS	อนุญาตให้กระบวนการตั้งค่าหรือเขียนเร็กคอร์ด RAS การ บันทึกข้อผิดพลาด การติดตาม ดัมพ์ฟังก์ชัน	mtrace_set, mtrace_ctl
PV_KER_LVM	อนุญาตให้กระบวนการตั้งค่าระบบย่อย LVM	
PV_KER_NFS	อนุญาตให้กระบวนการตั้งค่าระบบย่อย NFS	
PV_KER_VMM	อนุญาตให้กระบวนการแก้ไขกระบวนการการสลับค่าและพารามิเตอร์ที่เปลี่ยนได้ VMM อื่นๆ ในเคอร์เนล	swapoff, _swapon_ext, vmgetinfo
PV_KER_WPAR	อนุญาตให้กระบวนการตั้งค่าเวิร์กโหลดพาร์ติชัน	brand, corral_config, corral_delete, corral_modify, wpar_mkdevexport, wpar_rmdevexport, wpar_lsdevexport
PV_KER_CONF	อนุญาตให้กระบวนการดำเนินการตั้งค่าระบบ ที่แตกต่างกัน	sethostname, sethostid, unameu, setdomainname
PV_KER_EXTCONF	อนุญาตให้กระบวนการดำเนินงานการตั้งค่าต่างๆ ในส่วนขยายเคอร์เนล (สำหรับเซอร์วิสส่วนขยายเคอร์เนล)	

สิทธิพิเศษ	คำอธิบาย	การอ้างอิงการเรียกใช้ระบบ
PV_KER_IPC	อนุญาตให้กระบวนการเพิ่มค่าของบัฟเฟอร์คิวข้อความ IPC และอนุญาตให้ shmget ที่มีช่วงที่จะรวม	msgctl, shm_open, shmget, ra_shmget, ra_shmgetv, shmctl
PV_KER_IPC_R	อนุญาตให้กระบวนการอ่านคิวข้อความ IPC ชุดเซมาฟอร์ หรือเซ็กเมนต์หน่วยความจำที่แบ่งใช้	msgctl, __msgrcv, _mq_open, semctl, shmat, shm_open, __semop, shmctl, __semtimedop, sem_post, _sem_wait, __msgrcv, __msgxrcv
PV_KER_IPC_W	อนุญาตให้กระบวนการเขียนลงคิวข้อความ IPC ชุดเซมาฟอร์ หรือเซ็กเมนต์หน่วยความจำที่แบ่งใช้	_mq_open, shmat, _sem_open, semctl, shm_open, shmctl, mq_unlink, sem_unlink, shm_unlink, msgctl, __msgsnd
PV_KER_IPC_O	อนุญาตให้กระบวนการแทนที่ความเป็นเจ้าของ DAC บนอ็อบเจ็กต์ IPC ทั้งหมด	msgctl, semctl, shmctl, fchmod, fchown
PV_KER_SECCONFIG	อนุญาตให้กระบวนการตั้งค่าแฟล็กการรักษาความปลอดภัยตั้งค่า	sec_setsecconf, sec_setrunmode, sec_setsyslab, sec_getsyslab
PV_KER_PATCH	อนุญาตให้กระบวนการแพ็ตช์ส่วนขยายเคอร์เนล	
PV_KER_	เทียบเท่าสิทธิพิเศษเคอร์เนลข้างต้นทั้งหมด (PV_KER_*) รวมกัน	
PV_DEV_CONFIG	อนุญาตให้กระบวนการตั้งค่าส่วนขยายเคอร์เนล และอุปกรณ์ในระบบ	sysconfig
PV_DEV_LOAD	อนุญาตให้กระบวนการโหลดและยกเลิกการโหลดส่วนขยายเคอร์เนลและ อุปกรณ์ในระบบ	sysconfig
PV_DEV_QUERY	อนุญาตให้กระบวนการเคียวรีเคอร์เนลโมดูล	sysconfig
PV_SU_ROOT	ให้สิทธิพิเศษทั้งหมดแก่กระบวนการที่เชื่อมโยงกับ AIX superuser มาตรฐาน	
PV_SU_EMUL	ให้สิทธิพิเศษทั้งหมดแก่กระบวนการที่เชื่อมโยงกับ AIX super user มาตรฐานถ้า UID เป็น 0	
PV_SU_UID	ทำให้การเรียกใช้ระบบ getuid ส่งกลับค่า 0	getuidx
PV_SU_	เทียบเท่าสิทธิพิเศษ superuser ข้างต้นทั้งหมด (PV_SU_*) รวมกัน	
PV_NET_CNTL	อนุญาตให้กระบวนการแก้ไขตารางเน็ตเวิร์ก	socket, bind, listen, _naccept, econnect, ioctl, rmsock, setsockopt

สิทธิพิเศษ	คำอธิบาย	การอ้างอิงการเรียกใช้ระบบ
PV_NET_PORT	อนุญาตให้กระบวนการประมวลผลโยงพอร์ตที่มีสิทธิพิเศษ	bind
PV_NET_RAWSOCK	อนุญาตให้กระบวนการมีการเข้าถึงโดยตรงไปยังเน็ตเวิร์กเลเยอร์	socket, _send, _sendto, sendmsg, _nsendmsg
PV_NET_CONFIG	อนุญาตให้กระบวนการตั้งค่าพารามิเตอร์เกี่ยวกับเน็ตเวิร์ก	
PV_NET_	เทียบเท่ากับสิทธิ์ในระบบเครือข่ายด้านบนทั้งหมด (PV_NET_*) รวม	

สิทธิพิเศษที่แสดงในตารางต่อไปนี้เฉพาะ Trusted AIX:

สิทธิพิเศษ Trusted AIX	คำอธิบาย	การอ้างอิงการเรียกใช้ระบบ
PV_LAB_CL	อนุญาตให้กระบวนการแก้ไข subject SCLs เพื่อล้างค่าของ กระบวนการ	
PV_LAB_CLTL	อนุญาตให้กระบวนการแก้ไข subject TCLs เพื่อล้างค่าของ กระบวนการ	
PV_LAB_LEF	อนุญาตให้กระบวนการอ่านไฟล์การเข้ารหัสเลเบล	
PV_LAB_SLDG	อนุญาตให้กระบวนการดาวน์เกรด SLs เพื่อล้างค่าของ กระบวนการ	
PV_LAB_SLDG_STR	อนุญาตให้กระบวนการดาวน์เกรด SL ของแพ็กเก็ต เพื่อ ล้างค่าของกระบวนการ	
PV_LAB_SL_FILE	อนุญาตให้กระบวนการเปลี่ยน object SLs เพื่อล้างค่า ของกระบวนการ	
PV_LAB_SL_PROC	อนุญาตให้กระบวนการเพื่อเปลี่ยน subject SL เพื่อล้างค่าของ กระบวนการ	
PV_LAB_SL_SELF	อนุญาตให้กระบวนการเปลี่ยน SL ของตนเพื่อล้างค่าของ กระบวนการ	
PV_LAB_SLUG	อนุญาตให้กระบวนการอัพเกรด SLs เพื่อล้างค่าของ กระบวนการ	
PV_LAB_SLUG_STR	อนุญาตให้กระบวนการอัพเกรด SL ของแพ็กเก็ต เพื่อ ล้างค่าของกระบวนการ	
PV_LAB_TL	อนุญาตให้กระบวนการแก้ไข subject และ object TLs	
PV_LAB_	เทียบเท่าสิทธิพิเศษเลเบลข้างต้นทั้งหมด (PV_LAB_*) รวมกัน	

สิทธิพิเศษ Trusted AIX	คำอธิบาย	การอ้างอิงการเรียกใช้ระบบ
PV_MAC_CL	อนุญาตให้กระบวนการข้ามข้อจำกัดการล้างค่าระดับความลับ	
PV_MAC_R_PROC	อนุญาตให้กระบวนการข้ามข้อจำกัดการอ่าน MAC เมื่อรับ ข้อมูลเกี่ยวกับกระบวนการ โดยที่เลเบลของกระบวนการเป้าหมาย อยู่ภายในการล้างค่าของกระบวนการที่กระทำการ	
PV_MAC_W_PROC	อนุญาตให้กระบวนการข้ามข้อจำกัดการเขียน MAC เมื่อส่ง สัญญาณไปยังกระบวนการ โดยที่เลเบลของกระบวนการเป้าหมาย อยู่ภายในการล้างค่าของกระบวนการที่กระทำการ	
PV_MAC_R	อนุญาตให้กระบวนการข้ามข้อจำกัดการอ่าน MAC	
PV_MAC_R_CL	อนุญาตให้กระบวนการข้ามข้อจำกัดการอ่าน MAC เมื่อ เลเบลของอ็อบเจ็กต์อยู่ภายในการล้างค่าของกระบวนการ	
PV_MAC_R_STR	อนุญาตให้กระบวนการข้ามข้อจำกัดการอ่าน MAC เมื่ออ่าน ข้อความจาก STREAM โดยที่เลเบลของข้อความอยู่ภายใน การล้างค่าของกระบวนการ	
PV_MAC_W	อนุญาตให้กระบวนการข้ามข้อจำกัดการเขียน MAC	
PV_MAC_W_CL	อนุญาตให้กระบวนการข้ามข้อจำกัดการเขียน MAC เมื่อ เลเบลของอ็อบเจ็กต์อยู่ภายในการล้างค่าของกระบวนการ	
PV_MAC_W_DN	อนุญาตให้กระบวนการข้ามข้อจำกัดการเขียน MAC เมื่อ เลเบลกระบวนการควบคุมเลเบลของอ็อบเจ็กต์ และเลเบลของอ็อบเจ็กต์ อยู่ภายในการล้างค่าของกระบวนการ	
PV_MAC_W_UP	อนุญาตให้กระบวนการข้ามข้อจำกัดการเขียน MAC เมื่อ เลเบลกระบวนการถูกควบคุมโดยเลเบลของอ็อบเจ็กต์ และเลเบลของอ็อบเจ็กต์อยู่ภายในการล้างค่าของกระบวนการ	
PV_MAC_OVRRD	ข้ามข้อจำกัด MAC สำหรับไฟล์ที่แฟล็กเป็นได้รับยกเว้น จาก MAC	
PV_MAC_	เทียบเท่าสิทธิพิเศษ MAC ข้างต้นทั้งหมด (PV_MAC_*) รวมกัน	

สิทธิพิเศษ Trusted AIX	คำอธิบาย	การอ้างอิงการเรียกใช้ระบบ
PV_MIC	อนุญาตให้กระบวนการข้ามข้อจำกัด integrity	
PV_MIC_CL	อนุญาตให้กระบวนการข้ามข้อจำกัดการล้างค่า integrity	

โดเมน RBAC

การควบคุมการเข้าถึงตามบทบาท (Role-based access control - RBAC) เริ่มใช้งานครั้งแรกใน AIX 6.1 ซึ่งจัดเตรียมกลไกเพื่อแยกหลายๆ ฟังก์ชันของ ผู้ใช้ root ขั้นสูงออกเป็นบทบาทต่างๆ ซึ่งสามารถมอบหมายให้กับผู้ใช้อื่นในระบบได้ RBAC จัดเตรียม สิ่งอำนวยความสะดวกต่างๆ ในการมอบหมายหน้าที่ และปรับปรุงความปลอดภัยของระบบ เนื่องจากการตรวจสอบและติดตามกิจกรรมต่างๆ บนระบบสามารถทำได้ง่ายขึ้น RBAC ยังจัดเตรียมการมอบหมายความรับผิดชอบให้กับผู้ใช้อื่น (อ้างอิงเป็น ผู้ใช้ที่ได้รับอนุญาต) แต่ไม่ได้จัดเตรียมกลไกให้ด้วยเพื่อจำกัดสิทธิการควบคุมดูแล ของผู้ใช้ที่ได้รับอนุญาตเฉพาะรีซอร์สที่กำหนดของ ระบบ ตัวอย่างเช่น ผู้ใช้ที่มีสิทธิการควบคุมดูแลเครือข่าย สามารถจัดการอินเตอร์เฟสเครือข่ายทุกอย่างของบนระบบได้ แต่คุณไม่สามารถ จำกัดผู้ใช้ที่ได้รับอนุญาตให้แก้ไขชุดอินเตอร์เฟส

คุณลักษณะโดเมนสำหรับ RBAC ถูกใช้เพื่อจำกัดสิทธิของผู้ใช้ที่ได้รับอนุญาต ผู้ใช้และรีซอร์สของระบบจะถูกกำหนดชื่อโดยแท็กที่แนบที่เรียกว่า โดเมน และกฎการเข้าถึงเฉพาะจะกำหนดสิทธิใช้งาน รีซอร์สตามผู้ใช้

นิยาม นิยามต่อไปนี้สัมพันธ์กับกฎการเข้าถึง:

หัวเรื่อง (subject): หัวเรื่องคือรายการที่ร้องขอการเข้าถึง อ็อบเจ็กต์ ตัวอย่างของหัวเรื่องคือ โปรเซส

อ็อบเจ็กต์ (object): อ็อบเจ็กต์คือรายการที่เก็บข้อมูลของ ค่า ตัวอย่างของอ็อบเจ็กต์คือ ไฟล์ อุปกรณ์ และพอร์ตเครือข่าย

โดเมน (domain): โดเมนถูกกำหนดเป็นหมวดหมู่ที่รวมรายการไว้ เมื่อรายการต่างๆ ถูกรวมเข้าในโดเมน สิทธิควบคุมการเข้าถึงรายการ จะเป็นไปตามกฎการเข้าถึงต่อไปนี้:

กฎการเข้าถึง

- หัวเรื่องสามารถเข้าอ็อบเจ็กต์เมื่อหัวเรื่องมีโดเมนทั้งหมด ที่เป็นของอ็อบเจ็กต์ ซึ่งระบุรายชื่อโดเมน ที่เป็นเจ้าของหัวเรื่องคือ super set ของโดเมนของอ็อบเจ็กต์ ค่านี้เป็นลักษณะการทำงานดีฟอลต์
- หัวเรื่องสามารถเข้าถึงอ็อบเจ็กต์เมื่อหัวเรื่องมีอย่างน้อยหนึ่งโดเมนของอ็อบเจ็กต์ ซึ่งหัวเรื่องและอ็อบเจ็กต์มีหนึ่งโดเมนร่วมกัน พฤติกรรมนี้ขึ้นอยู่กับแฟล็กความปลอดภัยของอ็อบเจ็กต์
- อ็อบเจ็กต์สามารถปฏิเสธการเข้าถึงโดเมนบางอย่างได้ ถ้าอ็อบเจ็กต์กำหนด ชุดโดเมนที่เรียกชุดที่ขัดแย้ง และถ้าหนึ่งในโดเมนของหัวเรื่อง เป็นส่วนหนึ่งของชุดที่ขัดแย้ง อ็อบเจ็กต์สามารถปฏิเสธการเข้าถึง หัวเรื่องได้

ฐานข้อมูลโดเมน (Domains Database)

โดเมนที่สนับสนุนโดยระบบต้องถูกเก็บไว้ในไฟล์คอนฟิกูเรชัน ภายใต้ `/etc/security/domains` รูปแบบ ของ stanza ในไฟล์เป็นดังนี้:

```
domain-name:
id = <number>
dfltmsg = <Message>
msgcat = <Message catalog>
msgset = <Message set in catalog>
msgnum = <Message id in catalog>
```

ฐานข้อมูลสามารถ จัดการได้โดยใช้คำสั่ง **mkdom** และ **chdom** ใช้คำสั่ง **lsdom** เพื่อดูฐานข้อมูล หากต้องการลบ รายการให้ใช้คำสั่ง **rmdom**

รายการในฐานข้อมูลจะไม่มีผลใช้งานจนกว่ารายการนั้นถูกดาวน์โหลด เข้าสู่เคอร์เนลโดยใช้คำสั่ง **setkst**

โดเมนสูงสุด 1024 โดเมนที่สนับสนุนบนระบบ และค่าสูงสุดที่เป็นไปได้ ของตัวบ่งชี้โดเมน (แอ็ตทริบิวต์ ID) คือ 1024

อ็อบเจ็กต์ที่กำหนดโดเมน (Domain-Assigned Objects)

เมื่อต้องการกำหนดโดเมนให้กับอ็อบเจ็กต์ อ็อบเจ็กต์นั้นต้องกำหนดไว้ในฐานข้อมูล Domain-Assigned Objects โดเมนสำหรับรายการทั้งหมดบนระบบจะ ถูกเก็บไว้ในไฟล์คอนฟิกูเรชันภายใต้ /etc/security/domobjs รูปแบบของ stanza ในไฟล์เป็นดังต่อไปนี้ ซึ่งเป็นตัวอย่าง ในการกำหนดโดเมนให้กับอ็อบเจ็กต์

```
/dev/hrvg:
domains=HR,IT
conflictsets=payroll
objtype=device
secflags=FSF_DOM_ANY
```

โดเมน (domains): ระบุโดเมนที่อนุญาตให้เข้าถึง อ็อบเจ็กต์ ตัวอย่างของโดเมนได้แก่ IT, HR และ Payroll

ชนิดอ็อบเจ็กต์ (objtype): บ่งชี้ชนิดของอ็อบเจ็กต์ที่ได้รับมอบหมาย โดเมน โดยมี objtypes ที่แตกต่างกันคือ device, file, netint, และ netport

ชุดที่ขัดแย้ง (conflict sets): บ่งชี้เมื่อหัวเรื่องเป็นของ โดเมนใดๆ ที่ที่แสดงในแอ็ตทริบิวต์นี้ในชุดนี้ ซึ่งไม่อนุญาตให้เข้าถึง อ็อบเจ็กต์

secflags: แฟล็กนี้ระบุคุณสมบัติพิเศษของ อ็อบเจ็กต์ แฟล็กสามารถตั้งค่าเป็น **FSF_DOM_ANY** หรือ **FSF_DOM_ALL** ถ้าแฟล็กถูกตั้งค่าเป็น **FSF_DOM_ANY** หัวเรื่องสามารถเข้าถึงอ็อบเจ็กต์ ได้ถ้าอ็อบเจ็กต์มีหนึ่งในโดเมนระบุไว้ในรายการแอ็ตทริบิวต์ domains แต่ถ้าแฟล็กถูกตั้งค่าเป็น **FSF_DOM_ALL** โดเมนทั้งหมด ในรายการต้องเป็นตรงกับหัวเรื่องเพื่อเข้าถึงอ็อบเจ็กต์ ถ้าไม่มีค่าถูกระบุ ค่าดีฟอลต์ **FSF_DOM_ALL** จะถูกใช้งาน **secflag** มีผลใช้งานเฉพาะพฤติกรรมของแอ็ตทริบิวต์ domains ของอ็อบเจ็กต์

โดเมนสามารถกำหนดเป็นไฟล์ในระบบไฟล์ได้ ตามค่าดีฟอลต์ โดเมนทั้งหมดของอ็อบเจ็กต์ต้องเป็นเซ็ตย่อยของโดเมนของโปรเซส เพื่ออนุญาตให้โปรเซสเข้าถึงอ็อบเจ็กต์

1. อุปกรณ์ (Devices): อุปกรณ์ทั้งหมด (รวมถึงระบบไฟล์) สามารถกำหนดให้กับ โดเมนได้ ซึ่งการตรวจสอบโดเมนดำเนินการเสร็จสิ้นระหว่างกิจกรรมการจัดการ เช่น การกำหนดคอนฟิกอุปกรณ์

   ```
   /dev/hrvg:
   domains=HR,IT
   conflictsets=payroll
   objtype=device
   secflags=FSF_DOM_ANY
   ```

2. อินเตอร์เฟสเครือข่าย (Network interfaces): เมื่ออินเตอร์เฟสเครือข่าย (เช่น: en0) ถูกกำหนดให้กับโดเมน กิจกรรมการจัดการ เช่น การปิดอินเตอร์เฟส จะต้องการให้อินเตอร์เฟสทำการตรวจสอบโดเมน

```
en0:
domains=NETIF,ADMIN
objtype=netint
flags=FSF_DOM_ALL
```

3. พอร์ตเครือข่าย (Network ports): พอร์ต TCP และ UDP สามารถกำหนดให้แก่โดเมนได้ การตรวจสอบโดเมนถูกบังคับใช้เมื่อแอ็พพลิเคชันพยายามรวม พอร์ต

```
TCP_<port#>:
domains=NETIF,ADMIN
type=netport
flags=FSF_DOM_ALL
```

4. โปรเซส (Processes): โปรเซสสืบทอดโดเมนของผู้ใช้ที่สั่งการ ให้โปรเซสรัน เมื่อผู้ใช้ล็อกอิน โปรเซสเชลล์ของผู้ใช้ จะมีโดเมนของผู้ใช้อยู่ เมื่อโดเมนถูกตั้งค่า โดเมนเหล่านี้ของโปรเซส จะยังคงอยู่ตลอดช่วงอายุ โดเมนของโปรเซส ไม่สามารถเปลี่ยนแปลงได้โดยส่วนติดต่อผู้ใช้หรือการเรียกระบบ เฉพาะโปรเซส ที่สามารถตั้งค่าโดเมนคือโปรเซสที่ล็อกอิน โปรเซสไม่มีแอ็ตทริบิวต์ **conflict set** และ **secflags**

ข้อจำกัด Current

รายการ ต่อไปนี้เป็นข้อจำกัดสำหรับโปรแกรมอำนวยความสะดวก RBAC โดเมนปัจจุบัน:

- ในตอนนี้ไฟล์คอนฟิกูเรชันของโดเมน สนับสนุนบนระบบโลคัล และไม่มีเวอร์ชันขนาดเล็ก สำหรับเซิร์ฟเวอร์ directory access protocol (LDAP)
- โดเมน RBAC ไม่สามารถใช้ได้ภายใน AIX workload partitions (WPARs)
- คุณไม่สามารถใช้โดเมน RBAC กับไฟล์ชั่วคราว

ข้อกำหนด RBAC ที่ปรับปรุง

โดเมน RBAC ถูกสร้างบน RBAC ที่ปรับปรุง และต้องการ RBAC ที่ปรับปรุงเพื่อเปิดใช้งาน บนระบบ

ตารางความปลอดภัยของเคอร์เนล

โดเมนและอ็อบเจ็กต์ที่กำหนดโดเมน ที่ได้กำหนดไว้ในฐานข้อมูลโดเมน และฐานข้อมูล Domain-Object จะมีผลใช้งานหลังจากที่ถูกดาวน์โหลดเข้าสู่เคอร์เนลโดยใช้คำสั่ง **setkst** ตารางสองตารางถูกอ้างถึงเป็น Kernel Domain Table (KDOMT) และ Kernel Domain Object Table (KDOT)

สำหรับรายละเอียดเพิ่มเติมเกี่ยวกับ ตารางความปลอดภัยของเคอร์เนล และ **setkst** โปรดดูหัวข้อ role based access control (RBAC) ในคำแนะนำด้านความปลอดภัยของ AIX

คำสั่งโดเมน

ตารางต่อไปนี้แสดงรายการ คำสั่งที่เกี่ยวข้องกับโดเมน RBAC ที่จัดเตรียมไว้ในระบบปฏิบัติการ AIX เพื่อจัดการ และใช้งานกรอบงาน domain-RBAC:

คำสั่ง	คำอธิบาย
mkdom	สร้างโดเมนใหม่
lsdom	แสดงแอ็ตทริบิวต์ของโดเมน
rmdom	ลบโดเมน
chdom	เปลี่ยนแอ็ตทริบิวต์ของโดเมน
setsecattr	ตั้งค่าแอ็ตทริบิวต์ความปลอดภัย ของฐานข้อมูลโดเมน-อ็อบเจ็กต์
lssecattr	แสดงผลแอ็ตทริบิวต์ ความปลอดภัยของฐานข้อมูลโดมน-อ็อบเจ็กต์
rmsecattr	ลบนิยามของฐานข้อมูล โดเมน-อ็อบเจ็กต์
setkst	ส่งรายการในฐานข้อมูลระดับผู้ใช้ ของโดเมน RBAC ให้กับตารางความปลอดภัยของ Kernel

ไฟล์ที่เกี่ยวข้องกับโดเมน RBAC

ตารางต่อไปนี้ แสดงไฟล์ที่เกี่ยวข้องกับ RBAC ซึ่งจัดเตรียมไว้ในระบบปฏิบัติการ AIX เพื่อกำหนดคอนฟิก และจัดเก็บข้อมูลของฐานข้อมูล:

ไฟล์	คำอธิบาย
`/etc/security/domains`	ฐานข้อมูลโดเมน
`/etc/security/domobjs`	ฐานข้อมูลโดเมน-อ็อบเจ็กต์

การใช้โดเมน

การกำหนดโดเมน: โดเมน ถูกกำหนดไว้ในฐานข้อมูลโดเมนโดยใช้คำสั่ง **mkdom**

```
mkdom id=24 HR
```

การมอบหมาย โดเมน: โดเมนสามารถมอบหมายให้กับรายการต่างๆ เช่น ผู้ใช้ ไฟล์ อุปกรณ์ พอร์ตเครือข่าย และอินเตอร์เฟส รายการทั้งหมดนอกจาก ชุดที่ขัดแย้งและแฟล็กความปลอดภัย (**secflags**) ที่สนับสนุนโดยผู้ใช้

ผู้ใช้: ผู้ใช้ถูกกำหนดให้กับโดเมนโดยใช้คำสั่ง **chuser**, และ **chsec**

ไวยากรณ์:

```
chuser domains = <รายการที่คั่นด้วยคอมมาของ โดเมน> ชื่อผู้ใช้
```

ตัวอย่าง:

```
chuser domains=INET john
```

ในระหว่างการล็อกอิน โดเมนจะกำหนดให้กับผู้ใช้ที่ถูกเรียกใช้งาน คุณต้องล็อกอินใหม่อีกครั้ง ในกรณีที่โดเมนเปลี่ยนแปลงขณะที่เซสชันของคุณถูกใช้งานอยู่ เพื่อให้โดเมนใหม่มีผลใช้งาน

อ็อบเจ็กต์: เมื่อต้องการจำกัด การเข้าถึงอ็อบเจ็กต์ผ่านโดเมน อ็อบเจ็กต์ต้องกำหนดไว้ในฐานข้อมูล Domain-Object โดยใช้คำสั่ง **setsecattr**

ไวยากรณ์:

```
setsecattr -o domains=<comma-separated list of allowed domains>
conflictsets=<comma-separated list of restricted domains>
secflags=<FSF_DOM_ALL or FSF_DOM_ANY>
objtype=<ไฟล์หรืออุปกรณ์หรือ netint หรือ netport>
object-path
```

ตัวอย่าง:

```
setsecattr -o domains=INET,WEB conflictsets=DB secflags=FSF_DOM_ANY objtype=netint en0
```

Access Control Lists

โดยปกติ ACL ประกอบด้วยชุดของรายการที่เรียกว่า Access Control Entry (ACE) แต่ละ ACE จะกำหนดสิทธิการเข้าถึงสำหรับ ผู้ใช้ที่เกี่ยวกับอ็อบเจ็กต์

เมื่อมีความพยายามที่จะเข้าถึง ระบบปฏิบัติการจะใช้ ACL ที่เชื่อมโยงกับอ็อบเจ็กต์นั้นเพื่อดูว่าผู้ใช้มีสิทธิ เข้าถึงหรือไม่ ACLs และการเข้าถึงที่เกี่ยวข้องเหล่านี้จะตรวจสอบจากค่าหลักของ กลไก Discretionary Access Control (DAC) ที่สนับสนุนโดย AIX

ระบบปฏิบัติการสนับสนุนอ็อบเจ็กต์ระบบหลายประเภทที่ อนุญาตให้ผู้ใช้ดำเนินการเพื่อจัดเก็บหรือสื่อสารข้อมูล ประเภทของอ็อบเจ็กต์ที่ถูกควบคุมการเข้าถึงที่สำคัญที่สุดเป็นดังนี้:

- ไฟล์และไดเร็กทอรี
- ไพพ์ที่มีชื่อ
- อ็อบเจ็กต์ IPC เช่นคิวข้อความ เซ็กเมนต์หน่วยความจำที่แบ่งใช้ และ เซมาฟอร์

การตรวจสอบสิทธิการเข้าถึงทั้งหมดสำหรับอ็อบเจ็กต์เหล่านี้ถูกกระทำที่ ระดับการเรียกใช้ระบบเมื่ออ็อบเจ็กต์ถูกเข้าถึงเป็นครั้งแรก เนื่องจากอ็อบเจ็กต์ System V Interprocess Communication (SVIPC) ถูกเข้าถึงแบบไม่แสดงสถานะ การตรวจสอบจะกระทำกับการเข้าถึงทุกครั้ง สำหรับอ็อบเจ็กต์ที่มีชื่อระบบไฟล์ จำเป็นต้องสามารถระบุชื่อของอ็อบเจ็กต์ที่แท้จริงได้ ชื่อถูกระบุเป็นแบบเชิงสัมพันธ์ (กับไดเร็กทอรีที่กำลังทำงานของกระบวนการ) หรือแบบสัมบูรณ์ (กับไดเร็กทอรี root ของกระบวนการ) การระบุชื่อทั้งหมด เริ่มต้นโดยการค้นหาไดเร็กทอรีใดไดเร็กทอรีหนึ่งต่อไปนี้

กลไกการควบคุมการเข้าถึงที่ยอดเยี่ยมอนุญาตการควบคุมการเข้าถึง แหล่งข้อมูลอย่างมีประสิทธิภาพ และมีการป้องกันแยกต่างหากสำหรับ ข้อมูลที่เป็นความลับและความถูกต้อง กลไกการควบคุมการเข้าถึงที่ควบคุมโดยเจ้าของมีประสิทธิภาพเท่ากับผู้ใช้กระทำเท่านั้น ผู้ใช้ทั้งหมดต้องเข้าใจว่ามีการให้สิทธิ และการปฏิเสธ การเข้าถึงอย่างไร และค่าเหล่านี้ถูกตั้งค่าอย่างไร

ตัวอย่าง ACL ที่เชื่อมโยงกับอ็อบเจ็กต์ระบบไฟล์ (ไฟล์ หรือไดเร็กทอรี) สามารถบังคับเพื่อให้สิทธิการเข้าถึงแก่ผู้ใช้ที่แตกต่างกัน โดยไม่คำนึงถึงการเข้าถึงอ็อบเจ็กต์ เป็นไปได้ที่ ACL อาจ บังคับให้สิทธิการเข้าถึงในระดับที่ต่างออกไป เช่นการอ่านหรือเขียน สำหรับผู้ใช้ที่แตกต่างกัน

โดยปกติ แต่ละอ็อบเจ็กต์จะมีเจ้าของที่กำหนด และในบางกรณี จะเชื่อมโยงกับกลุ่มหลัก เจ้าของอ็อบเจ็กต์ที่เจาะจง จะควบคุมแอ็ตทริบิวต์เข้าถึงที่เกี่ยวข้อง แอ็ตทริบิวต์ของเจ้าของ ถูกตั้งค่าเป็น ID ผู้ใช้ที่มีผลของกระบวนการสร้าง

รายการต่อไปนี้มีแอ็ตทริบิวต์การควบคุมการเข้าถึงโดยตรงสำหรับ อ็อบเจ็กต์ชนิดต่างกัน:

เจ้าของ

สำหรับอ็อบเจ็กต์ System V Interprocess Communication (SVIPC) ผู้สร้าง หรือเจ้าของสามารถเปลี่ยนแปลงค่าความเป็นเจ้าของของอ็อบเจ็กต์ อ็อบเจ็กต์ SVIPC มี ผู้สร้างที่เชื่อมโยง ซึ่งมีสิทธิทั้งหมดของเจ้าของ (รวมถึง การอนุญาตเข้าถึง) ผู้สร้างไม่สามารถเปลี่ยนแปลง แม้ว่าจะใช้สิทธิ ระดับ root

อ็อบเจ็กต์ SVIPC ถูกเตรียมข้อมูลเบื้องต้นให้แก่ ID กลุ่มประสิทธิภาพ ของกระบวนการสร้าง สำหรับอ็อบเจ็กต์ระบบไฟล์ แอ็ตทริบิวต์ควบคุม การเข้าถึงโดยตรงถูกเตรียมข้อมูลให้แก่ ID กลุ่มที่มีผล ของกระบวนการสร้าง หรือ ID กลุ่มของไดเร็กทอรีพาเรนต์ (ค่านี้ ถูกกำหนดโดยแฟล็กการสืบทอดกลุ่มของไดเร็กทอรีพาเรนต์)

กลุ่ม เจ้าของอ็อบเจ็กต์สามารถเปลี่ยนกลุ่มได้ กลุ่มใหม่ต้องเป็น ID กลุ่มที่มีผลของกระบวนการสร้าง หรือ ID กลุ่มของไดเร็กทอรีพาเรนต์อย่างใดอย่างหนึ่ง (เช่นด้านบน อ็อบเจ็กต์ SVIPC มี กลุ่มการสร้างที่สัมพันธ์ซึ่งไม่สามารถเปลี่ยนแปลงได้ และใช้การอนุญาต เข้าถึงของกลุ่มอ็อบเจ็กต์ร่วมกัน)

โหมด คำสั่ง **chmod** (โหมดตัวเลขที่มีเครื่องหมาย ฐานแปด) สามารถตั้งค่าสิทธิและแอ็ตทริบิวต์พื้นฐาน รูทีนย่อย **chmod** ที่ถูกเรียกใช้โดยคำสั่ง จะปิดใช้งานสิทธิเพิ่มเติม สิทธิ เพิ่มเติมถูกปิดใช้งานถ้าคุณใช้โหมดตัวเลขของคำสั่ง **chmod** บนไฟล์ที่มี ACL โหมดสัญลักษณ์ของคำสั่ง **chmod** ปิดใช้งาน ACLs เพิ่มเติมสำหรับประเภท NSF4 ACL แต่ไม่ปิดใช้งานสิทธิ เพิ่มเติมสำหรับประเภท AIXC ACLs สำหรับข้อมูลเกี่ยวกับโหมดตัวเลข และสัญลักษณ์ ดูที่ **chmod**

หลายอ็อบเจ็กต์ในระบบปฏิบัติการ เช่นอ็อบเจ็กต์ซ็อกเก็ตและ ระบบไฟล์ ที่มี ACLs ซึ่งเชื่อมโยงสำหรับอ็อบเจ็กต์ที่ต่างกัน รายละเอียด ของ ACLs สำหรับประเภทอ็อบเจ็กต์เหล่านี้อาจแตกต่างกันไป

โดยทั่วไป AIX ให้การสนับสนุน บิตโหมดสำหรับการควบคุมการเข้าถึงอ็อบเจ็กต์ระบบไฟล์ ทั้งยัง สนับสนุนค่าเฉพาะจากบิตโหมดของ ACL ACL นี้ประกอบด้วย บิตโหมดพื้นฐาน และยังอนุญาตให้มีนิยามของ ACE หลายรายการ แต่ละรายการ ACE จะกำหนดสิทธิการเข้าถึงสำหรับผู้ใช้ หรือกลุ่ม ภายในบิตโหมด ชนิดคลาสลิคของ ลักษณะการทำงาน ACL นี้จะยังคงได้รับการสนับสนุน, และมีชื่อชนิดว่า ชนิด AIXC ACL

โปรดทราบว่าการสนับสนุน ACL บนอ็อบเจ็กต์ระบบไฟล์จะขึ้นกับ ระบบฟิสิคัลไฟล์ (PFS) ที่ต้องการ PFS ต้องเข้าใจใน ข้อมูล ACL และสามารถเก็บ เรียกออกมา และบังคับการเข้าถึง สำหรับผู้ใช้หลากหลาย ทั้งยังเป็นไปได้ที่บางระบบฟิสิคัลไฟล์ไม่สนับสนุน ACLs ใดๆ เลย (อาจสนับสนุนเพียงบิตโหมดพื้นฐาน) ดังที่เปรียบเทียบกับระบบฟิสิคัลไฟล์ที่สนับสนุน ACL หลายประเภท ระบบไฟล์จำนวนไม่มากภายใต้ AIX ได้รับการปรับปรุงให้สนับสนุนชนิด ACL จำนวนมาก JFS2 และ GPFS™ จะมีความสามารถในการสนับสนุน ประเภท ACL ที่อิงตามโปรโตคอล NFS เวอร์ชัน 4 เช่นกัน ACL นี้ชื่อประเภท ACL NFS4 บน AIX ประเภท ACL นี้ ยึดถือตามนิยาม ACL เป็นส่วนใหญ่ในข้อกำหนดคุณสมบัติโปรโตคอล NFS เวอร์ชัน 4 รวมทั้งสนับสนุนการควบคุมการเข้าถึงแบบรวมมากขึ้นเมื่อ เปรียบเทียบกับประเภท AIXC ACL และจัดให้มีความสามารถเช่น การสืบทอด

การสนับสนุนเฟรมเวิร์กประเภทหลายรายการค่าควบคุมการเข้าใช้

เริ่มต้นด้วยเวอร์ชัน 5.3.0, ระบบปฏิบัติการ AIX สนับสนุน โครงสร้างพื้นฐานสำหรับ Access Control List (ACL) ชนิดอื่นๆ ที่มีอยู่สำหรับอ็อบเจ็กต์ระบบไฟล์ที่แตกต่างกันภายในระบบปฏิบัติการ

โครงสร้างพื้นฐานนี้อนุญาตให้ใช้วิธีการแบบเดียวกันในการจัดการ ACLs โดยไม่จำกัด ว่าประเภท ACL เชื่อมโยงกับอ็อบเจ็กต์หรือไม่ เฟรมเวิร์กประกอบด้วย คอมโพเนนต์ต่อไปนี้:

คำสั่งการดูแล ACL

คำสั่งเหล่านี้เช่น **aclget, aclput, acledit, aclconvert, aclgetttypes** คำสั่งเหล่านี้เรียกใช้ไลบรารีอินเตอร์เฟสที่เรียกใช้โมดูลเฉพาะสำหรับ ประเภท ACL

ไลบรารีอินเตอร์เฟส ACL

ไลบรารีอินเตอร์เฟส ACL ทำหน้าที่เป็นส่วนหน้าของแอ็พพลิเคชัน ที่จำเป็นต้องเข้าถึง ACLs

โมดูล ACL ที่โหลดได้แบบไดนามิกที่เฉพาะสำหรับประเภท ACL

ระบบปฏิบัติการ AIX จัดเตรียมชุดของโมดูลที่ระบุไว้สำหรับชนิด ACL สำหรับ AIX Classic ACLs (**AIXC**) และ NFS4 ACLs (**nfs4**)

ความเข้ากันได้ระดับไบนารี:

ไม่มีปัญหาในเรื่องความเข้ากันได้สำหรับแอ็พพลิเคชันที่รันอยู่บน ระบบไฟล์ JFS2 ที่มีอยู่เดิม, โดยมีหรือไม่มี AIX ACLs ที่มีอยู่เดิม

อย่างไรก็ตาม โปรดทราบว่าแอ็พพลิเคชันอาจพบว่าการเข้าถึงไฟล์ต่างๆ อาจล้มเหลวถ้าแอ็พพลิเคชันพบอ็อบเจ็กต์ระบบไฟล์ที่มี ACLs ที่มีข้อจำกัดมากกว่า (เช่น NFS4) เชื่อมโยงอยู่ ทำการตรวจสอบง่ายๆ เพื่อดูว่าไฟล์ที่มีอยู่ จะต้องการใช้ระดับสิทธิการอ่านใน NFS4 ACL หรือไม่

ชนิด Access Control List สนับสนุนบนระบบปฏิบัติการ AIX

ระบบปฏิบัติการ AIX สนับสนุนชนิด AIXC และ NFS4 ACL ในปัจจุบัน

ดังที่กล่าว ยังสนับสนุนโครงสร้างพื้นฐานสำหรับ การเพิ่มประเภท ACL อื่นใดๆ ที่สนับสนุนโดยระบบไฟล์ฟิสิคัลไฟล์ ที่จำเป็น โปรดทราบว่า JFS2 PFS สนับสนุน NFS4 ACL โดยเริ่มแรก ถ้า instance ระบบไฟล์ถูกสร้างโดยมีความเข้ากันได้กับ Extended Attributes เวอร์ชัน 2

รายการค่าควบคุมการเข้าใช้ AIXC:

ชนิด AIXC Access Control List แทนค่าลักษณะการทำงานของชนิด ACL ที่สนับสนุนบน AIX รีลีสก่อนหน้า 5.3.0 AIXC ACLs ประกอบด้วยสิทธิพื้นฐานและสิทธิเพิ่มเติม

ชนิด AIXC Access Control List (ACL) แทนค่าลักษณะการทำงานของชนิด ACL ที่สนับสนุนบน AIX รีลีสก่อนหน้า 5.3.0 AIXC ACLs ประกอบด้วยสิทธิพื้นฐานและสิทธิเพิ่มเติม ระบบไฟล์ JFS2 อนุญาตให้มีขนาดสูงสุด 4 KB สำหรับ AIXC ACLs

การตั้งค่าสิทธิพื้นฐานสำหรับ AIXC ACL

สิทธิ พื้นฐานคือโหมดการเข้าถึงไฟล์แบบดั้งเดิมที่กำหนดให้แก่ เจ้าของไฟล์ กลุ่มไฟล์ และผู้ใช้อื่น โหมดการเข้าถึงได้แก่: อ่าน (r) เขียน (w) และทำงาน/ค้นหา (x)

ใน ACL สิทธิพื้นฐาน จะอยู่ในรูปแบบต่อไปนี้ที่มีพารามิเตอร์ *Mode* แสดง เป็น rwx (โดยที่เครื่องหมายยัติภังค์ (-) แทนสิทธิแต่ละสิทธิที่ไม่ระบุ):

```
สิทธิพื้นฐาน:
   owner(name): Mode
   group(group): Mode
   others: Mode
```

การตั้งค่าแอ็ตทริบิวต์สำหรับ AIXC ACL

แอ็ตทริบิวต์ ต่อไปนี้สามารถเพิ่มใน AIXC ACL:

setuid (SUID)
บิตโหมด Set-user-ID แอ็ตทริบิวต์นี้ตั้งค่า ID ผู้ใช้ที่มีผลและ บันทึกของกระบวนการให้แก่ ID เจ้าของของไฟล์ตอนรันไทม์

setgid (SGID)
บิตโหมด Set-group-ID แอ็ตทริบิวต์นี้ตั้งค่า ID กลุ่มที่มีผลและ บันทึกของกระบวนการให้แก่ ID กลุ่มของไฟล์ตอนรันไทม์

savetext (SVTX)
สำหรับไดเร็กทอรี บ่งชี้ว่าเจ้าของไฟล์เท่านั้นที่สามารถลิงก์ หรือยกเลิกการลิงก์ ไฟล์ในไดเร็กทอรีที่ระบุ

แอ็ตทริบิวต์เหล่านี้ถูกเพิ่มในรูปแบบต่อไปนี้:

```
แอ็ตทริบิวต์: SUID, SGID, SVTX
```

การตั้งค่าสิทธิเพิ่มเติมสำหรับ AIXC Access ACL

สิทธิ เพิ่มเติมอนุญาตให้เจ้าของไฟล์สามารถกำหนดการเข้าถึงสำหรับไฟล์นั้น ได้ละเอียดมากขึ้น สิทธิเพิ่มเติมจะแก้ไขสิทธิไฟล์พื้นฐาน (เจ้าของ กลุ่ม อื่น) โดยการอนุญาต การปฏิเสธ หรือการระบุโหมด การเข้าถึงสำหรับบุคคล กลุ่ม หรือผู้ใช้และกลุ่มร่วมกันที่เจาะจง สิทธิถูกแก้ไขโดยการใช้คีย์เวิร์ด

คีย์เวิร์ด **permit, deny** และ **specify** ถูกกำหนดดังนี้:

permit ให้สิทธิการเข้าถึงที่เจาะจงเพื่อใช้ไฟล์แก่ผู้ใช้หรือกลุ่ม

deny จำกัดผู้ใช้หรือกลุ่มจากการใช้การเข้าถึงที่เจาะจง เพื่อใช้ไฟล์

specify กำหนดการเข้าถึงไฟล์อย่างละเอียดสำหรับผู้ใช้หรือกลุ่ม

ถ้าผู้ใช้ถูกปฏิเสธการเข้าถึงที่เจาะจงโดยคีย์เวิร์ด **deny** หรือ **specify** จะไม่มีรายการอื่นใดที่สามารถแทนที่การปฏิเสธ การเข้าถึงได้

คีย์เวิร์ด **enabled** ต้องถูกระบุใน ACL เพื่อให้สิทธิเพิ่มเติมมีผลใช้ได้ ค่าดีฟอลต์ คือคีย์เวิร์ด **disabled**

ใน ACL สิทธิเพิ่มเติม อยู่ในรูปแบบต่อไปนี้:

```
สิทธิเพิ่มเติม:
  enabled | disabled
    permit   Mode  UserInfo...
    deny     Mode  UserInfo...
    specify  Mode  UserInfo...
```

ใช้ บรรทัดแยกแต่ละรายการ **permit, deny** หรือ **specify** พารามิเตอร์ *Mode* แสดงเป็น **rwx** (โดย ที่เครื่องหมายยัติภังค์ (-) แทนสิทธิแต่ละสิทธิที่ไม่ระบุ) พารามิเตอร์ *UserInfo* แสดงเป็น `u:UserName` หรือ `g:GroupName` หรือการผสม `u:UserName` และ `g:GroupName` โดยคั่นด้วยจุลภาค

หมายเหตุ: เนื่องจากกระบวนการมีหนึ่ง ID ผู้ใช้เท่านั้น ถ้ามีชื่อผู้ใช้ มากกว่าหนึ่งชื่อถูกระบุในรายการ จะไม่สามารถใช้รายการนั้นได้ใน การกำหนดการควบคุมการเข้าถึง

การแทนค่าข้อความของ AIXC ACL

stanza ต่อไปนี้แสดงการแทนค่าข้อความของ AIXC ACL:

```
Attributes:  { SUID | SGID | SVTX }
Base Permissions:
   owner(name): Mode
   group(group): Mode
   others: Mode
Extended Permissions:
   enabled | disabled
     permit   Mode  UserInfo...
     deny     Mode  UserInfo...
     specify  Mode  UserInfo...
```

รูปแบบไบนารีของ AIXC ACL

รูปแบบไบนารี AIXC ACL ถูกนิยามอยู่ใน `/usr/include/sys/acl.h` และถูกใช้ในรีลีส AIX ปัจจุบัน

ตัวอย่าง AIXC ACL

ต่อไปนี้คือตัวอย่าง ของ AIXC ACL:

```
attributes: SUID
base permissions:
      owner(frank):  rw-
      group(system): r-x
      others: ---
extended permissions:
      enabled
        permit  rw-  u:dhs
        deny    r--  u:chas, g:system
        specify r--  u:john, g:gateway, g:mail
        permit  rw-  g:account, g:finance
```

รายการ ACL ถูกแสดงดังนี้:

- บรรทัดแรกบ่งชี้ว่าบิต **setuid** ถูก เปิดทำงาน
- บรรทัดถัดไป ซึ่งแสดงสิทธิพื้นฐาน เป็นทางเลือก
- สามบรรทัดถัดไประบุสิทธิพื้นฐาน ชื่อเจ้าของและ กลุ่มในวงเล็บมีเพื่อเป็นข้อมูลเท่านั้น การเปลี่ยนชื่อ เหล่านี้จะไม่มีผล เปลี่ยนแปลงเจ้าของไฟล์ หรือกลุ่มไฟล์ เฉพาะคำสั่ง **chown** และคำสั่ง **chgrp** เท่านั้นที่สามารถเปลี่ยนแปลงไฟล์แอ็ตทริบิวต์เหล่านี้
- บรรทัดถัดไป ซึ่งแสดงสิทธิเพิ่มเติม เป็นทางเลือก
- บรรทัดถัดไปบ่งชี้ว่าสิทธิที่เพิ่มตามมา ถูกเปิดใช้งาน
- สี่บรรทัดสุดท้ายคือรายการที่เพิ่ม รายการที่เพิ่มอันแรก ให้ผู้ใช้ *dhs* มีสิทธิอ่าน (r) และเขียน (w) ในไฟล์
- รายการที่เพิ่มรายการที่สองปฏิเสธการเข้าถึงเพื่ออ่าน (r) สำหรับผู้ใช้ *chas* เท่านั้น เมื่อผู้ใช้เป็นสมาชิกของกลุ่ม *system*
- รายการที่เพิ่มรายการที่สามระบุว่าตราบใดที่ผู้ใช้ *john* เป็นสมาชิกของทั้งกลุ่ม *gateway* และกลุ่ม *mail* ผู้ใช้จะสามารถเข้าถึงเพื่ออ่าน (r) ถ้าผู้ใช้ *john* มิได้เป็นสมาชิกของทั้งสอง กลุ่ม สิทธิที่เพิ่มนี้จะไม่มีผลใช้
- รายการที่เพิ่มรายการสุดท้ายให้สิทธิผู้ใช้ใดๆ ใน *ทั้งสอง* กลุ่ม *account* และกลุ่ม *finance* มีสิทธิการอ่าน (r) และเขียน (w)

หมายเหตุ: รายการที่เพิ่มมากกว่าหนึ่งรายการสามารถมีผลใช้กับกระบวนการที่ กำลังร้องขอการเข้าถึงอ็อบเจ็กต์ที่ควบคุมได้ ด้วยรายการที่จำกัด ที่มีการบังคับใช้เหนือโหมดการอนุญาต

สำหรับไวยากรณ์โดยสมบูรณ์ทั้งหมด ดูที่คำสั่ง **acledit** ใน *การอ้างอิงคำสั่ง*

รายการค่าควบคุมการเข้าใช้ NFS4:

AIX ยังสนับสนุนชนิด NFS4 Access Control List (ACL)

ประเภท NFS4 ACL ประยุกต์ใช้การควบคุมการเข้าถึงดังระบุในโปรโตคอล *Network File System (NFS) เวอร์ชัน 4 RFC 3530* ระบบไฟล์ JFS2 อนุญาตให้มีขนาดสูงสุด 64KB สำหรับ NFS4 ACLs

เฉพาะไคลเอ็นต์ NFS V4 สนับสนุน NFS V4 ACL ทั้ง Cachefs และ Proxy ไม่สนับสนุน NFS V4 ACL

การแทนค่าข้อความของ NFS4 ACL

NFS V4 ACL แบบข้อความคือรายการของ ACEs (Access Control Entries) โดยแต่ละ ACE ต่อ หนึ่งบรรทัด ACE มีองค์ประกอบสี่ส่วนในรูปแบบต่อไปนี้

```
IDENTITY   ACE_TYPE      ACE_MASK     ACE_FLAGS

โดยที่:
IDENTITY => มีรูปแบบของ 'IDENTITY_type:(IDENTITY_name หรือ IDENTITY_ID หรือ IDENTITY_who):'
         โดยที่:
         IDENTITY_type => ประเภท Identity หนึ่งในต่อไปนี้:
                  u : ผู้ใช้
                  g : กลุ่ม
                  s : สตริง who พิเศษ (IDENTITY_who ต้องเป็น who พิเศษ)
                          IDENTITY_name => ชื่อผู้ใช้/กลุ่ม
                          IDENTITY_ID   => ID ผู้ใช้/กลุ่ม
                          IDENTITY_who  => สตริง who พิเศษ (เช่น OWNER@, GROUP@, EVERYONE@)
ACE_TYPE => ประเภท ACE หนึ่งต่อไปนี้:
                a : allow
                d : deny
                l : alarm
                u : audit
ACE MASK => หนึ่งหรือหลายค่าของ Key ค่า Mask ต่อไปนี้โดยไม่มีตัวคั่น:
                r : READ_DATA   หรือ LIST_DIRECTORY
                w : WRITE_DATA      หรือ ADD_FILE
                p : APPEND_DATA     หรือ ADD_SUBDIRECTORY
                R : READ_NAMED_ATTRS
                W : WRITE_NAMED_ATTRS
                x : EXECUTE         หรือ SEARCH_DIRECTORY
                D : DELETE_CHILD
                a : READ_ATTRIBUTES
                A : WRITE_ATTRIBUTES
                d : DELETE
                c : READ_ACL
                C : WRITE_ACL
                o : WRITE_OWNER
                s : SYNCHRONIZE
ACE_FLAGS (ทางเลือก) => หนึ่งหรือหลายค่าของ Attribute Key ต่อไปนี้โดยไม่มีตัวคั่น:
                fi : FILE_INHERIT
                di : DIRECTORY_INHERIT
```

```
oi : INHERIT_ONLY
ni : NO_PROPAGATE_INHERIT
sf : SUCCESSFUL_ACCESS_ACE_FLAG
ff : FAILED_ACCESS_ACE_FLAG
```

หมายเหตุ: ส่วนที่เกี่ยวกับคีย์ค่า SYNCHRONIZE Ace_Mask s, AIX ไม่มีการดำเนินการใดๆ ที่เกี่ยวกับคีย์ค่านี้ ระบบปฏิบัติการ AIX เก็บและสงวนคีย์ค่า s แต่คีย์ค่านี้ไม่มีความหมายกับ AIX

เมื่อ WRITE_OWNER Ace_Mask ถูกตั้งค่าเป็น Ace_Type `allow` ผู้ใช้สามารถเปลี่ยนความเป็นเจ้าของของไฟล์ไปเป็นตนเองเท่านั้น

การลบ ไฟล์ที่ขึ้นกับ ACEs สองรายการ ได้แก่รายการ DELETE ของอ็อบเจ็กต์จะถูกลบ และรายการ DELETE_CHILD ของพาเรนต์ไดเร็กทอรี ระบบปฏิบัติการ AIX จัดเตรียมสองโหมดของลักษณะการทำงานให้แก่ผู้ใช้ ในโหมด *ปลอดภัย* DELETE ทำงานคล้ายกับ AIXC ACLs ใน โหมด *ความเข้ากันได้* DELETE ทำงานคล้ายการใช้งานหลักอื่นๆ ของ NFS4 ACLs ในการเปิดใช้โหมดความเข้ากันได้ ใช้คำสั่ง **chdev** ดังนี้:

```
chdev -l sys0 -a nfs4_acl_compat=compatible
```

คุณ ต้องบูตระบบใหม่หลังจากรันคำสั่ง **chdev** ก่อนที่การเปลี่ยนแปลงการตั้งค่าจะมีผล

ถ้า คุณสับเปลี่ยนระบบของคุณระหว่างสองโหมด, คุณต้องทราบว่า NFS4 ACLs ที่สร้างขึ้นโดยระบบปฏิบัติการ AIX ในโหมดความปลอดภัยอาจไม่ยอมรับแพลตฟอร์มอื่นแม้ว่าระบบ เปลี่ยนกลับไปเป็นโหมดที่มีความเข้ากันได้

ตัวอย่าง:

```
        u:user1(aa@ibm.com):     a    rwp    fidi
        *s:(OWNER@):             d    x      dini          * This line is a comment
        g:staff(jj@jj.com):      a    rx
        s:(GROUP@):              a    rwpx   fioi
        u:2:                     d    r     di             * This line shows user bin (uid=2)
        g:7:                     a    ac    fi             * This line shows group security (gid=7)
        s:(EVERYONE@):           a    rca   ni
```

รูปแบบฐานสองสำหรับ NFS4 ACL

รูปแบบไบนารี NFS4 ACL ถูกนิยามอยู่ใน `/usr/include/sys/acl.h` และถูกใช้ในรีลีส AIX ปัจจุบัน

ตัวอย่าง NFS4 ACL

ตัวอย่าง ต่อไปนี้แสดง NFS4 ACL ที่นำไปใช้บนไดเร็กทอรี (เช่น `j2eav2/d0`):

```
s:(OWNER@):         a         rwpRWxDdo         difi         * 1st   ACE
s:(OWNER@):         d         D                 difi         * 2nd  ACE
s:(GROUP@):         d         x                 ni           * 3rd   ACE
s:(GROUP@):         a         rx                difi         * 4th   ACE
s:(EVERYONE@):      a         c                 difi         * 5th   ACE
s:(EVERYONE@):      d         C                 difi         * 6th   ACE
u:user1:            a         wp                oi           * 7th   ACE
g:grp1:             d         wp                             *  8th  ACE
u:101:              a         C                              * 9th   ACE
g:100:              d         c                              * 10th  ACE
```

รายการ ACL ถูกแสดงดังต่อไปนี้:

- ACE แรกบ่งชี้ว่าเจ้าของมีสิทธิต่อไปนี้บน /j2eav2/d0 และลูกหลานที่สร้างขึ้น หลังจาก ACL นี้ถูกนำไปใช้:
  - READ_DATA (= LIST_DIRECTORY)
  - WRITE_DATA (=ADD_FILE)
  - APPEND_DATA (= ADD_SUBDIRECTORY)
  - READ_NAMED_ATTR
  - WRITE_NAMED_ATTR
  - EXECUTE (=SEARCH_DIRECTORY)
  - DELETE_CHILD
  - DELETE
  - WRITE_OWNER
- ACE ที่สองบ่งชี้ว่าเจ้าของถูกปฏิเสธสิทธิสำหรับ DELETE_CHILD (การลบ ไฟล์หรือไดเร็กทอรีย่อยที่สร้างขึ้นภายใต้ /j2eav2) แต่เจ้าของยังคงสามารถลบได้เนื่องจาก ACE แรก ซึ่งอนุญาตให้ เจ้าของมีสิทธิในการ DELETE_CHILD
- ACE ที่สามบ่งชี้ว่าสมาชิกทั้งหมดของกลุ่มสำหรับอ็อบเจ็กต์ (/j2eav2/d0) ถูกปฏิเสธสิทธิสำหรับ EXECUTE (=SEARCH_DIRECTORY) แต่เจ้าของยังคงได้รับอนุญาตให้ มีสิทธิโดย ACE แรก ACE นี้ไม่สามารถกระจายไปยังลูกหลานทั้งหมดได้เนื่องจากแฟล็ก NO_PROPAGATE_INHERIT ถูกระบุ ACE นี้ถูกนำใช้เฉพาะกับไดเร็กทอรี /j2eav2/d0 และไฟล์และไดเร็กทอรีย่อยชายด์ที่ติดกันเท่านั้น
- ACE ที่สี่บ่งชี้ว่าสมาชิกทั้งหมดของกลุ่มของ อ็อบเจ็กต์ (/j2eav2/d0) ได้รับอนุญาตให้มีสิทธิ สำหรับ READ_DATA (= LIST_DIRECTORY) และ EXECUTE (=SEARCH_DIRECTORY) บน /j2eav2/d0 และ ลูกหลานทั้งหมด อย่างไรก็ตาม เนื่องจากสมาชิกกลุ่ม ACE ที่สาม (ยกเว้น เจ้าของ) ไม่ได้รับอนุญาตให้มีสิทธิสำหรับ EXECUTE (=SEARCH_DIRECTORY) บนไดเร็กทอรี /j2eav2/d0 และไฟล์และไดเร็กทอรีย่อยชายด์ ที่ติดกัน
- ACE ที่ห้าบ่งชี้ว่าทุกคนได้รับอนุญาตให้มีสิทธิ สำหรับ READ_ACL บนไดเร็กทอรี /j2eav2/d0 และลูกหลานใดๆ ที่ถูกสร้างขึ้นหลังจาก ACL นี้ถูกนำไปใช้
- ACE ที่หกบ่งชี้ว่าทุกคนถูกปฏิเสธสิทธิ สำหรับ WRITE_ACL บนไดเร็กทอรี /j2eav2/d0 และลูกหลานใดๆ เจ้าของมีสิทธิสำหรับ WRITE_ACL เสมอบน ไฟล์และไดเร็กทอรีที่มี NFS4 ACLs
- ACE ที่เจ็ดบ่งชี้ว่า user1 มีสิทธิสำหรับ WRITE_DATA (=ADD_FILE) และ APPEND_DATA (= ADD_SUBDIRECTORY) บน ลูกหลานทั้งหมดของไดเร็กทอรี /j2eav2/d0 แต่ไม่ใช่บนไดเร็กทอรี /j2eav2/d0 เอง
- ACE ที่แปดบ่งชี้ว่าสมาชิกทั้งหมดของ grp1 ถูกปฏิเสธ สิทธิสำหรับ WRITE_DATA (=ADD_FILE) และ APPEND_DATA (= ADD_SUBDIRECTORY) ACE นี้ไม่มีผลใช้กับเจ้าของ แม้ว่าจะอยู่ใน grp1 เนื่องจาก ACE แรก
- ACE ที่เก้าบ่งชี้ว่าผู้ใช้ที่มี **UID 101** มี สิทธิสำหรับ WRITE_ACL แต่ไม่มีคนใด ยกเว้น เจ้าของที่มีสิทธิสำหรับ WRITE_ACL เนื่องจาก ACE ที่หก
- ACE ที่สิบบ่งชี้ว่าสมาชิกทั้งหมดของกลุ่มที่มี **GID 100** ถูกปฏิเสธสำหรับ READ_ACL แต่จะมี สิทธินี้เนื่องจาก ACE ที่เก้า

การจัดการ Access Control List

คุณสามารถใช้คำสั่งเพื่อดูและตั้งค่า ACLs

แอ็พพลิเคชันโปรแกรมเมอร์และผู้พัฒนาระบบย่อยอื่นๆ สามารถใช้ ไลบรารีอินเตอร์เฟส ACL และรูทีนการแปลง ACL ดังอธิบายใน ส่วนนี้

คำสั่งการจัดการ ACL

คุณสามารถใช้ คำสั่งต่อไปนี้เพื่อทำงานกับ ACLs สำหรับอ็อบเจ็กต์ระบบไฟล์:

aclget เขียน ACL ของไฟล์อ็อบเจ็กต์ชื่อ *FileObject* ไปยังเอาต์พุตมาตรฐาน โดยแสดงในรูปแบบที่อ่านได้ หรือเขียนในแบบเดียวกันไปยังเอาต์พุตไฟล์ ชื่อ *outAclFile*

aclput ตั้งค่า ACL ของ *FileObject* บนระบบไฟล์โดยใช้ อินพุตที่ระบุผ่านอินพุตมาตรฐานหรือ *inAclFile*

acledit เปิดเอดิเตอร์เพื่อทำการแก้ไข ACL ของ *FileObject* ที่ระบุ

aclconvert
แปลง ACL จากประเภทหนึ่งไปเป็นอีกประเภทหนึ่ง คำสั่งนี้จะล้มเหลว ถ้าไม่มีการสนับสนุนการแปลง

aclgettypes
รับค่าประเภท ACL ที่พาธระบบไฟล์สนับสนุน

ไลบรารีอินเตอร์เฟส ACL

ไลบรารีอินเตอร์เฟส ACL ทำหน้าที่เป็นส่วนหน้าของแอ็พพลิเคชันที่จำเป็นต้องเข้าถึง ACLs แอ็พพลิเคชัน (รวมถึงคำสั่งการจัดการ ACL ทั่วไปที่กำหนด ด้านบน) จะไม่เรียกใช้ ACL syscalls โดยตรง แต่ เข้าถึง syscalls ทั่วไปและโมดูลที่โหลดได้เฉพาะประเภท ผ่านไลบรารีอินเตอร์เฟส วิธีนี้จะเป็นการป้องกันโปรแกรมเมอร์แอ็พพลิเคชันลูกค้า จากความซับซ้อนของการใช้โมดูลที่สามารถโหลดได้, และลดปัญหาความเข้ากันได้ของ ไบนารีแบบย้อนหลังสำหรับรีลีส AIX ในอนาคต

ไลบรารีอินเตอร์เฟสต่อไปนี้ เรียกใช้ syscalls

aclx_fget and aclx_get
ฟังก์ชัน **aclx_get** และ **aclx_fget** เรียกค้น ข้อมูลการควบคุมการเข้าถึงสำหรับอ็อบเจ็กต์ระบบไฟล์ และนำไปไว้ในส่วนพื้นที่หน่วยความจำที่ระบุโดย **acl** ข้อมูลขนาดและประเภท สำหรับ **acl** ถูกเก็บใน ***acl_sz** และ ***acl_type**

aclx_fput and aclx_put
ฟังก์ชัน **aclx_put** และ **aclx_fput** เก็บค่าข้อมูลการควบคุม การเข้าถึงที่ระบุใน **acl** สำหรับอ็อบเจ็กต์ไฟล์อินพุต ฟังก์ชันเหล่านี้ไม่ทำการแปลงประเภท ACL สำหรับการแปลงประเภท ACL ผู้เรียกใช้ต้องเรียกใช้ฟังก์ชัน **aclx_convert** อย่างชัดเจน

aclx_gettypes
ฟังก์ชัน **aclx_gettypes** รับค่ารายการของประเภท ACL ที่สนับสนุน บนระบบไฟล์เฉพาะ ประเภทระบบไฟล์สามารถสนับสนุนได้พร้อมกันมากกว่าหนึ่ง ประเภท ACL แต่ละอ็อบเจ็กต์ระบบไฟล์ที่เชื่อมโยงกับ ประเภท ACL เฉพาะเป็นสมาชิกของรายการของประเภท ACL ที่สนับสนุน โดยระบบไฟล์

aclx_gettypeinfo
ฟังก์ชัน **aclx_gettypeinfo** รับค่าคุณสมบัติและ ความสามารถของประเภท ACL บนระบบไฟล์ที่ระบุโดยพาธ โปรดทราบว่าคุณสมบัติ ACL โดยปกติจะเป็นประเภทโครงสร้างของ ซึ่งเป็นข้อมูลเฉพาะสำหรับแต่ละประเภท ACL ที่เจาะจง โครงสร้างข้อมูล ที่ใช้สำหรับ AIXC และ NFS4 ACLs จะอธิบายในเอกสาแยก

aclx_print และ aclx_printStr
สองฟังก์ชันนี้จะแสดง ACL ที่กำหนดในรูปแบบไบนารีไปเป็น การแสดงแบบข้อความ ฟังก์ชันเหล่านี้ถูกเรียกใช้โดยคำสั่ง **aclget** และ **acledit**

aclx_scan และ aclx_scanStr
สองฟังก์ชันนี้แปลงการแสดงด้วยข้อความของ ACL ไปเป็นรูปแบบไบนารี

aclx_convert
แปลง ACL จากประเภทหนึ่งไปเป็นอีกประเภทหนึ่ง ฟังก์ชันนี้ใช้ สำหรับการแปลงโดยนัยโดยใช้คำสั่ง เช่น **cp, mv** หรือ **tar**

การแปลง ACL

การแปลง ACL อนุญาตให้คุณ แปลงประเภท ACL หนึ่งไปเป็นอีกประเภทหนึ่ง การสนับสนุนประเภท ACL หลายประเภทขึ้นอยู่กับประเภท ACL ที่ได้รับการสนับสนุนบนระบบฟิสิคัลไฟล์ ที่เจาะจง ระบบไฟล์ทั้งหมดไม่สนับสนุนทุกประเภทของ ACL ตัวอย่าง ระบบไฟล์หนึ่งอาจสนับสนุนเฉพาะประเภท AIXC ACL และระบบไฟล์ สองอาจสนับสนุนประเภท AIXC และ NFS4 ACL คุณสามารถทำสำเนา AIXC ACLs ระหว่างระบบไฟล์สองระบบ แต่คุณต้องใช้การแปลง ACL เพื่อทำสำเนา NFS ACLs จากระบบไฟล์สองไปยังระบบไฟล์หนึ่ง การแปลง ACL จะปกป้องข้อมูลการควบคุมการเข้าถึงมากที่สุดเท่าที่เป็นไปได้

หมายเหตุ: กระบวนการแปลงใกล้เคียงและอาจส่งผล ให้สูญเสียข้อมูลการควบคุมการเข้าถึง คุณควรพิจารณาสิ่งนี้เมื่อ วางแผนทำการแปลง ACL ของคุณ

การแปลง ACL ในระบบปฏิบัติการ AIX ถูกสนับสนุนด้วยโครงสร้างพื้นฐานต่อไปนี้:

ไลบรารีรูทีน
รูทีนและเฟรมเวิร์ก ACL ระดับผู้ใช้เหล่านี้เปิดใช้การแปลง ACL จากประเภท ACL หนึ่งไปเป็นอีกประเภทหนึ่ง

คำสั่ง aclconvert
คำสั่งนี้แปลง ACLs

คำสั่ง aclput และ acledit
คำสั่งเหล่านี้ถูกใช้แก้ไขประเภท ACL

คำสั่ง cp และ mv
คำสั่งเหล่านี้ได้ถูกเปิดใช้งานเพื่อจัดการประเภท ACL หลายประเภท และดำเนินการแปลง ACL ภายในใดๆ ที่จำเป็น

คำสั่ง backup
คำสั่งนี้แปลงข้อมูล ACL ไปเป็นประเภทที่ทราบและ จาก (ประเภท AIXC ACL) ถ้าถูกร้องขอให้ทำการสำรองข้อมูลในรูปแบบเก่า ในการเรียกข้อมูล ACL ในรูปแบบดั้งเดิม ให้ระบุอ็อพชัน **-U** ดูที่ สำรองข้อมูล สำหรับข้อมูลเพิ่มเติม

แต่ละประเภท ACL จะเป็นค่าเฉพาะ และการแบ่งละเอียดของมาสก์ค่าควบคุมการเข้าใช้ แตกต่างกันอย่างมากจากประเภทหนึ่งไปยังอีกประเภทหนึ่ง อัลกอริทึมการแปลง เป็นการประมาณการ และไม่เท่ากับการแปลง ACL ด้วยตนเอง ในบางกรณีการแปลงจะไม่ได้ค่าที่แน่นอน ตัวอย่าง NFS4 ACLs ไม่สามารถถูกแปลงเป็น AIXC ACLs ได้อย่างแม่นยำ เนื่องจาก NFS4 ACLs มี สูงสุด 16 มาสก์การเข้าถึง และมีคุณลักษณะการสืบทอดที่ไม่ได้รับการสนับสนุน ในประเภท AIXC ACL) คุณไม่ควรใช้โปรแกรมช่วยการแปลง ACL และอินเตอร์เฟสถ้าคุณต้องกังวลเกี่ยวกับการสูญเสียข้อมูลการควบคุม การเข้าถึง

หมายเหตุ: อัลกอริทึมการแปลง ACL โดยทั่วไปมีเจ้าของ และอาจเปลี่ยนแปลงได้

บิต S และ Access Control Lists

คุณสามารถใช้โปรแกรม **setuid** และ **setgid** และ การใช้บิต S กับ ACLs

การใช้โปรแกรม setuid และ setgid

กลไก บิตสิทธิอนุญาตให้มีการควบคุมการเข้าถึงสำหรับรีซอร์สใน สถานการณ์ส่วนใหญ่ แต่สำหรับการควบคุมการเข้าถึงที่มีความสำคัญมากขึ้น ระบบปฏิบัติการ จะจัดให้มีโปรแกรม **setuid** และ **setgid**

ระบบปฏิบัติ AIX นิยาม identity เฉพาะในรูปของ uids และ gids ชนิด ACL ที่ไม่นิยาม identity ที่มี uids และ gids ถูกแม็พกับโมเดล identity AIX ตัวอย่าง ประเภท NFS4 ACL กำหนด identity ผู้ใช้เป็นสตริง ในรูปแบบ user@domain และสตริงนี้ถูกแม็พกับ UIDs และ GIDs ที่เป็นตัวเลข

โปรแกรมส่วนใหญ่รันโดยใช้สิทธิ์การเข้าถึง ผู้ใช้และกลุ่มของผู้ใช้ที่เรียกใช้โปรแกรม เจ้าของโปรแกรมสามารถเชื่อมโยง สิทธิ์การเข้าถึงของผู้ใช้ที่เรียกใช้ โดยจัดทำโปรแกรมให้เป็น โปรแกรม **setuid** หรือ **setgid** นั่นคือ โปรแกรมที่มี การตั้งค่าบิต setuid หรือ setgid ในฟิลด์สิทธิ์ของตน เมื่อโปรแกรมนั้น ถูกรันโดยกระบวนการ กระบวนการจะได้รับสิทธิ์การเข้าถึงของ เจ้าของโปรแกรม โปรแกรม **setuid** รันด้วยสิทธิ์การเข้าถึง ของเจ้าของ ขณะนี้โปรแกรม **setgid** มีสิทธิ์การเข้าถึง ของกลุ่ม และทั้งสองบิตสามารถถูกตั้งค่าตามกลไก สิทธิการใช้งาน

แม้กระบวนการถูกกำหนดสิทธิ์การเข้าถึง เพิ่มเติม สิทธิ์เหล่านี้จะถูกควบคุมโดยโปรแกรมที่ถือครอง สิทธิ์ ดังนั้น โปรแกรม **setuid** และ **setgid** อนุญาต สำหรับการควบคุมการเข้าถึงที่ผู้ใช้โปรแกรมซึ่งเป็นการให้สิทธิ์การเข้าถึง ทางอ้อม โปรแกรมทำหน้าที่เป็นระบบย่อยที่ไว้วางใจ ให้การปกป้อง สิทธิการเข้าถึงของผู้ใช้

แม้โปรแกรมเหล่านี้จะสามารถใช้ได้อย่างมีประสิทธิภาพ แต่ก็มีความเสี่ยงด้านความปลอดภัยถ้าไม่ได้รับการกำหนด อย่างระมัดระวัง โดยเฉพาะ โปรแกรมต้องไม่ส่งการควบคุม ไปยังผู้ใช้ขณะที่โปรแกรมยังคงมีสิทธิ์การเข้าถึงของเจ้าของ เนื่องจากอาจเป็นการอนุญาตให้ผู้ใช้ใช้สิทธิ์ที่ไม่จำกัดของสิทธิ์ของเจ้าของ

หมายเหตุ: เพื่อเหตุผลด้านความปลอดภัย ระบบปฏิบัติการไม่สนับสนุนการเรียกใช้โปรแกรม **setuid** หรือ **setgid** ภายในเชลล์สคริปต์

การนำใช้บิต S กับ ACLs

ACLs เช่น NFS4 ไม่ได้จัดการกับบิต S โดยตรง NFS4 ACL ไม่ได้ระบุวิธี ที่บิตเหล่านี้สามารถใช้เป็นส่วนหนึ่งของ ACL ระบบปฏิบัติการ AIX ประสบปัญหาที่บิต S ซึ่งจะถูกใช้ขณะดำเนินการเข้าถึง การตรวจสอบและจะส่งเสริม NFS4 ACL ใดๆ ที่เกี่ยวข้องกับการตรวจสอบการเข้าถึง คำสั่ง **chmod** ที่จัดเตรียมไว้พร้อมกับระบบปฏิบัติการ AIX สามารถใช้เพื่อตั้งค่าหรือรีเซ็ตบิต S บนอ็อบเจ็กต์ระบบไฟล์ที่มี ACLs เช่น NFS4

สิทธิการเข้าถึงเพื่อการดูแล

ระบบปฏิบัติการจัดให้มีสิทธิการเข้าถึงพิเศษ สำหรับการดูแลระบบ

สิทธิพิเศษของระบบจะอิงตาม ID ผู้ใช้และกลุ่ม ผู้ใช้ที่มี ID ผู้ใช้ที่มีผล หรือกลุ่มเป็น 0 จะถูกจัดเป็นมีสิทธิพิเศษ

กระบวนการที่มี ID ผู้ใช้ที่มีผลเป็น 0 คือกระบวนการผู้ใช้ root และสามารถ:

- อ่านหรือเขียนอ็อบเจ็กต์ใดๆ
- เรียกใช้ฟังก์ชันระบบใดๆ
- ดำเนินการควบคุมระบบย่อยโดยการเรียกใช้งานโปรแกรม **setuid-root**

คุณสามารถจัดการระบบได้โดยใช้สิทธิพิเศษสองประเภท: สิทธิพิเศษคำสั่ง **su** และสิทธิพิเศษโปรแกรม **setuid-root** คำสั่ง **su** อนุญาตให้โปรแกรมทั้งหมดที่คุณเรียกใช้ทำงานเป็นกระบวนการผู้ใช้ root คำสั่ง **su** เป็นวิธีที่ยืดหยุ่นในการจัดการ ระบบ แต่ก็ยังปลอดภัยไม่มาก

การทำให้โปรแกรมอยู่ในโปรแกรม **setuid-root** หมายความว่าโปรแกรม เป็นโปรแกที่ผู้ใช้ root เป็นเจ้าของที่มีชุดบิต setuid โปรแกรม **setuid-root** จัดให้มีฟังก์ชันการดูแลที่ผู้ใช้ทั่วไปสามารถดำเนินการ โดยไม่ต้องเกรงเรื่องความปลอดภัย สิทธิพิเศษจะถูกป้องกันไว้ใน โปรแกรมแทนการให้สิทธิโดยตรงแก่ผู้ใช้ อาจเป็นการยาก ที่จะป้องกันฟังก์ชันการดูแลที่จำเป็นทั้งหมดใน โปรแกรม **setuid-root** แต่ช่วยให้มีความปลอดภัยมากยิ่งขึ้นแก่ผู้จัดการระบบ

การอนุญาตเข้าถึง

เมื่อผู้ใช้ล็อกอินเข้าสู่บัญชีผู้ใช้ (โดยใช้คำสั่ง **login** หรือ **su**) ID ผู้ใช้ และ ID กลุ่มที่กำหนดให้แก่บัญชีผู้ใช้ชื่อนั้น จะถูกเชื่อมโยงเข้ากับกระบวนการของผู้ใช้ ID เหล่านี้ใช้พิจารณาสิทธิ การเข้าถึงของกระบวนการ

กระบวนการที่มี ID ผู้ใช้เป็น 0 คือ *กระบวนการของผู้ใช้ root* โดยทั่วไปกระบวนการเหล่านี้อนุญาตให้มีสิทธิการเข้าถึงทั้งหมด แต่ ถ้าการร้องขอกระบวนการผู้ใช้ root เรียกใช้งานสิทธิสำหรับโปรแกรม สิทธิการเข้าถึงถูกให้อย่างน้อย หนึ่งผู้ใช้

การอนุญาตเข้าถึงสำหรับ AIXC ACLs

เจ้าของ แหล่งข้อมูลเป็นผู้มีหน้าที่จัดการสิทธิการเข้าถึง รีซอร์สได้รับการป้องกันโดย *บิตสิทธิการใช้งาน* ซึ่งถูกรวมใน โหมดของอ็อบเจ็กต์ บิตสิทธิการใช้งานกำหนดสิทธิการเข้าถึง ที่ให้แก่เจ้าของอ็อบเจ็กต์ กลุ่มของอ็อบเจ็กต์ และสำหรับ คลาสดีฟอลต์ อื่นๆ ระบบปฏิบัติการสนับสนุน โหมดการเข้าถึงที่แตกต่างกันสามโหมด (อ่าน เขียน และ เรียกใช้งาน) ที่สามารถให้สิทธิแยกกันได้

สำหรับไฟล์ ไดเร็กทอรี ไพพ์ที่มีชื่อ และอุปกรณ์ (ไฟล์พิเศษ) ซึ่งการเข้าถึงได้รับอนุญาตดังนี้:

- สำหรับแต่ละรายการค่าควบคุมการเข้าใช้ (ACE) ใน ACL รายการ identifier ถูกเปรียบเทียบกับ identifiers ของกระบวนการ ถ้ามีรายการ ที่ตรงกัน กระบวนการจะได้รับสิทธิและข้อจำกัดที่กำหนด สำหรับรายการนั้น การรวมกันแบบโลจิคัลสำหรับทั้งสิทธิและข้อจำกัด จะถูกคำนวณสำหรับรายการที่ตรงกันแต่ละรายการใน ACL ถ้าการประมวลผล การร้องขอไม่ตรงกับรายการใดๆ ใน ACL จะได้รับ สิทธิและข้อจำกัดของรายการดีฟอลต์
- ถ้าโหมดการเข้าถึงที่ร้องขอได้รับอนุมัติ (รวมในการรวม ของสิทธิ) และไม่ถูกจำกัด (รวมในการรวมของ ข้อจำกัด) การเข้าถึงจะได้รับอนุญาต มิฉะนั้น การเข้าถึงถูกปฏิเสธ

รายการ identifier ของ ACL จับคู่การประมวลผล ถ้า identifiers ทั้งหมดในรายการตรงกับประเภทที่สัมพันธ์ของ identifier ทีมีผลสำหรับ การประมวลผลที่ร้องขอ identifier ประเภท USER จะตรงถ้าเท่ากับ ID ผู้ใช้ที่ใช้งานอยู่ของการประมวลผล และ identifier ประเภท GROUP ตรงถ้าเท่ากับ ID กลุ่มที่ใช้งานอยู่ของการประมวลผล หรือ เท่ากับ ID กลุ่มเพิ่มเติมค่าหนึ่ง ตัวอย่างเช่น ACE ที่มี รายการ identifier ดังเช่นต่อไปนี้:

```
USER:fred, GROUP:philosophers, GROUP:software_programmer
```

จะ จับคู่กระบวนการกับ ID ผู้ใช้ที่ใช้งานอยู่ของ *fred* และ ชุดกลุ่มของ:

```
philosophers, philanthropists, software_programmer, doc_design
```

แต่ จะไม่จับคู่กระบวนการที่มี ID ผู้ใช้ที่ใช้งานอยู่ของ *fred* และ ชุดกลุ่มของ:

```
philosophers, iconoclasts, hardware_developer, graphic_design
```

โปรด ทราบว่า ACE ที่มีรายการ identifier ของค่าต่อไปนี้จะจับคู่สำหรับ ทั้งสองกระบวนการ:

```
USER:fred, GROUP:philosophers
```

หรือ อีกนัยหนึ่ง รายการ identifier ในฟังก์ชัน ACE คือชุดของ เงื่อนไขที่ต้องเก็บรักษาไว้เพื่อให้การเข้าถึงที่ระบุได้รับอนุญาต

สิทธิ การเข้าถึงทั้งหมดที่ตรวจสอบสำหรับอ็อบเจ็กต์เหล่านี้ถูกดำเนินการที่ระดับ การเรียกใช้ระบบเมื่ออ็อบเจ็กต์ถูกเข้าถึงเป็นครั้งแรก เนื่องจากอ็อบเจ็กต์ System V Interprocess Communication (SVIPC) ถูกเข้าถึงแบบไม่มีการแสดงสถานะ การตรวจสอบถูก ดำเนินการสำหรับทุกการเข้าถึง สำหรับอ็อบเจ็กต์ที่มีชื่อระบบไฟล์ จำเป็นต้อง สามารถระบุชื่อของอ็อบเจ็กต์ที่แท้จริง ชื่อถูกระบุ แบบสัมพันธ์ (ไปยังไดเร็กทอรีการทำงานของกระบวนการ) หรือแบบสมบูรณ์ (ไปยังไดเร็กทอรี root ของกระบวนการ) การระบุชื่อทั้งหมดเริ่มต้นโดยการค้นหา ไดเร็กทอรีใดไดเร็กทอรีหนึ่งต่อไปนี้

กลไกการควบคุมการเข้าถึงที่ยอดเยี่ยม อนุญาตการควบคุมการเข้าถึงแหล่งข้อมูลอย่างมีประสิทธิภาพ และมี การป้องกันแยกต่างหากสำหรับข้อมูลที่เป็นความลับและ ความถูกต้อง กลไกการควบคุมการเข้าถึงที่ควบคุมโดยเจ้าของมีประสิทธิภาพเท่ากับผู้ใช้กระทำเท่านั้น ผู้ใช้ทั้งหมดต้องเข้าใจว่ามีการให้สิทธิ และการปฏิเสธ การเข้าถึงอย่างไร และค่าเหล่านี้ถูกตั้งค่าอย่างไร

การอนุญาตเข้าถึงสำหรับ NFS4 ACLs

ผู้ใช้ใด ที่มีสิทธิพิเศษสำหรับ WRITE_ACL จะสามารถ ควบคุมสิทธิการเข้าถึง เจ้าของรีซอร์สข้อมูลจะมี สิทธิพิเศษสำหรับ WRITE_ACL เสมอ สำหรับไฟล์และ ไดเร็กทอรีที่มี NFS4 ACLs การเข้าถึงได้รับอนุญาตดังนี้:

- รายการของ ACEs ถูกประมวลผลตามลำดับ และเฉพาะ ACEs ซึ่ง มี "who" (เช่น Identity) ที่ตรงกับผู้ร้องขอเท่านั้นที่จะได้รับการพิจารณา เพื่อประมวลผล credentials ของผู้ร้องขอไม่ถูกตรวจสอบขณะ ประมวลผล ACE ด้วยค่าพิเศษ who EVERYONE@
- แต่ละ ACE ถูกประมวลผลจนบิตทั้งหมดของการเข้าถึงของผู้ร้องขอ ได้รับอนุญาต เมื่อมีบิตหนึ่งได้รับอนุญาต จะไม่ถูกพิจารณา ในการประมวลผล ACEs ในภายหลัง
- ถ้าบิตใดๆ ที่สัมพันธ์กับการเข้าถึงของผู้ร้องขอถูกปฏิเสธ การเข้าถึงจะถูกปฏิเสธ และ ACEs ที่เหลือจะไม่ถูกประมวลผล
- ถ้าบิตทั้งหมดของการเข้าถึงของผู้ร้องขอไม่ได้รับอนุญาต และไม่มี ACE เหลือสำหรับการประมวลผล การเข้าถึงจะถูกปฏิเสธ

ถ้าการเข้าถึงที่ร้องขอถูกปฏิเสธโดย ACEs และ ผู้ใช้ที่ร้องขอเป็น superuser หรือ root โดยทั่วไปแล้วการเข้าถึงจะได้รับอนุญาต โปรดทราบว่าเจ้าของอ็อบเจ็กต์ได้รับอนุญาตสำหรับ READ_ACL, WRITE_ACL, READ_ATTRIBUTES และ WRITE_ATTRIBUTES เสมอ สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ อัลกอริทึมสำหรับการการอนุญาตให้เข้าถึง ดูที่ “รายการค่าควบคุมการเข้าใช้ NFS4” ในหน้า 141

การแก้ปัญหา Access Control List

ข้อมูลดังต่อไปนี้สามารถถูกใช้เพื่อแก้ปัญหา Access Control List (ACL)

NFS4 Access Control List บนอ็อบเจ็กต์ที่ทำให้แอ็พพลิเคชันแอ็พพลิเคชัน

คุณสามารถ ใช้โค้ดส่งคืนหรือความสามารถการติดตามเพื่อแก้ไขปัญหา โดยการตั้งค่า NFS4 ACL บนอ็อบเจ็กต์ เช่นไฟล์หรือไดเร็กทอรี ทั้งสองวิธีใช้คำสั่ง **aclput** และคำสั่ง **acledit** เพื่อค้นหาสาเหตุของปัญหา

การใช้โค้ดส่งคืนสำหรับการแก้ปัญหา

เมื่อต้องการ แสดงโค้ดส่งคืน ให้ใช้คำสั่ง `echo $?` หลังจากคุณรันคำสั่ง **aclput** รายการดังต่อไปนี้ แสดงโค้ดส่งคืนและคำอธิบาย:

22 (EINVAL, defined in /usr/include/sys/errno.h)

ข้อมูลต่อไปนี้เป็นสาเหตุที่เป็นไปได้สำหรับโค้ดนี้:

- การจัดรูปแบบข้อความไม่ถูกต้องในฟิลด์ใดๆ ของฟิลด์ 4 ฟิลด์
- ขนาดของอินพุต NFS4 ACL มากกว่า 64 KB
- ACL ถูกใช้กับไฟล์ที่มีอย่างน้อยหนึ่ง ACE โดยมี ACE มาสก์เซ็ตเป็น w (WRITE_DATA) แต่ไม่ใช่ p (APPEND_DATA) หรือ p (APPEND_DATA) แต่ไม่ใช่ w (WRITE_DATA)
- ACL ถูกใช้กับไดเร็กทอรีที่มีอย่างน้อยหนึ่ง ACE โดยมี ACE มาสก์เซ็ตเป็น w (WRITE_DATA) แต่ไม่ใช่ p (APPEND_DATA) หรือ p (APPEND_DATA) แต่ไม่ใช่ w (WRITE_DATA) และแฟล็ก ACE fi (FILE_INHERIT)
- มีอย่างน้อยหนึ่ง ACE ที่มี OWNER@ เซ็ตเป็น **who** (**Identity**) พิเศษและอย่างน้อยหนึ่ง ACE มาสก์ c (READ_ACL), C (WRITE_ACL), a (READ_ATTRIBUTE) และ A (WRITE_ATTRIBUTE) ถูก ปฏิเสธโดยชนิด ACE d

124 (ENOTSUP, defined in /usr/include/sys/errno.h)

ข้อมูลต่อไปนี้เป็นสาเหตุที่เป็นไปได้สำหรับโค้ดนี้:

- ค่าพิเศษซึ่งอาจไม่ได้เป็นหนึ่งในสามค่า (OWNER@, GROUP@, หรือ EVERYONE@) ในหนึ่งของค่า ACE
- มีอย่างน้อยหนึ่ง ACE ที่มีชนิด ACE u (AUDIT) หรือ l (ALARM)

13 (EACCES, ที่กำหนดใน /usr/include/sys/errno.h)

ข้อมูลต่อไปนี้เป็นสาเหตุที่เป็นไปได้สำหรับโค้ดนี้:

- คุณไม่ได้รับอนุญาตให้อ่านไฟล์อินพุตที่มี NFS4 ACE
- คุณไม่ได้รับอนุญาตให้ค้นหาไดเร็กทอรีพาเรนท์ของอ็อบเจ็กต์ปลายทาง เนื่องจากคุณไม่มีสิทธิ x (EXECUTE) ในไดเร็กทอรีพาเรนท์ของอ็อบเจ็กต์ปลายทาง
- คุณอาจไม่ได้รับอนุญาตให้เขียนหรือเปลี่ยน ACL ถ้าอ็อบเจ็กต์ เชื่อมโยงอยู่แล้วกับ NFS4 ACL ประกันว่าคุณจะมี privilege สำหรับ ACE มาสก์ C (WRITE_ACL)

การใช้ความสามารถ Trace สำหรับการแก้ปัญหา

คุณยังสามารถ สร้างรายงานการติดตามเพื่อค้นหาสาเหตุของปัญหา สถานการณ์ดังต่อไปนี้แสดงวิธีใช้การติดตามเพื่อค้นหาสาเหตุของ ปัญหาใช้กับ NFS4 ACL ถ้าคุณมีไฟล์ /j2v2/file1 โดย มี NFS4 ACL:

```
 s:(EVERYONE@):  a        acC
```

และ ACL ดังต่อไปนี้อยู่ในไฟล์อินพุต input_acl_file:

```
s:(EVERYONE@):  a        rwxacC
```

ทำขั้นตอน ดังต่อไปนี้ให้สมบูรณ์เพื่อแก้ปัญหาด้วยความสามารถในการติดตาม:

1. รันการติดตาม **aclput** และ **trcrpt** การใช้คำสั่ง ดังต่อไปนี้:

   ```
   $ trace -j  478 -o trc.raw
   $->!aclput -i input_acl_file -t NFS4 /j2v2/file1
   $ ->quit
   $ trcrpt trc.raw > trc.rpt
   ```

2. วิเคราะห์รายงานการติดตาม เมื่อ ACL ถูกนำมาใช้กับไฟล์หรือไดเร็กทอรี จะมีการตรวจสอบการเข้าถึงเพื่อ เขียน หรือเปลี่ยนแปลง ACL จากนั้นใช้ ACL ไฟล์ที่มีบรรทัดเหมือนกับดังต่อไปนี้:

```
478 xxx  xxx  ACL ENGINE: chk_access entry: type=NFS4 obj_mode=33587200 size=68 ops=16384 uid=100

478 xxx  xxx  ACL ENGINE: chk_access exit: type=NFS4 rc=0 ops=16384 priv=0 against=0
478 xxx  xxx  ACL ENGINE: set_acl entry: type=NFS4 ctl_flg=2 obj_mode=33587200 mode=0 size=48

478 xxx  xxx  ACL ENGINE: validate_acl: type=NFS4 rc=22 ace_cnt=1 acl_len=48 size=12
478 xxx  xxx  ACL ENGINE: set_acl exit: type=NFS4 rc=22 obj_mode=33587200 size=68 cmd=536878912
```

บรรทัด ที่สองมี `chk_access exit` ชี้ว่า การเข้าถึงสามารถทำได้ (`rc = 0`) เพื่อเขียน ACL บรรทัด ที่สี่มี `validate_acl`, และบรรทัดที่ห้า มี `set_acl exit` ชี้ว่า ACL ถูกนำมาใช้ไม่สำเร็จ (`rc=22` ระบุ **EINVAL**) บรรทัดที่สี่ มีค่า `validate_acl` แสดงว่า มีปัญหาในบรรทัดแรกของ ACE (`ace_cnt=1`) ถ้าคุณอ้างอิงถึง ACE แรก `s:(EVERYONE@): a rwxacC)` ไม่มี **p** เป็นมาสก์การเข้าถึง **p** จำเป็นนอกเหนือ จาก **w** เมื่อใช้ ACL

การแก้ปัญหาการปฏิเสธการเข้าถึง

การดำเนินการ filesystem (ตัวอย่างเช่น อ่านหรือเขียน ข้อมูล) อาจล้มเหลวในอ็อบเจ็กต์ที่สัมพันธ์ กับ NFS4 ACL โดยปกติ ข้อความแสดงความผิดพลาดถูกแสดง แต่ข้อความ นั้นอาจไม่มีข้อมูลเพียงพอในการกำหนด ปัญหาในการเข้าถึง คุณสามารถใช้ความสามารถในการติดตามเพื่อค้นหาปัญหาในการเข้าถึง ตัวอย่างเช่น ถ้าคุณมีไฟล์ `/j2v2/file2` ด้วย NFS4 ACL ดังต่อไปนี้:

```
s:(EVERYONE@):  a       rwpx
```

คำสั่ง ดังต่อไปนี้รายงานข้อผิดพลาด "Permission denied":

```
ls -l /j2v2/file2
```

ดำเนินการ ขั้นตอนดังต่อไปนี้ให้สมบูรณ์เพื่อแก้ไขปัญหานี้:

1. รันการติดตาม `ls -l /j2v2/file2` และ trcrpt โดยใช้คำสั่งดังต่อไปนี้:

   ```
   $ trace -j  478 -o trc.raw
   $->!ls -l /j2v2/file2
   $ ->quit
   $ trcrpt trc.raw > trc.rpt
   ```

2. วิเคราะห์รายงานการติดตาม ไฟล์ที่มีบรรทัดเหมือนกับดัง ต่อไปนี้:

   ```
   478  xxx  xxx   ACL ENGINE: chk_access entry: type=NFS4 obj_mode=33587711 size=68 ops=1024 uid=100
   478  xxx  xxx   ACL ENGINE: nfs4_chk_access_self: type=NFS4 aceN=1 aceCnt=1 req=128 deny=0
   478  xxx  xxx   ACL ENGINE: nfs4_mask_privcheck: type=NFS4 deny=128 priv=128
   478  xxx  xxx   ACL ENGINE: chk_access exit: type=NFS4 rc=13 ops=1024 priv=0 against=0
   ```

   บรรทัด ที่สามบ่งชี้ว่าการเข้าถึงถูกปฏิเสธสำหรับ access `mask = 128` (**0x80**) ซึ่งคือ `READ_ATTRIBUTES` เท่านั้น (ดูที่ไฟล์ `/usr/include/sys/acl.h`)

ภาพรวมการตรวจสอบ

ระบบย่อยการตรวจสอบช่วยให้ผู้ดูแลระบบสามารถบันทึก ข้อมูลที่เกี่ยวข้องกับความปลอดภัย ซึ่งสามารถนำไปวิเคราะห์ตรวจหาการฝ่าฝืนนโยบาย การรักษาความปลอดภัยระบบที่อาจเกิดขึ้นและที่เกิดขึ้นจริง

ระบบย่อยการตรวจสอบ

ระบบย่อยการตรวจสอบมีการตรวจหา การรวบรวม และการประมวลผลฟังก์ชัน

• "การตรวจหาเหตุการณ์การตรวจสอบ" ในหน้า 151

- “การรวบรวมข้อมูลเหตุการณ์”
- “การประมวลผลข้อมูล การติดตามการตรวจสอบ” ในหน้า 152

ผู้ดูแลระบบสามารถตั้งค่าฟังก์ชันแต่ละฟังก์ชันเหล่านี้

การตรวจหาเหตุการณ์การตรวจสอบ

การตรวจหา เหตุการณ์ถูกแจกจ่ายไปยัง Trusted Computing Base (TCB) ทั้ง ในเคอร์เนล (โค้ดสถานะผู้ดูแล) และโปรแกรมที่ไว้วางใจ (โค้ด สถานะผู้ใช้) เหตุการณ์ที่ตรวจสอบได้คือการเกิดเหตุการณ์ที่เกี่ยวกับความปลอดภัยในระบบ การเกิดเหตุการณ์ที่เกี่ยวกับความปลอดภัยคือการเปลี่ยนแปลงใดๆ ที่เกิดกับสถานะความปลอดภัยของ ระบบ การพยายามฝ่าฝืน หรือการฝ่าฝืนที่เกิดขึ้นจริงใดๆ ของการควบคุมการเข้าถึงระบบหรือ นโยบายการรักษาความปลอดภัยต่อความรับผิดชอบในหน้าที่ หรือทั้งสอง โปรแกรมและโมดูลเคอร์เนล ที่ตรวจหาเหตุการณ์ที่ตรวจสอบได้มีหน้าที่ในการรายงานเหตุการณ์เหล่านี้ไปยังตัวบันทึกการตรวจสอบระบบ ที่รันเป็นส่วนหนึ่งของเคอร์เนลและสามารถเข้าถึง ได้โดยใช้รูทีนย่อย (สำหรับการตรวจสอบโปรแกรมที่ไว้วางใจ) หรือภายในการเรียกใช้ โพรซีเดอร์เคอร์เนล (สำหรับการตรวจสอบสถานะผู้ดูแล) ข้อมูลที่รายงานประกอบด้วย ชื่อเหตุการณ์ที่ตรวจสอบได้ เหตุการณ์สำเร็จหรือล้มเหลว และ ข้อมูลเฉพาะเหตุการณ์เพิ่มเติมที่เกี่ยวข้องกับการตรวจสอบการรักษาความปลอดภัย

การตั้งค่า การตรวจหาเหตุการณ์ประกอบด้วยการเปิดใช้หรือปิดใช้การตรวจหาเหตุการณ์และ การระบุว่าเหตุการณ์ใดที่จะถูกตรวจสอบสำหรับผู้ใช้ใด ในการเรียกทำงานการตรวจหา เหตุการณ์ ใช้คำสั่ง **audit** เพื่อเปิดใช้งานหรือปิดใช้งาน ระบบย่อยการตรวจสอบ ไฟล์ `/etc/security/audit/config` ประกอบด้วยเหตุการณ์และผู้ใช้ที่ประมวลผลโดยระบบย่อยการตรวจสอบ

การรวบรวมข้อมูลเหตุการณ์

การรวบรวม ข้อมูลเป็นการรวมการบันทึกการทำงานของเหตุการณ์ที่ตรวจสอบได้ที่เลือกไว้ ฟังก์ชันนี้ ดำเนินการโดยตัวบันทึกการตรวจสอบเคอร์เนล ซึ่งมีทั้งส่วนการติดต่อการเรียกใช้ระบบ และการเรียกใช้โพรซีเดอร์ระหว่างเคอร์เนลที่บันทึกเหตุการณ์ที่ตรวจสอบได้

ตัวบันทึกการตรวจสอบมีหน้าที่สร้างเร็กคอร์ดการตรวจสอบที่สมบูรณ์ อันประกอบด้วย ส่วนหัวการตรวจสอบ ที่มีข้อมูลทั่วไปสำหรับทุกเหตุการณ์ (เช่น ชื่อเหตุการณ์ ผู้ใช้ที่รับผิดชอบ เวลาและสถานะที่ส่งคืน ของเหตุการณ์) และการติดตามการตรวจสอบ ซึ่งมีข้อมูลเฉพาะเหตุการณ์ ตัวบันทึกการตรวจสอบผนวกเร็กคอร์ดที่เกิดขึ้นในภายหลังต่อท้ายการติดตามการตรวจสอบเคอร์เนล ซึ่งสามารถถูกเขียนได้โดยใช้โหมดใดโหมดหนึ่งในสองโหมด (หรือใช้ทั้งสองโหมด):

โหมด BIN
: การติดตามถูกเขียนลงในไฟล์สำรอง ที่จัดให้มีเพื่อความปลอดภัย และสื่อบันทึกระยะยาว

โหมด STREAM
: การติดตามถูกเขียนลงในบัฟเฟอร์แบบวนซ้ำที่ถูกอ่านแบบซิงโครนัส ในอุปกรณ์จำลองการตรวจสอบ โหมด STREAM มีการตอบกลับในทันที

การรวบรวมข้อมูลสามารถตั้งค่าได้ทั้งที่ front end (การบันทึกเหตุการณ์) และที่ back end (การประมวลผลการติดตาม) การบันทึกเหตุการณ์ สามารถเลือกได้ขึ้นกับผู้ใช้แต่ละคน ผู้ใช้แต่ละคนมีชุดที่กำหนดของเหตุการณ์การตรวจสอบ ที่ถูกบันทึกการทำงานในการติดตามการตรวจสอบเมื่อเกิดเหตุการณ์นั้นขึ้น ที่ back end โหมด สามารถตั้งค่าแยกเฉพาะได้ ดังนั้นผู้ดูแลระบบสามารถนำการประมวลผล back-end ที่เหมาะสมที่สุดสำหรับสภาวะแวดล้อมเฉพาะไปใช้ได้ นอกจากนั้น การตวจสอบโหมด BIN สามารถตั้งค่าเพื่อสร้างการแจ้งเตือนในกรณีที่พื้นที่ของระบบไฟล์ ที่มีอยู่สำหรับการติดตามนั้นเหลือน้อยมาก

การประมวลผลข้อมูล การติดตามการตรวจสอบ

ระบบปฏิบัติการมีหลายอ็อพชันให้เลือกเพื่อประมวลผล การติดตามการตรวจสอบเคอร์เนล การติดตามโหมด BIN สามารถถูกบีบอัด กรอง หรือ จัดรูปแบบสำหรับเอาต์พุต หรือการผสมรวมกันอย่างเหมาะสมก่อนบันทึกลง สื่อบันทึกของการติดตามการตรวจสอบ ถ้ามี การบีบอัดทำโดยใช้การเข้ารหัส Huffman การกรองทำโดยใช้การเลือกบันทึกการตรวจสอบแบบ standard query language (SQL) (โดยใช้คำสั่ง **auditselect**) ซึ่งจัดให้มี ทั้งการดูแบบเลือกและการจัดเก็บแบบเลือกของการติดตามการตรวจสอบ การจัดรูปแบบ ของเร็กคอร์ดการติดตามการตรวจสอบสามารถใช้เพื่อตรวจสอบการติดตามการตรวจสอบ เพื่อสร้าง รายงานความปลอดภัยเป็นระยะ และเพื่อพิมพ์การติดตามการตรวจสอบลงกระดาษ

การติดตาม การตรวจสอบโหมด STREAM สามารถมอนิเตอร์ ณ เวลาจริง เพื่อให้มีความสามารถในการมอนิเตอร์การคุมคามได้ทันที การตั้งค่าของอ็อพชันเหล่านี้จัดการโดยโปรแกรมแยก ที่สามารถเรียกใช้เป็นแบบกระบวนการ daemon เพื่อกรองการติดตามโหมด BIN หรือ STREAM อย่างใดอย่างหนึ่ง แม้ว่าโปรแกรมการกรองบางโปรแกรมโดยปกติแล้วเหมาะสมกับ โหมดใดโหมดหนึ่งเพียงโหมดเดียวมากกว่า

การตั้งค่าระบบย่อยการตรวจสอบ

ระบบย่อยการตรวจสอบมีตัวแปรสถานะโกลบอลที่บ่งชี้ ว่าระบบย่อยการตรวจสอบเปิดใช้หรือไม่ นอกจากนั้น แต่ละกระบวนการจะมี ตัวแปรสถานะโลคัลที่บ่งชี้ว่าระบบย่อยการตรวจสอบควรบันทึก ข้อมูลเกี่ยวกับกระบวนการนี้หรือไม่

ตัวแปรทั้งสองนี้กำหนดว่าเหตุการณ์จะถูกตรวจพบโดยโมดูลและโปรแกรม Trusted Computing Base (TCB) หรือไม่ การปิดใช้การตรวจสอบ TCB สำหรับ กระบวนการที่เจาะจงทำให้กระบวนการทำการตรวจสอบด้วยตนเอง และไม่ต้องข้าม นโยบายความรับผิดชอบระบบ การอนุญาตให้โปรแกรมที่ไว้วางใจทำการตรวจสอบตัวเอง ช่วยให้การรวบรวมข้อมูลเป็นไปอย่างมีประสิทธิภาพและประสิทธิผลมากยิ่งขึ้น

การรวบรวมข้อมูลระบบย่อยการตรวจสอบ

การรวบรวม ข้อมูลขึ้นกับการเลือกเหตุการณ์และโหมดการติดตามการตรวจสอบเคอร์เนล ซึ่งทำโดย รูทีนเคอร์เนลที่มีส่วนการติดต่อกับข้อมูลการบันทึกการทำงาน ที่ใช้โดย คอมโพเนนต์ TCB ที่ตรวจหาเหตุการณ์ที่ตรวจสอบได้ และส่วนการติดต่อการตั้งค่า ที่ใช้โดยระบบย่อยการตรวจสอบเพื่อควบคุมรูทีนการบันทึกการตรวจสอบ

การบันทึกการตรวจสอบ

เหตุการณ์ที่ตรวจสอบได้ถูกบันทึกโดย ส่วนการติดต่อต่อไปนี้: สถานะผู้ใช้และสถานะผู้ดูแล ส่วนของสถานะผู้ใช้ ของ TCB ใช้รูทีนย่อย **auditlog** หรือ **auditwrite** ขณะที่ส่วนสถานะผู้ดูแลของ TCB ใช้ชุดการเรียกใช้งานโพรซีเดอร์ เคอร์เนล

สำหรับแต่ละเร็กคอร์ด ตัวบันทึกเหตุการณ์การตรวจสอบจะเพิ่มส่วนหัวการตรวจสอบไว้ข้างหน้า ของข้อมูลเฉพาะของเหตุการณ์ ส่วนหัวนี้ระบุผู้ใช้และกระบวนการ ที่เหตุการณ์นี้กำลังถูกตรวจสอบ รวมถึงเวลาที่เกิดเหตุการณ์ โค้ด ที่ตรวจหาเหตุการณ์จะเพิ่มประเภทเหตุการณ์และโค้ดส่งกลับ หรือสถานะ พร้อมข้อมูลเฉพาะของเหตุการณ์ที่เป็นทางเลือก (การติดตามเหตุการณ์) ข้อมูล เฉพาะของเหตุการณ์ประกอบด้วยชื่ออ็อบเจ็กต์ (ตัวอย่างเช่น ไฟล์ที่ถูกปฏิเสธ การเข้าถึง หรือ tty ที่ใช้ในการพยายามล็อกอินที่ล้มเหลว) พารามิเตอร์รูทีนย่อยและ ข้อมูลอื่นที่ถูกแก้ไข

เหตุการณ์ถูกกำหนดในเชิงสัญลักษณ์มากกว่าเชิงจำนวน วิธีนี้ช่วยลดโอกาสเกิดเหตุการณ์ที่ชื่อจะชนกัน โดยไม่ต้องใช้รูปแบบการลงทะเบียน เหตุการณ์ เนื่องจากรูทีนย่อยเป็นข้อกำหนดเคอร์เนลส่วนเพิ่มและตรวจสอบได้ ที่ไม่มีจำนวน switched virtual circuit (SVC) ที่คงที่ ทำให้ยากที่จะบันทึก เหตุการณ์ตามจำนวน การแม็พหมายเลขต้องได้รับการตรวจทานและบันทึกการทำงาน ทุกครั้งที่ส่วนการติดต่อเคอร์เนลถูกขยายหรือกำหนดใหม่

รูปแบบเร็กคอร์ดการตรวจสอบ

เร็กคอร์ดการตรวจสอบประกอบด้วย ส่วนหัวทั่วไป ตามด้วยการติดตามการตรวจสอบที่เฉพาะสำหรับเหตุการณ์การตรวจสอบของ เร็กคอร์ด โครงสร้างสำหรับส่วนหัวถูกกำหนดไว้ในไฟล์ /usr/include/sys/audit.h รูปแบบของข้อมูลในการติดตามการตรวจสอบเป็นรูปแบบเฉพาะสำหรับแต่ละเหตุการณ์พื้นฐาน และแสดงในไฟล์ /etc/security/audit/events

ข้อมูล ในส่วนหัวของการตรวจสอบโดยปกติรวบรวมโดยรูทีนบันทึกการทำงาน เพื่อให้แน่ใจในความถูกต้อง ขณะที่ข้อมูลในการติดตามการตรวจสอบจะจัดทำโดย โค้ดที่ตรวจหาเหตุการณ์ ตัวบันทึกการตรวจสอบไม่มีข้อมูลของ โครงสร้าง หรือซีแมนทิกส์ของการติดตามการตรวจสอบ ตัวอย่าง เมื่อคำสั่ง **login** ตรวจพบล็อกอินที่ล้มเหลว จะบันทึกเหตุการณ์เฉพาะนั้นด้วย เทอร์มินัลที่ เกิดเหตุการณ์ และเขียนเร็กคอร์ดลงในการติดตามการตรวจสอบโดยใช้รูทีนย่อย **auditlog** คอมโพเนนต์เคอร์เนล ตัวบันทึกการตรวจสอบบันทึกข้อมูลส่วนหัวเรื่อง (ID ผู้ใช้ ID กระบวนการ เวลา) ลงในส่วนหัว และนำไปผนวกกับข้อมูลอื่น ผู้เรียกใช้ระบุเฉพาะชื่อเหตุการณ์และฟิลด์ผลลัพธ์ในส่วนหัวเท่านั้น

การตั้งค่าตัวบันทึกการตรวจสอบ

ตัวบันทึกการตรวจสอบมีหน้าที่สร้างรายการบันทึกการตรวจสอบ ที่สมบูรณ์ คุณต้องเลือกเหตุการณ์ที่จะตรวจสอบที่คุณต้องการบันทึก

การเลือกเหตุการณ์ที่จะตรวจสอบ

การเลือกเหตุการณ์ที่จะตรวจสอบ มีประเภทต่อไปนี้:

การตรวจสอบก่อนการประมวลผล

ในการเลือกเหตุการณ์การประมวลผลได้อย่างมีประสิทธิภาพ ผู้ดูแลระบบสามารถกำหนด คลาสการตรวจสอบ คลาสการตรวจสอบเป็นเซ็ตย่อยของเหตุการณ์การตรวจสอบพื้นฐานใน ระบบ คลาสการตรวจสอบจัดให้มีการจัดกลุ่มเชิงตรรกะของเหตุการณ์การตรวจสอบพื้นฐาน เพื่อความสะดวก

สำหรับผู้ใช้แต่ละรายบนระบบ ผู้ดูแลระบบ จะกำหนดชุดของคลาสการตรวจสอบที่พิจารณาเหตุการณ์พื้นฐานที่สามารถ บันทึกได้สำหรับผู้ใช้รายนั้น แต่ละกระบวนการที่ทำงานโดยผู้ใช้จะถูกแท็กกับคลาส การตรวจสอบของกระบวนการ

การตรวจสอบรายอ็อบเจ็กต์

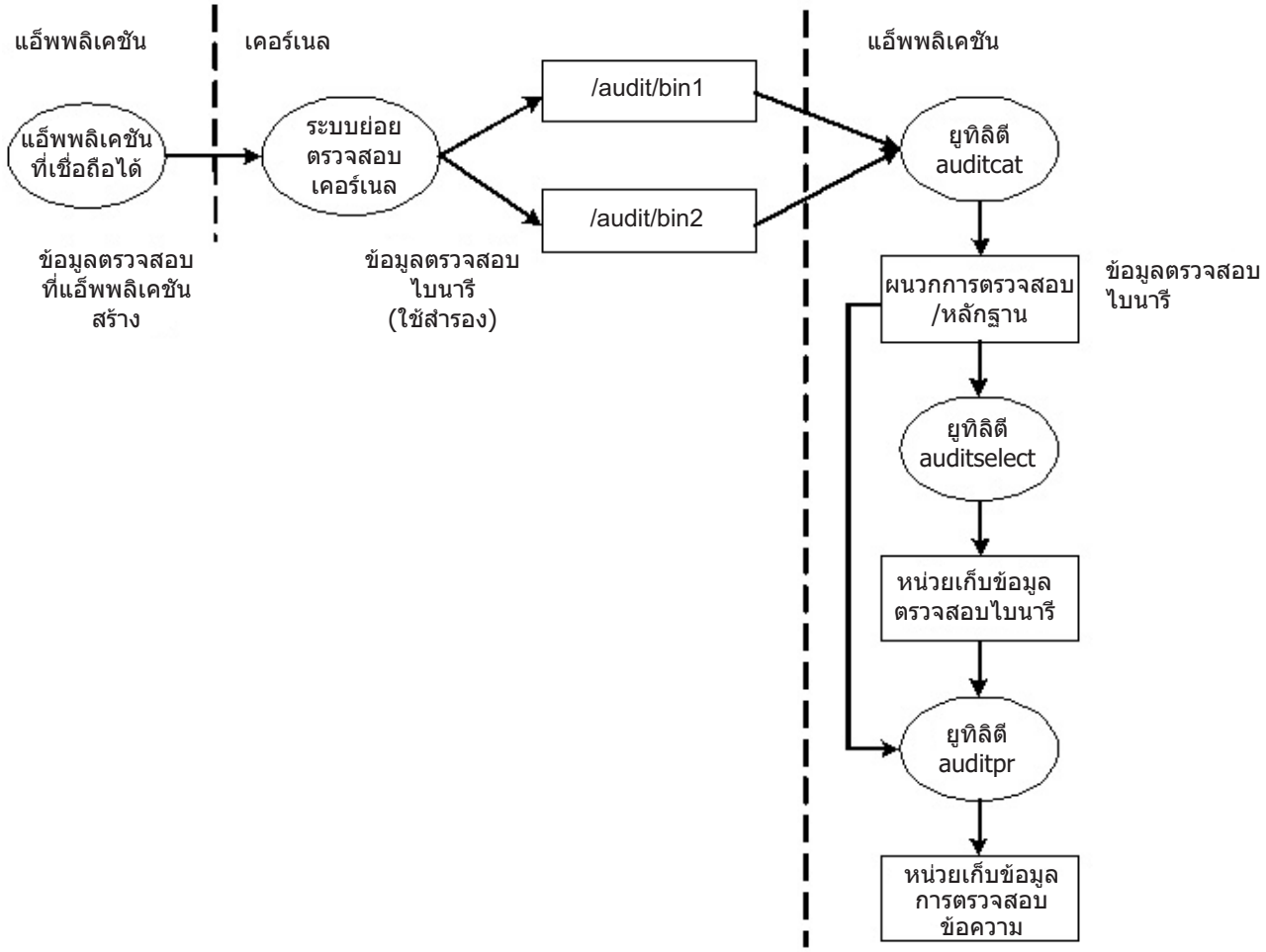
ระบบปฏิบัติการจัดให้มีการตรวจสอบการเข้าถึงอ็อบเจ็กต์ตามชื่อ ซึ่งคือ การตรวจสอบอ็อบเจ็กต์ที่เจาะจง (ปกติเป็นไฟล์) การตรวจสอบอ็อบเจ็กต์ตามชื่อ ช่วยป้องกันการบันทึกครอบคลุมการเข้าถึงอ็อบเจ็กต์ทั้งหมดให้เหลือตรวจสอบอ็อบเจ็กต์ ที่ต้องการบางอ็อบเจ็กต์ นอกจากนั้น โหมดการตรวจสอบสามารถถูกระบุ เพื่อให้การเข้าถึงเฉพาะของโหมดที่ระบุเท่านั้น (read/write/execute) ที่ถูกบันทึก

โหมดการติดตามการตรวจสอบเคอร์เนล

การบันทึกเคอร์เนลสามารถตั้งค่า เป็นโหมด BIN หรือ STREAM เพื่อกำหนดตำแหน่งที่จะใช้เขียนการติดตามการตรวจสอบเคอร์เนล ถ้าใช้โหมด BIN ตัวบันทึกการตรวจสอบเคอร์เนลต้องถูกระบุ (ก่อนเริ่มทำงาน การตรวจสอบ) อย่างน้อยหนึ่ง file descriptor ที่จะบันทึกรายการบันทึกต่อ

โหมด BIN ประกอบด้วยการเขียนบันทึกการตรวจสอบลงในไฟล์ที่เกี่ยวข้อง เมื่อเริ่มทำงาน การตรวจสอบ เคอร์เนลถูกส่ง file descriptors สองตัวและขนาด bin สูงสุด ที่แนะนำ โดยหยุดทำงานกระบวนการเรียกใช้ชั่วคราวและเริ่มการเขียนบันทึกการตรวจสอบ ลงใน file descriptor แรก เมื่อขนาดของ bin แรกมีขนาดถึง ขนาด bin สูงสุด และถ้า file descriptor ที่สองใช้ได้ ตัวบันทึกจะสลับ มาที่ bin ที่สองและเรียกใช้งานกระบวนการเรียกใช้อีกครั้ง เคอร์เนลยังคง เขียนลงใน bin ที่สองจนกว่าจะถูก

เขียนใช้อีกครั้งด้วยไฟล์ที่ใช้ได้ file descriptor อื่น ถ้าถึงจุดที่ bin ที่สองเต็ม จะสลับกลับมาที่ bin แรก และกระบวนการเรียกใช้ส่งค่ากลับทันที มิฉะนั้น กระบวนการเรียกใช้ จะถูกหยุดทำงานชั่วคราว และเคอร์เนลยังคงเขียนบันทึกลงใน bin ที่สองจนกว่าจะเต็ม กระบวนการยังคงดำเนินไปเช่นนั้นจนกระทั่งการตรวจสอบ ถูกปิด ดูภาพต่อไปนี้เพื่อดูการแสดงโหมด BIN ที่ตรวจสอบ:

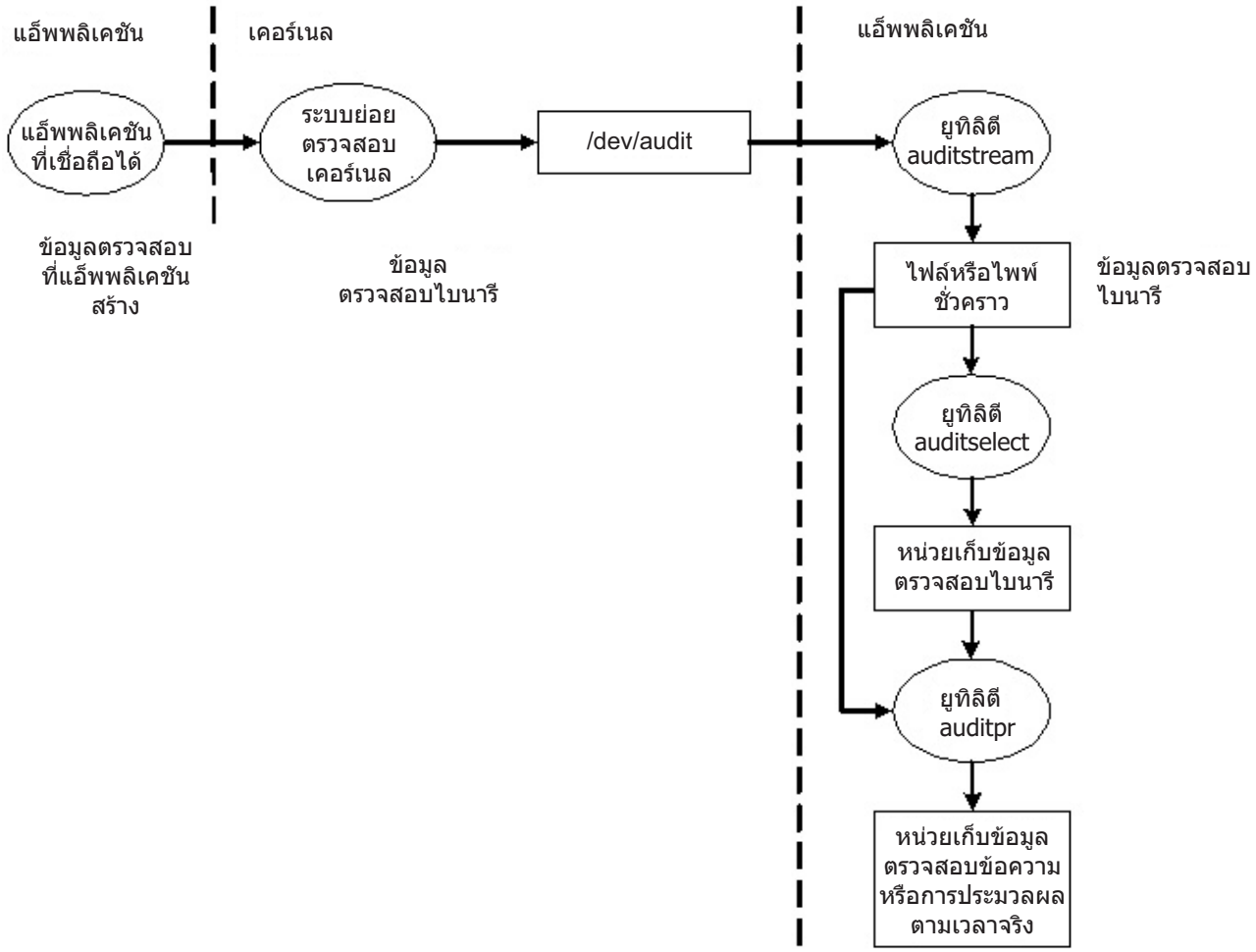


รูปที่ 1. กระบวนการของโหมด BIN ที่ตรวจสอบ. ภาพประกอบนี้แสดง กระบวนการของโหมด BIN ที่ตรวจสอบ

กลไก bin ทางเลือกถูกใช้เพื่อให้แน่ใจว่าระบบย่อย การตรวจสอบมีการเขียนเสมอในขณะดำเนินการบันทึกการตรวจสอบ เมื่อระบบย่อยการตรวจสอบสลับไป bin อื่น ระบบจะลบเนื้อหา bin แรกไปไว้ที่ไฟล์ `trace` เมื่อถึงเวลาสลับ bin อีกครั้ง bin แรกจะพร้อมใช้งาน ระบบแยกสื่อบันทึกและวิเคราะห์ ข้อมูลจากการสร้างข้อมูล โดยทั่วไป โปรแกรม **auditcat** ถูกใช้เพื่ออ่านข้อมูลจาก bin ที่เคอร์เนลไม่ได้เขียนอยู่ ในขณะนี้ เพื่อให้มั่นใจว่าระบบจะไม่ขาดแคลนพื้นที่สำหรับ การติดตามการตรวจสอบ (เอาต์พุตของโปรแกรม **auditcat**) พารามิเตอร์ *freespace* จึงถูกระบุในไฟล์ `/etc/security/audit/config` ถ้าระบบมีขนาดน้อยกว่าขนาดบล็อก 512 ไบต์ที่ระบุในที่นี้ ระบบจะสร้างข้อความ syslog

ถ้าเปิดใช้การตรวจสอบ พารามิเตอร์ *binmode* ใน `start` stanza ใน `/etc/security/audit/config` ควร ถูกตั้งค่าเป็น `panic` พารามิเตอร์ *freespace* ใน `bin` stanza ควรถูกตั้งค่าต่ำสุดเป็นค่าที่ เท่ากับ 25 เปอร์เซ็นต์ของพื้นที่ดิสก์ที่ใช้เป็นที่เก็บหลักฐานการตรวจสอบ พารามิเตอร์ *bytethreshold* และ *binsize* แต่ละค่าควรตั้งเป็น 65536 ไบต์

ในโหมด STREAM เคอร์เนลเขียน บันทึกลงในบัฟเฟอร์แบบวงกลม เมื่อเคอร์เนลเขียนถึงท้ายบัฟเฟอร์ เคอร์เนลจะกลับมาเขียนที่จุดเริ่มต้นบัฟเฟอร์ใหม่ กระบวนการจะอ่านข้อมูลจาก อุปกรณ์จำลองที่ถูกเรียก `/dev/audit` เมื่อกระบวนการ เปิดใช้อุปกรณ์นี้ จะสร้างแชนเนลสำหรับกระบวนการนั้น เป็นทางเลือก เหตุการณ์จะถูกอ่านบนแชนเนลที่สามารถระบุเป็นรายการของคลาสการตรวจสอบ ดูภาพต่อไปนี้เพื่อดูการแสดงโหมด STREAM ที่ตรวจสอบ:



รูปที่ 2. กระบวนการของโหมด STREAM ที่ตรวจสอบ. ภาพประกอบนี้แสดง กระบวนการของโหมด STREAM ที่ตรวจสอบ

วัตถุประสงค์หลักของโหมด STREAM คืออนุญาตให้มีการอ่าน หลักฐานการตรวจสอบตามเวลา ซึ่งเป็นสิ่งที่เป็นที่ต้องการสำหรับการมอนิเตอร์หาการคุกคาม ณ เวลาจริง การใช้งาน อีกประการคือเพื่อสร้างการติดตามที่ถูกเขียนในทันที เป็นการป้องกันการเปลี่ยนแปลงเพื่อทำลายข้อมูล ใดๆ ที่อาจเกิดขึ้นกับหลักฐานการตรวจสอบ เนื่องจากเป็นไปได้ถ้าหลักฐานนั้นถูกเก็บบนสื่อบันทึกที่สามารถเขียนได้

อีกวิธีหนึ่งในการใช้โหมด STREAM คือเพื่อเขียน สตรีมการตรวจสอบลงในโปรแกรมที่เก็บข้อมูลการตรวจสอบบนระบบรีโมต ซึ่งอนุญาตให้ทำการประมวลผลใกล้เวลากลาง และยัง ช่วยป้องกันข้อมูลการตรวจสอบจากการเปลี่ยนแปลงเพื่อทำที่โฮสต์ต้นทาง

การประมวลผลบันทึกการตรวจสอบ

คำสั่ง **auditselect, auditpr** และ **auditmerge** มีพร้อมเพื่อประมวลผลบันทึกการตรวจสอบโหมด BIN หรือ STREAM ยูทิลิตีทั้งสองทำหน้าที่เป็นตัวกรองจึงช่วยให้ใช้งานได้ง่ายบนไพพ์ ซึ่งเหมาะเป็นอย่างยิ่งสำหรับการตรวจสอบโหมด STREAM

auditselect

สามารถใช้เพื่อเลือกเฉพาะบันทึกการตรวจสอบที่เจาะจงด้วยคำสั่งที่เหมือนคำสั่ง SQL ตัวอย่าง ในการเลือกเฉพาะเหตุการณ์ **exec( )** ที่สร้างโดยผู้ใช้ *afx* เท่านั้น ให้พิมพ์ต่อไปนี้:

```
auditselect -e "login==afx && event==PROC_Execute"
```

auditpr ใช้เพื่อแปลงบันทึกการตรวจสอบแบบไบนารีเป็นรูปแบบที่สามารถอ่านได้ จำนวนข้อมูล ที่แสดงจะขึ้นกับแฟล็กที่ระบุบนบรรทัด คำสั่ง เพื่อให้ได้ข้อมูลที่มีอยู่ทั้งหมด ให้รันคำสั่ง **auditpr** ดังนี้:

```
auditpr -v -hhelrtRpPTc
```

โดยที่แฟล็ก **–v** ถูกระบุ หลักฐานการตรวจสอบที่เป็นสตริงเฉพาะสำหรับเหตุการณ์ (ดูที่ไฟล์ `/etc/security/audit/events`) จะถูกแสดงนอกเหนือจากข้อมูลการตรวจสอบมาตรฐานที่เคอร์เนล ส่งสำหรับทุกเหตุการณ์

auditmerge

ใช้เพื่อผสมรวมหลักฐานการตรวจสอบแบบไบนารี วิธีนี้มีประโยชน์อย่างยิ่งถ้ามี หลักฐานการตรวจสอบจากหลายๆ ระบบที่จำเป็นต้องนำมารวมกัน คำสั่ง **auditmerge** จะนำชื่อของการติดตามบนบรรทัดคำสั่งและส่งการติดตามแบบไบนารีที่ถูกรวมกัน ไปยังเอาต์พุตมาตรฐาน ดังนั้นคุณยังจำเป็นต้องใช้คำสั่ง **auditpr** เพื่อช่วยให้สามารถอ่านได้ ตัวอย่าง คำสั่ง **auditmerge** และ **auditptr** สามารถรันดังนี้:

```
auditmerge trail.system1 trail.system2 | auditpr -v -hhelrRtpc
```

การใช้ระดับย่อยการตรวจสอบเพื่อตรวจสอบความปลอดภัยอย่างรวดเร็ว:

เมื่อต้องการมอนิเตอร์โปรแกรมเดี่ยวที่น่าสงสัยโดยไม่ตั้งค่าระบบย่อย การตรวจสอบ สามารถใช้คำสั่ง **watch** ซึ่งจะบันทึกเหตุการณ์ที่ร้องขอหรือเหตุการณ์ทั้งหมดที่ถูกสร้างโดยโปรแกรมที่ระบุ

ตัวอย่าง เมื่อต้องการดูเหตุการณ์ **FILE_Open** ทั้งหมดเมื่อรัน **vi /etc/hosts** ให้พิมพ์:

```
watch -eFILE_Open -o /tmp/vi.watch vi /etc/hosts
```

ไฟล์ `/tmp/vi.watch` แสดงเหตุการณ์ **FILE_Open** ทั้งหมดสำหรับ เซสชันเอดิเตอร์

การเลือกเหตุการณ์

การเลือกเหตุการณ์ต้องมีความสมดุลระหว่างรายละเอียดที่ไม่เพียงพอ กับที่มากเกินไป

ชุดของเหตุการณ์ที่ตรวจสอบได้บนระบบเป็นตัวกำหนดการมีอยู่ใดที่สามารถ ถูกตรวจสอบได้จริง และกลุ่มของการตรวจสอบที่จัดให้มี เหตุการณ์ที่ตรวจสอบได้ ต้องครอบคลุมเหตุการณ์ที่เกี่ยวกับความปลอดภัยบนระบบ ดังที่กำหนดก่อนหน้านี้ ระดับของรายละเอียดที่คุณใช้สำหรับข้อกำหนดเหตุการณ์ที่ตรวจสอบได้ต้องคง ความสมดุลระหว่างรายละเอียดที่ไม่เพียงพอ ซึ่งทำให้ผู้ดูแลระบบเข้าใจในข้อมูล ที่เลือกได้ยากมากขึ้น และรายละเอียดที่มากเกินไป ซึ่งนำไปสู่การรวบรวมข้อมูล มากเกิน ข้อกำหนดของเหตุการณ์ใช้ประโยชน์ของ ความเหมือนกันในเหตุการณ์ที่ตรวจพบ สำหรับวัตถุประสงค์ของการอภิปรายนี้ *เหตุการณ์ที่ตรวจพบ* คือ instance เดี่ยวของเหตุการณ์ที่ตรวจสอบได้ ตัวอย่างเช่น เหตุการณ์ ที่กำหนดอาจถูกตรวจพบในหลายที่ หลักการสำคัญคือเหตุการณ์ ที่ตรวจพบที่มีคุณสมบัติด้านความปลอดภัยคล้ายกันจะถูกเลือกเป็น เหตุการณ์ที่ตรวจสอบได้เดียวกัน รายการต่อไปนี้แสดงการจัดประเภทเหตุการณ์ของนโยบาย การรักษาความปลอดภัย:

- เหตุการณ์เป้าหมาย
  - การสร้างกระบวนการ
  - การลบกระบวนการ
  - การตั้งค่าแอ็ตทริบิวต์ความปลอดภัยหลัก: ID ผู้ใช้ ID กลุ่ม
  - กลุ่มกระบวนการ เทอร์มินัลการควบคุม
- เหตุการณ์อ็อบเจ็กต์
  - การสร้างอ็อบเจ็กต์
  - การลบอ็อบเจ็กต์
  - การเปิดอ็อบเจ็กต์ (รวมถือกระบวนการเป็นอ็อบเจ็กต์)
  - การปิดอ็อบเจ็กต์ (รวมถือกระบวนการเป็นอ็อบเจ็กต์)
  - การตั้งค่าแอ็ตทริบิวต์ความปลอดภัยอ็อบเจ็กต์: เจ้าของ กลุ่ม ACL
- อิมพอร์ต/เอ็กซ์พอร์ตเหตุการณ์
  - การอิมพอร์ตหรือการเอ็กซ์พอร์ตอ็อบเจ็กต์
- การตรวจสอบเหตุการณ์ได้
  - การเพิ่มผู้ใช้ การเปลี่ยนแอ็ตทริบิวต์ผู้ใช้ในฐานข้อมูลรหัสผ่าน
  - การเพิ่มกลุ่ม การเปลี่ยนแอ็ตทริบิวต์กลุ่มในฐานข้อมูลกลุ่ม
  - ล็อกอินผู้ใช้
  - ล็อกออฟผู้ใช้
  - การเปลี่ยนข้อมูลการพิสูจน์ตัวตนของผู้ใช้
  - การตั้งค่าเทอร์มินัลพาธที่ไว้วางใจ
  - การตั้งค่าการพิสูจน์ตัวตน
  - การดูแลจัดการการตรวจสอบ: การเลือกเหตุการณ์และหลักฐานการตรวจสอบ การเปิดใช้ หรือปิด การกำหนดคลาสการตรวจสอบผู้ใช้
- เหตุการณ์การดูแลจัดการระบบทั่วไป
  - การใช้สิทธิพิเศษ
  - การตั้งค่าระบบไฟล์
  - ข้อกำหนดและการตั้งค่าอุปกรณ์
  - ข้อกำหนดพารามิเตอร์การตั้งค่าระบบ
  - IPL ระบบปกติและการปิดระบบ
  - การตั้งค่า RAS
  - การตั้งค่าระบบอื่น
  - การเริ่มทำงานระบบย่อยการตรวจสอบ
  - การหยุดทำงานระบบย่อยการตรวจสอบ
  - การเคียวรีระบบย่อยการตรวจสอบ
  - การรีเซ็ตระบบย่อยการตรวจสอบ
- การฝ่าฝืนการรักษาความปลอดภัย (ร้ายแรง)

- การปฏิเสธสิทธิในการเข้าถึง
- สิทธิพิเศษล้มเหลว
- ข้อผิดพลาดและข้อผิดพลาดระบบที่ตรวจพบในการวินิจฉัย
- การเปลี่ยนแปลงที่มีการพยายามทำของ TCB

การตั้งค่าการตรวจสอบ

ขั้นตอนนี้แสดงวิธีตั้งค่าระบบย่อยการตรวจสอบ สำหรับข้อมูลที่เจาะจงเพิ่มเติม อ้างอิงไฟล์คอนฟิกูเรชันที่กล่าวถึง ในขั้นตอนเหล่านี้

1. เลือกกิจกรรมระบบ (เหตุการณ์) เพื่อตรวจสอบจากรายการในไฟล์ /etc/security/audit/events ถ้าคุณเพิ่มเหตุการณ์การตรวจสอบใหม่ในแอ็พพลิเคชัน หรือส่วนขยายเคอร์เนล คุณต้องแก้ไขไฟล์เพื่อเพิ่มเหตุการณ์ใหม่
   - คุณเพิ่มเหตุการณ์ในไฟล์นี้ถ้าคุณได้รวมโค้ดเพื่อบันทึก เหตุการณ์นั้นในแอ็พพลิเคชันโปรแกรม (โดยใช้รูทีนย่อย **auditwrite** หรือ **auditlog**) หรือใน ส่วนขยายเคอร์เนล (โดยใช้เคอร์เนลเซอร์วิส **audit_svcstart, audit_svcbcopy** และ **audit_svcfinis**)
   - ตรวจสอบว่าให้แน่ใจว่าคำสั่งการจัดรูปแบบสำหรับเหตุการณ์การตรวจสอบใหม่ใดๆ ถูกรวมในไฟล์ /etc/security/audit/events ข้อกำหนดคุณสมบัติเหล่านี้เปิดให้คำสั่ง **auditpr** สามารถบันทึกการติดตามการตรวจสอบเมื่อจัดรูปแบบเร็กคอร์ดการตรวจสอบ
2. จัดกลุ่มเหตุการณ์การตรวจสอบที่เลือกเป็นชุดของรายการที่คล้ายกันเรียกว่า *คลาส การตรวจสอบ* กำหนดคลาสการตรวจสอบเหล่านี้ใน stanza คลาสของไฟล์ /etc/security/audit/config
3. กำหนดคลาสการตรวจสอบให้แก่ผู้ใช้แต่ละคน และกำหนดเหตุการณ์ การตรวจสอบไปยังไฟล์ (อ็อบเจ็กต์) ที่คุณต้องการตรวจสอบ ดังนี้:
   - ในการกำหนดคลาสการตรวจสอบให้แก่ผู้ใช้รายคน เพิ่มบรรทัดใน stanza ผู้ใช้ของไฟล์ /etc/security/audit/config ในการกำหนดคลาสการตรวจสอบให้แก่ผู้ใช้ คุณสามารถใช้คำสั่ง **chuser**
   - ในการกำหนดเหตุการณ์การตรวจสอบให้แก่อ็อบเจ็กต์ (ข้อมูลหรือไฟล์เรียกทำงาน) ให้เพิ่ม stanza ของไฟล์นั้นในไฟล์ /etc/security/audit/objects
   - คุณยังสามารถระบุคลาสการตรวจสอบดีฟอลต์สำหรับผู้ใช้ใหม่โดยการแก้ไข ไฟล์ /usr/lib/security/mkuser.default ไฟล์นี้ เก็บแอ็ตทริบิวต์ผู้ใช้ที่จะใช้เมื่อสร้าง ID ผู้ใช้ ใหม่ ตัวอย่าง ใช้คลาสการตรวจสอบ general สำหรับ ID ผู้ใช้ใหม่ทั้งหมด ดังนี้:

     ```
     user:
         auditclasses = general
         pgrp = staff
         groups = staff
         shell = /usr/bin/ksh
         home = /home/$USER
     ```

     ในการรับค่าเหตุการณ์การตรวจสอบทั้งหมด ให้ระบุ คลาส ALL เมื่อทำเช่นนั้นบนระบบมียุ่งพอสมควรเสมอ จะมีการสร้างข้อมูลเป็นปริมาณมาก โดยทั่วไป ควรจำกัดจำนวนเหตุการณ์ที่จะถูกบันทึกให้มากขึ้น
4. ในไฟล์ /etc/security/audit/config ตั้งค่า ประเภทของการรวบรวมข้อมูลที่คุณต้องการโดยใช้การรวบรวมแบบ BIN, การรวบรวมแบบ STREAM หรือทั้งสองวิธี ตรวจสอบให้แน่ใจว่าข้อมูลการตรวจสอบไม่แย่ง พื้นที่ไฟล์กับข้อมูลอื่นโดยใช้ระบบไฟล์แยกต่างหาก สำหรับข้อมูลการตรวจสอบ นี้จะช่วยให้แน่ใจว่ามีพื้นที่เพียงพอสำหรับข้อมูล การตรวจสอบ ตั้งค่าประเภทของการรวบรวมข้อมูลดังนี้:
   - ในการตั้งค่าการรวบรวมแบบ BIN:

a. เปิดใช้งานการรวบรวมโหมด BIN โดยตั้งค่า `binmode = on` ใน stanza เริ่มต้น
b. แก้ไข binmode stanza เพื่อตั้งค่า bins และการติดตาม และระบุ พาธของไฟล์ที่มีคำสั่งการประมวลผลส่วนหลังของโหมด BIN ไฟล์ค่าดีฟอลต์สำหรับคำสั่งส่วนหลังคือไฟล์ `/etc/security/audit/bincmds`
c. ตรวจสอบให้แน่ใจว่า bins การตรวจสอบมีขนาดใหญ่เพียงพอต่อความต้องการของคุณ และตั้งค่าพารามิเตอร์ *freespace* ให้สอดคล้องกัน เพื่อรับการแจ้งเตือนถ้าระบบไฟล์เต็ม
d. รวมคำสั่งเชลล์ที่ประมวลผล bins การตรวจสอบในไพพ์ การตรวจสอบในไฟล์ `/etc/security/audit/bincmds`

- ในการตั้งค่าการรวบรวมแบบ STREAM:
  a. เปิดใช้งานการรวบรวมโหมด STREAM โดยตั้งค่า `streammode = on` ใน stanza เริ่มต้น
  b. แก้ไข streammode stanza เพื่อระบุพาธไปยังไฟล์ที่มี คำสั่งการประมวลผล streammode ไฟล์ค่าดีฟอลต์ที่มี ข้อมูลนี้ในไฟล์ `/etc/security/audit/streamcmds`
  c. รวมคำสั่งเชลล์ที่ประมวลผลเร็กคอร์ด stream ในไพพ์ การตรวจสอบในไฟล์ `/etc/security/audit/streamcmds`

5. เมื่อคุณทำการเปลี่ยนแปลงใดๆ ที่จำเป็นในไฟล์คอนฟิกูเรชันเสร็จเรียบร้อย คุณก็พร้อมใช้คำสั่ง **audit start** เพื่อเปิดใช้งานระบบย่อยการตรวจสอบ นี้จะสร้างเหตุการณ์ **AUD_It** ที่มีค่าเป็น 1
6. ใช้คำสั่ง **audit query** เพื่อดูว่าเหตุการณ์ และอ็อบเจ็กต์ใดที่ถูกตรวจสอบ นี้จะสร้างเหตุการณ์ **AUD_It** ที่มีค่าเป็น 2
7. ใช้คำสั่ง **audit shutdown** เพื่อปิดการทำงาน ระบบย่อยการตรวจสอบอีกครั้ง นี้จะสร้างเหตุการณ์ **AUD_It** ที่มีค่าเป็น 4

การสร้างบันทึกการติดตามทั่วไป:

ต่อไปนี้คือตัวอย่างการสร้างบันทึกการติดตาม ทั่วไป

ในตัวอย่างนี้ สมมติว่าผู้ดูแลระบบต้องการ ใช้ระบบย่อยการตรวจสอบเพื่อมอนิเตอร์ระบบเซิร์ฟเวอร์ที่มีผู้ใช้จำนวนมากขนาดใหญ่ ไม่มีการรวมเข้ากับ IDS โดยตรง บันทึกการตรวจสอบทั้งหมด จะถูกตรวจสอบด้วยตนเองเพื่อหาความผิดปกติ มีเพียงเหตุการณ์การตรวจสอบ ที่จำเป็นบางส่วนเท่านั้นที่ถูกบันทึก เพื่อให้จำนวนข้อมูลที่ถูกสร้างขึ้น มีขนาดที่สามารถจัดการได้

เหตุการณ์การตรวจสอบที่ถูกพิจารณาเพื่อ ทำการตรวจหาเหตุการณ์มีดังต่อไปนี้:

FILE_Write
เราต้องการทราบเกี่ยวกับการเขียนลงไฟล์ในไฟล์คอนฟิกูเรชัน ดังนั้น เหตุการณ์นี้จะถูกใช้กับไฟล์ทั้งหมดในแผนผัง `/etc`

PROC_SetUserIDs
การเปลี่ยนแปลงทั้งหมดของ ID ผู้ใช้

AUD_Bin_Def
คอนฟิกูเรชัน bin การตรวจสอบ

USER_SU
คำสั่ง **su**

PASSWORD_Change
คำสั่ง **passwd**

AUD_Lost_Rec
การแจ้งเตือนในกรณีที่มีบันทึกสูญหาย

CRON_JobAdd
งาน cron ใหม่

AT_JobAdd
งาน at ใหม่

USER_Login
การล็อกอินทั้งหมด

PORT_Locked
การล็อกทั้งหมดบนเทอร์มินัล เนื่องจากมีความพยามที่ไม่สำเร็จเกิน

ต่อไปนี้คือตัวอย่างของวิธีการสร้าง บันทึกการตรวจสอบทั่วไป:

1. ตั้งค่ารายการไฟล์สำคัญที่จะถูกมอนิเตอร์การเปลี่ยนแปลง เช่น ไฟล์ทั้งหมดใน /etc และกำหนดคอนฟิกสำหรับ เหตุการณ์ **FILE_Write** ในไฟล์ objects ดังนี้:

   ```
   find /etc -type f | awk '{printf("%s:\n\tw = FILE_Write\n\n",$1)}' >> /etc/security/audit/objects
   ```

2. ใช้คำสั่ง **auditcat** เพื่อตั้งค่าการตรวจสอบ โหมด BIN ไฟล์ /etc/security/audit/bincmds มีลักษณะคล้ายต่อไปนี้:

   ```
   /usr/sbin/auditcat -p -o $trail $bin
   ```

3. แก้ไขไฟล์ /etc/security/audit/config และเพิ่มคลาสสำหรับเหตุการณ์ที่เราสนใจ แสดงรายการผู้ใช้ที่มีอยู่ ทั้งหมด และระบุคลาส custom สำหรับผู้ใช้

   ```
   start:
           binmode = on
           streammode = off

   bin:
           cmds = /etc/security/audit/bincmds
           trail = /audit/trail
           bin1 = /audit/bin1
           bin2 = /audit/bin2
           binsize = 100000
           freespace = 100000

   classes:
           custom = FILE_Write,PROC_SetUser,AUD_Bin_Def,AUD_Lost_Rec,USER_SU, \
                    PASSWORD_Change,CRON_JobAdd,AT_JobAdd,USER_Login,PORT_Locked

   users:
           root = custom
           afx = custom
           ...
   ```

4. เพิ่มคลาสการตรวจสอบ custom ไปยังไฟล์ /usr/lib/security/mkuser.default เพื่อที่ IDs จะมีการเรียกใช้การตรวจสอบที่ถูกต้องถูกเชื่อมโยงโดยอัตโนมัติ:

   ```
   user:
       auditclasses = custom
       pgrp = staff
       groups = staff
       shell = /usr/bin/ksh
       home = /home/$USER
   ```

5. สร้างระบบไฟล์ใหม่ชื่อ /audit โดยใช้ SMIT หรือคำสั่ง **crfs** ระบบไฟล์ ควรมีขนาดใหญ่พอสำหรับเก็บค่าทั้งสอง bins และหลักฐานการตรวจสอบขนาดใหญ่
6. รันอ็อพชันคำสั่ง **audit start** และ ตรวจสอบไฟล์ /audit คุณควรเห็นไฟล์ bin ทั้งสองนี้และไฟล์ trail ว่างในตอนเริ่มต้น หลังจากคุณได้ใช้ระบบมาระยะหนึ่ง คุณควรมีบันทึก การติดตามอยู่ในไฟล์ trail ที่สามารถอ่านได้ด้วยคำสั่ง:

   ```
   auditpr  -hhelpPRtTc -v | more
   ```

ตัวอย่างนี้ใช้เหตุการณ์เพียงสองสามเหตุการณ์เท่านั้น ในการดูเหตุการณ์ทั้งหมด คุณ สามารถระบุชื่อคลาส ALL ให้แก่ผู้ใช้ทุกคน การดำเนินการนี้ จะสร้างข้อมูลขนาดใหญ่ คุณอาจต้องการเพิ่ม เหตุการณ์ทั้งหมดที่เกี่ยวข้องกับการเปลี่ยนแปลงผู้ใช้ และการเปลี่ยนแปลงสิทธิพิเศษให้แก่คลาส custom ของคุณ

การมอนิเตอร์การเข้าถึงไฟล์สำคัญ เวลาจริง:

ขั้นตอนเหล่านี้สามารถใช้เพื่อมอนิเตอร์การเข้าถึงไฟล์สำคัญ ณ เวลาจริง

ดำเนินขั้นตอนเหล่านี้:

1. ตั้งค่ารายการไฟล์สำคัญที่จะถูกมอนิเตอร์ดูการเปลี่ยนแปลง ตัวอย่างเช่นไฟล์ทั้งหมดใน /etc และตั้งค่า สำหรับเหตุการณ์ **FILE_Write** ในไฟล์ objects:

   ```
   find /etc -type f | awk '{printf("%s:\n\tw = FILE_Write\n\n",$1)}' >> /etc/security/audit/objects
   ```

2. ตั้งค่าการตรวจสอบสตรีมเพื่อแสดงการเขียนไฟล์ทั้งหมด (ตัวอย่างนี้ แสดงรายการการเขียนไฟล์ทั้งหมดไปยังคอนโซล แต่ในสภาวะแวดล้อมการทำงานจริง คุณอาจต้องการ backend ที่ส่งเหตุการณ์ไปใน Intrusion Detection System) ไฟล์ /etc/security/audit/streamcmds คล้ายคำสั่งต่อไปนี้:

   ```
   /usr/sbin/auditstream | /usr/sbin/auditselect -e "event == FILE_Write" |
   auditpr  -hhelpPRtTc -v > /dev/console &
   ```

3. ตั้งค่าการตรวจสอบโหมด STREAM ใน /etc/security/audit/config เพิ่มคลาสสำหรับเหตุการณ์การเขียนไฟล์และตั้งค่าผู้ใช้ทั้งหมดที่ ควรถูกตรวจสอบกับคลาสนั้น:

   ```
   start:
           binmode = off
           streammode = on

   stream:
           cmds = /etc/security/audit/streamcmds

   classes:
           filemon = FILE_write

   users:
           root = filemon
           afx = filemon
           ...
   ```

4. ตอนนี้รัน **audit start** เหตุการณ์ **FILE_Write** ทั้งหมด ถูกแสดงไปยังคอนโซล

การเลือกเหตุการณ์ที่จะตรวจสอบ:

วัตถุประสงค์ของการตรวจสอบคือเพื่อตรวจหากิจกรรมที่อาจเป็นอันตราย ต่อความปลอดภัยของระบบของคุณ

เมื่อดำเนินการโดยผู้ใช้ที่ได้รับอนุญาต กิจกรรมต่อไปนี้ละเมิด ความปลอดภัยระบบและอาจเป็นแคนดิเดตสำหรับการตรวจสอบ:

- การมีส่วนในกิจกรรมใน Trusted Computing Base
- การพิสูจน์ตัวตนผู้ใช้
- การเข้าถึงระบบ
- การเปลี่ยนการตั้งค่าของระบบ
- การหลีกเลี่ยงระบบการตรวจสอบ
- การเตรียมข้อมูลระบบ
- การติดตั้งโปรแกรม
- การแก้ไขบัญชีผู้ใช้
- การถ่ายโอนข้อมูลไปยังหรือออกจากระบบ

ระบบตรวจสอบไม่มีชุดของเหตุการณ์ที่เป็นค่าดีฟอลต์ที่จะถูกตรวจสอบ คุณ ต้องเลือกเหตุการณ์หรือคลาสเหตุการณ์ตามความต้องการของคุณ

ในการตรวจสอบกิจกรรม คุณต้องระบุคำสั่งหรือกระบวนการที่เริ่มต้น เหตุการณ์ตรวจสอบ และทำให้แน่ใจว่าเหตุการณ์ถูกแสดงในไฟล์ /etc/security/audit/events สำหรับระบบของคุณ จากนั้นคุณต้องเพิ่มเหตุการณ์ในคลาสที่เหมาะสม ในไฟล์ /etc/security/audit/config หรือใน stanza ของอ็อบเจ็กต์ในไฟล์ /etc/security/audit/objects ดูที่ไฟล์ /etc/security/audit/events บนระบบของคุณเพื่อดูรายการเหตุการณ์ตรวจสอบและคำแนะนำการจัดรูปแบบการรติดตาม สำหรับรายละเอียดวิธีที่รูปแบบเหตุการณ์การตรวจสอบถูกเขียน หรือใช้งาน ดูที่คำสั่ง **auditpr**

หลังจากคุณเลือกเหตุการณ์ที่จะตรวจสอบ คุณต้องรวมเหตุการณ์ที่คล้ายกัน เข้าเป็นคลาสการตรวจสอบ จากนั้นคลาสการตรวจสอบจะถูกกำหนดให้แก่ผู้ใช้

การเลือกคลาสการตรวจสอบ

คุณสามารถสนับสนุนการกำหนด เหตุการณ์การตรวจสอบให้แก่ผู้ใช้โดยการรวมเหตุการณ์ที่คล้ายกันเข้าเป็นคลาสการตรวจสอบ คลาส การตรวจสอบเหล่านี้ถูกกำหนดใน stanza คลาสของไฟล์ /etc/security/audit/config

โดยทั่วไป บางคลาสการตรวจสอบอาจเป็นดังนี้:

ทั่วไป เหตุการณ์ที่เปลี่ยนสถานะของระบบและเปลี่ยนการพิสูจน์ตัวตนผู้ใช้ การตรวจสอบพยายามหลีกเลี่ยงการควบคุมการเข้าถึงระบบ

อ็อบเจ็กต์
การเข้าถึงเพื่อเขียนไฟล์คอนฟิกูเรชันความปลอดภัย

เคอร์เนล
เหตุการณ์ในคลาสเคอร์เนลถูกสร้างโดยฟังก์ชันการจัดการ กระบวนการของเคอร์เนล

ตัวอย่างของ stanza ในไฟล์ /etc/security/audit/config เป็นดังนี้:

```
classes:
    general = USER_SU,PASSWORD_Change,FILE_Unlink,FILE_Link,FILE_Rename
    system = USER_Change,GROUP_Change,USER_Create,GROUP_Create
    init = USER_Login,USER_Logout
```

การเลือกวิธีการรวบรวมข้อมูลการตรวจสอบ

การเลือก วิธีการรวบรวมข้อมูลของคุณขึ้นอยู่กับวิธีที่คุณต้องการใช้ข้อมูลการตรวจสอบ ถ้าคุณต้องการพื้นที่จัดเก็บข้อมูลระยะยาวที่เก็บข้อมูลขนาดใหญ่ เลือกการรวบรวมแบบ BIN ถ้าคุณต้องการประมวลผลข้อมูลขณะที่ถูกรวบรวม เลือกการรวบรวมแบบ STREAM ถ้าคุณจำเป็นต้องใช้ทั้งพื้นที่จัดเก็บข้อมูลระยะยาว และการประมวลผลทันที เลือกทั้งสองวิธี คำอธิบายของแต่ละเมธอดเหล่านี้เป็นดังนี้:

การรวบรวมแบบ Bin

อนุญาตให้มีพื้นที่จัดเก็บข้อมูลการติดตามการตรวจสอบขนาดใหญ่เป็นระยะเวลานาน เร็กคอร์ดการตรวจสอบ ถูกเขียนลงไฟล์ที่เป็น bin ชั่วคราว หลังจากเก็บลงไฟล์ ข้อมูลจะถูกประมวลผลโดย **auditbin** daemon ขณะระบบย่อยการตรวจสอบ เขียนลงไฟล์ bin อื่น และเร็กคอร์ดถูกเขียนลงไฟล์การติดตามการตรวจสอบ สำหรับจัดเก็บ

การรวบรวมแบบ Stream

อนุญาตให้มีการประมวลผลข้อมูลการตรวจสอบขณะที่ถูกรวบรวม เร็กคอร์ดการตรวจสอบ ถูกเขียนลงในบัฟเฟอร์วนซ้ำภายในเคอร์เนล และถูกเรียกข้อมูลโดย การอ่าน `/dev/audit` เร็กคอร์ดการตรวจสอบสามารถแสดง พิมพ์เพื่อให้มีการติดตามการตรวจสอบบนกระดาษ หรือแปลงเป็นเร็กคอร์ด bin โดยใช้คำสั่ง **auditcat**

การตรวจสอบพาร์ติชันเวิร์กโหลด

การตรวจสอบสามประเภทมีให้ใช้ในสภาวะแวดล้อม WPAR: โกลบอล ระบบ และการตรวจสอบจากโกลบอล

คุณสามารถเปิดใช้การตรวจสอบใน WPAR โกลบอล ภายใน WPAR หรือทั้งสอง การตั้งค่าการตรวจสอบสำหรับ WPAR ระบบและ WPAR โกลบอลนั้นเหมือนกับ การตั้งค่าในสภาวะแวดล้อมที่มิใช่ wpar คุณสามารถเริ่มการตรวจสอบ WPAR โกลบอลสำหรับ WPAR ระบบและแอ็พพลิเคชัน

หมายเหตุ: การตรวจสอบสำหรับ WPAR แอ็พพลิเคชันไม่สามารถเริ่มจากภายใน WPAR แต่สามารถ เริ่มโดยใช้การตรวจสอบ WPAR โกลบอล

การตรวจสอบ WPAR โกลบอลช่วยให้ผู้ดูแลระบบโกลบอลตรวจสอบ WPARs จากระบบโกลบอล ผู้ดูแลระบบโกลบอลสามารถควบคุม ระดับของการตรวจสอบสำหรับแต่ละ WPAR ได้จากที่เดียวโดยการระบุ คลาสที่จะตรวจสอบสำหรับแต่ละ WPAR ในไฟล์ `/etc/security/audit/config` โกลบอล

โดยการเพิ่ม WPARS stanza ในไฟล์ `/etc/security/audit/config` ผู้ดูแลระบบโกลบอลสามารถจัดให้มีรายการคลาสที่จะ ตรวจสอบสำหรับ WPAR ตัวอย่าง:

```
WPARS:
<wpar_name> = <auditclass>, ... <auditclass>
```

ในตัวอย่างที่ผ่านมา <wpar_name> ต้องเป็นชื่อ WPAR ของระบบ และแต่ละพารามิเตอร์ auditclass ควรกำหนดใน stanza คลาส

ในการตั้งค่าการตรวจสอบของ testwpar WPAR ด้วยคลาสทั่วไป tcpip และ lvm ให้เพิ่ม stanza ต่อไปนี้ในไฟล์ `/etc/security/audit/config`:

```
WPARS:
testwpar = general,tcpip,lvm
```

ผู้ดูแลระบบโกลบอลสามารถเริ่มและหยุดทำงานการตรวจสอบบน WPAR โดยใช้คำสั่ง **audit** และระบุ ชื่อ WPAR ดังนี้:

```
audit start -@ <wparname1> -@ <wparname2> ...
audit shutdown -@ <wparname1> -@ <wparname2> ...
```

คุณสามารถตรวจสอบอ็อบเจ็กต์ WPAR จากสภาวะแวดล้อม โกลบอลโดยการระบุพาธสัมบูรณ์ไปยังอ็อบเจ็กต์ที่คุณ ต้องการตรวจสอบ ตัวอย่างเช่น, เมื่อต้องการนิยามเหตุการณ์การตรวจสอบสำหรับไฟล์ /wpars/wpar1/etc/security/passwd, ให้เพิ่ม stanza ต่อไปนี้ให้กับไฟล์ /etc/security/audit/objects ในระบบ AIX ที่ถูกโฮสต์ WPAR:

```
/wpars/wpar1/etc/security/passwd:
     r = "WPAR1_PASSWD_RD"
     w = "WPAR1_PASSWD_WR"
```

stanza ก่อนหน้านี้ถูกแยกวิเคราะห์เมื่อเริ่มการ ตรวจสอบ (-@ <wpar1>) เวลาที่เปิดใช้การตรวจสอบอ็อบเจ็กต์สำหรับอ็อบเจ็กต์ /etc/security/passwd ของ wpar1 แอ็ตทริบิวต์เหล่านี้สร้างเหตุการณ์การตรวจสอบ WPAR1_PASSWD_RD ในทุกครั้งที่ไฟล์ /wpars/wpar1/etc/security/passwd ถูกอ่าน แอ็ตทริบิวต์เหล่านี้ยังสร้างเหตุการณ์การตรวจสอบ WPAR1_PASSWD_WR ในทุกครั้งที่ไฟล์ถูกเปิดเพื่อทำการเขียน

หมายเหตุ: คุณต้องเปิดใช้การตรวจสอบ สำหรับสภาวะแวดล้อมโกลบอลก่อนที่คุณเปิดใช้การตรวจสอบ WPAR จาก สภาวะแวดล้อมโกลบอล

คำสั่ง **auditpr** สามารถใช้เพื่อสร้าง รายงานการตรวจสอบที่แสดงชื่อ WPAR ตัวอย่าง:

```
auditpr -v < /audit/trail
```

การตรวจสอบในสภาพแวดล้อม NFS

ระบบย่อยการตรวจสอบ AIX สนับสนุนการตรวจสอบของระบบไฟล์ที่ติดตั้ง การกำหนดคอนฟิกของระบบไฟล์ที่ติดตั้งในไคลเอ็นต์ เหมือนกับระบบไฟล์โลคัล การดำเนินการตรวจสอบ อ็อบเจ็กต์ที่ติดตั้งที่สามารถตรวจสอบได้เหมือนกับอ็อบเจ็กต์โลคัลที่อธิบายไว้ใน ภาพรวมการตรวจสอบ พฤติกรรมการตรวจสอบในไคลเอ็นต์ และเซิร์ฟเวอร์สำหรับ ระบบไฟล์ที่ติดตั้งอธิบายไว้ภายหลังในข้อมูลในหัวเรื่องนี้

การตรวจสอบไคลเอ็นต์ NFS

การดำเนินการ ทั้งหมดรันบนอ็อบเจ็กต์ที่ตรวจสอบได้ ซึ่งอยู่บนระบบไฟล์ที่ติดตั้งโดย ไคลเอ็นต์ที่ล็อกออนไคลเอ็นต์ นี่สามารถใช้ได้โดยไม่มี การดำเนินการบนอ็อบเจ็กต์โดยเซิร์ฟเวอร์ NFS หรือไคลเอ็นต์ NFS อื่นๆ หรือตรวจสอบพาธเต็มต้องเปิดใช้งานบนไคลเอ็นต์

อ้างอิงคำสั่ง **audit** man page สำหรับข้อมูลเพิ่มเติม ถ้าการตรวจสอบพาธเต็มไม่ถูกเปิดใช้งาน และไฟล์ถูกแก้ไขโดยเซิร์ฟเวอร์หรือไคลเอ็นต์อื่น การตรวจสอบที่ตามมา จะไม่สามารถคาดเดาได้ พฤติกรรมนี้สามารถแก้ไขได้โดย รีสตาร์ทการตรวจสตาร์ทบนไคลเอ็นต์ ถ้าระบบไฟล์ถูกติดตั้งไว้ในหลายๆ ไคลเอ็นต์ เราขอแนะนำให้คุณตรวจสอบกการดำเนินการบนเซิร์ฟเวอร์เพื่อรับล็อกเหตุการณ์จริง หรือเปิดใช้การตรวจสอบพาธเต็ม บนไคลเอ็นต์

หมายเหตุ: คอนฟิกูเรชันระบบย่อยการตรวจสอบไม่สนับสนุน การใช้ระบบล็อกไฟล์การตรวจสอบเป็นระบบไฟล์ NFS ที่ติดตั้ง

การตรวจสอบบนเซิร์ฟเวอร์ NFS

การดำเนินการทั้งหมด ดำเนินการต่อไปบนระบบไฟล์ที่ติดตั้งโดยทั้งไคลเอ็นต์และเซิร์ฟเวอร์ ล็อกออนเข้าสู่เซิร์ฟเวอร์ NFS

ข้อจำกัดของฝั่งเซิร์ฟเวอร์

- ถ้าการดำเนินการใดๆ ทำต่อเนื่องไปโดยไคลเอ็นต์ NFS ไม่ส่งไปถึงเซิร์ฟเวอร์ เนื่องจากการแคชข้อมูลของ NFS หรือเนื่องจากสถาปัตยกรรมที่สืบทอดของ NFS การดำเนินการนั้นจะไม่ถูกตรวจสอบโดยเซิร์ฟเวอร์

  ตัวอย่าง: หลังจากติดตั้งไฟล์ระบบ เฉพาะการดำเนินการอ่านไฟล์ครั้งแรกเท่านั้น ที่ถูกตรวจสอบโดยเซิร์ฟเวอร์ การดำเนินการอ่านในครั้งต่อไปไม่ถูกล็อกออนเข้าสู่เซิร์ฟเวอร์ ข้อจำกัดนี้นำไปใช้กับการดำเนินการอ่านไฟล์ ลิงก์ และไดเร็กทอรีด้วย
- การทำงานที่ดำเนินการโดยไคลเอ็นต์ถูกล็อกออนในเซิร์ฟเวอร์ เป็น **nfsd** และมีผู้ใช้ **root** เป็นชื่อผู้ใช้

ตัวอย่าง

ระบบไฟล์ที่ชื่อ *File_System* ถูกติดตั้ง บนไคลเอ็นต์ด้วยคำสั่ง **mount server:/File_system /mnt** ถ้าไฟล์ชื่อ *A* ในระบบไฟล์ *File_System* จำเป็นต้องตรวจสอบในเซิร์ฟเวอร์ แล้ว `/File_system/A` ต้องกำหนดคอนฟิกในไฟล์คอนฟิกกูเรชันการตรวจสอบ

ถ้าคุณตัดสินใจที่จะ ตรวจสอบไฟล์ *A* ในระบบไฟล์ *File_System* บนไคลเอ็นต์ แล้ว `/mnt/A` ต้องถูกกำหนดคอนฟิก ให้ตรวจสอบในไคลเอ็นต์

ถ้าไฟล์ *A* ถูกกำหนดคอนฟิก ให้ตรวจสอบทั้งในเซิร์ฟเวอร์และไคลเอ็นต์ แล้วการดำเนินการที่ทำต่อไป ทั้งเซิร์ฟเวอร์และไคลเอ็นต์กับไฟล์ *A* จะถูกตรวจสอบ และล็อกออนเข้าสู่เซิร์ฟเวอร์ และดำเนินการต่อไปโดยไคลเอ็นต์ที่ล็อกออน เข้าสู่ไคลเอ็นต์

การดำเนินการใดๆ ที่ทำต่อไปโดยไคลเอ็นต์ในไฟล์ *A* จะถูกล็อกออนเข้าสู่เซิร์ฟเวอร์เป็น **nfsd** daemon แทนการดำเนินการหรือชื่อคำสั่ง

Lightweight Directory Access Protocol

Lightweight Directory Access Protocol (LDAP) กำหนดวิธีมาตรฐาน สำหรับการเข้าถึงและการอัพเดตข้อมูลในไดเร็กทอรี (ฐานข้อมูล) แบบโลคัลหรือแบบรีโมตอย่างใดอย่างหนึ่งในโมเดลไคลเอ็นต์–เซิร์ฟเวอร์

โปรโตคอลได้รับการอ็อพติไมซ์สำหรับการอ่าน การเรียกดู และการค้นหา ไดเร็กทอรี และเริ่มแรกถูกพัฒนาขึ้นเป็นโปรแกรมส่วนหน้าที่ขนาดเล็ก (lightweight) ไปยัง X.500 Directory Access Protocol วิธี LDAP ถูกใช้โดยคลัสเตอร์ของโฮสต์ เพื่ออนุญาตให้มีการพิสูจน์ตัวตนด้านความปลอดภัยจากศูนย์กลางรวมถึงการเข้าถึงข้อมูล ผู้ใช้และกลุ่ม ฟังก์ชันการทำงานนี้มีจุดมุ่งหมายเพื่อใช้ในสภาพแวดล้อมแบบคลัสเตอร์ เพื่อเก็บรักษาข้อมูลการพิสูจน์ตัวตน ผู้ใช้ และกลุ่มร่วมกับของทั้ง คลัสเตอร์

อ็อบเจ็กต์ใน LDAP ถูกเก็บอยู่ในโครงสร้างแบบลำดับขั้นที่รู้จักในชื่อ Directory Information Tree (DIT) ไดเร็กทอรีที่ดีจะเริ่มต้นด้วยการออกแบบ DIT อย่างมีโครงสร้าง DIT ควรได้รับการออกแบบอย่างระมัดระวังก่อนนำ LDAP ไปใช้ เป็นวิธีในการการพิสูจน์ตัวตน

โหลดโมดูลการพิสูจน์ตัวตน LDAP

การใช้ประโยชน์ LDAP ของระบบย่อยการรักษาความปลอดภัยที่ถูกนำไปใช้ เป็นโหลดโมดูลการพิสูจน์ตัวตน LDAP โดยความคิดแล้วเหมือนกับ โหลดโมดูลอื่นๆ เช่น NIS, DCE และ KRB5 โหลดโมดูลถูกกำหนดใน ไฟล์ `/usr/lib/security/methods.cfg`

โหลดโมดูล LDAP จัดให้มีการพิสูจน์ตัวตนผู้ใช้และฟังก์ชันการจัดการผู้ใช้และ กลุ่มแบบรวมศูนย์ผ่านโปรโตคอล LDAP ผู้ใช้ที่กำหนดบน เซิร์ฟเวอร์ LDAP สามารถถูกตั้งค่าให้ล็อกอินเข้าสู่ไคลเอ็นต์ LDAP แม้ผู้ใช้ จะไม่ได้ถูกกำหนดแบบสามารถโลคัล

โหลดโมดูล AIX LDAP ถูกรวมเข้ากับระบบปฏิบัติการ AIX โดยสมบูรณ์ หลังจากโหลดโมดูลการพิสูจน์ตัวตน LDAP ถูกเปิดใช้งานเพื่อให้บริการข้อมูลผู้ใช้และ กลุ่ม, APIs ระบบสูง คำสั่ง และเครื่องมือการจัดการระบบ ที่ทำงานในลักษณะปกติ แฟล็ก **-R** ถูกนำใช้สำหรับคำสั่งระดับสูง เพื่อทำงานผ่านโหลดโมดูลที่ต่างกัน ตัวอย่าง ในการสร้าง ผู้ใช้ LDAP ชื่อ *joe* จากเครื่องไคลเอ็นต์ ให้ใช้คำสั่งต่อไปนี้:

```
mkuser -R LDAP joe
```

หมายเหตุ: แม้ว่าโครงสร้างพื้นฐาน LDAP จะสามารถสนับสนุนจำนวนผู้ใช้ในกลุ่มได้ไม่จำกัดจำนวน โดยสร้างไว้สูงสุด 25 000 คนในหนึ่งกลุ่ม และมีการทดสอบโดยการทำงานที่ต่างกันในกลุ่มนั้น บางส่วน ของอินเตอร์เฟส POSIX ประวัติอาจไม่ส่งคืนข้อมูลโดยสมบูรณ์ ของกลุ่ม อ้างอิงเอกสารคู่มือของ API แต่ละตัวเพื่อดูข้อจำกัดเหล่านั้น

การพิสูจน์ตัวตนฐาน LDAP:

มีข้อกำหนดบน entities ต่างๆ กันอันเป็นส่วนหนึ่งของ การพิสูจน์ตัวตนฐาน LDAP บน AIX

โปรดทราบว่าโดยตัวโครงสร้างพื้นฐาน LDAP เองไม่ได้ระบุข้อจำกัดใดๆ ของเนื้อหาฐานข้อมูล อย่างไรก็ตาม ในส่วนนี้ให้ข้อมูลผลลัพธ์ ที่ได้จากการตั้งค่าการทดสอบเพื่อใช้เป็นข้อจำกัด ข้อจำกัดต่อไปนี้ ได้รับการทดสอบตามการพิสูจน์ตัวตนแบบอิง LDAP บนระบบปฏิบัติการ AIX:

จำนวนผู้ใช้ทั้งหมด: สูงสุด 500 000 คนได้ถูกสร้างขึ้น บนระบบเดียว และได้รับการทดสอบการพิสูจน์ตัวตนผู้ใช้พร้อมกันทีละหลายร้อยคน

จำนวนกลุ่มทั้งหมด: สูงสุด 500 กลุ่มได้ถูกสร้างขึ้น บนระบบเดียวและได้รับการทดสอบ

จำนวนผู้ใช้ต่อกลุ่มสูงสุด: สูงสุด 25 000 คนได้ถูกสร้างขึ้น ในกลุ่มเดียวและมีการทดสอบโดยการทำงานที่ต่างกัน ในกลุ่มนั้น

บางส่วนของอินเตอร์เฟส POSIX ประวัติอาจไม่ส่งคืนข้อมูลโดยสมบูรณ์ ของกลุ่ม อ้างอิงเอกสารคู่มือของ API แต่ละตัวเพื่อดูข้อจำกัดเหล่านั้น รวมทั้งค่าด้านบนยึดตามการทดสอบ ที่กระทำ โดยไม่ได้ขัดขวางความเป็นไปได้ที่จะสามารถตั้งค่า ระบบที่มีผู้ใช้และกลุ่มจำนวนมากที่จัดให้มีรีซอร์สที่จำเป็น อยู่

การตั้งค่าเซิร์ฟเวอร์ข้อมูลความปลอดภัย IBM Tivoli Directory Server:

เมื่อต้องการตั้งค่าระบบเป็นเซิร์ฟเวอร์ข้อมูลความปลอดภัย LDAP ที่ใช้การพิสูจน์ตัวตน, ผู้ใช้, และข้อมูลกลุ่ม ผ่าน LDAP, คุณต้องติดตั้งเซิร์ฟเวอร์ LDAP และแพ็กเกจไคลเอ็นต์

ถ้า Secure Sockets Layer (SSL) จำเป็นต้องมี, คุณยังต้องติดตั้งแพ็กเกจ **GSKitV7** สำหรับ IBM Tivoli Directory Server เวอร์ชัน 6.2, หรือก่อนหน้านี้, หรือ GSKitV8 สำหรับ IBM Tivoli Directory Server เวอร์ชัน 6.3, หรือเวอร์ชันถัดมา ผู้ดูแลระบบต้องสร้างคีย์โดยใช้คำสั่งการจัดการคีย์ GSKit คำสั่งนี้คือ **gsk7ikm** ใน GSKitV7 หรือคำสั่ง **ikeyman** ที่มี GSKitV8 สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ การตั้งค่าเซิร์ฟเวอร์เพื่อใช้ SSL ดูที่ Secure Communication with SSL

รันคำสั่ง **mksecldap** เพื่อกำหนดคอนฟิกเซิร์ฟเวอร์ คำสั่ง **mksecldap** สร้างเซิร์ฟเวอร์ LDAP และฐานข้อมูลแบบ back-end ที่ชื่อ *ldapdb2*, ระบุเซิร์ฟเวอร์ LDAP ด้วยข้อมูลผู้ใช้และกลุ่มจากโลคัลโฮสต์, และตั้งค่าผู้ดูแลระบบเซิร์ฟเวอร์ LDAP DN (ชื่อจำเพาะ) และรหัสผ่าน ทางเลือก สามารถตั้งค่า SSL สำหรับการสื่อสารไคลเอ็นต์/เซิร์ฟเวอร์ คำสั่ง **mksecldap** ยังเพิ่มรายการ ลงในไฟล์ `/etc/inittab` เพื่อเริ่มต้นเซิร์ฟเวอร์ LDAP เมื่อรีบู๊ตทุกครั้ง

ผู้ใช้และกลุ่ม AIX ถูกเก็บอยู่ในเซิร์ฟเวอร์ LDAP โดยใช้หนึ่งในสกีมาต่อไปนี้:

AIX schema

รวมคลาสอ็อบเจ็กต์ aixAccount และ aixAccessGroup schema นี้มีชุดแอ็ตทริบิวต์ทั้งหมดสำหรับผู้ใช้และกลุ่ม AIX

สกีมา RFC 2307

สอดแทรกคลาสอ็อบเจ็กต์ posixAccount, shadowAccount, และ posixGroup และถูกใช้โดยผลิตภัณฑ์ ไดเร็กทอรีของผู้จำหน่ายต่างๆ RFC 2307 schema กำหนดเฉพาะเซ็ตย่อย ของแอ็ตทริบิวต์ขนาดเล็กที่ AIX ใช้เท่านั้น

RFC2307AIX schema

รวมคลาสอ็อบเจ็กต์ posixAccount, shadowAccount และ posixGroup บวกกับคลาสอ็อบเจ็กต์ aixAuxAccount และ aixAuxGroup คลาสอ็อบเจ็กต์ aixAuxAccount and aixAuxGroup มีแอ็ตทริบิวต์ซึ่งใช้โดย AIX แต่ไม่ถูกกำหนดโดย RFC 2307 schema

การใช้ประเภท RFC2307AIX schema สำหรับผู้ใช้และ กลุ่มแนะนำให้ใช้เป็นอย่างมาก ชนิดสกีมา RFC2037AIX เข้ากันได้กับ RFC 2307 ที่มีแอ็ตทริบิวต์พิเศษเพื่อสนับสนุนการทำงานการจัดการผู้ใช้ AIX เพิ่มเติม เซิร์ฟเวอร์ IBM Tivoli Directory Server ที่มีคอนฟิกูเรชันสกีมา RFC2307AIX ไม่สนับสนุนไคลเอ็นต์ AIX LDAP เท่านั้น, แต่ยัง เข้ากันได้กับ RFC 2307 UNIX และไคลเอ็นต์ Linux LDAP

ข้อมูลผู้ใช้และกลุ่มทั้งหมดถูกเก็บอยู่ภายใต้แผนผัง AIX (ส่วนต่อท้าย) คำต่อท้ายดีฟอลต์คือ "cn=aixdata" คำสั่ง **mksecldap** ยอมรับคำต่อท้ายที่ผู้ใช้ระบุผ่านแฟล็ก **-d** ชื่อสำหรับ แผนผังย่อยที่ต้องถูกสร้างขึ้นสำหรับผู้ใช้, กลุ่ม, ID, และอื่นๆ, ถูกควบคุมโดย ไฟล์คอนฟิกูเรชัน sectoldif.cfg อ้างถึงไฟล์ sectoldif.cfg สำหรับ ข้อมูลเพิ่มเติม

แผนผัง AIX ACL (Access Control List) ถูกป้องกันไว้ ACL ดีฟอลต์อนุญาตให้สิทธิ์พิเศษการดูแลจัดการ เฉพาะ entity ที่ระบุเป็นผู้ดูแลระบบเท่านั้นกับอ็อพชันคำสั่ง **-a** สิทธิ์พิเศษเพิ่มเติมสามารถอนุญาตให้แก่ identity พร็อกซีถ้า อ็อพชันคำสั่ง **-x** และ **-X** ถูกใช้ การใช้อ็อพชันเหล่านี้ สร้าง identity พร็อกซีและกำหนดคอนฟิกสิทธิ์พิเศษ ตามที่กำหนดในไฟล์ /etc/security/ldap/proxy.ldif.template การสร้าง proxy identity อนุญาตให้ไคลเอ็นต์ LDAP โยงกับ เซิร์ฟเวอร์โดยไม่มีการใช้ identity ผู้ดูแลระบบ, ซึ่งจำกัด สิทธิพิเศษของผู้ดูแลระบบบนเซิร์ฟเวอร์ LDAP

คุณสามารถรันคำสั่ง **mksecldap** บนเซิร์ฟเวอร์ LDAP ที่ตั้งค่าสำหรับวัตถุประสงค์อื่นๆ; ตัวอย่างเช่น, สำหรับข้อมูลการค้นหา ID ผู้ใช้ในตัวอย่างนี้, **mksecldap** เพิ่มแผนผัง AIX และระบุด้วยข้อมูลความปลอดภัย AIX ให้กับเซิร์ฟเวอร์ LDAP ที่มีอยู่เดิม แผนผังนี้ป้องกันด้วย ACL ที่เป็นอิสระจากแผนผังที่มีอยู่

หมายเหตุ: คุณควรสำรองเซิร์ฟเวอร์ LDAP ที่มีอยู่ ก่อนที่คุณจะรันคำสั่ง **mksecldap** และขยายเซิร์ฟเวอร์ ไปเป็นเซิร์ฟเวอร์ข้อมูลความปลอดภัย AIX

หลังจากที่เซิร์ฟเวอร์ข้อมูลความปลอดภัย LDAP ติดตั้งเป็นผลสำเร็จ, คุณสามารถตั้งค่าโฮสต์ที่เหมือนกันกับไคลเอ็นต์ เพื่อจัดการกับผู้ใช้และกลุ่ม LDAP และอนุญาตให้ผู้ใช้ LDAP ล็อกออนเข้าสู่ เซิร์ฟเวอร์นี้

ถ้าการติดตั้ง เซิร์ฟเวอร์ข้อมูลการรักษาความปลอดภัย LDAP ไม่สำเร็จ คุณ สามารถยกเลิกการติดตั้งได้โดยการรันคำสั่ง **mksecldap** ด้วยแฟล็ก **-U** คำสั่งนี้จะเรียกคืนไฟล์ ibmslapd.conf (หรือ slapd.conf หรือ slapd32.conf) ไปเป็นสถานะก่อนติดตั้ง รันคำสั่ง **mksecldap** ด้วยแฟล็ก **-U** หลังพยายามทำการติดตั้งไม่สำเร็จก่อน ลองรันคำสั่ง **mksecldap** อีกครั้ง มิฉะนั้น ข้อมูลการติดตั้งที่มีอยู่อาจยังคงอยู่ในไฟล์คอนฟิกูเรชัน และทำสาเหตุให้การติดตั้งภายหลังล้มเหลว เพื่อเป็นการป้องกันความปลอดภัย อ็อพชัน เลิกทำ จะไม่ทำสิ่งใดกับฐานข้อมูล หรือข้อมูล เนื่องจาก ฐานข้อมูลอาจมีอยู่ก่อนการรันคำสั่ง **mksecldap** ลบฐานข้อมูลใดๆ ด้วยตนเอง ถ้าถูกสร้างขึ้นโดยคำสั่ง **mksecldap** ถ้าคำสั่ง **mksecldap** ได้เพิ่มข้อมูลในฐานข้อมูลที่มีอยู่ ก่อน ให้ตัดสินใจว่าจะใช้ขั้นตอนใดเรียกคืนจากการพยายามติดตั้ง ที่ล้มเหลว

หลักการที่เกี่ยวข้อง:

การสื่อสารอย่างปลอดภัยด้วย SSL
โดยขึ้นอยู่กับประเภทการพิสูจน์ตัวตนที่ใช้ระหว่างไคลเอ็นต์ LDAP และเซิร์ฟเวอร์ รหัสผ่านถูกส่งในรูปแบบที่เข้ารหัส (unix_auth) หรือแบบข้อมูลธรรมดา (ldap_auth) อย่างใดอย่างหนึ่ง ใช้ Secure Socket Layer (SSL) เพื่อป้องกันจาก การเปิดเผยความปลอดภัยแม้เมื่อคุณส่งรหัสผ่านที่เข้ารหัสบนเน็ตเวิร์ก หรือในบางกรณีบนอินเตอร์เน็ต AIX มี แพ็กเกจสำหรับ SSL ที่สามารถให้มีการสื่อสารอย่างปลอดภัยระหว่างไดเร็กทอรีเซิร์ฟเวอร์ และไคลเอ็นต์

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง **mksecldap**

การตั้งค่าไคลเอ็นต์ LDAP:

ในการตั้งค่าไคลเอ็นต์เพื่อใช้ LDAP สำหรับการพิสูจน์ตัวตน และข้อมูลผู้ใช้/กลุ่ม ตรวจสอบให้แน่ใจว่าแต่ละไคลเอ็นต์มีแพ็กเกจไคลเอ็นต์ LDAP ติดตั้งอยู่ ถ้าต้องการ Secure Sockets Layer (SSL), GSKit ต้องถูกติดตั้งไว้, คีย์ต้องถูกสร้าง, และเซิร์ฟเวอร์ LDAP ใบรับรองคีย์ SSL ต้องถูกเพิ่มไว้ในคีย์นี้

คล้ายกับการตั้งค่าเซิร์ฟเวอร์ LDAP การตั้งค่าไคลเอ็นต์สามารถกระทำได้โดย ใช้คำสั่ง **mksecldap** ในการให้ไคลเอ็นต์นี้ติดต่อกับ LDAP security information server ชื่อเซิร์ฟเวอร์ต้องถูกระบุระหว่างการเซ็ตอัพ การโยง DN และรหัสผ่านของเซิร์ฟเวอร์ยังต้องมีสำหรับให้ไคลเอ็นต์เข้าถึงแผนผัง AIX บนเซิร์ฟเวอร์ คำสั่ง **mksecldap** บันทึก DN การโยงของเซิร์ฟเวอร์ รหัสผ่าน ชื่อเซิร์ฟเวอร์ แผนผัง AIX DN บนเซิร์ฟเวอร์ พาธและรหัสผ่านคีย์ SSL และแอ็ตทริบิวต์การตั้งค่าอื่นๆ ไปยังไฟล์ `/etc/security/ldap/ldap.cfg`

คำสั่ง **mksecldap** บันทึกการโยงรหัสผ่านและรหัสผ่านคีย์ SSL (ถ้าคุณกำลังกำหนดคอนฟิก SSL) ไปยังไฟล์ `/etc/security/ldap/ldap.cfg` ในรูปแบบ ที่เข้ารหัสแล้ว รหัสผ่านที่เข้ารหัสเป็นค่าเฉพาะของระบบ และสามารถใช้โดย **secldapclntd** daemon บนระบบที่ รหัสผ่านนั้นถูกสร้างขึ้นเท่านั้น **secldapcIntd** daemon สามารถใช้งาน ข้อความปกติ หรือรหัสผ่านที่เข้ารหัสจากไฟล์ `/etc/security/ldap/ldap.cfg`

หลาย เซิร์ฟเวอร์สามารถถูกจัดเตรียมให้กับคำสั่ง **mksecldap** ระหวางเซ็ตอัพไคลเอ็นต์ ในกรณีนี้ ไคลเอ็นต์ติดต่อเซิร์ฟเวอร์ตามลำดับที่ให้ และสร้างการเชื่อมต่อไปยังเซิร์ฟเวอร์แรก ที่ไคลเอ็นต์สามารถโยงได้สำเร็จ ถ้ามีข้อผิดพลาดการเชื่อมต่อเกิดขึ้น ระหว่างไคลเอ็นต์และเซิร์ฟเวอร์ จะมีการพยายามส่งการร้องขอการเชื่อมต่อใหม่ โดยใช้ตรรกะแบบเดิม โมเดลการใช้ประโยชน์ Security LDAP ไม่ สนับสนุนการอ้างอิง สิ่งสำคัญคือเซิร์ฟเวอร์ที่เป็นสำเนาควรถูกเก็บโดย มีการซิงโครไนซ์

ไคลเอ็นต์สื่อสารกับเซิร์ฟเวอร์ข้อมูลการรักษาความปลอดภัย LDAP ผ่าน daemon ฝั่งไคลเอ็นต์ (**secldapcIntd**) ถ้าโหลดโมดูล LDAP ถูกเปิดใช้งานบนไคลเอ็นต์ คำสั่งระดับสูง จะถูกกำหนดเส้นทางไปยัง daemon ผ่าน APIs ไลบรารีสำหรับผู้ใช้ที่ระบุใน LDAP daemon คอยดูแลรักษาแคชของรายการ LDAP ที่ร้องขอ ถ้า การร้องขอไม่ตรงตามความต้องการของแคช daemon จะเคียวรี เซิร์ฟเวอร์ อัพเดตแคช และส่งข้อมูลกลับไปที่ ผู้เรียกใช้

อ็อพชันการปรับให้ใช้งานได้เต็มที่อื่นๆ สามารถกำหนดให้แก่คำสั่ง **mksecldap** ระหว่างการเซ็ตอัพไคลเอ็นต์ เช่นการตั้งค่าจำนวนเธรดที่จะใช้โดย daemon ขนาดรายการแคช และการหมดเวลาหมดอายุแคช อ็อพชันเหล่านี้สำหรับผู้ใช้ที่มีประสบการณ์เท่านั้น สำหรับสภาวะแวดล้อมโดยส่วนใหญ่ ใช้ค่าดีฟอลต์ก็เพียงพอ

ในขั้นตอนสุดท้ายของ เซ็ตอัพไคลเอ็นต์ คำสั่ง **mksecldap** เริ่มทำงาน daemon ฝั่งไคลเอ็นต์และเพิ่มรายการในไฟล์ `/etc/inittab` ดังนั้น daemon จะเริ่มทำงานในทุกครั้งที่ทำการรีบูต คุณสามารถตรวจสอบว่าเซ็ตอัพทำสำเร็จหรือไม่โดยการตรวจสอบ การประมวลผล **secldapcIntd** daemon ผ่านคำสั่ง **ls-secldapclntd** โดยจัดให้มี LDAP security information server ถูกเซ็ตอัพและกำลังทำงาน daemon นี้ จะกำลังทำงานอยู่ถ้าการเซ็ตอัพทำสำเร็จ

เซิร์ฟเวอร์ต้อง ถูกตั้งค่าก่อนไคลเอ็นต์ การเซ็ตอัพไคลเอ็นต์ขึ้นอยู่กับข้อมูลที่โอนย้ายที่จะนำมาไว้ในเซิร์ฟเวอร์ ทำตามขั้นตอนเหล่านี้เพื่อตั้งค่าไคลเอ็นต์:

1. ติดตั้งชุดไฟล์ไคลเอ็นต์ IBM Tivoli Directory Server บนระบบปฏิบัติการ AIX
   - บน IBM Tivoli Directory Server 5.2 ให้ติดตั้งชุดไฟล์ `ldap.client`
   - บน IBM Tivoli Directory Server 7.1 ให้ติดดตั้งชุดไฟล์ `idsldap`
2. ในการตั้งค่าไคลเอ็นต์ LDAP ให้รันคำสั่งต่อไปนี้:

   ```
   # mksecldap -c -h server1.ibm.com -a cn=admindn -p adminpwd -d cn=basedn
   ```

   แทนค่าด้านบนเป็นเหมาะสำหรับสภาวะแวดล้อมของคุณ

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mksecldap

คำสั่ง secldapclntd

การเปิดใช้งานไคลเอ็นต์สำหรับ LDAP netgroups:

คุณสามารถใช้ netgroups เป็นส่วนหนึ่งของ NIS–LDAP (วิธีการระบุ ชื่อ)

ดำเนินขั้นตอนต่อไปนี้สำหรับการเปิดใช้งานไคลเอ็นต์สำหรับ LDAP netgroups:

1. ติดตั้งและตั้งค่าการจัดการกลุ่มผู้ใช้ตาม LDAP ดัง แสดงรายละเอียดใน “การตั้งค่าไคลเอ็นต์ LDAP” ในหน้า 168 ถ้าการตั้งค่า netgroup ไม่สำเร็จ ผู้ใช้ที่กำหนด LDAP ใดๆ จะถูแสดงโดยระบบ ตัวอย่าง ถ้า *nguser* เป็น ผู้ใช้ netgroup เป็นสมาชิกของ netgroup *mygroup* ที่กำหนดแล้วใน เซิร์ฟเวอร์ LDAP ดังนั้น `lsuser -R LDAP nguser` จะแสดงรายการ ผู้ใช้
2. ในการเปิดใช้งานฟังก์ชัน netgroup นิยามโมดูล สำหรับ LDAP ในำฟล์ `/usr/lib/security/methods.cfg` จำเป็นต้องมีแอ็ตทริบิวต์อ็อพชันกับค่า netgroup แก้ไ ไฟล์ `/usr/lib/security/methods.cfg` และเพิ่ม บรรทัด `options = netgroup` ใน LDAP stanza คำสั่งนี้ ทำเครื่องหมายโหลดโมดูล LDAP เป็นโหลดโมดูลที่สามารถใช้ netgroup ตัวอย่าง:

   ```
   LDAP:
         program = /usr/lib/security/LDAP
         program_64 =/usr/lib/security/LDAP64
         options = netgroup
   ```

   ขณะนี้ คำสั่ง `lsuser -R LDAP nguser` หรือ `lsuser nguser` หรือ `lsuser -R LDAP -a ALL` จะไม่แสดง รายการผู้ใช้ใดๆ ขณะนี้ LDAP ถูกพิจารณาว่าเป็นฐานข้อมูล netgroup เท่านั้นจาก ไคลเอ็นต์นี้ แจะไม่มี netgroups ใดถูกเปิดใช้งานสำหรับการเข้าถึงไคลเอ็นต์นี้ เลย
3. แก้ไขไฟล์ `/etc/passwd` และผนวก บรรทัดสำหรับ netgroup ที่สนับสนุนการเข้าถึงระบบ ตัวอย่างถ้า *mygroup* เป็น netgroup บนเซิร์ฟเวอร์ LDAP ที่มี ผู้ใช้ที่ต้องการ ให้ผนวกบรรทัดต่อไปนี้:

   ```
   +@mygroup
   ```
4. แก้ไขไฟล์ `/etc/group` และผนวก บรรทัด `+:` เพื่อเปิดใช้งานการค้นหา NIS สำหรับกลุ่ม:

   ```
   +:
   ```

   การรันคำสั่ง `lsuser nguser` ขณะนี้ส่งกลับผู้ใช้เนื่องจาก *nguser* อยู่ใน netgroup *mygroup*

   คำสั่ง `lsuser -R LDAP nguser` ไม่ค้นหาผู้ใช้ แต่คำสั่ง `lsuser -R compat nguser` ค้นหา เนื่องจากผู้ใช้ถูกพิจารณาเป็นผู้ใช้ **compat** ในขณะนี้

5. หากผู้ใช้ netgroup เพื่อพิสูจน์ตัวตนกับระบบ, กลไกการพิสูจน์ตัวตน AIX ต้องทราบเมธอดที่ต้องการใช้ ถ้า stanza ดีฟอลต์ในไฟล์ /etc/security/user มี SYSTEM = compat ดังนั้นผู้ใช้ netgroup ทั้งหมด ใน netgroup ที่ถูกเพิ่มในไฟล์ /etc/passwd จะสามารถพิสูจน์ตัวตน อีกอ็อพชันหนึ่งคือการตั้งค่าผู้ใช้แต่ละคน โดยการเพิ่ม stanzas ด้วยตนเองในไฟล์ /etc/security/user สำหรับผู้ใช้ที่ต้องการ stanza ตัวอย่างสำหรับ *nguser* คือ:

   ```
   nguser:
           SYSTEM = compat
           registry = compat
   ```

   ผู้ใช้ Netgroup ใน netgroups ที่อนุญาตขณะนี้สามารถพิสูจน์ตัวตนกับระบบได้

   การเปิดใช้งาน คุณลักษณะ netgroup ยังเรียกทำงานเงื่อนไขต่อไปนี้:

   - ผู้ใช้ที่กำหนดในไฟล์ /etc/security/user เป็นสมาชิกของรีจิสทรี LDAP (ที่มี registry=LDAP และ SYSTEM="LDAP") ไม่สามารถพิสูจน์ตัวตนเป็นผู้ใช้ LDAP ผู้ใช้เหล่านี้ในขณะนี้เป็นผู้ใช้ **nis_ldap** และจำเป็นต้องมีความเป็นสมาชิก NIS netgroup โดยเริ่มต้น
   - ความหมายของ compat ในรีจิสทรีถูกขยายเพื่อรวมโมดูล ที่ใช้ netgroup ตัวอย่าง ถ้าโมดูล LDAP ถูกเปิดใช้งาน netgroup compat จะรวมรีจิสทรีไฟล์ NIS และ LDAP ผู้ใช้ถูกเรียกคืน จากโมดูลเหล่านั้นมีค่ารีจิสทรีที่ compat

ข้อมูลที่เกี่ยวข้อง

- เอกสาร exports File for NFS
- เอกสาร .rhosts File Format for TCP/IP
- เอกสาร hosts.equiv File Format for TCP/IP

เซิร์ฟเวอร์ LDAP ที่สนับสนุน:

การจัดการผู้ใช้และกลุ่มตาม AIX LDAP สนับสนุน IBM Tivoli Directory Servers เซิร์ฟเวอร์ที่ไม่ใช่ IBM ที่มี schema เป็นไปตาม RFC 2307 และ Microsoft active directory servers

IBM Tivoli Directory Server

ขอแนะนำว่า การจัดการผู้ใช้/กลุ่ม AIX ถูกกำหนดคอนฟิกโดยใช้ IBM Tivoli Directory Servers สำหรับ ข้อมูลเพิ่มเติมเกี่ยวกับการตั้งค่า IBM Tivoli Directory Server สำหรับการจัดการผู้ใช้และกลุ่ม, โปรดดู การตั้งค่าเซิร์ฟเวอร์ข้อมูลความปลอดภัย IBM Tivoli Directory Server

ไดเร็กทอรีเซิร์ฟเวอร์ที่มิใช่ IBM

AIX สนับสนุน ไดเร็กทอรีเซิร์ฟเวอร์หลากหลายที่ผู้ใช้หรือกลุ่มถูกกำหนด โดยใช้ RFC 2307 schema เมื่อกำหนดคอนฟิกเป็นไคลเอ็นต์ LDAP ให้กับเซิร์ฟเวอร์, AIX ใช้เซิร์ฟเวอร์ในวิธีเดียวกับ IBM Tivoli Directory Server ด้วยสกีมา RFC 2037 เซิร์ฟเวอร์ เหล่านี้ต้องสนับสนุนโปรโตคอล LDAP เวอร์ชัน 3

เนื่องจาก RFC 2307 schema กำหนดเซ็ตย่อยของแอ็ตทริบิวต์ผู้ใช้และกลุ่มที่ AIX สามารถใช้ได้เท่านั้น บางฟังก์ชันการทำงานการจัดการผู้ใช้และกลุ่ม AIX ไม่สามารถทำได้ถ้า AIX ถูกตั้งค่าเพื่อใช้ เซิร์ฟเวอร์ LDAP (ตัวอย่าง การบังคับใช้การตั้งค่ารหัสผ่านใหม่ ประวัติรหัสผ่าน ขีดจำกัดรีซอร์สต่อหนึ่งผู้ใช้ การควบคุมล็อกอินเข้าสู่ ระบบผ่านแอ็ตทริบิวต์ AIX hostsallowedlogin และ hostsdeniedlogin ความสามารถ และอื่นๆ)

AIX ไม่สนับสนุนไดเร็กทอรีเซิร์ฟเวอร์ที่ไม่เข้ากันกับ RFC 2307 อย่างไรก็ตาม AIX อาจถูกทำให้ทำงาน กับเซิร์ฟเวอร์ที่ไม่เข้ากันกับ RFC 2307 แต่ผู้ใช้และกลุ่ม ถูกกำหนดด้วยแอ็ตทริบิวต์ UNIX ที่จำเป็นทั้งหมด ชุดขั้นต่ำของแอ็ตทริบิวต์ผู้ใช้ และกลุ่มที่ต้องการใช้โดย AIX คือชุดที่กำหนดใน RFC 2307 การสนับสนุนไดเร็กทอรีเซิร์ฟเวอร์เหล่านั้นจำเป็นต้องมี การตั้งค่าด้วยตนเอง AIX จัดให้มี กลไกการแม็พ schema สำหรับวัตถุประสงค์นี้ สำหรับข้อมูลเพิ่มเติม เกี่ยวกับรูปแบบไฟล์ schema และการใช้งานไฟล์ schema ดูที่ รูปแบบไฟล์การแม็พแอ็ตทริบิวต์ LDAP

Microsoft Active Directory

AIX สนับสนุน Microsoft Active Directory (AD) เป็นเซิร์ฟเวอร์ LDAP สำหรับการจัดการผู้ใช้และกลุ่ม เซิร์ฟเวอร์ AD ต้องมี schema ที่สนับสนุน UNIX ถูกติดตั้ง schema การสนับสนุน UNIX ของ AD มาจากแพ็กเกจ Microsoft Service For UNIX (SFU) แต่ละ เวอร์ชัน SFU มีความแตกต่างของนิยาม schema ผู้ใช้และกลุ่มเล็กน้อย จากเวอร์ชันก่อนหน้า AIX สนับสนุน AD ที่ทำงานบน Windows 2000 และ 2003 ที่มี SFU schema เวอร์ชัน 3.0 และ 3.5 และ AD ที่ทำงานบน Windows 2003 R2 ที่มี UNIX schema ในตัว

เนื่องจาก ความแตกต่างในการจัดการผู้ใช้และกลุ่มระหว่างระบบ UNIX และระบบ Windows ทำให้ไม่ทุกคำสั่ง AIX ที่สามารถทำงานได้กับ ผู้ใช้ LDAP ถ้าเซิร์ฟเวอร์เป็น AD คำสั่งที่ไม่ทำงานรวม **mkuser** และ **mkgroup** คำสั่งการจัดการผู้ใช้และกลุ่มส่วนใหญ่ทำงานได้ ทั้งนี้ขึ้นอยู่กับ สิทธิการเข้าถึงที่กำหนดให้แก่ identity ซึ่งโยง AIX กับ AD คำสั่ง เหล่านี้มี **lsuser, chuser, rmuser, lsgroup, chgroup, rmgroup, id, groups, passwd** และ **chpasswd**

AIX สนับสนุนสองกลไกการพิสูจน์ตัวตน ผู้ใช้กับเซิร์ฟเวอร์ Windows: การพิสูจน์ตัวตน LDAP และการพิสูจน์ตัวตน Kerberos ด้วยการใช้กลไกอย่างใดอย่างหนึ่ง AIX สนับสนุน identification ผู้ใช้ ผ่านโปรโตคอล LDAP บน AD ที่ไม่มีข้อกำหนดสำหรับบัญชีผู้ใช้ ที่สัมพันธ์กันบน AIX

การกำหนดคอนฟิกระบบปฏิบัติการ AIX เพื่อทำงานกับ Active Directory ผ่าน LDAP:

AIX สนับสนุน Microsoft Active Directory (AD) เป็นเซิร์ฟเวอร์ LDAP สำหรับการจัดการผู้ใช้และกลุ่ม เป็นสิ่งจำเป็น ที่เซิร์ฟเวอร์ AD ต้องมี schema การสนับสนุน UNIX ติดตั้งอยู่

ผู้ดูแลระบบสามารถใช้คำสั่ง **mksecldap** เพื่อกำหนดคอนฟิก AIX บนเซิร์ฟเวอร์ AD ใน ลักษณะเดียวกับ IBM Tivoli Directory Server คำสั่ง **mksecldap** ซ่อนรายละเอียดการตั้งค่าทั้งหมดเพื่อให้ง่ายต่อดำเนินการ ก่อน รันคำสั่ง **mksecldap** เพื่อตั้งค่า AIX บนเซิร์ฟเวอร์ AD:

1. เซิร์ฟเวอร์ AD ต้องมี schema การสนับสนุน UNIX ติดตั้งอยู่
2. เซิร์ฟเวอร์ AD ต้องมีผู้ใช้ซึ่งเปิดใช้งาน UNIX

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการติดตั้ง UNIX schema ให้แก่ AD และการเปิดใช้งานผู้ใช้ AD ที่มีการสนับสนุน UNIX ดูที่เอกสารคู่มือ Microsoft ที่เกี่ยวข้อง

AD schema โดยส่วนใหญ่มีนิยามแอ็ตทริบิวต์หลายนิยามสำหรับแอ็ตทริบิวต์ UNIX เดียวกัน (ตัวอย่างเช่น มีนิยาม รหัสผ่านผู้ใช้ และสมาชิกกลุ่มหลายค่า) แม้ว่า AIX จะสนับสนุนเป็นส่วนใหญ่ แต่ควรคำนึงถึงข้อควรพิจารณาและการวางแผนอย่างระมัดระวังเมื่อเลือก นิยามที่จะใช้ ขอแนะนำให้ระบบ AIX และระบบอื่นที่มิใช่ AIX แบ่งใช้ AD เดียวกัน ให้ใช้นิยามเดียวกันเพื่อหลีกเลี่ยงความขัดแย้ง

การเลือกแอ็ตทริบิวต์รหัสผ่าน Active Directory:

AIX สนับสนุน กลไกการพิสูจน์ตัวตนสองวิธี **unix_auth** และ l**dap_auth**

ด้วย **unix_auth** รหัสผ่านใน Microsoft Active Directory (AD) จำเป็นต้อง อยู่ในรูปแบบเข้ารหัส ระหว่างการพิสูจน์ตัวตน รหัสผ่านที่เข้ารหัส จะถูกเรียกออกมาจาก AD และเปรียบเทียบกับรูปแบบที่เข้ารหัสของรหัสผ่าน ที่ผู้ใช้ป้อน การพิสูจน์ตัวตนสำเร็จถ้าทั้งสองมีค่าตรงกัน ในโหมด **ldap_auth** AIX จะพิสูจน์ตัวตนผู้ใช้โดยการดำเนินการเชื่อม LDAP กับเซิร์ฟเวอร์ด้วย identity ของผู้ใช้และ รหัสผ่านที่ให้ ผู้ใช้ได้รับการพิสูจน์ตัวตนถ้าการดำเนินการเชื่อม สำเร็จ AD สนับสนุนแอ็ตทริบิวต์รหัสผ่านผู้ใช้แบบหลายค่า โหมดการพิสูจน์ตัวตน AIX ที่แตกต่าง จำเป็นต้องใช้แอ็ตทริบิวต์รหัสผ่านผู้ใช้ AD ที่ต่างกัน

โหมด unix_auth

แอ็ตทริบิวต์รหัสผ่าน AD ต่อไปนี้สามารถใช้สำหรับโหมด **unix_auth**:

- **userPassword**
- **unixUserPassword**
- **msSFU30Password**

การจัดการรหัสผ่านบน AIX อาจทำได้ยากเนื่องจาก แอ็ตทริบิวต์รหัสผ่านแบบหลายค่าของ AD การทราบว่าแอ็ตทริบิวต์การจัดการ รหัสผ่านใดควรใช้โดยไคลเอ็นต์ UNIX อาจเกิดความสับสน ความสามารถในการแม็พแอ็ตทริบิวต์ AIX LDAP ช่วยให้คุณสามารถกำหนดการจัดการ รหัสผ่านได้เองตามความต้องการของคุณ

โดยค่าดีฟอลต์ AIX ใช้แอ็ตทริบิวต์ **msSFU30Password** สำหรับ AD ที่กำลังทำงานบน Windows 2000 และ 2003 และแอ็ตทริบิวต์ **userPassword** บน Windows 2003 R2 ถ้าใช้รหัสผ่านอื่นที่ต่างออกไป คุณจำเป็นต้องแก้ไขไฟล์ `/etc/security/ldap/sfu30user.map` (หรือไฟล์ `/etc/security/ldap/sfur2user.map` ถ้า AD กำลังทำงานบน Windows 2003 R2) ค้นหาบรรทัดที่ขึ้นต้นด้วยคำว่า **spassword** และ เปลี่ยนฟิลด์ที่สามของบรรทัดเป็นชื่อแอ็ตทริบิวต์รหัสผ่าน AD ที่ต้องการ สำหรับข้อมูลเพิ่มเติม ดูที่ LDAP Attribute Mapping File Format รันคำสั่ง **mksecldap** เพื่อตั้งค่า ไคลเอ็นต์ AIX LDAP หลังการเปลี่ยนแปลง ถ้าไคลเอ็นต์ AIX LDAP ถูกตั้งค่าอยู่แล้ว ให้รันคำสั่ง **restart-secldapclntd** เพื่อทำงาน **secldapclntd** daemon ต่อเพื่อรวม การเปลี่ยนแปลง

ในโหมด **unix_auth** รหัสผ่านอาจไม่ซิงคี กันระหว่าง Windows และ UNIX เป็นผลให้มีรหัสผ่าน ต่างกันสำหรับแต่ละระบบ นี้เกิดขึ้นเมื่อคุณเปลี่ยนรหัสผ่านจาก AIX เป็น Windows เนื่องจาก Windows ใช้แอ็ตทริบิวต์รหัสผ่าน **uncodepwd** คำสั่ง AIX **passwd** สามารถรีเซ็ตรหัสผ่าน UNIX ให้เหมือนกับ รหัสผ่าน Windows ได้ แต่ AIX ไม่สนับสนุนการเปลี่ยนรหัสผ่านของ Window โดยอัตโนมัติเมื่อคุรเปลี่ยนรหัสผ่าน UNIX ของคุณจาก AIX

โหมด ldap_auth

Active Directory ยังมี แอ็ตทริบิวต์รหัสผ่าน **unicodepwd** แอ็ตทริบิวต์รหัสผ่านนี้ใช้โดยระบบ Windows เพื่อพิสูจน์ตัวตนผู้ใช้ Windows ในการดำเนินการเชื่อมกับ AD รหัสผ่าน **unicodePwd** ต้อง ถูกใช้ ไม่มีรหัสผ่านใดที่กล่าวขึ้นภายใต้โหมด **unix_auth** ที่ใช้ได้สำหรับการดำเนินการเชื่อม ถ้าอ็อพชัน **ldap_auth** ถูกระบุ จากบรรทัดคำสั่ง คำสั่ง **mksecldap** แม็พ แอ็ตทริบิวต์รหัสผ่านกับแอ็ตทริบิวต์ **unicodePwd** ของ AD ตอนทำการตั้งค่า ไคลเอ็นต์ที่ไม่จำเป็นต้องมีขั้นตอนการทำด้วยตนเอง

โดยการแม็พรหัสผ่าน AIX กับแอ็ตทริบิวต์ **unicodePwd** ผู้ใช้ที่กำหนดใน AD สามารถล็อกอินเข้าสู่ระบบ Windows และ AIX โดยใช้รหัสผ่าน เดียวกัน รหัสผ่านที่รีเซ็ตจากระบบ AIX หรือ Windows จะมีผลสำหรับทั้งสองระบบคือ AIX และ Windows

การเลือกแอ็ตทริบิวต์สมาชิกกลุ่ม Active Directory:

Microsoft's Service for UNIX กำหนดกลุ่มสมาชิก **memberUid, msSFU30MemberUid** และ **msSFU30PosixMember**

แอ็ตทริบิวต์ **memberUid** และ **msSFU30MemeberUid** ยอมรับ ชื่อบัญชีผู้ใช้ ขณะที่ **msSFU30PosixMember** ยอมรับ DN แบบเต็มเท่านั้น ตัวอย่าง สำหรับบัญชีผู้ใช้ *foo* (ที่มีนามสกุล *bar*) ที่กำหนดใน AD:

- **memberUid: foo**
- **msSFU30MemberUid: foo**
- **msSFU30PosixMember: CN=foo bar,CN=Users,DC=austin,DC=ibm,DC=com**

ระบบปฏิบัติการ AIX สนับสนุนแอ็ตทริบิวต์เหล่านี้ทั้งหมด ปรึกษากับผู้ดูแลระบบ AD ของคุณเพื่อพิจารณาว่า ควรใช้แอ็ตทริบิวต์ใด โดยดีฟอลต์ คำสั่ง **mksecldap** กำหนดคอนฟิกระบบปฏิบัติการ AIX เพื่อ ใช้แอ็ตทริบิวต์ **msSFU30PosixMember** กับ AD ที่รันบน Windows 2000 และ 2003 และแอ็ตทริบิวต์ **uidMember** กับ AD ที่รันบน Windows 2003 R2 การเลือกเช่นนั้นเนื่องจากลักษณะการทำงาน AD เนื่องจาก AD เลือกแอ็ตทริบิวต์เมื่อเพิ่ม ผู้ใช้ในกลุ่มจาก Windows กลยุทธ์ธุรกิจของคุณอาจจำเป็นต้องใช้แอ็ตทริบิวต์ที่เป็นสมาชิกกลุ่ม ที่ไม่ใช่ค่าดีฟอลต์เพื่อการสนับสนุนแบบหลายแพล็ตฟอร์ม

ถ้าจำเป็นต้องใช้แอ็ตทริบิวต์ของสมาชิกกลุ่มอื่น คุณสามารถเปลี่ยน การแม็พได้โดยการแก้ไขไฟล์การแม็พกลุ่ม ไฟล์การแม็พกลุ่ม สำหรับ AD คือ `/etc/security/ldap/sfu30group.map` ที่กำลังทำงาน บน Windows 2000 และ 2003 และ `/etc/security/ldap/sfur2group.map` สำหรับ Windows 2003 R2 ค้นหาบรรทัด ที่ขึ้นต้นด้วยคำว่า **users** และแทนที่ฟิลด์ที่สามด้วยชื่อแอ็ตทริบิวต์ที่ต้องการสำหรับสมาชิกกลุ่ม สำหรับข้อมูลเพิ่มเติม ดูที่ LDAP Attribute Mapping File Format รันคำสั่ง **mksecldap** เพื่อกำหนดคอนฟิกไคลเอ็นต์ AIX LDAP หลังจากเปลี่ยน, หรือถ้าไคลเอ็นต์ AIX ได้ถูกกำหนดคอนฟิกไว้, ให้รันคำสั่ง **restart-secldapclntd** เพื่อรีสตาร์ท **secldapclntd** daemon เพื่อรับรู้การเปลี่ยนแปลง

หน่วยระดับองค์กรหลายหน่วย:

เซิร์ฟเวอร์ AD ของคุณอาจมีหน่วยระดับองค์กรหลายหน่วย ถูกกำหนด แต่ละหน่วยมีชุดของผู้ใช้

ผู้ใช้ Windows AD ส่วนใหญ่ ถูกกำหนดในแผนผังย่อย **cn=users,...** แต่บางส่วนอาจถูกกำหนด ที่อื่นได้ คุณลักษณะ AIX DN หลักจำนวนมากสามารถใช้ได้สำหรับเซิร์ฟเวอร์ AD สำหรับข้อมูลเพิ่มเติม ดูที่ การสนับสนุน หลาย base DN

การพิสูจน์ตัวตน Kerberos สำหรับเซิร์ฟเวอร์ Windows:

นอกจากกลไกการพิสูจน์ตัวตน LDAP แล้ว, ระบบปฏิบัติการ AIX ยังสนับสนุนการพิสูจน์ตัวตนผู้ใช้ผ่านโปรโตคอล Kerberos สำหรับเซิร์ฟเวอร์ Windows

ระบบปฏิบัติการ AIX สนับสนุนการพิสูจน์ตัวตน Kerberos สำหรับ Windows KDC และ LDAP identification สำหรับ Windows Active Directory โดยสร้างโหลดโมดูลผสม KRB5ALDAP เนื่องจากข้อมูล identification ของผู้ใช้ถูกดึงมาจาก Microsoft Active Directory, คุณไม่จำเป็นต้องสร้างแอคเคาต์ผู้ใช้ ที่สอดคล้องกันบนระบบปฏิบัติการ AIX

การจัดการผู้ใช้ LDAP:

คุณสามารถจัดการผู้ใช้และกลุ่มบน LDAP security information server ได้จากไคลเอ็นต์ LDAP ใดๆ โดยใช้คำสั่งระดับสูง

แฟล็ก **-R** ที่เพิ่มในคำสั่งระดับสูงส่วนใหญ่สามารถจัดการผู้ใช้ และกลุ่มโดยใช้ LDAP รวมถึงโหลดโมดูลการพิสูจน์ตัวตนอื่นๆ เช่น DCE, NIS และ KRB5 สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการใช้แฟล็ก **-R** ให้อ้างอิงในแต่ละคำสั่งของคำสั่งการจัดการผู้ใช้หรือกลุ่ม

ในการเปิดใช้การพิสูจน์ตัวตนผู้ใช้ผ่าน LDAP ให้รันคำสั่ง **chuser** เพื่อเปลี่ยนค่าแอ็ตทริบิวต์ **SYSTEM** ของผู้ใช้เป็น LDAP โดยการตั้งค่าแอ็ตทริบิวต์ **SYSTEM** ตามหลักไวยากรณ์ที่กำหนด ผู้ใช้สามารถถูกพิสูจน์ตัวตนผ่าน โหลดโมดูลมากกว่าหนึ่ง

โมดูล (ตัวอย่างเช่น compat และ LDAP) สำหรับข้อมูลเพิ่มเติม เกี่ยวกับการตั้งค่าวิธีการพิสูจน์ตัวตนของผู้ใช้ ดูที่ ไวยากรณ์แอ็ตทริบิวต์ “การพิสูจน์ตัวตนผู้ใช้” ในหน้า 80 **SYSTEM** ที่กำหนดในไฟล์ /etc/security/user

ผู้ใช้สามารถเป็นผู้ใช้ LDAP ได้ในตอนเซ็ตอัพไคลเอ็นต์โดยการรันคำสั่ง **mksecldap** ด้วยแฟล็ก **–u** ในรูปแบบใดรูปแบบหนึ่งต่อไปนี้:

1. รันคำสั่ง:

   ```
   mksecldap -c -u user1,user2,...
   ```

   โดย *user1*,*user2*,... คือ รายการผู้ใช้ ผู้ใช้ในรายชื่อนี้สามารถผู้ใช้ที่กำหนดแบบโลคัล หรือที่กำหนดแบบ LDAP รีโมต แอ็ตทริบิวต์ **SYSTEM** ถูกตั้งค่าเป็น LDAP ในแต่ละ stanzas ของผู้ใช้ด้านบนในไฟล์ /etc/security/user ผู้ใช้เหล่านั้นจะถูกพิสูจน์ตัวตนผ่าน LDAP เท่านั้น ผู้ใช้ในรายชื่อนี้ต้อง มีอยู่บน LDAP security information server มิฉะนั้นจะไม่สามารถล็อกอิน จากโฮสต์นี้ รันคำสั่ง **chuser** เพื่อแก้ไขแอ็ตทริบิวต์ **SYSTEM** และอนุญาตให้ทำการพิสูจน์ตัวตนโดยใช้หลายๆ วิธี (ตัวอย่าง ทั้งแบบโลคัล และ LDAP)
2. รัน

   ```
   mksecldap -c -u ALL
   ```

   คำสั่งนี้ตั้งค่าแอ็ตทริบิวต์ **SYSTEM** เป็น LDAP ในแต่ละ stanza ของผู้ใช้ในไฟล์ /etc/security/user สำหรับผู้ใช้ที่กำหนดแบบโลคัลทั้งหมด ผู้ใช้ทั้งหมดพิสูจน์ตัวตนผ่าน LDAP เท่านั้น ผู้ใช้ที่กำหนดแบบโลคัลต้องมีอยู่บน LDAP security information server มิฉะนั้นจะไม่สามารถล็อกอินจากโฮสต์นี้ ผู้ใช้ที่ถูกกำหนดบน เซิร์ฟเวอร์ LDAP แต่ไม่ได้ถูกกำหนดแบบโลคัลจะไม่สามารถล็อกอินจากโฮสต์นี้ ในการอนุญาต ให้ผู้ใช้ที่กำหนดแบบ LDAP รีโมตเพื่อล็อกอินจากโฮสต์นี้ ให้รันคำสั่ง **chuser** เพื่อตั้งค่าแอ็ตทริบิวต์ **SYSTEM** เป็น LDAP สำหรับผู้ใช้นั้น

อีกทางหนึ่ง คุณสามารถเปิดใช้งานผู้ใช้ LDAP ทั้งหมด ไม่ว่าจะถูกกำหนดแบบ โลคัลหรือไม่ เพื่อพิสูจน์ตัวตนผ่าน LDAP บนโฮสต์โลคัลโดยการแก้ไข stanza "ดีฟอลต์" ของไฟล์ /etc/security/user เป็น ใช้ "LDAP" เป็นค่า ผู้ใช้ทั้งหมดที่ไม่มีค่าถูกกำหนดสำหรับแอ็ตทริบิวต์ **SYSTEM** ของตน ต้องทำตามคำสั่งที่กำหนดใน stanza ดีฟอลต์ ตัวอย่าง ถ้า stanza ดีฟอลต์มี `"SYSTEM = "compat""` การเปลี่ยนเป็น `"SYSTEM = "compat OR LDAP""` จะอนุญาตให้ทำการพิสูจน์ตัวตนผู้ใช้เหล่านี้โดยผ่าน AIX หรือ LDAP การเปลี่ยน stanza ดีฟอลต์เป็น `"SYSTEM = "LDAP""` จะเปิดให้ผู้ใช้เหล่านี้ พิสูจน์ตัวตนผ่าน LDAP โดยเฉพาะ เหล่าผู้ใช้ที่มีค่าแอ็ตทริบิวต์ **SYSTEM** ถูกกำหนดจะไม่ได้รับผลโดย stanza ดีฟอลต์

การสนับสนุนหลาย base DN:

AIX สนับสนุน DNs หลัก โดยมีได้สูงสุด 10 base DNs สำหรับแต่ละ entity ที่ สามารถระบุได้ในไฟล์ /etc/security/ldap/ldap.cfg

Base DNs เรียงลำดับความสำคัญตามลำดับที่ปรากฏในไฟล์ /etc/security/ldap/ldap.cfg การดำเนินการโดยคำสั่ง AIX ในกรณีของหลาย base DNs ถูกดำเนินการตามลำดับความสำคัญ base DN ที่มีลักษณะการทำงานต่อไปนี้:

- การดำเนินการเคียวรี (ตัวอย่าง โดยคำสั่ง **lsuser**) ถูกดำเนินการกับ base DNs ตามระดับความสำคัญจนกระทั่งพบ บัญชีผู้ใช้ที่ตรง หรือส่งกลับค่าความล้มเหลวถ้า base DNs ทั้งหมดที่ถูกค้นหา ไม่พบรายการที่ตรง การเคียวรีสำหรับ ALL ส่งผลให้บัญชีผู้ใช้ ทั้งหมดจากทุก base DN จะถูกส่งกลับ
- การดำเนินการแก้ไข (ตัวอย่าง โดยคำสั่ง **chuser**) ถูกดำเนินการกับบัญชีผู้ใช้แรกที่ตรง
- การดำเนินการลบ (ตัวอย่าง โดยคำสั่ง **rmuser**) ถูกดำเนินการกับบัญชีผู้ใช้แรกที่ตรง
- การดำเนินการสร้าง (ตัวอย่าง โดยคำสั่ง **mkuser**) ถูกดำเนินการกับ base DN แรกเท่านั้น AIX ไม่ สนับสนุนการสร้างบัญชีผู้ใช้ให้แก่ base DNs อื่น

ถ้าความรับผิดชอบของผู้ดูแลระบบไดเร็กทอรีเซิร์ฟเวอร์คือต้องดูแลจัดการ ฐานข้อมูลบัญชีผู้ใช้ไม่ให้มีการชนกัน ถ้ามีหลายนิยาม สำหรับบัญชีผู้ใช้เดียวกัน แต่ละนิยามจะอยู่ภายใต้แผนผังย่อยต่างกัน บัญชีผู้ใช้แรกเท่านั้น ที่เห็นได้ใน AIX การดำเนินการ ค้นหาส่งกลับบัญชีผู้ใช้แรกที่ตรงเท่านั้น เช่นเดียวกับ การดำเนินการแก้ไขหรือการลบที่ถูกดำเนินการกับบัญชีผู้ใช้แรกที่ตรงเท่านั้น

คำสั่ง **mksecldap** เมื่อใช้เพื่อตั้งค่าไคลเอ็นต์ LDAP จะค้นหา base DN สำหรับแต่ละ entity และบันทึกค่าไว้ในไฟล์ `/etc/security/ldap/ldap.cfg` เมื่อมีหลาย base DNs อยู่บนเซิร์ฟเวอร์ LDAP สำหรับหนึ่ง entity คำสั่ง **mksecldap** จะสุ่มในค่าใดค่าหนึ่งจากค่า เหล่านี้ ในการให้ AIX ทำงานกับ หลาย base DNs คุณจำเป็นต้องแก้ไขไฟล์ `/etc/security/ldap/ldap.cfg` หลังจากคำสั่ง **mksecldap** ดำเนินการเสร็จเรียบร้อย ค้นหานิยาม base DN ที่เหมาะสมและเพิ่ม base DNs เพิ่มเติม ที่จำเป็นต้องใช้ AIX สนับสนุนสูงสุด 10 base DNs สำหรับแต่ละ entity โดย base DNs ที่เกินมาจะถูกข้าม

AIX ยังสนับสนุนตัวกรองที่ผู้ใช้ กำหนดและขอบเขตการค้นสำหรับแต่ละ base DN base DN สามารถมีตัวกรองและ ขอบเขตของตนเองที่อาจแตกต่างจาก base DNs เพียร์ของตน ตัวกรอง สามารถใช้เพื่อกำหนดชุดของบัญชีผู้ใช้ที่เห็นได้ใน AIX

บัญชีผู้ใช้เหล่านั้นเท่านั้นเท่านั้นที่ตรงตามเงื่อนไขตัวกรองจึงจะเห็นได้ใน AIX

การตั้งค่า SSL บนเซิร์ฟเวอร์ LDAP:

เมื่อต้องการตั้งค่า Secure Sockets Layer (SSL) บนเซิร์ฟเวอร์ LDAP, ให้ติดตั้งชุดไฟล์ LDAP crypto และชุดไฟล์ **GSKit** เพื่อเปิดใช้งานส่วนขยายการเข้ารหัสเซิร์ฟเวอร์ ชุดไฟล์เหล่านี้สามารถพบได้ใน AIX expansion pack

ทำตามขั้นตอนเหล่านี้เพื่อเปิดใช้งานการสนับสนุน SSL สำหรับการพิสูจน์ตัวตนไดเร็กทอรีเซิร์ฟเวอร์ IBM

1. ติดตั้ง IBM Tivoli Directory Server **GSKit** สำหรับ IBM Tivoli Directory Server เวอร์ชัน 6.2, หรือ **GSKitv8** สำหรับ IBM Tivoli Directory Server เวอร์ชัน 6.3, หากยังไม่ได้ติดตั้งไว้
2. สร้างคีย์ไพรเวตของเซิร์ฟเวอร์ IBM Directory และใบรับรองเซิร์ฟเวอร์โดยใช้ยูทิลิตีคีย์ GSKit ที่ถูกต้อง ใช้ยูทิลิตี **gsk7ikm** ด้วย IBM Tivoli Directory Server เวอร์ชัน 6.2, และใช้เครื่องมือ ikeyman สำหรับ IBM Tivoli Directory Server เวอร์ชัน 6.3, หรือเวอร์ชันถัดมา ใบรับรอง ของเซิร์ฟเวอร์อาจถูกลงนามโดย Certification Authority (CA) เชิงพาณิชย์, เช่น VeriSign, หรืออาจลงนามด้วยตนเองด้วยเครื่องมือการจัดการคีย์ GSKit ใบรับรองพับลิก (หรือใบรับรองการลงนามด้วยตนเอง) ของ CA ต้องถูกกระจายไปยังไฟล์ฐานข้อมูลหลัก ของไคลเอ็นต์แอ็พพลิเคชัน
3. ที่เก็บไฟล์ฐานข้อมูลหลักของเซิร์ฟเวอร์ และเชื่อมโยงรหัสผ่านไฟล์บนเซิร์ฟเวอร์ พาธดีฟอลต์สำหรับฐานข้อมูลคีย์ `/usr/ldap/etc` directory คือตำแหน่งที่ตั้งทั่วไป
4. รันคำสั่งต่อไปนี้ให้ตั้งค่าเซิร์ฟเวอร์, โดยที่ **mykey.kdb** คือฐานข้อมูลหลักและ *keypwd* เป็นรหัสผ่านไปยังฐานข้อมูลหลัก:

   ```
   # mksecldap -s -a cn=admin -p pwd -S rfc2307aix -k /usr/ldap/etc/mykey.kdb -w keypwd
   ```

การตั้งค่า SSL บนไคลเอ็นต์ LDAP:

เมื่อต้องการใช้ SSL บนไคลเอ็นต์ LDAP, ให้ติดตั้ง `ldap.max_crypto_client` และติดตั้งชุดไฟล์ `GSKit` ของแพ็กเสริม AIX

ทำตามขั้นตอนเหล่านี้เพื่อเปิดใช้งานการสนับสนุน SSL สำหรับ LDAP หลังจาก เซิร์ฟเวอร์ถูกเปิดใช้งานสำหรับ SSL

1. รัน `gsk7ikm` เพื่อสร้างฐานข้อมูลคีย์ บนแต่ละไคลเอ็นต์
2. คัดลอกใบรับรองเซิร์ฟเวอร์ไปยังแต่ละไคลเอ็นต์ ถ้า SSL เซิร์ฟเวอร์ใช้ใบรับรองที่ลงนามด้วยตนเอง ใบรับรองต้อง ถูกเอ็กซ์พอร์ตเป็นอันดับแรก
3. บนแต่ละระบบไคลเอ็นต์ รัน `gsk7ikm` เพื่อ อิมพอร์ตใบรับรองเซิร์ฟเวอร์ไปยังฐานข้อมูลคีย์

4. เปิดใช้งาน SSL สำหรับแต่ละไคลเอ็นต์:

   ```
   # mksecldap -c -h servername -a adminDN -p pwd -k /usr/ldap/etc/mykey.kdb -p keypwd
   ```

   โดยที่ `/usr/ldap/etc/mykey.kdb` คือพาธ แบบเต็มไปยังฐานข้อมูลคีย์และ *keypwd* คือรหัสผ่านสำหรับ คีย์ ถ้ารหัสผ่านคีย์ไม่ถูกป้อนจากบรรทัดคำสั่ง ไฟล์รหัสผ่านที่จัดเก็บ จากไดเร็กทอรีเดียวกันจะถูกนำมาใช้ไฟล์ที่จัดเก็บจำเป็น ต้องมีชื่อเดียวกับฐานข้อมูลคีย์ด้วยมีส่วนขยาย **.sth** (ตัวอย่าง `mykey.sth`)

ค่าควบคุมการเข้าใช้โฮสต์ LDAP:

AIX จัดให้มี การควบคุมการเข้าถึงโฮสต์ระดับผู้ใช้ (ล็อกอิน) สำหรับระบบ ผู้ดูแลระบบ สามารถตั้งค่าผู้ใช้ LDAP เพื่อล็อกอินเข้าสู่ระบบ AIX โดยการตั้งค่าแอ็ตทริบิวต์ **SYSTEM** ของตน เป็น LDAP

แอ็ตทริบิวต์ **SYSTEM** อยู่ในไฟล์ `/etc/security/user` คำสั่ง **chuser** สามารถใช้เพื่อตั้งค่า คล้ายกับตัวอย่างต่อไปนี้:

```
# chuser -R LDAP SYSTEM=LDAP registry=LDAP foo
```

หมายเหตุ: ด้วยการใช้การควบคุมประเภทนี้ อย่าตั้งค่าแอ็ตทริบิวต์ **SYSTEM** ดีฟอลต์ เป็น LDAP ซึ่งจะอนุญาตให้ผู้ใช้ LDAP ทั้งหมดล็อกอินเข้าสู่ระบบได้

ตัวอย่างนี้ตั้งค่าแอ็ตทริบิวต์ LDAP ให้อนุญาตให้ผู้ใช้ *foo* ล็อกอิน เข้าสู่ระบบนี้ รวมทั้งต้องค่ารีจิสทรีเป็น LDAP ซึ่งอนุญาตให้กระบวนการล็อกอินบันทึกการพยายามล็อกอินของ *foo* เข้าสู่ LDAP และยัง อนุญาตสามารถดำเนินงานการจัดการผู้ใช้ได้บน LDAP

ผู้ดูแลระบบจำเป็นต้องรันการเซ็ตอัพบนแต่ละระบบไคลเอ็นต์ เพื่อเปิดใช้การล็อกอินโดยผู้ใช้นั้นๆ

AIX มีคุณลักษณะ เพื่อจำกัดผู้ใช้ LDAP เพื่อล็อกอินเข้าสู่ระบบไคลเอ็นต์ LDAP เท่านั้น คุณลักษณะนี้ อนุญาตให้มีการจัดการค่าควบคุมการเข้าใช้โฮสต์จากศูนย์กลาง ผู้ดูแลระบบ สามารถระบุรายการค่าควบคุมการเข้าใช้โฮสต์สองรายการสำหรับหนึ่งบัญชีผู้ใช้: รายการ ที่อนุญาต และรายการที่ปฏิเสธ แอ็ตทริบิวต์ผู้ใช้สองค่านี้ถูกเก็บใน เซิร์ฟเวอร์ LDAP กับบัญชีผู้ใช้ ผู้ใช้ได้รับอนุญาตให้เข้าถึงระบบ หรือเน็ตเวิร์กที่ถูกระบุในรายการที่อนุญาต ขณะเดียวกันก็ถูกปฏิเสธ มิให้เข้าถึงระบบหรือเน็ตเวิร์กในรายการที่ปฏิเสธ ถ้าระบบถูกระบุ ในทั้งรายการที่อนุญาตและรายการที่ปฏิเสธ ผู้ใช้จะไม่ได้รับอนุญาตให้เข้าถึง ระบบ มีสองวิธีที่จะระบุรายการการเข้าถึงสำหรับ ผู้ใช้: ด้วยคำสั่ง **mkuser** เมื่อผู้ใช้ถูก สร้างขึ้น หรือด้วยคำสั่ง **chuser** สำหรับผู้ใช้ ที่มีอยู่แล้ว สำหรับความเข้ากันได้ย้อนหลัง ถ้าทั้งรายการที่อนุญาตและรายการ ที่ปฏิเสธไม่มีสำหรับผู้ใช้ ผู้ใช้จะได้รับอนุญาตให้ล็อกอินระบบไคลเอ็นต์ LDAP ใดๆ เป็นค่าดีฟอลต์

ตัวอย่างของการตั้งค่ารายการสิทธิการอนุญาตและปฏิเสธสำหรับผู้ใช้ มีดังนี้:

```
# mkuser -R LDAP hostsallowedlogin=host1,host2 foo
```

คำสั่งนี้สร้างผู้ใช้ *foo* และผู้ใช้ *foo* ได้รับอนุญาต ให้ล็อกอินเข้าสู่ `host1` และ `host2` เท่านั้น

```
# mkuser -R LDAP hostsdeniedlogin=host2 foo
```

คำสั่งนี้สร้างผู้ใช้ *foo* และผู้ใช้ *foo* สามารถล็อกอิน เข้าสู่ระบบไคลเอ็นต์ LDAP ใดๆ ยกเว้น `host2`

```
# chuser -R LDAP hostsallowedlogin=192.9.200.1 foo
```

คำสั่งนี้ตั้งค่าผู้ใช้ *foo* ให้มีสิทธิล็อกอินเข้าสู่ระบบไคลเอ็นต์ ที่แอดเดรส `192.9.200.1`

```
# chuser -R LDAP hostsallowedlogin=192.9.200/24 hostsdeniedlogin=192.9.200.1 foo
```

คำสั่งนี้ตั้งค่าผู้ใช้ *foo* ให้มีสิทธิล็อกอินเข้าสู่ระบบไคลเอ็นต์ใดๆ ภายในซับเน็ต 192.9.200/24 ยกเว้น ระบบไคลเอ็นต์ที่แอดเดรส 192.9.200.1

สำหรับข้อมูลเพิ่มเติม ดูที่คำสั่ง **chuser**

การสื่อสารอย่างปลอดภัยด้วย SSL:

โดยขึ้นอยู่กับประเภทการพิสูจน์ตัวตนที่ใช้ระหว่างไคลเอ็นต์ LDAP และเซิร์ฟเวอร์ รหัสผ่านถูกส่งในรูปแบบที่เข้ารหัส (unix_auth) หรือแบบข้อมูลธรรมดา (ldap_auth) อย่างใดอย่างหนึ่ง ใช้ Secure Socket Layer (SSL) เพื่อป้องกันจาก การเปิดเผยความปลอดภัยแม้เมื่อคุณส่งรหัสผ่านที่เข้ารหัสบนเน็ตเวิร์ก หรือในบางกรณีบนอินเตอร์เน็ต AIX มี แพ็กเกจสำหรับ SSL ที่สามารถให้มีการสื่อสารอย่างปลอดภัยระหว่างไดเร็กทอรีเซิร์ฟเวอร์ และไคลเอ็นต์

สำหรับข้อมูลเพิ่มเติม ดูที่:

- “การตั้งค่า SSL บนเซิร์ฟเวอร์ LDAP” ในหน้า 175
- “การตั้งค่า SSL บนไคลเอ็นต์ LDAP” ในหน้า 175

การใช้โหมดการพิสูจน์ตัวตน LDAPA เท่านั้นauthentication-only mode:

โมดูล LDAP คือโมดูลที่มีฟังก์ชันสมบูรณ์ที่สนับสนุนทั้ง การพิสูจน์ตัวตนผู้ใช้และ identification ผู้ใช้ โมดูล LDAPA มีโหมดการพิสูจน์ตัวตนเท่านั้น โมดูล LDAPA เหมือนกับโมดูล LDAP แต่คุณสามารถระบุเพื่อให้โหมดการพิสูจน์ตัวตนเท่านั้น

ในโหมดการพิสูจน์ตัวตนเท่านั้น โมดูล LDAPA ต้องรวมเข้ากับ อีกโมดูลฐานข้อมูลเพื่อสร้างโมดูลผสมแทน โมดูลสแตนด์อะโลน โมดูล LDAPA ดำเนินการพิสูจน์ตัวตนผู้ใช้ ขณะที่โมดูลที่สองดำเนินการ identification โมดูลที่รวมเข้าด้วยกันนี้ เรียกว่า โมดูลผสม คุณต้องกำหนดผู้ใช้ทั้งในเซิร์ฟเวอร์ LDAP และเซิร์ฟเวอร์ฐานข้อมูลสำหรับใช้โมดูลผสมนี้

ด้วยโมดูล LDAPA ข้อมูลกลุ่มจะได้มาจาก เซิร์ฟเวอร์ฐานข้อมูล ตัวอย่าง ในกรณีของไฟล์ LDAPA ข้อมูล กลุ่มได้มาจากไฟล์ /etc/group โลคัล ถ้าผู้ใช้ LDAP ของคุณบางคนอยู่ในกลุ่ม LDAP เท่านั้น คุณต้องสร้าง กลุ่ม LDAP ที่สอดคล้องกันบนเซิร์ฟเวอร์ฐานข้อมูลก่อนที่คุณตั้งค่า โมดูลไฟล์ LDAPA โดยการสร้างกลุ่มที่สอดคล้องกันนี้ คุณ สามารถหลีกเลี่ยงกรณีที่ที่ผู้ใช้ไฟล์ LDAPA ไม่สามารถแก้ไขการตั้งค่ากลุ่ม ได้เนื่องจากการตั้งค่ากลุ่มไม่มีอยู่บนเซิร์ฟเวอร์ฐานข้อมูล

หมายเหตุ: โมดูล LDAPA ไม่สนับสนุนการสร้างและการลบผู้ใช้ออก ในการสร้าง ผู้ใช้ไฟล์ LDAPA ผู้ดูแลระบบต้องสร้างผู้ใช้ LDAP โดยใช้โมดูล LDAP จากนั้นสร้างผู้ใช้เดียวกันแบบโลคัล จากนั้นกำหนดให้ผู้ใช้เป็นผู้ใช้ไฟล์ LDAPA โดยการตั้งค่า SYSTEM และ รีจิสทรีของผู้ใช้ไปที่ LDAPAfiles โดยใช้คำสั่ง **chuser**

ในการ ตั้งค่า LDAP ในโหมดการพิสูจน์ตัวตนเท่านั้นโดยใช้โมดูล LDAPA ใช้คำสั่ง **mksecldap** ที่มีอ็อพชัน -i *<databaseModule>* คำสั่งนี้สร้างโมดูล LDAPA ที่มีการตั้งค่า options = authonly และโหลดโมดูลผสม LDAPA *<databaseModule>*

ตัวอย่าง ในการตั้งค่า LDAP ในโหมดการพิสูจน์ตัวตนเท่านั้น และในการใช้ไฟล์โลคัลสำหรับโมดูลฐานข้อมูล ให้ใช้ตัวอย่างต่อไปนี้:

```
mksecldap -c –h <ldap server> -a <binddn> -p <bind password> -i files
```

ไฟล์ /usr/lib/security/methods.cfg ถูกอัพเดตด้วยค่าต่อไปนี้:

```
LDAPA:
        program = /usr/lib/security/LDAP
        program_64 =/usr/lib/security/LDAP64
        options = authonly

LDAP:
        program = /usr/lib/security/LDAP
        program_64 =/usr/lib/security/LDAP64

LDAPAfiles:
        options = db=BUILTIN,auth=LDAPA
```

ใน LDAPA stanza การตั้งค่า `options = authonly` บ่งชี้ว่าตั้งค่า โมดูล LDAPA เป็นโหมดการพิสูจน์ตัวตนเท่านั้น `LDAPAfiles` stanza กำหนดโหลดโมดูลผสม

โมดูล LDAP ถูกเก็บไว้เพื่อการแก้ไขข้อมูลที่มิใช่ผู้ใช้/กลุ่ม เช่น RBAC โมดูล LDAP ยังสามารถใช้เป็นโมดูลการพิสูจน์ตัวตนสแตนด์อะโลนอิสระของโมดูล LDAPA

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง mksecldap

แอ็ตทริบิวต์ที่สนับสนุน LDAPA:

โมดูล LDAPA ในโหมดการพิสูจน์ตัวตนเท่านั้นสนับสนุน จำนวนของแอ็ตทริบิวต์นโยบายรหัสผ่าน AIX ที่จำกัด ส่วนที่เหลือของแอ็ตทริบิวต์ AIX เป็นไปตาม โมดูลฐานข้อมูล

โมดูล LDAPA การพิสูจน์ตัวตนเท่านั้นสนับสนุนแอ็ตทริบิวต์ต่อไปนี้:

- maxage
- minage
- minlen
- lastupdate
- flags
- maxrepeats
- minalpha
- mindiff
- minother
- pwdwarntime
- pwdchecks
- histsize
- histexpire
- time_last_login
- time_last_unsuccessful_login
- tty_last_login

- tty_last_unsuccessful_login
- host_last_login
- host_last_unsuccessful_login
- unsuccessful_login_count
- account_locked
- loginretries
- logintimes

เฉพาะบางเซิร์ฟเวอร์ LDAP ที่สนับสนุนแอ็ตทริบิวต์เหล่านี้ หากเซิร์ฟเวอร์ LDAP ไม่สนับสนุนแอ็ตทริบิวต์ที่แสดงรายการทั้งหมด แอ็ตทริบิวต์ที่สนับสนุน จะเป็นแอ็ตทริบิวต์ที่ใช้ร่วมกันทั้งในรายการนี้และใน ไฟล์การแม็พผู้ใช้-แอ็ตทริบิวต์เท่านั้น ไฟล์การแม็พนี้อยู่ในไดเร็กทอรี `/etc/security/ldap`

สำหรับเซิร์ฟเวอร์การยินยอม RFC2307 โดยไม่มีส่วนสนับสนุนสกีมา AIX, แอ็ตทริบิวต์ AIX ต่อไปนี้ได้รับการสนับสนุน:

- maxage
- minage
- lastupdate
- pwdwarntime
- lastupdate

การโยง Kerberos:

นอกเหนือจากการโยงแบบง่ายโดยใช้ DN การโยงและรหัสผ่านการโยงแล้ว **secldapclntd** daemon ยังสนับสนุนการโยงโดยใช้ Kerberos V credentials

คีย์สำหรับหลักการโยงถูกเก็บในไฟล์ keytab และจำเป็นต้อง ทำให้ใช้ได้กับ **secldapclntd** daemon เพื่อใช้การโยง Kerberos เมื่อเปิดใช้การโยง Kerberos **secldapclntd** daemon จะทำการพิสูจน์ตัวตน Kerberos ไปยังเซิร์ฟเวอร์ LDAP โดยใช้ชื่อหลักการและ keytab ที่ระบุในไฟล์คอนฟิกูเรชันไคลเอ็นต์ `/etc/security/ldap/ldap.cfg` การใช้การโยง Kerberos ทำให้ **secldapclntd** daemon ข้าม DN การโยงและรหัสผ่านการโยงที่ระบุในไฟล์ `/etc/security/ldap/ldap.cfg`

เมื่อการพิสูจน์ตัวตน Kerberos สำเร็จ **secldapclntd** daemon จะบันทึก credentials การโยงไปยังไดเร็กทอรี `/etc/security/ldap/krb5cc_secldapclntd` credentials ที่บันทึกไว้จะถูกใช้สำหรับการโยงอีกครั้งในภายหลัง ถ้า credentials มีอายุ มากกว่าหนึ่งชั่วโมงในเวลาที่ **secldapclntd** daemon พยายาม โยงกับเซิร์ฟเวอร์ LDAP อีกครั้ง **secldapclntd** daemon จะเตรียมข้อมูลเพื่อ ต่ออายุ credentials ใหม่อีกครั้ง

ในการตั้งค่าระบบไคลเอ็นต์ LDAP เพื่อใช้การโยง Kerberos คุณต้องตั้งค่า ไคลเอ็นต์โดยใช้คำสั่ง **mksecldap** โดยใช้ DN การโยงและรหัสผ่านการโยง ถ้าการตั้งค่าสำเร็จ ให้แก้ไขไฟล์ `/etc/security/ldap/ldap.cfg` ด้วยค่าที่ถูกต้องสำหรับแอ็ตทริบิวต์ที่เกี่ยวกับ Kerberos **secldapclntd** daemon ใช้การโยง Kerberos เมื่อเริ่มทำต่อ หลังจากการตั้งค่าสำเร็จ DN การโยงและรหัสผ่านการโยงจะไม่ถูกใช้อีกต่อไป โดยสามารถลบออกได้อย่างปลอดภัย หรือใส่เครื่องหมายความคิดเห็นในไฟล์ `/etc/security/ldap/ldap.cfg`

การสร้าง Kerberos principal:

คุณจำเป็นต้องสร้างอย่างน้อยสอง principals บน Key Distribution Center (KDC) เพื่อใช้โดยเซิร์ฟเวอร์และไคลเอ็นต์ IDS เพื่อสนับสนุน การโยง Kerberos principal แรกคือ principal ของเซิร์ฟเวอร์ LDAP และ อันที่สองคอ principal ที่ใช้โดยระบบไคลเอ็นต์เพื่อโยงกับ เซิร์ฟเวอร์

แต่ละคีย์ principal ต้องวางในไฟล์ keytab เพื่อให้สามารถใช้เริ่มทำงานประมวลผลเซิร์ฟเวอร์หรือการประมวลผล daemon ไคลเอ็นต์ได้

ตัวอย่างต่อไปนี้ยึดตาม IBM Network Authentication Service ถ้าคุณติดตั้งซอฟต์แวร์ Kerberos จากซอร์สอื่นๆ คำสั่งที่แท้จริงอาจแตกต่างกับที่แสดงในที่นี้

- เริ่มทำงานเครื่องมือ kadmin บนเซิร์ฟเวอร์ KDC ในฐานะผู้ใช้ root

  ```
  #/usr/krb5/sbin/kadmin.local
  kadmin.local:
  ```

- สร้าง ldap/*serverhostname* principal สำหรับเซิร์ฟเวอร์ LDAP *serverhostname* เป็นโฮสต์ DNS แบบเต็มที่จะ รันเซิร์ฟเวอร์ LDAP

  ```
  kadmin.local: addprinc ldap/plankton.austin.ibm.com
  WARNING: no policy specified for "ldap/plankton.austin.ibm.com@ud3a.austin.ibm.com":
  Re-enter password for principal "ldap/plankton.austin.ibm.com@ud3a.austin.ibm.com":
  Principal "ldap/plankton.austin.ibm.com@ud3a.austin.ibm.com" created.
  kadmin.local:
  ```

- สร้าง keytab สำหรับ principal เซิร์ฟเวอร์ที่สร้าง คีย์นี้จะ ใช้โดยเซิร์ฟเวอร์ LDAP ระหว่างเริ่มทำงานเซิร์ฟเวอร์ ในการสร้าง keytab ชื่อ `slapd_krb5.keytab`:

  ```
  kadmin.local: ktadd -k /etc/security/slapd_krb5.keytab ldap/plankton.austin.ibm.com
  Entry for principal ldap/plankton.austin.ibm.com with kvno 2,
  encryption type Triple DES cbc mode with HMAC/sha1 added to keytab
  WRFILE:/etc/security/slapd_krb5.keytab.
  Entry for principal ldap/plankton.austin.ibm.com with kvno 2,
  encryption type ArcFour with HMAC/md5 added to keytab WRFILE:/etc/security/slapd_krb5.keytab.
  Entry for principal ldap/plankton.austin.ibm.com with kvno 2,
  encryption type AES-256 CTS mode with 96-bit SHA-1 HMAC added to keytab
  WRFILE:/etc/security/slapd_krb5.keytab.
  Entry for principal ldap/plankton.austin.ibm.com with kvno 2,
  encryption type DES cbc mode with RSA-MD5 added to keytab WRFILE:/etc/security/slapd_krb5.keytab.
  kadmin.local:
  ```

- สร้าง principal ชื่อ ldapadmin สำหรับผู้ดูแลระบบ IDS

  ```
  kadmin.local: addprinc ldapadmin
  WARNING: no policy specified for ldapadmin@ud3a.austin.ibm.com; defaulting to no policy.
  Note that policy may be overridden by ACL restrictions.
  Enter password for principal "ldapadmin@ud3a.austin.ibm.com":
  Re-enter password for principal "ldapadmin@ud3a.austin.ibm.com":
  Principal "ldapadmin@ud3a.austin.ibm.com" created.
  kadmin.local:
  ```

- สร้าง keytab สำหรับ principal การโยง `kdapadmin.keytab` คีย์นี้สามารถใช้โดย daemon ไคลเอ็นต์ **secldapclntd**

  ```
  kadmin.local: ktadd -k /etc/security/ldapadmin.keytab ldapadmin
  Entry for principal ldapadmin with kvno 2, encryption type
  Triple DES cbc mode with HMCA/sha1 added to keytab WRFILE:/etc/security/ldapadmin.keytab.
  ```

```
Entry for principal ldapadmin with kvno 2, encryption type
ArcFour with HMAC/md5 added to keytab WRFILE:/etc/security/ldapadmin.keytab.
Entry for principal ldapadmin with kvno 2, encryption type
AES-256 CTS mode with 96-bit SHA-1 HMAC added to keytab WRFILE:/etc/security/ldapadmin.keytab.
Entry for principal ldapadmin with kvno 2, encryption type
DES cbc mode with RSA-MD5 added to keytab WRFILE:/etc/security/ldapadmin.keytab.
kadmin.local
```

- สร้าง principal ชื่อ ldapproxy สำหรับไคลเอ็นต์เพื่อโยงกับ เซิร์ฟเวอร์ LDAP

```
kadmin.local: addprinc ldapproxy
WARNING: no policy specified for ldapproxy @ud3a.austin.ibm.com; defaulting to no policy.
Note that policy may be overridden by ACL restriction
Enter password for principal "ldapproxy@ud3a.austin.ibm.com":
Re-enter password for principal "ldapproxy@ud3a.austin.ibm.com":
Principal "ldapproxy@ud3a.austin.ibm.com" created.
kadmin.local:
```

- สร้าง keytab ชื่อ *ldapproxy.keytab* สำหรับ principal การโยง **ldapproxy** คีย์นี้สามารถใช้โดย daemon ไคลเอ็นต์ **secldapclntd**

```
kadmin.local: ktadd -k /etc/security/ldapproxy.keytab ldapproxy
Entry for principal ldapproxy with kvno 2, encryption type
Triple DES cbc mode with HMAC/sh1 added to keytab WRFILE:/etc/security/ldapproxy.keytab.
Entry for principal ldapproxy with kvno 2, encryption type
ArcFour with HMAC/md5 added to keytab WRFILE:/etc/security/ldapproxy.keytab
Entry for principal ldapproxy with kvno 2, encryption type
AES-256 CTS mode with 96-bit SHA-1 HMAC added to keytab WRFILE:/etc/security/ldapproxy.keytab
Entry for principal ldapproxy with kvno 2,
encryption type DES cbc mode with RSA-MD5 added to keytab WRFILE:/etc/security/ldapproxy.keytab.
kadmin.local:
```

การเปิดใช้การโยง Kerberos เซิร์ฟเวอร์ IDS:

โพรซีเดอร์ต่อไปนี้เปิดใช้การโยงเซิร์ฟเวอร์ IDS สำหรับ Kerberos

ตัวอย่างต่อไปนี้แสดงวิธีตั้งค่าการโยง เซิร์ฟเวอร์ IDS สำหรับ Kerberos

ตัวอย่างนี้ได้รับการทดสอบโดยใช้ IDS v5.1:

1. ติดตั้งชุดไฟล์ krb5.client
2. ตรวจให้แน่ใจว่าไฟล์ /etc/krb5/krb5.conf มีอยู่และตั้งค่าอย่างเหมาะสม ถ้าคุณจำเป็นต้องตั้งค่าไฟล์ คุณ สามารรันคำสั่ง **/usr/sbin/config.krb5**

```
# config.krb5 -r ud3a.austin.ibm.com -d austin.ibm.com -c KDC -s alyssa.austin.ibm.com
Initializing configuration...
Creating /etc/krb5/krb5_cfg_type...
Creating /etc/krb5/krb5.conf...
The command completed successfully.
# cat /etc/krb5/krb5.conf
[libdefaults]
     default_realm = ud3a.austin.ibm.com
     default_keytab_name = FILE:/etc/krb5/krb5.keytab
     default_tkt_enctypes = des3-cbc-sha1 arcfour-hmac aes256-cts des-cbc-md5 des-cbc-crc
     defaut_tgs_enctypes = des3-cbc-shal1 arcfour-hmac aes256-cts des-cbc-md5 des-cbc-crc
[realms]
```

```
  ud3a.austin.ibm.com = {
     kdc = alyssa.austin.ibm.com:88
     admin_server = alyssa.austin.ibm.com:749
     default_domain = austin.ibm.com
  }

[domain_realm]
     .austin.ibm.com = ud3a.austin.ibm.com
     alyssa.austin.ibm.com = ud3a.austin.ibm.com

[logging]
     kdc = FILE:/var/krb5/log/krb5
     admin_server = FILE:/var/krb5/log/kadmin.log
     default = FILE:/var/krb5/log/krb5lib.log
```

3. รับไฟล์ keytab ของ ldap:*/serverhostname* principal และวางไว้ในไดเร็กทอรี `/usr/ldap/etc` ตัวอย่าง: `/usr/ldap/etc/slapd_krb5.keytab`
4. ตั้งค่าสิทธิเพื่ออนุญาตให้กระบวนการเซิร์ฟเวอร์เข้าถึง ไฟล์

```
# chown ldap:ldap/usr/ldap/etc/slapd_krb5.keytab
#
```

5. ในการเปิดใช้งานเซิร์ฟเวอร์ IDS สำหรับการโยง Kerberos แก้ไขไฟล์ `/etc/ibmslapd.conf` และต่อท้ายด้วยรายการต่อไปนี้:

```
dn: cn=Kerberos, cn-Configuration
cn: Kerberos
ibm-slapdKrbAdminDN: ldapadmin
ibm-slapdKrbEnable: true
ibm-slapdKrbIdentityMap: true
ibm-slapdKrbKeyTab: /usr/ldap/etc/slapd_krb5.keytab
ibm-slapdKrbRealm: ud3a.austin.ibm.com
objectclass: ibm-slapdKerberos
objectclass: ibm-slapdconfigEntry
objectclass: top
```

6. แม็พ ldapproxy principal กับ DN การโยงชื่อ cn=proxyuser,cn=aixdata
   a. ถ้ารายการ DN การโยงมีอยู่ในเซิร์ฟเวอร์ IDS ให้สร้าง ไฟล์ชื่อ `ldapproxy.ldif` ด้วยเนื้อหา ต่อไปนี้:

```
dn: cn=proxyuser,cn=aixdata
changetype: modify
add: objectclass
objectclass: ibm-securityidentities
-
add:altsecurityidentities
altsecurityidentities: Kerberos:ldapproxy@ud3a.austin.ibm.com
```

   หรือ

   b. ถ้ายังไม่ได้เพิ่มรายการ DN การโยงในเซิร์ฟเวอร์ให้สร้างไฟล์ชื่อ `proxyuser.ldif` ด้วยเนื้อหา ต่อไปนี้:

หมายเหตุ: คุณจะต้องแทนที่ *proxyuserpwd* ด้วย รหัสผ่านของคุณ

```
dn: cn=proxyuser,cn=mytest
cn: proxyuser
sn: proxyuser
userpassword: proxyuserpwd
```

```
objectclass: person
objectclass: top
objectclass: ibm-securityidentities
altsecurityidentities: Kerberos:ldapproxy@ud3a.austin.ibm.com
```

เพิ่ม รายการ DN การโยงที่สร้างไปยังเซิร์ฟเวอร์ IDS โดยใช้คำสั่ง **ldapmodify**

```
# ldapmodify -D cn-admin -w adminPwd -f /tmp/proxyuser.ldif modifying entry cn=proxyuser,cn=mytest
#
```

7. เริ่มทำงานเซิร์ฟเวอร์ IDS ต่อ

การเปิดใช้งานการโยงไคลเอ็นต์ AIX LDAP Kerberos:

คุณสามารถตั้งค่าระบบไคลเอ็นต์ AIX LDAP เพื่อใช้ Kerberos ในการโยงเกับเซิร์ฟเวอร์ LDAP เริ่มต้น

เซิร์ฟเวอร์ IDS ต้องถูกตั้งค่าในลักษณะนี้สำหรับให้โฮสต์เซิร์ฟเวอร์ ทำหน้าที่เป็นไคลเอ็นต์ด้วย

ตัวอย่างนี้ได้รับการทดสอบโดยใช้ IDS v 5.1:

1. ติดตั้งชุดไฟล์ `krb5.client`
2. ตรวจให้แน่ใจว่าไฟล์ `/etc/krb.conf` มีอยู่และตั้งค่าอย่างเหมาะสม ถ้ายังตั้งค่าไม่เหมาะสม คุณสามารถรันคำสั่ง **/usr/sbin/config.krb5** เพื่อตั้งค่าใหม่
3. รับไฟล์ keytab ของ bind principal และวางในไดเร็กทอรี `/etc/security/ldap`
4. ตั้งค่าสิทธิเป็น 600
5. ตั้งค่าไคลเอ็นต์โดยใช้คำสั่ง **mksecldap** โดยใช้ DN การโยงและรหัสผ่านการโยง ตรวจสอบให้แน่ใจว่าคำสั่ง AIX ทำงานบนผู้ใช้ LDAP
6. แก้ไขไฟล์ `/etc/security/ldap/ldap.cfg` เพื่อตั้งค่าแอ็ตทริบิวต์ที่เกี่ยวข้องกับ Kerberos ในตัวอย่างต่อไปนี้ bind principal คือ **ldapproxy** และไฟล์ keytab คือ **ldapproxy.keytab** ถ้าคุณต้องการสิทธิพิเศษผู้ดูแลระบบเซิร์ฟเวอร์ IDS ให้แทน *ldapproxy* ด้วย *ldapadmin* และ แทน *ldapproxy.keytab* ด้วย *ldapadmin.keytab*

   ```
   useKRB5:yes
   krbprincipal:ldapproxy
   krbkeypath:/etc/security/ldap/ldapproxy.keytab
   krbcmddir:/usr/krb5/bin/
   ```

   ในตอนนี้ DN การโยงและรหัสผ่าน การโยงสามารถลบออก หรือใม่เครื่องหมายความคิดเห็นในไฟล์ `ldap.cfg` เนื่องจากขณะนี้ **secldapclntd** daemon ในการโยง Kerberos
7. เริ่มทำงาน **secldapclntd** daemon ต่อ
8. ขณะนี้ไฟล์ `/etc/security/ldap/ldap.cfg` สามารถถูกกระจายไปยังระบบไคลเอ็นต์อื่นๆ

การตรวจสอบ LDAP security information server:

SecureWay Directory เวอร์ชัน 3.2 (และใหม่กว่า) มีฟังก์ชันบันทึกการตรวจสอบเซิร์ฟเวอร์ดีฟอลต์ เมื่อเปิดใช้งาน, ปลั๊กอินการตรวจสอบดีฟอลต์นี้ ล็อกกิจกรรมเซิร์ฟเวอร์ LDAP ไปยังล็อกไฟล์ โปรดดูเอกสารคู่มือ LDAP ใน *คู่มือแพ็กเกจสำหรับการติดตั้ง LPP* สำหรับข้อมูลเพิ่มเติม เกี่ยวกับปลั๊กอินดีฟอลต์

ฟังก์ชันการตรวจสอบเซิร์ฟเวอร์ข้อมูลความปลอดภัย LDAP ที่จัดเตรียมไว้พร้อมกับระบบปฏิบัติการ AIX ถูกเรียกว่า *ปลั๊กอินการตรวจสอบความปลอดภัย* ซึ่งเป็นอิสระต่อกัน ของเซอร์วิสการตรวจสอบดีฟอลต์ SecureWay Directory ดังนั้นสามารถ

เลือกส่วนใดส่วนหนึ่งหรือทั้งสองของระบบย่อยการตรวจสอบเหล่านี้ เปิดใช้งานได้ ปลั๊กอินการตรวจสอบ AIX เร็กคอร์ดเหตุการณ์เหล่านั้นที่อัพเดตหรือเคียวรีข้อมูลความปลอดภัย AIX เกี่ยวกับเซิร์ฟเวอร์ LDAP เท่านั้น โดยทำงานภายในเฟรมเวิร์กของการตรวจสอบ AIX

ในการจัดเตรียม LDAP เหตุการณ์การตรวจสอบต่อไปนี้มีอยู่ใน ไฟล์ /etc/security/audit/event:

- LDAP_Bind
- LDAP_Unbind
- LDAP_Add
- LDAP_Delete
- LDAP_Modify
- LDAP_Modifydn
- LDAP_Search

นิยามคลาสการตรวจสอบ ldapserver ยังถูกสร้าง ในไฟล์ /etc/security/audit/config ที่มี เหตุการณ์ด้านบนทั้งหมด

ในการตรวจสอบ LDAP security information server ให้เพิ่มบรรทัด ต่อไปนี้ให้ stanza ของผู้ใช้แต่ละรายในไฟล์ /etc/security/audit/config:

```
ldap = ldapserver
```

เนื่องจากปลั๊กอินการตรวจสอบ LDAP security information server ถูกนำใช้ ภายในกรอบของการตรวจสอบระบบ AIX ถือเป็นส่วนหนึ่งของระบบย่อยการตรวจสอบระบบ AIX เปิดใช้งานหรือปิดใช้งานการตรวจสอบเซิร์ฟเวอร์ข้อมูลความปลอดภัย LDAP โดยใช้คำสั่งการตรวจสอบ ระบบ, เช่น **audit start** หรือ **audit shutdown** เร็กคอร์ดการตรวจสอบทั้งหมดถูกเพิ่มลงในหลักฐานการตรวจสอบ ระบบ ซึ่งสามารถตรวจทานได้ด้วยคำสั่ง **auditpr** สำหรับข้อมูลเพิ่มเติม ดูที่ “ภาพรวมการตรวจสอบ” ในหน้า 150

คำสั่ง LDAP:

มีคำสั่ง LDAP หลายคำสั่ง

คำสั่ง lsldap

คำสั่ง **lsldap** สามารถใช้เพื่อแสดง entity เซอร์วิสการกำหนดชื่อจากเซิร์ฟเวอร์ LDAP ที่ตั้งค่า entities เหล่านี้ได้แก่ aliases, automount, bootparams, ethers, groups, hosts, netgroups, networks, passwd, protocols, rpc และ services

คำสั่ง mksecldap

คำสั่ง **mksecldap** สามารถใช้เพื่อตั้งค่าเซิร์ฟเวอร์และไคลเอ็นต์ IBM SecureWay Directory สำหรับการพิสูจน์ตัวตนด้านความปลอดภัย และการจัดการข้อมูล คำสั่ง นี้ต้องรันบนเซิร์ฟเวอร์และไคลเอ็นต์ทั้งหมด

secldapclntd daemon

secldapclntd daemon รับการร้องขอจาก โหมดโมดูล LDAP ส่งการร้องขอต่อไปยัง LDAP Security Information Server และส่งผลลัพธ์จากเซิร์ฟเวอร์กลับไปที่โหลดโมดูล LDAP

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับรูปแบบไฟล์การแม็พแอ็ตทริบิวต์ LDAP ดูที่ **รูปแบบไฟล์การแม็พแอ็ตทริบิวต์ LDAP** ใน *การอ้างอิงไฟล์*

ข้อมูลที่เกี่ยวข้อง

คำสั่ง mksecldap, start-secldapclntd, stop-secldapclntd, restart-secldapclntd, ls-secldapclntd, sectoldif และ **flush-secldapclntd**

secldapclntd daemon

ไฟล์ /etc/security/ldap/ldap.cfg

รูปแบบไฟล์การแม็พแอ็ตทริบิวต์ LDAP

คำสั่งการจัดการ LDAP:

หลายคำสั่งถูกใช้สำหรับการจัดการ LDAP

คำสั่ง start-secldapclntd

คำสั่ง **start-secldapclntd** เริ่มทำงาน **secldapclntd** daemon ถ้ายังไม่ได้ทำงาน

คำสั่ง stop-secldapclntd

คำสั่ง **stop-secldapclntd** จบการรันการประมวลผล **secldapclntd** daemon

คำสั่ง restart-secldapclntd

สคริปต์ **restart-secldapclntd** หยุดทำงาน **secldapclntd** daemon ถ้ารันอยู่ จากนั้นรีสตาร์ท ถ้า **secldapclntd** daemon ไม่ได้กำลังรัน ให้เริ่มทำงาน

คำสั่ง ls-secldapclntd

คำสั่ง **ls-secldapclntd** แสดงรายการสถานะ **secldapclntd** daemon

คำสั่ง flush-secldapclntd

คำสั่ง **flush-secldapclntd** ล้างค่าแคชสำหรับการประมวลผล **secldapclntd** daemon

คำสั่ง sectoldif

คำสั่ง **sectoldif** อ่านผู้ใช้และ กลุ่มที่กำหนดแบบโลคัล และพิมพ์ผลลัพธ์ไปยังเอาต์พุตมาตรฐานในรูปแบบ **ldif**

รูปแบบไฟล์การแม็พสำหรับแอ็ตทริบิวต์ LDAP:

ไฟล์แม็พเหล่านี้ใช้โดยโมดูล /usr/lib/security/LDAP และ **secldapclntd** daemon เพื่อทำการแปลระหว่างชื่อแอ็ตทริบิวต์ AIX ไปเป็น ชื่อแอ็ตทริบิวต์ LDAP

แต่ละรายการในไฟล์การแม็พแทนการแปล สำหรับแอ็ตทริบิวต์ รายการที่ไฟล์คั่นด้วยช่องว่างสี่ฟิลด์:

```
AIX_Attribute_Name AIX_Attribute_Type LDAP_Attribute_Name LDAP_Value_Type
```

คำอธิบายสำหรับฟิลด์เหล่านี้มีดังนี้:

AIX_Attribute_Name
ระบุชื่อแอ็ตทริบิวต์ AIX

AIX_Attribute_Type
ระบุประเภทแอ็ตทริบิวต์ AIX ค่า ได้แก่ SEC_CHAR, SEC_INT, SEC_LIST และ SEC_BOOL

LDAP_Attribute_Name
ระบุชื่อแอ็ตทริบิวต์ LDAP

LDAP_Value_Type
ระบุประเภทค่า LDAP ค่าได้แก่ **s** สำหรับ ค่าเดียวและ **m** สำหรับหลายค่า

LDAP และ KRB5LDAP ในไคลเอ็นต์เดี่ยว

ถ้า LDAP เป็นส่วนหนึ่งของโมดูลผสม เช่น KRB5LDAP เฉพาะการอ่านเท่านั้นที่สามารถดำเนินการได้ ไม่ใช่การเขียน อย่างไรก็ตาม ด้วยการเปลี่ยนแปลงคอนฟิกูเรชันด้านล่างในไฟล์ **/usr/lib/security/methods.cfg** ทั้ง LDAP และโมดูลโหลดผสม เช่น KRB5LDAP จะถูกจัดให้เป็นไฟล์เดียวกัน โดยปฏิบัติตามขั้นตอนต่อไปนี้:

1. กำหนดคอนฟิกไคลเอ็นต์ LDAP และไคลเอ็นต์ KRB5LDAP ตามปกติ
2. แก้ไขไฟล์ **/usr/lib/security/methods.cfg** ดังต่อไปนี้:

```
LXAP:   program = /usr/lib/security/LDAP program_64
        =/usr/lib/security/LDAP64

LDAP:   program = /usr/lib/security/LDAP program_64
        =/usr/lib/security/LDAP64

NIS:    program = /usr/lib/security/NIS program_64 =
        /usr/lib/security/NIS_64

DCE:    program = /usr/lib/security/DCE

KRB5:   program = /usr/lib/security/KRB5


KRB5LXAP: options = db=LXAP,auth=KRB5
```

3. แก้ไขไฟล์ **/etc/security/user** สำหรับค่าดีฟอลต์ของ stanza ดังต่อไปนี้:

```
        SYSTEM = "KRB5LXAP OR LDAP OR compat"
```

ผู้ใช้ LDAP สามารถประมวลผลได้ตามปกติ ตัวอย่างต่อไปนี้จะแสดงการประมวลผลของผู้ใช้ KRB5LDAP:

```
mkuser -R KRB5LXAP <user_name>
rmuser -R KRB5LXAP <user_name>
lsuser -R KRB5LXAP <user_name>
passwd -R KRB5LXAP <user_name>
```

EFS Encrypted File System

Encrypted Files System เปิดให้ผู้ใช้แต่ละคนบนระบบ เข้ารหัสข้อมูลของตนบนระบบไฟล์ J2 ผ่านทางที่เก็บคีย์ของแต่ละคน

คีย์ถูกเชื่อมโยงเข้ากับผู้ใช้แต่ละคน คีย์เหล่านี้ถูกเก็บในที่เก็บคีย์ที่ได้รับการป้องกัน โดยการเข้ารหัสและและเมื่อล็อกอินสำเร็จ คีย์ของผู้ใช้ถูกโหลด มาไว้ในเคอร์เนล และเชื่อมโยงกับ credentials กระบวนการ ภายหลัง เมื่อ กระบวนการต้องการเปิดไฟล์ที่ป้องกันด้วย EFS credentials เหล่านี้จะถูกทดสอบ และถ้าพบคีย์ที่ตรงกับการป้องกันไฟล์ กระบวนการจะสามารถ ถอดรหัสคีย์ไฟล์และต่อด้วยเนื้อหาไฟล์ สนับสนุนการจัดการคีย์ตามกลุ่ม เช่นกัน

หมายเหตุ: EFS เป็นส่วนหนึ่งของยุทธวิธีการรักษาความปลอดภัยโดยรวม โดยออกแบบมาให้ทำงาน ร่วมกับแนวทางการรักษาความปลอดภัยคอมพิวเตอร์เสียและการควบคุม

การใช้งาน Encrypted File System

การจัดการคีย์ Encrypted File System (EFS) การเข้ารหัสไฟล์ และการเข้ารหัสไฟล์เห็นได้โดยผู้ใช้ในการดำเนินการปกติ

EFS คือส่วนหนึ่งของระบบปฏิบัติการ AIX หลัก ในการเปิดใช้งาน EFS นั้น root (หรือผู้ใช้ใดที่ได้รับอนุญาต RBAC **aix.security.efs** ดูที่ EFS actors เพื่อ ดูข้อมูลเพิ่มเติม) ต้องใช้คำสั่ง **efsenable** เพื่อเรียกทำงาน EFS และสร้างสภาวะแวดล้อม EFS นี้เป็นการเปิดใช้งาน ระบบเพียงครั้งเดียว หลังจาก EFS ถูกเปิดใช้งาน เมื่อผู้ใช้ล็อกอิน คีย์และที่เก็บคีย์จะถูกสร้างขึ้นและป้องกัน หรือเข้ารหัสด้วย รหัสผ่านล็อกอินของผู้ใช้ จากนั้นคีย์ผู้ใช้จะถูกใช้โดยระบบไฟล์ J2 เมื่อทำการเข้ารหัสหรือถอดรหัสไฟล์ EFS ทุกไฟล์ EFS ได้รับการป้องกันด้วยคีย์ไฟล์เฉพาะของไฟล์นั้น และในทางกลับกันคีย์ไฟล์นี้ ก็ได้รับการป้องกันหรือเข้ารหัสด้วยคีย์เจ้าของไฟล์ หรือกลุ่ม ทั้งนี้ขึ้นอยู่กับสิทธิของไฟล์

โดยดีฟอลต์ ระบบไฟล์ J2 ไม่ถูกเปิดใช้งาน EFS เมื่อถูกเปิดใช้งาน EFS ระบบไฟล์ J2 จะจัดการการเข้ารหัสและการถอดรหัสในเคอร์เนลสำหรับการร้องขอเพื่ออ่านและเขียน คำสั่งการดูแล ผู้ใช้และกลุ่ม (เช่น **mkgroup, chuser** และ **chgroup**) จะจัดการแบบไม่แสดงให้เห็นกับที่เก็บคีย์ของผู้ใช้ และของกลุ่ม

คำสั่ง EFS ต่อไปนี้จัดให้มีเพื่ออนุญาตให้ผู้ใช้สามารถจัดการ คีย์และการเข้ารหัสไฟล์ของตน:

efskeymgr
จัดการและดูแลคีย์

efsmgr จัดการการเข้ารหัสไฟล์/ไดเร็กทอรี/ระบบไฟล์

ผู้ดำเนินการ Encrypted File System

มีผู้ใช้สามประเภทที่สามารถจัดการและใช้คีย์ EFS:

การเข้าถึงแบบเต็มหรือแบบจำกัดเมื่อเป็น root:

การเข้าถึงคีย์โดยเป็น root สามารถเป็นแบบไม่จำกัด หรือแบบจำกัด ไม่ว่าใน โหมดใด root ไม่สามารถทำได้เพียง **su** ไปยังผู้ใช้แล้วจะได้รับการเข้าถึงไฟล์และที่เก็บคีย์ที่เข้ารหัสของผู้ใช้

ในโหมดหนึ่ง root สามารถตั้งค่ารหัสผ่านที่เก็บคีย์ของผู้ใช้ใหม่ และอาจได้รับการเข้าถึง คีย์ของผู้ใช้ภายในที่เก็บคีย์นี้ โหมดนี้ช่วยให้มีความยืดหยุ่นในการการจัดการ ระบบมากขึ้น

ในโหมดอื่น root สามารถตั้งค่ารหัสผ่านล็อกออนของผู้ใช้ใหม่ แต่ไม่สามารถตั้งค่า รหัสผ่านที่เก็บคีย์ของผู้ใช้ใหม่ได้ root ไม่สามารถเปลี่ยนเป็นผู้ (ด้วยคำสั่ง **su**) และสืบทอดที่เก็บคีย์ที่เปิด ในขณะที่ root สามารถ สร้างและลบผู้ใช้และกลุ่ม รวมทั้งที่เก็บคีย์ที่กุญแจเชื่อมโยง ก็ไม่สามารถได้รับสิทธิการเข้าถึงคีย์ภายในที่เก็บคีย์เหล่านี้ โหมดนี้มีระดับของ การป้องกันมากขึ้น เพื่อป้องกันการโจมตีจาก root ที่ประสงค์ร้าย

มีสองโหมดสำหรับการจัดการและการใช้ที่เก็บคีย์ Root Admin และ Root Guard รวมทั้งมีคีย์การดูแล EFS

คีย์การดูแล EFS เปิดใช้การเข้าถึงเพื่อตั้งค่ารหัสผ่านใหม่ให้แก่ที่เก็บคีย์ทั้งหมด ในโหมด Root Admin คีย์นี้อยู่ในที่เก็บคีย์พิเศษ **efs_admin** การเข้าถึงที่เก็บคีย์พิเศษ **efs_admin** ให้สิทธิเฉพาะผู้ใช้ ที่ได้รับอนุญาตเท่านั้น (ผู้ใช้ root และกลุ่มการรักษาความปลอดภัยในตอนทำการติดตั้ง หรือการอนุญาต RBAC **aix.security.efs**)

เมื่อที่เก็บคีย์อยู่ในโหมด Root Guard คีย์ที่มีอยู่ในที่เก็บคีย์นี้ ไม่สามารถถูกเรียกออกมาโดยปราศจากรหัสผ่านที่เก็บคีย์ที่ถูกต้อง วิธีนี้ช่วยให้มีการรักษาความปลอดภัย ที่เข้มแข็งเพื่อป้องกัน root ที่ประสงค์ร้าย แต่ก็อาจก่อให้เกิดปัญหาถ้าผู้ใช้ลืม รหัสผ่านของตน เนื่องจากไม่มีวิธีใดที่จะสร้างรหัสผ่านนั้นใหม่ได้หากปราศจาก คีย์ที่สูญไปในที่เก็บคีย์ และทำให้ผู้ใช้ไม่สามารถเข้าถึงข้อมูล ได้อีกต่อไป ในโหมดที่เก็บคีย์นี้ การดำเนินการบางอย่างไม่สามารถกระทำได้ในทันที และได้รับการกำหนดการให้เป็นการดำเนินการที่ค้างอยู่ การดำเนินการที่ค้างอยู่เหล่านี้ถูกสร้างขึ้น ในกรณีเช่นการเพิ่มหรือยกเลิกคีย์การเข้าถึงกลุ่มในที่เก็บคีย์ผู้ใช้ หรือการสร้างไพรเวตคีย์ใหม่ ซึ่งจัดการโดยเจ้าของที่เก็บคีย์

คีย์การจัดการ efs_admin:

ที่เก็บคีย์ efs_admin มีคีย์พิเศษที่สามารถเปิดที่เก็บคีย์ ใดๆ ของผู้ใช้หรือกลุ่มในโหมดผู้ดูแล root (ดีฟอลต์โหมด)

รหัสผ่านที่ใช้เปิดที่เก็บคีย์พิเศษนี้ถูกเก็บในที่เก็บคีย์ผู้ใช้ root และ ที่เก็บคีย์กลุ่มการรักษาความปลอดภัยเมื่อ EFS ถูกเรียกทำงาน รหัสผ่านนี้สามารถมอบให้แก่กลุ่มและผู้ใช้ อื่น หรือลบออกด้วยคำสั่ง **efskeymgr** คีย์นี้ รวมกับการอนุญาต RBAC **aix.security.efs** อนุญาตให้ผู้ใช้จัดการ EFS (นั่นคือ เข้าถึงที่เก็บคีย์ในโหมด ผู้ดูแล root)

ข้อควรพิจารณา efs_admin RBAC

บนระบบที่เปิดใช้ Role Based Access Control คำสั่ง **efs_admin** ถูกป้องกัน ด้วยการอนุญาต **aix.security.efs**

ที่เก็บคีย์ผู้ใช้:

ที่เก็บคีย์ผู้ใช้ถูกจัดการโดยอัตโนมัติสำหรับการดำเนินการทั่วไปโดยส่วนใหญ่ คำสั่ง **efskeymgr** ใช้เพื่อการบำรุงรักษางานและการใช้งาน EFS ระดับสูง ผู้ใช้สามารถสร้างไฟล์และไดเร็กทอรีที่เข้ารหัสด้วยคำสั่ง **efsmgr** การจัดการที่เก็บคีย์ถูกรวมเข้ากับคำสั่งผู้ดูแลผู้ใช้ส่วนใหญ่ ถ้าผู้ใช้ ถูกเพิ่มลงในกลุ่ม ผู้ใช้จะมีสิทธิเข้าถึงที่เก็บคีย์กลุ่ม โดยอัตโนมัติ

เจ้าของไฟล์ที่มีการเข้าถึง EFS ไฟล์สามารถใช้คำสั่ง **efsmgr** เพื่อให้สิทธิการเข้าถึง EFS แก่ผู้ใช้และกลุ่มรายอื่นๆ (คล้ายกับการควบคุมที่ เจ้าของไฟล์มีกับ ACLs ใน UNIX) ผู้ใช้สามารถเปลี่ยนรหัสผ่านของตนโดยไม่มีผลต่อกระบวนการที่กำลังทำงานแยกต่างหาก ภายใต้ UID เดียวกันกับที่เก็บคีย์แบบเปิด

ที่เก็บคีย์ Encrypted File System

ที่เก็บคีย์ได้รับการป้องกันด้วยรหัสผ่าน ผู้ใช้สามารถเลือกรหัสผ่าน ที่เก็บคีย์อื่นนอกเหนือจากล็อกอินรหัสผ่านของตน ในกรณีนี้ ที่เก็บคีย์ ไม่ถูกเปิดและพร้อมใช้ได้ระหว่างการล็อกอินมาตรฐานของผู้ใช้ ผู้ใช้ต้อง โหลดที่เก็บคีย์โดยใช้คำสั่ง **efskey** เพื่อให้รหัสผ่านที่เก็บคีย์แทน

รูปแบบที่เก็บคีย์คือ **PKCS # 12** ที่เก็บคีย์ถูกเก็บใน ไฟล์ต่อไปนี้:

ที่เก็บคีย์ผู้ใช้

/var/efs/users//keystore

ที่เก็บคีย์กลุ่ม

/var/efs/groups//keystore

ที่เก็บคีย์ efsadmin

/var/efs/efs_admin/keystore

ถ้าผู้ใช้ตั้งค่ารหัสผ่านล็อกออนของตนและรหัสผ่านที่เก็บคีย์เป็น รหัสผ่านเดียวกัน ที่เก็บคีย์ของผู้ใช้ถูกเปิดและใช้งานเมื่อผู้ใช้ล็อกอิน

ผู้ใช้สามารถใช้คำสั่ง EFS **efskeymgr** เพื่อเลือก ประเภทของอัลกอริทึมการเข้ารหัสและความยาวคีย์

การเข้าถึงที่เก็บคีย์ถูกสืบทอดโดยกระบวนการชายด์ใดๆ

โดยสนับสนุนการจัดการคีย์ตามกลุ่มเช่นกัน มีเพียงสมาชิกกลุ่มเท่านั้นที่สามารถ เพิ่มหรือลบคีย์กลุ่มในที่เก็บคีย์ของสมาชิก ถ้าที่เก็บคีย์กลุ่มอยู่ในโหมด ป้องกัน ที่เก็บคีย์ผู้ใช้มีไพรเวตคีย์ของผู้ใช้รวมถึงรหัสผ่าน เพื่อเปิดที่เก็บคีย์กลุ่มของผู้ใช้ซึ่งมีไพรเวตคีย์ของกลุ่ม

หมายเหตุ: ที่เก็บคีย์ EFS ถูกเปิดโดยอัตโนมัติเป็นส่วนหนึ่งของล็อกอิน AIX มาตรฐานเฉพาะเมื่อรหัสผ่าน ที่เก็บคีย์ของผู้ใช้ตรงกับรหัสผ่านล็อกอิน ค่านี้ถูกตั้งค่าโดยดีฟอลต์ระหว่าง การเริ่มสร้างที่เก็บคีย์ของผู้ใช้ วิธีล็อกอินที่นอกเหนือจากล็อกอิน AIX มาตรฐาน เช่นโมดูลการพิสูจน์ตัวตนที่โหลดได้ และโมดูลการพิสูจน์ตัวตนแบบปลั๊กได้อาจไม่เปิด ที่เก็บคีย์โดยอัตโนมัติ

การเข้ารหัสและการสืบทอด

EFS เป็นคุณลักษณะของ J2 อ็อพชัน **efs** ของระบบไฟล์ ต้องตั้งค่าเป็น **yes** (ดูที่คำสั่ง **mkfs** และ **chfs**)

J2 EFS เข้ารหัสและถอดรหัสข้อมูลผู้ใช้โดยอัตโนมัติ อย่างไรก็ตาม ถ้าผู้ใช้มีการเข้าถึงเพื่ออ่านไฟล์ที่เปิดทำงาน EFS แต่ไม่มีคีย์ที่ถูกต้อง ผู้ใช้จะไม่สามารถอ่านไฟล์ในรูปแบบปกติ ถ้าผู้ใช้ไม่มีคีย์ที่ถูกต้อง ก็จะไม่สามารถถอดรหัสข้อมูล ได้

ฟังก์ชันการเข้ารหัสทั้งหมดมาจากเคอร์เนลเซอร์วิส CLiC และไลบรารีผู้ใช้ CLiC

โดยดีฟอลต์ ระบบไฟล์ J2 ไม่เปิดใช้ EFS ระบบไฟล์ J2 ต้องเปิดใช้ EFS ก่อนที่การสืบทอด File System EFS จะสามารถเรียกทำงาน หรือการเข้ารหัส EFS ใดๆ ของข้อมูลผู้ใช้สามารถเกิดขึ้นได้ ไฟล์ถูกสร้าง ในรูปของไฟล์ที่เข้ารหัสโดยใช้คำสั่ง **efsmgr** โดยชัดเจน หรือโดยนัยผ่านทางการสืบทอด EFS การสืบทอด EFS สามารถเรียกทำงาน ได้ที่ระดับระบบไฟล์ ที่ระดับไดเร็กทอรี หรือทั้งสอง

คำสั่ง **ls** แสดงรายการของไฟล์ ที่เข้ารหัสที่มี **e** นำหน้า

คำสั่ง**cp** และ **mv** สามารถ จัดการ metadata และข้อมูลที่เข้ารหัสได้อย่างเรียบร้อยทั้งในสถานการณ์ EFS-กับ-EFS และ EFS-กับไม่ใช่-EFS

คำสั่ง **backup**, **restore** และ **tar** และคำสั่งที่เกี่ยวข้องสามารถสำรองข้อมูล และเรียกคืนข้อมูลที่เข้ารหัส รวมถึง EFS meta-data ที่ใช้สำหรับการเข้ารหัสและการถอดรหัส

การสำรองข้อมูลและเรียกคืน

ถือเป็นสิ่งสำคัญที่จะต้องจัดการการจัดเก็บลงสื่อถาวรหรือสำรองข้อมูล ที่เก็บคีย์ที่เชื่อมโยงกับไฟล์ EFS ที่จัดเก็บลงสื่อถาวร อย่างเหมาะสม คุณยังต้องจัดการและ รักษารหัสผ่านที่เก็บคีย์ที่เชื่อมโยงกับที่เก็บคีย์ที่จัดเก็บลงสื่อถาวร หรือสำรองข้อมูล การไม่ดำเนินการงานใดงานหนึ่งเหล่านี้อาจส่งผลให้ข้อมูลสูญหาย

เมื่อทำการทำสำเนาสำรองไฟล์ที่เข้ารหัส EFS คุณสามารถใช้อ็อพชัน **–Z** ด้วยคำสั่ง **backup** เพื่อสำรองข้อมูลรูปแบบที่เข้ารหัสของ ไฟล์ รวมถึง meta-data การเข้ารหัสของไฟล์ ทั้งข้อมูลไฟล์ และ meta-data ได้รับการป้องกันด้วยการเข้ารหัสที่น่าเชื่อถือ นี้มีประโยชน์ในด้าน การรักษาความปลอดภัยของการป้องกันไฟล์ที่ทำสำเนาสำรองด้วยการเข้ารหัสที่น่าเชื่อถือ ถือเป็น สิ่งจำเป็นที่จะต้องสำรองข้อมูลที่เก็บคีย์ของเจ้าของไฟล์และกลุ่มที่เชื่อมโยงกับ ไฟล์ที่กำลังถูกสำรองข้อมูล ที่เก็บคีย์เหล่านี้อยู่ในไฟล์ ต่อไปนี้:

ที่เก็บคีย์ผู้ใช้

/var/efs/users/*user_login*/*

ที่เก็บคีย์กลุ่ม

/var/efs/groups//keystore

ที่เก็บคีย์ efsadmin

/var/efs/efs_admin/keystore

ใช้คำสั่ง **restore** เพื่อเรียกคืนสำเนาสำรอง EFS (ที่ทำ โดยใช้คำสั่ง **backup** และอ็อพชัน **–Z**) คำสั่ง restore ทำให้แน่ใจว่า crypto-meta data ถูกเรียกคืนเช่นกัน ระหว่าง กระบวนการเรียกคืน ไม่จำเป็นต้องเรียกคืนที่เก็บคีย์ที่สำรองข้อมูล ถ้าผู้ใช้ไม่ได้เปลี่ยนคีย์ในที่เก็บคีย์ส่วนตัวของตน เมื่อ ผู้ใช้เปลี่ยนรหัสผ่านของตนเพื่อเปิดที่เก็บคีย์ คีย์ภายในที่เก็บคีย์ของผู้ใช้ จะไม่เปลี่ยนแปลง ใช้คำสั่ง **efskeymgr** เพื่อเปลี่ยน คีย์ภายในที่เก็บคีย์

ถ้าคีย์ภายในที่เก็บคีย์ของผู้ใช้ยังคงเหมือนเดิม ผู้ใช้สามารถ เปิดและถอดรหัสไฟล์ที่เรียกคืนได้ทันทีโดยใช้ที่เก็บคีย์ปัจจุบันของผู้ใช้ อย่างไรก็ตาม ถ้าภายในคีย์ของที่เก็บคีย์ของผู้ใช้เปลี่ยนแปลง ผู้ใช้ ต้องเปิดที่เก็บคีย์ที่ถูกสำรองข้อมูลที่สัมพันธ์กับไฟล์ ที่ถูกสำรองข้อมูล ที่เก็บคีย์คีย์สามารถเปิดด้วยคำสั่ง **efskeymgr –o** คำสั่ง**efskeymgr** พร้อมต์ให้ผู้ใช้ใส่รหัสผ่านเพื่อ เปิดที่เก็บคีย์ รหัสผ่านนี้ถูกใช้ร่วมกับที่เก็บคีย์ ในตอนทำการสำรองข้อมูล

ตัวอย่าง สมมติว่าที่เก็บคีย์ของผู้ใช้ Bob ได้รับการป้องกันด้วย รหัสผ่าน **foo** (รหัสผ่าน 'foo' ไม่ใช่รหัสผ่านที่ปลอดภัย และใช้ในตัวอย่างนี้เพื่อความง่ายเท่านั้น) และสำเนาสำรองของไฟล์ที่เข้ารหัสของ Bob ถูกดำเนินการในเดือนมกราคมพร้อมกับที่เก็บคีย์ของ Bob ในตัวอย่างนี้ บ๊อบยังใช้ **foo** สำหรับรหัสผ่านการล็อกอิน AIX ของตน ในเดือนกุมภาพันธ์ บ๊อบเปลี่ยนรหัสผ่านเป็น **bar** ซึ่ง มีผลต่อการเปลี่ยนรหัสผ่านการเข้าถึงที่เก็บคีย์ของตนเป็น **bar** เช่นกัน ถ้า ในเดือนมีนาคม ไฟล์ EFS ของบ๊อบถูกเรียกคืน เขาจะสามารถเปิดและ ดูไฟล์เหล่านี้ได้ด้วยที่เก็บคีย์ปัจจุบัน เนื่องจากเขาไม่ได้ เปลี่ยนคีย์ภายในของที่เก็บคีย์

อย่างไรก็ตามถ้าจำเป็นต้องเปลี่ยนคีย์ภายในของดีฟอลต์ของบ๊อบ (ด้วย คำสั่ง **efskeymgr**) โดยค่าดีฟอลต์แล้วคีย์ภายในของที่เก็บคีย์เก่าจะถูกเลิกใช้และเก็บในที่เก็บคีย์ของบ๊อบ เมื่อผู้ใช้เข้าถึง ไฟล์ EFS จะทราบโดยอัตโนมัติว่าไฟล์ที่เรียกคืนนั้นใช้ คีย์ภายในที่เป็นค่าเก่า จากนั้น EFS จะใช้คีย์ที่เลิกใช้เพื่อถอดรหัส ระหว่าง instance การเข้าถึงเดียวกันนี้ EFS จะแปลงไฟล์ให้เป็นใช้ คีย์ภายในใหม่ การทำงานนี้ไม่ส่งผลต่อผลการทำงานมากใน กระบวนการ เนื่องจากถูกจัดการผ่านที่เก็บคีย์ และ crypto meta-data ของไฟล์ และไม่ข้อมูลไฟล์ไม่จำเป็นต้องเข้ารหัสใหม่

ถ้าคีย์ภายในที่เลิกใช้ถูกลบโดยใช้ **efskeymgr** ที่เก็บคีย์เก่าที่มีคีย์ภายในเก่าต้องถูกเรียกคืนและ ใช้ร่วมกับไฟล์ที่เข้ารหัสด้วยคีย์ภายในนี้

นี้ทำให้เกิดคำถามว่าจะรักษาและจัดเก็บรหัสผ่านลงสื่อบันทึกถาวรอย่างปลอดภัยได้อย่างไร มีหลายวิธีและเครื่องมือหลายอย่างที่จะจัดก็บรหัสผ่านลงสื่อบันทึกถาวน โดยทั่วไป วิธีเหล่านี้ เกี่ยวข้องกับการดูว่าไฟล์ใดที่มีรายการรหัสผ่านเก่าทั้งหมด จากนั้น เข้ารหัสไฟล์เหล่านี้และป้องกันโดยใช้ที่เก็บคีย์ปัจจุบัน ซึ่งในทางกลับกัน จะได้รับการป้องกันโดยรหัสผ่านปัจจุบัน อย่างไรก็ตาม สภาวะแวดล้อมด้านไอทีและ นโยบายการรักษาความปลอดภัยจะแตกต่างกันไปในแต่ละองค์กร และข้อควรพิจารณาและความคิด จะก่อให้เกิดความต้องการที่เจาะจงกับความต้องการขององค์กรของคุณเพื่อพัฒนาเป็น นโยบายการรักษาความปลอดภัยและแนวปฏิบัติที่เหมาะสมที่สุดสำหรับสภาวะแวดล้อมของคุณ

กลไกภายใน J2 EFS

แต่ละไฟล์ที่เรียกทำงาน J2 EFS ถูกเชื่อมโยงกับแอ็ตทริบิวต์เพิ่มพิเศษ ที่มี EFS meta-data ที่ใช้ตรวจสอบความถูกต้องสิทธิการเข้ารหัสและ ข้อมูลที่ใช้เพื่อเข้ารหัสและถอดรหัสไฟล์ (คีย์ อัลกอริทึมการเข้ารหัส และอื่นๆ)

เนื้อหา EA ไม่ชัดเจนสำหรับ J2 ทั้ง credentials ผู้ใช้และ EFS meta-data จำเป็นต้องใช้เพื่อพิจารณาสิทธิการเข้ารหัส (ค่าควบคุมการเข้าใช้) สำหรับไฟล์ที่ เรียกทำงาน EFS

หมายเหตุ: ควรมีข้อควรพิจารณาพิเศษสำหรับสถานการณ์ที่ไฟล์หรือข้อมูล อาจสูญหาย (ตัวอย่าง การลบ EA ของไฟล์)

การสืบทอดการป้องกันด้วย EFS

หลังจากไดเร็กทอรีถูกเรียกทำงานแบบ EFS ชายด์ที่เพิ่งสร้างใหม่ใดๆ จะถูกเรียกทำงานแบบ EFS โดยอัตโนมัติถ้าไม่มีการแทนที่ด้วยตนเอง แอ็ตทริบิวต์ EFS ของพาเรนต์ไดเร็กทอรี มีความสำคัญเหนือแอ็ตทริบิวต์ EFS ของระบบไฟล์

ขอบเขตของการสืบทอดของไดเร็กทอรี คือหนึ่งระดับเท่านั้น ชายด์ที่เพิ่งสร้างขึ้นใหม่ยังสืบทอดแอ็ตทริบิวต์ EFS ของพาเรนต์ถ้าพาเรนต์ไดเร็กทอรีถูกเรียกใช้งาน EFS ชายด์ที่มีอยู่คงรักษาสถานะที่เข้ารหัส หรือไม่เข้ารหัส ปัจจุบันของตน ห่วงโซ่การการสืบทอดโลจิคัลจะขาดออกถ้าพาเรนต์เปลี่ยนแอ็ตทริบิวต์ EFS ของตน การเปลี่ยนแปลงเหล่านี้ไม่กระจายลงไปยังชายด์ต่างๆ ที่มีอยู่ของ ไดเร็กทอรี และต้องถูกนำใช้กับไดเร็กทอรีเหล่านั้นต่างหาก

การตรวจสอบพาร์ติชันเวิร์กโหลด

ก่อนการเปิดใช้งานหรือการใช้ Encrypted File System ภายใน Workload Partition อันดับแรก EFS ต้องถูกเปิดใช้งานบนระบบโกลบอลด้วยคำสั่ง **efsenable** การเปิดใช้งานนี้จำเป็นต้องดำเนินการเพียงครั้งเดียว นอกจากนั้น ระบบไฟล์ทั้งหมด รวมถึงระบบไฟล์ที่เปิดใช้ EFS ต้องสร้างขึ้นจากระบบโกลบอล

การตั้งค่า Encrypted File System

คุณต้องทำสิ่งนี้เป็นอันดับแรก

จำเป็นต้องตั้งค่าลำดับขั้นตามลำดับ

1. ติดตั้งชุดไฟล์ **clic.rte** ชุดไฟล์นี้มี ไลบรารีการเข้ารหัสลับ และส่วนขยายเคอร์เนลที่จำเป็นสำหรับ EFS ชุดไฟล์ **clic.rte** พบได้ใน AIX Expansion Pack
2. เปิดใช้งาน EFS บนระบบด้วยคำสั่ง **efsenable** (ตัวอย่าง >efsenable –a) เมื่อได้รับพร้อมต์ป้อนรหัสผ่าน จำเป็นต้องใช้รหัสผ่าน root ที่เก็บคีย์ผู้ใช้ถูกสร้างโดยอัตโนมัติ จากนั้น ผู้ใช้ล็อกอิน หรือล็อกอินซ้ำ หลังจากคำสั่ง **efsenable** ได้ถูกรันเมื่อ **efsenable –a** รัน บนระบบ ระบบจะมี EFS เปิดใช้งานและคำสั่ง **efsenable** ไม่จำเป็นต้องรันอีกครั้ง
3. สร้างระบบไฟล์ที่เปิดใช้ EFS ด้วยอ็อพชัน **–a efs=yes** ตัวอย่าง `crfs -v jfs2 -m /foo –A yes -a efs=yes -g rootvg -a size=20000`

4. หลังการเมาต์ระบบไฟล์ เปิดการสืบทอดวิธีการเข้ารหัสลับ บนระบบไฟล์ที่เปิดใช้งาน EFS ทำได้โดยใช้คำสั่ง **efsmgr** ในการดำเนินการตัวอย่างก่อนหน้านี้ต่อโดยที่ระบบไฟล์ **/foo** ถูกสร้างขึ้น ให้รันคำสั่งต่อไปนี้ : `efsmgr -s -E /foo` คำสั่งนี้อนุญาต ให้ทุกไฟล์ที่สร้างและใช้ในระบบไฟล์นี้เป็นไฟล์ที่เข้ารหัส

ตั้งแต่จุดนี้เป็นต้นไป เมื่อผู้ใช้หรือกระบวนการที่มีที่เก็บคีย์แบบเปิด สร้างไฟล์บนระบบไฟล์นี้ ไฟล์จะถูกเข้ารหัส เมื่อผู้ใช้หรือไฟล์ อ่านไฟล์ ไฟล์จะถูกถอดรหัสโดยอัตโนมัติสำหรับผู้ใช้ที่ ได้รับอนุญาตให้เข้าถึงไฟล์

ดูที่หัวข้อต่อไปนี้เพื่อดูข้อมูลเพิ่มเติม:

- คำสั่ง **chfs, chgroup, chuser, cp, efsenable, efskeymgr, efsmgr, lsuser, ls, mkgroup, mkuser,** และ **mv**
- ไฟล์ `/etc/security/group` และ `/etc/security/user`

การเข้าถึงแบบรีโมตไปยังที่เก็บคีย์ Encrypted File System

ในสภาวะแวดล้อมเอ็นเตอร์ไพรซ์ คุณสามารถรวมศูนย์ที่เก็บคีย์ Encrypted File System (EFS) ของคุณ เมื่อคุณเก็บฐานข้อมูลที่ควบคุม ที่เก็บคีย์บนแต่ละระบบโดยอิสระ อาจทำให้ยากต่อการ จัดการที่เก็บคีย์ ที่เก็บคีย์ AIX Centralized EFS อนุญาตให้คุณเก็บฐานข้อมูลที่เก็บคีย์ผู้ใช้และกลุ่ม ใน Lightweight Directory Access Protocol (LDAP) เพื่อให้คุณสามารถจัดการ ที่เก็บคีย์ EFS แบบรวมศูนย์

หลักการที่เกี่ยวข้อง:

"Lightweight Directory Access Protocol" ในหน้า 165
Lightweight Directory Access Protocol (LDAP) กำหนดวิธีมาตรฐาน สำหรับการเข้าถึงและการอัพเดตข้อมูลในไดเร็กทอรี (ฐานข้อมูล) แบบโลคัลหรือแบบรีโมตอย่างใดอย่างหนึ่งในโมเดลไคลเอ็นต์–เซิร์ฟเวอร์

ภาพรวมการเข้าถึงที่เก็บคีย์ Encrypted File System แบบรีโมต:

เรียนรู้เกี่ยวกับฐานข้อมูล Encrypted File System (EFS) การเปิดใช้งาน LDAP สำหรับคำสั่ง EFS และการเข้าถึงที่เก็บคีย์เฉพาะ

คุณสามารถเก็บฐานข้อมูลที่เก็บคีย์ AIX EFS ทั้งหมดใน LDAP, ซึ่งประกอบด้วยฐานข้อมูล EFS ต่อไปนี้:

- ที่เก็บคีย์ผู้ใช้
- ที่เก็บคีย์กลุ่ม
- ที่เก็บคีย์ผู้ดูแลล
- คุกกี้

ระบบปฏิบัติการ AIX จัดเตรียมยูทิลิตีเพื่อช่วยคุณดำเนินการกับงานการจัดการ ต่อไปนี้:

- เอ็กซ์พอร์ตข้อมูลที่เก็บคีย์โลคัลไปยังเซิร์ฟเวอร์ LDAP
- ตั้งค่าไคลเอ็นต์เพื่อใช้ข้อมูลที่เก็บคีย์ EFS ใน LDAP
- ควบคุมการเข้าถึงข้อมูลที่เก็บคีย์ EFS
- จัดการข้อมูล LDAP จากระบบไคลเอ็นต์

คำสั่งการจัดการฐานข้อมูลที่เก็บคีย์ EFS ทั้งหมดถูกเปิดใช้งาน เพื่อใช้ฐานข้อมูลที่เก็บคีย์ LDAP ถ้าลำดับการค้นหาทั้งระบบไม่ถูกระบุในไฟล์ `/etc/nscontrol.conf` การดำเนินการกับที่เก็บคีย์จะเป็นอิสระตามแอ็ตทริบิวต์ efs_keystore_access ผู้ใช้และกลุ่ม ถ้าคุณตั้งค่า efs_keystore_access เป็น ldap คำสั่ง EFS จะดำเนินการที่เก็บคีย์บนที่เก็บคีย์ LDAP

ตารางต่อไปนี้อธิบายการเปลี่ยนแปลงคำสั่ง EFS สำหรับ LDAP

ตารางที่ 11. การเปิดใช้คำสั่ง EFS สำหรับ LDAP

คำสั่ง	**ข้อมูล LDAP**
คำสั่ง EFS ใดๆ	เมื่อคุณตั้งค่าแอ็ตทริบิวต์ efs_keystore_access เป็น ldap คุณไม่จำเป็นต้องใช้อ็อพชันพิเศษ -L *domain* กับ คำสั่งใดๆเพื่อดำเนินการที่เก็บคีย์บน LDAP
efskeymgr	รวมอ็อพชัน -L load_module เพื่อให้คุณ สามารถดำเนินการที่เก็บคีย์บน LDAP อย่างชัดเจน
efsenable	รวมอ็อพชัน -d Basedn เพื่อให้คุณสามารถ ดำเนินการตั้งค่าเริ่มต้นบน LDAP เพื่อช่วยในการใช้งานที่เก็บคีย์ EFS การตั้งค่าเริ่มต้นประกอบด้วย การเพิ่ม distinguished names (DNs) ฐานสำหรับ ที่เก็บคีย์ EFS และการสร้างโครงสร้างไดเร็กทอรีโลคัล (/var/efs/)
efskstoldif	สร้างข้อมูลที่เก็บคีย์ EFS สำหรับ LDAP จาก ฐานข้อมูลต่อไปนี้บนระบบโลคัล: • /var/efs/users/*username*/keystore • /var/efs/groups/*groupname*/keystore • /var/efs/efs_admin/keystore • คุกกี้สำหรับที่เก็บคีย์ทั้งหมด ถ้ามีอยู่

รายการที่เก็บคีย์ทั้งหมดต้องเป็นค่าเฉพาะ รายการที่เก็บคีย์แต่ละรายการ ที่สอดคล้องโดยตรงกับ DN ของรายการที่มีชื่อผู้ใช้และกลุ่ม ระบบเคียวรี ID ผู้ใช้(uidNumber) ID กลุ่ม (gidNumber) และ DNs เคียวรีสำเร็จเมื่อชื่อผู้ใช้และ กลุ่มตรงกับ DNs ที่สอดคล้องกัน ก่อนที่คุณจะสร้างหรือโอนย้าย รายการที่เก็บคีย์ EFS บน LDAP ทำให้แน่ใจว่าชื่อผู้ใช้และกลุ่ม และ IDs บนระบบไม่มีค่าซ้ำ

งานที่เกี่ยวข้อง:

"การเอ็กซ์พอร์ตข้อมูลที่เก็บคีย์ Encrypted File System ไปยัง LDAP"
คุณต้องกระจายเซิร์ฟเวอร์ LDAP ที่มีข้อมูลที่เก็บคีย์ ไปใช้ LDAP เป็นที่เก็บแบบรวมศูนย์สำหรับที่เก็บคีย์ Encrypted File System (EFS)

"การตั้งค่าไคลเอ็นต์ LDAP สำหรับที่เก็บคีย์ Encrypted File System" ในหน้า 194
ในการใช้ข้อมูลที่เก็บคีย์ Encrypted File System (EFS) ที่ เก็บใน LDAP คุณต้องตั้งค่าระบบเป็นไคลเอ็นต์ LDAP

การเอ็กซ์พอร์ตข้อมูลที่เก็บคีย์ Encrypted File System ไปยัง LDAP:

คุณต้องกระจายเซิร์ฟเวอร์ LDAP ที่มีข้อมูลที่เก็บคีย์ ไปใช้ LDAP เป็นที่เก็บแบบรวมศูนย์สำหรับที่เก็บคีย์ Encrypted File System (EFS)

ก่อนคุณสร้างหรือโอนย้ายรายการที่เก็บคีย์บน LDAP ขอให้มั่นใจว่าชื่อผู้ใช้และกลุ่มบนระบบเป็นค่าเฉพาะ

ในการกระจายเซิร์ฟเวอร์ EFS ที่มีข้อมูลที่เก็บคีย์ของผม ให้ดำเนิน ขั้นตอนต่อไปนี้:

1. ติดตั้ง schema ของที่เก็บ EFS สำหรับ LDAP บนเซิร์ฟเวอร์ LDAP เอง:
   a. ดึงสกีมาที่เก็บคีย์ EFS สำหรับ LDAP จากไฟล์ /etc/security/ldap/sec.ldif บนระบบ AIX
   b. รันคำสั่ง **ldapmodify** เพื่ออัพเดต schema ของเซิร์ฟเวอร์ LDAP ที่มี schema ที่เก็บคีย์ EFS สำหรับ LDAP

2. รันคำสั่ง **efskstoldif** เพื่ออ่าน ข้อมูลในไฟล์ที่เก็บคีย์ EFS โลคัลและเอาต์พุตข้อมูลใน รูปแบบที่เหมาะสำหรับ LDAP ใน การคงการเข้าถึง ที่เก็บคีย์เฉพาะ ให้พิจารณาวางข้อมูลที่เก็บคีย์ EFS ที่อยู่ภายใต้ distinguished name (DN) พาเรนต์เดียวกันคือข้อมูลผู้ใช้ และข้อมูลกลุ่ม
3. บันทึกข้อมูลลงในไฟล์
4. รันคำสั่ง **ldapadd -b** เพื่อกระจายเซิร์ฟเวอร์ LDAP ที่มีข้อมูลที่เก็บคีย์

หลักการที่เกี่ยวข้อง:

"ภาพรวมการเข้าถึงที่เก็บคีย์ Encrypted File System แบบรีโมต" ในหน้า 192
เรียนรู้เกี่ยวกับฐานข้อมูล Encrypted File System (EFS) การเปิดใช้งาน LDAP สำหรับคำสั่ง EFS และการเข้าถึงที่เก็บคีย์เฉพาะ

การตั้งค่าไคลเอ็นต์ LDAP สำหรับที่เก็บคีย์ Encrypted File System:

ในการใช้ข้อมูลที่เก็บคีย์ Encrypted File System (EFS) ที่ เก็บใน LDAP คุณต้องตั้งค่าระบบเป็นไคลเอ็นต์ LDAP

ในการตั้งค่าไคลเอ็นต์ LDAP สำหรับที่เก็บคีย์ EFS ดำเนินขั้นตอน ต่อไปนี้:

1. รันคำสั่ง **/usr/sbin/mksecldap** เพื่อตั้งค่าระบบเป็นไคลเอ็นต์ LDAP คำสั่ง **mksecldap** ค้นหาเซิร์ฟเวอร์ LDAP ที่ระบุแบบไดนามิกเพื่อพิจารณาตำแหน่ง ของข้อมูลที่เก็บคีย์ EFS จากนั้น บันทึกผลลัพธ์ไปยังไฟล์ `/etc/security/ldap/ldap.cfg` คำสั่ง **mksecldap** พิจารณาตำแหน่งสำหรับ ข้อมูลที่เก็บคีย์ผู้ใช้ กลุ่ม ผู้ดูแล และ efscookies
2. ดำเนินขั้นตอนใดขั้นตอนหนึ่งต่อไปนี้เพื่อเปิดใช้งาน LDAP เป็น โดเมนการค้นหาสำหรับข้อมูลที่เก็บคีย์ EFS:
   - ตั้งค่าแอ็ตทริบิวต์ `efs_keystore_access` ผู้ใช้และกลุ่ม ไปยัง `file` หรือ `ldap`
   - กำหนดลำดับการค้นหาสำหรับที่เก็บคีย์ที่ระดับระบบ โดยใช้ไฟล์ `/etc/nscontrol.conf` ตาราง ต่อไปนี้แสดงตัวอย่าง

ตารางที่ 12. ตัวอย่างการตั้งค่าสำหรับไฟล์ /etc/nscontrol.conf

แอ็ตทริบิวต์	คำอธิบาย	ลำดับการค้นหา (secorder)
efsusrkeystore	ลำดับการค้นหานี้เป็นค่าทั่วไปสำหรับผู้ใช้ทั้งหมด	LDAP, ไฟล์
efsgrpkeystore	ลำดับการค้นหานี้เป็นค่าทั่วไปสำหรับกลุ่มทั้งหมด	ไฟล์, LDAP
efsadmkeystore	ลำดับการค้นหานี้ค้นหาที่เก็บคีย์ผู้ดูแล สำหรับที่เก็บคีย์ปลายทางใดๆ	LDAP, ไฟล์

ข้อควรสนใจ: การตั้งค่าที่กำหนดในไฟล์ `/etc/nscontrol.conf` แทนที่ชุดค่าใดๆ แก่แอ็ตทริบิวต์ `efs_keystore_access` ผู้ใช้และกลุ่ม เป็นจริงเช่นเดียวกันสำหรับแอ็ตทริบิวต์ `efs_adminks_access` ผู้ใช้

หลังจากคุณตั้งค่าระบบเป็นไคลเอ็นต์ LDAP และเปิดใช้งาน LDAP เป็นโดเมนการค้นหาสำหรับข้อมูลที่เก็บคีย์ EFS, daemon ไคลเอ็นต์ `/usr/sbin/secldapclntd` เรียกค้นข้อมูลที่เก็บคีย์ EFS จากการตั้งค่า เมื่อใดก็ตามที่คุณดำเนินการที่เก็บคีย์ LDAP

หลักการที่เกี่ยวข้อง:

"ภาพรวมการเข้าถึงที่เก็บคีย์ Encrypted File System แบบรีโมต" ในหน้า 192
เรียนรู้เกี่ยวกับฐานข้อมูล Encrypted File System (EFS) การเปิดใช้งาน LDAP สำหรับคำสั่ง EFS และการเข้าถึงที่เก็บคีย์เฉพาะ

Public Key Cryptography Standards #11

ระบบย่อย Public Key Cryptography Standards #11 (PKCS #11) จัดให้มีแอ็พพลิเคชันที่มีวิธีสำหรับการเข้าถึงอุปกรณ์ฮาร์ดแวร์ (โทเค็น) ไม่ว่าจะเป็นอุปกรณ์ประเภทใด

เนื้อหาในส่วนนี้เป็นไปตามมาตรฐาน PKCS #11 เวอร์ชัน 2.20

ระบบย่อย PKCS #11 ใช้คอมโพเนนต์ต่อไปนี้:

- อ็อบเจ็กต์ที่แบ่งใช้ API (`/usr/lib/pkcs11/ibm_pks11.so`) ถูกจัดให้มีเป็นอินเตอร์เฟสทั่วไปแก่ไดรเวอร์อุปกรณ์ที่รองรับ มาตรฐาน PKCS #11 การออกแบบที่แบ่งเป็นชั้นนี้เปิดใช้งานอุปกรณ์ PKCS #11 ใหม่ เมื่อพร้อมใช้งานโดยไม่ต้องคอมไพล์แอ็พพลิเคชันที่มีอยู่ใหม่
- ไดรเวอร์อุปกรณ์ PKCS #11 ที่จัดให้มีความสามารถ ในแอ็พพลิเคชันที่คล้ายกับความสามารถที่มีให้แก่คอมโพเนนต์เคอร์เนลอื่นๆ เช่น Encrypted File System (EFS) หรือ IP Security (IPSec)
- เมื่อแพล็ตฟอร์มสนับสนุนโปรแกรมอำนวยความสะดวกตัวประมวลผลร่วม การเข้ารหัสลับ ไดรเวอร์อุปกรณ์ PKCS #11 ใช้การเพิ่มความเร็วด้วยฮาร์ดแวร์ ที่มีอยู่กับการดำเนินการ Advanced Encryption Standard (AES), Secure Hash Algorithm (SHA) และ hash message authentication code (HMAC) เพื่อประสิทธิภาพการทำงานที่ดีขึ้น คุณสามารถเปิดใช้งานความสัมพันธ์หน่วยความจำ เครือข่าย

ข้อมูลที่เกี่ยวข้อง:

การสนับสนุนความสัมพันธ์หน่วยความจำ AIX

IBM 4758 Model 2 Cryptographic Coprocessor

IBM 4758 Model 2 Cryptographic Coprocessor จัดให้มีสภาวะแวดล้อมการคำนวณที่ปลอดภัย

ก่อนพยายามตั้งค่าระบบย่อย PKCS #11 ให้ตรวจสอบว่า อะแด็ปเตอร์ได้รับการตั้งค่าอย่างเหมาะสมด้วยไมโครโค้ดที่สนับสนุน

IBM 4960 Cryptographic Accelerator

IBM 4960 Cryptographic Accelerator จัดให้มีวิธีการออฟโหลดทรานแซกชันที่มี การเข้ารหัสลับ ก่อนพยายามตั้งค่าระบบย่อย PKCS #11 ให้ตรวจสอบว่า อะแด็ปเตอร์ได้รับการตั้งค่าอย่างเหมาะสม

การตรวจสอบ IBM 4758 Model 2 Cryptographic Coprocessor เพื่อใช้กับระบบย่อย Public Key Cryptography Standards #11:

ระบบย่อย PKCS #11 ถูกออกแบบเพื่อตรวจสอบอย่างอัตโนมัติถึงความสามารถของ อะแด็ปเตอร์ในการสนับสนุนการเรียก PKCS #11 ระหว่างการติดตั้ง และขณะบูตใหม่ สำหรับเหตุผลนี้, IBM 4758 Model 2 Cryptographic Coprocessor ที่ไม่ได้ถูกกำหนดค่าอย่างถูกต้องจะไม่สามารถ ถูกเข้าถึงได้จากอินเตอร์เฟส PKCS #11 และการเรียกที่ส่งไปที่อะแด็ปเตอร์ จะล้มเหลว

เมื่อต้องการตรวจสอบว่าอะแด็ปเตอร์ของคุณถูกติดตั้งอย่างถูกต้องทำตาม ขั้นตอนต่อไปนี้ให้สมบูรณ์:

1. ตรวจสอบว่าซอฟต์แวร์สำหรับอะแด็ปเตอร์ถูกติดตั้งอย่างถูกต้อง โดยพิมพ์คำสั่งดังต่อไปนี้:

   ```
   lsdev -Cc adapter | grep crypt
   ```

   ถ้า IBM 4758 Model 2 Cryptographic Coprocessor ไม่ถูกรวมอยู่ใน รายการผลลัพธ์ให้ตรวจสอบว่าการ์ดถูกติดตั้งถูกต้อง และซอฟต์แวร์ที่สนับสนุนถูกติดตั้งอย่างถูกต้อง

2. ตรวจสอบว่าเฟิร์มแวร์ที่ถูกต้องได้ถูกโหลดลงใน การ์ดโดยพิมพ์ดังต่อไปนี้:

   ```
   csufclu /tmp/l ST device_number_minor
   ```

   ตรวจสอบว่า Segment 3 Image มีแอ็พพลิเคชัน PKCS #11 โหลดอยู่ ถ้ายังไม่ได้ถูกโหลดให้อ้างอิงเอกสารอะแด็ปเตอร์เพื่อรับ microcode ล่าสุดและคำแนะการติดตั้ง

   หมายเหตุ: ถ้ายูทิลิตีนี้ไม่พร้อมใช้งาน ซอฟต์แวร์สนับสนุน ไม่ได้ถูกติดตั้ง

การตรวจสอบ IBM 4960 Model 2 Cryptographic Accelerator เพื่อใช้กับระบบย่อย Public Key Cryptography Standards #11:

ระบบย่อย PKCS #11 ถูกออกแบบเพื่อตรวจสอบอย่างอัตโนมัติถึงความสามารถของ อะแด็ปเตอร์ในการสนับสนุนการเรียก PKCS #11 ระหว่างการติดตั้ง และขณะบูตใหม่ สำหรับเหตุผลนี้, IBM 4960 Cryptographic Accelerator ที่ไม่ได้ถูกกำหนดค่าอย่างถูกต้องจะไม่สามารถ ถูกเข้าถึงได้จากอินเตอร์เฟส PKCS #11 และการเรียกที่ส่งไปที่อะแด็ปเตอร์ จะล้มเหลว

เพื่อประกันว่าซอฟต์แวร์สำหรับอะแด็ปเตอร์ถูกติดตั้ง อย่างถูกต้อง ให้พิมพ์คำสั่งดังต่อไปนี้:

```
lsdev -Cc adapter | grep ica
```

ถ้า IBM 4960 Cryptographic Accelerator ไม่ถูกรวมอยู่ในรายการผลลัพธ์ ให้ตรวจสอบว่าการ์ดถูกติดตั้งถูกต้อง และไดรเวอร์อุปกรณ์ที่สนับสนุนถูกติดตั้งอย่างถูกต้อง

การตั้งค่าระบบย่อย Public Key Cryptography Standards #11

ระบบย่อย PKCS #11 ตรวจหาอุปกรณ์ที่สนับสนุน PKCS #11 โดยอัตโนมัติ อย่างไรก็ตาม สำหรับบางแอ็พพลิเคชันที่ใช้อุปกรณ์เหล่านี้ อาจจำเป็นต้องตั้งค่าเริ่มต้น

งานเหล่านี้สามารถดำเนินการผ่าน API (โดยการเขียน แอ็พพลิเคชัน PKCS #11) หรือโดยการใช้อินเตอร์เฟส SMIT อ็อพชัน PKCS #11 SMIT ถูกเข้าถึงผ่าน **Manage the PKCS11 subsystem** จาก เมนู SMIT หลัก หรือโดยการใช้พาธด่วน **smit pkcs11**

การเตรียมข้อมูลเบื้องต้นโทเค็น:

แต่ละอะแด็ปเตอร์หรือโทเค็น PKCS #11 ต้องถูกเตรียมข้อมูลเบื้องต้นก่อน ถูกนำมาใช้

ขั้นตอนการเตรียมข้อมูลเกี่ยวข้องกับการตั้งค่าเลเบลเฉพาะ ให้กับโทเค็น เลเบลนี้อนุญาตให้แอ็พพลิเคชันระบุโทเค็นเฉพาะ ดังนั้นเลเบลไม่ควรถูกทำซ้ำ อย่างไรก็ตาม API ไม่ได้ตรวจสอบ ว่าเลเบลไม่ได้ถูกนำกลับมาใช้ การเตรียมข้อมูลนี้ทำได้ผ่านแอ็พพลิเคชัน PKCS #11 หรือโดยผู้ดูแลระบบโดยใช้ SMIT ถ้าโทเค็นของคุณมี Security Officer PIN, ค่าดีฟอลต์ถูกเซ็ตเป็น 87654321 เพื่อประกัน ความปลอดภัยของระบบย่อย PKCS #11 ค่านี้ควรถูกเปลี่ยนหลังการเตรียมข้อมูล

เมื่อต้องการ เตรียมข้อมูลเบื้องต้นโทเค็น:

1. เข้าสู่จอภาพการจัดการโทเค็นโดยพิมพ์ `smit pkcs11`
2. เลือก **Initialize a Token**
3. เลือกอะแด็ปเตอร์ PKCS #11 จากรายการของสนับสนุนที่สนับสนุน
4. ยืนยันการเลือกของคุณโดยกด Enter

   หมายเหตุ: นี่จะ เป็นการลบข้อมูลทั้งหมดบนโทเค็น

5. ป้อน Security Officer PIN (SO PIN) และเลเบลโทเค็นเฉพาะ

ถ้า PIN ที่ถูกต้องถูกป้อน อะแด็ปเตอร์จะถูกเตรียมข้อมูลเบื้องต้น หรือกำหนดข้อมูลใหม่ หลังจากคำสั่งรันเสร็จสิ้น

การตั้งค่า PIN เจ้าหน้าที่รักษาความปลอดภัย:

ทำตามขั้นตอนเหล่านี้เพื่อเปลี่ยน SO PIN จากค่าดีฟอลต์

ในการเปลี่ยน PIN จากค่าดีฟอลต์:

1. พิมพ์ `smit pkcs11`
2. เลือก **Set the Security Officer PIN**
3. เลือกอะแด็ปเตอร์ที่เตรียมข้อมูลไว้ซึ่งคุณต้องการตั้งค่า PIN
4. ป้อน PIN ปัจจุบันและ PIN ใหม่
5. ยืนยัน PIN ใหม่

การเตรียมข้อมูล PIN ผู้ใช้:

หลังจากโทเค็นได้ถูกเตรียมข้อมูล อาจจำเป็นต้อง ตั้งค่า PIN ผู้ใช้เพื่ออนุญาตให้แอ็พพลิเคชันเข้าถึงโทเค็นอ็อบเจ็กต์

อ้างอิงเอกสารคู่มือของอุปกรณ์เฉพาะของของคุณเพื่อพิจารณาว่า อุปกรณ์จำเป็นต้องให้ผู้ใช้ล็อกอินก่อนเข้าถึงอ็อบเจ็กต์หรือไม่

เมื่อต้องการเตรียมข้อมูล PIN ผู้ใช้:

1. เข้าสู่หน้าจอการจัดการโทเค็นโดยพิมพ์ `smit pkcs11`
2. เลือก **Initialize the User PIN**
3. เลือกอะแด็ปเตอร์ PKCS #11 จากรายการอะแด็ปเตอร์ที่สนับสนุน
4. ป้อน SO PIN และ User PIN
5. ยืนยัน User PIN
6. เมื่อทำการยืนยันเสร็จ ต้องเปลี่ยน User PIN

การรีเซ็ต PIN ผู้ใช้:

ในการรีเซ็ต PIN ผู้ใช้ คุณสามารถเตรียมข้อมูล PIN ใหม่โดยใช้ SO PIN หรือตั้งค่า PIN ผู้ใช้โดยใช้ PIN ผู้ใช้ที่มีอยู่

เมื่อต้องการรีเซ็ต PIN:

1. เข้าสู่หน้าจอการจัดการโทเค็นโดยพิมพ์ `smit pkcs11`
2. เลือก **Set the User PIN**
3. เลือกอะแด็ปเตอร์ที่เตรียมข้อมูลเบื้องต้นที่คุณต้องการตั้งค่า PIN ผู้ใช้
4. ป้อน PIN ผู้ใช้ปัจจุบันและ PIN ใหม่
5. ตรวจสอบ PIN ผู้ใช้ใหม่

การใช้งาน Public Key Cryptography Standards #11

สำหรับแอ็พพลิเคชันที่จะใช้ระบบย่อย PKCS #11 daemon ตัวจัดการสล็อตของระบบย่อยต้องกำลังทำงานและแอ็พพลิเคชันต้องโหลดในอ็อบเจ็กต์ ที่แบ่งใช้ของ APIs

ตัวจัดการสล็อตโดยปกติจะเริ่มทำงานในตอนบูตระบบโดย **inittab** เรียกใช้ สคริปต์ /etc/rc.pkcs11 สคริปต์นี้ตรวจสอบอะแด็ปเตอร์ในระบบก่อนที่จะเริ่มทำงาน daemon ตัวจัดการสล็อต เป็นผลให้ daemon ตัวจัดการสล็อตไม่พร้อมใช้งานก่อนที่ผู้ใช้จะล็อกออนเข้าสู่ระบบ หลังจาก daemon เริ่มทำงาน ระบบย่อยจะรวบรวมการเปลี่ยนแปลงใดๆ ตามจำนวนและ ประเภทของอะแด็ปเตอร์ที่สนับสนุนโดยไม่มีการแทรกแซงจากผู้ดูแลระบบ

API สามารถถูกโหลดโดยการลิงก์ในอ็อบเจ็กต์ตอนรันไทม์ หรือโดย การใช้การกำหนดค่าสัญลักษณ์ที่ถูกเลื่อนออกไป ตัวอย่าง แอ็พพลิเคชันสามารถรับรายการ ฟังก์ชัน PKCS #11 ได้ในลักษณะต่อไปนี้:

```
d CK_RV (*pf_init)();
void *d;
CK_FUNCTION_LIST *functs;

d = dlopen(e, RTLD_NOW);
if ( d == NULL ) {
   return FALSE;
}

pfoo = (CK_RV (*)())dlsym(d, "C_GetFunctionList");
if (pfoo == NULL) {
   return FALSE;
}

rc = pf_init(&functs);
```

เครื่องมือมาตรฐานการเข้ารหัสพับลิกคีย์ #11

สองเครื่องมือพร้อมใช้งานสำหรับการจัดการระบบการเข้ารหัส ภายในระบบปฏิบัติการ AIX : เครื่องมือ PKCS #11 Key Management, และเครื่องมือ PKCS #11 Administration คุณสามารถเข้าถึงเครื่องมือเหล่านี้ได้โดยใช้ GUI ที่อิง Curse หรืออินเตอร์เฟสบรรทัดคำสั่ง

หมายเหตุ: ความสามารถในการเข้าถึงสำหรับเครื่องมือกรอบงานการเข้ารหัส AIX ต้องการใช้ความสามารถ ในการประมวลผลแบบแบตช์ สำหรับข้อมูลโดยละเอียด เกี่ยวกับการใช้คุณลักษณะการประมวลผลแบบแบตช์สำหรับการเข้าถึง โปรดดู "การประมวลผลแบบแบตช์" ในหน้า 200

เครื่องมือ PKCS #11 Key Management เป็นเครื่องมือกลางสำหรับการจัดการคีย์, ใบรับรอง, และข้อมูล PKCS #11 บนระบบปฏิบัติการ AIX อ็อบเจ็กต์ ที่ถูกจัดการโดยเครื่องมือนี้ถูกเก็บไว้ภายในผู้ให้บริการ PKCS #11 ที่สนับสนุน, เช่น ตระกูล IBM ของอะแด็ปเตอร์การเข้ารหัส (ตัวอย่างเช่น, IBM 4758, 4960, และ 4764), หรือ AIX Cryptographic Framework คุณสามารถ ดำเนินการต่างๆ ได้โดยใช้เครื่องมือการจัดการคีย์ PKCS #11 การดำเนินการนี้รวมไปถึงการสร้างคำร้องขอให้ลงนามรับรอง (CSR) PKCS #10 หรือสร้างใบรับรองการลงนามด้วยตนเอง นอกจากนี้ คุณสามารถใช้ เครื่องมือนี้เพื่อค้นหา ดู ลบ อิมพอร์ต เอ็กซ์พอร์ต และสำรองข้อมูลอ็อบเจ็กต์ PKCS #11 เช่นเดียวกับถ่ายโอนข้อมูลอ็อบเจ็กต์ PKCS #11 ระหว่างโทเค็น PKCS #11 คุณสามารถเริ่มต้นเวอร์ชัน GUI ของเครื่องมือได้ โดยรันคำสั่ง **p11km** เครื่องมือโหลดโทเค็น PKCS #11 ที่พร้อมใช้งานทั้งหมด คุณสามารถดูรายละเอียดเกี่ยวกับโทเค็นเหล่านี้ได้ โดใช้ปุ่มลูกศรเลื่อนขึ้น และเลื่อนลงในรายการโทเค็น หากต้องการเลือกโทเค็น ใช้ปุ่มลูกศรเพื่อไฮไลต์โทเค็น และกดปุ่ม Enter คุณสามารถเริ่มต้นเวอร์ชันบรรทัดคำสั่งของเครื่องมือได้ โดยรันคำสั่งต่อไปนี้:

```
p11km -b <ไฟล์แบตช์>
```

เครื่องมือ PKCS #11 Administration เป็นเครื่องมือกลางสำหรับการจัดการกับกรอบงานการเข้ารหัส AIX PKCS #11 เครื่องมือนี้อนุญาตให้ผู้ดูแลระบบหรือพนักงานรักษาความปลอดภัย จัดการกับโทเค็นที่ควบคุมโดย AIX Cryptographic Framework คุณสามารถใช้เครื่องมือนี้เพื่อเริ่มต้น, สร้าง, และทำลายโทเค็น PKCS #11, จัดการสล็อต, รีเซ็ตรหัสผ่านผู้ใช้, ยืนยันการลบอ็อบเจ็กต์, ระบุความเชื่อถือได้ของอ็อบเจ็กต์, และดำเนินการปรับ AIX Cryptographic Framework สำหรับประสิทธิภาพการทำงานและการดูแลระบบทั่วไป คุณสามารถเริ่มต้นเวอร์ชัน GUI ของเครื่องมือได้โดยรันคำสั่ง **p11admin** เครื่องมือโหลดโทเค็น PKCS #11 ที่พร้อมใช้งานทั้งหมด คุณสามารถดูรายละเอียดเกี่ยวกับโทเค็นเหล่านี้ได้โดใช้ปุ่มลูกศรเลื่อนขึ้น และเลื่อนลงในรายการโทเค็น หากต้องการเลือกโทเค็น ใช้ปุ่มลูกศรเพื่อไฮไลต์โทเค็น และกดปุ่ม Enter คุณสามารถเริ่มต้นเวอร์ชันบรรทัดคำสั่งของเครื่องมือได้โดยรันคำสั่งต่อไปนี้:

```
p11admin -b <ไฟล์แบตช์>
```

โปรไฟล์คำสั่ง:

เครื่องมือ AIX Cryptographic Framework ใช้ไลบรารี OpenSSL เพื่อวิเคราะห์คำไฟล์คอนฟิกูเรชัน ที่ถูกใช้เพื่อสร้างโปรไฟล์แบบกำหนดเอง คุณสามารถใช้โปรไฟล์เหล่านี้เพื่อตั้งค่าแอ็ตทริบิวต์ต่างๆ ของเครื่องมือ เช่น สีของ GUI สำหรับคำสั่ง **p11km** และคำสั่ง **p11admin**

โดยการใช้รูปแบบไฟล์ที่ระบุไว้ใน “การประมวลผลแบบแบตช์” ในหน้า 200, คุณสามารถสร้าง และแก้ไขไฟล์ของโปรไฟล์ต่อไปนี้เพื่อกำหนดลักษณะเฉพาะของ GUI

หมายเหตุ: หลังจากที่คุณสร้างไฟล์ของโปรไฟล์แล้ว ให้ตั้งชื่อไฟล์นั้น และจัดเก็บไฟล์ไว้ในไดเร็กทอรีหลักของคุณดังนี้:

```
$HOME/.p11km

$HOME/.p11admin
```

แอ็ตทริบิวต์สีของ GUI ต่อไปนี้ได้รับการสนับสนุน:

```
action_name = “GUI_COLORS”
gui_fg_color = “<ชื่อสี>” ## Foreground Color
gui_bg_color = “<ชื่อสี>” ## Background Color
gui_vc_color = “<ชื่อสี>” ## View Content Color
```

โดย <*ชื่อสี*> คือหนึ่งในค่าต่อไปนี้:

LIGHT GRAY

WHITE

BLACK

DARK GRAY

RED

LIGHT RED

YELLOW

ORANGE or BROWN

GREEN

LIGHT GREEN

BLUE

LIGHT BLUE

CYAN

LIGHT CYAN

MAGENTA

LIGHT MAGENTA

ตัวอย่าง: p11km profile ($HOME/.p11km)

```
[p11km_cmd]
gui_fg_color = "RED"
gui_bg_color = "BLACK"
gui_vc_color = "WHITE"
```

ตัวอย่าง: p11admin Profile ($HOME/.p11admin)

```
[p11admin_cmd]
gui_fg_color = "BLUE"
gui_bg_color = "LIGHT GRAY"
gui_vc_color = "BLACK"
```

การประมวลผลแบบแบตช์:

คุณสามารถรันคำสั่งประมวลผลแบบแบตช์จากบรรทัดคำสั่ง เพื่อดำเนินการงานเดียวกันที่พร้อมใช้งานในเวอร์ชัน GUI ของเครื่องมือ PKCS #11

รูปแบบคำสั่งสำหรับเครื่องมือจัดการหลัก PKCS #11 (p11km) เป็นดังต่อไปนี้:

```
p11km -b <ไฟล์แบตช์>
```

รูปแบบคำสั่งสำหรับเครื่องมือควบคุมดูแลหลัก PKCS #11 (p11admin) เป็นดังต่อไปนี้:

```
p11admin -b <ไฟล์แบตช์>
```

เนื่องจากเครื่องมือเหล่านี้ใช้ไลบรารี OpenSSL เพื่อวิเคราะห์ไฟล์แบตช์ รูปแบบของไฟล์แบตช์เป็นไปตามรูปแบบไฟล์การกำหนดคอนฟิก OpenSSL ทั่วไป แต่ละส่วนคือคำสั่งที่แยกกัน และการจับคู่ของค่าแอ็ตทริบิวต์ จัดเตรียมข้อมูลที่จำเป็นสำหรับการประมวลผล แต่ละคำสั่งของส่วนคือการประมวลผลแบตช์ตามลำดับจากบนลงล่าง ถ้าคำสั่งแบตช์ใดล้มเหลว ข้อผิดพลาดจะถูกพิมพ์ และยุติการประมวลผลแบตช์นั้น โดยไม่ประมวลคำสั่งของส่วนถัดไป

ข้อมูลต่อไปนี้คือตัวอย่างรูปแบบไฟล์คอนฟิกูเรชันของ OpenSSL

```
[section1]
attribute1 = "value1"
attribute2 = "value2"
...
attributeN = "valueN"
[section2]
attribute1 = "value1"
attribute2 = "value2"
...
attributeN = "valueN"
```

```
...
...
[sectionN]
attribute1 = "value1"
attribute2 = "value2"
...
attributeN = "valueN"
```

หากต้องการให้มั่นใจว่าส่วนคำสั่งของเครื่องมือ PKCS #11 อยู่ร่วมกับ ส่วนไฟล์คอนฟิกูเรชัน OpenSSL ให้ใช้คำเสริมหน้าต่อไปนี้สำหรับส่วน PKCS #11:

p11km tool

p11km_cmd

p11admin tool

p11admin_cmd

แต่ละส่วนของ p11km_cmd หรือ p11admin_cmd ต้องมีเพียงแอ็ตทริบิวต์ action_name เดียว พร้อมค่าเดี่ยวที่บ่งชี้คำสั่งเฉพาะที่เกี่ยวข้องกับส่วน ตัวอย่างที่ง่ายที่สุดคือไฟล์ที่มีส่วนคำสั่งเดียว ที่อธิบายคำสั่งที่ไม่มีพารามิเตอร์เพิ่มเติม ข้อมูลต่อไปนี้คือตัวอย่างของวิธีการใช้เครื่องมือ p11km เพื่อรันคำสั่งแบตช์ที่แสดงรายการโทเค็น PKCS #11 ที่พร้อมใช้งานในระบบ:

```
[p11km_cmd_list_my_tokens]
action_name="LIST_TOKENS"
```

คำสั่งแบตช์แต่ละคำสั่งสนับสนุนแอ็ตทริบิวต์บูลีนทางเลือก:

```
start_gui="<boolean>"
```

ถ้าคุณรันคำสั่งแบตช์ที่มีแอ็ตทริบิวต์บูลีน ที่มีค่าเป็น TRUE การประมวลผลแบบแบตช์จะยุติหลังคำสั่งดังกล่าวเสร็จสิ้น และ GUI เริ่มทำงาน

หมายเหตุ: ถ้าไฟล์แบตช์มีคำสั่งที่รวมแอ็ตทริบิวต์ **start_gui** ทางเลือก ไม่มีคำสั่งแบตช์ที่แสดงในรายการหลังประมวลผล

คำสั่งแบตช์:

คำสั่งแบตช์จัดเตรียมบรรทัดคำสั่งเข้าถึงเครื่องมือ PKCS #11

คำสั่งแบตช์ต่อไปนี้สามารถใช้ได้ในเครื่องมือการจัดการคีย์ PKCS #11 (p11km)

หมายเหตุ: หากต้องการใช้คำสั่งแบตช์ให้ปฏิบัติดังต่อไปนี้:

1. สร้างและแก้ไขไฟล์แบตช์ตามที่อธิบายไว้ใน "การประมวลผลแบบแบตช์" ในหน้า 200
2. สร้างส่วน p11km_cmd ใหม่ที่มีแอ็ตทริบิวต์สำหรับ คำสั่งแบตช์ที่คุณต้องการใช้

รายการโทเค็น PKCS #11 ที่พร้อมใช้งาน

สร้างรายงานและแสดงผลโทเค็นและข้อมูลสล็อตสำหรับ โทเค็น PKCS #11 ที่พร้อมใช้งาน

แอ็ตทริบิวต์ที่จำเป็น

action_name = "LIST_TOKENS"

แอ็ตทริบิวต์ทางเลือก

start_gui = "<*boolean*>"

โดย <*boolean*> เป็นTRUE หรือ FALSE

ตัวอย่าง

```
[p11km_cmd_list_tokens]
action_name = "LIST_TOKENS"
```

รายการกลไกของ PKCS#11 ที่พร้อมใช้งาน

สร้างรายงานและแสดงผลกลไกของ PKCS #11 ที่สนับสนุนโดยโทเค็น PKCS #11 ที่ระบุ (ตรงกับไดรเวอร์ และค่าแอ็ตทริบิวต์ของสล็อตที่เลือกไว้)

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "LIST_MECHANISMS"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
```

โดย <*slot number*> เป็นค่าจำนวนเต็มบวก และ <*driver name*> เป็นหนึ่งในค่าต่อไปนี้:

ค่า	คำอธิบาย
AIX	AIX OS Cryptographic Framework
IBM_4758_4960	IBM 4758/4960 Cryptographic Hardware Adapters
IBM_4764	IBM 4764 Cryptographic Hardware Adapter
อื่นๆ	ถ้าคุณระบุ OTHER คุณต้องระบุแอ็ตทริบิวต์ **p11_driver_path** ด้วย

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

แอ็ตทริบิวต์เสริม

```
p11_driver_path = "<path to PKCS#11 driver>"
```

โดยที่ <*path to PKCS#11 driver*> คือพาธแบบเต็ม UNIX และชื่อไฟล์ ของไลบรารี PKCS #11 ที่ถูกใช้สำหรับคำสั่ง แอ็ตทริบิวต์นี้สามารถระบุได้เฉพาะเมื่อ แอ็ตทริบิวต์ **p11_driver** ถูกตั้งค่าเป็น OTHER

ตัวอย่าง

```
[p11km_cmd_list_4764_slot_0_mechs]
action_name = "LIST_MECHANISMS"
p11_driver = "IBM_4764"
p11_slot = "0"
start_gui = "TRUE"
```

รายการอ็อบเจ็กต์ PKCS #11 ที่พร้อมใช้งาน

สร้างรายงานและแสดงผลอ็อบเจ็กต์ PKCS #11 ที่พร้อมใช้งาน ซึ่งสนับสนุนโดยโทเค็น PKCS #11 (ตรงกับไดรเวอร์ และค่าแอ็ตทริบิวต์ของสล็อตที่ระบุ)

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "LIST_OBJECTS"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
```

แอ็ตทริบิวต์ทางเลือก

```
p11_login = "<boolean>"
p11_label = "<string>"
p11_class = "<PKCS#11 Object Class>"
p11_private = "<boolean>"
p11_trusted = "<boolean>"
p11_sensitive = "<boolean>"
start_gui = "<boolean>"
```

โดย *<PKCS#11 Object Class>* เป็นหนึ่งในค่าต่อไปนี้ตามที่กำหนดไว้ใน ข้อกำหนด PKCS #11 จาก RSA:

```
CKO_DATA
CKO_CERTIFICATE
CKO_PUBLIC_KEY
CKO_PRIVATE_KEY
CKO_SECRET_KEY
CKO_HW_FEATURE
CKO_DOMAIN_PARAMETERS
CKO_MECHANISM
CKO_VENDOR_DEFINED
```

ตัวอย่าง

```
[p11km_cmd_list_private_objs]
action_name = "LIST_OBJECTS"
p11_login = "TRUE"
p11_private = "TRUE"
p11_driver = "AIX"
p11_slot = "5"
```

เปลี่ยนแปลง PIN ของผู้ใช้โทเค็น PKCS #11:

เปลี่ยนแปลงPIN ของผู้ใช้โทเค็น PKCS #11 ที่ใช้งานเมื่อล็อกออน เข้าสู่โทเค็น

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "CHANGE_USER_PIN"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
```

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11km_cmd_change_my_pin]
action_name = "CHANGE_USER_PIN"
p11_slot = "1337"
p11_driver = "IBM_4764"
```

ลบอ็อบเจ็กต์ PKCS #11

ลบอ็อบเจ็กต์ PKCS #11 อ็อบเจ็กต์ถูกลบตามลำดับหมายเลขของอ็อบเจ็กต์ ซึ่งเกิดจากการรันคำสั่ง **LIST_OBJECTS** และใช้เท็มเพลตเดียวกันแอ็ตทริบิวต์ต่อไปนี้:

```
p11_label = "<string>"
p11_class = "<PKCS#11 Object Class>"
p11_private = "<boolean>"
p11_trusted = "<boolean>"
p11_sensitive = "<boolean>"
p11_login = "<boolean>"
```

ข้อควรสนใจ: เนื่องจาก สถานะและความสอดคล้องกันของโทเค็นไม่ถูกรักษาไว้ระหว่างขั้นตอนแบ็ตช์ อ็อบเจ็กต์สามารถถูกลบได้โดยไม่ตั้งใจ ลำดับรายการของอ็อบเจ็กต์เปลี่ยนแปลง เมื่ออ็อบเจ็กต์ถูกเพิ่มหรือลบโดยขั้นตอนอื่นที่รัน ซ้อนในโทเค็นเดียวกันระหว่างเวลาที่อ็อบเจ็กต์แสดงในรายการ กับเวลาที่อ็อบเจ็กต์ถูกลบ

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "DELETE_OBJECTS"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
p11_objects = "<CSV>"
```

โดย <*CSV*> เป็นคำว่า ALL (อ็อบเจ็กต์โทเค็นทั้งหมด) หรือรายการที่แบ่งด้วยจุลภาคของ ค่าจำนวนเต็มบวกที่สอดคล้องกับอ็อบเจ็กต์ตามลำดับหมายเลขที่ปรากฏ โดยใช้แอ็ตทริบิวต์ทางเลือกต่อไปนี้

แอ็ตทริบิวต์ทางเลือก

```
p11_label = "<string>"
p11_class = "<PKCS#11 Object Class>"
p11_private = "<boolean>"
p11_trusted = "<boolean>"
p11_sensitive = "<boolean>"
p11_login = "<boolean>"
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11km_cmd_delete_seven_objects]
action_name = "DELETE_OBJECTS"
p11_slot = "0"
p11_driver = "AIX"
p11_objects = "1,5,10,11,12,27,33"
p11_login = "TRUE"
```

ย้ายอ็อบเจ็กต์ PKCS #11:

ย้ายอ็อบเจ็กต์ PKCS #11 อ็อบเจ็กต์ถูกย้ายตามลำดับหมายเลขของอ็อบเจ็กต์ ซึ่งเกิดจากการรันคำสั่ง **LIST_OBJECTS** และใช้เท็มเพลตเดียวกัน

ข้อควรสนใจ: เนื่องจากสถานะและความสอดคล้องกัน ของอ็อบเจ็กต์ไม่ถูกรักษาไว้ระหว่างขั้นตอนแบ็ตช์ อ็อบเจ็กต์สามารถถูกย้ายได้โดยไม่ตั้งใจ ลำดับรายการของอ็อบเจ็กต์เปลี่ยนแปลง เมื่ออ็อบเจ็กต์ถูกเพิ่มหรือลบโดยขั้นตอนอื่นที่รัน ซ้อนในโทเค็นเดียวกันระหว่างเวลาที่อ็อบเจ็กต์แสดงในรายการ กับเวลาที่อ็อบเจ็กต์ถูกย้าย

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "MOVE_OBJECTS"
##############################################
###### Source Token Identification: ######
p11_driver = "<driver name>"
p11_slot = "<slot number>"
##############################################
```

```
###### Target Token Identification: ######
p11_driver_target = "<driver name>"
p11_slot_target = "<slot number>"
##########################################
###### Objects being moved to target: ######
p11_objects = "<CSV>"
```

แอ็ตทริบิวต์ทางเลือก

```
p11_label = "<string>"
p11_class = "<PKCS#11 Object Class>"
p11_private = "<boolean>"
p11_trusted = "<boolean>"
p11_sensitive = "<boolean>"
p11_login = "<boolean>"
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11km_cmd_move_three_objects]
action_name = "MOVE_OBJECTS"
p11_slot = "0"
p11_slot_target = "1"
p11_driver = "AIX"
p11_driver_target = "AIX"
p11_objects = "15,20,60"
p11_login = "FALSE"
```

คัดลอกอ็อบเจ็กต์ PKCS #11

คัดลอกอ็อบเจ็กต์ PKCS #11 อ็อบเจ็กต์ถูกคัดลอกตามลำดับหมายเลขของอ็อบเจ็กต์ ซึ่งเกิดจากการรันคำสั่ง **LIST_OBJECTS** และใช้เท็มเพลตเดียวกัน

ข้อควรสนใจ: เนื่องจาก สถานะและความสอดคล้องกันของโทเค็นไม่ถูกรักษาไว้ระหว่างขั้นตอนแบ็ตช์ อ็อบเจ็กต์สามารถถูกคัดลอกได้โดยไม่ตั้งใจ ลำดับรายการของอ็อบเจ็กต์เปลี่ยนแปลง เมื่ออ็อบเจ็กต์ถูกเพิ่มหรือลบโดยขั้นตอนอื่นที่รัน ซ้อนในโทเค็นเดียวกันระหว่างเวลาที่อ็อบเจ็กต์แสดงในรายการ กับเวลาที่อ็อบเจ็กต์ถูกคัดลอก

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "COPY_OBJECTS"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
p11_driver_target = "<driver name>"
p11_slot_target = "<slot number>"
p11_objects = "<CSV>"
```

แอ็ตทริบิวต์ทางเลือก

```
p11_label = "<string>"
p11_class = "<PKCS#11 Object Class>"
p11_private = "<boolean>"
p11_trusted = "<boolean>"
p11_sensitive = "<boolean>"
p11_login = "<boolean>"
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11km_cmd_copy_one_private_object]
action_name = "COPY_OBJECTS"
p11_slot = "0"
p11_slot_target = "1"
p11_driver = "AIX"
p11_driver_target = "AIX"
p11_objects = "3"
p11_login = "TRUE" ## REQUIRED FOR PRIVATE OBJECT MGT.
```

ส่งออกและสำรองข้อมูลอ็อบเจ็กต์ PKCS #11 ไปยังไฟล์

ส่งออกและสำรองข้อมูลอ็อบเจ็กต์ PKCS #11 อ็อบเจ็กต์ถูกส่งออกและสำรองข้อมูลตามลำดับหมายเลขของอ็อบเจ็กต์ ซึ่งเกิดจากการรันคำสั่ง **LIST_OBJECTS** และใช้เท็มเพลตเดียวกัน

ข้อควรสนใจ: เนื่องจากสถานะและความสอดคล้องกัน ของอ็อบเจ็กต์ไม่ถูกรักษาไว้ระหว่างขั้นตอนแบ็ตช์ อ็อบเจ็กต์สามารถถูกส่งออกได้โดยไม่ตั้งใจ ลำดับรายการของอ็อบเจ็กต์เปลี่ยนแปลง เมื่ออ็อบเจ็กต์ถูกเพิ่มหรือลบโดยขั้นตอนอื่นที่รัน ซ้อนในโทเค็นเดียวกันระหว่างเวลาที่อ็อบเจ็กต์แสดงในรายการ กับเวลาที่อ็อบเจ็กต์ถูกส่งออก

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "EXPORT_OBJECTS"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
p11_object_file = "<file name>"
p11_objects = "<CSV>"
```

แอ็ตทริบิวต์ทางเลือก

```
p11_label = "<string>"
p11_class = "<PKCS#11 Object Class>"
p11_private = "<boolean>"
p11_trusted = "<boolean>"
p11_sensitive = "<boolean>"
p11_login = "<boolean>"
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11km_cmd_backup_objects]
action_name = "EXPORT_OBJECTS"
p11_slot = "0"
p11_driver = "AIX"
p11_objects = "ALL"
p11_login = "TRUE"
p11_object_file = "/home/user1/p11km.backup"
```

นำเข้าอ็อบเจ็กต์ PKCS #11 จากไฟล์

นำเข้าอ็อบเจ็กต์ PKCS #11 ที่สร้างจากไฟล์ส่งออกของ PKCS #11

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "IMPORT_OBJECTS"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
p11_object_file = "<file name>"
```

แอ็ตทริบิวต์ทางเลือก

```
p11_login = "<boolean>" # REQUIRED TO IMPORT ANY PRIVATE OBJECTS
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11km_cmd_import_my_backed_up_objects]
action_name = "IMPORT_OBJECTS"
p11_slot = "0"
p11_driver = "AIX"
p11_login = "TRUE"
p11_object_file = "/home/user1/p11km.backup"
```

สร้างใบรับรองที่ลงนามเอง

สร้างใบรับรอง X.509 ที่ลงนามเอง และอ็อบเจ็กต์ PKCS #11 ที่เกี่ยวข้องกับโทเค็น PKCS #11

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "CREATE_SSC"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
p11_login = "TRUE"
p11_ssc_label = "<string>"
p11_ssc_config = "<openssl configuration file>"
```

โดยที่ *<openssl configuration file>* คือพาธแบบเต็ม UNIX และชื่อไฟล์ของไฟล์คอนฟิกูเรชัน OpenSSL ที่ถูกระบุค่าที่ถูกใช้ใน การสร้างใบรับรองที่ลงนามด้วยตนเอง

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11km_cmd_self_signed_certificate]
action_name = "CREATE_SSC"
p11_slot = "0"
p11_driver = "AIX"
p11_login = "TRUE"
p11_ssc_label = "Lab RADIUS Server"
p11_ssc_config = "/etc/radius/EAP-TLS/openssl.cnf"
```

สร้างคำร้องขอลงนามใบรับรอง PKCS #10

สร้างคำร้องขอใบรับรอง PKCS #10 หรือคำร้องขอลงนามใบรับรอง (CSR)

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "CREATE_CSR"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
p11_login = "TRUE"
p11_csr_label = "<string>"
p11_csr_file = "<path to CSR output file>"
p11_csr_type = "<DER or Base64>'
p11_csr_config = "<openssl configuration file>"
```

โดยที่ <*DER or Base64*> สร้าง ASN.1 (DER) ไฟล์เอาต์พุต CSR ที่เข้ารหัสแล้ว หรือไฟล์เอาต์พุต CSR ที่เข้ารหัส Base64 และ <*path to CSR output file*> อ้างถึงพาธแบบเต็ม UNIX และชื่อไฟล์ไปยังเอาต์พุต CSR

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11km_cmd_my_pkcs10_base64]
action_name = "CREATE_SSC"
p11_slot = "0"
p11_driver = "AIX"
p11_login = "TRUE"
p11_csr_label = "Lab RADIUS Server"
p11_csr_type = "Base64"
p11_csr_file = "/etc/radius/EAP-TLS/certreq.b64"
p11_csr_config = "/etc/radius/EAP-TLS/openssl.cnf"
```

คำสั่งแบตช์ต่อไปนี้สามารถใช้ได้ในเครื่องมือการควบคุมดูแล PKCS #11 (p11admin)

หมายเหตุ: หากต้องการใช้คำสั่งแบตช์ ให้ปฏิบัติดังต่อไปนี้:

1. สร้างและแก้ไขไฟล์แบตช์ตามที่อธิบายไว้ใน "การประมวลผลแบบแบตช์" ในหน้า 200
2. สร้างส่วน p11km_cmd ใหม่ที่มีแอ็ตทริบิวต์สำหรับ คำสั่งแบตช์ที่คุณต้องการใช้

รายการโทเค็น PKCS #11 ที่พร้อมใช้งาน

สร้างรายงานและแสดงผลโทเค็นและข้อมูลสล็อตสำหรับ โทเค็น PKCS #11 ที่พร้อมใช้งาน

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "ADM_LIST_TOKENS"
```

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

โดย <*boolean*> เป็นTRUE หรือ FALSE

ตัวอย่าง

```
[p11admin_cmd_list_tokens]
action_name = "ADM_LIST_TOKENS"
```

รายการกลไกของ PKCS#11 ที่พร้อมใช้งาน

สร้างรายงานและแสดงผลกลไกของ PKCS #11 ที่สนับสนุน โดยโทเค็น PKCS #11 (ตรงกับไดรเวอร์ และค่าแอ็ตทริบิวต์ของสล็อตที่เลือกไว้)

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "ADM_LIST_MECHANISMS"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
```

โดย <*slot number*> เป็นค่าจำนวนเต็มบวก และ <*driver name*> เป็นหนึ่งในค่าต่อไปนี้:

ค่า	คำอธิบาย
AIX	AIX OS Cryptographic Framework
IBM_4758_4960	IBM 4758/4960 Cryptographic Hardware Adapters
IBM_4764	IBM 4764 Cryptographic Hardware Adapter
อื่นๆ	ถ้าคุณระบุ OTHER คุณต้องระบุแอ็ตทริบิวต์ **p11_driver_path** ด้วย

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

แอ็ตทริบิวต์เสริม

```
p11_driver_path = "<path to PKCS#11 driver>"
```

โดยที่ *<path to PKCS#11 driver>* คือพาธแบบเต็ม UNIX และชื่อไฟล์ไลบรารี PKCS #11 ที่ถูกใช้สำหรับคำสั่ง แอ็ตทริบิวต์นี้สามารถระบุได้เฉพาะเมื่อ แอ็ตทริบิวต์ **p11_driver** ถูกตั้งค่าเป็น OTHER

ตัวอย่าง

```
[p11admin_cmd_list_4764_slot_0_mechs]
action_name = "ADM_LIST_MECHANISMS"
p11_driver = "IBM_4764"
p11_slot = "0"
start_gui = "TRUE"
```

แสดงข้อมูลสำหรับโทเค็น PKCS #11

แสดงผลโทเค็น PKCS #11 และข้อมูลสล็อตสำหรับโทเค็น PKCS #11

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "ADM_SHOW_TOKEN_INFO"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
```

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11admin_cmd]
action_name = "ADM_SHOW_TOKEN_INFO"
p11_slot = "411"
p11_driver = "IBM_4764"
```

เตรียมข้อมูลโทเค็น PKCS #11:

เตรียมข้อมูลโทเค็น PKCS #11 การเตรียมข้อมูลจะรีเซ็ตโทเค็น ลบข้อมูลและอ็อบเจ็กต์ทั้งหมดที่เก็บไว้ใน PKCS#11 และอนุญาตให้กำหนดเลเบลใหม่

ข้อควรสนใจ: เนื่องจากข้อมูลและอ็อบเจ็กต์ทั้งหมดของ PKCS #11 จะถูกลบในขั้นตอนการเตรียมข้อมูล ตรวจดูให้มั่นใจว่าคุณไม่จำเป็นต้องใช้อ็อบเจ็กต์และข้อมูลนั้นก่อนที่คุณจะเตรียมข้อมูล โทเค็น PKCS #11

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "ADM_INIT_TOKEN"
p11_driver = "<driver name>"
p11_slot = "<slot number>" ## SAME AS 'p11_init_slot'
p11_init_slot = "<slot number>" ## SAME AS 'p11_slot'
p11_init_label = "<string>" ## NEW TOKEN LABEL
```

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11admin_cmd]
action_name = "ADM_INIT_TOKEN"
p11_slot = "1"
p11_driver = "IBM_4764"
p11_init_slot = "1"
p11_init_label = "ABC Token"
```

ดูนาฬิกาสำหรับโทเค็น PKCS #11

แสดงผลนาฬิกาของฮาร์ดแวร์สำหรับโทเค็น PKCS #11 ถ้าโทเค็นนั้น มีนาฬิกา

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "ADM_CLOCK_VIEW"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
```

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11admin_cmd]
action_name = "ADM_CLOCK_VIEW"
p11_slot = "1"
p11_driver = "IBM_4764"
```

ตั้งค่านาฬิกาสำหรับโทเค็น PKCS #11

ตั้งค่านาฬิกาของฮาร์ดแวร์สำหรับโทเค็น PKCS #11 ถ้าโทเค็นนั้น มีนาฬิกา

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "ADM_CLOCK_SET"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
p11_clock_set = "<clock data>"
```

โดย <*clock data*> คือเวลาและวันที่ UTC ปัจจุบันที่มีรูปแบบดังนี้: HH:MM:SS mm-dd-YYYY

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11admin_cmd]
action_name = "ADM_CLOCK_SET"
p11_slot = "1"
p11_driver = "IBM_4764"
p11_clock_set = "23:59:59 12-31-1999"
```

รีเซ็ต PIN สำหรับผู้ใช้โทเค็น PKCS #11

รีเซ็ต PIN สำหรับผู้ใช้โทเค็น PKCS #11

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "ADM_RESET_USER_PIN"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
```

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11admin_cmd_change_so_pin]
action_name = "ADM_RESET_USER_PIN"
p11_driver = "AIX"
p11_slot = "0"
```

เปลี่ยนแปลง PIN สำหรับเจ้าหน้าที่ความปลอดภัยโทเค็น PKCS #11

เปลี่ยนแปลง PIN สำหรับเจ้าหน้าที่ความปลอดภัยโทเค็น PKCS #11 PIN นี้ถูกใช้เมื่อดำเนินการควบคุมดูแลโทเค็น

แอ็ตทริบิวต์ที่จำเป็น

```
action_name = "ADM_CHANGE_SO_PIN"
p11_driver = "<driver name>"
p11_slot = "<slot number>"
```

แอ็ตทริบิวต์ทางเลือก

```
start_gui = "<boolean>"
```

ตัวอย่าง

```
[p11admin_cmd_change_so_pin]
action_name = "ADM_CHANGE_SO_PIN"
p11_slot = "888"
p11_driver = "IBM_4764"
```

Pluggable Authentication Modules

เฟรมเวิร์ก Pluggable authentication module (PAM) ช่วยให้ ผู้ดูแลระบบมีความสามารถในการรวมกลไกการพิสูจน์ตัวตนหลายวิธี เข้าไว้ในระบบที่มีอยู่ผ่านการใช้โมดูลแบบปลั๊กได้

แอ็พพลิเคชันที่เปิดใช้งานเพื่อใช้ PAM สามารถ *ปลั๊กอิน* กับ เทคโนโลยีใหม่โดยไม่มีการแก้ไขแอ็พพลิเคชันที่มีอยู่ ความยืดหยุ่นนี้ ช่วยให้ผู้ดูแลระบบสามารถทำสิ่งต่อไปนี้:

- เลือกเซอร์วิสการพิสูจน์ตัวตนบนระบบสำหรับแอ็พพลิเคชัน
- ใช้กลไกการพิสูจน์ตัวตนหลายวิธีสำหรับเซอร์วิสที่เลือก
- เพิ่มเซอร์วิสโมดูลการพิสูจน์ตัวตนใหม่โดยไม่มีการแก้ไขแอ็พพลิเคชัน ที่มีอยู่

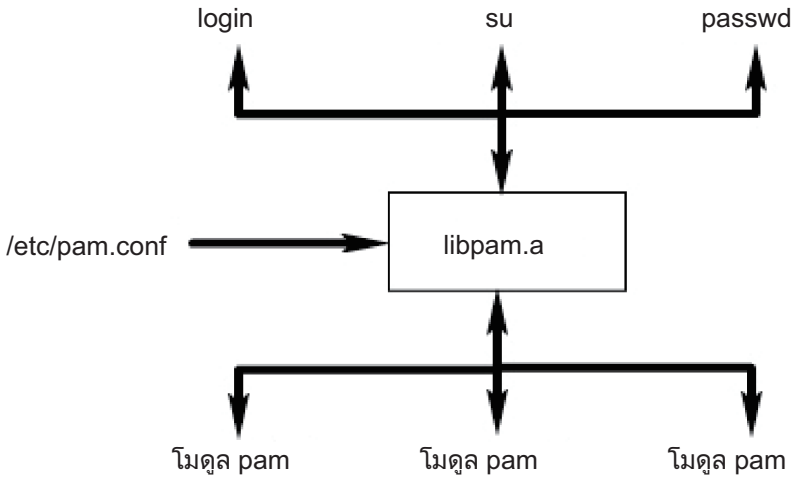
- ใช้รหัสผ่านที่ป้อนก่อนหน้านี้เพื่อทำการพิสูจน์ตัวตนกับหลาย โมดูล

เฟรมเวิร์ก PAM ประกอบด้วยไลบรารี โมดูลแบบปลั๊กได้ และ ไฟล์คอนฟิกูเรชัน ไลบรารี PAM นำ PAM application programming interface (API) ไปใช้และทำหน้าที่จัดการทรานแซกชัน PAM และร้องขอ PAM service programming interface (SPI) ที่กำหนด ในโมดูลแบบปลั๊กได้ โมดูลแบบปลั๊กได้ถูกโหลดแบบไดนามิก โดยไลบรารีโดยยึดตามเซอร์วิสการร้องขอและรายการในไฟล์คอนฟิกูเรชัน การดำเนินการสำเร็จนั้นไม่เพียงพิจารณาโดยโมดูลแบบปลั๊กได้ แต่ยัง โดยลักษณะการทำงานที่กำหนดสำหรับเซอร์วิสนั้น จากแนวความคิดของ *การสแต็ก* เซอร์วิสสามารถถูกกำหนดคอนฟิกเพื่อพิสูจน์ตัวตนผ่านวิธีการพิสูจน์ตัวตน หลายวิธี ถ้าสนับสนุน โมดูลยังสามารถถูกกำหนดคอนฟฟิกเพื่อใช้รหัสผ่านที่ส่งไป ก่อนหน้านี้แทนการพร้อมต์เพื่อให้ป้อนอินพุตอีก

ผู้ดูแลระบบสามารถกำหนดคอนฟิกระบบ AIX เพื่อใช้ PAM ผ่านการปรับแต่งของแอ็ตทริบิวต์ **auth_type** ใน usw stanza ของไฟล์ /etc/security/login.cfg ค่าติดตั้ง auth_type = PAM_AUTH กำหนดคอนฟิกคำสั่งที่เปิดใช้งาน PAM เพื่อเรียกใช้ PAM API โดยตรงกับการพิสูจน์ตัวตนแทน การใช้รูทีนการพิสูจน์ตัวตน AIX เชิงประวัติ การตั้งค่านี้ เป็นการตัดสินใจตอนรันไทม์และไม่จำเป็นต้องบูตระบบใหม่ เพื่อให้มีผล สำหรับข้อมูลเพิ่มเติมเกี่ยวกับแอ็ตทริบิวต์ **auth_type** โปรดดูที่ไฟล์ /etc/security/login.cfg เพื่อการอ้างอิง คำสั่งดั้งเดิม AIX และแอ็พพลิเคชันต่อไปนี้ได้ถูกปรับเปลี่ยนเพื่อจดจำแอ็ตทริบิวต์ **auth_type** และเปิดใช้งานสำหรับการพิสูจน์ตัวตน PAM:

- **login**
- **passwd**
- **su**
- **ftp**
- **telnet**
- **rlogin**
- **rexec**
- **rsh**
- **snappd**
- **imapd**
- **dtaction**
- **dtlogin**
- **dtsession**

ภาพประกอบต่อไปนี้แสดงการโต้ตอบระหว่างแอ็พพลิเคชันที่เปิดใช้งาน PAM ไลบรารี PAM ไฟล์คอนฟิกูเรชัน และโมดูล PAM บน ระบบที่ถูกกำหนดคอนฟิกเพื่อใช้ PAM แอ็พพลิเคชันที่เปิดใช้งาน PAM ร้องขอ PAM API ในไลบรารี PAM ไลบรารีจะพิจารณาโมดูล ที่เหมาะสมเพื่อโหลดตามรายการแอ็พพลิเคชันในไฟล์คอนฟิกูเรชัน และการเรียกใช้ PAM SPI ในโมดูล การสื่อสารที่เกิดขึ้นระหว่าง โมดูล PAM และแอ็พพลิเคชันผ่านการใช้ฟังก์ชันการสื่อสาร ที่ถูกใช้งานในแอ็พพลิเคชัน ความสำเร็จหรือความล้มเหลวจาก โมดูลและลักษณะการทำงานที่กำหนดในไฟล์คอนฟิกูเรชัน จะพิจารณาว่าจำเป็นต้องโหลดโมดูลอื่นอีกหรือไม่ ถ้าจำเป็น การประมวลผล จะดำเนินต่อไป ถ้าไม่ ผลลัพธ์จะถูกส่งกลับไปที่แอ็พพลิเคชัน



รูปที่ 3. เฟรมเวิร์ก PAM และ Entities. ภาพประกอบนี้ แสดงวิธีที่คำสั่งที่เปิดใช้งาน PAM ใช้ไลบรารี PAM เพื่อเข้าถึง โมดูล PAM ที่เหมาะสม

ไลบรารี PAM

ไลบรารี PAM `/usr/lib/libpam.a` มี PAM API ที่ทำหน้าที่เป็นอินเตอร์เฟสร่วมไปยังแอ็พพลิเคชัน PAM ทั้งหมดและยังควบคุมการโหลดโมดูล

โมดูลถูกโหลดโดยไลบรารี PAM โดยยึดตามลักษณะการทำงานการสแต็ก ที่กำหนดในไฟล์ `/etc/pam.conf`

ฟังก์ชัน PAM API ต่อไปนี้ร้องขอ PAM SPI ที่สัมพันธ์ ที่จัดให้มีโดยโมดูล PAM ตัวอย่าง pam_authenticate API ร้องขอ pam_sm_authenticate SPI ในโมดูล PAM

- `pam_authenticate`
- `pam_setcred`
- `pam_acct_mgmt`
- `pam_open_session`
- `pam_close_session`
- `pam_chauthtok`

ไลบรารี PAM ยังรวม APIs เฟรมเวิร์กมากมายที่เปิดใช้งาน แอ็พพลิเคชันเพื่อร้องขอโมดูล PAM และส่งข้อมูลไปยังโมดูล PAM ตารางต่อไปนี้แสดง APIs เฟรมเวิร์ก PAM ที่ถูกนำไปใช้ใน AIX รวมถึงฟังก์ชัน:

API เฟรมเวิร์ก PAM	ฟังก์ชัน
pam_start	สร้างเซสชัน PAM
pam_end	จบการทำงานเซสชัน PAM
pam_get_data	เรียกข้อมูลเฉพาะของโมดูล
pam_set_data	ตั้งค่าข้อมูลเฉพาะของโมดูล
pam_getenv	เรียกข้อมูลค่าของตัวแปรสภาวะแวดล้อม PAM ที่กำหนด
pam_getenvlist	เรียกข้อมูลรายการของตัวแปรสภาวะแวดล้อม PAM ที่กำหนดทั้งหมด รวมถึงค่าตัวแปร
pam_putenv	ตั้งค่าตัวแปรสภาวะแวดล้อม PAM
pam_get_item	เรียกข้อมูล PAM ร่วม
pam_set_item	ตั้งค่าข้อมูล PAM ร่วม
pam_get_user	เรียกข้อมูลชื่อผู้ใช้
pam_strerror	รับข้อความแสดงความผิดพลาดมาตรฐาน PAM

โมดูล PAM

โมดูล PAM อนุญาตให้ใช้กลไกการพิสูจน์ตัวตนหลายวิธี แบบเป็นกลุ่มหรือเป็นอิสระต่อกันบนระบบ

โมดูล PAM ที่กำหนดต้องนำใช้ประเภทโมดูลอย่างน้อยหนึ่งประเภทจากสี่ประเภท ประเภทโมดูลถูกอธิบายดังนั้น พร้อมกับ PAM SPIs ที่สัมพันธ์ที่จำเป็นต้องใช้เพื่อยืนยันประเภทโมดูล

โมดูลการพิสูจน์ตัวตน

พิสูจน์ตัวตนผู้ใช้และตั้งค่า รีเฟรช หรือทำลาย credentials โมดูล เหล่านี้ระบุผู้ใช้โดยยึดตามการพิสูจน์ตัวตนและ credentials

ฟังก์ชันของโมดูล การพิสูจน์ตัวตน:

- pam_sm_authenticate
- pam_sm_setcred

โมดูลการจัดการบัญชีผู้ใช้

พิจารณาความถูกต้องของบัญชีผู้ใช้และการเข้าถึงต่อมาหลังจาก identification จากโมดูลการพิสูจน์ตัวตน การตรวจสอบที่ดำเนินการโดย โมดูลเหล่านี้โดยทั่วไปจะรวมการหมดอายุบัญชีผู้ใช้และข้อจำกัดรหัสผ่าน

ฟังก์ชันของโมดูลการจัดการบัญชีผู้ใช้:

- pam_sm_acct_mgmt

โมดูลการจัดการเซสชัน

Initiate และยกเลิกเซสชันผู้ใช้ นอกจากนั้น การสนับสนุนสำหรับ การตรวจสอบเซสชันต้องถูกจัดให้มี

ฟังก์ชันของโมดูลการจัดการเซสชัน:

- pam_sm_open_session
- pam_sm_close_session

โมดูลการจัดการรหัสผ่าน

ทำการแก้ไขรหัสผ่าน และการจัดการแอ็ตทริบิวต์ที่เกี่ยวข้อง

ฟังก์ชัน ของโมดูลการจัดการรหัสผ่าน:

- pam_sm_chauthtok

ไฟล์คอนฟิกูเรชัน PAM

ไฟล์คอนฟิกูเรชัน /etc/pam.conf ประกอบด้วย รายการเซอร์วิสสำหรับแต่ละประเภทโมดูล PAM และทำหน้าที่จัดเส้นทางเซอร์วิส ผ่านโมดูลพาธที่กำหนด

รายการในไฟล์ประกอบด้วยฟิลด์ที่คั่นด้วย whitespace ต่อไปนี้:

service_name module_type control_flag module_path module_options

คำอธิบายของฟิลด์เหล่านี้มีดังนี้:

service_name

ระบุชื่อของเซอร์วิส คีย์เวิร์ด OTHER ถูกใช้เพื่อ กำหนดโมดูลดีฟอลต์เพื่อใช้สำหรับแอพพลิเคชันที่ไม่ถูกระบุ ในรายการ

module_type

ระบุประเภทโมดูลสำหรับเซอร์วิส ประเภทโมดูล ที่ถูกต้องคือ **auth, account, session** หรือ **password** โมดูลที่กำหนดจะให้การสนับสนุน สำหรับประเภทโมดูลอย่างน้อยหนึ่งประเภท

control_flag

ระบุลักษณะการสแต็กสำหรับโมดูล แฟล็กการควบคุม ที่สนับสนุนได้แก่ required, requisite, sufficient หรือ optional

module_path

ระบุโมดูลเพื่อโหลดเซอร์วิส ค่าที่ใช้ได้สำหรับ *module_path* สามารถ ระบุเป็นพาธเต็มไปยังโมดูล หรือระบุเพียงชื่อโมดูล ถ้าระบุพาธเต็มไปยังโมดูล ไลบรารี PAM จะใช้ *module_path* นั้นเพื่อโหลดเซอร์วิส 32 บิต หรือใช้ไดเรกทอรีย่อย 64 ไดเร็กทอรีสำหรับเซอร์วิส 64 บิต ถ้าพาธเต็มไปยังโมดูลไม่ถูกระบุ ไลบรารี PAM จะเพิ่มส่วนนำหน้า /usr/lib/security (สำหรับเซอร์วิส 32 บิต) หรือ /usr/lib/security/64 (สำหรับเซอร์วิส 64 บิต) ไปยังชื่อโมดูล

module_options

ระบุรายการของอ็อพชันที่คั่นด้วยช่องว่างที่สามารถ ส่งไปยังเซอร์วิสโมดูล ค่าสำหรับฟิลด์นี้ขึ้นอยู่กับ อ็อพชันที่โมดูลสนับสนุนซึ่งกำหนดในฟิลด์ *module_path* ฟิลด์นี้เป็นทางเลือก

รายการที่ผิดรูปแบบ หรือรายการที่มีค่าไม่ถูกต้องสำหรับฟิลด์ **module_type** หรือ **control_flag** ถูกละเว้นโดยไลบรารี PAM รายการที่ขึ้นต้นด้วยอักขระเครื่องหมายตัวเลข (#) ที่ตำแหน่งเริ่มต้นของบรรทัดจะถูกละเว้นเช่นกันเนื่องจากเครื่องหมายนี้แสดงว่าเป็นความคิดเห็น

PAM สนับสนุนแนวคิดโดยทั่วไปที่ถูกอ้างถึงเป็น "การสแต็ก" ซึ่งอนุญาตให้ใช้หลาย กลไกสำหรับแต่ละเซอร์วิส การสแต็กถูกนำไปใช้ในไฟล์คอนฟิกูเรชัน โดยการสร้างรายการหลายรายการสำหรับเซอร์วิสที่มีฟิลด์ **module_type** เหมือนกัน โมดูลถูกร้องขอตามลำดับที่แสดงรายการในไฟล์ สำหรับเซอร์วิสที่กำหนด ที่มีผลลัพธ์สุดท้ายถูกกำหนดโดยฟิลด์ **control_flag** ที่ระบุสำหรับแต่ละรายการ ค่าที่ถูกต้องสำหรับฟิลด์ **control_flag** และลักษณะการทำงานที่สัมพันธ์กันในสแต็กเป็นดังนี้:

ค่าสำหรับฟิลด์ control_flag	ลักษณะการทำงาน
required	โมดูลที่จำเป็นทั้งหมดในสแต็กต้องได้ผลลัพธ์ ที่สำเร็จ ถ้ามีอย่างน้อยหนึ่งโมดูลที่จำเป็นล้มเหลว โมดูลที่จำเป็นทั้งหมด ในสแต็กจะถูกพยายามดำเนินการ แต่ข้อผิดพลาดจากโมดูลที่จำเป็นที่ล้มเหลวอันดับแรก จะถูกส่งกลับ
requisite	คล้ายกับที่จำเป็น ยกเว้นว่าถ้าโมดูลที่ต้องการ ล้มเหลว จะไม่มีโมดูลอื่นในสแต็กถูกประมวลผลต่อและจะส่งกลับ โค้ดความล้มเหลวแรกจากโมดูลที่จำเป็นหรือที่ต้องการโดยทันที
sufficient	ถ้าโมดูลแฟล็กเป็นความสำเร็จแบบเพียงพอ และไม่มีโมดูลที่ จำเป็น หรือต้องการอยู่ก่อนหน้าเกิดล้มเหลว โมดูลที่เหลือทั้งหมดในสแต็ก จะถูกข้ามและส่งกลับโดยแสดงว่าสำเร็จ
optional	ถ้าไม่มีโมดูลใดในสแต็กเป็นโมดูลที่จำเป็น และไม่มีโมดูลที่เพียงพอทำงานสำเร็จ ดังนั้นอย่างน้อยหนึ่งโมดูลทางเลือกสำหรับ เซอร์วิสต้องถูกดำเนินการสำเร็จ ถ้ามีโมดูลอื่นในสแต็กทำสำเร็จ ความล้มเหลวในโมดูลทางเลือกจะถูกละเว้น

เซ็ตย่อย/etc/pam.conf ต่อไปนี้เป็นตัวอย่างของ การสแต็กในประเภทโมดูล auth สำหรับเซอร์วิสการล็อกอิน

```
#
# PAM configuration file /etc/pam.conf
#

# Authentication Management
login   auth     required       /usr/lib/security/pam_ckfile    file=/etc/nologin
login   auth     required       /usr/lib/security/pam_aix
login   auth     optional       /usr/lib/security/pam_test      use_first_pass
OTHER   auth     required       /usr/lib/security/pam_prohibit
```

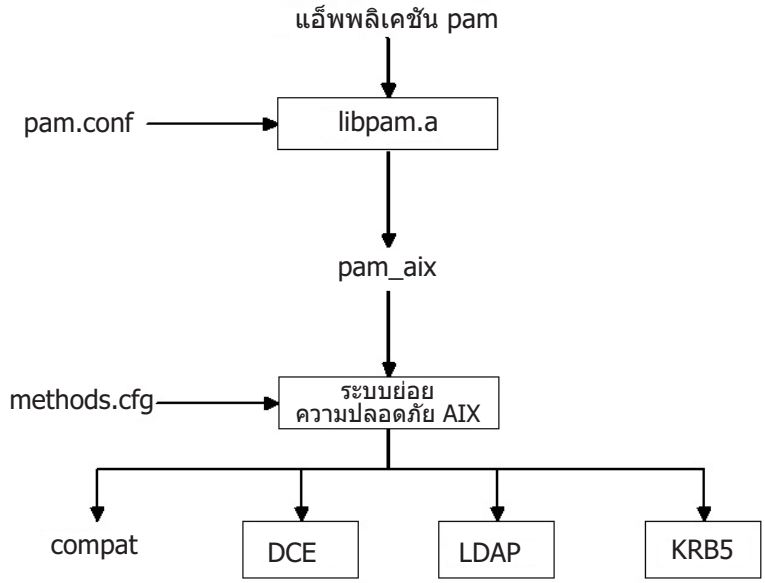
ตัวอย่างของไฟล์คอนฟิกูเรชันมีสามรายการสำหรับ เซอร์วิสการล็อกอิน การระบุทั้ง pam_ckfile และ pam_aix เป็น ที่จำเป็น ทั้งสองโมดูลจะถูกรันและทั้งสองต้องทำสำเร็จเพื่อให้ผลลัพธ์โดยรวมแสดงเป็นสำเร็จ รายการที่สามสำหรับโมดูล pam_test ที่ไม่จริง เป็นทางลือกและการสำเร็จหรือล้มเหลวของโมดูลนี้จะไม่มีผลว่าผู้ใช้ ล็อกอินได้หรือไม่ อ็อพชัน use_first_pass สำหรับโมดูล pam_test จำเป็นต้องใช้รหัสผ่านที่ป้อนก่อนหน้านี้เพื่อใช้แทนการพร้อมต์เพื่อรับ ค่าใหม่

การใช้คีย์เวิร์ด OTHER เป็นชื่อเซอร์วิสทำให้ค่าดีฟอลต์ถูกตั้งค่าสำหรับ เซอร์วิสอื่นที่ไม่ได้ประกาศโดยชัดเจนใน ไฟล์คอนฟิกูเรชัน การตั้งค่าดีฟอลต์ทำให้แน่ใจว่ากรณีที่เป็นไปได้ทั้งหมดสำหรับประเภทโมดูลที่กำหนด จะครอบคลุมอย่างน้อยหนึ่งโมดูล ในกรณีของตัวอย่างนี้ เซอร์วิสทั้งหมด นอกเหนือจากล็อกอินจะล้มเหลวเสมอ เนื่องจากโมดูล pam_prohibit ส่งกลับค่าความล้มเหลว PAM สำหรับการร้องขอทั้งหมด

โมดูล pam_aix

โมดูล pam_aix คือโมดูล PAM ที่จัดให้มีการเข้าถึงแอ็พพลิเคชันที่เปิดใช้งาน PAM ในเซอร์วิสการรักษาความปลอดภัย AIX โดยการจัดให้มีอินเตอร์เฟสที่เรียกใช้เซอร์วิส AIX ที่เทียบเท่าที่ มีอยู่

ในทางกลับกันเซอร์วิสเหล่านี้ถูกดำเนินการโดยโมดูลการพิสูจน์ตัวตนที่ โหลดได้หรือฟังก์ชันในตัว AIX ในตัว โดยยึดตามนิยามของผู้ใช้และการตั้งค่าที่เกี่ยวข้อง ในไฟล์ methods.cfg โค้ดระบุความผิดพลาดใดๆ ที่สร้างขึ้น ระหว่างการทำงานของเซอร์วิส AIX จะถูกแม็พกับโค้ดระบุความผิดพลาด PAM ที่เกี่ยวข้อง



รูปที่ 4. แอ็พพลิเคชัน PAM ไปยังพาธระบบย่อยการรักษาความปลอดภัย AIX

ภาพประกอบนี้แสดงพาธที่การเรียกใช้ API ของแอ็พพลิเคชัน PAM จะเป็นไปตามถ้าไฟล์ /etc/pam.conf ถูกตั้งค่า เพื่อใช้งานโมดูล pam_aix ดังแสดงใน แผนภาพ การผสมผสานนี้อนุญาตให้ผู้ใช้ได้รับการพิสูจน์ตัวตนโดยใช้โมดูลการพิสูจน์ตัวตนที่โหลดได้ (DCE, LDAP หรือ KRB5) หรือในไฟล์ AIX (compat)

โมดูล pam_aix ถูกติดตั้งในไดเร็กทอรี /usr/lib/security การผสมผสานของโมดูล pam_aix จำเป็นต้อง ตั้งค่าไฟล์ /etc/pam.conf เพื่อใช้งาน โมดูล การสแต็กยังคงใช้ได้แต่ไม่แสดงใน ตัวอย่างของไฟล์ /etc/pam.conf ต่อไปนี้:

```
#
# Authentication management
#
OTHER   auth     required       /usr/lib/security/pam_aix

#
# Account management
#
OTHER   account  required       /usr/lib/security/pam_aix

#
# Session management
#
OTHER   session  required       /usr/lib/security/pam_aix

#
# Password management
#
OTHER   password required       /usr/lib/security/pam_aix
```

โมดูล pam_aix มีการนำไปใช้สำหรับ ฟังก์ชัน pam_sm_authenticate, pam_sm_chauthok และ pam_sm_acct_mgmt SPI pam_sm_setcred, pam_sm_open_session และ pam_sm_close_session SPI ยังถูกนำไปใช้ในโมดูล pam_aix แต่ฟังก์ชัน SPI เหล่านี้ ส่งกลับการร้องขอ PAM_SUCCESS

ต่อไปนี้คือการแม็พอย่างย่อ ของการเรียกใช้ PAM SPI ไปยังระบบย่อยการรักษาความปลอดภัย AIX:

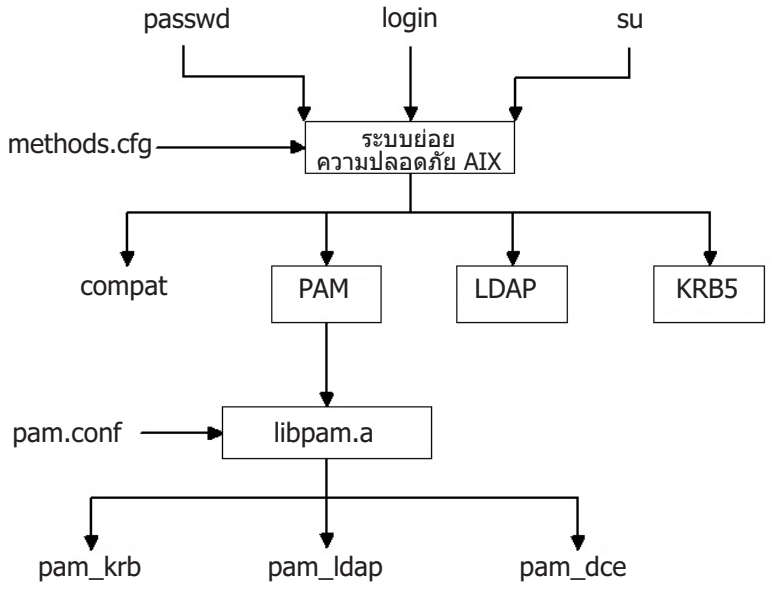
```
 PAM SPI                    AIX
=========                  =====
pam_sm_authenticate   -->  authenticate
pam_sm_chauthtok      -->  passwdexpired, chpass
                           Note: passwdexpired is only checked if the
                           PAM_CHANGE_EXPIRED_AUTHTOK flag is passed in.
pam_sm_acct_mgmt      -->  loginrestrictions, passwdexpired
pam_sm_setcred        -->  No comparable mapping exists, PAM_SUCCESS returned
pam_sm_open_session   -->  No comparable mapping exists, PAM_SUCCESS returned
pam_sm_close_session  -->  No comparable mapping exists, PAM_SUCCESS returned
```

ข้อมูลที่จะต้องถูกส่งไปยังระบบย่อยการรักษาความปลอดภัย AIX สามารถถูกตั้งค่าโดยใช้ฟังก์ชัน pam_set_item ก่อน การใช้โมดูล หรือโมดูล pam_aix สำหรับข้อมูลถ้า ยังไม่มีอยู่

โมดูลการพิสูจน์ตัวตนแบบโหลดได้ของ PAM

เซอร์วิสการรักษาความปลอดภัย AIX สามารถตั้งค่าเพื่อเรียกใช้โมดูล PAM ผ่านการใช้งาน เฟรมเวิร์กโมดูลการพิสูจน์ตัวตนแบบโหลดได้ของ AIX ที่มีอยู่

เมื่อไฟล์ /usr/lib/security/methods.cfg ถูกตั้งค่า อย่างถูกต้อง โหลดโมดูล PAM จะจัดเส้นทางเซอร์วิสการรักษาความปลอดภัย AIX (passwd, login และอื่นๆ) ไปยังไลบรารี PAM ไลบรารี PAM ตรวจสอบไฟล์ /etc/pam.conf เพื่อพิจารณาว่าโมดูล PAM ใดที่จะใช้ และทำการเรียกใช้ PAM SPI ที่สัมพันธ์ ค่าส่งกลับจาก PAM ถูกแม็พกับโค้ดระบุความผิดพลาด AIX และถูกส่งกลับ ไปยังโปรแกรมที่เรียกใช้



รูปที่ 5. เซอร์วิสการรักษาความปลอดภัย AIX ไปยัง โมดูลพาธ PAM

ภาพประกอบนี้แสดงพาธที่การเรียกใช้เซอร์วิสการรักษาความปลอดภัย AIX เกิดขึ้นเมื่อ PAM ถูกตั้งค่าอย่างถูกต้อง โมดูล PAM ที่แสดง (pam_krb, pam_ldap และ pam_dce) ถูกแสดงรายเป็นเป็นตัวอย่างของวิธีแก้ปัญหา ของบุคคลที่สาม

โหลดโมดูล PAM ถูกติดตั้งในไดเร็กทอรี /usr/lib/security และเป็นโมดูลที่ทำการพิสูจน์ตัวตนเท่านั้น โมดูล PAM ต้องถูกรวมเข้ากับ ฐานข้อมูลเพื่อจัดรูปแบบโหลดโมดูลผสม ตัวอย่างต่อไปนี้ แสดง stanzas ที่สามารถเพิ่มในไฟล์ methods.cfg เพื่อจัดรูปแบบโมดูล PAM ผสมที่มีฐานข้อมูลที่เรียกใช้ไฟล์ คีย์เวิร์ด BUILTIN สำหรับแอ็ตทริบิวต์ **db** กำหนดฐานข้อมูลเป็นไฟล์ UNIX

```
PAM:
        program = /usr/lib/security/PAM

PAMfiles:
        options = auth=PAM,db=BUILTIN
```

การสร้างและการแก้ไขผู้ใช้จะถูกดำเนินการโดยใช้อ็อพชัน –R กับคำสั่งการจัดการและโดยการตั้งค่าแอ็ตทริบิวต์ *SYSTEM* เมื่อสร้างผู้ใช้ ตัวอย่าง:

```
mkuser -R PAMfiles SYSTEM=PAMfiles registry=PAMfiles pamuser
```

การดำเนินการนี้แจ้งการเรียกใช้เพิ่มไปยังเซอร์วิสการรักษาความปลอดภัย AIX (login, passwd และอื่นๆ) เพื่อใช้โหลดโมดูล PAM สำหรับการพิสูจน์ตัวตน ในขณะที่ฐานข้อมูล **ไฟล์** ถูกใช้สำหรับโมดูลผสมในตัวอย่างนี้ ฐานข้อมูลอื่นๆ เช่น LDAP ยังสามารถถูกใช้ได้ถ้าได้รับการติดตั้งไว้ การสร้างผู้ใช้ ดังอธิบายก่อนหน้านี้จะส่งผลต่อการแม็พต่อไปนี้ของการรักษาความปลอดภัย AIX ไปยังการเรียกใช้ PAM API:

```
 AIX                    PAM API
=====                  =========
authenticate       -->  pam_authenticate
chpass             -->  pam_chauthtok
passwdexpired      -->  pam_acct_mgmt
passwdrestrictions -->  No comparable mapping exists, success returned
```

การกำหนดไฟล์ /etc/pam.conf เองช่วยให้ การเรียกใช้ PAM API ถูกเปลี่ยนเส้นทางไปยังโมดูล PAM ที่ต้องการสำหรับการพิสูจน์ตัวตน ในการปรับกลไกการพิสูจน์ตัวตนเพิ่มเติม สามารถนำการสแต็กมาใช้

ข้อมูลที่พร้อมต์เพื่อรับค่าโดยเซอร์วิสการรักษาความปลอดภัย AIX ถูกส่งไปยัง PAM ผ่านฟังก์ชัน pam_set_item เนื่องจากไม่สามารถจัดการไดอะล็อกผู้ใช้ให้เหมาะสมจาก PAM โมดูล PAM ถูกเขียนเพื่อการรวมกันกับโมดูล PAM ควรเรียก ข้อมูลทั้งหมดด้วยการเรียกใช้ pam_get_item และไม่ควร พยายามพร้อมต์ผู้ใช้ให้ป้อนข้อมูลเนื่องจากการดำเนินการนี้ถูกจัดการ โดยเซอร์วิสการรักษาความปลอดภัย

การตรวจหาลูปถูกจัดให้มีเพื่อหาข้อผิดพลาดการตั้งค่าที่อาจเกิดขึ้น ที่ซึ่งเซอร์วิสการรักษาความปลอดภัย AIX ถูกจัดเส้นทางไปยัง PAM จากนั้นโมดูล PAM ก็จะพยายาม เรียกใช้เซอร์วิสการรักษาความปลอดภัย AIX เพื่อดำเนินการ การตรวจพบเหตุการณ์ที่เป็นแบบลูปนี้จะ ส่งผลให้เกิดความล้มเหลวของการดำเนินการที่ต้องการโดยทันที

หมายเหตุ: ไฟล์ /etc/pam.conf *ไม่* ควร ถูกเขียนเพื่อการใช้งานโมดูล pam_aix เมื่อใช้การรวม PAM จากเซอร์วิสการรักษาความปลอดภัย AIX ไปยังโมดูล PAM เนื่องจากทำให้เกิดสภาวะของการลูป

การเพิ่มโมดูล PAM

คุณสามารถเพิ่มโมดูล PAM เพื่อเปิดใช้กลไกการพิสูจน์ตัวตนหลายวิธี

1. วางโมดูลเวอร์ชัน 32 บิตในไดเร็กทอรี /usr/lib/security และโมดูลเวอร์ชัน 64 บิตในไดเร็กทอรี /usr/lib/security/64

2. ตั้งค่าความเป็นเจ้าของไฟล์เป็น root และสิทธิ เป็น 555 ไลบรารี PAM ไม่โหลด โมดูลใดๆ ที่ไม่ได้เป็นเจ้าของโดยผู้ใช้ root
3. อัพเดตไฟล์คอนฟิกูเรชัน /etc/pam.conf เพื่อรวมโมดูลในรายการสำหรับชื่อเซอร์วิสที่ต้องการ
4. ทดสอบเซอร์วิสที่ได้รับผลเพื่อให้แน่ใจว่าทำงานได้ อย่าล็อกออกจากระบบจนกว่าจะทำการทดสอบการล็อกอินแล้ว

การเปลี่ยนแปลงไฟล์ /etc/pam.conf

มีสองสามสิ่งที่คุณควรคำนึงถึงก่อนการเปลี่ยนแปลงไฟล์ /etc/pam.conf

เมื่อทำการเปลี่ยนแปลงไฟล์คอนฟิกูเรชัน /etc/pam.conf ขอให้พิจารณาข้อกำหนดต่อไปนี้:

- ไฟล์ควรมีเจ้าของเป็นผู้ใช้ root และการรักษาความปลอดภัยกลุ่ม สิทธิ ของไฟล์ต้องเป็น 644 เพื่ออนุญาตให้ทุกคนสามารถเข้าถึงเพื่ออ่าน แต่อนุญาต root ให้แก้ไขได้เท่านั้น
- สำหรับการรักษาความปลอดภัยที่ยอดเยี่ยมยิ่งขึ้น ให้พิจารณาการกำหนดคอนฟิกแต่ละเซอร์วิสที่เปิดใช้งาน PAM โดยชัดเจนและใช้โมดูล pam_prohibit สำหรับ คีย์เวิร์ดเซอร์วิส OTHER
- อ่านเอกสารคู่มือที่จัดให้สำหรับโมดูลที่พิจารณาเลือก และพิจารณาว่า แฟล็กการควบคุม และอ็อพชันใดที่ได้รับการสนับสนุน และผลกระทบที่จะมีขึ้น
- เลือกการจัดลำดับโมดูลและแฟล็กการควบคุมอย่างระมัดระวัง โปรดตระหนักถึง การทำงานของแฟล็กการควบคุมจะเป็น จำเป็น ต้องการ เพียงพอ และ เป็นทางเลือก ในโมดูลแบบสแต็ก

หมายเหตุ: การกำหนดคอนฟิกของไฟล์คอนฟิกูเรชัน PAM ที่ไม่ถูกต้อง อาจส่งผลในระบบไม่สามารถล็อกอินได้ เนื่องจากคอนฟิกูเรชัน จะใช้กับผู้ใช้ทั้งหมดรวมถึง root หลังจากทำการเปลี่ยนแปลงไฟล์ ให้ทดสอบ แอ็พพลิเคชันที่ได้รับผลเสมอก่อนที่จะล็อกออกจากระบบ ระบบ ที่ไม่สามารถล็อกอินได้สามารถกู้คืนได้โดยการบูตระบบในโหมดการบำรุงรักษา และแก้ไขไฟล์คอนฟิกูเรชัน /etc/pam.conf ให้ถูกต้อง

การเปิดใช้งานการดีบัก PAM

ไลบรารี Pluggable Authentication Modules (PAM) สามารถจัดเตรียมข้อมูลดีบักในระหว่างการเรียกทำงาน หลังจากที่เปิดใช้งานระบบ เพื่อรวบรวมเอาต์พุตการดีบัก, ข้อมูลที่รวบรวมจะสามารถใช้เพื่อติดตามการเรียก PAM API และกำหนดจุดล้มเหลวในการติดตั้ง PAM ปัจจุบัน

เมื่อต้องการเปิดใช้งานเอาต์พุตดีบัก PAM, ให้ทำตามขั้นตอนต่อไปนี้:

1. สร้างไฟล์ว่างที่ชื่อ pam_debug ในไดเร็กทอรี /etc/pam_debug โดยใช้คำสั่ง **touch**, หากไม่มีไฟล์อยู่ ไลบรารี PAM ตรวจสอบไฟล์ /etc/pam_debug และเปิดใช้งานเอาต์พุต syslog หากพบ
2. แก้ไขไฟล์ /etc/syslog.conf เพื่อระบุไฟล์ที่จะล็อก auth ข้อความ syslog ที่ลำดับความสำคัญที่คุณต้องการ ตัวอย่าง เช่น, เพื่อส่งข้อความระดับดีบักของ PAM ไปยังไฟล์ /var/log/auth.log, ให้เพิ่มข้อความต่อไปนี้เป็นบรรทัดใหม่ในไฟล์ syslog.conf :

   ```
   *.debug /var/log/auth.log
   ```

3. สร้างไฟล์เอาต์พุตที่ถูกโอนย้ายในขั้นตอน 2, /var/log/auth.log, โดยใช้คำสั่ง **touch**, หากไม่มีอยู่
4. เมื่อต้องการรีสตาร์ท syslogd daemon เพื่อให้การเปลี่ยนคอนฟิกูเรชัน จดจำได้, และทำตามขั้นตอนต่อไปนี้:
   a. หยุด syslog daemon โดยป้อนคำสั่งต่อไปนี้:

      ```
      stopsrc -s syslogd
      ```

   b. เริ่มต้น syslog daemon โดยป้อนคำสั่งต่อไปนี้ข:

```
startsrc -s syslogd
```

เมื่อแอ็พพลิเคชัน PAM ถูกรีสตาร์ท, ข้อความดีบัก ถูกรวบรวมไว้ในไฟล์เอาต์พุตที่ถูกนิยามในไฟล์คอนฟิกูเรชัน `/etc/syslog.conf`

การสนับสนุน OpenSSH และ Kerberos เวอร์ชัน 5

Kerberos คือกลไกการพิสูจน์ตัวตนที่มีวิธี การพิสูจน์ตัวตนสำหรับผู้ใช้เน็ตเวิร์กอย่างปลอดภัย โดยป้องกันการส่งข้อมูล รหัสผ่านที่เป็นแบบข้อความโดยตรงผ่านเน็ตเวิร์กโดยการเข้ารหัสข้อความ การพิสูจน์ตัวตนระหว่างไคลเอ็นต์กับข้อความ นอกจากนั้น Kerberos ยังจัดให้มี ระบบสำหรับการอนุญาตในรูปของโทเค็นการจัดการ หรือ credentials

ในการพิสูจน์ตัวตนผู้ใช้โดยใช้ Kerberos ผู้ใช้จะรันคำสั่ง **kinit** เพื่อให้ได้รับ credentials เริ่มต้นจากเซิร์ฟเวอร์ Kerberos กลางที่รู้จักในชื่อ KDC (Key Distribution Center) KDC ตรวจสอบผู้ใช้และส่ง credentials เริ่มต้นกลับไปให้ผู้ใช้ หรือที่เรียก TGT (Ticket-Granting Ticket) จากนั้นผู้ใช้สามารถเริ่มทำงานเซสชันล็อกอินรีโมตโดยใช้เซอร์วิส เช่น Telnet ที่ปิดใช้ Kerberos หรือ OpenSSH และ Kerberos พิสูจน์ตัวตน ผู้ใช้โดยการรับ credentials ผู้ใช้จาก KDC Kerberos ดำเนิน การพิสูจน์ตัวตนนี้โดยไม่จำเป็นต้องมีการโต้ตอบกับผู้ใช้ ดังนั้น ผู้ใช้ไม่จำเป็นต้องป้อนรหัสผ่านเพื่อล็อกอิน Kerberos เวอร์ชันของ IBM หรือรู้จักในชื่อ Network Authentication Service (NAS) NAS สามารถถูกติดตั้ง จากซีดี AIX Expansion Pack ซึ่งมีอยู่ในแพ็กเกจ `krb5.client.rte` และ `krb5.server.rte` เริ่มตั้งแต่ OpenSSH 3.6 รีลีสของเดือนกรกฎาคม 2545 นั้น OpenSSH สนับสนุน การพิสูจน์ตัวตน Kerberos 5 และการอนุญาตผ่าน NAS เวอร์ชัน 1.3

OpenSSH เวอร์ชัน 3.8 และใหม่กว่าสนับสนุนการพิสูจน์ตัวตน Kerberos 5 และการอนุญาตผ่าน NAS เวอร์ชัน 1.4 การโอนย้ายระบบใดๆ จาก NAS (Kerberos) เวอร์ชันก่อนหน้าจำเป็นต้องทำก่อนการอัพเดต OpenSSH OpenSSH เวอร์ชัน 3.8.x จะทำงานได้กับ NAS เวอร์ชัน 1.4 หรือใหม่กว่าเท่านั้น

AIX ได้สร้าง OpenSSH ที่มีการพิสูจน์ตัวตน Kerberos เป็นวิธีการทางเลือก ถ้า ไลบรารี Kerberos ไม่ถูกติดตั้งบนระบบ เมื่อ OpenSSH รัน การพิสูจน์ตัวตน Kerberos จะถูกข้ามและ OpenSSH พยายามใช้วิธีการพิสูจน์ตัวตน ที่ตั้งค่าไว้รายการถัดไป (เช่นการพิสูจน์ตัวตน AIX)

หลังจากคุณติดตั้ง Kerberos ขอแนะนำให้คุณอ่าน เอกสาร Kerberos ก่อนทำการตั้งค่าเซิร์ฟเวอร์ Kerberos สำหรับ ข้อมูลเพิ่มเติมเกี่ยวกับวิธีติดตั้งและจัดการ Kerberos โปรดอ้างอิง *IBM Network Authentication Service Version 1.3 for AIX: Administrator's and User's Guide* ที่อยู่ใน พาธ `/usr/lpp/krb5/doc/html/lang/ADMINGD.htm`

ข้อมูลที่เกี่ยวข้อง:

OpenSSH

อิมเมจ OpenSSH

ใช้ขั้นตอนต่อไปนี้เพื่อติดตั้งอิมเมจ OpenSSH:

1. ไปที่เว็บไซต์ AIX Expansion Pack (http://www.ibm.com/systems/power/software/aix/expansionpack/index.html)
2. คลิก **ดาวน์โหลด** ใน ส่วน ข้อมูลเพิ่มเติม
3. ล็อกอินโดยใช้ ID และรหัสผ่านของคุณเพื่อเข้าถึง แพ็กเกจที่มี
4. เลือก **OpenSSH** และคลิก **ทำต่อ**
5. ยอมรับข้อตกลงการอนุญาตใช้สิทธิ์เพื่อดาวน์โหลดแพ็คเกจ

6. แยกอิมเมจแพ็กเกจโดยใช้คำสั่ง **uncompress** *packagename* ตัวอย่าง:

   ```
   uncompress OpenSSH_6.0.0.6102.tar.Z
   ```

7. Untar แพ็กเกจด้วยคำสั่ง **tar -xvf** *packagename* ตัวอย่าง:

   ```
   tar -xvf OpenSSH_6.0.0.6102.tar
   ```

8. รันคำสั่ง **inutoc**
9. รันคำสั่ง **smitty install**
10. เลือก **Install and Update Software**
11. เลือก **Update Installed Software to Latest Level (Update All)**
12. พิมพ์จุด (.) ในฟิลด์สำหรับ **INPUT device / directory for software** และกด Enter
13. เลื่อนลงไปที่ **ACCEPT new license agreements** และ กดปุ่ม **Tab** เพื่อเปลี่ยนฟิลด์เป็น **Yes**
14. กดปุ่ม Enter สองครั้งเพื่อเริ่มทำการติดตั้ง

อิมเมจ OpenSSH คืออิมเมจระดับเบื้องต้น ไม่ใช่ Program Temporary Fixes (PTFs) เมื่อทำการติดตั้งโค้ดก่อนหน้าทั้งหมดของ เวอร์ชันก่อนหน้าถูกเขียนทับด้วยอิมเมจของเวอร์ชันใหม่

การตั้งค่าการคอมไพล์ OpenSSH

ข้อมูลต่อไปนี้อธิบายวิธีที่โค้ด OpenSSH ถูกคอมไพล์สำหรับ AIX

เมื่อตั้งค่า OpenSSH สำหรับ AIX เวอร์ชัน 6.1 เอาต์พุตจะคล้าย ตัวอย่างต่อไปนี้:

```
OpenSSH has been configured with the following options:
                     User binaries: /usr/bin
                   System binaries: /usr/sbin
               Configuration files: /etc/ssh
                   Askpass program: /usr/sbin/ssh-askpass
                      Manual pages: /usr/man
                          PID file: /etc/ssh
  Privilege separation chroot path: /var/empty
            sshd default user PATH: /usr/bin:/bin:/usr/sbin:/sbin:/usr/
                                    local/bin

                    Manpage format: man
                       PAM support: yes
                   OSF SIA support: no
                  KerberosV support: yes
                  Smartcard support: no
                    SELinux support: no
                      S/KEY support: no
              TCP Wrappers support: yes
              MD5 password support: no
                   libedit support: no
  Solaris process contract support: no
           Solaris project support: no
       IP address in $DISPLAY hack: no
           Translate v4 in v6 hack: no
                 BSD Auth support: no
              Random number source: OpenSSL internal ONLY

              Host: powerpc-ibm-aix6.1.0.0
          Compiler: cc
```

```
Compiler flags: -bloadmap:file -qnostdinc -qnolm -qlist -qsource -qattr=full
Preprocessor flags: -I/gsa/ausgsa/projects/o/openssh/freeware5/openssl-0.9.8r/
            include -I/gsa/ausgsa/projects/o/openssh/zlib -I/usr/include

      Linker flags: -L/gsa/ausgsa/projects/o/openssh/freeware5/
                lib -L/gsa/ausgsa/projects/o/openssh/zlib -L/usr/include
                -Wl,-blibpath:/usr/lib:/lib
         Libraries: -lcrypto -lz -lc  -lcrypt -lefs   -lwrap -lpam -ldl
```

หมายเหตุ: อ็อพชันการคอมไพล์สำหรับ AIX เวอร์ชัน 6.1 และ AIX เวอร์ชัน 7.1 คล้ายกันเนื่องจาก ไบนารีสำหรับทั้งสองเวอร์ชันนั้นเหมือนกัน

การใช้ OpenSSH กับ Kerberos

บางการเซ็ตอัพเริ่มต้นจำเป็นต้องใช้ OpenSSH กับ Kerberos

ขั้นตอนดังต่อไปนี้จัดเตรียมข้อมูลเกี่ยวกับเซ็ตอัพ เริ่มต้นที่จำเป็นในการใช้ OpenSSH กับ Kerberos:

1. บนไคลเอ็นต์และเซิร์ฟเวอร์ OpenSSH ของคุณต้องมีไฟล์ `/etc/krb5.conf` อยู่ ไฟล์นี้ให้ข้อมูลกับ Kerberos ถึง KDC ที่จะใช้ ระยะเวลาที่จะกำหนดให้กับ แต่ละตั๋วและอื่นๆ ต่อไปนี้เป็นไฟล์ `krb5.conf` ตัวอย่าง:

```
[libdefaults]
ticket_lifetime = 600
default_realm = OPENSSH.AUSTIN.XYZ.COM
default_tkt_enctypes = des3-hmac-sha1 des-cbc-crc
default_tgs_enctypes = des3-hmac-sha1 des-cbc-crc

[realms]
OPENSSH.AUSTIN.xyz.COM = {
    kdc = kerberos.austin.xyz.com:88
    kdc = kerberos-1.austin.xyz.com:88
    kdc = kerberos-2.austin.xyz.com:88
    admin_server = kerberos.austin.xyz.com:749
    default_domain = austin.xyz.com
}

[domain_realm]
    .austin.xyz.com = OPENSSH.AUSTIN.XYZ.COM
    kdc.austin.xyz.com = OPENSSH.AUSTIN.XYZ.COM
```

2. นอกจากนี้คุณต้องเพิ่มเซอร์วิส Kerberos ดังต่อไปนี้ให้กับแต่ละไฟล์ `/etc/services` ของเครื่องไคลเอ็นต์:

```
kerberos      88/udp    kdc    # Kerberos V5 KDC
kerberos      88/tcp    kdc    # Kerberos V5 KDC
kerberos-adm  749/tcp          # Kerberos 5 admin/changepw
kerberos-adm  749/udp          # Kerberos 5 admin/changepw
krb5_prop     754/tcp          # Kerberos slave
                               #  propagation
```

3. ถ้า KDC ของคุณใช้ LDAP เป็นรีจิสทรีเพื่อเก็บข้อมูลผู้ใช้ อ่าน “โหลดโมดูลการพิสูจน์ตัวตน LDAP” ในหน้า 165 และเอกสาร Kerberos นอกจากนี้ตรวจสอบว่าได้ดำเนินการดังต่อไปนี้:
   - KDC กำลังรันไคลเอ็นต์ LDAP คุณสามารถสตาร์ท LDAP client daemon ด้วย คำสั่ง **secldapclntd**
   - เซิร์ฟเวอร์ LDAP กำลังรัน `slapd` LDAP server daemon
4. บนเซิร์ฟเวอร์ OpenSSH แก้ไขไฟล์ `/etc/ssh/sshd_config` ให้มีบรรทัดต่อไปนี้:

```
KerberosAuthentication yes
KerberosTicketCleanup yes
GSSAPIAuthentication yes
GSSAPICleanupCredentials yes
UseDNS yes
```

ถ้า UseDNS ถูกตั้งค่าเป็น **Yes** เซิร์ฟเวอร์ ssh กลับการค้นหาโฮสต์เพื่อค้นหาชื่อของไคลเอ็นต์ การเชื่อมต่อ ซึ่งจำเป็นเมื่อการพิสูจน์ตัวตนโดยมีโฮสต์ถูกใช้หรือเมื่อคุณ ต้องการข้อมูลล็อกอินล่าสุดเพื่อแสดงชื่อโฮสต์แทนที่จะเป็น IP addresses

หมายเหตุ: บางเซสชัน ssh หยุดทำงานเมื่อทำการกลับการค้นหาชื่อ เนื่องจากเซิร์ฟเวอร์ DNS ไม่สามารถติดต่อได้ ถ้าเหตุการณ์นี้เกิดขึ้น คุณสามารถข้ามการค้นหา DNS โดยตั้งค่า UseDNS เป็น `no` ถ้า UseDNS ไม่ได้ถูกตั้งค่าในไฟล์ `/etc/ssh/sshd_config` ค่าดีฟอลต์คือ `UseDNS yes`

5. บนเซิร์ฟเวอร์ SSH รันคำสั่ง **startsrc -g ssh** เพื่อสตาร์ท ssh server daemon
6. บนเครื่องไคลเอ็นต์ SSH รันคำสั่ง **kinit** เพื่อรับ credential เริ่มต้น (TGT) คุณสามารถตรวจสอบว่าคุณได้รับ TGT โดยรันคำสั่ง **klist** ซึ่งแสดง credentials ทั้งหมดที่เป็น ของคุณ
7. เชื่อมต่อไปที่เซิร์ฟเวอร์โดยรันคำสั่ง **ssh** *username@servername*
8. ถ้า Kerberos ถูกกำหนดค่าอย่างถูกต้องเพื่อพิสูจน์ตัวตนผู้ใช้ พร้อมต์สำหรับรหัสผ่าน จะไม่แสดง และผู้ใช้จะถูกล็อกอินเข้าสู่ เซิร์ฟเวอร์ SSH โดยอัตโนมัติ

การรักษาความปลอดภัยเน็ตเวิร์ก

ส่วนต่อไปนี้อธิบายวิธีติดตั้งและตั้งค่า IP Security วิธีระบุเน็ตเวิร์กเซอร์วิสที่จำเป็นและไม่จำเป็น และ การตรวจสอบและการมอนิเตอร์การรักษาความปลอดภัยเน็ตเวิร์ก

ความปลอดภัย TCP/IP

ถ้าคุณติดตั้งซอฟต์แวร์ Transmission Control Protocol/Internet Protocol (TCP/IP) และ Network File System (NFS) คุณสามารถตั้งค่าระบบของคุณ ให้สื่อสารข้ามเน็ตเวิร์กได้

ข้อมูลแนะนำนี้ไม่อธิบายแนวคิดพื้นฐานของ TCP/IP แต่อธิบายถึง เรื่องความปลอดภัยของ TCP/IP สำหรับข้อมูลเกี่ยวกับการติดตั้งและ คอนฟิกูเรชันเริ่มต้นของ TCP/IP อ้างอิงที่ส่วน Transmission Control Protocol/Internet Protocol ใน *การจัดการเครือข่ายและการสื่อสาร*

จะด้วยเหตุผลใดก็ตาม ผู้ที่ดูแลระบบของคุณ จะต้องปฏิบัติตาม ระดับการรักษาความปลอดภัยระดับใดระดับหนึ่ง ตัวอย่างระดับความปลอดภัย อาจเป็นเรื่องของนโยบายองค์กร หรือระบบอาจจำเป็นต้องเข้าถึงระบบของรัฐบาล ดังนั้นจึงจำเป็นต้องสื่อสารกันที่ระดับการรักษาความปลอดภัยในระดับหนึ่ง มาตรฐานความปลอดภัย เหล่านี้อาจถูกใช้กับเน็ตเวิร์ก ระบบปฏิบัติการ แอ็พพลิเคชันซอฟต์แวร์ แม้แต่โปรแกรมที่เขียนโดยผู้ดูแล ระบบของคุณ

ส่วนนี้อธิบายถึงคุณลักษณะการรักษาความปลอดภัยที่จัดเตรียมด้วย TCP/IP ทั้งใน โหมดมาตรฐานและในแบบระบบที่มีความปลอดภัย และพูดถึงข้อควรพิจารณาเรื่องความปลอดภัย บางข้อ ที่เกี่ยวข้องในสภาวะแวดล้อมเน็ตเวิร์ก

หลังจากที่คุณติดตั้งซอฟต์แวร์ TCP/IP และ NFS แล้ว, ให้ใช้พาธด่วน System Management Interface Tool (SMIT) **tcpip** เพื่อกำหนดคอนฟิก ระบบของคุณ

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับคำสั่ง **dacinet** อ้างอิงถึง *การอ้างอิงคำสั่ง*

การรักษาความปลอดภัยระบบปฏิบัติการ

คุณลักษณะการรักษาความปลอดภัยจำนวนมาก เช่นการควบคุมการเข้าถึงเน็ตเวิร์ก และ การตรวจสอบเน็ตเวิร์ก ที่พร้อมใช้สำหรับ TCP/IP มาจากที่มีอยู่ ในระบบปฏิบัติการ

ส่วนต่อไปนี้ให้กรอบเกี่ยวกับการรักษาความปลอดภัย TCP/IP

ค่าควบคุมการเข้าใช้เน็ตเวิร์ก:

นโยบายความปลอดภัยสำหรับเน็ตเวิร์กเป็นส่วนขยายของนโยบาย ความปลอดภัยสำหรับระบบปฏิบัติการ และประกอบด้วยการพิสูจน์ตัวตนผู้ใช้ การพิสูจน์ตัวตน การเชื่อมต่อ และความปลอดภัยของข้อมูล

ประกอบด้วยคอมโพเนนต์หลักดังต่อไปนี้:

- การพิสูจน์ตัวตนผู้ใช้ ถูกจัดเตรียมที่รีโมตโฮสต์ โดยชื่อผู้ใช้และรหัสผ่านในวิธีเดียวกับเมื่อผู้ใช้ล็อกอินเข้าสู่ ระบบโลคัล คำสั่ง Trusted TCP/IP เช่น **ftp, rexec,** และ **telnet** มีข้อกำหนดเหมือนกันและผ่านกระบวนการ ตรวจสองเหมือนกับคำสั่งที่ไว้วางใจในระบบปฏิบัติการ
- การพิสูจน์ตัวตนการเชื่อมต่อ ถูกจัดเตรียมเพื่อประกันว่า รีโมตโฮสต์มีแอดเดรสและชื่อ Internet Protocol (IP) ที่คาดไว้ ซึ่งป้องกันรีโมตโฮสต์จากการปลอมแปลงข้อมูลเป็นรีโมตโฮสต์อื่น
- การรักษาความปลอดภัยการอิมพอร์ตและเอ็กซ์พอร์ตข้อมูล อนุญาตให้ข้อมูลที่ ระดับความปลอดภัยที่ระบุส่งผ่านไปมากับอินเตอร์เฟสเน็ตเวิร์กอะแด็ปเตอร์ที่ระดับ ความปลอดภัยและสิทธิเดียวกัน ตัวอย่างเช่น ข้อมูลลับสุดยอดสามารถส่งผ่านเฉพาะ ระหว่างอะแด็ปเตอร์ที่ถูกเซ็ตเป็นระดับการรักษาความปลอดภัยลับสุดยอด

การตรวจสอบเน็ตเวิร์ก:

การตรวจสอบเครือข่ายถูกระบุโดย TCP/IP โดยใช้ระบบย่อย การตรวจสอบเพื่อตรวจอสบแอ็พพลิเคชันโปรแกรม

วัตถุประสงค์ของการตรวจสอบคือเพื่อบันทึกการดำเนินการเหล่านี้ที่มีผลต่อความปลอดภัย ของระบบและผู้ใช้ต้องรับผิดชอบต่อการดำเนินการเหล่านั้น

เหตุการณ์แอ็พพลิเคชันต่อไปนี้ถูกตรวจสอบ:

- การเข้าถึงเน็ตเวิร์ก
- การเชื่อมต่อ
- การเอ็กซ์พอร์ตข้อมูล
- การอิมพอร์ตข้อมูล

การสร้างและการลบของอ็อบเจ็กต์จะถูกตรวจสอบโดยระบบปฏิบัติการ เร็กคอร์ดการตรวจสอบแอ็พพลิเคชันจะหยุดทำงานชั่วคราวหรือทำงานต่อการตรวจสอบเพื่อหลีกเลี่ยงการตรวจสอบซ้ำซ้อนกันกับ ที่ดำเนินการโดยเคอร์เนล

พาธที่ไว้วางใจ เชลล์ที่ไว้วางใจ และ Secure Attention Key:

ระบบปฏิบัติการจัดเตรียม *พาธที่ไว้วางใจ* เพื่อป้องกัน โปรแกรมที่ไม่ได้รับอนุญาต จากการอ่านข้อมูลจากเทอร์มินัลผู้ใช้ พาธนี้ถูกใช้ เมื่อพาธการสื่อสารที่ปลอดภัยกับระบบเป็นสิ่งจำเป็น เช่น เมื่อคุณเปลี่ยนรหัสผ่านหรือการล็อกอินเข้าสู่ระบบ

ระบบปฏิบัติการยังจัดเตรียม *เชลล์ที่ไว้วางใจ* (**tsh**) ซึ่งรันเฉพาะโปรแกรมที่ไว้วางใจที่ถูกทดสอบและตรวจสอบแล้ว ว่าปลอดภัย TCP/IP สนับสนุนทั้งสองคุณลักษณะนี้ พร้อมกับ *secure attention key* (SAK) ซึ่งสร้างสภาวะแวดล้อมที่จำเป็นสำหรับการสื่อสาร ที่ปลอดภัยระหว่างคุณและระบบ โลคัล SAK พร้อมใช้งานเมื่อไรก็ตาม ที่คุณใช้ TCP/IP รีโมต SAK มีอยู่ผ่านคำสั่ง **telnet**

โลคัล SAK มีฟังก์ชันเดียวกับใน **telnet** นั้นคือ มีในโปรแกรมแอ็พพลิเคชันระบบปฏิบัติการอื่น: ซึ่งจบการทำงานกระบวนการ **telnet** และกระบวนการอื่นทั้งหมดที่เกี่ยวข้องกับเทอร์มินัลซึ่ง **telnet** รัน อยู่ ภายในโปรแกรม telnet คุณสามารถส่งการร้องขอสำหรับพาธที่ไว้วางใจไปที่ระบบรีโมตโดยใช้คำสั่ง **telnet send sak** (ขณะอยู่ในโหมดคำสั่ง **telnet**) คุณ ยังสามารถกำหนดคีย์เดี่ยวเพื่อเริ่มการร้องขอ SAK โดยใช้คำสั่ง **telnet set sak**

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ Trusted Computing Base, ดูที่ “Trusted Computing Base” ในหน้า 3

ความปลอดภัยของคำสั่ง TCP/IP

บางคำสั่งใน TCP/IP จัดเตรียมสภาวะแวดล้อมที่ปลอดภัยระหว่างการดำเนินการ คำสั่งเหล่านี้คือ **ftp, rexec** และ **telnet**

ฟังก์ชัน **ftp** จัดเตรียมความปลอดภัยระหว่างการถ่ายโอนไฟล์ คำสั่ง **rexec** จัดเตรียมสภาวะแวดล้อมที่ปลอดภัยสำหรับการรันคำสั่งบนโฮสต์ foreign ฟังก์ชัน **telnet** จัดเตรียม ความปลอดภัยสำหรับล็อกอินไปที่โฮสต์ foreign

คำสั่ง **ftp, rexec** และ **telnet** จัดเตรียม ความปลอดภัยระหว่างการดำเนินการของตัวคำสั่งเองเท่านั้น นั่นคือคำสั่งไม่ได้ตั้งค่าสภาวะแวดล้อมที่ปลอดภัยสำหรับการใช้กับคำสั่งอื่น เพื่อรักษาความปลอดภัยระบบของคุณ สำหรับการดำเนินการอื่น ให้ใช้คำสั่ง **securetcpip** คำสั่งนี้ ทำให้คุณสามารถรักษาความปลอดภัยระบบของคุณโดยการปิด daemons และแอ็พพลิเคชัน ที่ไม่ไว้ใจ และโดยการให้ตัวเลือกแก่คุณในการรักษาความปลอดภัย เน็ตเวิร์กโปรโตคอลของ IP เลเยอร์เช่นกัน

คำสั่ง **ftp, rexec, securetcpip,** และ **telnet** จัดเตรียมฟอร์มของระบบ และความปลอดภัยของข้อมูลดังต่อไปนี้:

ftp คำสั่ง **ftp** จัดเตรียมสภาวะแวดล้อมที่ปลอดภัยสำหรับการ ถ่ายโอนไฟล์ เมื่อผู้ใช้ร้องขอคำสั่ง **ftp** กับโฮสต์ foreign ผู้ใช้จะถูกพร้อมท์ของล็อกอิน ID ล็อกอิน ID ดีฟอลต์ถูกแสดง: ล็อกอิน ID ปัจจุบันของผู้ใช้บนโลคัลโฮสต์ ผู้ใช้จะได้รับพร้อมต์ ขอรหัสผ่านสำหรับรีโมตโฮสต์

กระบวนการล็อกอินอัตโนมัติค้นหา ไฟล์ `$HOME/.netrc` ของผู้ใช้โลคัลสำหรับ ID และรหัสผ่านของผู้ใช้ที่จะใช้ที่โฮสต์ foreign เพื่อความปลอดภัย สิทธิบนไฟล์ `$HOME/.netrc` ต้องถูกเซ็ตเป็น 600 (อ่านและเขียน ได้โดยเจ้าของเท่านั้น) มิฉะนั้นการล็อกอินอัตโนมัติจะล้มเหลว

หมายเหตุ: เนื่องจากการใช้ไฟล์ `.netrc` ต้องการหน่วยเก็บข้อมูลของรหัสผ่านในไฟล์ที่ไม่ถูกเข้ารหัส คุณลักษณะการล็อกอินอัตโนมัติของคำสั่ง **ftp** ไม่สามารถใช้ได้ เมื่อระบบของคุณถูกกำหนดค่าด้วยคำสั่ง **securetcpip** คุณลักษณะนี้สามารถถูกเปิดใช้อีกครั้งโดยการเอาคำสั่ง **ftp** ออกจาก tcpip stanza ในไฟล์ `/etc/security/config`

เมื่อต้องการ ใช้ฟังก์ชันการถ่ายโอนไฟล์ คำสั่ง **ftp** ต้องใช้สอง การเชื่อมต่อ TCP/IP, สำหรับ File Transfer Protocol (FTP) และ การถ่ายโอนข้อมูล การเชื่อมต่อโปรโตคอลเป็นเรื่องหลักและถูกรักษาความปลอดภัย เนื่องจากถูกสร้างขึ้นบนพอร์ตการสื่อสารที่เชื่อถือได้ การเชื่อมต่อรองจำเป็น สำหรับการถ่ายโอนข้อมูลจริง ทั้งโฮสต์โลคัลและรีโมตตรวจสองว่าอีกด้านหนึ่งของการเชื่อมต่อถูกสร้างขึ้นด้วยโฮสต์เดียวกัน กับการเชื่อมต่อหลัก ถ้าการเชื่อมต่อหลักและรองไม่ได้ถูกสร้างขึ้นจาก โฮสต์เดียวกันคำสั่ง **ftp** จะแสดง ข้อความแสดงความผิดพลาด แจ้งว่าการเชื่อมต่อข้อมูลไม่ถูกพิสูจน์ตัวตน จากนั้นจบการทำงาน การตรวจสอบของการเชื่อมต่อรองนี้ป้องกัน โฮสต์ที่สามดักจับข้อมูลที่ต้องการส่งไปยังโฮสต์อื่น

rexec คำสั่ง **rexec** จัดเตรียมสภาวะแวดล้อมที่ปลอดภัย สำหรับดำเนินคำสั่งบนโฮสต์ foreign ผู้ใช้จะได้รับพร้อมท์ ทั้งล็อกอิน ID และรหัสผ่าน

คุณลักษณะล็อกอินอัตโนมัติทำให้ คำสั่ง **rexec** ค้นหาไฟล์ `$HOME/.netrc` ของผู้ใช้โลคัล เพื่อหา ID และรหัสผ่านของผู้ใช้บนโฮสต์ foreign เพื่อความปลอดภัย สิทธิบนไฟล์ `$HOME/.netrc` ต้องถูกเซ็ตเป็น 600 (อ่านและเขียน ได้โดยเจ้าของเท่านั้น) มิฉะนั้นการล็อกอินอัตโนมัติจะล้มเหลว

หมายเหตุ: เนื่องจากการใช้ไฟล์ `.netrc` ต้องการหน่วยเก็บข้อมูลรหัสผ่านในไฟล์ที่ไม่ถูกเข้ารหัส คุณลักษณะการล็อกอินอัตโนมัติของคำสั่ง **rexec** ไม่สามารถใช้ได้ เมื่อระบบของคุณกำลังดำเนินการอย่างปลอดภัย คุณลักษณะนี้สามารถ ถูกเปิดใช้อีกครั้งโดยการเอารายการ ออกจาก tcpip stanza ในไฟล์ `/etc/security/config`

securetcpip

คำสั่ง **securetcpip** เปิดใช้งานคุณลักษณะความปลอดภัย TCP/IP การเข้าถึงคำสั่งที่ไม่ไว้วางใจถูกเอาออกจากระบบเมื่อ คำสั่งนี้ถูกใช้ แต่ละคำสั่งดังต่อไปนี้ถูกเอาออกโดย การรันคำสั่ง **securetcpip**:

- **rlogin** และ **rlogind**
- **rcp**, **rsh**, และ **rshd**
- **tftp** และ **tftpd**
- **trpt**

คำสั่ง **securetcpip** ถูกใช้เพื่อแปลง ระบบจากระดับมาตรฐานของความปลอดภัยไปเป็นระดับความปลอดภัยที่สูงกว่า หลังจาก ระบบของคุณได้ถูกแปล คุณไม่จำเป็นต้องใช้คำสั่ง **securetcpip** อีกครั้งนอกจากคุณติดตั้ง TCP/IP ใหม่

telnet หรือ tn

คำสั่ง **telnet** (TELNET) จัดเตรียมสภาวะแวดล้อมที่มี การรักษาความปลอดภัยสำหรับล็อกอินสู่โฮสต์ foreign ผู้ใช้จะได้รับพร้อมท์ ทั้งล็อกอิน ID และรหัสผ่าน เทอร์มินัลของผู้ใช้จะถูกปฏิบัติเหมือนเทอร์มินัลที่เชื่อมต่อ โดยตรงกับโฮสต์ นั่นคือ การเข้าถึงเทอร์มินัลถูกควบคุมโดยบิตด สิทธิ ผู้ใช้อื่น (กลุ่มหรืออื่นๆ) ไม่มีสิทธิอ่านกับเทอร์มินัล แต่สามารถเขียนความส่งไปที่เทอร์มินัล ถ้าเจ้าของให้สิทธิ เขียน คำสั่ง **telnet** ยังจัดเตรียมการเข้าถึงเชลล์ที่ไว้วางใจ บนระบบรีโมตผ่าน SAK ลำดับคีย์นี้ต่างจากลำดับ ที่เรียกพาธที่ไว้วางใจโลคัลและสามารถถูกกำหนดภายในคำสั่ง **telnet**

การเข้าถึงเพื่อทำงานคำสั่งรีโมต:

ผู้ใช้บนโฮสต์ที่แสดงในไฟล์ `/etc/hosts.equiv` สามารถรันบางคำสั่งบนระบบของคุณโดยไม่ต้องใช้รหัสผ่าน

ตารางต่อไปนี้จัดเตรียมข้อมูลเกี่ยวกับวิธีการแสดง, เพิ่ม, และย้ายโฮสต์แบบรีโมตโดยใช้อินเตอร์เฟส SMIT หรืออินเตอร์เฟสบรรทัดรับคำสั่ง

ตารางที่ 13. งานการเข้าถึงเพื่อทำงาน คำสั่งรีโมต

งาน	พาธด่วน SMIT	คำสั่งหรือไฟล์
แสดงรายการรีโมตโฮสต์ที่มีการเข้าถึงเพื่อทำงานคำสั่ง	**smit lshostsequiv**	ดูไฟล์ `/etc/hosts.equiv`
เพิ่มรีโมตโฮสต์สำหรับการเข้าถึงเพื่อทำงานคำสั่ง	**smit mkhostsequiv**	แก้ไขไฟล์ `/etc/hosts.equiv` หมายเหตุ
ลบรีโมตโฮสต์ออกจากการเข้าถึงเพื่อทำงานคำสั่ง	**smit rmhostsequiv**	แก้ไขไฟล์ `/etc/hosts.equiv` หมายเหตุ

หมายเหตุ: สำหรับข้อมูลเพิ่มเติมเกี่ยวกับโพรซีเดอร์ไฟล์เหล่านี้ ดูที่ "hosts.equiv File Format for TCP/IP" ใน *การอ้างอิงไฟล์*

ผู้ใช้ที่จำกัดการถ่ายโอนไฟล์:

ผู้ใช้ที่แสดงรายการในไฟล์ /etc/ftpusers ได้รับการป้องกันจากการเข้าถึง FTP รีโมต ตัวอย่าง สมมติผู้ใช้ A ถูกล็อกอิน เข้าสู่ระบบรีโมต และผู้ใช้ทราบรหัสผ่านของผู้ใช้ B บนระบบของคุณ ถ้าผู้ใช้ B ถูกแสดงรายการในไฟล์ /etc/ftpusers ผู้ใช้ A ไม่สามารถ FTP ไฟล์ไปยัง หรือจากบัญชีผู้ใช้ของผู้ใช้ B แม้ว่าผู้ใช้ A จะทราบ รหัสผ่านของผู้ใช้ B

ตารางต่อไปนี้จัดเตรียมข้อมูลเกี่ยวกับวิธีการแสดง, เพิ่ม, และลบผู้ใช้ที่จำกัดไว้โดยใช้ SMIT หรือบรรทัดรับคำสั่ง

งานผู้ใช้รีโมต FTP

งาน	พาธด่วน SMIT	คำสั่งหรือไฟล์
แสดงรายการผู้ใช้ที่จำกัด FTP	**smit lsftpusers**	ดูไฟล์ /etc/ftpusers
เพิ่มผู้ใช้ที่จำกัด	**smit mkftpusers**	แก้ไขไฟล์ /etc/ftpusers หมายเหตุ
ลบผู้ใช้ที่จำกัดออก	**smit rmftpusers**	แก้ไขไฟล์ /etc/ftpusers หมายเหตุ

หมายเหตุ: สำหรับข้อมูลเพิ่มเติมเกี่ยวกับโพรซีเดอร์ไฟล์เหล่านี้ ดูที่ "ftpusers File Format for TCP/IP" ใน *การอ้างอิงไฟล์*

กระบวนการที่ไว้วางใจ

โปรแกรมที่ไว้วางใจ หรือกระบวนการที่ไว้วางใจ เป็นเชลล์สคริปต์ daemon หรือโปรแกรมที่ตรงตามมาตรฐานความปลอดภัย มาตรฐานความปลอดภัยเหล่านี้ ถูกกำหนดและดูแลโดย U.S. Department of Defense ซึ่งยัง ให้การรับรองโปรแกรมที่ไว้วางใจบางโปรแกรม

โปรแกรมที่ไว้วางใจได้รับการไว้วางใจที่ระดับต่างกัน ระดับความปลอดภัยประกอบด้วย A1, B1, B2, B3, C1, C2 และ D โดยที่ระดับ A1 ให้ระดับการรักษาความปลอดภัย สูงสุด แต่ละระดับความปลอดภัยต้องตรงตามข้อกำหนด ตัวอย่างเช่น ระดับ C2 ของการรักษาความปลอดภัยประกอบด้วยมาตรฐานดังต่อไปนี้:

ความสมบูรณ์ของโปรแกรม
ประกันว่ากระบวนการทำงานตามที่กำหนดไว้อย่างแม่นยำ

modularity
ซอร์สโค้ดของกระบวนการถูกแยกเป็นโมดูลซึ่งจะไม่ได้รับผลกระทบ หรือเข้าถึง โดยตรงจากโมดูลอื่น

หลักการ privilege ที่น้อยที่สุด
กำหนดว่าผู้ใช้ดำเนินการที่ระดับ privilege ต่ำที่สุดที่ได้รับอนุญาตตลอดเวลา นั้นคือ ถ้าผู้ใช้มีการเข้าถึงเพียงการดูไฟล์ ผู้ใช้จะไม่มีการเข้าถึงเพื่อเปลี่ยนแปลงไฟล์นั้น โดยไม่ได้ตั้งใจ

การจำกัดการนำอ็อบเจ็กต์กลับมาใช้
ตัวอย่างเช่น ป้องกันผู้ใช้จาก ค้นหาส่วนของหน่วยความจำ ที่ถูกแฟล็กสำหรับการเขียนทับแต่ยังไม่ได้ถูกเคลียร์โดยไม่ได้ตั้งใจ ซึ่งอาจมี ข้อมูลสำคัญ

TCP/IP มี daemons ที่ไว้วางใจจำนวนหนึ่งและ daemons ที่ไม่ไว้วางใจจำนวนมาก

ตัวอย่างของ daemons ที่ไว้วางใจมีดังนี้:

- **ftpd**
- **rexecd**
- **telnetd**

ตัวอย่างของ daemons ที่ไม่ไว้วางใจมีดังนี้:

- **rshd**
- **rlogind**
- **tftpd**

สำหรับระบบที่จะได้รับการไว้วางใจ ระบบต้องทำงานกับ trusted computing base นั้นคือสำหรับโฮสต์เดี่ยว เครื่องต้องถูกรักษาความปลอดภัย สำหรับเน็ตเวิร์ก ไฟล์เซิร์ฟเวอร์ เกตเวย์ และโฮสต์อื่น ทั้งหมดต้องถูกรักษาความปลอดภัย

Network Trusted Computing Base

The Network Trusted Computing Base (NTCB) ประกอบด้วยฮาร์ดแวร์ และซอฟต์แวร์สำหรรับการรับประกันความปลอดภัยของเน็ตเวิร์ก ส่วนนี้กำหนดคอมโพเนนต์ ของ NTCB ตามความสัมพันธ์กับ TCP/IP

คุณลักษณะความปลอดภัยฮาร์ดแวร์สำหรับเน็ตเวิร์กถูกจัดเตรียม โดยเน็ตเวิร์กอะแด็ปเตอร์ที่ใช้กับ TCP/IP อะแด็ปเตอร์เหล่านี้ควบคุมข้อมูลที่เข้ามา โดยรับเฉพาะข้อมูลที่เป้าหมายอยู่ที่ระบบโลคัลและกระจาย ข้อมูลที่รับได้โดยระบบทั้งหมด

คอมโพเนนต์ซอฟต์แวร์ของ NTCB ประกอบด้วยโปรแกรมที่ ไว้วางใจเท่านั้น โปรแกรมและไฟล์ที่เชื่อมโยงที่เป็นส่วนหนึ่งของ ระบบที่ปลอดภัยถูกแสดงในตารางดังต่อไปนี้ในแบบ ไดเรกทอรีต่อไดเรกทอรี

ไดเร็กทอรี /etc

ชื่อ	เจ้าของ	กลุ่ม	โหมด	สิทธิ
gated.conf	root	system	0664	rw-rw-r—
gateways	root	system	0664	rw-rw-r—
hosts	root	system	0664	rw-rw-r—
hosts.equiv	root	system	0664	rw-rw-r—
inetd.conf	root	system	0644	rw-r—r—
named.conf	root	system	0644	rw-r—r—
named.data	root	system	0664	rw-rw-r—
networks	root	system	0664	rw-rw-r—
protocols	root	system	0644	rw-r—r—
rc.tcpip	root	system	0774	rwxrwxr—
resolv.conf	root	system	0644	rw-rw-r—
services	root	system	0644	rw-r—r—
3270.keys	root	system	0664	rw-rw-r—
3270keys.rt	root	system	0664	rw-rw-r—

ไดเร็กทอรี /usr/bin

ชื่อ	เจ้าของ	กลุ่ม	โหมด	สิทธิ
host	root	system	4555	r-sr-xr-x
hostid	bin	bin	0555	r-xr-xr-x
hostname	bin	bin	0555	r-xr-xr-x
finger	root	system	0755	rwxr-xr-x
ftp	root	system	4555	r-sr-xr-x
netstat	root	bin	4555	r-sr-xr-x
rexec	root	bin	4555	r-sr-xr-x
ruptime	root	system	4555	r-sr-xr-x
rwho	root	system	4555	r-sr-xr-x
talk	bin	bin	0555	r-xr-xr-x
telnet	root	system	4555	r-sr-xr-x

ไดเร็กทอรี /usr/sbin

ชื่อ	เจ้าของ	กลุ่ม	โหมด	สิทธิ
arp	root	system	4555	r-sr-xr-x
fingerd	root	system	0554	r-xr-xr—
ftpd	root	system	4554	r-sr-xr—
gated	root	system	4554	r-sr-xr—
ifconfig	bin	bin	0555	r-xr-xr-x
inetd	root	system	4554	r-sr-xr—
named	root	system	4554	r-sr-x—
ping	root	system	4555	r-sr-xr-x
rexecd	root	system	4554	r-sr-xr—
route	root	system	4554	r-sr-xr—
routed	root	system	0554	r-xr-x—-
rwhod	root	system	4554	r-sr-xr—
securetcpip	root	system	0554	r-xr-xr—
setclock	root	system	4555	r-sr-xr-x
syslogd	root	system	0554	r-xr-xr—
talkd	root	system	4554	r-sr-xr—
telnetd	root	system	4554	r-sr-xr—

ไดเร็กทอรี /usr/ucb

ชื่อ	เจ้าของ	กลุ่ม	โหมด	สิทธิ
tn	root	system	4555	r-sr-xr-x

ไดเร็กทอรี /var/spool/rwho

ชื่อ	เจ้าของ	กลุ่ม	โหมด	สิทธิ
rwho (directory)	root	system	0755	drwxr-xr-x

ความปลอดภัยของข้อมูลและการปกป้องข้อมูล

คุณลักษณะความปลอดภัยสำหรับ TCP/IP ไม่ได้เข้ารหัสข้อมูลผู้ใช้ที่ส่ง ผ่านเน็ตเวิร์ก

ระบุความเสี่ยงที่มีในการสื่อสารซึ่งอาจมีผลในการเปิดเผย รหัสผ่านและข้อมูลสำคัญอื่น และจากข้อมูลความเสี่ยงดังกล่าว ให้ใช้วิธีป้องกันที่เหมาะสม

การใช้คุณลักษณะความปลอดภัย TCP/IP ในสภาวะแวดล้อม Department of Defense (DOD) environment อาจต้องการการปฏิบัติตาม DOD 5200.5 และ NCSD-11 สำหรับการรักษาความปลอดภัยการสื่อสาร

ค่าควบคุมการเข้าใช้พอร์ต TCP ตามผู้ใช้โดยใช้ discretionary access control for internet ports

Discretionary Access Control for Internet Ports (DACinet) มีคุณลักษณะการควบคุมการเข้าถึงสำหรับพอร์ต TCP สำหรับการสื่อสาร ระหว่างโฮสต์ AIX

AIX สามารถใช้ส่วนหัว TCP เพิ่มเติมเพื่อส่งข้อมูลผู้ใช้และกลุ่มระหว่างระบบ คุณลักษณะ DACinet อนุญาตให้ผู้ดูแลระบบบนระบบปลายทาง ควบคุมการเข้าถึงบนพอร์ตปลายทาง ด้วย ID ผู้ใช้ และโฮสต์ ต้นทาง

นอกจากนั้น คุณลักษณะ DACinet ยังอนุญาตให้ผู้ดูแลระบบจำกัด พอร์ตโลคัลสำหรับ root ใช้งานเท่านั้น ระบบ UNIX อย่างเช่น AIX ถือว่าพอร์ตที่ต่ำกว่า 1024 เป็นพอร์สที่มีสิทธิพิเศษซึ่งสามารถเปิดใช้ได้โดย root เท่านั้น AIX อนุญาตให้คุณระบุพอร์ตเพิ่มเติมมากกว่า 1024 ซึ่งสามารถเปิดได้เฉพาะ root เท่านั้น, ดังนั้นจึงป้องกัน ผู้ใช้จากการรันเซิร์ฟเวอร์บนพอร์ตที่รู้จักกันดี

โดยขึ้นอยู่กับการตั้งค่าระบบที่มิใช่ DACinet ที่อาจจะ หรืออาจจะไม่ สามารถเชื่อมต่อกับระบบ DACinet การเข้าถึงจะถูกปฏิเสธในขั้นเริ่มเตรียมค่า ของคุณลักษณะ DACinet เมื่อ DACinet ถูกเปิดใช้งานแล้ว จะไม่ มีวิธีปิดใช้งาน DACinet

คำสั่ง **dacinet** ยอมรับแอดเดรสซึ่ง ถูกระบุเป็นชื่อโฮสต์ เป็นโฮสต์แอดเดรสแบบจุดทศนิยม หรือเน็ตเวิร์กแอดเดรส ตามด้วยความยาวส่วนนำหน้าเน็ตเวิร์ก

ตัวอย่างต่อไปนี้ระบุโฮสต์เดียวซึ่งรู้จักโดย ใช้ชื่อโฮสต์แบบเต็ม *host.domain.org*:

```
host.domain.org
```

ตัวอย่างต่อไปนี้ระบุโฮสต์เดียวซึ่งรู้จักโดย IP แอดเดรส 10.0.0.1:

```
10.0.0.1
```

ตัวอย่างต่อไปนี้ระบุทั้งเน็ตเวิร์กซึ่งมี 24 บิตแรก (ความยาวของส่วนนำหน้าเน็ตเวิร์ก) ที่มีค่าเป็น 10.0.0.0:

```
10.0.0.0/24
```

เน็ตเวิร์ก นี้รวม IP addresses ทั้งหมดตั้งแต่ 10.0.0.1 ถึง 10.0.0.254

ค่าควบคุมการเข้าใช้สำหรับเซอร์วิสบน TCP:

DACinet ใช้ไฟล์เริ่มทำงาน /etc/rc.dacinet และไฟล์คอนฟิกูเรชันที่ใช้คือ /etc/security/priv, /etc/security/services และ /etc/security/acl

พอร์ตที่แสดงใน /etc/security/services ถูก พิจารณาให้ยกเว้นจากการตรวจสอบ ACL ไฟล์มีรูปแบบเดียวกับ /etc/services วิธีง่ายที่สุดในการเตรียมข้อมูลเบื้องต้นคือทำสำเนาไฟล์จาก /etc ไปยัง /etc/security และ จากนั้นลบพอร์ตทั้งหมดที่ ACLs ควรถูกนำไปใช้ ACLs ถูกเก็บ ไว้สองที่ ACLs ที่แอ็คทีฟขณะนี้ถูกเก็บในเคอร์เนลและ สามารถอ่านได้โดยรัน dacinet aclls ACLs ที่จะถูกเปิดทำงานใหม่ ในการบูตระบบครั้งถัดไปโดย /etc/rc.tcpip ถูกเก็บใน /etc/security/acl โดยใช้รูปแบบต่อไปนี้:

```
service host/prefix-length [user|group]
```

โดยที่เซอร์วิสสามารถเป็นแบบตัวเลข หรือดังแสดงใน /etc/services โฮสต์สามารถกำหนดเป็นชื่อโฮสต์ หรือเน็ตเวิร์กแอดเดรสที่มีค่ากำหนด subnet mask และผู้ใช้หรือกลุ่มถูกระบุด้วยส่วนนำหน้า u: or g: เมื่อไม่มีการระบุผู้ใช้หรือกลุ่ม ACL จะพิจารณาเฉพาะโฮสต์ การส่ง การนำหน้าเซอร์วิสด้วย – จะปิด การเข้าถึงโดยชัดเจน ACLs ถูกประเมินตามการจับคู่ตรงกันในครั้งแรก ดังนั้นคุณ สามารถระบุการเข้าถึงสำหรับกลุ่มของผู้ใช้ แต่ปฏิเสธอย่างชัดแจ้งสำหรับผู้ใช้ในกลุ่มโดยการวางกฎสำหรับผู้ใช้นี้หน้ากฎกลุ่ม

ไฟล์ /etc/services สอดแทรกสองรายการ ด้วยค่าหมายเลขพอร์ตซึ่งไม่สนับสนุนใน AIX ผู้ดูแลระบบต้องลบทั้งสองบรรทัดออกจากไฟล์นั้นก่อนทำงาน คำสั่ง **mkCCadmin** ลบบรรทัดต่อไปนี้ออกจาก ไฟล์ /etc/services:

```
sco_printer     70000/tcp     sco_spooler    # For System V print IPC
sco_s5_port     70001/tcp     lpNet_s5_port  # For future use
```

ตัวอย่างการใช้งาน DACinet:

ตัวอย่างเช่น, เมื่อใช้ DACinet เพื่อจำกัดการเข้าถึงพอร์ต TCP/25 ขาเข้าไปยัง root ด้วยคุณลักษณะ DACinet เท่านั้น, เฉพาะผู้ใช้ root เท่านั้น จากโฮสต์ AIX อื่นๆ สามารถเข้าถึงพอร์ตนี้, ดังนั้น การจำกัดความเป็นไปได้ของผู้ใช้ปกติ กับอีเมลหลอกโดย telnet ไปยังพอร์ต TCP/25 บนเครื่องที่เสียหาย

ตัวอย่างต่อไปนี้แสดงวิธีตั้งค่า โปรโตคอล X (X11) สำหรับการเข้าถึงโดย root เท่านั้น ตรวจให้แน่ใจว่ารายการ X11 ใน /etc/security/services ถูก ลบออกแล้วเพื่อที่ ACLs จะนำใช้สำหรับเซอร์วิสนี้

โดยการสมมติซับเน็ตของ 10.1.1.0/24 สำหรับระบบที่เชื่อมต่อทั้งหมด รายการ ACL ที่จะจำกัดการเข้าถึงให้แก่ผู้ใช้ root เท่านั้นสำหรับ X (TCP/6000) ใน /etc/security/acl จะเป็นดังนี้:

```
      6000    10.1.1.0/24 u:root
```

เมื่อจำกัดเซอร์วิส Telnet ให้เฉพาะผู้ใช้ในกลุ่ม friends ไม่ว่าจะมาจากระบบใด ให้ใช้รายการ ACL ต่อไปนี้ หลังจากลบรายการ telnet ออกจาก /etc/security/services:

```
telnet    0.0.0.0/0   g:friends
```

ไม่อนุญาตผู้ใช้ที่มีการเข้าถึง fred ไปยังเว็บเซิร์ฟเวอร์ แต่อนุญาตทุกคน เข้าถึงอย่างอื่น:

```
-80     0.0.0.0/0 u:fred
80      0.0.0.0/0
```

พอร์ตสิทธิพิเศษสำหรับการทำงานโลคัลเซอร์วิส:

เพื่อป้องกันมิให้ผู้ใช้ทั่วไปทำงานเซิร์ฟเวอร์ที่พอร์ต ที่เจาะจง พอร์ตเหล่านี้สามารถกำหนดให้เป็นสิทธิพิเศษ

โดยปกติผู้ใช้ใดๆ จะสามารถเปิดพอร์ตใดๆ ที่เหนือกว่า 1024 ตัวอย่าง ผู้ใช้สามารถกำหนดเซิร์ฟเวอร์ที่พอร์ต 8080 ซึ่งมักใช้บ่อย เพื่อรันเว็บพร็อกซี่หรือที่ 1080 ที่ซึ่งปกติจะพบ SOCKS server คำสั่ง **dacinet setpriv** สามารถใช้เพื่อเพิ่ม พอร์ตสิทธิพิเศษแก่ระบบที่กำลังทำงาน พอร์ตที่ถูกกำหนด เป็นสิทธิพิเศษเมื่อระบบเริ่มทำงานได้ถูกแสดงใน `/etc/security/priv`

พอร์ตสามารถแสดงรายการในไฟล์นี้โดยใช้ชื่อสัญลักษณ์ ตามที่กำหนดใน `/etc/services` หรือโดยการระบุ หมายเลขพอร์ต รายการต่อไปนี้จะไม่อนุญาตให้ผู้ใช้ที่มิใช่ root รัน SOCKS servers หรือเซิร์ฟเวอร์ Lotus Notes® บนพอร์ตปกติ:

```
1080
lotusnote
```

หมายเหตุ: คุณลักษณะนี้มิได้ป้องกัน ผู้ใช้ในการทำงานโปรแกรม จะป้องกันผู้ใช้มิให้ทำงาน เซอร์วิสที่พอร์ตที่รู้จักดีที่โดยทั่วไปเซอร์วิสเหล่านั้นถูกคาด ว่ามีเท่านั้น

เน็ตเวิร์กเซอร์วิส

ข้อมูลเกี่ยวกับการระบุและการรักษาความปลอดภัยเน็ตเวิร์กเซอร์วิสที่มี พอร์ตการสื่อสารที่เปิดถูกแสดง

การใช้งานพอร์ต

ตารางต่อไปนี้กล่าวถึงการใช้งานพอร์ตที่รู้จักบนระบบปฏิบัติการ AIX

หมายเหตุ: รายการนี้สร้างขึ้นโดยตรวจทานระบบ AIX จำนวนมากด้วยคอนฟิกูเรชันอื่นของ ซอฟต์แวร์ที่ติดตั้ง

รายการต่อไปนี้อาจไม่รวมการใช้งานพอร์ตสำหรับซอฟต์แวร์ทั้งหมด ที่มีอยู่บนระบบปฏิบัติการ AIX :

Port/Protocol	ServiceName	Aliases
13/tcp	daytime	Daytime (RFC 867)
13/udp	daytime	Daytime (RFC 867)
21/tcp	ftp	File Transfer [Control]
21/udp	ftp	File Transfer [Control]
23/udp	telnet	Telnet
23/udp	telnet	Telnet
25/tcp	smtp	Simple Mail Transfer
25/udp	smtp	Simple Mail Transfer
37/tcp	time	Time
37/udp	time	Time
111/tcp	sunrpc	SUN Remote Procedure Call
111/udp	sunrpc	SUN Remote Procedure Call
161/tcp	snmp	SNMP

Port/Protocol	ServiceName	Aliases
161/udp	snmp	SNMP
199/tcp	smux	SMUX
199/udp	smux	SMUX
512/tcp	exec	remote process execution;
513/tcp	login	remote login a la telnet;
514/tcp	shell	cmd
514/udp	syslog	Syslog
518/tcp	ntalk	Talk
518/udp	ntalk	Talk
657/tcp	rmc	RMC
657/udp	rmc	RMC
1334/tcp	writesrv	writesrv
1334/udp	writesrv	writesrv
2279/tcp	xmquery	xmquery
2279/udp	xmquery	xmquery
32768/tcp	filenet-tms	FileNet® TMS
32768/udp	filenet-tms	FileNet TMS
32769/tcp	filenet-rpc	FileNet RPC
32769/udp	filenet-rpc	FileNet RPC
32770/tcp	filenet-nch	FileNet NCH
32770/udp	filenet-nch	FileNet NCH
32771/tcp	filenet-rmi	FileNet RMI
32771/udp	filenet-rmi	FileNet RMI
32772/tcp	filenet-pa	FileNet Process Analyzer
32772/udp	filenet-pa	FileNet Process Analyzer
32773/tcp	filenet-cm	FileNet Component Manager
32773/udp	filenet-cm	FileNet Component Manager
32774/tcp	filenet-re	FileNet Rules Engine
32774/udp	filenet-re FileNET Rules Engine	FileNet Rules Engine
32775/tcp	filenet-pch	Performance Clearinghouse
32775/udp	filenet-pch	Performance Clearinghouse
32776/tcp	filenet-peior	FileNet BPM IOR

Port/Protocol	ServiceName	Aliases
32776/udp	filenet-peior	FileNet BPM IOR
32777/tcp	filenet-obrok	FileNet BPM CORBA
32777/udp	filenet-obrok	FileNet BPM CORBA

การระบุเน็ตเวิร์กเซอร์วิสด้วยพอร์ตการสื่อสารที่เปิด

แอ็พพลิเคชันไคลเอ็นต์-เซิร์ฟเวอร์เปิดพอร์ตการสื่อสารบนเซิร์ฟเวอร์โดยอนุญาตให้แอ็พพลิเคชันรอฟังการร้องขอไคลเอ็นต์ขาเข้า

เนื่องจากพอร์ตที่เปิดอาจเป็นช่องโหวของการโจมตีด้านความปลอดภัยได้ ให้ระบุแอ็พพลิเคชันใดมีพอร์ตที่เปิด และปิดพอร์ตที่ถูกเปิด โดยไม่จำเป็น การปฏิบัติเช่นนี้เป็นประโยชน์เนื่องจากช่วยให้คุณเข้าใจ สิ่งที่ระบบทำให้พร้อมใช้ได้สำหรับบุคคลที่มีการเข้าถึงอินเตอร์เน็ต

ในการ พิจารณาว่าพอร์ตใดถูกเปิด ดำเนินขั้นตอนเหล่านี้:

1. ระบุเซอร์วิสโดยการใช้คำสั่ง **netstat** ดังนี้:

   ```
   # netstat -af inet
   ```

   ต่อไปนี้เป็น ตัวอย่างของเอาต์พุตคำสั่งนี้ คอลัมน์สุดท้ายของเอาต์พุตคำสั่ง **netstat** บ่งชี้สถานะของแต่ละเซอร์วิส เซอร์วิสที่กำลังรอ การเชื่อมต่อที่จะมีเข้ามาจะอยู่ในสถานะ LISTEN

นี้คือตัวอย่างของเอาต์พุตคำสั่งเมื่อรันคำสั่ง **netstat**

การเชื่อมต่อ อินเตอร์เน็ตที่แอ็คทีฟ (รวมถึงเซิร์ฟเวอร์)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	(สถานะ)
tcp4	0	0	*.echo	*.*	LISTEN
tcp4	0	0	*.discard	*.*	LISTEN
tcp4	0	0	*.daytime	*.*	LISTEN
tcp	0	0	*.chargen	*.*	LISTEN
tcp	0	0	*.ftp	*.*	LISTEN
tcp4	0	0	*.telnet	*.*	LISTEN
tcp4	0	0	*.smtp	*.*	LISTEN
tcp4	0	0	*.time	*.*	LISTEN
tcp4	0	0	*.www	*.*	LISTEN
tcp4	0	0	*.sunrpc	*.*	LISTEN
tcp	0	0	*.smux	*.*	LISTEN
tcp	0	0	*.exec	*.*	LISTEN

นี้คือตัวอย่างของเอาต์พุตคำสั่งเมื่อรันคำสั่ง **netstat**

การเชื่อมต่อ อินเตอร์เน็ตที่แอ็คทีฟ (รวมถึงเซิร์ฟเวอร์)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	(สถานะ)
tcp	0	0	*.login	*.*	LISTEN
tcp4	0	0	*.shell	*.*	LISTEN
tcp4	0	0	*.klogin	*.*	LISTEN
udp4	0	0	*.kshell	*.*	LISTEN
udp4	0	0	*.echo	*.*	
udp4	0	0	*.discard	*.*	
udp4	0	0	*.daytime	*.*	
udp4	0	0	*.chargen	*.*	
udp4	0	0	*.time	*.*	
udp4	0	0	*.bootpc	*.*	
udp4	0	0	*.sunrpc	*.*	
udp4	0	0	255.255.255.255.ntp	*.*	
udp4	0	0	1.23.123.234.ntp	*.*	
udp4	0	0	localhost.domain.ntp	*.*	
udp4	0	0	name.domain..ntp	*.*	

.....

2. เปิดไฟล์ /etc/services และ ตรวจสอบเซอร์วิส Internet Assigned Numbers Authority (IANA) เพื่อแม็พเซอร์วิสกับหมายเลขพอร์ตภายในระบบปฏิบัติการ

   ต่อไปนี้คือแฟรกเมนต์ ตัวอย่างของไฟล์ /etc/services:

   ```
   tcpmux  1/tcp  # TCP Port Service Multiplexer
   tcpmux  1/tcp  # TCP Port Service Multiplexer
   Compressnet  2/tcp  # Management Utility
   Compressnet  2/udp  # Management Utility
   Compressnet  3/tcp  # Compression Process
   Compressnet  3/udp  Compression Process
   Echo  7/tcp
   Echo  7/udp
   discard  9/tcp  sink null
   discard  9/udp  sink null
   ..............
   rfe  5002/tcp  # Radio Free Ethernet
   rfe  5002/udp  # Radio Free Ethernet
   rmonitor_secure  5145/tcp
   rmonitor_secure  5145/udp
   ```

```
pad12sim  5236/tcp
pad12sim  5236/udp
sub-process  6111/tcp  # HP SoftBench Sub-Process Cntl.
sub-process  6111/udp  # HP SoftBench Sub-Process Cntl.
xdsxdm  6558/ucp
xdsxdm  6558/tcp
afs3-fileserver  7000/tcp  # File Server Itself
afs3-fileserver  7000/udp  # File Server Itself
af3-callback  7001/tcp  # Callbacks to Cache Managers
af3-callback  7001/udp  # Callbacks to Cache Managers
```

3. ปิดพอร์ตที่ไม่จำเป็นโดยการลบเซอร์วิสที่กำลังทำงานอยู่ออก

หมายเหตุ: พอร์ต 657 ถูกใช้โดย Resource Monitoring and Control (RMC) เพื่อ การสื่อสารระหว่างโหนด คุณไม่สามารถบล็อกหรือมิฉะนั้นจำกัดพอร์ตนี้

การระบุซ็อกเก็ต TCP และ UDP

ใช้คำสั่ง **lsof** ตัวแปรของคำสั่ง **netstat -af** เพื่อระบุซ็อกเก็ต TCP ที่อยู่ในสถานะ LISTEN และซ็อกเก็ต UDP ที่ไม่ได้ทำงานที่กำลังรอข้อมูลเข้ามา

ตัวอย่าง ในการแสดงซ็อกเก็ต TCP ในสถานะ LISTEN และซ็อกเก็ต UDP ในสถานะ IDLE รันคำสั่ง **lsof** ดังนี้:

```
# lsof -i | egrep "COMMAND|LISTEN|UDP"
```

เอาต์พุต ที่เกิดขึ้นคล้ายกับเอาต์พุตต่อไปนี้:

คำสั่ง	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
dtlogin	2122	root	5u	IPv4	0x70053c00	0t0	UDP	*:xdmcp
dtlogin	2122	root	6u	IPv4	0x70054adc	0t0	TCP	*:32768(LISTEN)
syslogd	2730	root	4u	IPv4	0x70053600	0t0	UDP	*:syslog
X	2880	root	6u	IPv4	0x70054adc	0t0	TCP	*:32768(LISTEN)
X	2880	root	8u	IPv4	0x700546dc	0t0	TCP	*:6000(LISTEN)
dtlogin	3882	root	6u	IPv4	0x70054adc	0t0	TCP	*:32768(LISTEN)
glbd	4154	root	4u	IPv4	0x7003f300	0t0	UDP	*:32803
glbd	4154	root	9u	IPv4	0x7003f700	0t0	UDP	*:32805
dtgreet	4656	root	6u	IPv4	0x70054adc	0t0	TCP	*:32768(LISTEN)

หลังจากระบุ ID กระบวนการ คุณสามารถหาข้อมูลเพิ่มเติม เกี่ยวกับโปรแกรมได้โดยการรันคำสั่งต่อไปนี้:

```
" # ps -fp PID#"
```

เอาต์พุต มีพาธไปยังชื่อคำสั่ง ซึ่งคุณสามารถใช้เข้าถึง หน้าหลักของโปรแกรม

การรักษาความปลอดภัย Internet Protocol

IP Security ช่วยให้มีการสื่อสารแบบปลอดภัยบนอินเตอร์เน็ตและ ภายในเน็ตเวิร์กภายในบริษัทโดยการปกป้องการรับส่งข้อมูลที่เลเยอร์ IP

ภาพรวมการรักษาความปลอดภัย IP

การรักษาความปลอดภัย IP อนุญาตให้ผู้ใช้แต่ละคนหรือแต่ละองค์กรรักษาความปลอดภัย การรับส่งข้อมูลสำหรับแอ็พพลิเคชันทั้งหมด โดยไม่ต้องทำการแก้ไข ใดๆ ในแอ็พพลิเคชัน ดังนั้น การส่งข้อมูลใดๆ เช่นอีเมล หรือข้อมูลบริษัทที่เจาะจงสำหรับการใช้ สามารถทำให้มีความปลอดภัยได้

การรักษาความปลอดภัย IP และระบบปฏิบัติการ:

ระบบปฏิบัติการใช้ IP Security (IPsec) ซึ่งเป็น เทคโนโลยีการรักษาความปลอดภัยมาตรฐานแบบเปิดที่พัฒนาโดย Internet Engineering Task Force (IETF)

IPsec จัดให้มีการป้องกันโดยใช้การเข้ารหัสข้อมูลทั้งหมดที่ เลเยอร์ IP ของสแต็กการสื่อสาร ไม่จำเป็นต้องมีการเปลี่ยนแปลงสำหรับ แอ็พพลิเคชันที่มีอยู่แล้ว IPsec เป็นเฟรมเวิร์กการรักษาความปลอดภัยเน็ตเวิร์กมาตรฐานอุตสาหกรรม ที่เลือกโดย IETF สำหรับสภาวะแวดล้อมทั้ง IP Version 4 และ 6

IPsec ปกป้องการรับส่งข้อมูลของคุณโดยใช้เทคนิคการเข้ารหัสลับ ต่อไปนี้:

การพิสูจน์ตัวตน
: กระบวนการที่ identity ของโฮสต์หรือจุดหมายถูกตรวจสอบความถูกต้อง

การตรวจสอบ Integrity
: กระบวนการของการทำให้แน่ใจว่าไม่มีการแก้ไขใดในข้อมูล ขณะส่งผ่านเน็ตเวิร์ก

การเข้ารหัส
: กระบวนการของการทำให้แน่ใจในความเป็นส่วนตัวโดย "การซ่อน" ข้อมูลและ IP addresses ไพรเวต ขณะส่งผ่านเน็ตเวิร์ก

อัลกอริทึมการพิสูจน์ตัวตนพิสูจน์ identity ของผู้ส่งและ data integrity โดยการใช้ฟังก์ชันการแฮชที่มีการการเข้ารหัสเพื่อประมวลผล แพ็กเก็ตของข้อมูล (ที่มีฟิลด์ส่วนหัว IP คงที่รวมอยู่) โดยใช้ คีย์ลับเพื่อสร้างส่วนย่อยเฉพาะ ที่ฝั่งผู้รับ ข้อมูล ถูกประมวลผลโดยใช้ฟังก์ชันและคีย์เดียวกัน ถ้าข้อมูลถูก เปลี่ยนแปลง หรือคีย์ผู้ส่งไม่ถูกต้อง เดทาแกรมจะถูกข้าม

การเข้ารหัสใช้อัลกอริทึมการเข้ารหัสเพื่อแก้ไขและส่ม ข้อมูลโดยใช้อัลกอริทึมและคีย์เฉพาะเพื่อสร้างข้อมูลที่เข้ารหัส ที่รู้จักเป็น *ข้อความเข้ารหัส* การเข้ารหัสทำให้ ไม่สามารถอ่านข้อมูลได้ระหว่างการส่ง หลังจากได้รับข้อมูล ข้อมูลจะ ถูกกู้คืนโดยใช้อัลกอริทึมและคีย์เดียวกัน (ด้วยอัลกอริทึมการเข้ารหัส แบบสมมาตร) การเข้ารหัสต้องเกิดขึ้นคู่กับการพิสูจน์ตัวตนเพื่อตรวจสอบ data integrity ของข้อมูลที่เข้ารหัส

เซอร์วิสเบื้องต้นเหล่านี้ถูกนำไปใช้ใน IPsec โดยการใช้ Encapsulating Security Payload (ESP) และ Authentication Header (AH) ESP ช่วยให้มีการรักษาความลับโดยการเข้ารหัสแพ็กเก็ต IP ต้นฉบับ สร้างส่วนหัว ESP และใส่ข้อความเข้ารหัสใน ESP payload

AH สามารถใช้ตามลำพังเพื่อการพิสูจน์ตัวตนและการตรวจสอบ integrity ถ้าการรักษาความลับไม่ใช่ปัญหาสำคัญ ด้วย AH ฟิลด์สแตติกของ ส่วนหัว IP และข้อมูลจะมีอัลกอริทึมการแฮชนำใช้เพื่อคำนวณ ส่วนย่อยที่เป็นคีย์ ผู้รับใช้คีย์เพื่อคำนวณและเปรียบเทียบ ส่วนย่อยเพื่อทำให้แน่ใจว่าแพ็กเก็ตไม่ถูกเปลี่ยนแปลง และ identity ของผู้ส่ง ได้รับการพิสูจน์ตัวตน

คุณลักษณะ IP security:

ต่อไปนี้คือคุณลักษณะของ IP Security

คุณลักษณะต่อไปนี้พร้อมใช้งานกับ Internet Key Exchange สำหรับระบบปฏิบัติการ AIX :

- สนับสนุนอัลกอริทึม AES 128 บิต 192 บิต และ 256 บิต
- การเร่งด้วยฮาร์ดแวร์ที่มี 10/100 Mbps Ethernet PCI Adapter II
- ส่วนสนับสนุน AH โดยใช้ RFC 2402, และส่วนสนับสนุน ESP โดยใช้ RFC 2406
- Manual tunnels สามารถตั้งค่าเพื่อจัดให้มีการทำงานร่วมกันกับ ระบบอื่นๆ ที่ไม่สนับสนุนวิธีการรีเฟรชคีย์ IKE อัตโนมัติ และสำหรับการใช้ IP Version 6 tunnels
- โหมด Tunnel และโหมดการส่งของ encapsulation สำหรับ tunnel โฮสต์ หรื เกตเวย์
- อัลกอริทึมการพิสูจน์ตัวตนของ HMAC (Hashed Message Authentication Code) MD5 (Message Digest 5) และ HMAC SHA (Secure Hash Algorithm)
- อัลกอริทึมการเข้ารหัสประกอบด้วย Data Encryption Standard (DES) 56 บิต Cipher Block Chaining (CBC) ที่มี initial vector (IV) 64 บิต, Triple DES, DES CBC 4 (IV 32 บิต), และ AES CBC
- Dual IP Stack Support (IP version 4 และ IP version 6)
- การรับส่งข้อมูลทั้ง IP Version 4 และ IP Version 6 สามารถ encapsulated และกรองได้ เนื่องจากสแต็ก IP เป็นค่าแยกฟังก์ชัน IP Security สำหรับแต่ละสแต็กสามารถตั้งค่าแยกกันได้
- การกรองทราฟิกที่มีความปลอดภัยและไม่มีความปลอดภัยด้วยคุณสมบัติ IP ที่หลากหลาย เช่น IP address ต้นทางและปลายทาง, อินเตอร์เฟส, โปรโตคอล, หมายเลขพอร์ต, และอื่นๆ
- การสร้างและการลบกฎตัวกรองอัตโนมัติกับประเภท tunnel ส่วนใหญ่
- ใช้ชื่อโฮสต์สำหรับแอดเดรสปลายทางเมื่อคุณนิยาม tunnels และกฎการกรอง ชื่อโฮสต์ถูกแปลงเป็น IP addresses โดยอัตโนมัติ (เมื่อ DNS พร้อมใช้งาน)
- การบันทึกการทำงานเหตุการณ์ IP Security ลงใน **syslog**
- การใช้การติดตามระบบและสถิติสำหรับการพิจารณาปัญหา
- แอ็คชันดีฟอลต์ที่กำหนดโดยผู้ใช้อนุญาตให้ผู้ใช้ระบุ ทราฟิกที่ไม่ตรงกับ tunnels ที่นิยามไว้ซึ่งได้รับอนุญาต

คุณลักษณะเพิ่มเติมต่อไปนี้ มีอยู่พร้อมกับ Internet Key Exchange สำหรับ AIX 6.1 TL 05 หรือใหม่กว่า:

- IPSec สนับสนุนโดยใช้ RFC 4301, AH สนับสนุนโดยใช้ RFC 4302, และสนับสนุน ESP โดยใช้ RFC 4303
- อัลกอริทึมการพิสูจน์ตัวตนของ Cipher-based Message Authentication Code (CMAC) AES XCBC
- อัลกอริทึมการเข้ารหัสประกอบด้วย AES 128 บิต, 192 บิต, GCM 256 บิต (IV 16 บิต), AES-128-GMAC, AES-192-GMAC, และ AES-256-GMAC
- การสนับสนุนช่วงพอร์ตสำหรับกฎตัวกรอง
- Extended Sequence Numbers

คุณลักษณะ Internet Key Exchange:

ต่อไปนี้คือคุณลักษณะที่มาพร้อมกับ Internet Key Exchange สำหรับ AIX

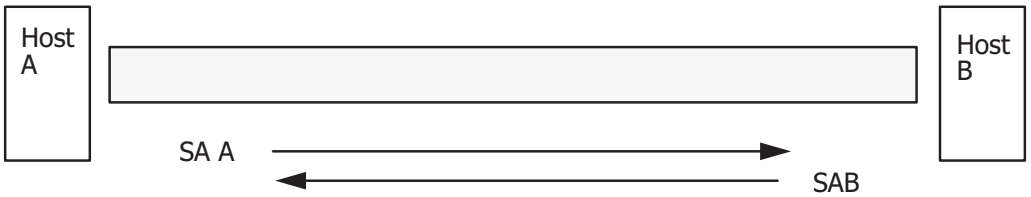
คุณลักษณะเพิ่มเติมต่อไปนี้มาพร้อมกับ Internet Key Exchange สำหรับ AIX 6.1 หรือใหม่กว่า:

- การสนับสนุน AH สำหรับการแฮช HMAC SHA2 256 บิต (TL 04 หรือใหม่กว่า)
- การเข้ารหัส ESP สนับสนุน GCM AES 128 บิต 192 บิต 256 บิตที่มี (16 bit IV) อัลกอริทึม GMAC AES 128 บิต 192 บิต 256 บิต การพิสูจน์ตัวตน ESP สนับสนุนโดย HMAC MD5 และ HMAC SHA1 (TL 04 หรือใหม่กว่า)
- IKEv1 (RFC2409) และ IKEv2 (RFC4306) ได้รับการสนับสนุน (TL 02 หรือใหม่กว่า) IKEv1 ได้รับการสนับสนุนโดย **isakmpd** daemon และ IKEv2 ได้รับการสนับสนุนโดย **ikev2d** daemon (TL 02 หรือใหม่กว่า) IKEv1 และ IKEv2 tunnels สามารถมีคู่กัน
- การสนับสนุนสำหรับอัลกอริทึม integrity CMAC_AES_XCBC and HMAC_SHA2_256 (TL 04 หรือใหม่กว่า)
- การสนับสนุนสำหรับอัลกอริทึม PRF PRF_SHA2_256 (TL 04 หรือใหม่กว่า)
- การสนับสนุนสำหรับ Diffie Hellman กลุ่ม 14, 19 และ 24 (TL 04 หรือใหม่กว่า)

การรวมกลุ่มการรักษาความปลอดภัย:

บล็อกการสร้างที่สร้างการสื่อสารที่มีความปลอดภัย คือแนวความคิดที่รู้จักเป็น *การรวมกลุ่มการรักษาความปลอดภัย* การรวมกลุ่มการรักษาความปลอดภัยเกี่ยวข้องกับชุดพารามิเตอร์ความปลอดภัยที่เฉพาะ เจาะจงสำหรับประเภทของการรับส่งข้อมูล

ด้วยข้อมูลที่ป้องกันด้วย IP Security การรวมกลุ่มการรักษาความปลอดภัยแยก มีอยู่สำหรับแต่ละทิศทางและสำหรับแต่ละประเภทส่วนหัว AH หรือ ESP ข้อมูล ที่อยู่ในการรวมกลุ่มการรักษาความปลอดภัยประกอบด้วย IP addresses ของฝ่ายที่ทำการสื่อสาร identifier เฉพาะที่ รู้จักเป็น Security Parameters Index (SPI) อัลกอริทึมที่เลือกสำหรับ การพิสูจน์ตัวตนหรือการเข้ารหัส คีย์การพิสูจน์ตัวตนและการเข้ารหัส และอายุการใช้งานคีย์ รูปต่อไปนี้แสดงการรวมกลุ่มการรักษาความปลอดภัยระหว่าง Host A และ Host B



SA = การเชื่อมโยงความปลอดภัย ประกอบด้วย:

ที่อยู่ปลายทาง
SPI
คีย์
อัลกอริทึมและรูปแบบลับ
อัลกอริทึมการพิสูจน์ตัวตน
ไลฟ์ไทม์คีย์

รูปที่ 6. การจัดทำ Tunnel ที่ปลอดภัยระหว่าง Hosts A และ B

ภาพประกอบนี้แสดง tunnel เสมือนที่ทำงานระหว่าง Host A และ Host B Security association A คือลูกศรที่มีทิศทางจาก Host A ไป Host B Security association B คือลูกศรที่มีทิศทางจาก Host B ไป Host A การรวมกลุ่มการรักษาความปลอดภัยประกอบด้วย Destination Address, SPI, Key, Crypto Algorithm and Format, Authentication Algorithm และ Key Lifetime

เป้าหมายของการจัดการคีย์คือเพื่อเจรจาและคำนวณการรวมกลุ่ม การรักษาความปลอดภัยที่จะช่วยปกป้องการรับส่งข้อมูล IP

Tunnels และการจัดการคีย์:

ใช้ tunnel เพื่อเจรจาและจัดการการรวมกลุ่มการรักษาความปลอดภัย ที่จำเป็นสำหรับการตั้งค่าการสื่อสารที่ปลอดภัยระหว่างสองโฮสต์

ประเภท tunnels ต่อไปนี้ได้รับการสนับสนุน แต่ละประเภทใช้เทคนิคการจัดการ คีย์ต่างกัน:

- IKE tunnels (การเปลี่ยนคีย์แบบไดนามิก มาตรฐาน IETF)
- Manual tunnels (สแตติก คีย์คงที่ มาตรฐาน IETF)

การสนับสนุน Internet Key Exchange tunnel:

IKE Tunnels ยึดตามมาตรฐาน Internet Security Association and Key Management Protocol (ISAKMP)/Oakley ที่พัฒนาโดย IETF ด้วยโปรโตคอลนี้ พารามิเตอร์ด้านความปลอดภัยจะถูกเจรจาและ รีเฟรช และคีย์ถูกแลกเปลี่ยนอยากปลอดภัย

ประเภทการพิสูจน์ตัวตนต่อไปนี้ได้รับการสนับสนุน:

- คีย์ที่แบ่งใช้ล่วงหน้า
- ลายเซ็นใบรับรองดิจิตัล X.509v3
- บน AIX 6.1 TL 04 หรือใหม่กว่า IKEv2 สนับสนุนลายเซ็นสนับสนุนดิจิตัล ECDSA-256 ให้เป็นส่วนหนึ่งของวิธีการพิสูจน์ตัวตน X509v3 ที่ยึดตามใบรับรอง ดิจิตัล

การเจรจาใช้แนวการดำเนินการสองเฟส เฟส 1 พิสูจน์ตัวตน ฝ่ายที่เกี่ยวกับการสื่อสาร และระบุอัลกอริทึมที่จะใช้ สำหรับการสื่อสารอย่างปลอดภัยในเฟส 2 ระหว่างเฟส 2 พารามิเตอร์ IP Security ที่จะใช้ระหว่างการถ่ายโอนข้อมูลจะถูกเจรจา และการเชื่อมโยงด้าน ความปลอดภัยและคีย์ถูกสร้างและแลกเปลี่ยน

ตารางต่อไปนี้แสดงอัลกอริทึมการพิสูจน์ตัวตนที่สามารถ ใช้กับรการรักษาความปลอดภัย AH และ ESP สำหรับการสนับสนุน IKE tunnel

ตารางที่ 14. อัลกอริทึมการพิสูจน์ตัวตนสำหรับการสนับสนุน IKE tunnel

อัลกอริทึม	AH IP Version 4 & 6	ESP IP Version 4 & 6
HMAC MD5	X	X
HMAC SHA1	X	X
DES CBC 8		X
Triple DES CBC		X
AES CBC (128, 192, 256)		X
ESP Null		X
AES-XCBC-MAC-96	X	X
AES GCM (128, 192, 256)		X
AES GMAC (128, 192, 256)	X	

ตารางที่ 14. อัลกอริทึมการพิสูจน์ตัวตนสำหรับการสนับสนุน IKE tunnel (ต่อ)

อัลกอริทึม	AH IP Version 4 & 6	ESP IP Version 4 & 6
ESP_ENCR_NULL_ AUTH_AES_GMAC		X

การสนับสนุน Manual tunnel:

Manual tunnels มีความเข้ากันได้กับระบบเก่า และ สื่อสารระหว่างเครื่องที่ไม่สนับสนุนโปรโตคอลการจัดการคีย์ IKE ได้ ข้อด้อยของ manual tunnels คือค่าคีย์ เป็นค่าสแตติก คีย์การเข้ารหัสและการพิสูจน์ตัวตนจะเหมือนกันตลอด ระยะเวลาของ tunnel และต้องอัพเดตด้วยตนเอง

ตารางต่อไปนี้แสดงอัลกอริทึมการพิสูจน์ตัวตนที่สามารถ ใช้กับโปรโตคอลการรักษาความปลอดภัย AH และ ESP สำหรับการสนับสนุน manual tunnel

อัลกอริทึม	AH IP Version 4	AH IP Version 6	ESP IP Version 4	ESP IP Version 6
HMAC MD5	X	X	X	X
HMAC SHA1	X	X	X	X
AES CBC (128, 192, 256)			X	X
Triple DES CBC			X	X
DES CBC 8			X	X
DES CBC 4			X	X

เนื่องจาก IKE tunnels นำเสนอการรักษาความปลอดภัยที่มีประสิทธิภาพมากกว่า IKE จึงเป็น วิธีการจัดการคีย์ที่นิยมใช้

ความสามารถในการกรองเบื้องต้น:

การกรอง เป็นฟังก์ชันพื้นฐานที่ซึ่งแพ็กเก็ตขาเข้าและขาออก จะถูกยอมรับหรือปฏิเสธขึ้นกับคุณสมบัติอันหลากหลาย โดยอนุญาตให้ผู้ใช้หรือผู้ดูแลระบบตั้งค่าโฮสต์เพื่อควบคุม ปริมาณการรับส่งระหว่างโฮสต์นี้และโฮสต์อื่นๆ

การกรองถูกดำเนินการบนคุณสมบัติที่หลากหลายของแพ็กเก็ต เช่นแอดเดรส ต้นทางและปลายทาง เวอร์ชัน IP (4 หรือ 6) subnet masks โปรโตคอล พอร์ต คุณสมบัติการจัดเส้นทาง การแตกแฟรกเมนต์ อินเตอร์เฟส และนิยาม tunnel

กฎ ที่รู้จักเป็น *กฎตัวกรอง* ถูกใช้เพื่อเชื่อมโยงประเภทการรับส่ง เฉพาะกับ tunnel พิเศษ ในการตั้งค่าเบื้องต้นสำหรับ manual tunnels เมื่อผู้ใช้กำหนด host-to-host tunnel กฎตัวกรองจะถูกสร้างอัตโนมัติ เพื่อกำหนดทิศทางการรับส่งข้อมูลทั้งหมดจากโฮสต์จนถึง tunnel ที่ปลอดภัย ถ้าต้องการ ประเภทการรับส่งข้อมูลที่เจาะจงมากยิ่งขึ้น (เช่น ซับเน็ตไปยังซับเน็ต) สามารถแก้ไขหรือแทนที่กฎตัวกรองเพื่อให้มีการควบคุมการรับส่งที่เจาะจงมากขึ้นโดยใช้ tunnel จำเพาะ

สำหรับ IKE tunnels กฎตัวกรองยังถูกสร้างโดยอัตโนมัติและ แทรกไว้ในตารางตัวกรองทันทีที่ tunnel ถูกเรียกทำงาน

ในทำนองเดียวกัน เมื่อ tunnel ถูกแก้ไขหรือลบ กฎตัวกรองสำหรับ tunnel นั้นจะถูกลบโดยอัตโนมัติ ซึ่งช่วยให้ง่ายต่อการตั้งค่า IP Security และช่วยลดข้อผิดพลาดของบุคคล นิยาม Tunnel สามารถนำไปกระจายและ แบ่งใช้ระหว่างเครื่อง และไฟร์วอลล์ โดยใช้ยูทิลิตีการอิมพอร์ตและเอ็กซ์พอร์ต ซึ่งมีประโยชน์การดูแลเครื่องจำนวนมาก

กฎตัวกรองเชื่อมโยงประเภทการรับส่งข้อมูลเฉพาะกับ tunnel แต่ข้อมูล ที่ถูกกรองไม่จำเป็นต้องเดินทางใน tunnel ลักษณะการทำงาน ของกฎตัวกรองนี้ช่วยให้ระบบปฏิบัติการจัดให้มีฟังก์ชันการทำงานของไฟร์วอลล์ระดับต้น แก่บุคคลที่ต้องการจำกัดปริมาณรับส่งข้อมูล หรือจากเครื่องในอินทราเน็ต หรือในเน็ตเวิร์กที่ไม่มีการป้องกันด้วยไฟร์วอลล์อย่างแท้จริง ในสถานการณ์นี้ กฎตัวกรองจัดให้มีการปกป้องด่านที่สองแก่ กลุ่มของเครื่อง

หลังจากกฎตัวกรองถูกสร้าง จะถูกเก็ฐในตารางและโหลด เข้าสู่เคอร์เนล เมื่อแพ็กเก็ตพร้อมส่งหรือรับจากเน็ตเวิร์ก กฎตัวกรองจะถูกตรวจสอบตามรายการตั้งแต่บนมาถึงล่างเพื่อพิจารณาว่า แพ็กเก็ตควรได้รับอนุญาต ปฏิเสธ หรือส่งผ่าน tunnel เกณฑ์ของกฎถูกนำไปเปรียบเทียบกับคุณสมบัติแพ็กเก็ตจนว่าจะตรงกัน หรือถึงค่ากฎแพ็กเก็ต

ฟังก์ชัน IP Security ยังใช้การกรองของแพ็กเก็ตที่ไม่ปลอดภัย ตามขนาดที่เล็กมาก เกณฑ์ที่ผู้ใช้กำหนดเอง ซึ่งอนุญาตให้การควบคุม ปริมาณการรับส่งข้อมูล IP ระหว่างเน็ตเวิร์กและเครื่องไม่จำเป็นต้องมีคุณสมบัติการพิสูจน์ตัวตน หรือการเข้ารหัสของ IP Security

การสนับสนุนใบรับรองดิจิตัล:

IP Security สนับสนุนการใช้ใบรับรองดิจิตัล X.509 Version 3

เครื่องมือ Key Manager จัดการการร้องขอใบรับรอง รักษา ฐานข้อมูลคีย์ และดำเนินงานด้านการดูแลอื่นๆ

ใบรับรองดิจิตัลอธิบายอยู่ใน Digital Certificate Configuration Key Manager และฟังก์ชัน ถูกอธิบายใน Using the IBM Key Manager Tool

Virtual private networks และ IP security:

virtual private network (VPN) ขยายไพรเวต อินทราเน็ตไปยังพับลิกเน็ตเวิร์กเช่นอินเตอร์เน็ตได้อย่างปลอดภัย

VPNs ลำเลียงข้อมูลไปยัง tunnel ไพรเวตที่จำเป็น ผ่านอินเตอร์เน็ตไปยังและจากผู้ใช้รีโมต สำนักงานสาขา และ คู่ค่าทางธุรกิจ/ซัพพลายเออร์ บริษัทสามารถเลือกการเข้าถึงอินเตอร์เน็ต ผ่านผู้ให้บริการอินเตอร์เน็ต (ISPs) โดยใช้หมายเลขตรง หรือ หมายเลขโทรศัพท์ในพื้นที่และช่วยลดค่าใช้จ่าย leased lines การโทร ทางไกล และหมายเลขโทรศัพท์โทรฟรี วิธีแก้ปัญหาด้วย VPN สามารถใช้ มาตรฐานการรักษาความปลอดภัย IPsec เนื่องจาก IPsec คือเฟรมเวิร์ก การรักษาความปลอดภัยเน็ตเวิร์กมาตรฐานอุตสาหกรรมที่เลือกโดย IETF สำหรับทั้ง สภาวะแวดล้อมทั้ง IP Version 4 และ 6 และไม่จำเป็นต้องเปลี่ยนแปลงใดๆ ในแอ็พพลิเคชันที่มีอยู่แล้ว

รีซอร์สที่แนะนำสำหรับการวางแผนการใช้ VPN ในระบบปฏิบัติการ AIX อยู่ในบทที่ 9 ของ *A Comprehensive Guide to Virtual Private Networks, Volume III: Cross-Platform Key and Policy Management*, ISBN SG24-5309-00 คู่มือนี้ ยังมีอยู่บนอินเตอร์เน็ตเวิล์ดไวด์เว็บที่ http://www.redbooks.ibm.com/redbooks/SG245309.html

การติดตั้งคุณลักษณะ IP security

คุณลักษณะ IP Security ใน AIX สามารถติดตั้งและโหลด แยกกันได้

ชุดไฟล์ที่ต้องติดตั้งมีดังต่อไปนี้:

- `bos.net.ipsec.rte` (สภาวะแวดล้อมรันไทม์ สำหรับสภาวะแวดล้อมและคำสั่ง kernel IP Security)
- `bos.msg.LANG.net.ipsec` (โดยที่ *LANG* คือภาษาที่ระบุไว้, เช่น en_US)
- `bos.net.ipsec.keymgt`
- `clic.rte` (CryptoLite สำหรับ C ชุดไฟล์สำหรับการเข้ารหัส DES, triple DES และ AES)

สำหรับส่วนสนับสนุนการลงนามดิจิทัล IKE, คุณยังต้องติดตั้งชุดไฟล์ `gskit.rte` หรือ `gskkm.rte` จากแพ็กเสริม

หลังจากที่ติดตั้งแล้ว, IP Security สามารถโหลดแยกกันสำหรับ IP เวอร์ชัน 4 และ IP เวอร์ชัน 6, โดยใช้โพรซีเดอร์ที่แนะนำที่ได้จัดเตรียมไว้ใน "การโหลดการรักษาความปลอดภัย IP"หรือโดยใช้คำสั่ง **mkdev**

การโหลดการรักษาความปลอดภัย IP:

ใช้ SMIT เพื่อโหลดโมดูล IP security เมื่อเริ่มต้นทำงานกับ IP Security และ, SMIT ต้องมั่นใจว่าส่วนขยายเคอร์เนล และ IKE daemons ถูกโหลดอยู่ในลำดับที่ถูกต้อง

หมายเหตุ: การโหลด IP Security เปิดใช้ฟังก์ชัน การกรอง ก่อนการโหลด สิ่งสำคัญคือต้องแน่ใจว่าสร้างกฎตัวกรอง ที่ถูกต้อง มิฉะนั้น การสื่อสารภายนอกทั้งหมดอาจถูกบล็อก

ถ้า การโหลดเสร็จเรียบร้อย คำสั่ง **lsdev** จะแสดง อุปกรณ์ IP Security เป็น `Available`

```
lsdev -C -c ipsec

   ipsec_v4 Available IP Version 4 Security Extension
   ipsec_v6 Available IP Version 6 Security Extension
```

หลังส่วนขยายเคอร์เนล IP Security ได้ถูกโหลด tunnels และตัวกรองจะพร้อมให้ตั้งค่า

การวางแผนการตั้งค่าการรักษาความปลอดภัย IP

ในการตั้งค่า IP Security วางแผนเพื่อตั้งค่า tunnels และตัวกรอง เป็นอันดับแรก

เมื่อกำหนด tunnel แบบง่ายสำหรับการรับส่งข้อมูลทั้งหมดใช้ กฎ ตัวกรองสามารถสร้างขึ้นโดยอัตโนมัติ ถ้าต้องการการกรองที่ซับซ้อนยิ่งขึ้น คุณสามารถตั้งค่ากฎตัวกรองแยกต่างหาก

คุณสามารถ กำหนดคอนฟิก IP Security โดยใช้ปลั๊กอิน Virtual Private Network หรือ System Management Interface Tool (SMIT) ถ้าใช้ SMIT พาธด่วนต่อไปนี้จะมีให้ใช้ได้:

smit ips4_basic
: การตั้งค่าระดับต้นสำหรับ IP version 4

smit ips6_basic
: การตั้งค่าระดับต้นสำหรับ IP version 6

ก่อนการตั้งค่า IP Security สำหรับไซต์ของคุณ คุณต้อง ตัดสินใจว่าวิธีใดที่คุณต้องการใช้ ตัวอย่างเช่น คุณต้องการใช้ tunnels หรือตัวกรอง (หรือทั้งคู่) ประเภท tunnel ใดเหมาะสมที่สุดสำหรับความต้องการของคุณ และอื่นๆ ส่วนต่อไปนี้ให้ข้อมูลที่คุณต้องทำความเข้าใจ ก่อนตัดสินใจ:

การเร่งด้วยฮาร์ดแวร์:

10/100 Mbps Ethernet PCI Adapter II (รหัสผลิตภัณฑ์ 4962) มี IP Security แบบมาตรฐานและออกแบบเพื่อออฟโหลดฟังก์ชัน IP Security จากระบบปฏิบัติการ AIX

เมื่อ 10/100 Mbps Ethernet PCI Adapter II ถูกแสดงในระบบ AIX สแต็ก IP Security ใช้ความสามารถต่อไปนี้ของอะแด็ปเตอร์:

- การเข้ารหัสและการถอดรหัสโดยใช้อัลกอริทึม DES หรือ Triple DES
- การพิสูจน์ตัวตนโดยใช้อัลกอริทึม MD5 หรือ SHA-1
- สื่อบันทึกที่เก็บข้อมูลที่เกี่ยวกับความปลอดภัย

ฟังก์ชันบนอะแด็ปเตอร์ถูกใช้แทนอัลกอริทึมซอฟต์แวร์ 10/100 Mbps Ethernet PCI Adapter II พร้อมใช้สำหรับ manual และ IKE tunnels

คุณลักษณะการเร่งด้วยฮาร์ดแวร์ของ IP Security มีอยู่ใน 5.1.0.25 หรือ ระดับใหม่กว่าของชุดไฟล์ `bos.net.ipsec.rte` และ `devices.pci.1410ff01.rte`

มีการจำกัดจำนวนความเชื่อมโยงด้านความปลอดภัยที่สามารถ ออฟโหลดไปยังเน็ตเวิร์กอะแด็ปเตอร์บนฝั่งรับ (การรับส่งข้อมูลขาเข้า) บนฝั่งการส่ง (การรับส่งข้อมูลขาออก) แพ็กเก็ตทั้งหมดที่ใช้การตั้งค่า ที่สนับสนุนจะถูกออฟโหลดไปยังอะแด็ปเตอร์ การตั้งค่า tunnel บางส่วน ไม่สามารถถูกออฟโหลดไปยังอะแด็ปเตอร์

10/100 Mbps Ethernet PCI Adapter II สนับสนุนคุณลักษณะ ต่อไปนี้:

- การเข้ารหัส DES, 3DES หรือ NULL ผ่าน ESP
- การพิสูจน์ตัวตน HMAC-MD5 หรือ HMAC-SHA-1 ผ่าน ESP หรือ AH แต่ไม่ใช่ ทั้งสอง (ถ้าใช้ทั้ง ESP และ AH ต้องดำเนินงาน ESP ก่อน นี้เป็นจริง เสมอสำหรับ IKE tunnels แต่ผู้ใช้สามารถเลือกลำดับ เองได้สำหรับ manual tunnels)
- โหมด Transport และ Tunnel
- ออฟโหลดของแพ็กเก็ต IPV4

หมายเหตุ: 10/100 Mbps Ethernet PCI Adapter II ไม่สามารถจัดการแพ็กเก็ต ที่มีอ็อพชัน IP

ในการเปิดใช้งาน 10/100 Mbps Ethernet PCI Adapter II สำหรับ IP Security คุณอาจต้องแยกเน็ตเวิร์กอินเตอร์เฟสออก จากนั้นเปิดใช้งานคุณลักษณะ IPsec Offload

ในการแยกเน็ตเวิร์กอินเตอร์เฟสออก ดำเนินขั้นตอนต่อไปนี้โดยใช้ อินเตอร์เฟส SMIT:

ในการเปิดใช้งานคุณลักษณะ IPsec Offload ทำสิ่งต่อไปนี้โดยใช้ อินเตอร์เฟส SMIT:

1. ล็อกอินเป็นผู้ใช้ **root**
2. พิมพ์ `smitty eadap` ที่บรรทัดคำสั่งและกด Enter
3. เลือกอ็อพชัน **Change / Show Characteristics of an Ethernet Adapter** และกด Enter
4. เลือก 10/100 Mbps Ethernet PCI Adapter II และกด Enter
5. เปลี่ยนฟิลด์ IPsec Offload เป็น yes และ กด Enter

ในการแยกเน็ตเวิร์กอินเตอร์เฟสออกจากบรรทัดคำสั่ง พิมพ์ คำสั่งต่อไปนี้:

```
# ifconfig enX detach
```

ในการเปิดใช้งานแอ็ตทริบิวต์ออฟโหลด IPsec จากบรรทัดคำสั่ง พิมพ์ คำสั่งต่อไปนี้:

```
# chdev -l entX -a ipsec_offload=yes
```

ในการตรวจสอบว่าแอ็ตทริบิวต์ออฟโหลด IPsec ถูกเปิดใช้งานจาก บรรทัดคำสั่ง พิมพ์คำสั่งต่อไปนี้:

```
# lsattr -El entX detach
```

ในการปิดใช้งานแอ็ตทริบิวต์ออฟโหลด IPsec จากบรรทัดคำสั่ง พิมพ์ คำสั่งต่อไปนี้:

```
# chdev -l entX -a ipsec_offload=no
```

ใช้คำสั่ง **enstat** เพื่อทำให้แน่ใจว่าการตั้งค่า tunnel ของคุณกำลังใช้ประโยชน์แอ็ตทริบิวต์ออฟโหลด IPsec คำสั่ง **enstat** แสดงสถิติทั้งหมดของ แพ็กเก็ต IPsec ที่ส่งและรับเมื่อแอ็ตทริบิวต์ออฟโหลด IPsec ถูกเปิดใช้งาน ตัวอย่าง ถ้าอินเตอร์เฟสอีเทอร์เน็ตคือ **ent1** พิมพ์คำสั่งต่อไปนี้:

```
# entstat -d ent1
```

เอาต์พุตจะคล้ายตัวอย่างต่อไปนี้:

```
.
.
.
10/100 Mbps Ethernet PCI Adapter II (1410ff01) Specific Statistics:
---------------------------------------------
.
.
.
Transmit IPsec packets: 3
Transmit IPsec packets dropped: 0
Receive IPsec packets: 2
Receive IPsec packets dropped: 0
```

Tunnels เทียบกับตัวกรอง:

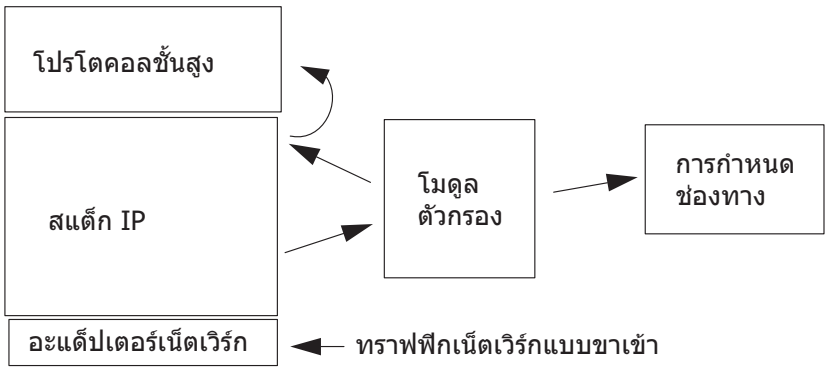
ส่วนต่างกันสองส่วนของ IP Security คือ *tunnels* และ *filters* Tunnels จำเป็นต้องใช้ตัวกรอง แต่ตัวกรองไม่จำเป็นต้องใช้ tunnels

การกรอง เป็นฟังก์ชันที่ซึ่งแพ็กเก็ตขาเข้าและขาออก จะถูกยอมรับหรือปฏิเสธขึ้นกับคุณสมบัติอันหลากหลายที่เรียกว่า *กฎ* ฟังก์ชันนี้อนุญาตให้ผู้ดูแลระบบตั้งค่าโฮสต์เพื่อควบคุม การรับส่งข้อมูลระหว่างโฮสต์นี้และโฮสต์อื่น การกรองถูกดำเนินการบนคุณสมบัติ ที่หลากหลายของแพ็กเก็ต เช่นแอดเดรสต้นทางและปลายทาง IP Version (4 หรือ 6) subnet masks โปรโตคอล พอร์ต คุณสมบัติการจัดเส้นทาง การแตกแฟรกเมนต์ อินเตอร์เฟส และนิยาม tunnel การกรองนี้ทำที่เลเยอร์ IP layer จึงไม่จำเป็นต้องทำการเปลี่ยนแปลงใดๆ ในแอ็พพลิเคชัน

Tunnels กำหนดการรวมกลุ่มการรักษาความปลอดภัยระหว่างสองโฮสต์ การรวมกลุ่ม ความปลอดภัยเหล่านี้เกี่ยวข้องกับพารามิเตอร์การรักษาความปลอดภัยที่เจาะจงที่แบ่งใช้ ระหว่างจุดหมายของ tunnel

ภาพประกอบต่อไปนี้บ่งชี้วิธีที่แพ็กเก็ตจากเน็ตเวิร์กอะแด็ปเตอร์เข้ามา ในสแต็ก IP จากที่นั่น โมดูลตัวกรองถูกเรียกใช้เพื่อพิจารณาว่า แพ็กเก็ตได้รับอนุญาตหรือปฏิเสธ ถ้า tunnel ID ถูกระบุ แพ็กเก็ต ถูกตรวจสอบกับนิยาม tunnel ที่มีอยู่ ถ้าการแยก

ออกจาก tunnel สำเร็จ แพ็กเก็ตจะถูกส่งไปยังโปรโตคอลเลเยอร์ระดับบนขึ้นไป ฟังก์ชันนี้เกิดขึ้นในลำดับที่ตรงข้ามกับแพ็กเก็ตขาออก tunnel ยึดตาม กฎตัวกรองเพื่อเชื่อมโยงแพ็กเก็ตกับ tunnel เฉพาะ แต่ ฟังก์ชันการกรองสามารถเกิดขึ้นได้โดยไม่ต้องส่งแพ็กเก็ตไปยัง tunnel



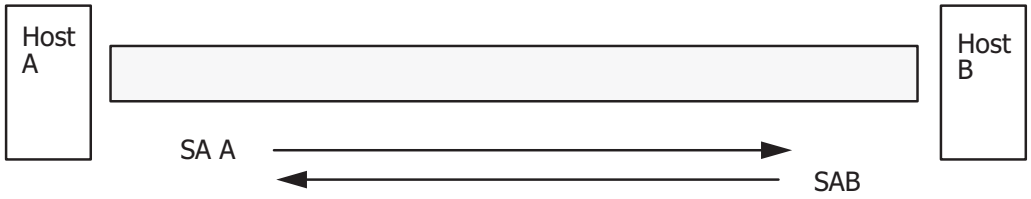
รูปที่ 7. การจัดเส้นทางเน็ตเวิร์กแพ็กเก็ต

ภาพประกอบแสดงเส้นทางที่เน็ตเวิร์กแพ็กเก็ตใช้ เข้ามาจากเน็ตเวิร์ก แพ็กเก็ตจะเข้าสู่เน็ตเวิร์กอะแด็ปเตอร์ จากที่นั้น จะไปที่สแต็ก IP ที่ซึ่งจะถูกส่งไปยังโมดูลตัวกรอง จากโมดูลตัวกรอง แพ็กเก็ตอาจถูกส่งไปยังนิยาม tunnel หรือส่งกลับไป ที่สแต็ก IP ที่ซึ่งจะถูกส่งต่อไปยังโปรโตคอลระดับบนขึ้นไป

Tunnels และการรวมกลุ่มการรักษาความปลอดภัย:

Tunnels ถูกใช้เมื่อคุณจำเป็นต้องพิสูจน์ตัวตนของข้อมูล หรือพิสูจน์ตัวตนและเข้ารหัส Tunnels ถูกกำหนดโดยการระบุ การรวมกลุ่มการรักษาความปลอดภัยระหว่างสองโฮสต์ การรวมกลุ่มการรักษาความปลอดภัย กำหนดค่าพารามิเตอร์สำหรับอัลกอริทึมการเข้ารหัสและการพิสูจน์ตัวตน รวมถึงคุณสมบัติของ tunnel

ภาพประกอบต่อไปนี้แสดง tunnel เสมือนระหว่าง Host A และ Host B



SA = การเชื่อมโยงความปลอดภัย ประกอบด้วย:

- ที่อยู่ปลายทาง
- SPI
- คีย์
- อัลกอริทึมและรูปแบบลับ
- อัลกอริทึมการพิสูจน์ตัวตน
- ไลฟ์ไทม์คีย์

รูปที่ 8. การสร้าง Secure Tunnel ระหว่าง Hosts A และ B

ภาพประกอบแสดง tunnel เสมือนที่ทำงานระหว่าง Host A และ Host B Security association A คือลูกศรที่มีทิศทางจาก Host A ไป Host B Security association B คือลูกศรที่มีทิศทางจาก Host B ไป Host A การรวมกลุ่มการรักษาความปลอดภัยประกอบด้วย Destination Address, SPI, Key, Crypto Algorithm and Format, Authentication Algorithm และ Key Lifetime

Security Parameter Index (SPI) และแอดเดรสปลายทาง ระบุการรวมกลุ่มการรักษาความปลอดภัยเฉพาะ พารามิเตอร์เหล่านี้จำเป็นสำหรับ การระบุ tunnel เฉพาะ พารามิเตอร์อื่นๆ เช่น อัลกอริทึมการเข้ารหัส อัลกอริทึมการพิสูจน์ตัวตน คีย์ และช่วงอายุสามารถถูกระบุ หรือใช้ค่าดีฟอลต์

ข้อควรพิจารณา Tunnel:

คุณควรพิจารณาหลายสิ่ง ก่อนตัดสินใจเลือก ชนิดของ tunnel ที่จะใช้สำหรับความปลอดภัย IP

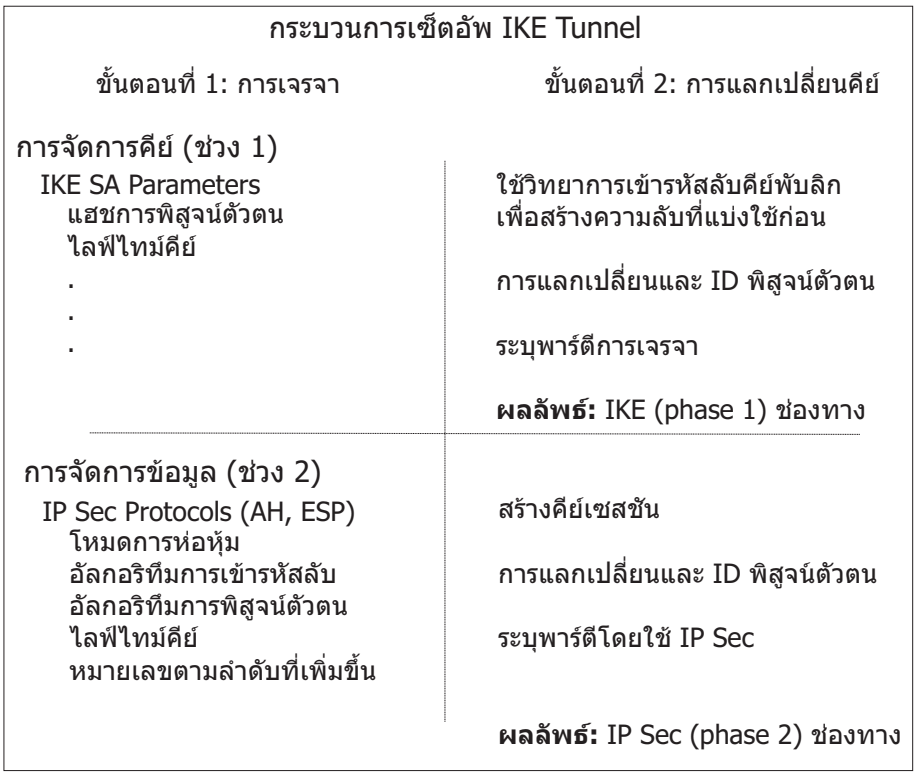
IKE tunnels ต่างจาก manual tunnels เนื่องจากคอนฟิกูเรชัน ของนโยบายความปลอดภัยเป็นกระบวนการแยกจากการกำหนดจุดสิ้นสุด tunnel

ใน IKE มีสองขั้นตอนในกระบวนการ แลกเปลี่ยนข้อมูล แต่ละกระบวนการแลกเปลี่ยน ข้อมูลเรียกว่า *เฟส* และแต่ละเฟสมีนโยบายความปลอดภัยแยกกัน

เมื่อการแลกเปลี่ยนข้อมูล Internet Key เริ่มขึ้น ต้องสร้างช่องทางที่ ปลอดภัยสำหรับการแลกเปลี่ยนข้อมูล ซึ่งเรียกว่าเฟส *การจัดการคีย์* หรือ *เฟส 1* ระหว่างเฟสนี้ แต่ละกลุ่มใช้คีย์ที่แบ่งใช้ล่วงหน้า หรือใบรับรองดิจิตัลเพื่อพิสูจน์ตัวตนของอีกฝ่ายหนึ่ง และส่งข้อมูล ID เฟสนี้สร้างกลุ่มความปลอดภัยระหว่างสองกลุ่ม กำหนดวิธีที่กลุ่มวางแผนในการสื่อสารอย่างปลอดภัย และการป้องกันที่ใช้ เพื่อสื่อสารระหว่างเฟสที่สอง ผลลัพธ์ของ เฟสนี้คือ *IKE* หรือ *เฟส 1* tunnel

เฟสที่สองเรียกว่าเฟส *การจัดการข้อมูล* หรือ *เฟส 2* และใช้ IKE tunnel เพื่อสร้างกลุ่มความปลอดภัย สำหรับ AH และ ESP ที่ป้องกันการเดินทางของข้อมูล เฟสที่สองยังกำหนด ข้อมูลที่จะใช้ IP Security tunnel ตัวอย่างเช่น สามารถกำหนดข้อมูลดังต่อไปนี้:

- subnet mask
- ขอบเขตแอดเดรส
- การรวมโปรโตคอลและหมายเลขพอร์ต



รูปที่ 9. กระบวนการเซ็ตอัพ IKE Tunnel

ตัวอย่างนี้แสดง สองขั้นตอน กระบวนการสองเฟสสำหรับ เซ็ตอัพ IKE tunnel

หมายเหตุ: IKEv2 มีสองเฟสเช่นกัน เฟสแรก รู้จักกันในชื่อของเฟส *IKE SA* หรือ *เฟส 1* เฟสที่สอง รู้จักกันในชื่อของเฟส *CHILD SA* หรือ *เฟส 2* ไม่เหมือนกับการสร้าง tunnels ใน IKEv1 เมื่อเฟส 1 tunnel ถูกสร้างใน IKEv2 เฟส 2 tunnel ถูกเรียกทำงานโดยอัตโนมัติ คอนฟิกูเรชันของ IKEv2 tunnels เหมือนกับ IKEv1 tunnels

ในกรณีส่วนมาก จุดสิ้นสุดของการจัดการคีย์ (IKE) tunnel จะเหมือนกับจุดสิ้นสุดของการจัดการข้อมูล (IP Security) tunnel จุดสิ้นสุด IKE tunnel คือ ID ของเครื่องที่ทำการ แลกเปลี่ยนข้อมูล จุดสิ้นสุดของ IP Security tunnel อธิบายชนิด ของการเดินทางของข้อมูลที่จะใช้ IP Security tunnel สำหรับ host-to-host tunnels ธรรมดา ซึ่งการเดินทางของข้อมูลทั้งหมดระหว่างสอง tunnels ถูกป้องกันด้วย tunnel เดียวกัน จุดสิ้นสุดของ เฟส 1 และเฟส 2 tunnel เหมือนกัน เมื่อกลุ่มการแลกเปลี่ยนข้อมูลเป็นสองเกตเวย์ จุดสิ้นสุด IKE tunnel เป็นสองเกตเวย์ และจุดสิ้นสุด IP Security เป็น เครื่องหรือ subnets (หลังเกตเวย์) หรือช่วงของแอดเดรส (หลังเกตเวย์) ของผู้ใช้ tunnel

พารามิเตอร์และนโยบายการจัดการคีย์:

คุณสามารถกำหนดนโยบายการจัดการคีย์เองโดยการระบุพารามิเตอร์ ที่จะใช้ระหว่างการเจรจา IKE ตัวอย่าง มีนโยบายการจัดการคีย์ สำหรับคีย์ที่แบ่งใช้ล่วงหน้า หรือการพิสูจน์ตัวตนโหมดลายเซ็น สำหรับเฟส 1 ผู้ใช้ ต้องพิจารณาคุณสมบัติการรักษาความปลอดภัยการจัดการคีย์ที่เจาะจง ซึ่งจะใช้ในการแลกเปลี่ยน

เฟส 1 (เฟสการจัดการคีย์) ตั้งค่าพารามิเตอร์ต่อไปนี้ ของการกำหนดคอนฟิกช่องสัญญาณ IKE:

Tunnel การจัดการคีย์ (เฟส 1)
ชื่อของ IKE tunnel นี้ สำหรับแต่ละ tunnel จุดหมาย ของการเจรจาต้องถูกระบุ ต่อไปนี้คือเครื่องสองเครื่องที่วางแผน ส่ง และตรวจสอบความถูกต้องข้อความ IKE ชื่อของ tunnel อาจอธิบาย ถึงจุดหมาย tunnel เช่น VPN Boston หรือ VPN Acme

ประเภท Identity โฮสต์
ประเภท ID ที่จะใช้ในการแลกเปลี่ยน IKE ประเภท ID และค่าต้องตรงกับค่าสำหรับคีย์ที่แบ่งใช้ล่วงหน้าเพื่อให้แน่ใจว่ามีการดำเนินการ ค้นหาคีย์ที่เหมาะสม ถ้า ID ต่างหากถูกใช้เพื่อค้นหาค่าคีย์ ที่แบ่งใช้ล่วงหน้า *host ID* คือ ID ของคีย์และ *type* คือ KEY_ID ประเภท KEY_ID จะเป็นประโยชน์ถ้าโฮสต์เดียวมีค่าคีย์ที่แบ่งใช้ล่วงหน้ามากกว่าหนึ่งค่า

Identity โฮสต์
ค่าของ ID โฮสต์ที่แทนเป็น IP แอดเดรส, fully qualified domain name (FQDN) หรือผู้ใช้ตามด้วยโดเมนเนม แบบเต็ม (*user@FQDN*) ตัวอย่าง jdoe@studentmail.ut.edu

IP Address
IP แอดเดรสของรีโมตโฮสต์ ค่านี้จำเป็นต้องใช้ เมื่อประเภท ID โฮสต์คือ KEY_ID หรือเมื่อใดที่ประเภท ID โฮสต์ไม่สามารถระบุ IP แอดเดรส ตัวอย่าง ถ้าผู้ใช้ไม่สามารถระบุด้วย เซิร์ฟเวอร์ชื่อโลคัล ต้องป้อน IP แอดเดรสสำหรับฝั่งรีโมต

พารามิเตอร์และนโยบายการจัดการข้อมูล:

พารามิเตอร์ข้อเสนอการจัดการข้อมูลถูกตั้งค่าระหว่าง เฟส 1 ของการตั้งค่า IKE tunnel โดยเหมือนกับพารามิเตอร์ IP Security ที่ใช้ใน manual tunnels และอธิบายประเภทการป้องกัน ที่ใช้สำหรับการป้องกันการรับส่งข้อมูลใน tunnel คุณสามารถเริ่มทำงาน tunnel เฟส 2 มากกว่าหนึ่ง tunnel ภายใต้ tunnel เฟส 1 เดียวกัน

ประเภท endpoint ID ต่อไปนี้อธิบายประเภทของข้อมูลที่ ใช้ IP Security Data tunnel:

โฮสต์, ซับเน็ต หรือ ช่วง
อธิบายว่าการเดินทางของปริมาณการรับส่งข้อมูล ใน tunnel จะเป็นลักษณะเฉพาะของโฮสต์, ซับเน็ต หรือช่วงแอดเดรส

ID โฮสต์/ซับเน็ต
มี identity โฮสต์หรือซับเน็ตของ ระบบโลคัลและรีโมตที่ส่งข้อมูลบน tunnel นี้ พิจารณา IDs ที่ส่งในการเจรจาของเฟส 2 และกฎตัวกรองที่ จะถูกสร้างถ้าการเจรจาสำเร็จ

Subnet mask
อธิบาย IP addresses ทั้งหมดภายในซับเน็ต (ตัวอย่าง host 9.53.250.96 และ mask 255.255.255.0)

ช่วง IP Address เริ่มต้น
จัดให้มี IP address เริ่มต้นสำหรับช่วง ของแอดเดรสที่จะใช้ tunnel (ตัวอย่าง 9.53.250.96 ของ 9.53.250.96 ถึง 9.53.250.93)

ช่วง IP Address สิ้นสุด
จัดให้มี IP address สิ้นสุดสำหรับช่วง ของแอดเดรสที่จะใช้ tunnel (ตัวอย่าง 9.53.250.93 ของ 9.53.250.96 ถึง 9.53.250.93)

พอร์ต อธิบายข้อมูลโดยใช้หมายเลขพอร์ตที่เจาะจง (ตัวอย่าง 21 หรือ 23)

โปรโตคอล

อธิบายข้อมูลที่กำลังถูกส่งด้วยโปรโตคอล ที่เจาะจง (ตัวอย่าง TCP หรือ UDP) พิจารณา โปรโตคอลที่ส่งในการเจรจาของเฟส 2 และกฎตัวกรองที่ จะถูกสร้างถ้าการเจรจาสำเร็จ โปรโตคอลสำหรับจุดหมายโลคัล ต้องตรงกับโปรโตคอลสำหรับจุดหมายรีโมต

พอร์ตสิ้นสุด

อธิบายพอร์ตสิ้นสุดสำหรับการส่งข้อมูล (ตัวอย่าง 100 หรือ 500) ค่าดีฟอลต์ 65355 คือค่าจุดหมาย

ข้อจำกัด: สำหรับ IKEv2 ใช้ ช่วง IPv4 หรือ IPv6 address เป็นตัวเลือกขอบเขตเท่านั้น พอร์ตสิ้นสุดสามารถใช้ได้กับ IKEv2 และ AIX 6.1 TL 04, หรือ เวอร์ชันถัดมา

การเลือกประเภท tunnel:

การตัดสินใจใช้ manual tunnels หรือ IKE tunnels ขึ้นอยู่กับ การสนับสนุน tunnel ของปลายทางรีโมต และประเภทของการจัดการคีย์ ที่ต้องการ

เมื่อใช้ได้ ให้ใช้ IKE tunnels เนื่องจากมีการเจรจาคีย์ความปลอดภัย ที่เป็นมาตรฐานอุตสาหกรรม และการรีเฟรชคีย์ รวมทั้งใช้ประโยชน์ ของประเภทส่วนหัว IETF ESP และ AH header และสนับสนุนการป้องกันการต่อต้านการถ่ายทอดต่อ เป็นทางเลือก คุณสามารถตั้งค่าโหมดลายเซ็นเพื่ออนุญาตใช้ใบรับรองดิจิตัล

ถ้า ปลายทางรีโมตใช้อัลกอริทึมใดอัลกอริทึมหนึ่งที่จำเป็นต้องใช้ manual tunnels ควรใช้ manual tunnels Manual tunnels ทำให้แน่ใจว่าสามารถใช้งานร่วมกันได้ กับโฮสต์จำนวนมาก เนื่องจากคีย์เป็นค่าสแตติก และเปลี่ยนแปลง ยาก และอาจยุ่งยากต่อการอัพเดต ทำให้คีย์อาจไม่ปลอดภัย Manual tunnels สามารถนำมาใช้ระหว่างที่โฮสต์กำลังทำงานระบบปฏิบัติการนี้ และเครื่องอื่นๆ กำลังทำงาน IP Security และมีชุดของอัลกอริทึมการเข้ารหัส และการพิสูจน์ตัวตนร่วมกัน ผู้จำหน่ายส่วนใหญ่เสนอ Keyed MD5 ที่มี DES หรือ HMAC MD5 ที่มี DES เซ็ตย่อยนี้ทำงานกับการนำใช้ IP Security เกือบทั้งหมด

โพรซีเดอร์ที่ใช้ในการตั้งค่า manual tunnels ขึ้นอยู่กับว่าคุณกำลังตั้งค่าโฮสต์แรก ของ tunnel หรือโฮสต์ที่สอง ซึ่งต้องมี พารามิเตอร์ตรงกับการตั้งค่าโฮสต์แรก เมื่อตั้งค่าโฮสต์แรก คีย์สามารถสามารถ และอัลกอริทึมสามารถเป็นค่าดีฟอลต์ เมื่อตั้งค่าโฮสต์ที่สอง ให้อิมพอร์ตข้อมูล tunnel จาก ปลายทางรีโมต ถ้าเป็นไปได้

ข้อควรพิจารณาที่สำคัญอีกประการคือ การพิจารณาว่าระบบรีโมตอยู่ภายในไฟร์วอลล์หรือไม่ ถ้าอยู่ การตั้งค่าต้องรวมข้อมูลเกี่ยวกับไฟร์วอลล์ที่ขวางกั้นอยู่

การใช้ IKE กับ DHCP หรือแอดเดรสที่กำหนดแบบไดนามิก:

สถานการณ์ทั่วไปสถานการณ์หนึ่งสำหรับการใช้ IP Security กับระบบปฏิบัติการ คือเมื่อระบบกำลังเตรียมข้อมูลเซสชัน IKE เบื้องต้นกับเซิร์ฟเวอร์ และ identity ของระบบไม่สามารถผูกเข้ากับ IP address เฉพาะได้

สถานการณ์นี้สามารถเกิดขึ้นได้ในสภาวะแวดล้อม Local Area Network (LAN) เช่น การใช้ IP Security เพื่อเชื่อมต่อเซิร์ฟเวอร์บน LAN และต้องการเข้ารหัส ข้อมูล การใช้งานทั่วไปอื่นๆ เกี่ยวข้องกับการต่อเลขหมายของรีโมตไคลเอ็นต์เข้ามาที่เซิร์ฟเวอร์และ โดยใช้โดเมนเนมแบบเต็ม (FQDN) หรืออีเมลแอดเดรส (user@FQDN) เพื่อระบุ ID รีโมต

ในเฟส Key Management phase (เฟส 1) RSA Signature เป็นโหมดการพิสูจน์ตัวตนโหมดเดียวเท่านั้นที่สนับสนุนถ้าคุณใช้โหมดหลัก กับ ID ที่ไม่ใช่ IP address หรือกล่าวอีกอย่าง ถ้าคุณต้องการใช้การพิสูจน์ตัวตนคีย์ที่แบ่งใช้ล่วงหน้า คุณต้องใช้โหมด aggressive หรือโหมดหลักที่มี IP แอดเดรส เป็น IDs ในความจริงๆ เมื่อจำนวนไคลเอ็นต์ DHCP ที่คุณต้องการสร้าง IPsec tunnels ด้วยมีจำนวนมาก ทำให้กำหนดคีย์ที่แบ่งใช้ล่วงหน้าเป็นค่าเฉพาะได้ยาก สำหรับแต่ละไคลเอ็นต์ DHCP ดังนั้น

ขอแนะนำให้คุณใช้การพิสูจน์ตัวตน RSA Signature ในสถานการณ์นี้ คุณยังสามารถใช้ Group ID เป็น ID รีโมตในนิยาม tunnel เพื่อที่คุณกำหนด tunnel เพียงครั้งเดียวกับไคลเอ็นต์ DHCP ทั้งหมด (ดูไฟล์ตัวอย่าง นิยาม tunnel `/usr/samples/ipsec/group_aix_responder.xml`) Group ID เป็นคุณลักษณะเฉพาะของ AIX IPsec คุณสามารถกำหนด ID กลุ่มเพื่อรวม IKE IDs ใดๆ (เหมือน IP address เดียว), FQDN, User FQDN ช่วงหรือชุดของ addresses และอื่นๆ จากนั้นใช้ Group ID นี้เป็น ID รีโมต ของเฟส 1 หรือเฟส 2 ในนิยาม tunnel ของคุณ

หมายเหตุ: เมื่อใช้ Group ID ควรกำหนด tunnel ให้มีบทบาทเป็น Responder เท่านั้น ซึ่งหมายความว่า คุณต้องเรียกทำงาน tunnel นี้จากฝั่งไคลเอ็นต์ DHCP

สำหรับ เฟส Data Management (เฟส 2) เมื่อการรวมกลุ่ม IP Security กำลัง ถูกสร้างขึ้นเพื่อเข้ารหัสการรับส่งข้อมูล TCP หรือ UDP สามารถตั้งค่า tunnel การจัดการข้อมูลทั่วไป ดังนั้นการร้องขอใดๆ ที่ถูกพิสูจน์ตัวตนระหว่างเฟส 1 จะใช้ tunnel ทั่วไป สำหรับเฟส Data Management ที่กำหนด ถ้า IP address ไม่ถูกตั้งค่าไว้อย่างชัดเจนในฐานข้อมูล นี้อนุญาตให้แอดเดรสใดๆ จับคู่ tunnel ทั่วไปและสามารถใช้ได้จนกว่าการตรวจสอบความถูกต้องของความปลอดภัย ที่ยึดตามพับลิกคีย์อย่างเข้มงวดทำสำเร็จในเฟส 1

การใช้ XML เพื่อกำหนด tunnel การจัดการข้อมูลทั่วไป:

คุณสามารถกำหนด tunnel การจัดการข้อมูลทั่วไปโดยใช้รูปแบบ XML ที่ **ikedb** เข้าใจ

ดูในส่วนที่ชื่อ "อินเตอร์เฟสบรรทัดคำสั่งสำหรับการตั้งค่า IKE tunnel" ในหน้า 254 เพื่อ ดูข้อมูลเพิ่มเติมเกี่ยวกับอินเตอร์เฟส IKE XML และคำสั่ง **ikedb** tunnels การจัดการข้อมูลทั่วไปถูกใช้กับ DHCP รูปแบบ XML ใช้ชื่อแท็ก IPSecTunnel นี้ยังถูกอ้างอิงเป็น *tunnel เฟส 2* ในบริบทอื่นๆ *tunnel การจัดการข้อมูล ทั่วไป* ไม่ใช่ tunnel แท้จริง แต่ IPSecProtection ที่ ถูกใช้ถ้าข้อความการจัดการข้อมูลขาเข้า (ภายใต้ Key Management tunnel ที่ระบุ) ไม่ตรงกับ tunnel การจัดการข้อมูลใดๆ ที่กำหนดสำหรับ Key Management tunnel นั้น โดยถูกใช้ในกรณีที่ระบบ AIX เป็นผู้ตอบกลับเท่านั้น การระบุ tunnel การจัดการข้อมูลทั่วไป IPSecProtection เป็น ทางเลือก

tunnel การจัดการข้อมูลทั่วไปถูกกำหนดในองค์ประกอบ IKEProtection มีแอ็ตทริบิวต์ XML สองแอ็ตทริบิวต์ชื่อ *IKE_IPSecDefaultProtectionRef* และ *IKE_IPSecDefaultAllowedTypes* ที่ใช้ในที่นี้

อันดับแรก คุณจำเป็นต้องกำหนด IPSecProtection ที่ คุณต้องการใช้เป็นค่าดีฟอลต์ถ้าไม่มี IPSecTunnels (tunnel การจัดการข้อมูล) ตรง IPSecProtection ที่จะ ถูกใช้เป็นค่าดีฟอลต์ต้องมี IPSec_ProtectionName ที่ ขึ้นต้นด้วย `_defIPSprot_`

ถึงตอนนี้ไปที่ IKEProtection ที่ คุณจะใช้ IPSecProtection ดีฟอลต์นี้ ระบุ แอ็ตทริบิวต์ **IKE_IPSecDefaultProtectionRef** ที่มีชื่อของ IPSec_Protection ดีฟอลต์

คุณยังต้องระบุ ค่าสำหรับแอ็ตทริบิวต์ **IKE_IPSecDefaultAllowedTypes** ใน IKEProtection นี้ ซึ่งมีค่าได้มากกว่าหนึ่งค่า สำหรับค่าต่อไปนี้ (ถ้ามีหลายค่า ควร คั่นด้วยช่องว่าง):

```
Local_IPV4_Address
Local_IPV6_Address
Local_IPV4_Subnet
Local_IPV6_Subnet
Local_IPV4_Address_Range
Local_IPV6_Address_Range
Remote_IPV4_Address
Remote_IPV6_Address
```

```
Remote_IPV4_Subnet
Remote_IPV6_Subnet
Remote_IPV4_Address_Range
Remote_IPV6_Address_Range
```

ค่าเหล่านี้สอดคล้องกับ ประเภท ID ที่ระบุโดย initiator ในการเจรจา IKE IDs แท้จริงจะถูกข้าม IPSecProtection ที่ระบุถูกใช้ ถ้าแอ็ตทริบิวต์ **IKE_IPSecDefaultAllowedTypes** มีสตริงที่ขึ้นต้น ด้วย `Local_` ที่สอดคล้องกับประเภท ID โลคัลของ initiator และมีสตริงที่ขึ้นต้นด้วย `Remote_` ที่สอดคล้อง กับประเภท ID รีโมตของ initiator หรือกล่าวอีกนัยหนึ่ง อย่างน้อยคุณต้องมี ค่า `Local_` หนึ่งค่าและค่า `Remote_` อย่างน้อยหนึ่งค่า ในแอ็ตทริบิวต์ **IKE_IPSecDefaultAllowedTypes** ใดๆ เพื่อใช้ IPSec_Protection ที่ สอดคล้อง

ตัวอย่าง tunnel การจัดการข้อมูลทั่วไป:

Data Management tunnel สามารถใช้ส่งข้อความไปยังระบบ

initiator ส่งสิ่งต่อไปนี้ไปยังระบบ AIX ในข้อความเฟส 2 (Data Management):

```
local ID type:    IPV4_Address
local ID:         192.168.100.104

remote ID type:   IPV4_Subnet
remote ID:        10.10.10.2
remote netmask:   255.255.255.192
```

ระบบ AIX system ไม่มี Data Management tunnel ที่ตรงกับ IDs เหล่านี้ แต่มี IPSecProtection ที่มี แอ็ตทริบิวต์ต่อไปนี้ถูกกำหนด:

```
IKE_IPSecDefaultProtectionRef="_defIPSprot_protection4"
IKE_IPSecDefaultAllowedTypes="Local_IPV4_Address
                              Remote_IPV4_Address
                              Remote_IPV4_Subnet
                              Remote_IPV4_Address_Range"
```

ประเภท ID โลคัลของข้อความขาเข้า IPV4_Address ตรงกับหนึ่งในค่า Local_ ของประเภทที่อนุญาต Local_IPV4_Address รวมทั้ง ID รีโมตของข้อความ IPV4_Subnet ตรงกับ ค่า Remote_IPV4_Subnet ดังนั้นการเจรจา Data Management tunnel จะดำเนินต่อไปโดยมี `_defIPSprot_protection4` เป็น IPSecProtection

ไฟล์ `/usr/samples/ipsec/default_p2_policy.xml` คือไฟล์ XML แบบเต็มที่กำหนด IPSecProtection ทั่วไปที่ สามารถใช้เป็นตัวอย่าง

การตั้งค่า Internet key exchange tunnels

คุณสามารถกำหนดคอนฟิก Internet Key Exchange (IKE) tunnels โดยใช้ System Management Interface Tool (SMIT) หรือบรรทัดรับคำสั่ง

การใช้อินเตอร์เฟส SMIT สำหรับการตั้งค่า IKE tunnel:

คุณสามารถใช้อินเตอร์เฟส SMIT เพื่อตั้งค่า IKE tunnels และใช้งาน ฟังก์ชันฐานข้อมูล IKE ระดับต้น

SMIT ใช้ฟังก์ชันคำสั่ง XML ที่สำคัญเพื่อดำเนินการเพิ่ม การลบ และการแก้ไขในนิยาม IKE tunnel IKE SMIT ถูกใช้ในการตั้งค่า IKE tunnels อย่างรวดเร็ว และให้ตัวอย่างของไวยากรณ์ XML ที่ใช้สร้างนิยาม IKE tunnel เมนู IKE SMIT ยังอนุญาตให้คุณ สำรองข้อมูล เรียกคืน และเตรียมข้อมูลเบื้องต้นสำหรับฐานข้อมูล IKE

ในการตั้งค่า IPv4 IKE tunnel ใช้พาธด่วน **smitty ike4** ในการตั้งค่า IPv6 IKE tunnel ใช้พาธด่วน **smitty ike6** ฟังก์ชัน ฐานข้อมูล IKE พบได้ในเมนู Advanced IP Security Configuration

อินเตอร์เฟสบรรทัดคำสั่งสำหรับการตั้งค่า IKE tunnel:

คำสั่ง **ikedb** อนุญาตให้ผู้ใช้เรียกค้น อัพเดต ลบ อิมพอร์ต และเอ็กซ์พอร์ตข้อมูลในฐานข้อมูล IKE โดยใช้ XML อินเตอร์เฟส

คำสั่ง **ikedb** อนุญาตให้ผู้ใช้เขียน (put) หรืออ่านจาก (get) ฐานข้อมูล IKE รูปแบบอินพุตและเอาต์พุต คือไฟล์ Extensible Markup Language (XML) รูปแบบของไฟล์ XML ถูกระบุโดย Document Type Definition (DTD) คำสั่ง **ikedb** อนุญาตให้ผู้ใช้ดู DTD ที่ใช้ในการตรวจสอบความถูกต้องไฟล์ XML เมื่อจะทำ put ขณะนี้การประกาศ entity สามารถเพิ่มใน DTD โดยใช้แฟล็ก **-e** วิธีนี้เป็นการแก้ไข DTD เพียงวิธีเดียว ที่สามารถทำได้ การประกาศ DOCTYPE ภายนอกใดๆ ในไฟล์ XML อินพุตจะถูกข้ามและการประกาศ DOCTYPE ภายในใดๆ อาจส่งผล ให้เกิดข้อผิดพลาด กฎที่ใช้แยกวิเคราะห์ไฟล์ XML ที่ใช้ DTD ถูกระบุในมาตรฐาน XML ไฟล์ `/usr/samples/ipsec` มีตัวอย่างของไฟล์ XML ปกติที่กำหนดสถานการณ์ tunnel ทั่วไป ดูที่รายละเอียดคำสั่ง **ikedb** ใช้ *การอ้างอิงคำสั่ง* สำหรับ รายละเอียดไวยากรณ์

คุณสามารถใช้คำสั่ง **ike** เพื่อเริ่มทำงาน หยุดทำงาน และมอนิเตอร์ IKE tunnels คำสั่ง **ike** ยังสามารถ ใช้เพื่อเรียกทำงาน, ลบ หรือแสดงรายการ IKE และ IP Security tunnels ดูที่รายละเอียดคำสั่ง **ike** ใช้ *การอ้างอิงคำสั่ง* สำหรับ รายละเอียดไวยากรณ์

ตัวอย่างต่อไปนี้แสดงวิธีใช้ **ike, ikedb** และคำสั่งอื่นๆ มากมายเพื่อตั้งค่าและตรวจสอบสถานะของ IKE tunnel ของคุณ:

1. ในการเริ่มทำงานการเจรจา tunnel (*เรียกทำงาน* tunnel) หรือ การอนุญาตให้ระบบขาเข้าทำหน้าที่เป็นผู้ตอบกลับ (ขึ้นอยู่กับ บทบาทที่ระบุ) ใช้คำสั่ง **ike** พร้อม หมายเลข tunnel ดังนี้:

   ```
   # ike cmd=activate numlist=1
   ```

   คุณ ยังสามารถใช้ id รีโมตหรือ IP addresses ดังแสดงในตัวอย่าง ต่อไปนี้:

   ```
   # ike cmd=activate remid=9.3.97.256
   # ike cmd=activate ipaddr=9.3.97.100, 9.3.97.256
   ```

   เนื่องจาก อาจใช้เวลาสักครู่เพื่อให้คำสั่งดำเนินการเสร็จสมบูรณ์ คำสั่ง คืนค่ากลับมาเมื่อการเจรจาเริ่มทำงาน
2. ในการแสดงสถานะ tunnel ใช้คำสั่ง **ike** ดังนี้:

   ```
   # ike cmd=list
   ```

   เอาต์พุต คล้ายกับตัวอย่างต่อไปนี้:

   ```
   Phase 1 Tunnel ID       [1]
   Phase 2 Tunnel ID       [1]
   ```

   เอาต์พุตแสดง tunnel เฟส 1 และเฟส 2 ที่แอ็คทีฟอยู่ขณะนี้
3. ในการรายการโดยละเอียดของ tunnel ใช้คำสั่ง **ike** ดังนี้:

   ```
   # ike cmd=list verbose
   ```

   เอาต์พุต คล้ายกับตัวอย่างต่อไปนี้:

   ```
   Phase 1 Tunnel ID       1
   Local ID Type:          Fully_Qualified_Domain_Name
   Local ID:               bee.austin.ibm.com
   ```

```
Remote ID Type:         Fully_Qualified_Domain_Name
Remote ID:              ipsec.austin.ibm.com
Mode:                   Aggressive
Security Policy:        BOTH_AGGR_3DES_MD5
Role:                   Initiator
Encryption Alg:         3DES-CBC
Auth Alg:               Preshared Key
Hash Alg:               MD5
Key Lifetime:           28800 Seconds
Key Lifesize:           0 Kbytes
Key Rem Lifetime:       28737 Seconds
Key Rem Lifesize:       0 Kbytes
Key Refresh Overlap:    5%
Tunnel Lifetime:        2592000 Seconds
Tunnel Lifesize:        0 Kbytes
Tun Rem Lifetime:       2591937 Seconds
Status:                 Active

Phase 2 Tunnel ID       1
Local ID Type:          IPv4_Address
Local ID:               10.10.10.1
Local Subnet Mask:      N/A
Local Port:             any
Local Protocol:         all
Remote ID Type:         IPv4_Address
Remote ID:              10.10.10.4
Remote Subnet Mask:     N/A
Remote Port:            any
Remote Portocol:        all
Mode:                   Oakley_quick
Security Policy:        ESP_3DES_MD5_SHA_TUNNEL_NO_PFS
Role:                   Initiator
Encryption Alg:         ESP_3DES
AH Transform:           N/A
Auth Alg:               HMAC-MD5
PFS:                    No
SA Lifetime:            600 Seconds
SA Lifesize:            0 Kbytes
SA Rem Lifetime:        562 Seconds
SA Rem Lifesize:        0 Kbytes
Key Refresh Overlap:    15%
Tunnel Lifetime:        2592000 Seconds
Tunnel Lifesize:        0 Kbytes
Tun Rem Lifetime:       2591962 Seconds
Assoc P1 Tunnel:        0
Encap Mode:             ESP_tunnel
Status:                 Active
```

4. ในการแสดงกฎตัวกรองในตารางการกรองไดนามิกสำหรับ IKE tunnel ที่เพิ่งถูกเรียกทำงาน ใช้คำสั่ง **lsfilt** ดังนี้:

   ```
   # lsfilt -d
   ```

   เอาต์พุต คล้ายกับตัวอย่างต่อไปนี้:

```
1 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 no udp eq 4001 eq 4001 both both no all
  packets 0 all
2 *** กฎการแทนที่การกรองไดนามิก *** no
0 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 yes all any 0 any 0 both both no all
  packets 0 all

*** Dynamic table ***

0 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 no udp eq 500 eq 500 local both no all
  packets 0
0 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 no ah any 0 any 0 both inbound no all
  packets 0
0 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 no esp any 0 any 0 both inbound no all
  packets 0
1 permit 10.10.10.1 255.255.255.255 10.10.10.4 255.255.255.255 no all any 0 any
  0 both outbound yes all packets 1
1 permit 10.10.10.4 255.255.255.255 10.10.10.1 255.255.255.255 no all any 0 any
  0 both inbound yes all packets 1
```

ตัวอย่างนี้ แสดงเครื่องที่มีหนึ่ง IKE tunnel และไม่มี tunnels อื่นอีก กฎ การแทนที่การกรองไดนามิก (กฎ #2 ในเอาต์พุตตัวอย่างนี้ของ ตารางสแตติก) สามารถถูกย้ายโดยผู้ใช้เพื่อควบคุมการจัดวางที่สัมพันธ์ กับกฎที่ผู้ใช้กำหนดเองข้ออื่นทั้งหมด กฎในตารางไดนามิกถูกประกอบ ขึ้นเป็น tunnels โดยอัตโนมัติคือกฎที่เจรจาและสอดคล้องกัน ถูกแทรกลงในตารางการกรอง กฎเหล่านี้สามารถแสดงได้ แต่ไม่สามารถแก้ไข

5. ในการเปิดใช้การบันทึกการทำงานของกฎตัวกรองไดนามิก ให้ตั้งค่าอ็อพชัน การ บันทึกการทำงานสำหรับกฎ #2 เป็น Yes ใช้คำสั่ง **chfilt** ดังแสดงในตัวอย่างต่อไปนี้:

   ```
   # chfilt -v 4 -n 2 -l y
   ```

   สำหรับ รายละเอียดเพิ่มเติมเกี่ยวกับการบันทึกการทำงาน IKE traffic ดูที่ “สิ่งอำนวยความสะดวกการบันทึกการทำงาน” ในหน้า 282

6. ในการปิดทำงาน tunnel ใช้คำสั่ง **ike** ดังนี้:

   ```
   # ike cmd=remove numlist=1
   ```

7. ในการดูนิยาม tunnel ใช้คำสั่ง **ikedb** ดังนี้:

   ```
   # ikedb -g
   ```

8. ในการใส่นิยามในฐานข้อมูล IKE จากไฟล์ XML ที่ถูก สร้างขึ้นบนเครื่องเพียร์ และบันทึกทับอ็อบเจ็กต์ใดๆ ที่มีอยู่ในฐานข้อมูลด้วยชื่อเดียวกัน ใช้คำสั่ง **ikedb** ดังนี้:

   ```
   # ikedb -pFs peer_tunnel_conf.xml
   ```

   `peer_tunnel_conf.xml` คือ ไฟล์ XML ที่สร้างขึ้นบนไฟล์เพียร์

9. ในการรับนิยามของ tunnel เฟส 1 ชื่อ *tunnel_sys1_and_sys2* และ tunnels เฟส 2 ที่ขึ้นต่อกันทั้งหมดที่มีข้อเสนอและการป้องกันตามลำดับ ใช้คำสั่ง **ikedb** ดังนี้:

   ```
   # ikedb -gr -t IKETunnel -n tunnel_sys1_and_sys2
   ```

10. ในการลบคีย์ที่แบ่งใช้ไว้แล้วทั้งหมดออกจากฐานข้อมูล ใช้คำสั่ง **ikedb** ดังนี้:

    ```
    # ikedb -d -t IKEPresharedKey
    ```

สำหรับข้อมูลทั่วไปเกี่ยวกับการสนับสนุนกลุ่ม IKE tunnel โปรดดูที่ “การสนับสนุนกลุ่ม” ในหน้า 257 คุณสามารถใช้คำสั่ง **ikedb** เพื่อกำหนดกลุ่ม จากบรรทัดคำสั่ง

AIX IKE และ Linux affinity:

คุณสามารถกำหนดคอนฟิก AIX IKE tunnel โดยใช้ไฟล์คอนฟิกูเรชัน Linux

เมื่อต้องการกำหนดคอนฟิก AIX IKE tunnel โดยใช้ไฟล์คอนฟิกูเรชัน Linux, ให้ใช้คำสั่ง **ikedb** พร้อมกับแฟล็ก **-c** (อ็อพชันการแปลง), ซึ่งอนุญาตให้คุณใช้ไฟล์คอนฟิกูเรชัน `/etc/ipsec.conf` และ `/etc/ipsec.secrets` Linux เป็นนิยาม IKE คำสั่ง **ikedb** แยกวิเคราะห์ไฟล์คอนฟิกูเรชัน Linux จะสร้าง ไฟล์ XML และอาจเลือกเพิ่มข้อกำหนด XML tunnel ลงใน ฐานข้อมูล IKE จากนั้นคุณสามารถดูนิยาม tunnel ได้โดยใช้คำสั่ง **ikedb -g**

การสนับสนุนกลุ่ม:

IP security สนับสนุนการจัดกลุ่ม IKE IDs ในนิยาม tunnel เพื่อ เชื่อมโยงหลาย IDs ด้วยนโยบายการรักษาความปลอดภัยเดียว โดยไม่ต้องสร้าง นิยาม tunnel แยกต่างหาก

การจัดกลุ่มมีประโยชน์อย่างมากเมื่อตั้งค่าการเชื่อมต่อไปยังโฮสต์รีโมต หลายโฮสต์ เนื่องจากคุณสามารถเลี่ยงการตั้งค่าหรือการจัดการนิยาม tunnel หลายๆ นิยาม รวมทั้ง ถ้าต้องมีการเปลี่ยนแปลงในนโยบายการรักษาความปลอดภัย คุณไม่จำเป็นต้องเปลี่ยน นิยาม tunnel หลายนิยาม

กลุ่มต้องถูกกำหนดก่อนการใช้ชื่อกลุ่มในนิยาม tunnel ขนาดของกลุ่มจำกัดไว้ที่ 1 KB ที่ด้านของผู้เริ่มต้นการเจรจา คุณสามารถใช้กลุ่มเป็น ID รีโมตในนิยาม tunnel การจัดการข้อมูลเท่านั้น ที่ด้านผู้ตอบการเจรจา คุณสามารถใช้กลุ่มเป็น ID รีโมตในการจัดการคีย์และนิยาม tunnel การจัดการข้อมูล

กลุ่มประกอบด้วยชื่อกลุ่มและรายการ IKE IDs และประเภท ID IDs สามารถเป็นประเภทเดียวกัน หรือผสมกันระหว่างตัวเลือกต่อไปนี้:

- IPv4 addresses
- IPv6 addresses
- FQDN
- user@FQDN
- ประเภท X500 DN

ระหว่างการเจรจา Security Association IDs ในกลุ่in a group are searched linearly for the first match.

โปรดอ้างถึง “อินเตอร์เฟสบรรทัดคำสั่งสำหรับการตั้งค่า IKE tunnel” ในหน้า 254 สำหรับข้อมูลเกี่ยวกับการนิยามกลุ่มจากบรรทัดรับคำสั่ง

สถานการณ์การตั้งค่า IKE tunnel:

สถานการณ์ต่อไปนี้อธิบายประเภทของสถานการณ์ ที่ลูกค้าส่วนใหญ่ประสบเมื่อพยายามตั้งค่า tunnels สถานการณ์เหล่านี้สามารถอธิบายเป็นกรณีสำนักงานสาขา คู่ค้าธุรกิจ และ การเข้าถึงแบบรีโมต

- ในกรณีสำนักงานสาขา ลูกค้ามีเน็ตเวิร์กที่ไว้วางใจสองเน็ตเวิร์ก ที่ต้องการเชื่อมต่อกัน กลุ่มวิศวกรของที่ตั้งหนึ่ง ไปยังกลุ่มวิศวกรของอีกที่หนึ่ง ในตัวอย่างนี้ มีเกตเวย์ ที่เชื่อมต่อซึ่งกันและกัน และการรับส่งข้อมูลทั้งหมดที่ส่งระหว่าง เกตเวย์จะใช้ tunnel เดียวกัน ปริมาณการรับส่งที่ปลายแต่ละด้านของ tunnel ถูกแยกส่วนและส่งเป็นแบบข้อความธรรมดาภายในอินทราเน็ตของบริษัท

ในเฟสแรกของการเจรจา IKE ความเชื่อมโยงด้านความปลอดภัย IKE ถูกสร้างระหว่างสองเกตเวย์ ปริมาณการรับส่งข้อมูลที่ส่งใน IP Security tunnel คือปริมาณการรับส่งระหว่างสองซับเน็ต และ IDs ซับเน็ตถูกใช้ในการเจรจาเฟส 2 หลังป้อนนโยบายการรักษาความปลอดภัยและพารามิเตอร์ tunnel สำหรับ tunnel แล้ว จะสร้าง หมายเลข tunner ขึ้น ใช้คำสั่ง **ike** เพื่อเริ่มทำงาน tunnel

- ในสถานการณ์คู่ค้าธุรกิจ เน็ตเวิร์กไม่ได้รับความไว้วางใจ และผู้บริหารเน็ตเวิร์กอาจต้องการจำกัดการเข้าถึงให้แก่โฮสต์จำนวนน้อยที่อยู่เบื้องหลังเกตเวย์ด้านความปลอดภัย ในกรณีนี้ tunnel ระหว่างโฮสต์จะส่งข้อมูลที่ป้องกันโดย IP Security เพื่อใช้ ระหว่างโฮสต์เฉพาะสองโฮสต์ โปรโตคอลของ tunnel เฟส 2 คือ AH หรือ ESP tunnel โฮสต์-ถึง-โฮสต์นี้ได้รับการรักษาความปลอดภัยภายในเกตเวย์-ต่อ-เกตเวย์
- ในกรณีการเข้าถึงแบบรีโมต tunnels ถูกตั้งค่าตามต้องการและ ใช้การรักษาความปลอดภัยระดับสูง IP addresses อาจไม่สื่อความหมาย ดังนั้น จึงควรใช้โดเมนเนมแบบเต็มหรือ *user@* fully qualified domain names มากกว่า ทางเลือกคุณสามารถใช้ KEYID เพื่อเชื่อมโยง คีย์กับ ID โฮสต์

ใบรับรองดิจิตัลและแนวคิดตัวจัดการคีย์

ใบรับรองดิจิตัลจะโยง identity เข้ากับพับลิกคีย์ ด้วยวิธี ซึ่งคุณสามารถตรวจสอบผู้ส่งหรือผู้รับของการถ่ายโอนที่เข้ารหัสได้

IP Security ใช้ใบรับรองดิจิทัลเพื่อเปิดใช้งาน *วิทยาการเข้ารหัสลับด้วยพับลิก*, ซึ่งรู้จักกันในนามของ *วิทยาการเข้ารหัสลับแบบอสมมาตร*, ซึ่งเข้ารหัสข้อมูล โดยใช้ไพรเวตคีย์ที่ผู้ใช้ทราบและถอดรหัสโดยใช้ พับลิกคีย์ที่เชื่อมโยง (แบ่งใช้) จากคู่พับลิก-ไพรเวตคีย์ *คู่คีย์* คือสตริงข้อมูล ขนาดยาวที่ทำหน้าที่เป็นคีย์สำหรับ scheme การเข้ารหัสของผู้ใช้

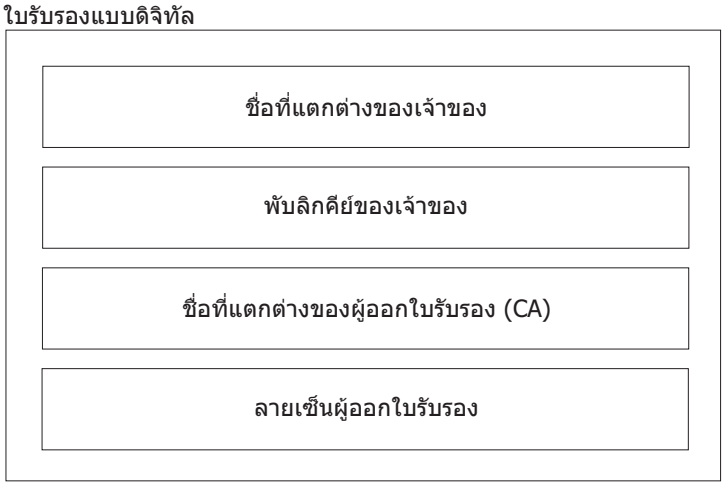
ในวิทยาการเข้ารหัสลับพับลิกคีย์ พับลิกคีย์จะถูกมอบให้แก่บุคคลที่ ผู้ใช้ต้องการสื่อสารด้วย ผู้ส่งลงนามแบบดิจิตัลการสื่อสารที่มีความปลอดภัย ทั้งหมดด้วยไพรเวตคีย์ที่สอดคล้องกับคู่คีย์ที่กำหนด ผู้รับ ใช้พับลิกคีย์เพื่อตรวจสอบลายเซ็นของผู้ส่ง ถ้าข้อความถูกถอดรหัส เสร็จเรียบร้อยโดยใช้พับลิกคีย์ ผู้รับสามารถตรวจสอบได้ว่าผู้ส่ง ได้ถูกพิสูจน์ตัวตนแล้ว

วิทยาการเข้ารหัสลับแบบพับลิกคีย์จะขึ้นกับ *certification authorities* (*CAs*) ที่ไว้วางใจและเป็นของบุคคลที่สาม เพื่อออกใบรับรองดิจิตัลที่เชื่อถือได้ ผู้รับ ระบุว่าองค์กรการออกใบรับรองใด หรือหน่วยงานใดที่ถือว่า ได้รับความไว้วางใจ ใบรับรองถูกออกมาเพื่อใช้ในช่วงเวลาหนึ่งที่ระบุ เมื่อเลยวันหมดอายุ ใบรับรองนั้นต้องถูกเปลี่ยนใหม่

AIX จัดเตรียมเครื่องมือ Key Manager, ซึ่งจัดการกับใบรับรองดิจิทัล ส่วน ต่อไปนี้จัดให้มีข้อมูลเกี่ยวกับแนวคิดเกี่ยวกับใบรับรอง

รูปแบบของใบรับรองดิจิตัล:

ใบรับรองดิจิตัลมีส่วนข้อมูลที่เฉพาะเจาะจง เกี่ยวกับ identity ของเจ้าของใบรับรองและเกี่ยวกับ certification authority ดูรูปภาพต่อไปนี้เพื่อดูภาพประกอบของใบรับรอง ดิจิตัล



เนื้อหาของใบรับรองดิจิทัล

รูปที่ 10. เนื้อหาของใบรับรองดิจิตัล

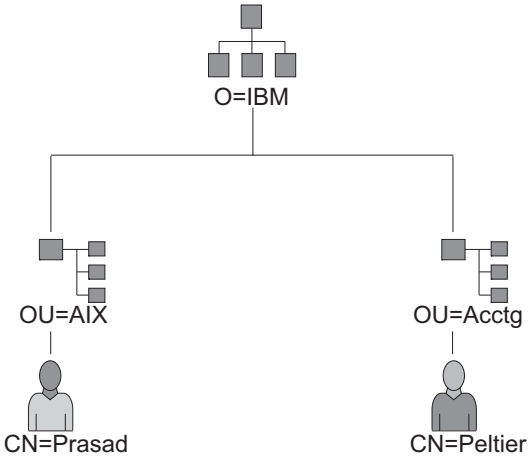
ภาพประกอบนี้แสดงสี่ส่วนของใบรับรองดิจิตัล จากด้านบนคือ Distinguished Name ของเจ้าของ พับลิกคีย์ของเจ้าของ Distinguished Name ของผู้ออก (CA) และลายเซ็นของผู้ออก

รายการต่อไปนี้ยังอธิบายถึงเนื้อหาของใบรับรอง ดิจิตัล:

Distinguished Name ของเจ้าของ

การรวมชื่อของเจ้าของ กับบริบทแวดล้อม (ตำแหน่ง) ในลำต้นไดเร็กทอรี ในรูปภาพต่อไปนี้ของลำต้นไดเร็กทอรีอย่างง่าย ตัวอย่างเช่น Prasad เป็นชื่อของเจ้าของ และบริบทแวดล้อมคือ ประเทศ=US, องค์กร=ABC, องค์กรย่อย=SERV ดังนั้น distinguished name คือ:

```
/C=US/O=ABC/OU=SERV/CN=prasad.austin.ibm.com
```



ตัวอย่างของการได้รับชื่อที่ไม่ซ้ำกันจากไดเร็กทอรีทรี

รูปที่ 11. ตัวอย่างของการรับ Distinguished Name จากลำต้นไดเร็กทอรี

ภาพประกอบนี้คือลำต้นไดเร็กทอรีที่มี O=ABC ที่ระดับ บนสุดและแตกสาขาออกเป็นสองหน่วยที่ระดับที่สอง ระดับที่สอง มี OU=AIX และ OU=Acctg อยู่บนสาขาแยกกัน แต่ละสาขามีสาขาที่ นำไปสูงหน่วยเดียวบนระดับล่าสุด ระบบสุดท้ายมี CN=Prasad และ CN=Peltier ตามลำดับ

พับลิกคีย์ของเจ้าของ
ใช้โดยผู้รับเพื่อถอดรหัสข้อมูล

Subject Alternate Name
สามารถเป็น identifier เช่น IP address อีเมลแอดเดรส โดเมนเนม แบบเต็ม และอื่น

วันที่ออก
วันที่ออกใบรับรองดิจิตัล

วันหมดอายุ
วันที่ที่ใบรับรองดิจิตัลจะหมดอายุ

Distinguished Name ของผู้ออก
Distinguished name ของ Certification Authority

ลายเซ็นดิจิตัลของผู้ออก
ลายเซ็นดิจิตัลที่ใช้ตรวจสอบความถูกต้องของใบรับรอง

ข้อควรพิจารณาเกี่ยวกับความปลอดภัยสำหรับใบรับรองดิจิตัล:

ใบรับรองดิจิตัลเพียงอย่างเดียวไม่สามารถใช้พิสูจน์ identity ได้

ใบรับรองดิจิตัลอนุญาตให้คุณตรวจสอบ identity ของเจ้าของ ใบรับรองดิจิตัลโดยการให้พับลิกคีย์ที่จำเป็นสำหรับการตรวจสอบ พับลิกดิจิตัลของเจ้าของ คุณสามารถส่งพับลิกคีย์ของคุณไปให้คนอื่นได้อย่างปลอดภัย เนื่องจากข้อมูลของคุณไม่สามารถถูกถอดรหัสได้โดยขาดส่วนใดส่วนหนึ่งของคู่คีย์ คือ ไพรเวตคีย์ของคุณ ดังนั้น เจ้าของต้องปกป้องไพรเวตคีย์ที่เป็นคู่ของ พับลิกคีย์ในใบรับรองดิจิตัล การสื่อสารทั้งหมดของเจ้าของ ใบรับรองดิจิตัลสามารถถอดรหัสได้ ถ้าทราบไพรเวตคีย์ หากปราศจาก ไพรเวตคีย์ ใบรับรองดิจิตัลจะไม่สามารถนำไปใช้งานทางที่ผิดได้

Certification authorities และลำดับชั้นการไว้วางใจ:

ใบรับรองดิจิตัลมีความเชื่อถือได้เทียบเท่ากับ certification authority (CA) ที่ออกใบรับรอง

ในฐานะส่วนหนึ่งของการไว้วางใจนี้ จึงควรทำความเข้าใจในนโยบายที่อยู่ภายใต้ ใบรับรองที่ออก แต่ละองค์กรหรือผู้ใช้ต้องพิจารณาว่า certification authorities ใดสามารถยอมรับว่าเป็นที่เชื่อถือได้

เครื่องมือ Key Manager ยังอนุญาตให้องค์กรสร้างใบรับรองที่ลงนามเอง ซึ่งเป็นประโยชน์สำหรับการทดสอบหรือในสภาวะแวดล้อมที่มีจำนวนผู้ใช้ หรือจำนวนเครื่องน้อย

ในฐานะผู้ใช้เซอร์วิสด้านความปลอดภัย คุณจำเป็นต้องทราบพับลิกคีย์เพื่อจัดหา และตรวจสอบความถูกต้องของใบรับรองดิจิตัล รวมถึงการรับใบรับรองดิจิตัลโดยง่าย ไม่ช่วยให้เชื่อในความถูกต้องของใบรับรอง ในการตรวจสอบความถูกต้องของใบรับรอง คุณจำเป็น ต้องใช้พับลิกคีย์ของ certification authority ที่ออกใบรับรองดิจิตัลนั้น ถ้าคุณยังไม่ได้ถือครองสำเนาพับลิกคีย์ของ CA คุณอาจ ต้องใช้ใบรับรองดิจิตัลเพิ่มเพื่อขอรับพับลิกคีย์ของ CA

รายการการเพิกถอนใบรับรอง:

ใบรับรองดิจิตัลคาดว่าจะถูกใช้ตลอดระยะเวลา ที่ใช้ได้ อย่างไรก็ตามถ้าจำเป็น ใบรับรองสามารถถูกทำให้ไม่สามารถใช้ได้ ก่อนวันที่ หมดอายุจริงของใบรับรอง

การทำให้ใบรับรองไม่สามารถใช้ได้อาจจำเป็น เช่น ถ้าพนักงานลาออก หรือถ้าไพรเวตคีย์ของใบรับรอง รั่วไหล ในการทำให้ใบรับรองไม่สามารถใช้ได้ คุณต้องแจ้ง Certificate Authority (CA) ที่เกี่ยวข้องของเหตุการณ์ เมื่อ CA เพิกถอนใบรับรอง จะเพิ่มหมายเลขลำดับใบรับรองที่ไม่สามารถใช้ได้นั้นใน Certificate Revocation List (CRL)

CRLs คือโครงสร้างข้อมูลที่มีการลงนามที่ออกเป็นระยุ และสามารถดูได้ในที่เก็บพับลิก CRLs สามารถถูกเรียกออกมาจากเซิร์ฟเวอร์ HTTP หรือ LDAP แต่ละ CRL มีการประทับเวลาปัจจุบันและการประทับเวลา nextUpdate แต่ละใบรับรองที่ถูกเพิกถอนในรายจะถูกระบุโดยหมายเลขลำดับ ใบรับรอง

เมื่อตั้งค่า IKE tunnel และใช้ใบรับรองดิจิตัลเป็นวิธีการพิสูจน์ตัวตน ของคุณ คุณสามารถยืนยันว่าใบรับรองยังไม่ถูกเพิกถอนได้โดยการเลือก RSA Signature with CRL Checking ถ้า CRL Checking ถูกเปิดใช้งาน รายการจะถูกโหลด และตรวจสอบระหว่างกระบวนการเจรจาเพื่อสร้าง tunnel การจัดการคีย์

> **หมายเหตุ:** ในการใช้คุณลักษณะนี้ของ IP Security ระบบของคุณต้องถูกตั้งค่า เพื่อใช้เซิร์ฟเวอร์ SOCKS (เวอร์ชัน 4 สำหรับเซิร์ฟเวอร์ HTTP) และเซิร์ฟเวอร์ LDAP หรือทั้งสอง ถ้าคุณทราบว่ากำลังใช้เซิร์ฟเวอร์ SOCKS หรือ LDAP เพื่อขอรับ CRLs, คุณสามารถเพิ่มไปยังไฟล์ `/etc/isakmpd.conf`

ใช้เป็นใบรับรองดิจิตัลในอินเตอร์เน็ตแอ็พพลิเคชัน:

อินเตอร์เน็ตแอ็พพลิเคชันที่ใช้ระบบวิทยาการเข้ารหัสลับพับลิกคีย์ ต้องใช้ใบรับรองดิจิตัลเพื่อขอรับพับลิกคีย์

มีหลายแอ็พพลิเคชันที่ใช้วิทยาการเข้ารหัสลับพับลิกคีย์ รวมถึง รายการต่อไปนี้:

Virtual Private Networks (VPN)

Virtual Private Networks หรือที่เรียก *secure tunnels* สามารถถูกตั้งค่า ระหว่างระบบ เช่นไฟร์วอลล์เพื่อเปิดใช้การเชื่อมต่อที่มีการป้องกันระหว่าง เน็ตเวิร์กที่ปลอดภัยบนลิงก์การสื่อสารที่ไม่ปลอดภัย การรับส่งข้อมูลทั้งหมด ที่กำหนดไปยังเน็ตเวิร์กเหล่านี้ถูกเข้ารหัสระหว่างระบบที่เกี่ยวข้อง

โปรโตคอล ที่ใช้ในการ tunnel ยึดตามมาตรฐาน IP Security และ IKE ซึ่งอนุญาตสำหรับ การเชื่อมต่อที่เข้ารหัสและปลอดภัยระหว่างรีโมตไคลเอ็นต์ (ตัวอย่างเช่น พนักงาน กำลังทำงานจากที่บ้าน) และโฮสต์หรือเน็ตเวิร์กที่ปลอดภัย

Secure Sockets Layer (SSL)

SSL คือโปรโตคอลที่จัดให้มีความเป็นส่วนตัวและ integrity สำหรับการสื่อสาร ใช้โดยเว็บเซิร์ฟเวอร์สำหรับการเชื่อมต่อแบบปลอดภัยระหว่างเว็บเซิร์ฟเวอร์และ เว็บเบราว์เซอร์ โดย Lightweight Directory Access Protocol (LDAP) สำหรับการเชื่อมต่อแบบปลอดภัย ระหว่างไคลเอ็นต์ LDAP และเซิร์ฟเวอร์ LDAP และโดย Host-on-Demand V.2 สำหรับการเชื่อมต่อ ระหว่างไคลเอ็นต์และระบบโฮสต์ SSL ใช้ใบรับรองดิจิตัลสำหรับ แลกเปลี่ยนคีย์ การพิสูจน์ตัวตนเซิร์ฟเวอร์ และเป็นทางเลือก การพิสูจน์ตัวตนไคลเอ็นต์

Secure Electronic Mail

ระบบจดหมายอิเล็กทรอนิกส์หลายระบบ ใช้มาตรฐานเช่น PEM หรือ S/MIME สำหรับ จดหมายอิเล็กทรอนิกส์แบบปลอดภัย ใช้ใบรับรองดิจิตัลสำหรับลายเซ็นดิจิตัลและ สำหรับการแลกเปลี่ยนคีย์เพื่อเข้ารหัสและถอดรหัสข้อความจดหมาย

ใบรับรองดิจิตัลและการร้องขอใบรับรอง:

การร้องขอใบรับรอง ต้องถูกสร้างและส่งไปยัง CA เพื่อ ร้องของใบรับรองดิจิตัล

ใบรับรองดิจิตัลที่ลงนามมีฟิลด์สำหรับ distinguished name ของเจ้าของ พับลิกคีย์ของเจ้าของ distinguished name ของ CA และลายเซ็นของ CA ใบรับรองดิจิตัลที่ลงนามเองมี distinguished name ของเจ้าของ พับลิกคีย์ และลายเซ็น

การร้องขอใบรับรองมีฟิลด์สำหรับ distinguished name, พับลิกคีย์ และลายเซ็นของผู้ขอ CA ตรวจสอบลายเซ็นของผู้ขอ ด้วยพับลิกคีย์ในใบรับรองดิจิทัลเพื่อให้แน่ใจว่า:

- การร้องขอใบรับรองไม่ถูกแก้ไขในการส่งผ่านระหว่าง ผู้ร้องขอ และ CA
- ผู้ร้องขอเป็นเจ้าของพับลิกคีย์ที่เกี่ยวข้องสำหรับ พับลิกคีย์ที่อยู่ในการร้องขอใบรับรอง

CA ยังมีหน้าที่ในการตรวจสอบในบางระดับสำหรับ identity ของ ผู้ร้องขอ ข้อกำหนดสำหรับการตรวจสอบนี้อาจมีช่างตั้งแต่การพิสูจน์เล็กน้อย ไปจนถึงการรับประกันความถูกต้องของ identity ของเจ้าของโดยสมบูรณ์

เครื่องมือ Key Manager:

เครื่องมือ Key Manager จัดการใบรับรองดิจิตัล และอยู่ใน ชุดไฟล์ `gskkm.rte` บนแพ็กเสริม

ในการตั้งค่าการสนับสนุนใบรับรองดิจิตัลและลายเซ็น อย่างน้อยคุณต้อง ดำเนินงาน 1, 2, 3, 4, 6 และ 7 จากนั้น สร้าง IKE tunnel และเชื่อมโยงนโยบายกับ tunnel ที่ใช้ RSA Signature เป็นเมธอดการพิสูจน์ตัวตน

คุณสามารถสร้างและกำหนดคอนฟิกฐานข้อมูลหลักโดยใช้คำสั่ง certmgr เพื่อเปิดเครื่องมือ Key Manager จาก บรรทัดรับคำสั่ง

ส่วนนี้อธิบายวิธีใช้ Key Manager เพื่อทำงานต่อไปนี้:

การสร้างฐานข้อมูลคีย์:

ฐานข้อมูลคีย์เปิดให้จุดหมาย VPN เชื่อมต่อโดยใช้ ใบรับรองดิจิตัลที่ถูกต้อง รูปแบบฐานข้อมูลคีย์ (*.kdb) ถูกใช้ กับ IP Security VPNs

ประเภทของใบรับรองดิจิตัล CA ต่อไปนี้ที่มี พร้อม Key Manager:

- RSA Secure Server Certification Authority
- Thawte Personal Premium Certification Authority
- Thawte Personal Freemail Certification Authority
- Thawte Personal Basic Certification Authority
- Thawte Personal Server Certification Authority
- Thawte Server Certification Authority
- Verisign Class 1 Public Primary Certification Authority
- Verisign Class 2 Public Primary Certification Authority
- Verisign Class 3 Public Primary Certification Authority
- Verisign Class 4 Public Primary Certification Authority

ใบรับรองดิจิตัลลายเซ็นเหล่านี้เปิดให้ไคลเอ็นต์ เชื่อมต่อกับเซิร์ฟเวอร์ที่มีใบรับรองดิจิตัลที่ใช้ได้จากผู้ลงนามเหล่านี้ หลังจากคุณสร้างฐานข้อมูลคีย์ คุณสามารถใช้เพื่อเชื่อมต่อ กับเซิร์ฟเวอร์ที่มีใบรับรองดิจิตัลที่ใช้ได้จากหนึ่งในผู้ลงนาม

ในการ ใช้ใบรับรองดิจิตัลลายเซ็นที่ไม่อยู่ในรายการนี้ คุณ ต้องร้องขอจาก CA และเพิ่มลงในฐานข้อมูลคีย์ของคุณ ดูที่ “การเพิ่มใบรับรองดิจิตัล CA root”

ในการ สร้างฐานข้อมูลคีย์โดยใช้คำสั่ง **certmgr** ใช้โพรซีเดอร์ต่อไปนี้:

1. เริ่มทำงานเครื่องมือ Key Manager โดยการพิมพ์:

   ```
   # certmgr
   ```

2. เลือก **New** จากรายการ Key Database File
3. ยอมรับค่าดีฟอลต์ CMS key database file สำหรับฟิลด์ **Key database type**
4. ป้อนชื่อไฟล์ต่อไปนี้ในฟิลด์ **File Name**:

   ```
   ikekey.kdb
   ```

5. ป้อนตำแหน่งของฐานข้อมูลต่อไปนี้ในฟิลด์ **Location**:

   ```
   /etc/security
   ```

   หมายเหตุ: ฐานข้อมูลคีย์ต้อง ชื่อ `ikekey.kbd` และต้องวางอยู่ใน ไดเร็กทอรี `/etc/security` มิฉะนั้น IP Security จะไม่สามารถทำงานได้อย่างถูกต้อง
6. คลิก **OK** หน้าจอ **Password Prompt** แสดง
7. ป้อนรหัสผ่านในฟิลด์ **Password** และป้อนอีกครั้งในฟิลด์ **Confirm Password**
8. ถ้าคุณต้องการเปลี่ยนวันที่รหัสผ่านหมดอายุ ให้ป้อนวันที่ที่ต้องการในฟิลด์ **Set expiration time?** ค่าดีฟอลต์สำหรับฟิลด์นี้คือ 60 วัน ถ้าคุณไม่ต้องการให้รหัสผ่านหมดอายุ ลบค่าในฟิลด์ **Set expiration time?**
9. เพื่อบันทึกรหัสผ่านเข้ารหัสของรหัสผ่านในไฟล์ stash เลือกฟิลด์ **Stash the password to a file?** แล้วเลือก Yes

   หมายเหตุ: คุณต้องเก็บ รหัสผ่านเพื่อทำให้ใช้ใบรับรองดิจิตัลกับ IP Security
10. คลิก **OK** หน้าจอการยืนยัน แสดง เพื่อยืนยันว่าคุณได้สร้างฐานข้อมูลคีย์
11. คลิก **OK** อีกครั้ง พร้อมทั้งคุณกลับไปที่ หน้าจอ IBM Key Management คุณสามารถเลือกดำเนินการงานอื่น หรือออกจากเครื่องมือ

การเพิ่มใบรับรองดิจิตัล CA root:

หลังจากคุณได้ร้องขอและได้รับใบรับรองดิจิตัล root จาก CA แล้วคุณสามารถเพิ่มใบรับรองลงในฐานข้อมูลของคุณ

ใบรับรองดิจิตัล root ส่วนใหญ่อยู่ในรูป *.arm เช่นตัวอย่างต่อไปนี้:

```
cert.arm
```

ในการ เพิ่มใบรับรองดิจิตัล CA root ลงในฐานข้อมูล ใช้โพรซีเดอร์ ต่อไปนี้:

1. ยกเว้นว่าคุณได้กำลังใช้ Key Manager อยู่ ให้เริ่มทำงานเครื่องมือ โดยการพิมพ์:

   ```
   # certmgr
   ```

2. จากหน้าจอหลัก เลือก **Open** จาก รายการ Key Database File
3. ไฮไลต์ไฟล์ฐานข้อมูลคีย์ที่คุณต้องการเพิ่ม ใบรับรองดิจิตัล CA root และคลิก **Open**

4. ป้อนรหัสผ่านและคลิก **OK** เมื่อรหัสผ่านของคุณได้รับการยอมรับ คุณจะกลับไปที่หน้าจอ IBM Key Management ขณะนี้แถบหัวเรื่อง แสดงชื่อของไฟล์ฐานข้อมูลคีย์ที่คุณเลือก บ่งชี้ว่า ขณะนี้ไฟล์ถูกเปิดและพร้อมใช้งาน
5. เลือก **Signer Certificates** จาก รายการ **Personal/Signer Certificates**
6. คลิก **Add**
7. เลือกชนิดข้อมูลจากรายการ **Data type** เช่น:

   ข้อมูล Base64-encoded ASCII
8. ป้อนชื่อไฟล์ใบรับรองและตำแหน่งสำหรับใบรับรองดิจิตัล CA root หรือคลิก **Browse** เพื่อเลือก ชื่อและตำแหน่ง
9. คลิก **OK**
10. ป้อนเลเบลสำหรับใบรับรองดิจิตัล CA root เช่น `Test CA Root Certificate` และคลิก **OK** คุณกลับไปที่หน้าจอ **Key Management** ขณะนี้ฟิลด์ **Signer Certificates** แสดงเลเบลของใบรับรองดิจิตัล CA root ที่คุณเพิ่งเพิ่ม คุณสามารถเลือกดำเนินการงานอื่น หรือออกจากเครื่องมือ

การสร้างการตั้งค่าความไว้วางใจ:

ใบรับรอง CA ที่ติดตั้งถูกตั้งค่าเป็น trusted โดย ดีฟอลต์ คุณสามารถเปลี่ยนการตั้งค่าความไว้วางใจได้ถ้าต้องการ

ในการเปลี่ยนแปลงการตั้งค่าความไว้วางใจ ทำขั้นตอนต่อไปนี้:

1. ยกเว้นว่าคุณได้กำลังใช้ Key Manager อยู่ ให้เริ่มทำงานเครื่องมือ โดยการพิมพ์:

   ```
   # certmgr
   ```
2. จากหน้าจอหลัก เลือก **Open** จาก รายการ **Key Database File**
3. ไฮไลต์ไฟล์ฐานข้อมูลคีย์ที่คุณต้องการเปลี่ยน ใบรับรองดิจิตัลดีฟอลต์และคลิก **Open**
4. ป้อนรหัสผ่านและคลิก **OK** หลังจากรหัสผ่านของคุณได้รับการยอมรับ คุณจะกลับไปที่หน้าจอ **IBM Key Management** แถบหัวเรื่องแสดงชื่อของไฟล์ฐานข้อมูลคีย์ที่คุณเลือก บ่งชี้ว่าขณะนี้ไฟล์ถูกเปิด
5. เลือก **Signer Certificates** จาก รายการ **Personal/Signer Certificates**
6. ไฮไลต์ใบรับรองที่คุณต้องการเปลี่ยนและคลิก **View/Edit** หรือดับเบิลคลิกบนรายการ หน้าจอ **Key Information** แสดงรายการใบรับรอง
7. ในการทำให้ใบรับรองนี้เป็นใบรับรอง root ที่ไว้วางใจ เลือก เช็คบ็อกซ์ติดกับ **Set the certificate as a trusted root** และคลิก **OK** ถ้าใบรับรอง ไม่ได้รับความไว้วางใจ ล้างค่าเช็คบ็อกซ์แทนและคลิก **OK**
8. คลิก **OK** จากหน้าจอ **Signer Certificates** คุณกลับไปที่หน้าจอ **IBM Key Management** คุณสามารถเลือกดำเนินการงานอื่น หรือออกจากเครื่องมือ

การลบใบรับรองดิจิตัล CA root:

ถ้าคุณไม่ต้องการใช้หนึ่งใน CAs ในรายการใบรับรองดิจิตัล ลายเซ็นของคุณอีกต่อไป คุณต้องลบใบรับรองดิจิตัล CA root

หมายเหตุ: ก่อนทำการลบใบรับรองดิจิตัล CA root สร้าง สำเนาสำรองข้อมูลในกรณีที่คุณอาจต้องการสร้าง CA root ใหม่ภายหลัง

ในการ ลบใบรับรองดิจิตัล CA root ออกจากฐานข้อมูล ใช้โพรซีเดอร์ ต่อไปนี้:

1. ยกเว้นว่าคุณได้กำลังใช้ Key Manager อยู่ ให้เริ่มทำงานเครื่องมือ โดยการพิมพ์:

   ```
   # certmgr
   ```

2. จากหน้าจอหลัก เลือก **Open** จาก รายการ **Key Database File**
3. ไฮไลต์ไฟล์ฐานข้อมูลคีย์ที่คุณต้องการ ลบใบรับรองดิจิตัล CA root และคลิก **Open**
4. ป้อนรหัสผ่านและคลิก **OK** หลังจากรหัสผ่านของคุณได้รับการยอมรับ คุณจะกลับไปที่หน้าจอ **Key Management** แถบหัวเรื่อง แสดงชื่อของไฟล์ฐานข้อมูลคีย์ที่คุณเลือก บ่งชี้ว่า ขณะนี้ไฟล์ถูกเปิดและแก้ไข
5. เลือก **Signer Certificates** จาก รายการ **Personal/Signer Certificates**
6. ไฮไลต์ใบรับรองที่คุณต้องการลบและคลิก **Delete** หน้าจอ **Confirm** แสดง
7. คลิก **Yes** คุณกลับไปที่หน้าจอ **IBM Key Management** เลเบลของใบรับรองดิจิตัล CA root ไม่แสดงใน ฟิลด์ **Signer Certificates** อีกต่อไป คุณสามารถเลือกดำเนินการงานอื่น หรือออกจากเครื่องมือ

การร้องขอใบรับรองดิจิตัล:

ในการขอรับใบรับรองดิจิตัล ให้สร้างการร้องขอโดยใช้ Key Manager และส่งการร้องขอไปยัง CA ไฟล์การร้องขอที่คุณสร้างอยู่ในรูปแบบ PKCS#10 จากนั้น CA จะตรวจสอบ identity ของคุณและ ส่งใบรับรองดิจิตัลให้คุณ

ในการร้องขอใบรับรองดิจิตัล ใช้โพรซีเดอร์ต่อไปนี้:

1. ยกเว้นว่าคุณได้กำลังใช้ Key Manager อยู่ ให้เริ่มทำงานเครื่องมือ โดยการพิมพ์:

   ```
   # certmgr
   ```

2. จากหน้าจอหลัก เลือก **Open** จาก รายการ **Key Database File**
3. ไฮไลต์ไฟล์ฐานข้อมูลคีย์ `/etc/security/ikekey.kdb` จากที่คุณต้องการสร้างการร้องขอและคลิก **Open**
4. ป้อนรหัสผ่านและคลิก **OK** หลังจากรหัสผ่านของคุณได้รับการยอมรับ คุณจะกลับไปที่หน้าจอ **IBM Key Management** แถบหัวเรื่องแสดงชื่อของไฟล์ฐานข้อมูลคีย์ที่คุณเลือก บ่งชี้ว่าขณะนี้ไฟล์ถูกเปิดและแก้ไข
5. เลือก **สร้าง** > **สร้างคำร้องขอใบรับรอง**
6. คลิก **New**
7. จากหน้าจอต่อไปนี้ ป้อน Key Label สำหรับ ใบรับรองดิจิตัลที่ลงนามเอง เช่น:

   ```
   keytest
   ```

8. ป้อน common name (ค่าดีฟอลต์คือ ชื่อโฮสต์) และ organization จากนั้นเลือก country สำหรับฟิลด์ที่เหลือ ให้ยอมรับค่าดีฟอลต์ หรือเลือกค่าใหม่
9. กำหนดชื่อ subject alternate ฟิลด์ ทางเลือกที่เชื่อมโยงกับ subject alternate คือ อีเมลแอดเดรส IP address และชื่อ DNS สำหรับประเภท tunnel ของ IP address ให้พิมพ์ IP address เดียวกับที่ตั้งค่าใน IKE tunnel ลงในฟิลด์ IP address สำหรับประเภท tunnel ID ของ *user@FQDN* กรอกฟิลด์อีเมลแอดเดรสให้สมบูรณ์ สำหรับประเภท tunnel ID ของ FQDN พิมพ์ชื่อโดเมนเนมแบบเต็ม (ตัวอย่าง *hostname.companyname*.com) ในฟิลด์ชื่อ DNS
10. ที่ด้านล่างของหน้าจอ ป้อนชื่อสำหรับไฟล์ เช่น:

    ```
    certreq.arm
    ```

11. คลิก **OK** หน้าจอการยืนยัน ถูกแสดง เพื่อตรวจสอบว่าคุณได้สร้างการร้องขอใบรับรอง ดิจิตัลใหม่
12. คลิก **OK** คุณกลับไปที่หน้าจอ **IBM Key Management** ขณะนี้ฟิลด์ **Personal Certificate Requests** แสดงคีย์เลเบลของการร้องขอ ใบรับรองดิจิตัล (PKCS#10) ใหม่ที่สร้าง
13. ส่งไฟล์ไปยัง CA เพื่อร้องขอใบรับรองดิจิตัลใหม่ คุณสามารถเลือกดำเนินการงานอื่น หรือออกจากเครื่องมือ

การเพิ่ม (การรับ) ใบรับรองดิจิตัลใหม่:

หลังจากคุณได้รับใบรับรองดิจิตัลใหม่จาก CA คุณต้องเพิ่มลงในฐานข้อมูลคีย์ที่คุณใช้สร้างการร้องขอ

ในการเพิ่ม (รับ) ใบรับรองดิจิตัลใหม่ใช้โพรซีเดอร์โพรซีเดอร์:

1. ยกเว้นว่าคุณได้กำลังใช้ Key Manager อยู่ให้เริ่มทำงานเครื่องมือ โดยการพิมพ์:

   ```
   # certmgr
   ```

2. จากหน้าจอหลัก เลือก **Open** จาก รายการ **Key Database File**
3. ไฮไลต์ไฟล์ฐานข้อมูลคีย์ที่คุณใช้สร้างการร้องขอ ใบรับรองและคลิก **Open**
4. ป้อนรหัสผ่านและคลิก **OK** หลังจากรหัสผ่านของคุณได้รับการยอมรับ คุณจะกลับไปที่หน้าจอ IBM Key Management แถบหัวเรื่อง แสดงชื่อของไฟล์ฐานข้อมูลคีย์ที่คุณเลือก บ่งชี้ว่า ขณะนี้ไฟล์ถูกเปิดและแก้ไข
5. เลือก **Personal Certificate Requests** จาก รายการ **Personal/Signer Certificates**
6. คลิก **Receive** เพื่อเพิ่มใบรับรองดิจิตัล ที่เพิ่งได้รับใหม่ลงในฐานข้อมูลของคุณ
7. เลือกประเภทข้อมูลของดิจิตัลใบรับรองจาก รายการ **Data type** ดีฟอลต์คือ **ข้อมูล Base64-encoded ASCII**
8. ป้อนชื่อไฟล์ใบรับรองและตำแหน่งสำหรับ ใบรับรองดิจิตัลใหม่ หรือคลิก **Browse** เพื่อเลือก ชื่อและตำแหน่ง
9. คลิก **OK**
10. ป้อนเลเบลอธิบายสำหรับใบรับรองดิจิตัลใหม่ เช่น:

    ```
    VPN Branch Certificate
    ```

11. คลิก **OK** คุณกลับไปที่หน้าจอ **IBM Key Management** ขณะนี้ฟิลด์ **Personal Certificates** แสดงเลเบลของใบรับรองดิจิตัลใหม่ ที่คุณเพิ่งเพิ่ม คุณสามารถเลือกดำเนินการงานอื่น หรือออกจากเครื่องมือ ถ้ามีข้อผิดพลาดในการโหลดใบรับรองเกิดขึ้น ตรวจสอบว่า ไฟล์ใบรับรองขึ้นต้นด้วยข้อความ —-BEGIN CERTIFICATE—- และ สิ้นสุดด้วยข้อความ —-END CERTIFICATE—-

    ตัวอย่าง:

    ```
    -----BEGIN CERTIFICATE-----
    ajdkfjaldfwwwwwwwwwwadafdw
    kajf;kdsajkflasasfkjafdaff
    akdjf;ldasjkf;safdfdasfdas
    kaj;fdljk98dafdas43adfadfa
    -----END CERTIFICATE-----
    ```

    ถ้าข้อความไม่ตรง ให้แก้ไขไฟล์ใบรับรองเพื่อให้เริ่มต้นและสิ้นสุดอย่างเหมาะสม

การลบใบรับรองดิจิตัล:

บางเวลาจำเป็นต้องลบใบรับรองดิจิตัล

หมายเหตุ: ก่อนทำการลบใบรับรองดิจิตัล สร้าง สำเนาสำรองข้อมูลในกรณีที่คุณอาจต้องการสร้างใหม่ภายหลัง

ในการลบ ใบรับรองดิจิตัลออกจากฐานข้อมูลของคุณ ใช้โพรซีเดอร์ต่อไปนี้:

1. ยกเว้นว่าคุณได้กำลังใช้ Key Manager อยู่ให้เริ่มทำงานเครื่องมือ โดยการพิมพ์:

   ```
   # certmgr
   ```

2. จากหน้าจอหลัก เลือก **Open** จาก รายการ **Key Database File**

3. ไฮไลต์ไฟล์ฐานข้อมูลคีย์ที่คุณต้องการ ลบใบรับรองดิจิตัล และคลิก **Open**
4. ป้อนรหัสผ่านและคลิก **OK** หลังจากรหัสผ่านของคุณได้รับการยอมรับ คุณจะกลับไปที่หน้าจอ **IBM Key Management** แถบหัวเรื่องแสดงชื่อของไฟล์ฐานข้อมูลคีย์ที่คุณเลือก บ่งชี้ว่าขณะนี้ไฟล์ถูกเปิดและแก้ไข
5. เลือก **Personal Certificate Requests** จาก รายการ **Personal/Signer Certificates**
6. ไฮไลต์ใบรับรองดิจิตัลที่คุณต้องการลบและ คลิก **Delete** หน้าจอ **Confirm** แสดง
7. คลิก **Yes** คุณกลับไปที่หน้าจอ **IBM Key Management** เลเบล ของใบรับรองดิจิตัลที่คุณเพิ่งลงจะไม่แสดง ในฟิลด์ **Personal Certificates** อีกต่อไป คุณสามารถ เลือกดำเนินการงานอื่น หรือออกจากเครื่องมือ

การเปลี่ยนรหัสผ่านฐานข้อมูล:

บางเวลาจำเป็นต้องเปลี่ยนรหัสผ่านฐานข้อมูล

ในการเปลี่ยนฐานข้อมูลคีย์ ใช้โพรซีเดอร์ต่อไปนี้:

1. ยกเว้นว่าคุณได้กำลังใช้ Key Manager อยู่ ให้เริ่มทำงานเครื่องมือ โดยการพิมพ์:

   ```
   # certmgr
   ```

2. จากหน้าจอหลัก เลือก **Change Password** จาก รายการ **Key Database File**
3. ป้อนรหัสผ่านใหม่ในฟิลด์ **Password** และป้อนอีกครั้งในฟิลด์ **Confirm Password**
4. ถ้าคุณต้องการเปลี่ยนจำนวนวันที่รหัสผ่านหมดอายุ ให้ป้อนจำนวนวันที่ต้องการในฟิลด์ **Set expiration time?** ค่าดีฟอลต์สำหรับฟิลด์นี้คือ 60 วัน ถ้าคุณไม่ต้องการให้รหัสผ่านหมดอายุ ลบค่าในฟิลด์ **Set expiration time?**
5. หากต้องการบันทึกเวอร์ชันเข้ารหัสของรหัสผ่านในไฟล์ stash ให้เลือกฟิลด์ **Stash the password to a file?** และเลือก `Yes`

   หมายเหตุ: คุณต้องเก็บ รหัสผ่านเพื่อทำให้ใช้ใบรับรองดิจิตัลกับ IP Security
6. คลิก **OK** ข้อความในแถบบอกสถานะ บ่งชี้ว่าการร้องขอเสร็จเรียบร้อย
7. คลิก **OK** อีกครั้งและคุณกลับไป ที่หน้าจอ **IBM Key Management** คุณสามารถเลือกดำเนินการงานอื่น หรือออกจากเครื่องมือ

การสร้าง IKE tunnels ที่ใช้ใบรับรองดิจิตัล:

เมื่อต้องการสร้าง IKE tunnels ที่ใช้ใบรับรองดิจิทัล, คุณต้องระบุการลงนาม RSA เป็นโหมดการพิสูจน์ตัวตนในไฟล์นโยบายการแปลงสภาพ IKE tunnel

ตัวอย่างต่อไปนี้แสดงตัวอย่างของไฟล์นโยบาย XML ที่ระบุการลงนาม RSA:

```
<!-- define the policy for IKE tunnel -->
<IKEProtection
 IKE ProtectionName="ike_3des_sha">
 <IKETTransform
  IKE AuthenticationMethod="RSA_signatures"
  IKE Encryption="3DES-CBC"
  IKE Hash="SHA"
  IKE DHGroup="1"/>
</IKEProtection>
```

IP Security สนับสนุนชนิดแบบเป็นเอกลักษณ์ของโฮสต์ IKE tunnel:

- IP address

- Fully Qualified Domain Name (FQDN)
- *user@FQDN*
- X.500 Distinguished Name
- Key identifier

เมื่อ IKE tunnel ใช้โหมดการลงนาม RSA, X.500 Distinguished Names ถูกใช้ในนิยาม IKE tunnel ตัวอย่างเช่น, ถ้าโฮสต์แบบโลคัลและแบบรีโมตของ tunnel ของคุณถูกระบุไว้เป็น **/C=US/O=ABC/OU=SERV/CN=localname.austin.ibm.com** และ **/C=US/O=ABC/OU=SERV/CN=remotename.austin.ibm.com,** นิยาม IKE tunnel ในไฟล์ XML อ่านเนื้อหาตัวอย่างต่อไปนี้:

```
<IKETunnel>
 IKE TunnelName="Key_Tunnel"
 IKE ProtectionRef="ike_3des_sha">
<IKELocalIdentity>
 <ASN1_DN Value="/C=US/O=ABC/OU=SERV/CN=localname.austin.ibm.com">
 </ASN1_DN>
</IKELocalIdentity>
<IKERemoteIdentity>
 <ASN1_DN Value="/C=US/O=ABC/OU=SERV/CN=remotename.austin.ibm.com">
 </ASN1_DN>
</IKERemoteIdentity>
</IKETunnel>
```

เมื่อต้องการขอรับใบรับรองที่ต้องการ จาก certificate authority (CA), ให้ใช้เครื่องมือ Key Manager เพื่อสร้างคำร้องขอใบรับรอง ตัวอย่างเช่น, ถ้าคุณใช้ **/C=US/O=ABC/OU=SERV/CN=name.austin.ibm.com** เป็น Subject Distinguished Name ในใบรับรองของคุณ, คุณต้องป้อนค่า ลงในเครื่องมือ Key Manager เมื่อคุณสร้างคำร้องขอใบรับรอง ดิจิทัล:

ชื่อทั่วไป
name.austin.ibm.com

องค์กร ABC

หน่วยองค์กร
SERV

ประเทศ
US

X.500 Distinguished Name ที่ป้อนไว้คือชื่อที่ตั้งค่าโดยระบบของคุณ หรือผู้ดูแลระบบ LDAP ค่าหน่วยองค์กรคือตัวเลือก

IP Security ยังสนับสนุนการป้อนชนิดแบบเป็นเอกลักษณ์อื่นๆ เป็น Subject Alternate Names ในใบรับรองดิจิทัล ตัวอย่างเช่น, ถ้าคุณใช้ IP address 10.10.10.1 เป็นโฮสต์แบบเป็นเอกลักษณ์สำรอง, ค่าต่อไปนี้ต้องถูกป้อนลงในคำร้องขอใบรับรองดิจิทัล:

ชื่อทั่วไป
name.austin.ibm.com

องค์กร ABC

หน่วยองค์กร

SERV

ประเทศ

US

ฟิลด์ Subject alternate IP address

10.10.10.1

หลังจากคุณสร้างการร้องขอใบรับรองดิจิตัลโดยใช้ ข้อมูลนี้ CA ใช้ข้อมูลนี้เพื่อสร้างใบรับรองดิจิตัล ส่วนบุคคล

เมื่อทำการร้องขอใบรับรองดิจิตัลส่วนบุคคล CA จำเป็นต้องใช้ข้อมูลต่อไปนี้:

- คุณกำลังร้องขอใบรับรอง X.509
- รูปแบบลายเซ็นเป็นการเข้ารหัส MD5 ที่มี RSA
- คุณระบุ Subject Alternate Name หรือไม่ ชนิดของชื่อสำรอง ได้จัดเตรียมไว้ในรายการต่อไป:
  - IP address
  - Fully qualified domain name (FQDN)
  - *user@FQDN*

  ข้อมูล subject alternate-name ต่อไปนี้ถูกรวม ในไฟล์การร้องขอใบรับรอง
- คีย์ที่คุณวางแผนใช้ (ต้องเลือกบิตลายเซ็นดิจิตัล
- ไฟล์การร้องขอใบรับรองดิจิตัล Key Manager (ในรูปแบบ PKCS#10)

สำหรับขั้นตอนที่ระบุไว้ซึ่งอธิบายถึงวิธีการใช้เครื่องมือ Key Manager เพื่อสร้างคำร้องขอใบรับรอง, โปรดดู “การร้องขอใบรับรองดิจิตัล” ในหน้า 265

ก่อนที่คุณจะเรียกใช้ IKE tunnel, คุณต้องเพิ่ม ใบรับรองดิจิทัลส่วนบุคคลที่คุณได้รับจาก CA ไปยังฐานข้อมูล Key Manager, `ikekey.kdb` สำหรับข้อมูลเพิ่มเติม ดูที่ “การเพิ่ม (การรับ) ใบรับรองดิจิตัลใหม่” ในหน้า 266

IP Security สนับสนุนประเภทใบรับรองดิจิตัลส่วนบุคคลต่อไปนี้:

Subject DN

Subject Distinguished Name ต้องอยู่ในรูปแบบหรือลำดับ ต่อไปนี้:

`/C=US/O=ABC/OU=SERV/CN=`*`name`*`.austin.ibm.com`

เครื่องมือ Key Manager อนุญาตให้มีค่า **OU** หนึ่งค่าเท่านั้น

Subject DN และ Subject Alternate Name เป็น IP address

Subject Distinguished Name และ Subject Alternate Name สามารถ ถูกกำหนดเป็น IP address ได้ ดังแสดงต่อไปนี้:

`/C=US/O=ABC/OU=SERV/CN=name.austin.ibm.com` and `10.10.10.1`

Subject DN และ Subject Alternate Name เป็น FQDN

Subject Distinguished Name และ Subject Alternate Name สามารถ ถูกกำหนดเป็นโดเมนเนมแบบเต็ม ดังแสดงต่อไปนี้:

`/C=US/O=ABC/OU=SERV/CN=name.austin.ibm.com` and `bell.austin.ibm.com`

Subject DN และ Subject Alternate Name เป็น *user@FQDN*

Subject Distinguished Name และ Subject Alternate Name สามารถ กำหนดเป็นแอดเดรสผู้ใช้ (*user_ID@fully_qualified_domain_name*) ดังแสดงต่อไปนี้:

`/C=US/O=ABC/OU=SERV/CN=name.austin.ibm.com` and `name@austin.ibm.com`

Subject DN และ Subject Alternate Names หลายชื่อ

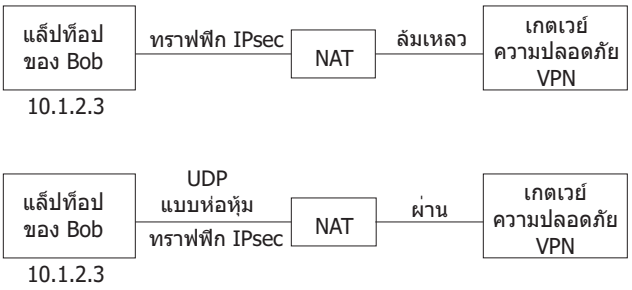
Subject Distinguished Name สามารถเชื่อมโยงกับ Subject Alternate Names หลายชื่อ ดังแสดงต่อไปนี้:

`/C=US/O=ABC/OU=SERV/CN=name.austin.ibm.com` and `bell.austin.ibm.com`, `10.10.10.1`, and `user@name.austin.ibm.com`

Network address translation

IP Security สามารถใช้อุปกรณ์ที่มีแอดเดรสอยู่ภายใต้ network address translation (NAT)

NAT ใช้อย่างกว้างขวางเป็นส่วนหนึ่งของเทคโนโลยีไฟร์วอลล์สำหรับการแบ่งใช้การเชื่อมต่อ อินเตอร์เน็ต และเป็นคุณลักษณะมาตรฐานบนอุปกรณ์เราเตอร์และ edge โปรโตคอล IP Security ขึ้นอยู่กับการระบุจุดหมายรีโมตและ นโยบายของจุดหมายที่ยึดตาม IP แอดเดรสรีโมต เมื่ออุปกรณ์สื่อกลาง เช่นเราเตอร์และไฟร์วอลล์แปลไพรเวตแอดเดรสเป็นพับลิกแอดเดรส กระบวนการพิสูจน์ตัวตนที่จำเป็นใน IP Security อาจ ล้มเหลวเนื่องจากแอดเดรสในแพ็กเก็ต IP ได้ถูกแก้ไขภายหลัง ส่วนย่อยการพิสูจน์ตัวตนถูกคำนวณ ด้วยการสนับสนุน IP Security NAT ใหม่ อุปกรณ์ที่ถูกตั้งค่าภายในโหนดที่ทำหน้าที่ การแปลเน็ตเวิร์กแอดเดรสจะสามารถสร้าง IP Security Tunnel โค้ด IP Security สามารถตรวจหาเมื่อรีโมตแอดเดรสถูก แปล โดยใช้การนำ IP Security ใหม่ไปใช้ด้วยการสนับสนุน NAT จะอนุญาตให้ไคลเอ็นต์ VPN เชื่อมต่อจากบ้านหรือบนถนนทุกที่ไปยัง สำนักงานผ่านการเชื่อมต่ออินเตอร์เน็ตที่มี NAT เปิดใช้งาน



รูปที่ 12. IP Security ที่เปิดใช้งาน NAT

แผนภาพนี้แสดงความแตกต่างระหว่างการนำ IP Security ที่เปิดใช้งาน NAT ไปใช้ ที่มีการรับส่งข้อมูลถูกห่อหุ้มด้วย UDP กับการนำ IP Security ที่ไม่ได้เปิดใช้งาน NAT

การตั้งค่า IP security เพื่อทำงานกับ NAT:

เพื่อใช้ NAT ใน IP Security คุณต้องตั้งค่าตัวแปร *ENABLE_IPSEC_NAT_TRAVERSAL* ในไฟล์ `/etc/isakmpd.conf` เมื่อตัวแปรนี้ ถูกตั้งค่า กฎตัวกรองถูกเพิ่มเพื่อส่งและรับข้อมูลบนพอร์ต 4500

ตัวอย่างต่อไปนี้แสดงกฎกรองเมื่อตัวแปร *ENABLE_IPSEC_NAT_TRAVERSAL* ถูกตั้งค่า

```
Dynamic rule 2:
Rule action         : permit
Source Address      : 0.0.0.0 (any)
Source Mask         : 0.0.0.0 (any)
```

```
Destination Address : 0.0.0.0 (any)
Destination Mask    : 0.0.0.0 (any)
Source Routing      : no
Protocol            : udp
Source Port         : 0 (any)
Destination Port    : 4500
Scope               : local
Direction           : inbound
Fragment control    : all packets
Tunnel ID number    : 0

Dynamic rule 3:
Rule action         : permit
Source Address      : 0.0.0.0 (any)
Source Mask         : 0.0.0.0 (any)
Destination Address : 0.0.0.0 (any)
Destination Mask    : 0.0.0.0 (any)
Source Routing      : no
Protocol            : udp
Source Port         : 4500
Destination Port    : 0 (any)
Scope               : local
Direction           : outbound
Fragment control    : all packets
Tunnel ID number    : 0
```

การตั้งค่าตัวแปร *ENABLE_IPSEC_NAT_TRAVERSAL* ยังเพิ่มกฎตัวกรองเพิ่มบางกฎในตารางตัวกรอง ข้อความ IPSEC NAT พิเศษใช้การห่อหุ้ม UDP และกฎตัวกรองต้อง ถูกเพิ่มเพื่ออนุญาตให้การรับส่งข้อมูลนี้ไหลไปได้ นอกจากนั้น ในเฟส 1 จำเป็น ต้องใช้โหมดลายเซ็น ถ้าใช้ IP Address เป็น identifier ในใบรับรอง ควรเป็น ip address ไพรเวต

IP Security ยัง จำเป็นต้องส่งข้อความ NAT keep alive เพื่อรักษาการแม็พของ IP Address ต้นฉบับและแอดเดรส NAT ช่วงเวลาถูกระบุ โดยตัวแปร *NAT_KEEPALIVE_INTERVAL* ในไฟล์ `/etc/isakmpd.conf` ตัวแปรนี้ระบุ ความถี่แพ็กเก็ต NAT keepalive ที่ถูกส่งเป็นวินาที ถ้าคุณ ไม่ระบุค่าสำหรับ *NAT_KEEPALIVE_INTERVAL* จะใช้ค่าดีฟอลต์เป็น 20 วินาที

ข้อจำกัดเมื่อใช้การแลกเปลี่ยน NAT:

จุดหมายเบื้องหลังอุปกรณ์ NAT ต้องป้องกันการรับส่งข้อมูลตน โดยใช้โปรโตคอล ESP

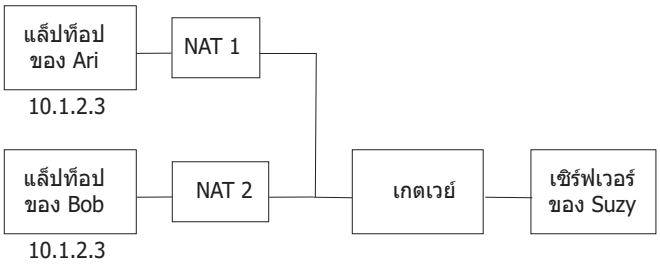
ESP เป็นส่วนหัวที่เหนือกว่าที่ถูกเลือกสำหรับ IP Security และจะ สามารถใช้ได้กับแอ็พพลิเคชันของลูกค้าส่วนใหญ่ ESP รวมการแฮชของ ข้อมูลผู้ใช้ แต่ไม่รวม IP Header การตรวจสอบ integrity ใน ส่วนหัว AH ที่รวมอยู่ใน IP addresses ต้นทางและปลายทาง ในการตรวจสอบ integrity ข้อความที่ถูกใช้คีย์ อุปกรณ์ NAT หรือ NAT ตรงข้ามที่ ทำการเปลี่ยนแปลงในฟิลด์แอดเดรสทำให้การตรวจสอบ integrity ของข้อความไม่ถูกต้อง ดังนั้น ถ้ามีเพียงโปรโตคอล AH เท่านั้นที่ถูกกำหนดในนโยบายเฟส 2 สำหรับ tunnel และ NAT ถูกตรวจพบในเฟสการแลกเปลี่ยนในเฟส 1 Notify Payload ที่แจ้ง `NO_PROPOSAL_CHOSEN` จะถูกส่งไป

นอกจากนั้น การเชื่อมต่อโดยใช้ NAT ต้องเลือกโหมด tunnel เพื่อที่ IP address ต้นฉบับถูกห่อหุ้มอยู่ในแพ็กเก็ต โหม Transport และแอดเดรสที่มี NAT ไม่สามารถทำงานร่วมกันได้ ถ้า NAT ถูกตรวจพบ และมีการนำเสนอเฉพาะโหมด transport ในเฟส 2 ดังนั้น Notify Payload ที่แจ้ง `NO_PROPOSAL_CHOSEN` จะถูกส่งไป

การหลีกเลี่ยงความขัดแย้งโหมด tunnel:

รีโมตเพียร์อาจเจรจารายการที่คาบเกี่ยวกันในเกตเวย์ การคาบเกี่ยวกันนี้เป็นสามารถจากความขัดแย้งของโหมด tunnel

รูปภาพต่อไปนี้แสดงความขัดแย้งของโหมด tunnel



รูปที่ 13. ความขัดแย้งโหมด Tunnel

เกตเวย์มี Security Associations (SAs) เป็นไปได้สอง SA สำหรับ 10.1.2.3 IP address รีโมตแอดเดรสที่ซ้ำกันเหล่านี้ทำให้เกิดความสับสน ว่าควรส่งแพ็กเก็ตที่มาจากเซิร์ฟเวอร์ไปที่ใด เมื่อตั้งค่า tunnel ระหว่างเซิร์ฟเวอร์ของ Suzy และแล็ปท็อปของ Ari ซึ่ง IP address จะถูกใช้และ Suzy ไม่สามารถ ตั้งค่า tunnel กับ Bob ที่มี IP address เดียวกัน เพื่อหลีกเลี่ยงความขัดแย้งในโหมด tunnel คุณไม่ควรกำหนด tunnel ด้วย IP address เดียวกัน เนื่องจาก รีโมตแอดเดรสไม่ได้อยู่ภายใต้การควบคุมของผู้ใช้รีโมต ควรใช้ ID ประเภทอื่น เพื่อระบุรีโมตโฮสต์เช่น โดเมนเนมแบบเต็ม หรือ ผู้ใช้ตามด้วยเครื่องหมาย @ และโดเมนเนมแบบเต็ม

การตั้งค่า manual tunnels

คุณสามารถกำหนดคอนฟิกช่องสัญญาณ IP Security ด้วยตนเอง ถ้าอุปกรณ์ไม่รองกับวิธีการคีย์อัตโนมัติ

ช่องสัญญาณและตัวกรองด้วยตนเอง:

กระบวนการของการตั้งค่า tunnel คือการกำหนด tunnel บน จุดหมายหนึ่ง นำเข้านิยามบนอีกจุดหมายหนึ่ง และเรียกทำงาน tunnel และกฎ ตัวกรองที่จุดหมายทั้งสอง จากนั้น tunnel จะพร้อมใช้งาน

ในการตั้งค่า manual tunnel ไม่จำเป็นต้องตั้งค่ากฎตัวกรอง แยกกัน ตราบใดที่การรับส่งข้อมูลทั้งหมดระหว่างสองโฮสต์ผ่าน tunnel กฎตัวกรองที่จำเป็นจะถูกสร้างโดยอัตโนมัติ

ข้อมูล เกี่ยวกับ tunnel ต้องจัดทำเพื่อให้ตรงกันทั้งสองฝั่งถ้าไม่ได้ระบุไว้โดยชัดแจ้ง ตัวอย่างเช่น อัลกอริทึมการเข้ารหัสและการพิสูจน์ตัวตนที่ระบุ สำหรับต้นทางจะถูกใช้สำหรับปลายทางด้วยถ้าค่าที่ปลายทาง ไม่ได้ระบุไว้

การสร้าง manual tunnel บนโฮสต์แรก:

คุณสามารถกำหนดคอนฟิก tunnel ได้โดยใช้พาธด่วน SMIT`ips4_basic` (สำหรับ IP เวอร์ชัน 4), พาธด่วน SMIT `ips6_basic` (สำหรับ IP เวอร์ชัน 6) หรือคุณสามารถสร้าง tunnel แบบแมนวลได้โดยใช้โพรซีเดอร์ต่อไปนี้

ต่อไปนี้คือตัวอย่างของคำสั่ง **gentun** ที่ใช้สร้าง manual tunnel:

```
gentun -v 4 -t manual -s 5.5.5.19 -d 5.5.5.8 \
   -a HMAC_MD5 -e DES_CBC_8 -N 23567
```

คุณสามารถ ใช้คำสั่ง **lstun -v 4** เพื่อแสดงรายการคุณสมบัติ ของ manual tunnel ที่สร้างโดยตัวอย่างก่อนหน้านี้ เอาต์พุต คล้ายกับตัวอย่างต่อไปนี้:

```
Tunnel ID           : 1
IP Version          : IP Version 4
Source              : 5.5.5.19
Destination         : 5.5.5.8
Policy              : auth/encr
Tunnel Mode         : Tunnel
Send AH Algo        : HMAC_MD5
Send ESP Algo       : DES_CBC_8
Receive AH Algo     : HMAC_MD5
Receive ESP Algo    : DES_CBC_8
Source AH SPI       : 300
Source ESP SPI      : 300
Dest AH SPI         : 23576
Dest ESP SPI        : 23576
Tunnel Life Time    : 480
Status              : Inactive
Target              : -
Target Mask         : -
Replay              : No
New Header          : Yes
Snd ENC-MAC Algo    : -
Rcv ENC-MAC Algo    : -
```

ในการเรียกทำงาน tunnel พิมพ์โค้ดต่อไปนี้:

```
mktun -v 4 -t1
```

กฎ ตัวกรองที่เชื่อมโยงกับ tunnel จะถูกสร้างโดยอัตโนมัติ

ในการ ดูกฎตัวกรอง โดยใช้คำสั่ง **lsfilt -v 4** เอาต์พุตคล้ายกับตัวอย่างต่อไปนี้:

```
Rule 4:
Rule action          : permit
Source Address       : 5.5.5.19
Source Mask          : 255.255.255.255
Destination Address  : 5.5.5.8
Destination Mask     : 255.255.255.255
Source Routing       : yes
Protocol             : all
Source Port          : any 0
Destination Port     : any 0
Scope                : both
Direction            : outbound
Logging control      : no
Fragment control     : all packets
Tunnel ID number     : 1
Interface            : all
Auto-Generated       : yes

Rule 5:
Rule action          : permit
Source Address       : 5.5.5.8
Source Mask          : 255.255.255.255
```

```
Destination Address   : 5.5.5.19
Destination Mask      : 255.255.255.255
Source Routing        : yes
Protocol              : all
Source Port           : any 0
Destination Port      : any 0
Scope                 : both
Direction             : inbound
Logging control       : no
Fragment control      : all packets
Tunnel ID number      : 1
Interface             : all
Auto-Generated        : yes
```

ในการเรียกทำงาน กฎตัวกรอง รวมทั้งกฎตัวกรองดีฟอลต์ ใช้คำสั่ง **mktun -v 4 -t 1**

ในการตั้งค่าอีกฝั่งหนึ่ง (เมื่อ อีกเครื่องหนึ่งใช้ระบบปฏิบัติการนี้) นิยาม tunnel สามารถถูกเอ็กซ์พอร์ตบนโฮสต์ A จากนั้นอิมพอร์ตเข้าโฮสต์ B

คำสั่ง ต่อไปนี้เอ็กซ์พอร์ตนิยาม tunnel ลงในไฟล์ชื่อ **ipsec_tun_manu.exp** และ กฎตัวกรองที่เชื่อมโยงใดๆ ไปยังไฟล์ **ipsec_fltr_rule.exp** ใน ไดเร็กทอรีที่บ่งชี้โดยแฟล็ก **-f**:

```
exptun -v 4 -t 1 -f /tmp
```

การสร้าง manual tunnel บนโฮสต์ที่สอง:

ในการสร้างปลายทาง tunnel ที่ตรงกัน ไฟล์ที่เอ็กซ์พอร์ต จะถูกทำสำเนาและอิมพอร์ตเข้าสู่เครื่องรีโมต

ใช้คำสั่งต่อไปนี้เพื่อสร้างปลายทาง ของ tunnel ที่ตรงกัน:

```
imptun -v 4 -t 1 -f /tmp
```

โดยที่

1 คือ tunnel ที่จะถูกอิมพอร์ต

/tmp คือไดเร็กทอรีที่มีไฟล์ที่อิมพอร์ตอยู่

หมายเลข tunnel ถูกสร้างโดยระบบ คุณ สามารถหาได้จากเอาต์พุตของคำสั่ง **gentun** หรือโดยการใช้คำสั่ง **lstun** เพื่อแสดงรายการ tunnels และพิจารณาหมายเลข tunnel ที่ถูกต้องเพื่ออิมพอร์ต ถ้ามีหนึ่ง tunnel เท่านั้นในไฟล์ที่อิมพอร์ต หรือถ้าจะอิมพอร์ต tunnels ทั้งหมด ไม่ต้องใช้อ็อพชัน **-t**

ถ้าเครื่องรีโมต ไม่ได้ทำงานระบบปฏิบัติการนี้ ไฟล์ที่เอ็กซ์พอร์ตสามารถใช้เป็น การอ้างอิงสำหรับการตั้งค่าอัลกอริทึม คีย์ และค่า security parameters index (SPI) สำหรับปลายทางของอีก tunnel หนึ่ง

ไฟล์ที่เอ็กซ์พอร์ต จากผลิตภัณฑ์ไฟร์วอลล์สามารถอิมพอร์ตเพื่อสร้าง tunnels ใน การทำนี้ ใช้อ็อพชัน **-n** เมื่ออิมพอร์ตไฟล์ดังนี้:

```
imptun -v 4 -f /tmp -n
```

การลบตัวกรองออก:

ในการลบตัวกรองและหยุดทำงานการรักษาความปลอดภัย IP โดยสมบูรณ์ใช้คำสั่ง **rmdev**

กฎตัวกรองดีฟอลต์ยังคงแอ็คทีฟแม้การกรองจะถูกปิดทำงาน ด้วยคำสั่ง **mkfilt -d** คำสั่งนี้อนุญาตให้คุณ หยุดทำงานชั่วคราว หรือลบกฎตัวกรองทั้งหมดและโหลดกฎใหม่ขณะที่ การป้องกันของกฎดีฟอลต์ยังคงทำงานอยู่ กฎตัวกรองดีฟอลต์คือ *DENY* ถ้าคุณปิดทำงานการกรองด้วยคำสั่ง **mkfilt -d** รายงานจากคำสั่ง **lsfilt** จะแสดงว่าการกรอง ถูกปิดทำงาน แต่ไม่มีแพ็กเก็ตใดได้รับอนุญาตให้เข้าหรือออก ถ้าคุณต้องการหยุดทำงาน การรักษาความปลอดภัย IP ทั้งหมด ใช้คำสั่ง **rmdev**

การตั้งค่าตัวกรองการรักษาความปลอดภัย IP

การกรองสามารถตั้งค่าเป็นตัวกรองอย่างง่าย โดยใช้กฎตัวกรอง ที่สร้างอัตโนมัติส่วนใหญ่ หรือสามารถกำหนดเองโดยการกำหนดฟังก์ชันตัวกรอง ที่เฉพาะเจาะจงอย่างมากโดยยึดตามคุณสมบัติของแพ็กเก็ต IP

แต่ละบรรทัดในตารางตัวกรองคือ *กฎ* ชุดรวม กฎจะพิจารณาว่าแพ็กเก็ตใดได้รับการยอมรับในและนอกเครือง และทิศทางใด การจับคู่กฎตัวกรองบนแพ็กเก็ตขาเข้า ทำได้โดยการเปรียบเทียบแอดเดรสต้นทางกับค่า SPI ของรายการที่แสดง ในตารางตัวกรอง ดังนั้น คู่นี้ต้องเป็นค่าเฉพาะ กฎตัวกรอง สามารถควบคุมการสื่อสารได้หลายรูปแบบ ประกอบด้วยแอดเดรสต้นทางและปลายทาง และ masks โปรโตคอล หมายเลขพอร์ต ทิศทาง การควบคุมแฟรกเมนต์ การจัดเส้นทางต้นทาง tunnel และประเภทอินเตอร์เฟส

ประเภทของกฎตัวกรองมีดังนี้:

- กฎตัวกรองแบบสแตติก ถูกสร้างในตารางตัวกรอง ที่จะใช้สำหรับการกรองทั่วไปของการรับส่งข้อมูล หรือสำหรับการเชื่อมโยง กับ manual tunnels โดยสามารถเพิ่ม ลบ แก้ไข และย้ายได้ ฟิลด์ข้อความอธิบายที่เป็นทางเลือกสามารถเพิ่มเพื่อระบุกฎที่เจาะจง
- กฎตัวกรองที่สร้างอัตโนมัติ และกฎตัวกรอง ของผู้ใช้ที่เจาะจง (หรือเรียกกฎตัวกรอง *สร้างอัตโนมัติ*) เป็น ชุดของกฎที่เจาะจงที่สร้างขึ้นสำหรับการใช้ช่องสัญญาณ IKE กฎตัวกรอง ทั้งแบบสแตติกและไดนามิกถูกสร้างขึ้นจากข้อมูล tunnel การจัดการข้อมูล และจากการเจรจา tunnel การจัดการข้อมูล
- กฎตัวกรองที่กำหนดไว้แล้ว คือกฎตัวกรองทั่วไป ที่ไม่สามารถแก้ไข ย้าย หรือลบได้ เช่นกฎ `all traffic` กฎ `ah` และกฎ `esp` ซึ่งเกี่ยวข้องกับการรับส่งข้อมูลทั้งหมด

แฟล็กทิศทาง (**-w**) ของคำสั่ง **genfilt** ถูกใช้ระบุว่าเมื่อใดกฎที่ระบุควรใช้ระหว่าง การประมวลผลแพ็กเก็ตอินพุต หรือระหว่างการประมวลผลแพ็กเก็ตเอาต์พุต เมื่อค่า **ทั้งสอง** สำหรับแฟล็กถูกใช้ จะระบุว่ากฎนี้ถูกใช้ระหว่าง การประมวลผลทั้งอินพุตและเอาต์พุต ใน AIX IPsec, เมื่อเปิดใช้การกรอง, อย่างน้อยหนึ่งกฎต้องกำหนดผลลัพธ์ของแพ็กเก็ตเครือข่ายใดๆ (เป็นแบบขาเข้าหรือขาออก) ถ้าคุณต้องการ ให้ใช้กฎระหว่างการประมวลผลของแพ็กเก็ตขาเข้าอย่างเดียวเท่านั้น (หรือแพ็กเก็ต ขาออก) คุณสามารถเลือกได้โดยใช้ตัวเลือก **-w** ของ คำสั่ง **genfilt** ตัวอย่างเช่น เมื่อแพ็กเก็ตถูกส่งออกจากโฮสต์ A ไปยังโฮสต์ B แพ็กเก็ต IP ขาออกมีแอดเดรสต้นทางของ *A* และแอดเดรสปลายทางของ *B* บนโฮสต์ A แพ็กเก็ตนี้ถูกประมวลผลโดยตัวกรอง IPsec ระหว่างการประมวลผลขาออกระหว่างการประมวลผลขาเข้าบนโฮสต์ B สมมติว่ามีเกตเวย์ G ระหว่างโฮศต์ A และโฮสต์ B บนเกตเวย์ G แพ็กเก็ตเดียวกันนี้ แพ็กเก็ตด้วยกันนี้ (ฟิลด์ที่เปลี่ยนแปลงได้มีค่าเดียวกัน) ถูกประมวลผลสองครั้ง: ครั้งหนึ่งสำหรับการประมวลผลขาเข้า และอีกครั้งสำหรับการประมวลผลขาออก (ถ้าอ็อพชัน **ipforwarding** ถูกเลือก) สำหรับแพ็กเก็ตที่จะเดินทาง จากโฮสต์ A ไปโฮสต์ B ผ่านเกตเวย์ G คุณต้องใช้กฎการอนุญาตที่มี:

- บนโฮสต์ A – **src addr** ตั้งค่าเป็น `A` **dest addr** เป็น `B` ทิศทางเป็นขาออก
- บนโฮสต์ B – **src addr** ตั้งค่าเป็น `A` **dest addr** เป็น `B` ทิศทางเป็นขาเข้า

แต่บนเกตเวย์ G คุณจะต้อใช้กฎสองข้อ:

1. **src addr** ตั้งค่าเป็น A **dest addr** เป็น B ทิศทางเป็นขาออก
2. **src addr** ตั้งค่าเป็น A **dest addr** เป็น B ทิศทางเป็นขาเข้า

กฎด้านบนสามารถแทนที่โดย: **src addr** ตั้งค่าเป็น A **dest addr** เป็น B และทิศทางเป็นทั้งสอง ดังนั้น ค่า ของ **ทั้งสอง** สำหรับทิศทางถูกใช้โดยทั่วไปในเกตเวย์ที่ มีการตั้งค่าอ็อพชัน **ipforwarding** เป็น no การตั้งค่าด้านบนใช้สำหรับการเดินทางของแพ็กเก็ตจากโฮสต์ A ไปโฮสต์ B ผ่านเกตเวย์ G เท่านั้น ถ้าคุณต้องการให้แพ็กเก็ตเดินทาง ในทิศทางตรงกันข้าม (จากโฮสต์ B ไปโฮสต์ A ผ่านเกตเวย์ G) คุณต้องใช้อีกกฎหนึ่ง

หมายเหตุ: ทิศทาง **ทั้งสอง** หมายความว่ากฎที่เชื่อมโยงถูกใช้ สำหรับทั้งแพ็กเก็ตขาเข้าและขาออก อย่างไรก็ตาม ไม่ได้หมายความว่า กฎถูกนำใช้เมื่อแอดเดรสต้นทางและปลายทาง เป็นตรงกันข้าม ตัวอย่างเช่น ถ้าเซิร์ฟเวอร์ A มีกฎที่มี *A* เป็นแอดเดรสต้นทาง และ *B* เป็นแอดเดรสปลายทาง และทิศทางถูกตั้งค่าเป็น **ทั้งสอง** ดังนั้น *A* จะเป็นแพ็กเก็ตขาเข้าที่มี *B* เป็น แอดเดรสต้นทางและ *A* เป็นปลายทางไม่ตรงกับกกฎนี้โดยทั่วไป อ็อพชัน **ทั้งสอง** ถูกใช้ในแพ็กเก็ตที่ส่งต่อ แพ็กเก็ต

ที่เชื่อมโยงกับกฎตัวกรองเหล่านี้คือ Subnet masks ซึ่ง ID กลุ่มที่ถูกเชื่อมโยงกับ กฎตัวกรอง และอ็อพชันการตั้งค่า host-firewall-host ส่วน ต่อไปนี้อธิบายประเภทต่างๆ ของกฎตัวกรองและ คุณลักษณะที่เชื่อมโยงกับกฎ

ตัวกรอง IP สำหรับ AIX:

IPFilter คือซอฟต์แวร์แพ็กเกจที่สามารถใช้เพื่อจัดให้มี เซอร์วิส network address translation (NAT) หรือไฟร์วอลล์

ซอฟต์แวร์ที่เปิดเผยซอร์ส IPFilter เวอร์ชัน 4.1.13 ถูกพอร์ตไปยัง AIX สอดคล้องกับ ไลเซนส์ที่แสดงบนเว็บไซต์ IP Filter (http://coombs.anu.edu.au/~avalon/) ซอฟต์แวร์ IPFilter ถูกจัดส่งมาบน แพ็กเสริม AIX แพ็กเกจ installp package, ipfl ประกอบด้วยหน้า man และไลเซนส์

บนระบบปฏิบัติการ AIX, ผลิตภัณฑ์ IPFilter โหลดเป็นส่วนขยายเคอร์เนล, /usr/lib/drivers/ipf ไบนารี **ipf**, **ipfs**, **ipfstat**, **ipmon** และ **ipnat** ยังมาพร้อมกับแพ็กเกจนี้ด้วย

หลังการติดตั้งแพ็กเกจ รันคำสั่งต่อไปนี้เพื่อโหลด ส่วนขยายเคอร์เนล:

```
/usr/lib/methods/cfg_ipf -l
```

รันคำสั่งต่อไปนี้เพื่อยกเลิกการโหลดส่วนขยายเคอร์เนล:

```
/usr/lib/methods/cfg_ipf -u
```

อย่าลืมเปิดใช้ ipforwarding (อ็อพชัน network) ถ้าจำเป็นต้องใช้การส่งต่อ แพ็กเก็ต สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ IPFilter รวมถึงหน้า man และ FAQ ให้ตรวจสอบเว็บไซต์ IPFilter (http://coombs.anu.edu.au/~avalon/)

กฎตัวกรองแบบสแตติก:

แต่ละกฎตัวกรองแบบสแตติกจะมีฟิลด์ที่คั่นด้วยช่องว่าง

รายการต่อไปนี้มีชื่อของแต่ละฟิลด์ใน กฎตัวกรองแบบสแตติกตามด้วยตัวอย่างจากกฎ 1 ในวงเล็บ:

- Rule_number (1)
- Action (permit)
- Source_addr (0.0.0.0)
- Source_mask (0.0.0.0)

- Dest_addr (0.0.0.0)
- Dest_mask (0.0.0.0)
- Source_routing (no)
- Protocol (udp)
- Src_prt_operator (eq)
- Src_prt_value (4001)
- Dst_prt_operator (eq)
- Dst_prt_value (4001)
- Scope (both)
- Direction (both)
- Logging (no)
- Fragment (all packets)
- Tunnel (0)
- Interface (all)

ตัวอย่างของกฎตัวกรองแบบสแตติก

```
1 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 no udp eq 4001 eq 4001 both both no all
   packets 0 all

2 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 no ah any 0 any 0 both both no all packets
   0 all

3 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 no esp any 0 any 0 both both no all packets
   0 all

4 permit 10.0.0.1 255.255.255.255 10.0.0.2 255.255.255.255 no all any 0 any 0 both
   outbound no all packets 1 all outbound traffic

5 permit 10.0.0.2 255.255.255.255 10.0.0.1 255.255.255.255 no all any 0 any 0 both
   inbound no all packets 1 all


6 permit 10.0.0.1 255.255.255.255 10.0.0.3 255.255.255.255 no tcp lt 1024 eq 514 local
   outbound yes all packets 2 all


7 permit 10.0.0.3 255.255.255.255 10.0.0.1 255.255.255.255 no tcp/ack eq 514 lt 1024
   local inbound yes all packets 2 all


8 permit 10.0.0.1 255.255.255.255 10.0.0.3 255.255.255.255 no tcp/ack lt 1024 lt 1024
   local outbound yes all packets 2 all


9 permit 10.0.0.3 255.255.255.255 10.0.0.1 255.255.255.255 no tcp lt 1024 lt 1024 local
   inbound yes all packets 2 all
```

```
10 permit 10.0.0.1 255.255.255.255 10.0.0.4 255.255.255.255 no icmp any 0 any 0 local
   outbound yes all packets 3 all


11 permit 10.0.0.4 255.255.255.255 10.0.0.1 255.255.255.255 no icmp any 0 any 0 local
   inbound yes all packets 3 all


12 permit 10.0.0.1 255.255.255.255 10.0.0.5 255.255.255.255 no tcp gt 1023 eq 21 local
   outbound yes all packets 4 all


13 permit 10.0.0.5 255.255.255.255 10.0.0.1 255.255.255.255 no tcp/ack eq 21 gt 1023 local
   inbound yes all packets 4 all


14 permit 10.0.0.5 255.255.255.255 10.0.0.1 255.255.255.255 no tcp eq 20 gt 1023 local
   inbound yes all packets 4 all


15 permit 10.0.0.1 255.255.255.255 10.0.0.5 255.255.255.255 no tcp/ack gt 1023 eq 20 local
   outbound yes all packets 4 all


16 permit 10.0.0.1 255.255.255.255 10.0.0.5 255.255.255.255 no tcp gt 1023 gt 1023 local
   outbound yes all packets 4 all


17 permit 10.0.0.5 255.255.255.255 10.0.0.1 255.255.255.255 no tcp/ack gt 1023 gt 1023 local
   inbound yes all packets 4 all


18 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 no all any 0 any 0 both both yes all
   packets
```

แต่ละกฎในตัวอย่างก่อนหน้านี้ได้รับการอธิบายดังนี้:

กฎ 1 สำหรับ **Session Key** daemon กฎนี้แสดงในตารางตัวกรอง IP Version 4 เท่านั้น โดยใช้หมายเลขพอร์ต 4001 เพื่อควบคุมแพ็กเก็ต ในการรีเฟรชเซสชันคีย์ กฎ 1 ตัวอย่างวิธีที่หมายเลขพอร์ต สามารถถูกใช้เพื่อวัตถุประสงค์ที่เจาะจง

หมายเหตุ: อย่าแก้ไข กฎตัวกรองนี้ ยกเว้นเพื่อวัตถุประสงค์ในการบันทึกการทำงาน

กฎ 2 และ 3

อนุญาตให้มีการประมวลผลส่วนหัวการพิสูจน์ตัวตน (AH) และส่วนหัว encapsulating security payload (ESP)

หมายเหตุ: อย่าแก้ไขกฎ 2 และ 3 ยกเว้นเพื่อวัตถุประสงค์ในการบันทึกการทำงาน

กฎ 4 และ 5

ชุดของกฎที่สร้างอัตโนมัติที่กรองการรับส่งข้อมูลระหว่างการกรอง 10.0.0.1 และ 10.0.0.2 ผ่าน tunnel 1 กฎ 4 สำหรับการรับส่งข้อมูลขาออก และกฎ 5 สำหรับการับส่งข้อมูลขาเข้า

หมายเหตุ: กฎ 4 มีรายละเอียดที่ ผู้ใช้กำหนดเองของ *outbound traffic*

กฎ 6 ถึง 9

ชุดของกฎที่ขาออกผู้ใช้กำหนดเองที่กรองเซอร์วิส rsh, rcp, rdump, rrestore และ rdist ขาออกระหว่างแอดเดรส 10.0.0.1 และ 10.0.0.3 ผ่าน tunnel 2 ในตัวอย่างนี้ การบันทึกการทำงานถูกตั้งค่าเป็น Yes ดังนั้น ผู้ดูแลระบบสามารถมอนิเตอร์การรับส่งข้อมูลประเภทนี้

กฎ 10 และ 11

ชุดของกฎที่ผู้ใช้กำหนดเองที่กรองเซอร์วิส icmp ทั้ง ขาเข้าและขาออกของประเภทใดๆ ระหว่าง 10.0.0.1 และ 10.0.0.4 ผ่าน tunnel 3

กฎ 12 ถึง 17

กฎที่ผู้ใช้กำหนดเองที่กรองเซอร์วิส file transfer protocol (FTP) ขาออกจาก 10.0.0.1 และ 10.0.0.5 ผ่าน tunnel 4

กฎ 18 กฎที่สร้างอัตโนมัติจะอยู่ที่ท้ายตารางเสมอ ในตัวอย่างนี้ อนุญาตแพ็กเก็ตทั้งหมดที่ไม่ตรงกับกฎตัวกรอง ข้ออื่นๆ โดยสามารถถูกตั้งค่าให้ปฏิเสธการรับส่งข้อมูลทั้งหมดที่ไม่ตรงกับกฎตัวกรอง อื่นๆ

แต่ละกฎสามารถดูแยกกันได้ (โดยใช้ **lsfilt**) เพื่อแสดงรายการแต่ละฟิลด์พร้อมค่าของฟิลด์ ตัวอย่าง:

```
Rule 1:
Rule action          : permit
Source Address       : 0.0.0.0
Source Mask          : 0.0.0.0
Destination Address  : 0.0.0.0
Destination Mask     : 0.0.0.0
Source Routing       : yes
Protocol             : udp
Source Port          : eq  4001
Destination Port     : eq  4001
Scope                : both
Direction            : both
Logging control      : no
Fragment control     : all packets
Tunnel ID number     : 0
Interface            : all
Auto-Generated       : yes
```

รายการต่อไปนี้มีพารามิเตอร์ทั้งหมดที่สามารถ ระบุในกฎตัวกรอง:

`-v` IP Version: 4 หรือ 6

`-a` การดำเนินการ:

- **d** ปฏิเสธ
- **p** อนุญาต

`-s` แอดเดรสต้นทาง สามารถเป็น IP address หรือชื่อโฮสต์

`-m` subnet mask ต้นทาง

`-d` แอดเดรสปลายทาง สามารถเป็น IP address หรือชื่อโฮสต์

`-M` subnet mask ปลายทาง

`-g` การควบคุมการจัดเส้นทางต้นทาง: `y` หรือ `n`

-c โปรโตคอล ค่าสามารถเป็น udp, icmp, tcp, tcp/ack, ospf, pip, esp, ah และ all

-o พอร์ตต้นทาง หรือการดำเนินการประเภท ICMP

-p พอร์ตต้นทาง หรือค่าประเภท ICMP

-O พอร์ตปลายทาง หรือการดำเนินการโค้ด ICMP

-P พอร์ตปลายทาง หรือค่าโค้ด ICMP

-r การจัดเส้นทาง:

- **r** แพ็กเก็ตที่ส่งต่อ
- **l** แพ็กเก็ตปลายทาง/ต้นทางโลคัล
- **b** ทั้งสอง

-l การควบคุมบันทึกการทำงาน

- **y** รวมในบันทึกการทำงาน
- **n** ไม่รวมในบันทึกการทำงาน

-f การแตกแฟรกเมนต์

- **y** ใช้กับส่วนหัวแฟรกเมนต์ แฟรกเมนต์ และที่ไม่ใช่แฟรกเมนต์
- **o** ใช้กับแฟรกเมนต์และส่วนหัวแฟรกเมนต์เท่านั้น
- **n** ใช้กับที่ไม่ใช่แฟรกเมนต์เท่านั้น
- **h** ใช้กับที่ไม่ใช่แฟรกเมนต์และส่วนหัวแฟรกเมนต์เท่านั้น

-t Tunnel ID

-i อินเตอร์เฟส เช่น tr0 หรือ en0

สำหรับข้อมูลเพิ่มเติม ดูที่คำอธิบายคำสั่ง **genfilt** และ **chfilt**

กฎตัวกรองที่สร้างอัตโนมัติ และกฎตัวกรองที่ผู้ใช้ระบุ:

กฎบางกฎถูกสร้างอัตโนมัติสำหรับการใช้งานตัวกรอง IP Security และโค้ด tunnel

กฎที่สร้างโดยอัตโนมัติประกอบด้วยชุดของกฎต่อไปนี้:

- กฎสำหรับ daemon คีย์เซสชันที่รีเฟรชคีย์ IP เวอร์ชัน 4 ใน IKE
- กฎสำหรับการประมวลผลของแพ็กเก็ต AH และ ESP

กฎตัวกรองยังถูกสร้างโดยอัตโนมัติเมื่อคุณนิยาม tunnels สำหรับ tunnels ด้วยตนเอง กฎทีซอร์สสร้างอัตโนมัติระบุแอดเดรสต้นทาง และปลายทางที่ค่า mask รวมถึง tunnel ID ทราฟิกทั้งหมด ระหว่างแอดเดรสเหล่านั้นจะโฟลว์ผ่าน tunnel

สำหรับ IKE tunnels กฎตัวกรองที่สร้างอัตโนมัติจะพิจารณา โปรโตคอลและหมายเลขพอร์ตระหว่างการเจรจา IKE กฎตัวกรอง IKE ถูกเก็บในตารางแยกต่างหาก ซึ่งถูกค้นหาหลังกฎตัวกรองสแตติก และก่อนกฎที่สร้างอัตโนมัติ กฎตัวกรอง IKE ถูกแทรกใน ตำแหน่งดีฟอลต์ภายในตารางตัวกรองสแตติก แต่สามารถย้ายตำแหน่ง ได้โดยผู้ใช้

กฎที่สร้างอัตโนมัติต้องทำการรับส่งทั้งหมดบน tunnel กฎที่ผู้ใช้กำหนดเองสามารถจัดวางข้อจำกัดบนการรับส่งบางประเภท จัดวางกฎที่ใช้ผู้ใช้กำหนดเองเหล่านี้ก่อนกฎที่สร้างอัตโนมัติ เนื่องจาก IP Security ใช้กฎแรกที่พบที่มีผลใช้กับแพ็กเก็ต ต่อไปนี้คือตัวอย่างของกฎตัวกรองที่ผู้ใช้กำหนดเองที่กรอง การรับส่งข้อมูลตามการดำเนินการ ICMP

```
1 permit 10.0.0.1 255.255.255.255 10.0.0.4 255.255.255.255 no icmp any 8 any 0
   local outbound no all packets 3 all
2 permit 10.0.0.4 255.255.255.255 10.0.0.1 255.255.255.255 no icmp any 0 any 0 local
   inbound no all packets 3 all
3 permit 10.0.0.4 255.255.255.255 10.0.0.1 255.255.255.255 no icmp any 8 any 0 local
   inbound no all packets 3 all
4 permit 10.0.0.1 255.255.255.255 10.0.0.4 255.255.255.255 no icmp any 0 any 0 local
   outbound no all packets 3 all
```

เพื่อให้ทำการตั้งค่า tunnel เดียวได้ง่าย กฎตัวกรอง ถูกสร้างอัตโนมัติเมื่อ tunnels ถูกกำหนด ฟังก์ชันนี้สามารถ ระงับได้โดยการระบุแฟล็ก **–g** ใน **gentun** คุณสามารถพบไฟล์ตัวกรองตัวอย่างที่มี **genfilt** เพื่อสร้างกฎตัวกรองสำหรับเซอร์วิส TCP/IP ที่แตกต่างได้ใน `/usr/samples/ipsec/filter.sample`

กฎตัวกรองที่กำหนดไว้แล้ว:

กฎตัวกรองที่กำหนดไว้แล้วหลายๆ กฎถูกสร้างอัตโนมัติด้วย เหตุการณ์ที่เจาะจง

เมื่อโหลดอุปกรณ์ `ipsec_v4` หรือ `ipsec_v6` กฎที่กำหนดไว้แล้วจะถูกแทรกในตารางตัวกรองและ จากนั้นถูกเรียกทำงานโดยค่าดีฟอลต์ กฎที่กำหนดไว้แล้วนี้จะอนุญาต แพ็กเก็ตทั้งหมด แต่ผู้ใช้สามารถตั้งค่าได้ และคุณสามารถตั้งค่าให้ปฏิเสธแพ็กเก็ตทั้งหมดได้

หมายเหตุ: เมื่อทำการตั้งค่ารีโมต ทำให้แน่ใจว่ากฎการปฏิเสธ ไม่ถูกเปิดใช้งานก่อนการตั้งค่าจะเสร็จสมบูรณ์ เพื่อป้องกันมิให้เซสชันของคุณถูกล็อกออกจากระบบของเครื่อง สถานการณ์ นี้สามารถเลี่ยงได้โดยการตั้งค่าการดำเนินการดีฟอลต์เพื่ออนุญาตหรือ โดยการตั้งค่า tunnel ไปยังเครื่องรีโมตก่อนการเรียกทำงาน IP Security

ตารางตัวกรอง IP Version 4 และ IP Version 6 ทั้งสอง มีกฎที่กำหนดไว้แล้ว อันใดอันหนึ่งอาจถูกเปลี่ยนแปลงโดยอิสระเพื่อให้ปฏิเสธทั้งหมด นี้จะป้องกันมิให้มีการรับส่งข้อมูล ยกเว้นการรับส่งอข้อมูลนั้นถูกกำหนด ไว้เป็นพิเศษโดยกฎตัวกรองเพิ่มอ็อพชันเดียวเท่านั้นที่จะเปลี่ยนแปลง กฎที่กำหนดไว้แล้วคือ **chfilt** ที่มีอ็อพชัน **–l** ซึ่งอนุญาตให้แพ็กเก็ตที่ตรงกับกฎนั้นถูกบันทึกการทำงาน

ในการสนับสนุน IKE tunnels กฎตัวกรองแบบไดนามิกถูกใส่ ในตารางตัวกรอง IP Version 4 นี้คือตำแหน่งที่กฎตัวกรอง แบบไดนามิกถูกแทรกลงในตารางตัวกรอง ตำแหน่งนี้สามารถ ควบคุมโดยผู้ใช้โดยการย้ายตำแหน่งขึ้นหรือลงในตาราง ตัวกรอง หลังจาก tunnel manager daemon และ **isakmpd** daemon ได้รับการเตรียมข้อมูลเบื้องต้นเพื่ออนุญาตให้ IKE tunnels เจรจา กฎจะถูกสร้าง โดยอัตโนมัติในตารางตัวกรองแบบไดนามิกเพื่อจัดการข้อความ IKE รวมถึง แพ็กเก็ต AH และ ESP

Subnet masks:

Subnet masks ถูกใช้เพื่อจัดกลุ่มชุดของ IDs ที่ถูกเชื่อมโยง ด้วยกฎตัวกรอง ค่า mask ถูก AND กับ ID ในกฎตัวกรอง และเปรียบเทียบกับ ID ทีร่ะบุในแพ็กเก็ต

ตัวอย่าง กฎตัวกรองที่มี IP address ต้นทาง `10.10.10.4` และ subnet mask `255.255.255.255` ที่ระบุว่าต้องมีการตรงกัน แน่นอนของ IP address ฐานสิบ ดังแสดงต่อไปนี้:

	ไบนารี	ฐานสิบ
IP address ต้นทาง	1010.1010.1010.0100	10.10.10.4
Subnet mask	11111111.11111111.11111111.11111111	255.255.255.255

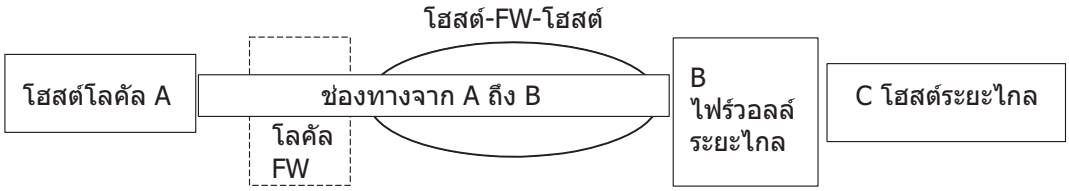
10.10.10.x subnet ถูกระบุเป็น 11111111.11111111.11111111.0 หรือ 255.255.255.0 แอดเดรสขาเข้าจะมี subnet mask ถูกใช้กับแอดเดรส จากนั้นการรวมกัน จะถูกเปรียบเทียบกับ ID ในกฎตัวกรอง ตัวอย่าง แอดเดรส 10.10.10.100 กลายเป็น 10.10.10.0 หลัง subnet mask ถูกนำใช้ ซึ่งตรงกับกฎตัวกรอง

subnet mask 255.255.255.240 อนุญาตให้เป็นค่าใดๆ สำหรับ สี่บิตสุดท้ายในแอดเดรส

การตั้งค่าโฮสต์-ไฟร์วอลล์-โฮสต์:

อ็อพชันการตั้งค่าโฮสต์-ไฟร์วอลล์-โฮสต์สำหรับ tunnels อนุญาตให้คุณสร้าง tunnel ระหว่างโฮสต์ของคุณกับไฟร์วอลล์ จากนั้น สร้างกฎตัวกรองที่จำเป็นโดยอัตโนมัติสำหรับการสื่อสารที่ถูกต้อง ระหว่างโฮสต์ของคุณกับโฮสต์ที่อยู่หลังไฟร์วอลล์

กฎตัวกรองที่สร้างอัตโนมัติอนุญาตให้กฎทั้งหมดระหว่าง โฮสต์ที่ไม่มีไฟร์วอลล์สองโฮสต์บน tunnel ถูกระบุ กฎดีฟอลต์สำหรับส่วนหัว user datagram protocol (UDP), Authentication Headers (AH) และ Encapsulating Security Payload (ESP) headers ควรมีการจัดการการสื่อสารจากโฮสต์ ไปยังไฟร์วอลล์ ไฟร์วอลล์จะต้องถูกตั้งค่าอย่างเหมาะสม เพื่อใช้ดำเนินการตั้งค่าให้เสร็จสมบูรณ์ คุณควรใช้ไฟล์ที่เอ็กซ์พอร์ตจาก tunnel ที่คุณสร้างเพื่อป้อนค่า SPI และคีย์ที่ไฟร์วอลล์ต้องใช้



รูปที่ 14. โฮสต์-ไฟร์วอลล์-โฮสต์

ภาพประกอบนี้แสดงการตั้งค่าโฮสต์-ไฟร์วอลล์-โฮสต์ โฮสต์ A มี tunnel กำลังทำงานผ่านโลคัลไฟร์วอลล์และออกไปยัง อินเตอร์เน็ต จากนั้นไปที่ Remote Firewall B และจากนั้นบน Remote Host C

สิ่งอำนวยความสะดวกการบันทึกการทำงาน

ขณะที่โฮสต์สื่อสารระหว่างกัน แพ็กเก็ตที่ถูกถ่ายโอน อาจถูกบันทึกการทำงานลงใน daemon บันทึกการทำงานระบบ syslogd ข้อความสำคัญอื่นๆ เกี่ยวกับ IP Security จะแสดงเช่นกัน

ผู้ดูแลระบบอาจเลือกมอนิเตอร์ข้อมูลบันทึกการทำงานนี้ เพื่อวิเคราะห์ปริมาณการรับส่งข้อมูลและการช่วยเหลือในการดีบัก ต่อไปนี้คือ ขั้นตอนสำหรับการตั้งค่าสิ่งอำนวยความสะดวกการบันทึกการทำงาน

1. แก้ไขไฟล์ /etc/syslog.conf เพื่อเพิ่ม รายการต่อไปนี้:

   ```
   local4.debug var/adm/ipsec.log
   ```

   ใช้ สิ่งอำนวยความสะดวก local4 เพื่อบันทึกปริมาณการรับส่งข้อมูลและเหตุการณ์ IP Security โดยใช้ระดับความสำคัญของระบบปฏิบัติการมาตรฐาน คุณควรตั้งค่า ระดับความสำคัญของ debug จนกว่าการรับส่งผ่าน IP Security tunnels และตัวกรองจะแสดงความเสถียรและการเคลื่อนย้ายที่เหมาะสม

หมายเหตุ: การบันทึกการทำงานของเหตุการณ์ตัวกรองอาจสร้างกิจกรรมจำนวนมากที่โฮสต์ IP Security และอาจใช้พื้นที่สื่อบันทึกเป็นจำนวนมาก

2. บันทึก `/etc/syslog.conf file`
3. ไปที่ไดเร็กทอรีที่คุณระบุสำหรับล็อกไฟล์และสร้าง ไฟล์ว่างที่มีชื่อเดียวกัน ในกรณีข้างต้น คุณควรเปลี่ยน เป็นไดเร็กทอรี `/var/adm` และเรียกใช้คำสั่ง:

   ```
   touch ipsec.log
   ```

4. เรียกใช้คำสั่ง **refresh** ไปยังระบบย่อย `syslogd`:

   ```
   refresh -s syslogd
   ```

5. ถ้าคุณกำลังใช้ IKE tunnels ทำให้แน่ใจว่าไฟล์ `/etc/isakmpd.conf` ระบุระดับการบันทึกการทำงาน **isakmpd** ที่ต้องการ (ดูที่ "การวินิจฉัยปัญหาการรักษาความปลอดภัย Internet Protocol" ในหน้า 288 เพื่อ ดูข้อมูลเพิ่มเติมเกี่ยวกับการบันทึกการทำงาน IKE)
6. ขณะกำลังสร้างกฎตัวกรองสำหรับของคุณ ถ้าคุณต้องการให้แพ็กเก็ต จับคู่กฎที่ระบุที่จะถูกบันทึกการทำงาน ให้ตั้งค่าพารามิเตอร์ *-l* สำหรับกฎเป็น **Y** (ใช่) โดยใช้คำสั่ง **genfilt** หรือ **chfilt**
7. เปิดใช้การบันทึกการทำงานแพ็กเก็ตและเริ่มทำงาน **ipsec_logd** daemon โดยใช้คำสั่ง:

   ```
   mkfilt -g start
   ```

   คุณสามารถ หยุดทำงานการบันทึกการทำงานโดยการออกคำสั่งต่อไปนี้:

   ```
   mkfilt -g stop
   ```

ตัวอย่างล็อกไฟล์ต่อไปนี้มีรายการการรับส่งข้อมูลและรายการล็อก IP Security อื่นๆ:

```
1. Aug 27 08:08:40 host1 : Filter logging daemon ipsec_logd (level 2.20)
   initialized at 08:08:40 on 08/27/97A
2. Aug 27 08:08:46 host1 : mkfilt: Status of packet logging set to Start
   at 08:08:46 on 08/27/97
3. Aug 27 08:08:47 host1 : mktun: Manual tunnel 2 for IPv4, 9.3.97.244, 9.3.97.130
   activated.
4. Aug 27 08:08:47 host1 : mkfilt: #:1 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
   udp eq  4001 eq  4001  both both l=n f=y t=0 e= a=
5. Aug 27 08:08:47 host1 : mkfilt: #:2 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
   ah any 0 any 0  both both l=n f=y t=0 e= a=
6. Aug 27 08:08:47 host1 : mkfilt: #:3 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
   esp any 0 any 0  both both l=n f=y t=0 e= a=
7. Aug 27 08:08:47 host1 : mkfilt: #:4 permit 10.0.0.1 255.255.255.255 10.0.0.2
   255.255.255.255 icmp any 0 any 0  local outbound l=y f=y t=1 e= a=
8. Aug 27 08:08:47 host1 : mkfilt: #:4 permit 10.0.0.2 255.255.255.255 10.0.0.1
   255.255.255.255 icmp any 0 any 0  local inbound l=y f=y t=1 e= a=
9. Aug 27 08:08:47 host1 : mkfilt: #:6 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
    all any 0 any 0  both both l=y f=y t=0 e= a=
10. Aug 27 08:08:47 host1 : mkfilt: Filter support (level 1.00) initialized at
    08:08:47 on 08/27/97
11. Aug 27 08:08:48 host1 : #:6 R:p  o:10.0.0.1 s:10.0.0.1 d:10.0.0.20 p:udp
    sp:3327 dp:53 r:l a:n f:n T:0 e:n l:67
12. Aug 27 08:08:48 host1 : #:6 R:p  i:10.0.0.1 s:10.0.0.20 d:10.0.0.1 p:udp
    sp:53 dp:3327 r:l a:n f:n T:0 e:n l:133
13. Aug 27 08:08:48 host1 : #:6 R:p  i:10.0.0.1 s:10.0.0.15 d:10.0.0.1 p:tcp
    sp:4649 dp:23 r:l a:n f:n T:0 e:n l:43
14. Aug 27 08:08:48 host1 : #:6 R:p  o:10.0.0.1 s:10.0.0.1 d:10.0.0.15 p:tcp
    sp:23 dp:4649 r:l a:n f:n T:0 e:n l:41
```

```
15. Aug 27 08:08:48 host1 : #:6 R:p  i:10.0.0.1 s:10.0.0.15 d:10.0.0.1 p:tcp
    sp:4649 dp:23 r:l a:n f:n T:0 e:n l:40
16. Aug 27 08:08:51 host1 : #:4 R:p  o:10.0.0.1 s:10.0.0.1 d:10.0.0.2 p:icmp
    t:8 c:0 r:l a:n f:n T:1 e:n l:84
17. Aug 27 08:08:51 host1 : #:5 R:p  i:10.0.0.1 s:10.0.0.2 d:10.0.0.1 p:icmp
    t:0 c:0 r:l a:n f:n T:1 e:n l:84
18. Aug 27 08:08:52 host1 : #:4 R:p  o:10.0.0.1 s:10.0.0.1 d:10.0.0.2 p:icmp
    t:8 c:0 r:l a:n f:n T:1 e:n l:84
19. Aug 27 08:08:52 host1 : #:5 R:p  i:10.0.0.1 s:10.0.0.2 d:10.0.0.1 p:icmp
    t:0 c:0 r:l a:n f:n T:1 e:n l:84
20. Aug 27 08:32:27 host1 : Filter logging daemon terminating at 08:32:27 on
    08/27/971
```

ย่อหน้าต่อไปนี้จะอธิบายเกี่ยวกับรายการบันทึกการทำงาน

1 daemon การบันทึกการทำงานตัวกรองถูกเรียกทำงาน

2 การบันทึกการทำงานแพ็กเก็ตตัวกรองถูกตั้งค่าเป็น on ด้วยคำสั่ง **mkfilt -g start**

3 การเรียกทำงาน Tunnel การแสดง tunnel ID แอดเดรสต้นทาง แอดเดรส ปลายทาง และการประทับเวลา

4-9 ตัวกรองถูกเรียกทำงาน การบันทึกการทำงานแสดงกฎตัวกรองที่ถูกโหลดทั้งหมด

10 ข้อความแสดงการเรียกทำงานของตัวกรอง

11-12
รายการเหล่านี้แสดงการค้นหา DNS สำหรับโฮสต์

13-15
รายการเหล่านี้แสดงการเชื่อมต่อ Telnet บางส่วน (รายการอื่นๆ ถูกลบออกจากตัวอย่างนี้ด้วยเหตุผลด้านพื้นที่

16-19
รายการเหล่านี้แสดงสอง pings

20 daemon การบันทึกการทำงานตัวกรองกำลังปิดทำงาน

ตัวอย่างต่อไปนี้แสดงการเจรจาของโฮสต์สองโฮสต์ใน tunnel เฟส 1 และ เฟส 2 จากมุมมองของโฮสต์ที่เป็นผู้เริ่ม (ระดับการบันทึกการทำงาน **isakmpd** ได้ถูกระบุเป็น **isakmp_events**)

```
1. Dec  6 14:34:42 host1 Tunnel Manager: 0: TM is processing a
   Connection_request_msg
 2. Dec  6 14:34:42 host1 Tunnel Manager: 1: Creating new P1 tunnel object (tid)
 3. Dec  6 14:34:42 host1 isakmpd: 192.168.100.103 >>> 192.168.100.104 ( SA PROPOSAL
    TRANSFORM  )
 4. Dec  6 14:34:42 host1 isakmpd: ::ffff:192.168.100.103 <<< 192.168.100.104 ( SA
    PROPOSAL TRANSFORM  )
 5. Dec  6 14:34:42 host1 isakmpd: Phase I SA Negotiated
 6. Dec  6 14:34:42 host1 isakmpd: 192.168.100.103 >>> 192.168.100.104 ( KE NONCE  )
 7. Dec  6 14:34:42 host1 isakmpd: ::ffff:192.168.100.103 <<< 192.168.100.104 ( KE
    NONCE  )
 8. Dec  6 14:34:42 host1 isakmpd: Encrypting the following msg to send: ( ID HASH
     )
 9. Dec  6 14:34:42 host1 isakmpd: 192.168.100.103 >>> 192.168.100.104 ( Encrypted
    Payloads )
10. Dec  6 14:34:42 host1 isakmpd: ::ffff:192.168.100.103 <<< 192.168.100.104 (
    Encrypted Payloads )
```

```
11. Dec  6 14:34:42 host1 Tunnel Manager: 1: TM is processing a P1_sa_created_msg
    (tid)
12. Dec  6 14:34:42 host1 Tunnel Manager: 1:   Received good P1 SA, updating P1
    tunnel (tid)
13. Dec  6 14:34:42 host1 Tunnel Manager: 0: Checking to see if any P2 tunnels need
    to start
14. Dec  6 14:34:42 host1 isakmpd: Decrypted the following received msg: ( ID HASH
     )
15. Dec  6 14:34:42 host1 isakmpd:  Phase I Done !!!
16. Dec  6 14:34:42 host1 isakmpd: Phase I negotiation authenticated
17. Dec  6 14:34:44 host1 Tunnel Manager: 0: TM is processing a
    Connection_request_msg
18. Dec  6 14:34:44 host1 Tunnel Manager: 0: Received a connection object for an
    active P1 tunnel
19. Dec  6 14:34:44 host1 Tunnel Manager: 1: Created blank P2 tunnel (tid)
20. Dec  6 14:34:44 host1 Tunnel Manager: 0: Checking to see if any P2 tunnels need
    to start
21. Dec  6 14:34:44 host1 Tunnel Manager: 1: Starting negotiations for P2 (P2 tid)
22. Dec  6 14:34:45 host1 isakmpd: Encrypting the following msg to send: ( HASH SA
    PROPOSAL TRANSFORM NONCE ID ID  )
23. Dec  6 14:34:45 host1 isakmpd: 192.168.100.103 >>> 192.168.100.104 ( Encrypted
    Payloads )
24. Dec  6 14:34:45 host1 isakmpd: ::ffff:192.168.100.103 <<< 192.168.100.104 (
    Encrypted Payloads )
25. Dec  6 14:34:45 host1 isakmpd: Decrypted the following received msg: ( HASH SA
    PROPOSAL TRANSFORM NONCE ID ID  )
26. Dec  6 14:34:45 host1 isakmpd: Encrypting the following msg to send: ( HASH  )
27. Dec  6 14:34:45 host1 isakmpd: 192.168.100.103 >>> 192.168.100.104 ( Encrypted
    Payloads )
28. Dec  6 14:34:45 host1 isakmpd: Phase II SA Negotiated
29. Dec  6 14:34:45 host1 isakmpd: PhaseII negotiation complete.
30. Dec  6 14:34:45 host1 Tunnel Manager: 0: TM is processing a P2_sa_created_msg
31. Dec  6 14:34:45 host1 Tunnel Manager: 1: received p2_sa_created for an existing
    tunnel as initiator (tid)
32. Dec  6 14:34:45 host1 Tunnel Manager: 1: Filter::AddFilterRules: Created filter
    rules for tunnel
33. Dec  6 14:34:45 host1 Tunnel Manager: 0: TM is processing a List_tunnels_msg
```

ย่อหน้าต่อไปนี้จะอธิบายเกี่ยวกับรายการบันทึกการทำงาน

1-2 คำสั่ง **ike cmd=activate phase=1** เริ่ม การเชื่อมต่อ

3-10 **isakmpd** daemon เจรจรกับ tunnel เฟส 1

11-12

Tunnel Manager ได้รับการเชื่อมโยงความปลอดภัยเฟส 1 ที่ถูกต้อง จากผู้ตอบกลับ

13 Tunnel Manager ตรวจสอบว่า **ike cmd=activate** มี ค่าเฟส 2 เพื่อทำงานต่อ ไม่มีค่าเพื่อทำงานต่อ

14-16

isakmpd daemon ดำเนินการเจรจาเฟส 1 เสร็จ

17-21

คำสั่ง **ike cmd=activate phase=2** เริ่ม tunnel เฟส 2

22-29

isakmpd daemon เจรจรกับ tunnel เฟส 2

30-31

Tunnel Manager ได้รับการเชื่อมโยงความปลอดภัยเฟส 2 ที่ถูกต้อง จากผู้ตอบกลับ

32 Tunnel Manager เขียนกฎตัวกรองแบบไดนามิก

33 คำสั่ง **ike cmd=list** ดู IKE tunnels

เลเบลในรายการฟิลด์:

ฟิลด์ในรายการบันทึกการทำงานที่ถูกย่อเพื่อลด ความต้องการใช้พื้นที่ DASD

ฟิลด์	ความหมาย
#	หมายเลขกฎที่เป็นสาเหตุให้แพ็กเก็ตนี้ถูกบันทึกการทำงาน
R	ประเภทกฎ **p** อนุญาต **d** ปฏิเสธ
i/o	ทิศทางที่แพ็กเก็ตกำลังไปเมื่อถูกขัดขวาง โดยโค้ดที่สนับสนุนการกรอง ระบุ IP address ของอะแด็ปเตอร์ที่เชื่อมโยงกับ แพ็กเก็ต: • สำหรับแพ็กเก็ตขาเข้า (i) นี้คืออะแด็ปเตอร์ที่แพ็กเก็ต มาถึง • สำหรับแพ็กเก็ตขาออก (o) นี้คืออะแด็ปเตอร์ที่ IP layer พิจารณาว่าควรจัดการการส่งข้อมูลแพ็กเก็ต
s	ระบุ IP address ของผู้ส่งแพ็กเก็ต (แยก ออกมาจากส่วนหัว IP)
d	ระบุ IP address ของผู้รับแพ็กเก็ต ที่ต้องการ (แยกออกมาจากส่วนหัว IP)
p	ระบุโปรโตคอลระดับสูงที่ใช้สร้าง ข้อความในส่วนข้อมูลของแพ็กเก็ต อาจเป็นหมายเลขหรือ ชื่อ ตัวอย่าง: `udp`, `icmp`, `tcp`, `tcp/ack`, `ospf`, `pip`, `esp`, `ah` หรือ `all`
sp/t	ระบุหมายเลขพอร์ตโปรโตคอลที่เชื่อมโยงกับผู้ส่ง แพ็กเก็ต (แยกออกมาจากส่วนหัว TCP/UDP) เมื่อโปรโตคอล เป็น ICMP หรือ OSPF ฟิลด์นี้จะถูกแทนที่ด้วย **t** ซึ่งระบุ ประเภท IP
dp/c	ระบุหมายเลขพอร์ตโปรโตคอลที่เชื่อมโยงกับผู้รับ แพ็กเก็ตที่ต้องการ (แยกออกมาจากส่วนหัว TCP/UDP) เมื่อ โปรโตคอลเป็น ICMP ฟิลด์นี้จะถูกแทนที่ด้วย **c** ซึ่ง ระบุโค้ด IP
-	ระบุว่าไม่มีข้อมูล
r	บ่งชี้ว่าแพ็กเก็ตมี affiliation โลคัลใดๆ **f** แพ็กเก็ตที่ส่งต่อ **l** แพ็กเก็ตโลคัล **o** ขาออก **b** ทั้งสอง
l	ระบุความยาวของแพ็กเก็ตเฉพาะเป็นไบต์
f	ระบุว่าแพ็กเก็ตเป็นแฟรกเมนต์
T	บ่งชี้ tunnel ID
i	ระบุอินเตอร์เฟสใดที่แพ็กเก็ตที่เข้ามาใช้

การบันทึกการทำงาน Internet Key-Exchange:

คุณสามารถเปิดใช้การบันทึกการทำงานเหตุการณ์ Internet Key-Exchange ลงในโปรแกรมอำนวยความสะดวก SYSLOG ที่มี **isakmpd** daemon

สำหรับ **isakmpd** daemon คุณเปิดใช้การบันทึกการทำงานโดยใช้คำสั่ง **ike cmd=log** คุณสามารถตั้งค่าระดับการบันทึกการทำงานในไฟล์คอนฟิกูเรชัน `/etc/isakmpd.conf` ที่มีพารามิเตอร์ `log_level` โดยขึ้นกับ จำนวนข้อมูลที่คุณต้องการบันทึกการทำงาน คุณสามารถตั้งค่า ระดับเป็น *none*, *errors*, *isakmp_events*, หรือ *information*

ตัวอย่าง ในการระบุว่าคุณต้องการบันทึกข้อมูลโปรโตคอล และข้อมูลการนำไปปฏิบัติ ให้ระบุพารามิเตอร์ต่อไปนี้:

```
log_level=INFORMATION
```

isakmpd daemon เริ่มทำงานหนึ่งในสองกระบวนการ โดยส่ง ข้อเสนอ หรือประเมินข้อเสนอ ถ้าข้อเสนอได้รับการยอมรับ การเชื่อมโยงด้านความปลอดภัยจะถูกสร้าง และ tunnel ถูกตั้งค่า ถ้า ข้อเสนอไม่ได้รับการยอมรับหรือการเชื่อมต่อสิ้นสุดลงก่อนการเจรจา จะเสร็จสมบูรณ์ **isakmpd** daemon บ่งชี้ข้อผิดพลาด รายการ ในโปรแกรมอำนวยความสะดวก SYSLOG จาก **tmd** บ่งชี้ว่าการเจรจา สำเร็จ ความล้มเหลวมีสาเหตุโดยใบรับรองที่ใช้ไม่ได้ ถูกบันทึกการทำงานลงในโปรแกรมอรรถประโยชน์ SYSLOG ในการพิจารณาสาเหตุที่แท้จริงของ การเจรจาที่ล้มเหลว ให้ตรวจทานข้อมูลในล็อกไฟล์ที่ถูกระบุ ใน `/etc/syslog.conf`

โปรแกรมอรรถประโยชน์ SYSLOG เพิ่มส่วนนำหน้าให้แก่แต่ละบรรทัดของล็อก บันทึก วันที่และเวลา เครื่อง และโปรแกรม ตัวอย่างต่อไปนี้ใช้ `googly` เป็นชื่อเครื่อง และ `isakmpd` เป็น ชื่อโปรแกรม:

```
Nov 20 09:53:50 googly isakmpd: ISAKMP_MSG_HEADER
Nov 20 09:53:50 googly isakmpd:  Icookie : 0xef06a77488f25315, Rcookie :0x0000000000000000
Nov 20 09:53:51 googly isakmpd:  Next Payload : 1(SA), Maj Ver : 1, Min Ver : 0
Nov 20 09:53:51 googly isakmpd:  Xchg Type : 2 (ID protected), Flag= 0, Encr : No,COMMIT : No
Nov 20 09:53:51 googly isakmpd:  Msg ID  : 0x00000000
```

เพื่อเพิ่มความชัดเจน ใช้คำสั่ง **grep** เพื่อ แยกบรรทัดล็อกที่สนใจ (เช่น การบันทึก **isakmpd** ทั้งหมด) และคำสั่ง **cut** เพื่อลบส่วนนำหน้าออกจาก แต่ละบรรทัด

ไฟล์ /etc/isakmpd.conf:

คุณสามารถตั้งค่าอ็อพชันสำหรับ **isakmpd** daemon ในไฟล์ `/etc/isakmpd.conf`

อ็อพชันต่อไปนี้มีอยู่ในไฟล์ `/etc/isakmpd.conf`

การตั้งค่าบันทึกการทำงาน

พิจารณาจำนวนข้อมูลที่คุณต้องการบันทึก จากนั้นตั้งค่า ระดับ IKE daemons ใช้อ็อพชันนี้เพื่อระบุระดับของบันทึกการทำงาน

ไวยากรณ์: `none | error | isakmp_events | information`

โดยที่ระดับ มีความหมายต่อไปนี้:

none ไม่มีการบันทึกการทำงาน นี้เป็นค่าดีฟอลต์

error บันทึกข้อผิดพลาดโปรโตคอลหรือข้อผิดพลาด appliation programming interface (API)

isakmp_events
บันทึกเหตุการณ์หรือข้อผิดพลาดโปรโตคอล IKE ใช้ระดับนี้เมื่อต้องการดีบักปัญหา

information
บันทึกข้อมูลโปรโตคอลและข้อมูลการนำไปปฏิบัติ

การเจรจา IP address ที่ไม่รู้จัก

คุณสามารถตั้งค่าอ็อพชันนี้เป็น YES หรือ NO เมื่อคุณตั้งค่าอ็อพชันนี้เป็น YES ฐานข้อมูล IKE โลคัลต้องมี IP address สำหรับทั้งจุดหมาย phase-1 tunnel ทั้งสอง คุณต้องระบุ YES สำหรับโฮสต์เพื่อยอมรับ tunnel โหมดหลักขาเข้า IP address สามารถเป็น ID หลักหรือ IP address ทางเลือกที่ เชื่อมโยงกับประเภท ID อื่นบางประเภท

ตั้งค่าอ็อพชันนี้เป็น NO เพื่อยอมรับ การเชื่อมโยงโหมดหลักขาเข้า เมื่อคุณตั้งค่าอ็อพชันเป็น NO โฮสต์ อาจยอมรรับการเชื่อมต่อแม้ว่าฐานข้อมูล IKE จะไม่ได้ระบุ IP addresses สำหรับจุดหมายเฟส 1 อย่างไรก็ตาม เพื่อให้โฮสต์ยอมรับ การเชื่อมต่อ คุณต้องใช้การพิสูจน์ตัวตนโดยใช้ใบรับรอง ซึ่งอนุญาต ให้โฮสต์ที่มี IP address ที่กำหนดแบบไดนามิกเพื่อเริ่มต้น tunnel โหมดหลัก กับเครื่อง

ถ้าคุณไม่ระบุพารามิเตอร์นี้ ค่าดีฟอลต์เป็น NO

ไวยากรณ์: MAIN_MODE_REQUIRES_IP= YES | NO

การตั้งค่าเซิร์ฟเวอร์ SOCKS4

อ็อพชัน SOCKS4_PORTNUM เป็นทางเลือก ถ้าคุณไม่ ระบุค่า ค่าพอร์ต SOCKS-server ดีฟอลต์คือ 1080 จะถูกใช้ ค่าพอร์ตถูกใช้เมื่อ SOCKS server สื่อสารกับเซิร์ฟเวอร์ HTTP

ไวยากรณ์: *mnemonic* = *value*
โดยที่ *mneumonic* และ *value* สามารถ เป็นค่าต่อไปนี้:

SOCKS4_SERVER= ระบุชื่อเซิร์ฟเวอร์

SOCKS4_PORTNUM= ระบุหมายเลขพอร์ต SOCKS-server

SOCKS4_USERID= ID ผู้ใช้

การตั้งค่าเซิร์ฟเวอร์ LDAP

ไวยากรณ์: *mnemonic* = *value*
โดยที่ *mnemonic* และ *value* สามารถ เป็นค่าต่อไปนี้:

LDAP_SERVER= ระบุชื่อเซิร์ฟเวอร์ LDAP

LDAP_VERSION= เวอร์ชันของเซิร์ฟเวอร์ LDAP (สามารถเป็น 2 หรือ 3)

LDAP_SERVERPORT= หมายเลขพอร์ต LDAP-server

LDAP_SEARCHTIME= ค่าหมดเวลาใช้งานการค้นหาไคลเอ็นต์

ลำดับการดึงใช้ CRL

อ็อพชันนี้กำหนดว่าเซิร์ฟเวอร์ HTTP หรือ LDAP ถูกเคียวรีเป็นอันดับแรก เมื่อทั้งสองเซิร์ฟเวอร์ถูกตั้งค่า อ็อพชัน CRL_FETCH_ORDER เป็นทางเลือก ลำดับการดึงใช้ดีฟอลต์คือ HTTP อันดับแรก จากนั้น LDAP ทั้งนี้ขึ้นกับ ว่าทั้งเซิร์ฟเวอร์ HTTP และ LDAP ถูกตั้งค่า

ไวยากรณ์: CRL_FETCH_ORDER= *protocol#*, *protocol#*
โดยที่ *protocol#* สามารถ เป็น HTTP หรือ LDAP

ข้อกำหนดคุณสมบัติพอร์ต IKEv1 และ IKEv2

สตริงนี้ระบุพอร์ตที่ใช้โดย **isakmpd** daemon (IKEv1) และ **ikev2d** daemon (IKEv2) **iked** daemon (IKE message broker daemon) ค้นหารายการนี้และเริ่มทำงาน **isakmpd** daemon และ **ikev2d** daemon บนพอร์ตตามลำดับ

ไวยากรณ์: v1=port-natport,v2=port-natport

การวินิจฉัยปัญหาการรักษาความปลอดภัย Internet Protocol

ต่อไปนี้เป็นคำแนะนำและเคล็ดลับบางอย่างที่อาจช่วยคุณได้ เมื่อคุณประสบปัญหา

ตั้งค่าการบันทึกการทำงานเมื่อตั้งค่า IPSec เป็นครั้งแรก การบันทึกการทำงานเป็นประโยชน์อย่างมากใน การใช้พิจารณาสิ่งที่เกิดขึ้นกับตัวกรองและ tunnels (สำหรับข้อมูลบันทึกการทำงานโดยละเอียด ดูที่ “สิ่งอำนวยความสะดวกการบันทึกการทำงาน” ในหน้า 282)

ในการพิจารณาว่า IP security daemons ใดกำลังทำงาน ให้ป้อนคำสั่งต่อไปนี้:

```
ps -ef
```

daemons ต่อไปนี้ เชื่อมโยงกับ IP security: **tmd, iked, isakmpd, ikev2d, cpsd**

หมายเหตุ: ถ้า ทั้ง IKEv1 และ IKEv2 ถูกตั้งค่า **iked** daemon จะทำงาน มิฉะนั้น **iskmpd** daemon จะทำงานหรือ **ikev2d** daemon จะทำงาน การตั้งค่า นี้อยู่ในไฟล์ **/etc/isakmpd.conf**

การแก้ปัญหาข้อผิดพลาด manual tunnel:

ต่อไปนี้คือรายละเอียดข้อผิดพลาด tunnel ต่างๆ ที่เป็นไปได้ พร้อมทั้งวิธีแก้ปัญหา

Error	**ปัญหาและโซลูชันที่เป็นไปได้**
การออกคำสั่ง **mktun** เกิดข้อผิดพลาด ต่อไปนี้: `insert_tun_man4(): write failed : The requested resource is busy.`	ปัญหา: tunnel ที่คุณร้องขอให้เรียกทำงานนั้นเรียกทำงานอยู่แล้ว หรือคุณ มีค่า SPI ที่ขัดแย้งกัน วิธีแก้ไข: ออกคำสั่ง **rmtun** เพื่อปิดทำงาน จากนั้นออกคำสั่ง mktun เพื่อเรียกทำงาน ตรวจสอบเพื่อดูว่าค่า SPI สำหรับ failing tunnel ตรงกับ tunnel อื่นใด ที่แอ็คทีฟหรือไม่ แต่ละ tunnel ควรมีค่า SPI เฉพาะของตนเอง

การออกคำสั่ง **mktun** เกิดข้อผิดพลาด ต่อไปนี้:
 	
`อุปกรณ์ ipsec_v4 อยู่ในสถานะ Defined`
 	
`การเรียกทำงาน Tunnel สำหรับ IP Version 4 ไม่ถูกดำเนินการ`
	ปัญหา: คุณยังไม่ได้ทำงานอุปกรณ์ IP Security พร้อมใช้งาน
	
	วิธีแก้ไข: ออกคำสั่งต่อไปนี้:
	`mkdev -l ipsec -t 4`
	
	คุณ อาจต้องเปลี่ยนอ็อพชัน **-t** เป็น 6 ถ้าคุณได้รับข้อผิดพลาด เหมือนกันสำหรับการเรียกทำงาน IP Version 6 tunnel อุปกรณ์ต้องอยู่ใน สถานะพร้อมใช้งาน ในการตรวจสอบสถานะอุปกรณ์ IP Security ให้ออก คำสั่งต่อไปนี้:
	`lsdev -Cc ipsec`

การออกคำสั่ง **gentun** เกิดข้อผิดพลาด ต่อไปนี้: `Invalid Source IP address`	ปัญหา: คุณไม่ได้ป้อน IP address ที่ถูกต้องสำหรับแอดเดรสต้นทาง วิธี แก้ไข: สำหรับ IP Version 4 tunnels ให้ตรวจสอบเพื่อดูว่า คุณได้ป้อน IP Version 4 address ที่มีสำหรับเครื่องโลคัล คุณไม่สามารถใช้ชื่อโฮสต์สำหรับต้นทางเมื่อสร้าง tunnels คุณสามารถใช้ชื่อโฮสต์สำหรับปลายทางเท่านั้น สำหรับ IP Version 6 tunnels ตรวจสอบเพื่อดูว่าคุณป้อน IP Version 6 address ที่มีอยู่ ถ้าคุณพิมพ์ `netstat -in` และไม่มี IP Version 6 addresses อยู่ ให้รัน `/usr/sbin/autoconf6` (อินเตอร์เฟส) สำหรับ แอดเดรสที่สร้างอัตโนมัติบนลิงก์โลคัล (โดยใช้ MAC address) หรือใช้คำสั่ง **ifconfig** เพื่อกำหนดแอดเดรสด้วยตนเอง

Error	ปัญหาและโซลูชันที่เป็นไปได้
การออกคำสั่ง **gentun** เกิดข้อผิดพลาด ต่อไปนี้: `Invalid Source IP address`	ปัญหา: คุณไม่ได้ป้อน IP address ที่ถูกต้องสำหรับแอดเดรสต้นทาง วิธี แก้ไข: สำหรับ IP Version 4 tunnels ให้ตรวจสอบเพื่อดูว่า คุณได้ป้อน IP Version 4 address ที่มีสำหรับเครื่องโลคัล คุณไม่สามารถ ใช้ชื่อโฮสต์สำหรับต้นทางเมื่อสร้าง tunnels คุณสามารถใช้ชื่อโฮสต์สำหรับปลายทางเท่านั้น สำหรับ IP Version 6 tunnels ตรวจสอบเพื่อดูว่าคุณป้อน IP Version 6 address ที่มีอยู่ ถ้า คุณพิมพ์ `netstat -in` และไม่มี IP Version 6 addresses อยู่ให้รัน `/usr/sbin/autoconf6` (อินเตอร์เฟส) สำหรับแอดเดรส ที่สร้างอัตโนมัติบนลิงก์โลคัล (โดยใช้ MAC address) หรือใช้ **ifconfig** เพื่อ กำหนดแอดเดรสด้วยตนเอง
การออกคำสั่ง **mktun** เกิดข้อผิดพลาด ต่อไปนี้: `insert_tun_man4(): write failed : A system call received a parameter that is not valid.`	ปัญหา: การสร้าง Tunnel เกิดขึ้นกับการรวม ESP และ AH ที่ไม่ถูกต้องหรือ ไม่มีการใช้รูปแบบส่วนหัวใหม่เมื่อจำเป็น วิธี แก้ไข: ตรวจสอบเพื่อดูว่าอัลกอริทึมการพิสูจน์ตัวตนใดใช้งานโดย tunnel เฉพาะที่มีปัญหา จำไว้ว่าอัลกอริทึม HMAC_MD5 และ HMAC_SHA จำเป็นต้องใช้รูปแบบส่วนหัวใหม่ รูปแบบส่วนหัวใหม่สามารถ เปลี่ยนได้โดยใช้พาธด่วน SMIT **ips4_basic** หรือพารามิเตอร์ *-z* กับคำสั่ง **chtun** รวมทั้งจำว่าไม่สามารถใช้ DES_CBC_4 กับรูปแบบส่วนหัวใหม่
การพยายามใช้ IP Security เกิดข้อผิดพลาดต่อไปนี้: `The installed bos.crypto is back level and must be updated.`	ปัญหา: ไฟล์ `bos.net.ipsec.*` ได้ถูกอัพเดตเป็น เวอร์ชันใหม่ แต่ไฟล์ `bos.crypto.*` ที่สอดคล้องยังไม่ได้รับการอัพเดต วิธีแก้ไข: อัพเดตไฟล์ `bos.crypto.*` เป็นเวอร์ชันที่สอดคล้องกับไฟล์ `bos.net.ipsec.*` ที่อัพเดต

การแก้ปัญหาข้อผิดพลาด Internet Key Exchange tunnel:

ส่วนต่อไปนี้อธิบายข้อผิดพลาดที่สามารถเกิดขึ้นได้เมื่อใช้ Internet Key Exchange (IKE) tunnels

โฟลว์กระบวนการ Internet Key Exchange tunnel:

ส่วนนี้อธิบายโฟลว์กระบวนการสำหรับ internet key exchange tunnel

IKE tunnels ถูกตั้งค่าโดยการสื่อสารของคำสั่ง **ike** ด้วย daemons ต่อไปนี้:

tmd Tunnel Manager daemon

iked IKE broker daemon (แอ็คทีฟเมื่อทั้ง IKEv1 และ IKEv2 daemons ถูกตั้งค่าบนระบบเท่านั้น)

isakmpd
IKEv1 daemon

ikev2d IKEv2 daemon

cpsd Certificate proxy daemon

เพื่อให้ IKE tunnels ตั้งค่าอย่างถูกต้อง **tmd** and **isakmpd** daemons ต้องทำงานอยู่ ถ้า IP Security ถูกตั้งค่าให้เริ่มทำงานตอนบูตใหม่ daemons เหล่านี้จะเริ่มทำงาน โดยอัตโนมัติ หรือ, เริ่มต้นทำงานโดยป้อนคำสั่งต่อไปนี้:

```
startsrc -g ike
```

Tunnel Manager ให้การร้องขอไปยังคำสั่ง **isakmpd** เพื่อเริ่มทำงาน tunnel ถ้า tunnel มีอยู่แล้วหรือใช้ไม่ได้ (ตัวอย่างเช่น มีรีโมตแอดเดรสที่ไม่ถูกต้อง) จะมีการรายงานข้อผิดพลาด ถ้าการเจรจาได้เริ่มทำงาน อาจต้องใช้เวลาชั่วครู่ ขึ้นกับเวลาแฝงเน็ตเ

วิร์ก ที่การเจรจาจะเสร็จ สมบูรณ์ คำสั่ง **ike cmd=list** สามารถแสดงรายการสถานะของ tunnel เพื่อพิจารณาว่าการเจรจาสำเร็จ รวมทั้ง Tunnel Manager บันทึกเหตุการณ์ไปยัง `syslog` ตามระดับของ debug, event และ information ซึ่งสามารถใช้มอนิเตอร์ความคืบหน้า ของการเจรจา

ลำดับเป็นดังนี้:

1. ใช้คำสั่ง **ike** เพื่อเริ่มทำงาน tunnel
2. **tmd** daemon ส่งการร้องขอให้ **isakmpd** daemon สำหรับการจัดการคีย์ (เฟส 1)
3. **isakmpd** daemon ตอบกลับโดย SA `created` หรือ ข้อความแสดงความผิดพลาด
4. **tmd** daemon ส่งการร้องขอให้ **isakmpd** daemon สำหรับ tunnel การจัดการข้อมูล (เฟส 2)
5. **isakmpd** daemon ตอบกลับโดย SA `created` หรือ ข้อความแสดงความผิดพลาด
6. พารามิเตอร์ Tunnel ถูกแทรกในแคช tunnel เคอร์เนล
7. กฎตัวกรองถูกเพิ่มในตารางตัวกรองแบบไดนามิกของเคอร์เนล

เมื่อเครื่องทำหน้าที่เป็นผู้ตอบกลับ **isakmpd** daemon แจ้ง Tunnel Manager **tmd** daemon ว่า tunnel ได้เจรจา เสร็จเรียบร้อย และ tunnel ใหม่ถูกแทรกในเคอร์เนล ในกรณีเช่นนี้ กระบวนการเริ่ทำงานขั้นตอน 3 และทำต่อไปจนถึงขั้นตอน 7 โดยไม่มี **tmd** daemon ออกการร้องขอการเชื่อมต่อ

ฟังก์ชันการแยกวิเคราะห์ Parse payload:

Security association (SA) ระหว่างจุดหมายสองสุด ถูกสร้างขึ้นโดยการแลกเปลี่ยนข้อความ IKE ฟังก์ชัน Parse Payload ทำหน้าที่แยกวิเคราะห์ข้อความที่อยู่ในรูปแบบที่มนุษย์อ่านได้

การบันทึกการทำงาน Parse payload สามารถเปิดใช้งานได้โดยการแก้ไขไฟล์ `/etc/isakmpd.conf` รายการบันทึกการทำงานในไฟล์ `/etc/isakmpd.conf` มีลักษณะคล้ายตัวอย่างต่อไปนี้:

```
information
```

ประเภทของ IKE payloads ที่การบันทึกการทำงาน Parse Payload ขึ้นอยู่กับ เนื้อหาของข้อความ IKE ตัวอย่างประกอบด้วย SA Payload, Key Exchange Payload, Certificate Request Payload, Certificate Payload และ Signature Payload ต่อไปนี้เป็นตัวอย่างของบันทึกการทำงาน Parse Payload ที่ ISAKMP_MSG_HEADER ตามด้วยห้า payloads:

```
ISAKMP_MSG_HEADER
        Icookie : 0x9e539a6fd4540990, Rcookie : 0x0000000000000000
        Next Payload : 1(SA), Maj Ver : 1, Min Ver : 0
        Xchg Type : 4 (Aggressive), Flag= 0, Encr : No,COMMIT : No
        Msg ID  : 0x00000000
        len     : 0x10e(270)
SA Payload:
        Next Payload : 4(Key Exchange), Payload len : 0x34(52)
        DOI          : 0x1(INTERNET)
        bitmask      : 1(SIT_IDENTITY_ONLY
   Proposal Payload:
        Next Payload : 0(NONE), Payload len : 0x28(40)
        Proposal # : 0x1(1), Protocol-ID : 1(ISAKMP)
        SPI size : 0x0(0), # of Trans : 0x1(1)
   Transform Payload:
        Next Payload : 0(NONE), Payload len : 0x20(32)
        Trans # : 0x1(1), Trans.ID : 1(KEY_IKE)
```

```
        Attr : 1(Encr.Alg     ), len=0x2(2)
        Value=0x1(1),(DES-cbc)
        Attr : 2(Hash Alg     ), len=0x2(2)
        Value=0x1(1),(MD5)
        Attr : 3(Auth Method  ), len=0x2(2)
        Value=0x3(3),(RSA Signature)
        Attr : 4(Group Desc   ), len=0x2(2)
        Value=0x1(1),(default 768-bit MODP group)
        Attr : 11(Life Type    ), len=0x2(2)
        Value=0x1(1),(seconds)
        Attr : 12(Life Duration), len=0x2(2)
        Value=0x7080(28800)
Key Payload:
        Next Payload : 10(Nonce), Payload len : 0x64(100)

        Key Data :
        33 17 68 10 91 1f ea da 38 a0 22 2d 84 a3 5d 5d
        a0 e1 1f 42 c2 10 aa 8d 9d 14 0f 58 3e c4 ec a3
        9f 13 62 aa 27 d8 e5 52 8d 5c c3 cf d5 45 1a 79
        8a 59 97 1f 3b 1c 08 3e 2a 55 9b 3c 50 cc 82 2c
        d9 8b 39 d1 cb 39 c2 a4 05 8d 2d a1 98 74 7d 95
        ab d3 5a 39 7d 67 5b a6 2e 37 d3 07 e6 98 1a 6b

Nonce Payload:
        Next Payload : 5(ID), Payload len : 0xc(12)

        Nonce Data:
        6d 21 73 1d dc 60 49 93
ID Payload:
        Next Payload : 7(Cert.Req), Payload len : 0x49(73)
        ID type      : 9(DER_DN), Protocol : 0, Port = 0x0(0)
Certificate Request Payload:
        Next Payload : 0(NONE), Payload len : 0x5(5)
        Certificate Encoding Type: 4(X.509 Certificate - Signature)
```

ภายในแต่ละ payload ฟิลด์ **Next Payload** ชี้ไปที่ payload ที่ต่อจาก payload ปัจจุบัน ถ้า payload ปัจจุบันเป็นอันสุดท้ายในข้อความ IKE ฟิลด์ **Next Payload** มีค่าเป็นศูนย์ (ไม่มี)

แต่ละ Payload ในตัวอย่างมีข้อมูลเกี่ยวกับการเจรจา ที่กำลังดำเนินอยู่ ตัวอย่าง SA payload มี Proposal and Transform Payloads ซึ่งแสดงอัลกอริทึมการเข้ารหัส โหมด การพิสูจน์ตัวตน อัลกอริทึมการแฮช ประเภทช่วงอายุ SA และช่วงเวลา SA ที่ผู้เริ่ม กำลังเสนอไปยังผู้ตอบกลับ

รวมทั้ง SA Payload ประกอบด้วย Proposal Payloads อย่างน้อยหนึ่ง payloads และ Transform Payloads อย่างน้อยหนึ่ง payloads ฟิลด์ **Next Payload** สำหรับ Proposal Payload มีค่าเป็น 0 ถ้ามี Proposal Payloads ค่าเดียว หรือค่าเป็น 2 ถ้าตามด้วย Proposal Payloads อย่างน้อยหนึ่ง payloads ในทำนองเดียวกัน ฟิลด์ **Next Payload** สำหรับ Transform Payload มีค่าเป็น 0 ถ้ามี Transform Payload ค่าเดียว หรือ ค่าเป็น 3 ถ้าตามด้วย Transform Payloads อย่างน้อยหนึ่ง payloads ดังแสดง ในตัวอย่างต่อไปนี้:

```
ISAKMP_MSG_HEADER
        Icookie : 0xa764fab442b463c6, Rcookie : 0x0000000000000000
        Next Payload : 1(SA), Maj Ver : 1, Min Ver : 0
        Xchg Type : 2 (ID protected), Flag= 0, Encr : No,COMMIT : No
        Msg ID  : 0x00000000
```

```
        len     : 0x70(112)
SA Payload:
        Next Payload : 0(NONE), Payload len : 0x54(84)
        DOI          : 0x1(INTERNET)
        bitmask      : 1(SIT_IDENTITY_ONLY
   Proposal Payload:
        Next Payload : 0(NONE), Payload len : 0x48(72)
        Proposal # : 0x1(1), Protocol-ID : 1(ISAKMP)
        SPI size : 0x0(0), # of Trans : 0x2(2)
   Transform Payload:
        Next Payload : 3(Transform), Payload len : 0x20(32)
        Trans # : 0x1(1), Trans.ID : 1(KEY_IKE)
        Attr : 1(Encr.Alg     ), len=0x2(2)
        Value=0x5(5),(3DES-cbc)
        Attr : 2(Hash Alg     ), len=0x2(2)
        Value=0x1(1),(MD5)
        Attr : 3(Auth Method  ), len=0x2(2)
        Value=0x1(1),(Pre-shared Key)
        Attr : 4(Group Desc   ), len=0x2(2)
        Value=0x1(1),(default 768-bit MODP group)
        Attr : 11(Life Type    ), len=0x2(2)
        Value=0x1(1),(seconds)
        Attr : 12(Life Duration), len=0x2(2)
        Value=0x7080(28800)
   Transform Payload:
        Next Payload : 0(NONE), Payload len : 0x20(32)
        Trans # : 0x2(2), Trans.ID : 1(KEY_IKE)
        Attr : 1(Encr.Alg     ), len=0x2(2)
        Value=0x1(1),(DES-cbc)
        Attr : 2(Hash Alg     ), len=0x2(2)
        Value=0x1(1),(MD5)
        Attr : 3(Auth Method  ), len=0x2(2)
        Value=0x1(1),(Pre-shared Key)
        Attr : 4(Group Desc   ), len=0x2(2)
        Value=0x1(1),(default 768-bit MODP group)
        Attr : 11(Life Type    ), len=0x2(2)
        Value=0x1(1),(seconds)
        Attr : 12(Life Duration), len=0x2(2)
        Value=0x7080(28800)
```

ส่วนหัวข้อความ IKE ของบันทึกการทำงาน Parse Payload แสดงประเภท การแลกเปลี่ยน (Main Mode หรือ Aggressive Mode) ความยาวของทั้งข้อความ identifier ข้อความ และอื่นๆ

Certificate Request Payload ร้องขอใบรับรองจาก ผู้ตอบกลับ ผู้ตอบกลับส่งใบรับรองในข้อความแยกต่างหาก ตัวอย่างต่อไปนี้แสดง Certificate Payload และSignature Payload ที่ถูกส่งไปยังเพียร์เป็นส่วนหนึ่งของการเจรจา SA ข้อมูล ใบรับรองและข้อมูลลายเซ็นถูกพิมพ์ในรูปฐานสิบหก

```
ISAKMP_MSG_HEADER
        Icookie : 0x9e539a6fd4540990, Rcookie : 0xc7e0a8d937a8f13e
        Next Payload : 6(Certificate), Maj Ver : 1, Min Ver : 0
        Xchg Type : 4 (Aggressive), Flag= 0, Encr : No,COMMIT : No
        Msg ID  : 0x00000000
        len     : 0x2cd(717)
Certificate Payload:
```

```
        Next Payload : 9(Signature), Payload len : 0x22d(557)
        Certificate Encoding Type: 4(X.509 Certificate - Signature)
        Certificate: (len 0x227(551) in bytes
        82 02 24 30 82 01 8d a0 03 02 01 02 02 05 05 8e
        fb 3e ce 30 0d 06 09 2a 86 48 86 f7 0d 01 01 04
        05 00 30 5c 31 0b 30 09 06 03 55 04 06 13 02 46
        49 31 24 30 22 06 03 55 04 0a 13 1b 53 53 48 20
        43 6f 6d 6d 75 6e 69 63 61 74 69 6f 6e 73 20 53
        65 63 75 72 69 74 79 31 11 30 0f 06 03 55 04 0b
        13 08 57 65 62 20 74 65 73 74 31 14 30 12 06 03
        55 04 03 13 0b 54 65 73 74 20 52 53 41 20 43 41
        30 1e 17 0d 39 39 30 39 32 31 30 30 30 30 30 30
        5a 17 0d 39 39 31 30 32 31 32 33 35 39 35 39 5a
        30 3f 31 0b 30 09 06 03 55 04 06 13 02 55 53 31
        10 30 0e 06 03 55 04 0a 13 07 49 42 4d 2f 41 49
        58 31 1e 30 1c 06 03 55 04 03 13 15 62 61 72 6e
        65 79 2e 61 75 73 74 69 6e 2e 69 62 6d 2e 63 6f
        6d 30 81 9f 30 0d 06 09 2a 86 48 86 f7 0d 01 01
        01 05 00 03 81 8d 00 30 81 89 02 81 81 00 b2 ef
        48 16 86 04 7e ed ba 4c 14 d7 83 cb 18 40 0a 3f
        55 e9 ad 8f 0f be c5 b6 6d 19 ec de 9b f5 01 a6
        b9 dd 64 52 34 ad 3d cd 0d 8e 82 6a 85 a3 a8 1c
        37 e4 00 59 ce aa 62 24 b5 a2 ea 8d 82 a3 0c 6f
        b4 07 ad 8a 02 3b 19 92 51 88 fb 2c 44 29 da 72
        41 ef 35 72 79 d3 e9 67 02 b2 71 fa 1b 78 13 be
        f3 05 6d 10 4a c7 d5 fc fe f4 c0 b8 b8 fb 23 70
        a6 4e 16 5f d4 b1 9e 21 18 82 64 6d 17 3b 02 03
        01 00 01 a3 0f 30 0d 30 0b 06 03 55 1d 0f 04 04
        03 02 07 80 30 0d 06 09 2a 86 48 86 f7 0d 01 01
        04 05 00 03 81 81 00 75 a4 ee 9c 3a 18 f2 de 5d
        67 d4 1c e4 04 b4 e5 b8 5e 9f 56 e4 ea f0 76 4a
        d0 e4 ee 20 42 3f 20 19 d4 25 57 25 70 0a ea 41
        81 3b 0b 50 79 b5 fd 1e b6 0f bc 2f 3f 73 7d dd
        90 d4 08 17 85 d6 da e7 c5 a4 d6 9a 2e 8a e8 51
        7e 59 68 21 55 4c 96 4d 5a 70 7a 50 c1 68 b0 cf
        5f 1f 85 d0 12 a4 c2 d3 97 bf a5 42 59 37 be fe
        9e 75 23 84 19 14 28 ae c4 c0 63 22 89 47 b1 b6
        f4 c7 5d 79 9d ca d0
Signature Payload:
        Next Payload : 0(NONE), Payload len : 0x84(132)

        Signature: len 0x80(128) in bytes
        9d 1b 0d 90 be aa dc 43 95 ba 65 09 b9 00 6d 67
        b4 ca a2 85 0f 15 9e 3e 8d 5f e1 f0 43 98 69 d8
        5c b6 9c e2 a5 64 f4 ef 0b 31 c3 cb 48 7c d8 30
        e3 a2 87 f4 7c 9d 20 49 b2 39 00 fa 8e bf d9 b0
        7d b4 8c 4e 19 3a b8 70 90 88 2c cf 89 69 5d 07
        f0 5a 81 58 2e 15 40 37 b7 c8 d6 8c 5c e2 50 c3
        4d 19 7e e0 e7 c7 c2 93 42 89 46 6b 5f f8 8b 7d
        5b cb 07 ea 36 e5 82 9d 70 79 9a fe bd 6c 86 36
```

ปัญหาใบรับรองดิจิตัลและโหมดลายเซ็น:

ต่อไปนี้คือโซลูชันสำหรับปัญหาใบรับรองดิจิทัลและโหมดลายเซ็นที่เป็นไปได้ที่คุณอาจประสบ:

Error	ปัญหาและโซลูชันที่เป็นไปได้
ข้อผิดพลาด: **cpsd** (Certificate Proxy Server daemon) ไม่เริ่มทำงาน มีรายการที่คล้ายกับข้อความต่อไปนี้ปรากฏในล็อกไฟล์: `Sep 21 6:02:00 ripple CPS[19950]: Init():Lo adCaCerts() failed, rc =-12`	ปัญหา: ฐานข้อมูลใบรับรองไม่ถูกเปิด หรือไม่พบ วิธี แก้ไข: ทำให้แน่ใจว่าฐานข้อมูลใบรับรอง Key Manager แสดงอยู่ใน `/etc/security` ไฟล์ต่อไปนี้รวมกันเป็น ฐานข้อมูล: `ikekey.crl`, `ikekey.kdb`, `ikekey.rdb`, `ikekey.sth` ถ้า มีไฟล์ `ikekey.sth` เท่านั้นที่หายไป อ็อพชัน `stash password` จะไม่ถูกเลือกเมื่อสร้างฐานข้อมูล Key Manager รหัสผ่านต้องถูกเก็บไว้เพื่อให้สามารถใช้ใบรับรอง ดิจิตัลกับ IP Security (ดูที่ Creating a Key Database เพื่อดูข้อมูลเพิ่มเติม)

ข้อผิดพลาด: Key Manager แสดงข้อผิดพลาดต่อไปนี้เมื่อได้รับ ใบรับรอง:
 	
`Invalid Base64-encoded data was found`
	ปัญหา: พบข้อมูล Superfluous ในไฟล์ใบรับรองหรือข้อมูล สูญหาย หรือเสียหาย
	
	วิธีแก้ไข: 'DER' Encoded Certificate ควรมีอยู่ภายในสตริงต่อไปนี้ (แสดงด้านล่าง) ไม่ควร มีอักขระอื่นๆ นำหน้า หรือต่อท้ายนอกเหนือจากสตริง BEGIN และ END CERTIFICATE
	
	`-----BEGIN CERTIFICATE-----`
	`MIICMTCCAZqgAwIBAgIFFKZtANowDQYJKoZIhvcNAQEFBQAwXDELMAkGA1UEBhMC`
	`RkkxJDAiBgNVBAoTG1NTSCBDb21tdW5pY2F0aW9ucyBTZWN1cml0eTERMA8GA1UE`
	`CxMIV2ViIHRlc3QxFDASBgNVBAMTC1Rlc3QgUlNBIENBMB4XDTk5MDkyMTAwMDAw`
	`MFoXDTk5MTAyMTIzNTk1OVowOzELMAkGA1UEBhMCVVMxDDAKBgNVBAoTA0lCTTEe`
	`MBwGA1UEAxMVcmlwcGxlLmF1c3Rpbi5pYm0uY29tMIGfMA0GCSqGSIb3DQEBAQUA`
	`A4GNADCBiQKBgQC5EZqo6n7tZrpAL6X4L7mf4yXQSm+m/NsJLhp6afbFpPvXgYWC`
	`wq4pvOtvxgum+FHrEOgysNjbKkE4Y6ixC9PGGAKHnhM3vrmvFjnl1G6KtyEz58Lz`
	`BWW39QS6NJ1LqqP1nT+y3+Xzvfv8Eonqzno8mglCWMXO9SguLmWoU1PcZQIDAQAB`
	`oyAwHjALBgNVHQ8EBAMCBaAwDwYDVRORBAgwBocECQNhhzANBgkqhkiG9w0BAQUF`
	`AOBgQA6bgp4Zay34/fyAlyCkNNAYJRrN3Vc4NHN7IGjUziN6jK5UyB5zL37FERW`
	`hT9ArPLzK7yEZs+MDNvBObosyGWEDYPZr7EZHhYcoBP4/cdOV5rBFmA8Y2gUthPi`
	`Ioxpi4+KZGHYyLqTrm+8Is/DVJaQmCGRPynHK35xjT6WuQtiYg==`
	`-----END CERTIFICATE-----`
	
	อ็อพชันต่อไปนี้สามารถช่วยคุณ วินิจฉัยและแก้ไขปัญหานี้
	• ถ้าข้อมูลสูญหายหรือเสียหาย ให้สร้างใบรับรองใหม่
	• ใช้ ASN.1 parser (มีอยู่บนอินเตอร์เน็ตเวิล์ดไวด์เว็บ) เพื่อตรวจสอบว่าใบรับรองถูกต้อง โดยการแยกวิเคราะห์ใบรับรอง ได้สำเร็จ

ข้อผิดพลาด: Key Manager แสดงข้อผิดพลาดต่อไปนี้เมื่อได้รับ ใบรับรองส่วนบุคคล: `No request key was found for the certificate`	ปัญหา: Personal Certificate Request ไม่มีอยู่สำหรับใบรับรองส่วนบุคคล ที่ได้รับ วิธีแก้ไข: สร้าง Personal Certificate Request อีกครั้งและร้องขอใบรับรองใหม่
ข้อผิดพลาด: IKE negotiation ล้มเหลว และรายการคล้ายกับ ต่อไปนี้ปรากฏในล็อกไฟล์: `inet_cert_service:: channelOpen(): clientInitIPC():error,rc =2 (No such file or directory)`	ปัญหา: **cpsd** ไม่ ได้ทำงานอยู่ หรือหยุดทำงาน วิธีแก้ไข: เริ่มทำงาน IP Security, ซึ่งเริ่มต้น daemons ที่เหมาะสม
ข้อผิดพลาด: IKE negotiation ล้มเหลว และรายการคล้ายกับ ต่อไปนี้ปรากฏในล็อกไฟล์: `CertRepo::GetCertObj: DN Does Not Match: ("/C=US/O=IBM/CN=ripple. austin.ibm.com")`	ปัญหา: X.500 Distinguished Name (DN) ที่ป้อนขณะกำหนด IKE tunnel ไม่ตรงกับ X.500 DN ในใบรับรองส่วนบุคคล วิธีแก้ไข: เปลี่ยนนิยาม IKE tunnel เพื่อจับคู่ชื่อจำเพาะในใบรับรอง

สิ่งอำนวยความสะดวกการติดตาม:

การติดตามเป็นสิ่งอำนวยความสะดวกการดีบักสำหรับการติดตาม เหตุการณ์เคอร์เนล การติดตามสามารถใช้เพื่อรับข้อมูลที่เจาะจงเพิ่มมากขึ้นเกี่ยวกับเหตุการณ์หรือ ข้อผิดพลาดที่เกิดขึ้นในตัวกรองเคอร์เนลและ tunnel code

สิ่งอำนวยความสะดวกการติดตาม SMIT IP Security มีอยู่ในเมนู Advanced IP Security Configuration ข้อมูลที่บันทึกโดยระบบการติดตามนี้มีข้อมูลเกี่ยวกับ Error, Filter, Filter Information, Tunnel, Tunnel Information, Capsulation/Decapsulation, ข้อมูล Capsulation, Crypto และข้อมูล Crypto โดยการออกแบบ hook การติดตามข้อผิดพลาดช่วยให้มีข้อมูลที่สำคัญยิ่ง hook การติดตามข้อมูลสามาถสร้างข้อมูลสำคัญและอาจ มีผลต่อผลการทำงานของระบบ การติดตามนี้แสดงแนวทางแก้ไขปัญหาและ จำเป็นต้องทำเมื่ออธิบายปัญหาแก่ช่างเทคนิคบริการ

เพื่อเปิดใช้การติดตาม กำหนดค่าอุปกรณ์ IPSec และตั้งค่า ระดับการติดตามของแต่ละคอมโพเนนต์ย่อย IPSec จนถึงระดับการติดตาม 7 เพื่อสร้าง ข้อมูลการติดตาม kernel ที่เป็นประโยชน์ ถ้าไม่ได้กำหนดค่าอุปกรณ์ IPSec คำสั่งควบคุมการติดตามคอมโพเนนต์จะไม่แสดงในรายการที่เกี่ยวข้องกับ IPSec entries. เมื่อต้องการเริ่มต้นติดตาม IPSec ใช้พาธด่วน SMIT **smit ips4_start** (สำหรับ IP Version 4) หรือ **smit ips6_start** (สำหรับ IP Version 6)

หมายเหตุ: ถ้าการติดตามคอมโพเนนต์ IPSec ไม่ได้ตั้งค่าอย่างถูกต้อง การติดตามที่บันทึกจะว่างเปล่า

เพื่อบันทึกข้อมูลการติดตาม kernel ให้ทำตามขั้นตอนต่อไปนี้:

1. คิวรีคอมโพเนนต์ทั้งหมดเพื่อดูการตั้งค่าระดับการติดตามปัจจุบัน:

   ```
   # ctctrl -q
   ```

2. ตรวจสอบคอมโพเนนต์และคอมโพเนนต์ย่อยของ IPSec คอมโพเนนต์ จะปรากฏขึ้นครั้งแรกดังต่อไปนี้ด้วยระดับการติดตามดีฟอลต์ 3 เพื่อดู ระดับการติดตามครั้งแรกของคอมโพเนนต์ ป้อน:

   ```
   # ctctrl -q -c ipsec -r
   ```

Component Name	มีชื่อย่อ	Memory Trace/Level	System Track/Level	Buffer Size/Allocated
ipsec	NO	ON/3	ON/3	40960/YES
.capsulate	NO	ON/3	ON/3	10240/YES
.filter	NO	ON/3	ON/3	10240/YES
.tunnel	NO	ON/3	ON/3	10240/YES

3. เพิ่มระดับการติดตามของ IPSec และคอมโพเนนต์ย่อยไปที่ 7 เพื่อ รองรับการติดตาม kernel ป้อน:

   ```
   # ctctrl systracelevel=7 -c ipsec -r
   ```

4. คิวรีเพื่อยืนยันว่าระดับการติดตามสำหรับ IPSec และคอมโพเนนต์ย่อย เปลี่ยนแปลง ป้อน:

   ```
   # ctctrl -q -c ipsec -r
   ```

Component Name	มีชื่อย่อ	Memory Trace/Level	System Track/Level	Buffer Size/Allocated
ipsec	NO	ON/3	ON/7	40960/YES
.capsulate	NO	ON/3	ON/7	10240/YES
.filter	NO	ON/3	ON/7	10240/YES
.tunnel	NO	ON/3	ON/7	10240/YES

ในการเาถึงสิ่งอำนวยความสะดวกการติดตาม ใช้พาธด่วน SMIT **smit ips4_tracing** (สำหรับ IP Version 4) หรือ **smit ips6_tracing** (สำหรับ IP Version 6) การติดตาม Kernel ที่ทำผ่าน **smit ips4_tracing, smit ips6_tracing** หรือผ่านระบบติดตามบรรทัดคำสั่ง สร้างข้อมูลติดตาม IPSec ที่ถูกต้อง

คำสั่ง ipsecstat:

คุณสามารถใช้คำสั่ง **ipsecstat** เพื่อแสดงรายการสถานะของอุปกรณ์ IP Security อัลกอริทึมการเข้ารหัส IP Security และสถิติของแพ็กเก็ต IP Security

การออกคำสั่ง **ipsecstat** จะ สร้างรายงานตัวอย่างต่อไปนี้ ซึ่งแสดงว่าอุปกรณ์ IP Security อยู่ในสถานะพร้อมใช้ มีอัลกอริทึมการพิสูจน์ตัวตนสามอัลกอริทึม ถูกติดตั้ง อัลกอริทึมการเข้ารหัสสามอัลกอริทึมถูกติดตั้ง และ มีรายการขณะปัจจุบันสำหรับกิจกรรมของแพ็กเก็ต ข้อมูลนี้อาจ เป็นประโยชน์ต่อคุณในการใช้พิจารณามีปัญหาที่ใดถ้าคุณกำลัง แก้ปัญหาการรับส่งข้อมูล IP Security ของคุณ

```
IP Security Devices:
ipsec_v4 Available
ipsec_v6 Available

Authentication Algorithm:
HMAC_MD5 -- Hashed MAC MD5 Authentication Module
HMAC_SHA -- Hashed MAC SHA Hash Authentication Module
KEYED_MD5 -- Keyed MD5 Hash Authentication Module

Encryption Algorithm:
CDMF -- CDMF Encryption Module
DES_CBC_4 -- DES CBC 4 Encryption Module
DES_CBC_8 -- DES CBC 8 Encryption Module
3DES_CBC -- Triple DES CBC Encryption Module

IP Security Statistics -
Total incoming packets:  1106
Incoming AH packets:326
Incoming ESP packets:  326
Srcrte packets allowed:  0
Total outgoing packets:844
Outgoing AH packets:527
Outgoing ESP packets:  527
Total incoming packets dropped:  12
  Filter denies on input:  12
  AH did not compute: 0
  ESP did not compute:0
  AH replay violation:0
  ESP replay violation:  0
Total outgoing packets dropped:0
```

```
  Filter denies on input:0
Tunnel cache entries added: 7
Tunnel cache entries expired:  0
Tunnel cache entries deleted:  6
```

หมายเหตุ: ไม่จำเป็นต้องใช้ CDMF เนื่องจาก DES พร้อมใช้งานได้ทั่วโลก ตั้งค่า tunnels ใดๆ ที่ใช้ CDMF ใหม่ให้ใช้ DES หรือ Triple DES

การอ้างอิงการรักษาความปลอดภัย IP

มีคำสั่งและวิธีการสำหรับการรักษาความปลอดภัย IP คุณยังสามารถโอนย้าย IKE tunnels ตัวกรอง และคีย์ที่แบ่งใช้ล่วงหน้า

รายการคำสั่ง:

ตารางต่อไปนี้มีรายการคำสั่ง

Command	วัตถุประสงค์
ike cmd=activate	เริ่มต้นการเจรจา Internet Key Exchange (IKE)
ike cmd=remove	ปิดการทำงาน IKE tunnels
ike cmd=list	แสดงรายการ IKE tunnels
ikedb	จัดเตรียมอินเตอร์เฟสไปยังฐานข้อมูล IKE tunnel
gentun	สร้างนิยาม tunnel
mktun	เรียกทำงานนิยาม tunnel
chtun	เปลี่ยนนิยาม tunnel
rmtun	ลบนิยาม tunnel ออก
lstun	แสดงรายการนิยาม tunnel
exptun	เอ็กซ์พอร์ตนิยาม tunnel
imptun	อิมพอร์ตนิยาม tunnel
genfilt	สร้างนิยามตัวกรอง
mkfilt	เรียกทำงานนิยามตัวกรอง
mvfilt	ย้ายกฎตัวกรอง
chfilt	เปลี่ยนนิยามตัวกรอง
rmfilt	ลบนิยามตัวกรองออก
lsfilt	แสดงรายการนิยามตัวกรอง
expfilt	เอ็กซ์พอร์ตนิยามตัวกรอง
impfilt	อิมพอร์ตนิยามตัวกรอง
ipsec_convert	แสดงรายการสถานะ IP security
ipsecstat	แสดงรายการสถานะ IP security
ipsectrcbuf	แสดงรายการเนื้อหาของบัฟเฟอร์การติดตาม IP security
unloadipsec	ยกเลิกการโหลดโมดูลที่เข้ารหัส

รายการเมธอด:

ต่อไปนี้จัดให้มีรายการเมธอด

defipsec

กำหนด instance ของ IP Security สำหรับ IP Version 4 หรือ IP Version 6

cfgipsec

ตั้งค่าและโหลด **ipsec_v4** หรือ **ipsec_v6**

ucfgipsec

ยกเลิกการตั้งค่า **ipsec_v4** หรือ **ipsec_v6**

การโอนย้าย IP security:

คุณสามารถโอนย้าย IKE tunnels, ตัวกรอง และคีย์ที่แบ่งใช้ก่อนของคุณจากเวอร์ชันก่อนหน้านี้ของระบบปฏิบัติการ AIX

การโอนย้าย IKE tunnels:

เมื่อต้องการโอนย้าย tunnels ของคุณ, ให้ทำตาม ขั้นตอนต่อไปนี้:

1. รันสคริปต์ `bos.net.ipsec.keymgt.pre_rm.sh` เมื่อคุณรันสคริปต์นี้ ไฟล์ต่อไปนี้จะถูกสร้างขึ้นในไดเร็กทอรี `/tmp`:
   a. `p2proposal.bos.net.ipsec.keymgt`
   b. `p1proposal.bos.net.ipsec.keymgt`
   c. `p1policy.bos.net.ipsec.keymgt`
   d. `p2policy.bos.net.ipsec.keymgt`
   e. `p1tunnel.bos.net.ipsec.keymgt`
   f. `p2tunnel.bos.net.ipsec.keymgt`

   ข้อควรสนใจ: รันสคริปต์นี้เพียงครั้งเดียวเท่านั้น ถ้าคุณอัพเดต ฐานข้อมูลและรันสคริปต์อีกครั้ง คุณจะสูญเสียไฟล์ทั้งหมด โดยที่คุณ ไม่สามารถเรียกคืนมาได้ อ่านสคริปต์ใน "สคริปต์ bos.net.ipsec.keymgt.pre_rm.sh" ในหน้า 300 ก่อนที่คุณจะโอนย้าย tunnels ของคุณ
2. บันทึกไฟล์ที่สร้างโดยสคริปต์และไฟล์ `/tmp/lpplevel` ไปยังสื่อบันทึกภายนอก เช่นซีดีหรือฟล็อปปีดิสก์

การโอนย้ายคีย์ที่แบ่งใช้ล่วงหน้า:

ดำเนินขั้นตอนต่อไปนี้เพื่ออัพเดตรูปแบบคีย์ที่แบ่งใช้ล่วงหน้า

ฐานข้อมูลคีย์ที่แบ่งใช้ล่วงหน้า IKE tunnel ถูกทำให้เสียหายเช่นกัน ระหว่างการโอนย้ายระบบ เมื่อต้องการอัพเดตรูปแบบคีย์ที่แบ่งใช้ก่อน, ให้ทำตามขั้นตอนต่อไปนี้บนระบบที่ถูกโอนย้าย:

1. บันทึกเอาต์พุตของคำสั่ง **ikedb -g** โดยการรันคำสั่งต่อไปนี้:
   ```
   ikedb -g > out.keys
   ```
2. แก้ไขไฟล์ `out.keys` เพื่อแทน FORMAT=ASCII ด้วย FORMAT=HEX สำหรับ รูปแบบคีย์ที่แบ่งใช้ล่วงหน้า
3. อินพุตไฟล์ XML โดยการรันคำสั่ง ต่อไปนี้:
   ```
   ikedb -pF out.keys
   ```

การโอนย้ายตัวกรอง:

ดำเนินขั้นตอนต่อไปนี้เพื่อโอนย้ายตัวกรอง

1. เอ็กซ์พอร์ตไฟล์กฎตัวกรองไปยังไดเร็กทอรี `/tmp` โดยใช้ SMIT โดยการดำเนินขั้นตอนต่อไปนี้:
   a. รันคำสั่ง **smitty ipsec4**
   b. เลือก `Advanced IP Security Configuration—>Configure IP Security Filter Rules—>Export IP Security filter rules`
   c. ป้อน `/tmp` สำหรับชื่อไดเร็กทอรี
   d. ภายใต้อ็อพชัน Filter Rules กด F4 และ เลือก **all** จากรายการ

e. กด enter เพื่อบันทึกกฎตัวกรองในไฟล์ /tmp/ipsec_fltr_rule.exp บนสื่อบันทึกภายนอก

ทำกระบวนการนี้ให้เสร็จสิ้นสำหรับระบบทั้งหมด ที่คุณกำลังโอนย้ายจากเวอร์ชันก่อนหน้านี้ของระบบปฏิบัติการ AIX

2. ทำสำเนาไฟล์ tunnel ทั้งหกไฟล์ที่สร้างโดยสคริปต์ไฟล์ /tmp/lpplevel และไฟล์ /tmp/ipsec_fltr_rule.exp ไปยังไดเร็กทอรี /tmp บนระบบที่โอนย้าย
3. รันสคริปต์ bos.net.ipsec.keymgt.post_i.sh เพื่อกระจายการตั้งค่าไปในฐานข้อมูลอีกครั้ง
4. รันคำสั่ง **ikedb -g** เพื่อตรวจสอบว่า tunnels อยู่ในฐานข้อมูล

หมายเหตุ: ถ้าคุณ ไม่เห็นข้อมูล tunnel ในฐานข้อมูล ให้รันสคริปต์อีกครั้ง แต่เปลี่ยนชื่อไฟล์ *.loaded ทั้งหมดในไดเร็กทอรี /tmp เป็นชื่อต้นฉบับ

บนระบบที่ถูกโอนย้าย, ฐานข้อมูลตัวกรองถูกทำให้ล้มเหลวหลังจากการโอนย้ายระบบ ถ้าคุณรันคำสั่ง **lsfilt** บนระบบที่โอนย้าย คุณจะได้รับข้อผิดพลาดต่อไปนี้:

```
Cannot get ipv4 default filter rule
```

ในการ อัพเดตฐานข้อมูลตัวกรอง ดำเนินขั้นตอนต่อไปนี้:

1. แทนที่ไฟล์ ipsec_filter และไฟล์ ipsec_filter.vc ในไดเร็กทอรี /etc/security ด้วยไฟล์ที่ไม่ล้มเหลวจาก จากระบบที่โอนย้ายใหม่ ถ้าคุณไม่มี ไฟล์เหล่านี้ คุณสามารถขอได้จาก IBM Service
2. อิมพอร์ตไฟล์กฎตัวกรองไปยังไดเร็กทอรี /tmp โดยใช้ SMIT โดยการดำเนินขั้นตอนต่อไปนี้:
   a. รันคำสั่ง **smitty ipsec4**
   b. เลือก Advanced IP Security Configuration—>Configure IP Security Filter Rules—>Import IP Security filter rules
   c. ป้อน /tmp สำหรับชื่อไดเร็กทอรี
   d. ภายใต้อ็อพชัน **Filter Rules** กด **F4** และ เลือก **all** จากรายการ
   e. กด Enter เพื่อสร้างกฎตัวกรองใหม่ คุณสามารถแสดงรายการกฎตัวกรอง ทาง SMIT หรือด้วยคำสั่ง **lsfilt**

สคริปต์ bos.net.ipsec.keymgt.pre_rm.sh:

สคริปต์ bos.net.ipsec.keymgt.pre_rm.sh บันทึกเนื้อหาของฐานข้อมูล tunnel บนระบบที่รันระบบปฏิบัติการ AIX

```
#!/usr/bin/ksh
keymgt_installed=`lslpp -Lqc bos.net.ipsec.keymgt 2>/dev/null | awk -F: '{print $6}' | head -1`

if [ ! "$keymgt_installed" ]
then
  exit 0
fi


#  Copy the database to a save directory in case changes fail
if [ -d /etc/ipsec/inet/DB ]
then
  cp -R /etc/ipsec/inet/DB /etc/ipsec/inet/DB.sav || exit $?
fi

#  Remember the level you are migrating from
VRM=$(LANG=C lslpp -Lqc bos.net.ipsec.keymgt 2>/dev/null | awk -F: '{print $3}' | \
```

```
awk -F. '{print $1"."$2"."$3}')
VR=${VRM%.*}
echo $VRM > /tmp/lpplevel

IKEDB=$(which ikedb) || IKEDB=/usr/sbin/ikedb

XMLFILE=/tmp/full_ike_database.bos.net.ipsec.keymgt
PSKXMLFILE=/tmp/psk_ike_database.bos.net.ipsec.keymgt

#  See if ikedb exists.
if [ -f $IKEDB ]
then

  # If either of the ikedb calls below fails, that's OK.  Just remove the
  # resulting file (which may contain garbage) and continue.  The post_i
  # script will simply not import the file if it doesn't exist, which will
  # mean part or all of the IKE database is lost, but this is preferable
  # to exiting the script with an error code, which causes the entire
  # migration to fail.

  $IKEDB -g > $XMLFILE
  if [ $? -ne 0 ]
  then
    rm -f $XMLFILE  || exit $?
  fi

  if [[ $VR = "5.1" ]]; then
    # This is a special case.  The 5.1 version of ikedb is the only
    # one that does not include preshared keys in the full database
    # output.  So we have to retrieve those separately.
    $IKEDB -g -t IKEPresharedKey > $PSKXMLFILE
    if [ $? -ne 0 ]
    then
     rm -f $PSKXMLFILE  || exit $?
    fi
  fi

#  Make sure ikegui command is installed
elif [ -f /usr/sbin/ikegui ]
then

  #  Get database information and save to /tmp
  /usr/sbin/ikegui 0 1 0 0 > /tmp/p1proposal.bos.net.ipsec.keymgt 2>/dev/null
  RC=$?
  if [[ $RC -ne 0 ]]
  then
    rm -f /tmp/p1proposal.bos.net.ipsec.keymgt || exit $?
  fi

  /usr/sbin/ikegui 0 1 1 0 > /tmp/p1policy.bos.net.ipsec.keymgt 2>/dev/null
  RC=$?
  if [[ $RC -ne 0 ]]
  then
    rm -f /tmp/p1policy.bos.net.ipsec.keymgt || exit $?
  fi
```

```
  /usr/sbin/ikegui 0 2 0 0 > /tmp/p2proposal.bos.net.ipsec.keymgt 2>/dev/null
  RC=$?
  if [[ $RC -ne 0 ]]
  then
    rm -f /tmp/p2proposal.bos.net.ipsec.keymgt || exit $?
  fi

  /usr/sbin/ikegui 0 2 1 0 > /tmp/p2policy.bos.net.ipsec.keymgt 2>/dev/null
  RC=$?
  if [[ $RC -ne 0 ]]
  then
    rm -f /tmp/p2policy.bos.net.ipsec.keymgt || exit $?
  fi

  /usr/sbin/ikegui 0 1 2 0 > /tmp/p1tunnel.bos.net.ipsec.keymgt 2>/dev/null
  RC=$?
  if [[ $RC -ne 0 ]]
  then
    rm -f /tmp/p1tunnel.bos.net.ipsec.keymgt || exit $?
  fi

  /usr/sbin/ikegui 0 2 2 0 > /tmp/p2tunnel.bos.net.ipsec.keymgt 2>/dev/null
  RC=$?
  if [[ $RC -ne 0 ]]
  then
    rm -f /tmp/p2tunnel.bos.net.ipsec.keymgt || exit $?
  fi

fi
```

สคริปต์ bos.net.ipsec.keymgt.post_i.sh:

สคริปต์ bos.net.ipsec.keymgt.post_i.sh โหลดเนื้อหาของฐานข้อมูล tunnel บนระบบที่ถูกโอนย้าย เพื่อรันระบบปฏิบัติการ AIX

```
#!/usr/bin/ksh

function PrintDot {
    echo "echo \c"
    echo "\".\c"
    echo "\\\c\c"
    echo "\"\c"
    echo
}

function P1PropRestore {
    while :
    do
        read NAME
        read MODE
        if [[ $? = 0 ]]; then
            echo "ikegui 1 1 0 $NAME $MODE \c"
            MORE=1
            while [[ $MORE = 1 ]];
```

```
            do
                read AUTH
                read HASH
                read ENCRYPT
                read GROUP
                read TIME
                read SIZE
                read MORE
                echo "$AUTH $HASH $ENCRYPT $GROUP $TIME $SIZE $MORE \c"
            done
            echo " > /dev/null 2>&1"
            PrintDot
        else
            return 0
        fi
    done
}

function P2PropRestore {
    while :
    do
        read NAME
        FIRST=yes
        MORE=1
        while [[ $MORE = 1 ]];
        do
            read PROT
            if [[ $? = 0 ]]; then
                read AH_AUTH
                read ESP_ENCR
                read ESP_AUTH
                read ENCAP
                read TIME
                read SIZE
                read MORE
                if [[ $FIRST = "yes" ]]; then
                    echo "ikegui 1 2 0 $NAME $MODE \c"
                fi
                echo "$PROT $AH_AUTH $ESP_ENCR $ESP_AUTH $ENCAP $TIME $SIZE $MORE \c"
                FIRST=no
            else
                return 0
            fi
        done
        echo " > /dev/null 2>&1"
        PrintDot
    done
}

function P1PolRestore {
    while :
    do
        read NAME
        read ROLE
        if [[ $? = 0 ]]; then
```

```
            read TIME
            read SIZE
            read OVERLAP
            read TTIME
            read TSIZE
            read MIN
            read MAX
            read PROPOSAL
            echo "ikegui 1 1 1 $NAME $ROLE $OVERLAP $TTIME $TSIZE $MIN $MAX 1 0 0 $PROPOSAL > \
/dev/null 2>&1"
            PrintDot
        else
            return 0
        fi
    done
}

function P2PolRestore {
    while :
    do
        read NAME
        read ROLE
        if [[ $? = 0 ]]; then
            read IPFS
            read RPFS
            read TIME
            read SIZE
            read OVERLAP
            read TTIME
            read TSIZE
            read MIN
            read MAX
            echo "ikegui 1 2 1 $NAME $ROLE $IPFS $RPFS $OVERLAP $TTIME $TSIZE $MIN $MAX 1 0 0 \c"
            MORE=1
            while [[ $MORE = 1 ]];
            do
                read PROPOSAL
                read MORE
                echo "$PROPOSAL $MORE \c"
                FIRST=no
            done
        else
            return 0
        fi
        echo " > /dev/null 2>&1"
        PrintDot
    done
}

function P1TunRestore {
    while :
    do
        read TUNID
        read NAME
        if [[ $? = 0 ]]; then
```

```
            read LID_TYPE
            read LID
            if [[ $LPPLEVEL = "4.3.3" ]]; then
                read LIP
            fi
            read RID_TYPE
            read RID
            read RIP
            read POLICY
            read KEY
            read AUTOSTART
            echo "ikegui 1 1 2 0 $NAME $LID_TYPE \"$LID\" $LIP $RID_TYPE \"$RID\" \
$RIP $POLICY $KEY $AUTOSTART > /dev/null 2>&1"
            PrintDot
        else
            return 0
        fi
    done
}

function P2TunRestore {
    while :
    do
        read TUNID
        read NAME
        if [[ $? = 0 ]]; then
            read P1TUN
            read LTYPE
            read LID
            read LMASK
            read LPROT
            read LPORT
            read RTYPE
            read RID
            read RMASK
            read RPROT
            read RPORT
            read POLICY
            read AUTOSTART
            echo "ikegui 1 2 2 0 $NAME $P1TUN $LTYPE $LID $LMASK $LPROT $LPORT $RTYPE
              \$RID $RMASK $RPROT $RPORT $POLICY $AUTOSTART > /dev/null 2>&1"
            PrintDot
        else
            return 0
        fi
    done
}

function allRestoreWithIkedb {

    ERRORS=/tmp/ikedb_msgs.bos.net.ipsec.keymgt
    echo > $ERRORS
    $IKEDB -p $XMLFILE 2>> $ERRORS
    if [ -f $PSKXMLFILE ]
    then
```

```
        $IKEDB -p $PSKXMLFILE 2>> $ERRORS
    fi

}

P1PROPFILE=/tmp/p1proposal.bos.net.ipsec.keymgt
P2PROPFILE=/tmp/p2proposal.bos.net.ipsec.keymgt
P1POLFILE=/tmp/p1policy.bos.net.ipsec.keymgt
P2POLFILE=/tmp/p2policy.bos.net.ipsec.keymgt
P1TUNFILE=/tmp/p1tunnel.bos.net.ipsec.keymgt
P2TUNFILE=/tmp/p2tunnel.bos.net.ipsec.keymgt
XMLFILE=/tmp/full_ike_database.bos.net.ipsec.keymgt
PSKXMLFILE=/tmp/psk_ike_database.bos.net.ipsec.keymgt
CMD_FILE=/tmp/commands
IKEDB=$(which ikedb) || IKEDB=/usr/sbin/ikedb

echo "building ISAKMP database \n"
$IKEDB -x || exit $?

if [ -f $XMLFILE ]; then
    echo "\nRestoring database entries\c"
    allRestoreWithIkedb
    echo "\ndone\n"

elif [ -f /tmp/*.bos.net.ipsec.keymgt ]; then
    echo "\nRestoring database entries\c"

    LPPLEVEL=`cat /tmp/lpplevel`

    echo > $CMD_FILE
    touch $P1PROPFILE; P1PropRestore < $P1PROPFILE >> $CMD_FILE
    touch $P2PROPFILE; P2PropRestore < $P2PROPFILE >> $CMD_FILE
    touch $P1POLFILE; P1PolRestore < $P1POLFILE >> $CMD_FILE
    touch $P2POLFILE; P2PolRestore < $P2POLFILE >> $CMD_FILE
    touch $P1TUNFILE; P1TunRestore < $P1TUNFILE >> $CMD_FILE
    touch $P2TUNFILE; P2TunRestore < $P2TUNFILE >> $CMD_FILE

    mv $P1PROPFILE ${P1PROPFILE}.loaded
    mv $P2PROPFILE ${P2PROPFILE}.loaded
    mv $P1POLFILE ${P1POLFILE}.loaded
    mv $P2POLFILE ${P2POLFILE}.loaded
    mv $P1TUNFILE ${P1TUNFILE}.loaded
    mv $P2TUNFILE ${P2TUNFILE}.loaded

    ksh $CMD_FILE

    echo "done\n"
fi
```

การรักษาความปลอดภัยด้วย Network File System

Network File System (NFS) คือเทคโนโลยีที่ใช้อย่างกว้างขวาง ซึ่งอนุญาตให้ข้อมูลถูกแบ่งใช้ระหว่างโฮสต์ต่างๆ บนเน็ตเวิร์ก

NFS ยังสนับสนุนการใช้การพิสูจน์ตัวตน Kerberos 5 เพิ่มเติมจาก DES การรักษาความปลอดภัยด้วย Kerberos 5 ถูกจัดให้มีภายใต้การใช้กลไกโปรโตคอล ชื่อ RPCSEC_GSS

นอกเหนือจากระบบการพิสูจน์ตัวตน UNIX มาตรฐาน แล้ว NFS ยังมีวิธีพิสูจน์ตัวตนผู้ใช้และเครื่องใน เน็ตเวิร์กด้วยวิธี message-by-message ระบบการพิสูจน์ตัวตน แบบดั้งเดิมนี้ใช้การเข้ารหัส Data Encryption Standard (DES) และวิทยาการเข้ารหัสลับ ด้วยพับลิกคีย์

NFS ยังสนับสนุนการใช้การพิสูจน์ตัวตน Kerberos 5 เพิ่มเติมจาก DES การรักษาความปลอดภัยด้วย Kerberos 5 ถูกจัดให้มีภายใต้การใช้กลไกโปรโตคอล ชื่อ RPCSEC_GSS สำหรับรายละเอียดวิธีดูแลและใช้การพิสูจน์ตัวตน Kerberos authentication กับ NFS ดูที่ *NFS Administration Guide*

คำแนะนำทั่วไปสำหรับการรักษาความปลอดภัย Network File System

มีคำแนะนำมากมายที่ช่วยคุณรักษาความปลอดภัย Network File System (NFS)

- ตรวจสอบว่าติดตั้งแพ็ตช์ซอฟต์แวร์ล่าสุดแล้ว แพ็ตช์ ที่แก้ไขปัญหาควรได้รับการพิจารณาว่ามีความสำคัญอย่างยิ่ง ซอฟต์แวร์ทั้งหมดในโครงสร้างพื้นฐานที่กำหนดให้ควรได้รับการดูแลรักษา ตัวอย่าง การติดตั้งแพ็ตช์ในระบบปฏิบัติการ แต่ไม่สามารถติดตั้งแพ็ตช์ บนเว็บเซิร์ฟเวอร์อาจทำให้ผู้โจมตีมีวิธีที่จะเข้าถึงสภาวะแวดล้อม ของคุณที่สามารถหลีกเลี่ยงได้หากเว็บเซิร์ฟเวอร์ได้รับการอัพเดต เช่นเดียวกัน เมื่อต้องการสมัครสมาชิกไปยัง IBM System p® Security Alerts สำหรับข้อมูลเกี่ยวกับข้อมูลความปลอดภัย ที่มีอยู่ล่าสุด, ให้เยี่ยมชนเว็บแอดเดรสต่อไปนี้: http://www14.software.ibm.com/webapp/set2/subscriptions/pqvcmjd
- ตั้งค่าเซิร์ฟเวอร์ NFS เพื่อเอ็กซ์พอร์ตระบบไฟล์ที่มีสิทธิพิเศษ น้อยที่สุดเท่าที่จำเป็น ถ้าผู้ใช้ต้องการอ่านจาก ระบบไฟล์เท่านั้น ผู้ใช้ไม่ควรสามารถเขียนลงระบบไฟล์ วิธีนี้ช่วยลดโอกาสเกิดความพยายามที่จะเขียนทับข้อมูลสำคัญ แก้ไขไฟล์คอนฟิกูเรชัน หรือเขียนโค้ดที่รันได้ที่อาจเป็นอันตรายลงในระบบไฟล์ที่เอ็กซ์พอร์ต ระบุสิทธิพิเศษโดยใช้ SMIT หรือโดยการแก้ไขไฟล์ `/etc/exports` โดยตรง
- ตั้งค่าเซิร์ฟเวอร์ NFS เพื่อเอ็กซ์พอร์ตระบบไฟล์อย่างชัดเจนสำหรับ ผู้ใช้ที่ควรสามารถเข้าถึงได้ การนำ NFS ไปใช้โดยส่วนใหญ่ จะอนุญาตให้คุณระบุว่าไคลเอ็นต์ NFS ใดที่สามารถเข้าถึง ระบบไฟล์ที่กำหนด วิธีนี้ช่วยลดโอกาสที่ผู้ใช้ที่ไม่ได้รับอนุญาต เข้าถึงระบบไฟล์ โดยเฉพาะอย่างยิ่งระบบ อย่ากำหนดค่าเซิร์ฟเวอร์ NFS เพื่อเอ็กซ์พอร์ตระบบไฟล์ไปยังตนเอง
- ระบบไฟล์ที่เอ็กซ์พอร์ตควรอยู่ในพาร์ติชันของตนเอง ผู้โจมตี สามารถลดความสามารถการทำงานของระบบ โดยการเขียนลงในระบบไฟล์ที่เอ็กซ์พอร์ต จนกระทั่งเต็ม ซึ่งอาจทำให้ระบบไฟล์ไม่พร้อมใช้งานสำหรับแอ็พพลิเคชัน อื่นๆ หรือผู้ใช้ที่จำเป็นต้องใช้ระบบไฟล์นั้น
- อย่าอนุญาตให้ไคลเอ็นต์ NFS เข้าถึงระบบไฟล์โดยใช้ credential ผู้ใช้เป็น root หรือ credential ผู้ใช้ที่ไม่รู้จัก การนำ NFS ไปใช้งานส่วนใหญ่ สามารถถูกตั้งค่าเพื่อแม็พการร้องขอจากผู้ใช้ที่มีสิทธิพิเศษ หรือที่ไม่รู้จัก กับผู้ใช้ที่ไม่มีสิทธิพิเศษ วิธีนี้จะป้องกันมิให้เกิดสถานการณ์ที่ผู้โจมตี พยายามเข้าถึงไฟล์และดำเนินงานกับไฟล์เสมือนเป็นผู้ใช้ ที่มีสิทธิพิเศษ
- อย่าอนุญาตให้ไคลเอ็นต์ NFS รันโปรแกรม suid และ sgid บนระบบไฟล์ ที่เอ็กซ์พอร์ต ซึ่งจะป้องกันมิให้ไคลเอ็นต์ NFS เรียกใช้งานโค้ดที่เป็นอันตราย ด้วยสิทธิพิเศษ ถ้าผู้โจมตีสามารถไฟล์ที่รันได้ มีเจ้าของเป็นเจ้าของหรือกลุ่มที่มีสิทธิพิเศษ อาจส่งผลให้เกิดอันตรายร้ายแรง ต่อเซิร์ฟเวอร์ NFS ได้ ซึ่งได้ได้โดยการระบุอ็อพชันของคำสั่ง **mknfsmnt -y**
- ใช้ Secure NFS Secure NFS ใช้การเข้ารหัส DES เพื่อพิสูจน์ตัวตน โฮสต์ที่เกี่ยวข้องในทรานแซกชัน RPC RPC คือโปรโตคอลที่ใช้โดย NFS เพื่อสื่อสารการร้องขอระหว่างโฮสต์ Secure NFS จะช่วยลดโอกาสที่ ผู้โจมตีจะลอกเลียนการร้องขอ RPC โดยการเข้ารหัสการประทับเวลา ในการร้องขอ RPC ผู้รับสามารถถอดรหัสการประทับเวลาให้สำเร็จ และยืนยันว่าการร้องขอนั้นถูกต้องถือเป็นคำยืนยันว่าการร้องขอ RPC นั้นมาจากโฮสต์ที่ไว้วางใจ
- ถ้าไม่จำเป็นต้องใช้ NFS ให้ปิดการทำงาน วิธีนี้ช่วยลดจำนวน แนวทางการโจมตีที่เป็นไปได้ที่อาจมีสำหรับผู้บุกรุก

NFS ยังสนับสนุนการใช้ชนิดการเข้ารหัส AES ด้วยการพิสูจน์ตัวตน Kerberos 5 เพิ่มเติมจาก Triple DES และ Single DES สำหรับรายละเอียดวิธี ตั้งค่า Kerberos 5 ให้ใช้ประเภทการเข้ารหัส AES ดูที่คู่มือ NFS System Management

หลักการที่เกี่ยวข้อง:

"การรักษาความปลอดภัยด้วย Network File System" ในหน้า 306

ข้อมูลที่เกี่ยวข้อง:

รายการตรวจสอบสำหรับการกำหนดค่า NFS

เริ่มทำงาน NFS daemons เมื่อเริ่มทำงานระบบ

การกำหนดค่าเซิร์ฟเวอร์ NFS

การกำหนดค่าไคลเอ็นต์ NFS

การแม็พเอกลักษณ์

การเอ็กซ์พอร์ตระบบไฟล์ NFS

การตั้งค่าเครือข่ายสำหรับ RPCSEC-GSS

การยกเลิกการเอ็กซ์พอร์ตระบบไฟล์ NFS

การเปลี่ยนแปลงระบบไฟล์ที่เอ็กซ์พอร์ต

ผู้ใช้ Root เข้าถึงระบบไฟล์ที่เอ็กซ์พอร์ต

การเมาท์ระบบไฟล์ NFS โดยชัดแจ้ง

ระบบย่อยการเมาท์อัตโนมัติ

การสร้างการเมาท์ NFS ที่กำหนดไว้ล่วงหน้า

การนำการเมาท์ NFS ที่กำหนดไว้ล่วงหน้าออก

เอ็กซ์พอร์ตไฟล์สำหรับ NFS

คำสั่ง mknfsmnt

การพิสูจน์ตัวตน Network File System

NFS ใช้อัลกอริทึม DES สำหรับวัตถุประสงค์แตกต่างกัน NFS ใช้ DES เพื่อเข้ารหัสการประทับเวลาในข้อความ remote procedure call (RPC) ที่ส่งระหว่างเซิร์ฟเวอร์ NFS และไคลเอ็นต์ การประทับเวลาที่เข้ารหัสนี้ พิสูจน์ตัวตนเครื่องเหมือนที่โทเค็นพิสูจน์ตัวตนผู้ส่ง

เนื่องจาก NFS สามารถพิสูจน์ตัวตนทุกข้อความ RPC ที่แลกเปลี่ยนระหว่าง ไคลเอ็นต์และเซิร์ฟเวอร์ NFS ึ ทำให้มีระดับทางเลือกเพิ่มเติมในการรักษาความปลอดภัย สำหรับแต่ละระบบไฟล์โดยดีฟอลต์ ระบบไฟล์ถูกเอ็กซ์พอร์ต โดยใช้การพิสูจน์ตัวตน UNIX มาตรฐาน ในการใช้ประโยชน์ของระดับการรักษาความปลอดภัยเพิ่มเติมนี้ คุณสามารถระบุ อ็อพชัน secure เมื่อคุณเอ็กซ์พอร์ตระบบไฟล์

วิทยาการเข้ารหัสลับพับลิกคีย์สำหรับ secure Network File System:

ทั้งพับลิกคีย์และคีย์ลับของผู้ใช้นั้นถูกเก็บและ จัดทำดัชนีโดย net name ในแม็พ `publickey.byname`

คีย์ลับคือ DES ที่เข้ารหัสด้วยรหัสผ่านของล็อกอินผู้ใช้ คำสั่ง **keylogin** ใช้คีย์ลับที่เข้ารหัส ถอดรหัสด้วยรหัสผ่านล็อกอิน จากนั้นส่งให้ แก่โลคัลคีย์เซิร์ฟเวอร์ที่มีความปลอดภัยเพื่อบันทึกไว้สำหรับใช้ในทรานแซกชัน RPC ในอนาคต ผู้ใช้ไม่ทราบพับลิกคีย์และคีย์ลับของตน เนื่องจากคำสั่ง **yppasswd** นอกจากการเปลี่ยนรหัสผ่านล็อกอินแล้ว ยังสร้างพับลิกคีย์และ คีย์ลับโดยอัตโนมัติ

keyserv daemon คือเซอร์วิส RPC ที่รันบน NIS แต่ละเครื่อง ภายใน NIS **keyserv** รัน รูทีนย่อยพับลิกคีย์ต่อไปนี้:

- รูทีนย่อย **key_setsecret**
- รูทีนย่อย **key_encryptsession**
- รูทีนย่อย **key_decryptsession**

รูทีนย่อย **key_setsecret** แจ้งคีย์เซิร์ฟเวอร์ให้ เก็บคีย์ลับของผู้ใช้ (SK_A) เพื่อใช้ในอนาคต ซึ่งโดยปกติถูกเรียกใช้โดยคำสั่ง **keylogin** โปรแกรม ไคลเอ็นต์เรียกใช้รูทีนย่อย **key_encryptsession** เพื่อสร้าง คีย์การสนทนาที่เข้ารหัส ซึ่งถูกส่งในทรานแซกชัน RPC แรก ที่ไปยังเซิร์ฟเวอร์ คีย์เซิร์ฟเวอร์ค้นหาพับลิกคีย์ของเซิร์ฟเวอร์และรวมเข้ากับ คีย์ลับของไคลเอ็นต์ (ตั้งค่าโดยรูทีนย่อย **key_setsecret** ก่อนหน้า) เพื่อสร้างคีย์ร่วม เซิร์ฟเวอร์ขอให้คีย์เซิร์ฟเวอร์ถอดรหัส คียีการสนทนาโดยการเรียกใช้รูทีนย่อย **key_decryptsession**

โดยนัยแล้วในการเรียกใช้รูทีนย่อยเหล่านี้คือชื่อของผู้เรียกใช้ ซึ่งต้อง ได้รับพิสูจน์ตัวตนพิสูจน์ตัวตนด้วยวิธีการบางอย่าง คีย์เซิร์ฟเวอร์ไม่สามารถใช้การพิสูจน์ตัวตน DES เพื่อทำสิ่งนี้ เนื่องจากจะทำให้เกิด deadlock คีย์เซิร์ฟเวอร์แก้ปัญหานี้โดยการเก็บคีย์ลับตาม ID ผู้ใช้ (UID) และให้สิทธิการร้องขอ กับการประมวลผล root โลคัลเท่านั้น จากนั้นการประมวลผลไคลเอ็นต์จะรัน รูทีนย่อย **setuid** ที่ผู้ใช้ root เป็นเจ้าของ ที่ทำการร้องขอในนามของไคลเอ็นต์โดยแจ้งคีย์เซิร์ฟเวอร์ให้ทราบ UID จริงของไคลเอ็นต์

ข้อกำหนดการพิสูจน์ตัวตน Network File System:

การพิสูจน์ตัวตน Secure NFS ขึ้นกับความสามารถของผู้ส่ง ในการเข้ารหัสเวลาปัจจุบัน ซึ่งผู้รับสามารถถอดรหัสและตรวจสอบกับ นาฬิกาตนเอง

กระบวนการนี้มีข้อกำหนดต่อไปนี้:

- สองเอเจนต์ต้องยอมรับเวลาปัจจุบัน
- ผู้ส่งและผู้รับต้องกำลังใช้คีย์การเข้ารหัส DES เดียวกัน

การยอมรับเวลาปัจจุบัน:

ถ้าเน็ตเวิร์กใช้การซิงโครไนซ์เวลา timed daemon จะคงในนาฬิกาของไคลเอ็นต์และเซิร์ฟเวอร์ซิงโครไนซ์กัน ถ้าไม่ ไคลเอ็นต์จะคำนวณ การประทับเวลาที่เหมาะสมตามค่านาฬิกาเซิร์ฟเวอร์

ในการทำเช่นนี้ ไคลเอ็นต์จะพิจารณาเวลาเซิร์ฟเวอร์ก่อนเริ่มเซสชัน RPC จากนั้นคำนวณเวลาที่ต่างกันระหว่างนาฬิกาของตนเองและ ของเซิร์ฟเวอร์ จากนั้นไคลเอ็นต์ปรับการประทับเวลาตามความเหมาะสม ถ้า ระหว่าง ดำเนินระหว่างของเซสชัน RPC นาฬิกาของไคลเอ็นต์และเซิร์ฟเวอร์เกิดการไม่ซิงโครไนซ์ ในจุดที่เซิร์ฟเวอร์กำลังปฏิเสธการร้องขอของไคลเอ็นต์ ไคลเอ็นต์จะพิจารณาเวลาเซิร์ฟเวอร์อีกครั้ง

การใช้คีย์ DES เดียวกัน:

ไคลเอ็นต์และเซิร์ฟเวอร์คำนวณคีย์การเข้ารหัส DES เดียวกันโดยใช้วิทยาการเข้ารหัสลับพับลิกคีย์

สำหรับไคลเอ็นต์ A และเซิร์ฟเวอร์ B ใดๆ คีย์ที่ชื่อ *คีย์ร่วม* สามารถ พิจารณาได้จาก A และ B เท่านั้น ที่คีย์นี้เป็น ไคลเอ็นต์สืบทอดคีย์ร่วม โดยการคำนวณสูตรต่อไปนี้:

$$K_{AB} = PK_B^{SK}A$$

โดยที่ *K* คือ คีย์ร่วม *PK* คือพับลิกคีย์และ *SK* คือคีย์ลับ และแต่ละคีย์เหล่านี้คือจำนวน 128 บิต เซิร์ฟเวอร์สืบทอดคีย์ร่วมเดียวกันโดยการคำนวณสูตรต่อไปนี้:

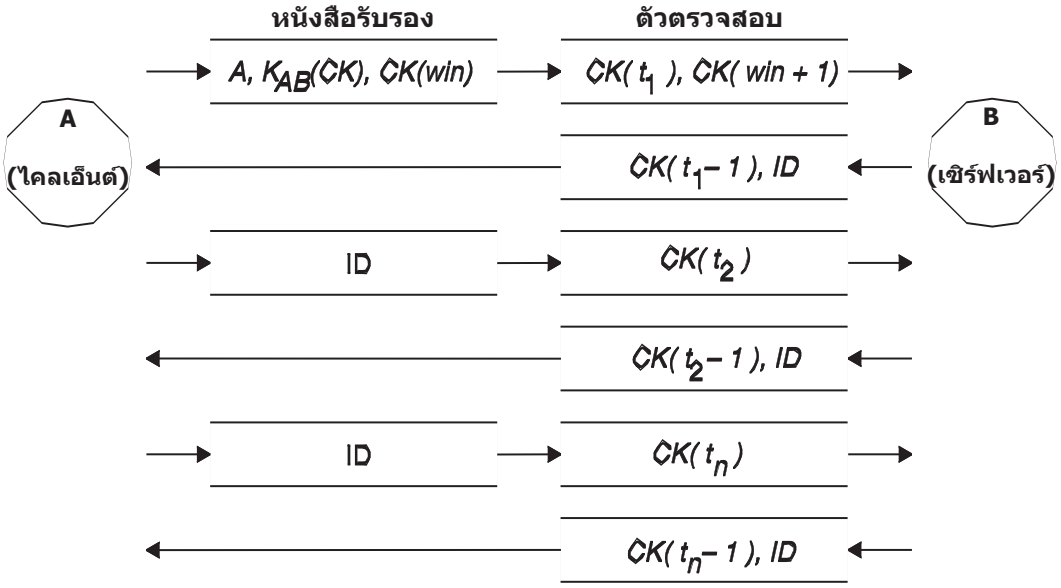
$$K_{AB} = PK_A^{SK}B$$

เซิร์ฟเวอร์ และไคลเอ็นต์เท่านั้นที่สามารถคำนวณคีย์ร่วมนี้เนื่องจากการคำนวณเช่นนั้น จำเป็นที่จะต้องทราบคีย์ลับ หรือพับลิกคีย์อย่างใดอย่างหนึ่ง เนื่องจากคีย์ร่วมยาว 128 บิต และ DES ใช้คีย์ 56 บิต ไคลเอ็นต์และเซิร์ฟเวอร์จะดึง 56 บิตจาก คีย์ร่วมเพื่อจัดทำเป็นคีย์ DES

กระบวนการพิสูจน์ตัวตน Network File System:

เมื่อไคลเอ็นต์ต้องการคุยกับเซิร์ฟเวอร์ ไคลเอ็นต์จะสร้างคีย์แบบสุ่ม ที่ใช้สำหรับการเข้ารหัสการประทับเวลา คีย์นี้คือ *conversation key* (*CK*)

ไคลเอ็นต์เข้ารหัส conversation key โดยใช้คีย์ร่วม DES (อธิบาย ใน ข้อกำหนด การพิสูจน์ตัวตน) และส่งคีย์ไปยังเซิร์ฟเวอร์ในทรานแซกชัน RPC แรก การประมวลผลนี้ยังแสดงในรูปภาพต่อไปนี้



รูปที่ 15. กระบวนการพิสูจน์ตัวตน. รูปภาพนี้แสดง กระบวนการพิสูจน์ตัวตน

รูปนี้แสดงไคลเอ็นต์ A กำลังเชื่อมต่อกับเซิร์ฟเวอร์ B คำว่า *K*(*CK*) หมายถึง *CK* ถูกเข้ารหัสด้วยคีย์ร่วม DES *K* ในการร้องขอแรก RPC credential ไคลเอ็นต์มีชื่อไคลเอ็นต์ (*A*) conversation key (*CK*) และตัวแปรชื่อ *win* (window) ที่เข้ารหัสด้วย *CK* (ขนาดหน้าต่างดีฟอลต์คือ 30 นาที) ตัวตรวจสอบ ไคลเอ็นต์ในการร้องขอแรกมีการประทับเวลาที่เข้ารหัสและตัวตรวจสอบที่เข้ารหัสของหน้าต่างที่ระบุ *win* + 1 ตัวตรวจสอบหน้าต่างทำให้การคาดเดา credential ที่ถูกต้องทำได้ยากมากขึ้นและเพิ่มการรักษาความปลอดภัย

หลังการพิสูจน์ตัวตนไคลเอ็นต์ เซิร์ฟเวอร์เก็บรายการต่อไปนี้ ในตาราง credential:

- ชื่อไคลเอ็นต์ *A*
- Conversation key, *CK*
- หน้าต่าง
- การประทับเวลา

เซิร์ฟเวอร์ยอมรับเฉพาะการประทับเวลาที่มากกว่าค่าล่าสุดตามลำดับเวลาที่พบ เท่านั้น ดังนั้นทรานแซกชันที่เล่นซ้ำใดๆ จะถูกปฏิเสธทั้งหมด เซิร์ฟเวอร์ส่งกลับ ID ดัชนีใน ตาราง credential ไปยังไคลเอ็นต์ในตัวตรวจสอบ บวกกับการประทับเวลาไคลเอ็นต์ลบ 1 ที่เข้ารหัสโดย *CK* ไคลเอ็นต์ ทราบว่าเซิร์ฟเวอร์เท่านั้นที่ส่งตัวตรวจสอบ เนื่องจากมีเพียงเซิร์ฟเวอร์ เท่านั้นที่ทราบการประทับเวลาที่ไคลเอ็นต์ส่ง เหตุผลในการลบ 1 จากการประทับเวลาคือเพื่อให้แน่ใจว่าค่าจะไม่ถูกต้องและไม่สามารถนำไปใช้ใหม่ เป็นตัวตรวจสอบไคลเอ็นต์ได้ หลังทรานแซกชัน RPC แรกแล้ว ไคลเอ็นต์เพียงส่ง ID และการประทับเวลาที่เข้ารหัสไปยังเซิร์ฟเวอร์ และเซิร์ฟเวอร์จะส่งกลับ การประทับเวลาไคลเอ็นต์ลบ 1 ที่เข้ารหัสโดย *CK*

การตั้งชื่อ entity เน็ตเวิร์กสำหรับการพิสูจน์ตัวตน DES

การพิสูจน์ตัวตน DES ทำการตั้งชื่อโดยใช้ net names

net name คือสตริงของอักขระที่สามารถพิมพ์ได้เพื่อใช้พิสูจน์ตัวตน พับลิกคีย์และคีย์ลับถูกเก็บตาม per-net-name มากกว่าใช้รูปแบบ per-user-name แม็พ `netid.byname` NIS จะแม็พ net name ใน UID โลคัลและรายการการเข้าถึงแบบกลุ่ม

ชื่อผู้ใช้เป็นค่าเฉพาะภายในแต่ละโดเมน Net names ถูกกำหนดโดยการต่อ ระบบปฏิบัติการและ ID ผู้ใช้ที่มี NIS และ อินเตอร์เน็ตโดเมนเนม ระเบียบวิธีที่เหมาะสมสำหรับการตั้งค่าโดเมนคือการต่อท้าย อินเตอร์เน็ตโดเมนเนม (com, edu, gov, mil) เข้ากับโลคัลโดเมนเนม

ชื่อเน็ตเวิร์กถูกกำหนดให้แก่เครื่องเช่นเดียวกับผู้ใช้ net name ของเครื่องถูกจัดรูปแบบเหมือนกับของผู้ใช้ ตัวอย่าง เครื่องชื่อ `hal` ใน โดเมน `eng.xyz.com` มี net name `unix.hal@eng.xyz.com` การพิสูจน์ตัวตนที่ถูกต้องของเครื่องเป็นสิ่งสำคัญสำหรับเครื่องที่ไม่มีดิสก์ที่ จำเป็นต้องมีเข้าถึงเต็มที่สำหรับโฮมไดเร็กทอรีของตนบนเน็ตเวิร์ก

ในการพิสูจน์ตัวตนผู้ใช้จากรีโมตโดเมนใดๆ ให้สร้างรายการสำหรับการพิสูจน์ตัวตนนั้นใน ฐานข้อมูล NIS สองฐานข้อมูลรายการหนึ่งคือ รายการสำหรับพับลิกคีย์และคีย์ลับ อีกรายการหนึ่งสำหรับการแม็พ UID โลคัลและรายการการเข้าถึงแบบกลุ่ม จากนั้นผู้ใช้ในรีโมตโดเมนสามารถเข้าถึง เซอร์วิสของโลคัลเน็ตเวิร์กทั้งหมดได้ เช่น NFS และรีโมตล็อกอิน

ไฟล์ /etc/publickey

ไฟล์ `/etc/publickey` มีชื่อ และพับลิกคีย์ซึ่ง NIS ใช้ เพื่อสร้างการแม็พ publickey

แม็พ publickey ถูกใช้สำหรับในระบบเน็ตเวิร์กที่มีความปลอดภัย แต่ละรายการในไฟล์ประกอบด้วยชื่อผู้ใช้ เน็ตเวิร์ก (ซึ่งอ้างอิงชื่อผู้ใช้หรือชื่อโฮสต์) ตามด้วย พับลิกคีย์ผู้ใช้ (ในรูปแบบ hexadecimal notation) โคลอน และคีย์ลับที่เข้ารหัส โดยผู้ใช้ (ในรูปแบบ hexadecimal notation เช่นกัน) โดยดีฟอลต์ เฉพาะผู้ใช้ในไฟล์ `/etc/publickey` เท่านั้น คือผู้ใช้ nobody

อย่าใช้เท็กซ์เอดิเตอร์เพื่อเปลี่ยนแปลงไฟล์ `/etc/publickey` เนื่องจากไฟล์มีคีย์การเข้ารหัส ในการเปลี่ยนแปลงไฟล์ `/etc/publickey` ให้ใช้คำสั่ง **chkey** หรือ **newkey**

ข้อควรพิจารณาการบูตระบบด้วยพับลิกคีย์

เมื่อรีสตาร์ทเครื่องหลังเกิดปัญหาไฟฟ้าขัดข้อง คีย์ลับที่เก็บไว้ทั้งหมด จะสูญหายไป และไม่มีกระบวนการใดที่สามารถเข้าถึงเน็ตเวิร์กเซอร์วิสที่มีการรักษาความปลอดภัยได้ เช่นการเมาท์ NFS การประมวลผล Root สามารถดำเนินต่อได้ถ้ามีบางคนป้อนรหัสผ่านที่ถอดรหัสคีย์ลับของผู้ใช้ root ได้ วิธีแก้ปัญหา คือให้เก็บคีย์ลับที่เข้ารหัสโดยผู้ใช้ root ในไฟล์ที่เซิร์ฟเวอร์คีย์สามารถอ่านได้

ไม่ใช่ว่าการเรียกใช้รูทีนย่อย **setuid** ทั้งหมดจะดำเนินงานได้อย่างถูกต้อง ตัวอย่าง ถ้ารูทีนย่อย **setuid** ถูกเรียกใช้โดยเจ้าของ A และเจ้าของ A ยังไม่ได้ล็อกอินเข้าสู่เครื่องตั้งแต่ที่เริ่มทำงาน รูทีนย่อยจะไม่สามารถเข้าถึง เน็ตเวิร์กเซอร์วิสที่มีการรักษาความปลอดภัยใดๆ เช่น A อย่างไรก็ตาม การเรียกใช้รูทีนย่อย **setuid** ส่วนใหญ่ถูกเรียกใช้ผู้ใช้ root และคีย์ลับของผู้ใช้ root จะถูกเก็บตอนเริ่มทำงานเสมอ

ข้อควรพิจารณาผลการทำงาน Secure Network File System

มีหลายแนวทางที่ secure NFS มีผลต่อผลการทำงานระบบ

- ทั้งไคลเอ็นต์และเซิร์ฟเวอร์ต้องคำนวณหาคีย์ร่วม เวลา ที่ใช้คำนวณคีย์ร่วมนั้นประมาณหนึ่งวินาที เป็นผลให้ ใช้เวลาประมาณสองวินาทีในการสร้างการเชื่อมต่อ RPC เริ่มต้น เนื่องจากทั้งไคลเอ็นต์และเซิร์ฟเวอร์ต้องกระทำการดำเนินการนี้หลังจาก การเชื่อมต่อ RPC เริ่มต้นแล้ว คีย์เซิร์ฟเวอร์แคชผลลัพธ์ของ การคำนวณครั้งก่อนหน้าไว้ และทำให้ไม่ต้องคำนวณหาคีย์ร่วม ใหม่ทุกครั้ง
- แต่ละทรานแซกชัน RPC จำเป็นต้องมีการดำเนินการเข้ารหัส DES ต่อไปนี้:
  1. ไคลเอ็นต์เข้ารหัสการประทับเวลาการร้องขอ
  2. เซิร์ฟเวอร์ถอดรหัสการประทับเวลา
  3. เซิร์ฟเวอร์เข้ารหัสการประทับเวลาตอบกลับ
  4. ไคลเอ็นต์ถอดรหัสการประทับเวลา

เนื่องจากผลการทำงานระบบอาจลดลงโดย secure NFS ขอให้ให้น้ำหนัก ผลดีของความปลอดภัยที่เพิ่มขึ้นเทียบกับความต้องการด้านผลการทำงานของระบบ

รายการตรวจสอบ Secure Network File System

รายการตรวจสอบนี้ช่วยให้แน่ใจว่า secure NFS ทำงานได้อย่างถูกต้อง

- เมื่อเมาท์ระบบไฟล์ด้วยอ็อพชัน **–secure** บนไคลเอ็นต์ ชื่อเซิร์ฟเวอร์ต้องตรงกับชื่อโฮสต์เซิร์ฟเวอร์ในไฟล์ /etc/hosts ถ้าเซิร์ฟเวอร์ชื่อถูกใช้สำหรับการหาชื่อโฮสต์ ขอให้แน่ใจว่าข้อมูล โฮสต์ที่ส่งกลับโดยเซิร์ฟเวอร์ชื่อจะตรงกับรายการในไฟล์ /etc/hosts ข้อผิดพลาดการพิสูจน์ตัวตนส่งผลต่อเมื่อชื่อเหล่านี้ไม่ตรงกันเนื่องจาก net names สำหรับเครื่องต้องอิงตามรายการหลักในไฟล์ /etc/hosts และคีย์ในแม็พ **publickey** ถูกเข้าถึงโดย net name
- อย่าใช้การเอ็กซ์พอร์ตและการเมาท์แบบที่มีและไม่มีความปลอดภัยผสมกัน มิฉะนั้น การเข้าถึงไฟล์ อาจถูกพิจารณาว่าไม่ถูกต้อง ตัวอย่าง ถ้าเครื่องไคลเอ็นต์เมาท์ ระบบไฟล์ที่มีความปลอดภัยโดยไม่มีอ็อพชัน **–secure** หรือเมาท์ระบบที่ไม่มีความปลอดภัยด้วยอ็อพชัน **–secure** ผู้ใช้จะมีการเข้าถึงเป็น nobody มากกว่าจะเป็นตัวผู้ใช้เอง เงื่อนไขนี้ยังเกิดขึ้นได้ถ้าผู้ใช้ไม่รู้จัก NIS และผู้ใช้รายนั้นพยายามสร้างหรือแก้ไขไฟล์บน ระบบไฟล์ที่มีความปลอดภัย
- เนื่องจาก NIS ต้องกระจาย(แม็พใหม่หลังจากการใช้แต่ละครั้งของคำสั่ง **chkey** และ **newkey** ให้ใช้คำสั่งเหล่านี้เมื่อเน็ตเวิร์กมีปริมาณงานไม่โหลดมากนักเท่านั้น
- อย่าสร้างไฟล์ /etc/keystore หรือไฟล์ /etc/.rootkey ถ้าคุณติดตั้งอีกครั้ง ย้าย หรืออัพเกรดเครื่อง ให้บันทึกไฟล์ /etc/keystore และ /etc/.rootkey

- แนะนำให้ผู้ใช้ใช้คำสั่ง **yppasswd** แทนคำสั่ง **passwd** เพื่อเปลี่ยนรหัสผ่าน การทำเช่นนั้นจะช่วยจัดเก็บรหัสผ่านและไพรเวตคีย์ให้ซิงโครไนซ์กัน
- เนื่องจากคำสั่ง **login** ไม่เรียกข้อมูลคีย์ออกมา จากแม็พ publickey สำหรับ **keyserv** daemon ผู้ใช้ต้องรันคำสั่ง **keylogin** คุณอาจต้องการ ใส่คำสั่ง **keylogin** ในไฟล์ `profile` ของผู้ใช้แต่ละราย เพื่อรันคำสั่งโดยอัตโนมัติระหว่างการล็อกอิน คำสั่ง **keylogin** บังคับให้ผู้ใช้ต้องป้อนรหัสผ่านอีกครั้ง
- เมื่อคุณสร้างคีย์สำหรับผู้ใช้ root ที่แต่ละโฮสต์โดยใช้คำสั่ง **newkey –h** หรือ **chkey** คุณต้องรันคำสั่ง **keylogin** เพื่อส่งคีย์ใหม่ไปยัง **keyserv** daemon คีย์ถูก เก็บในไฟล์ `/etc/.rootkey` ซึ่งถูกอ่านโดย **keyserv** daemon ในแต่ละครั้งที่ daemon เริ่มทำงาน
- ให้ตรวจสอบ **yppasswdd** และ **ypupdated** daemons เป็นระยะว่ากำลังทำงานอยู่บนเซิร์ฟเวอร์หลัก NIS Daemon เหล่านี้จำเป็นสำหรับการบำรุงรักษาแม็พ publickey
- ให้ตรวจสอบว่า **keyserv** daemon เป็นระยะว่ากำลังทำงาน บนเครื่องทุกเครื่องโดยใช้ secure NFS

การตั้งค่า Network File System ที่มีการรักษาความปลอดภัย

เมื่อต้องการกำหนดคอนฟิก NFS ด้วยความปลอดภัยบนเซิร์ฟเวอร์ NIS หลักหรือสำรอง, ให้ทำตามโพรซีเดอร์ต่อไปนี้

1. บนเซิร์ฟเวอร์มาสเตอร์ NIS ให้สร้างรายการสำหรับผู้ใช้แต่ละรายในไฟล์ NIS `/etc/publickey` โดยใช้คำสั่ง **newkey** ดังนี้:
   - สำหรับผู้ใช้ทั่วไป พิมพ์:

     ```
     smit newkey
     ```

     หรือ

     ```
     newkey -u username
     ```

     สำหรับผู้ใช้ root บนเครื่องโฮสต์ พิมพ์:

     ```
     newkey -h hostname
     ```

   - อีกทางหนึ่ง ผู้ใช้สามารถสร้างพับลิกคีย์ของตนเองโดยใช้คำสั่ง **chkey** หรือ **newkey**
2. สร้างแม็พ NIS publickey แม็พ NIS `publickey.byname` ที่เกี่ยวข้อง อยู่บนเซิร์ฟเวอร์ NIS เท่านั้น
3. ยกเลิกการใส่ข้อคิดเห็น stanzas ต่อไปนี้ในไฟล์ `/etc/rc.nfs`:

   ```
   #if [ -x /usr/sbin/keyserv ]; then
   #  startsrc -s keyserv
   #fi
   #if [ -x /usr/lib/netsvc/yp/rpc.ypupdated -a -d /etc/yp/`domainname` ]; then
   #  startsrc -s ypupdated
   #fi
   #DIR=/etc/passwd
   #if [ -x /usr/lib/netsvc/yp/rpc.yppasswdd -a -f $DIR/passwd ]; then
   #  startsrc -s yppasswdd
   #fi
   ```

4. เริ่มทำงาน **keyserv, ypupdated** และ **yppasswdd** daemons โดยใช้คำสั่ง **startsrc**

ในการตั้งค่า secure NFS บนไคลเอ็นต์ NIS ให้เริ่มทำงาน **keyserv** daemon โดยใช้คำสั่ง **startsrc**

การเอ็กซ์พอร์ตระบบไฟล์โดยใช้ Secure Network File System

คุณสามารถเอ็กซ์พอร์ต NFS ที่ปลอดภัยได้โดยใช้หนึ่งใน โพรซีเดอร์ต่อไปนี้

- ในการเอ็กซ์พอร์ตระบบไฟล์ secure NFS โดยใช้ SMIT ดำเนินการ ขั้นตอนต่อไปนี้:
  1. ตรวจสอบว่า NFS กำลังทำงานอยู่โดยการรันคำสั่ง **lssrc –g nfs** เอาต์พุตบ่งชี้ว่า nfsd และ rpc.mountd daemons แอ็คทีฟ
  2. ตรวจสอบว่ามีแม็พ publickey อยู่และ keyserv daemon กำลังทำงานอยู่ สำหรับข้อมูลเพิ่มเติม ดูที่ "การตั้งค่า Network File System ที่มีการรักษาความปลอดภัย" ในหน้า 313
  3. รันพาธด่วน **smit mknfsexp**
  4. ระบุค่าที่เหมาะสมสำหรับ PATHNAME ของไดเร็กทอรีที่จะ เอ็กซ์พอร์ต MODE ที่จะเอ็กซ์พอร์ตไดเร็กทอรี และ EXPORT ไดเร็กทอรีขณะนี้ ระบบ รีสตาร์ท หรือทั้งสองฟิลด์ ระบุ yes สำหรับฟิลด์ Use SECURE option
  5. ระบุคุณสมบัติเผื่อเลือกอื่นๆ หรือยอมรับค่า ดีฟอลต์
  6. ออกจาก SMIT ถ้าไฟล์ /etc/exports ไม่มี จะสร้างไฟล์ขึ้นมา
  7. ทำซ้ำขั้นตอน 3 ถึง 6 สำหรับแต่ละไดเร็กทอรีที่คุณต้องการเอ็กซ์พอร์ต
- ในการเอ็กซ์พอร์ตระบบไฟล์ secure NFS โดยใช้เท็กซ์เอดิเตอร์ ดำเนิน ขั้นตอนต่อไปนี้:
  1. เปิดไฟล์ /etc/exports ด้วยเท็กซ์เอดิเตอร์ ที่คุณตองการ
  2. สร้างรายการสำหรับแต่ละไดเร็กทอรีที่จะเอ็กซ์พอร์ต โดยใช้ชื่อพาธแบบเต็ม ของไดเร็กทอรี แสดงรายการแต่ละไดเร็กทอรีที่จะถูกเอ็กซ์พอร์ตเริ่มต้น ที่ขอบซ้าย ไม่มีไดเร็กทอรีใดที่ควรรวมไดเร็กทอรีอื่นใด ที่ถูกเอ็กซ์พอร์ตไปแล้ว ดูที่เอกสารคู่มือไฟล์ /etc/exports สำหรับรายละเอียดของไวยากรณ์ทั้งหมดสำหรับรายการใน ไฟล์ /etc/exports รวมถึงวิธีระบุ อ็อพชัน secure
  3. บันทึกและปิดไฟล์ /etc/exports
  4. ถ้าขณะนี้ NFS กำลังทำงานอยู่ พิมพ์:

     ```
     /usr/sbin/exportfs -a
     ```

     โดยใช้ อ็อพชัน **–a** กับคำสั่ง **exportfs** เพื่อส่งข้อมูลทั้งหมดในไฟล์ /etc/exports ไปยังเคอร์เนล
- ในการเอ็กซ์พอร์ตระบบไฟล์ NFS ชั่วคราว (คือ โดยไม่เปลี่ยนแปลง ไฟล์ /etc/exports) พิมพ์:

  ```
  exportfs -i -o secure /dirname
  ```

  โดยที่ dirname คือ ชื่อของระบบไฟล์ที่คุณต้องการเอ็กซ์พอร์ต คำสั่ง **exportfs –i** ระบุว่าไฟล์ /etc/exports จะไม่ถูกตรวจสอบสำหรับตรวจสอบที่ระบุ และอ็อพชันทั้งหมด ถูกดำเนินการจากบรรทัดคำสั่งโดยตรง

การเมาท์ระบบไฟล์โดยใช้ Secure Network File System

คุณสามารถเมาท์ไดเร็กทอรี secure NFS อย่างแน่นอน

ในการเมาท์ไดเร็กทอรี secure NFS อย่างแน่นอน ดำเนินขั้นตอน ต่อไปนี้:

1. ตรวจสอบว่าเซิร์ฟเวอร์ NFS ได้เอ็กซ์พอร์ตไดเร็กทอรี โดยการรันคำสั่ง:

   ```
   showmount -e ServerName
   ```

   โดยที่ ServerName คือ ชื่อของเซิร์ฟเวอร์ NFS คำสั่งนี้แสดงชื่อของไดเร็กทอรีที่ ถูกเอ็กซ์พอร์ตจากเซิร์ฟเวอร์ NFS ในขณะนี้ ถ้าไดเร็กทอรีที่คุณต้องการเมาท์ ไม่มีอยู่ในรายการ ให้เอ็กซ์พอร์ตไดเร็กทอรีจากเซิร์ฟเวอร์

2. สร้างจุดเมาท์โลคัลโดยใช้คำสั่ง **mkdir** สำหรับ NFS เพื่อให้ดำเนินการเมาท์ได้เสร็จสมบูรณ์ ไดเร็กทอรีที่ทำหน้าที่เป็นจุดเมาท์ (หรือจุดยึด) ของการเมาท์ NFS ต้องถูกแสดงอยู่ ไดเร็กทอรีนี้ควร ว่างเปล่า จุดเมาท์นี้สามารถสร้างขึ้นได้เช่นเดียวกับไดเร็กทอรีอื่นๆ และไม่จำเป็นต้อง ใช้แอ็ตทริบิวต์พิเศษใดๆ
3. ตรวจสอบว่าแม็พ publickey มีอยู่และ keyserv daemon กำลังทำงานอยู่ สำหรับข้อมูลเพิ่มเติม ดูที่ "การตั้งค่า Network File System ที่มีการรักษาความปลอดภัย" ในหน้า 313
4. ประเภท

   ```
   mount -o secure ServerName:/remote/directory /local/directory
   ```

   โดยที่ `ServerName` คือชื่อของเซิร์ฟเวอร์ NFS `/remote/directory` คือ ไดเร็กทอรีบนเซิร์ฟเวอร์ NFS ที่คุณต้องการเมาท์ และ `/local/directory` คือ จุดเมาท์บนไคลเอ็นต์ NFS

   หมายเหตุ: ผู้ใช้ root เท่านั้นที่สามารถเมาท์ secure NFS

การแม็พ identity เอ็นเตอร์ไพรซ์

สภาวะแวดล้อมเน็ตเวิร์กทุกวันนี้ประกอบด้วยกลุ่มของระบบและแอ็พพลิเคชัน ที่ซับซ้อน เป็นผลจากความต้องการในการจัดการรีจิสทรีผู้ใช้หลายรีจิสทรี การรับมือกับรีจิสทรีผู้ใช้หลายรีจิสทรีที่เติบโตอย่างรวดเร็วเป็นปัญหาการดูแล ที่ใหญ่ขึ้นที่ส่งผลต่อผู้ใช้ ผู้ดูแลระบบ และผู้พัฒนาแอ็พพลิเคชัน Enterprise Identity Mapping (EIM) อนุญาตให้ผู้ดูแลระบบและผู้พัฒนาแอ็พพลิเคชัน จัดการปัญหานี้

ส่วนนี้อธิบายเกี่ยวกับปัญหา แสดงกรอบวิธีการใช้ในอุตสาหกรรม ปัจจุบัน และอธิบายวิธีการ EIM

การจัดการรีจิสทรีผู้ใช้หลายรีจิสทรี

ผู้ดูแลระบบจำนวนมากจัดการเน็ตเวิร์กที่รวมระบบแ และเซิร์ฟเวอร์ที่แตกต่างกัน แต่ละเน็ตเวิร์กด้วยวิธีเฉพาะในการจัดการผู้ใช้ผ่านทางรีจิสทรีผู้ใช้ ที่แตกต่างกัน

ในเน็ตเวิร์กที่ซับซ้อนเหล่านี้ ผู้ดูแลระบบมีหน้าที่ จัดการ identities และรหัสผ่านของผู้ใช้แต่ละคนทั่วทั้งหลายระบบ นอกเหนือจากนั้น ผู้ดูแลระบบต้องซิงโครไนซ์ identities และรหัสผ่านเหล่านี้เป็นประจำ ผู้ใช้ ต้องรับภาระจดจำ identities และรหัสผ่านหลายค่า และต้องคอยทำให้ ซิงโครไนซ์กัน เนื่องจากค่าใช้จ่ายในการดำเนินการกับผู้ใช้และผู้ดูแลระบบในสภาวะแวดล้อมนี้สูง ผู้ดูแลระบบต้องใช้เวลาอันมีค่าในแก้ปัญหาความพยายามล็อกอิน ที่ล้มเหลว และการตั้งค่ารหัสผ่านที่ลืมใหม่แทนการจัดการเอ็นเตอร์ไพรซ์

ปัญหา ของการจัดการรีจิสทรีผู้ใช้หลายรีจิสทรียังส่งผลต่อผู้พัฒนาแอ็พพลิเคชัน ที่ต้องการให้มีแอ็พพลิเคชันแบบหลายเทียร์หรือที่แตกต่างกัน ลูกค้า มีข้อมูลทางธุรกิจที่สำคัญกระจายอยู่ในระบบประเภทต่างๆ มากหมาย ที่การประมวลผลระบบของตนเองจะมีรีจิสทรีผู้ใช้เป็นของตนเองในแต่ละระบบ ผลที่ตามมาคือ ผู้พัฒนา ต้องสร้างผู้ใช้รีจิสทรีของตนเองและเชื่อมโยงซีแมนทิกส์ด้านความปลอดภัย สำหรับแอ็พพลิเคชันของตน แม้วิธีนี้จะช่วยแก้ปัญหาสำหรับผู้พัฒนา แอ็พพลิเคชัน แต่ก็เพิ่มค่าใช้จ่ายในการดำเนินการกับผู้ใช้และผู้ดูแลระบบ

วิธีการ Current เพื่อการแม็พ identity เอ็นเตอร์ไพรซ์

วิธีการใช้งานในอุตสาหกรรมปัจจุบันหลายวิธีการเพื่อการแก้ปัญหา การจัดการรีจิสทรีผู้ใช้หลายรีจิสทรีที่มีอยู่ แต่ทั้งหมดที่มีไม่ได้ให้วิธีแก้ปัญหาที่สมบูรณ์ ตัวอย่าง Lightweight Directory Access Protocol (LDAP) มีวิธีแก้ปัญหารีจิสทรีผู้ใช้แบบกระจาย อย่างไรก็ตาม ในการใช้วิธีแก้ปัญหาเช่น LDAP นั้นผู้ดูแลระบบต้องจัดการอีกรีจิสทรีผู้ใช้ และซีแมนทิกส์ด้านความปลอดภัย หรือแทนที่แอ็พพลิเคชันที่มีอยู่ ที่ถูกสร้างเพื่อใช้รีจิสทรีเหล่านั้น

การใช้วิธีแก้ปัญหาประเภทนี้ ผู้ดูแลระบบต้องจัดการกลวิธีการรักษาความปลอดภัย หลายรูปแบบสำหรับแต่ละรีซอร์ส จึงเพิ่มโอเวอร์เฮดการดูแล และเพิ่มแนวโน้มความเป็นไปได้ในการเปิดช่องว่างด้านความปลอดภัย เมื่อมีหลายกลวิธีที่สนับสนุนรีซอร์สเดียว โอกาส ของการเปลี่ยนแปลงสิทธิโดยใช้กลวิธีหนึ่ง และลืม ที่จะเปลี่ยนสิทธิของกลวิธีอื่นๆ หนึ่งหรือหลายวิธี ก็ยิ่งเพิ่มมากขึ้น ตัวอย่าง การเปิดช่องโหวด้านความปลอดภัยอาจส่งผลเมื่อผู้ใช้ ถูกปฏิเสธอย่างเหมาะสมในการเข้าถึงผ่านส่วนการติดต่อหนึ่ง แต่ได้รับอนุญาต ให้เข้าถึงผ่านส่วนการติดต่ออื่นอย่างน้อยหนึ่งส่วนการติดต่อ

หลังจากทำงานนี้เสร็จ ผู้ดูแลระบบพบว่าเขายัง ไม่สามารถแก้ปัญหาได้โดยสมบูรณ์โดยทั่วไป เอ็นเตอร์ไพรซ์ได้ลงทุน เป็นเงินจำนวนมากในรีจิสทรีผู้ใช้ปัจจุบัน และในซีแมนทิกส์ด้านความปลอดภัย ที่เชื่อมโยงเพื่อให้วิธีแก้ปัญหาประเภทนี้สามารถใช้งานได้ การสร้างรีจิสทรีผู้ใช้อีกรีจิสทรีหนึ่ง รวมถึงซีแมนทิกส์ด้านความปลอดภัยที่เชื่อมโยงช่วย แก้ปัญหาสำหรับผู้ให้บริการแอ็พพลิเคชัน แต่ไม่ได้ช่วยแก้ปัญหาสำหรับ ผู้ใช้หรือผู้ดูแลระบบ

วิธีแก้ปัญหาอีกทางหนึ่งคือใช้วิธีการ single sign-on หลายๆ ผลิตภัณฑ์ ที่มีอยู่ จะอนุญาตให้ผู้ดูแลระบบจัดการไฟล์ที่มี identities และรหัสผ่านของผู้ใช้ทั้งหมด อย่างไรก็ตาม วิธีนี้ มีจุดอ่อนหลายข้อ:

- วิธีนี้แสดงปัญหาเพียงปัญหาเดียวที่ผู้ใช้เผชิญ แม้ว่า จะอนุญาตให้ผู้ใช้เข้าสู่ระบบได้หลายระบบโดยการใส่ identity และรหัสผ่านเดียว ผู้ใช้ยังจำเป็นต้องใส่รหัสผ่านบนระบบอื่นๆ หรือจำเป็นต้องจัดการรหัสผ่านเหล่านี้
- วิธีการนี่ก่อให้เกิดปัญหาใหม่โดยการสร้างช่องโหวด้านความปลอดภัยเนื่องจาก รหัสผ่านแบบข้อความ หรือสามารถถอดรหัสได้ถูกเก็บในไฟล์เหล่านี้ รหัสผ่าน ไม่ควรถูกเก็บในไฟล์แบบข้อความทั่วไป หรือเข้าถึงได้ง่าย ไม่ว่าบุคคลใด รวมถึงผู้ดูแลระบบ
- ไม่ได้ช่วยแก้ปัญหาของผู้พัฒนาแอ็พพลิเคชันบุคคลที่สาม ที่จัดให้มีแอ็พพลิเคชันที่ไม่เข้ากัน มีหลายเทียร์ บุคคลที่สามยังคงต้องจัดให้มีรีจิสทรีผู้ใช้ที่เป็นของตนสำหรับแอ็พพลิเคชันของพวกเขา

นอกเหนือจากจุดอ่อนเหล่านี้แล้ว บางเอ็นเตอร์ไพรซ์ใช้วิธีแก้ปัญหาเหล่านี้ เนื่องจากช่วยลดปัญหารีจิสทรีผู้ใช้หลายๆ ประการ

การใช้งานการแม็พ identity เอ็นเตอร์ไพรซ์

สถาปัตยกรรม EIM อธิบายความสัมพันธ์ระหว่างบุคคล หรือ entity (เช่น ไฟล์เซิร์ฟเวอร์และพรินต์เซิร์ฟเวอร์) ในเอ็นเตอร์ไพรซ์และ หลายๆ identities ที่แทนเอ็นเตอร์ไพรซ์ภายในเอ็นเตอร์ไพรซ์ นอกจากนั้น EIM ยังมีชุดของ APIs ที่อนุญาตให้แอ็พพลิเคชันถามคำถามเกี่ยวกับ ความสัมพันธ์ของตน

ตัวอย่าง การกำหนด identity ผู้ใช้ของบุคคลในรีจิสทรีผู้ใช้ คุณ สามารถกำหนดได้ว่า identity ในอีกรีจิสทรีผู้ใช้หนึ่งที่แทนบุคคล เดียวกันนั้น ถ้าผู้ใช้ได้รับอนุญาตด้วย identity หนึ่งและคุณสามารถแม็พ identity นั้นกับ identity ที่เหมาะสมในอีกรีจิสทรีผู้ใช้หนึ่ง ผู้ใช้ ไม่จำเป็นต้องใช้ credentials เพื่อการพิสูจน์ตัวตนอีกครั้ง คุณจำเป็นต้องทราบ เพียงว่า identity ใดแทนผู้ใช้คนนั้นในอีกรีจิสทรีผู้ใช้หนึ่ง ดังนั้น EIM จัดให้มีฟังก์ชันการแม็พ identity โดยสรุปสำหรับเอ็นเตอร์ไพรซ์

ความสามารถในการแม็พ identities ของผู้ใช้ในรีจิสทรีต่างกัน ช่วยให้เกิดประโยชน์หลายประการ อันดับแรก แอ็พพลิเคชันมีความยืดหยุ่นต่อ การใช้รีจิสทรีหนึ่งเพื่อการพิสูจน์ตัวตนขณะที่ใช้รีจิสทรีที่ต่างกัน เพื่อการอนุญาต ตัวอย่าง ผู้ดูแลระบบสามารถแม็พ SAP identity เพื่อเข้าถึงรีซอร์ส SAP

การแม็พ Identity จำเป็นที่ผู้ดูแลระบบต้องดำเนินขั้นตอนต่อไปนี้:

1. สร้าง EIM identifiers ที่แทนบุคคลหรือ entities ในเอ็นเตอร์ไพรซ์ของตน
2. สร้างนิยามรีจิสทรี EIM ที่อธิบายรีจิสทรีผู้ใช้ที่มีอยู่ ในเอ็นเตอร์ไพรซ์ของตน
3. กำหนดความสัมพันธ์ระหว่าง identities ผู้ใช้ในรีจิสทรีเหล่านั้น กับ EIM identifiers ที่สร้างขึ้น

ไม่จำเป็นต้องเปลี่ยนแปลงโค้ดใดๆ กับรีจิสทรีที่มีอยู่ ไม่จำเป็นต้องทำการแม็พ identities ทั้งหมดในรีจิสทรีผู้ใช้ EIM อนุญาตให้มีการแม็พแบบ one-to-many (หรือ กล่าวอีกอย่างคือ ผู้ใช้คนเดียวมี identity มากกว่าหนึ่งในรีจิสทรีผู้ใช้เดียว) EIM ยังอนุญาตให้มีการแม็พแบบ many-to-one (หรือกล่าวอีกอย่างคือ หลายผู้ใช้แบ่งใช้ identity เดียวในรีจิสทรีผู้ใช้เดียว ซึ่งแม้จะสนับสนุนให้ทำได้ แต่ไม่แนะนำให้ใช้ เนื่องจากเหตุผลด้านความปลอดภัย) ผู้ดูแลระบบสามารถแทนรีจิสทรีผู้ใช้ใดๆ เป็นประเภทใดๆ ใน EIM

EIM ไม่จำเป็นต้องทำสำเนาข้อมูลที่มีอยู่ไปยังที่เก็บใหม่ และพยายาม ให้ทั้งสองสำเนาซิงโครไนซ์กัน มีเพียงข้อมูลใหม่เท่านั้นที่ EIM สร้างขึ้นคือ ข้อมูลเกี่ยวกับความสัมพันธ์ ผู้ดูแลระบบจัดการข้อมูลนี้ไดเร็กทอรี LDAP ซึ่งช่วยให้มีความยืดหยุ่นต่อการจัดการข้อมูลในที่เดียว และมี เรพลิคาในที่ที่ข้อมูลถูกใช้

Kerberos

Kerberos คือเซอร์วิสในการพิสูจน์ตัวตนบนเน็ตเวิร์ก ที่จัดให้มี วิธีในการตรวจสอบ identities ของ principals บนเน็ตเวิร์กที่ไม่ปลอดภัย เชิงกายภาพ Kerberos จัดให้มีการพิสูจน์ตัวตนร่วมกัน data integrity และความเป็นส่วนตัวภายใต้สมมติฐานว่าที่ว่าการรับส่งข้อมูลบนเน็ตเวิร์กเสี่ยงต่อ การดักขโมย การตรวจสอบ และการสับเปลี่ยน

หลักการ Kerberos เป็น identity เฉพาะที่ใช้เซอร์วิสการพิสูจน์ตัวตน Kerberos Kerberos ตรวจสอบ identities โดยไม่เชื่อถือการพิสูจน์ตัวตน โดยระบบปฏิบัติการของโฮสต์ การให้ความไว้วางใจต่อโฮสต์แอดเดรส หรือความต้องการ ความปลอดภัยเชิงกายภาพของโฮสต์ทั้งหมดบนเน็ตเวิร์ก

ตั๋ว Kerberos คือ credentials ที่ใช้ยืนยัน identity ของคุณ ตั๋ว มีสองประเภท: *ตั๋วในการให้สิทธิตั๋ว ตั๋วสำหรับเซอร์วิส* ตั๋วในการให้สิทธิตั๋ว ใช้สำหรับการร้องขอ identity เริ่มต้นของคุณ เมื่อล็อกอินเข้าสู่ระบบโฮสต์ คุณจำเป็นต้องใช้อย่างอย่างเพื่อยืนยัน identity ของคุณ เช่นรหัสผ่าน หรือโทเค็น หลังจากที่คุณมีตั๋วที่ใช้ในการให้สิทธิตั๋วแล้ว คุณจะสามารถ ใช้เพื่อร้องขอตั๋วสำหรับเซอร์วิสเพื่อใช้งานเซอร์วิส เฉพาะ วิธีการของตั๋วทั้งสองนี้เรียกว่า *บุคคลที่สาม ที่ไว้วางใจ* ของ Kerberos ตั๋วที่ใช้ในการให้สิทธิตั๋วของคุณ จะพิสูจน์ตัวตนของคุณกับเซิร์ฟเวอร์ Kerberos และตั๋วสำหรับเซอร์วิสของคุณ คือการแนะนำตัวที่ปลอดภัยแก่เซอร์วิส

บุคคลที่สามที่ไว้วางใจหรือสื่อกลางใน Kerberos ถูกเรียก *Key Distribution Center* (KDC) KDC ออกตั๋วทั้งหมดของ Kerberos ให้แก่ไคลเอ็นต์

ภาพรวมคำสั่งรีโมตที่ปลอดภัย

ข้อมูลต่อไปนี้ให้รายละเอียดเกี่ยวกับ คำสั่งรีโมตที่ปลอดภัย

หมายเหตุ:

1. เริ่มต้นด้วย Distributed Computing Environment (DCE) เวอร์ชัน 2.2 เซิร์ฟเวอร์การรักษาความปลอดภัย DCE สามารถคืนค่าตั๋ว Kerberos Version 5
2. คำสั่งรีโมตที่มีความปลอดภัยทั้งหมด(rcmds) ใช้ไลบรารี Kerberos เวอร์ชัน 5 ที่จัดเตรียมไว้โดย IBM Network Authentication Service (NAS) ซึ่งพร้อมใช้งานบน DVD แพ็กเสริม คุณต้องติดตั้งชุดไฟล์ `krb5.client.rte`, ซึ่งยังมีอยู่บน DVD แพ็กเสริม
3. ถ้าคุณกำลังโอนย้ายระบบปฏิบัติการ AIX ของคุณโดยใช้สื่อบันทึก DVD และ Kerberos ถูกติดตั้งไว้แล้ว, สคริปต์การติดตั้งจะพร้อมต์ให้คุณติดตั้ง `krb5.client.rte` จาก DVD แพ็กเสริม
4. ถ้าคุณกำลังโอนย้ายระบบปฏิบัติการ AIX ของคุณโดยใช้รีซอร์ส NIM และ Kerberos ได้ถูกติดตั้งไว้แล้ว, ให้เพิ่ม `krb5` ไปยังไดเร็กทอรี lpp_source ของคุณ

คำสั่งรีโมตที่มีความปลอดภัย (rcmds) คือ **rlogin, rcp, rsh, telnet,** และ **ftp** คำสั่งเหล่านี้โดยส่วนใหญ่ยังรู้จักเป็นวิธีการพิสูจน์ตัวตน AIX มาตรฐาน เมธอดเพิ่มเติมที่จัดเตรียมไว้คือ Kerberos

เมื่อใช้วิธีการพิสูจน์ตัวตน Kerberos Version 5 ไคลเอ็นต์ จะได้รับตั๋ว Kerberos Version 5 จากเซิร์ฟเวอร์การรักษาความปลอดภัย DCE หรือ เซิร์ฟเวอร์ Kerboros ตั๋วคือส่วนหนึ่งของ DCE ปัจจุบันของผู้ใช้หรือ credentials โลคัลที่เข้ารหัสสำหรับเซิร์ฟเวอร์ TCP/IP กับฝ่ายที่ต้องการ เชื่อมต่อด้วย daemon บนเซิร์ฟเวอร์ TCP/IP ถอดรหัสตั๋ว การดำเนินการนี้ อนุญาตให้เซิร์ฟเวอร์ TCP/IP ระบุผู้ใช้อย่างแน่นอน ถ้า DCE หรือ principal โลคัลอธิบายในตั๋วได้รับอนุญาตให้เข้าถึง ระบบปฏิบัติการของบัญชีผู้ใช้ของผู้ใช้ การเชื่อมต่อจะดำเนินการ rcmds ที่ปลอดภัยสนับสนุนไคลเอ็นต์และเซิร์ฟเวอร์ Kerberos จากทั้ง Kerberos Version 5 และ DCE

นอกเหนือจากการพิสูจน์ตัวตนไคลเอ็นต์ Kerberos Version 5 จะส่งต่อ credentials ของผู้ใช้ปัจจุบันไปที่เซิร์ฟเวอร์ TCP/IP ถ้า credentials ถูกทำเครื่องหมายเป็นส่งต่อได้ ไคลเอ็นต์จะส่งไปที่ เซิร์ฟเวอร์เพื่อเป็นตั๋วที่ใช้ในการให้สิทธิตั๋ว บนฝั่งเซิร์ฟเวอร์ TCP/IP ถ้าผู้ใช้กำลังสื่อสารกับเซิร์ฟเวอร์การรักษาความปลอดภัย DCE daemon จะอัพเกรดตั๋วที่ใช้ในการให้สิทธิตั๋วเป็น DCE credentials เต็มโดยใช้ คำสั่ง **k5dcecreds**

คำสั่ง **ftp** ให้วิธีการพิสูจน์ตัวตน ที่แตกต่างจาก rcmds ที่ปลอดภัยอื่นๆ โดยใช้วิธีการรักษาความปลอดภัย GSSAPI เพื่อส่งการพิสูจน์ตัวตนระหวางคำสั่ง **ftp** และ **ftpd** daemon การใช้คำสั่งย่อย **clear**, **safe** และ **private** ไคลเอ็นต์ ftp สนับสนุนการเข้ารหัสข้อมูล

ระหว่างระบบปฏิบัติการไคลเอ็นต์และเซิร์ฟเวอร์ คำสั่ง **ftp** อนุญาตการถ่ายโอนหลายไบต์สำหรับการเชื่อมต่อข้อมูลที่เข้ารหัส ค่า มาตรฐานกำหนดเฉพาะการถ่ายโอนไบต์เดียวเท่านั้นสำหรับการเชื่อมต่อข้อมูลที่เข้ารหัส เมื่อเชื่อมต่อกับเครื่องของบุคคลที่สามและใช้การเข้ารหัสข้อมูล คำสั่ง **ftp** จะดำเนินตามข้อจำกัดการถ่ายโอน แบบไบต์เดียว

การตั้งค่าระบบ:

สำหรับ rcmds ที่ปลอดภัยทั้ง วิธีการตั้งค่าในระดับระบบ จะพิจารณาว่าวิธีการพิสูจน์ตัวตนใดที่อนุญาตให้กระทำได้สำหรับระบบนั้น การตั้งค่าจะทำหน้าที่ควบคุมการเชื่อมต่อทั้งขาออกและ ขาเข้า

การตั้งค่าการพิสูจน์ตัวตนประกอบด้วยไลบรารี `libauthm.a` และคำสั่ง **lsauthent** และ **chauthent** ที่จัดให้มีบรรทัดคำสั่งในการเข้าถึงไลบรารีรูทีน **get_auth_methods** และ **set_auth_methods**

วิธีการพิสูจน์ตัวตนเป็นตัวกำหนดวิธีที่ใช้ในการพิสูจน์ตัวตน การเข้าถึงเน็ตเวิร์กของผู้ใช้ ระบบให้การสนับสนุนวิธีการพิสูจน์ตัวตน ต่อไปนี้:

- Kerberos Version 5 เป็นวิธีการที่ใช้ทั่วไปมากสุด เนื่องจากเป็นวิธีพื้นฐานสำหรับ DCE
- Kerberos Version 4 ถูกใช้โดย rcmds ที่ปลอดภัย rlogin, rsh และ rcp เท่านั้น ซึ่งจัดเตรียมเพื่อสนับสนุนความเข้ากันได้กับเวอร์ชันก่อนหน้าเท่านั้น บนระบบ SP ตั๋ว Kerberos Version 4 ไม่ถูกอัพเกรดเป็น DCE credentials

ถ้ามีวิธีการพิสูจน์ตัวตนมากกว่าหนึ่งวิธีถูกตั้งค่าและวิธีแรก ไม่สามารถทำการเชื่อมต่อได้ ไคลเอ็นต์พยายามพิสูจน์ตัวตนโดยใช้วิธีการพิสูจน์ตัวตนที่ถูกตั้งค่าไว้ถัดไป

วิธีการพิสูจน์ตัวตนสามารถตั้งค่าลำดับใดๆ ก็ได้ มีข้อยกเว้น อย่างเดียวคือ AIX มาตรฐานต้อง เป็นวิธีการพิสูจน์ตัวตนสุดท้ายที่ถูกตั้งค่า เนื่องจากไม่มีอ็อพชัน fallback ถ้า AIX มาตรฐาน ไม่ใช่วิธีการพิสูจน์ตัวตนที่ตั้งค่าไว้ จะไม่มีการใช้การพิสูจน์ตัวตนด้วยรหัสผ่าน และการพยายามทำการเชื่อมต่อใดๆ โดยใช้วิธีนี้จะถูกปฏิเสธ

คุณยังสามารถตั้งค่าระบบได้โดยไม่ต้องใช้วิธีการพิสูจน์ตัวตนใดๆ ในกรณีนี้ ระบบปฏิเสธการเชื่อมต่อทั้งหมดที่มาจาก หรือไปยังระบบใดๆ โดยใช้ rcmds ที่ปลอดภัย รวมทั้ง เนื่องจาก Kerberos Version 4 สนับสนุน การใช้กับคำสั่ง **rlogin, rsh** และ **rcp** เท่านั้น ระบบที่ตั้งค่าเพื่อใช้เฉพาะ Kerberos Version 4 จะไม่อนุญาต ให้มีการเชื่อมต่อโดยใช้ telnet หรือ FTP

การตรวจสอบความถูกต้องผู้ใช้ Kerberos Version 5:

วิธีการพิสูจน์ตัวตน Kerberos Version 5 สามารถนำไปใช้ เพื่อตรวจสอบความถูกต้องผู้ใช้

เมื่อใช้วิธีการพิสูจน์ตัวตน Kerberos Version 5 ไคลเอ็นต์ TCP/IP จะรับตั๋วเซอร์วิสที่ข้ารหัสสำหรับเซิร์ฟเวอร์ TCP/IP เมื่อเซิร์ฟเวอร์ถอดรหัสตั๋ว จะมีวิธีการระบุผู้ใช้ อย่างปลอดภัย (โดย DCE หรือ local principal) อย่างไรก็ตาม เซิร์ฟเวอร์ต้องพิจารณาว่า DCE หรือor local principal นี้จะได้รับอนุญาตให้เข้าถึงบัญชีผู้ใช้โลคัล การแม็พ DCE หรือ local principal กับบัญชีผู้ใช้ระบบปฏิบัติการโลคัล ได้รับการจัดการโดยไลบรารีที่แบ่งใช้ `libvaliduser.a` ซึ่งมีรูทีนย่อยเดียว kvalid_user ถ้าต้องการใช้วิธีการแม็พแบบอื่น ผู้ดูแลระบบต้องให้ทางลอก สำหรับไลบรารี `libvaliduser.a`

การตั้งค่า DCE:

ในการใช้ rcmds ที่ปลอดภัย ต้องมีสอง DCE principals อยู่ สำหรับทุกอินเตอร์เฟสเน็ตเวิร์กที่สามารถใช้เชื่อมต่อได้

DCE principals ทั้งสองได้แก่:

```
host/FullInterfaceName
ftp/FullInterfaceName
```

โดยที่ *FullInterfaceName* คือชื่ออินเตอร์เฟสและ โดเมนเนม

การตั้งค่าโลคัล:

ในการใช้ rcmds ที่ปลอดภัย ต้องมีสอง local principals อยู่ สำหรับทุกอินเตอร์เฟสเน็ตเวิร์กที่สามารถใช้เชื่อมต่อได้

local principals ทั้งสองได้แก่:

```
host/FullInterfaceName@Realmname
ftp/FullInterfaceName@Realmname
```

โดยที่ *FullInterfaceName* คือ ชื่ออินเตอร์เฟสและโดเมนเนมและ *RealmName* คือชื่อ ของ local Kerberos Version 5 realm

ดูที่แหล่งข้อมูลต่อไปนี้เพื่อดูข้อมูลที่เกี่ยวข้อง:

- รูทีนย่อย get_auth_method และ set_auth_method ใน *ข้อมูลอ้างอิงด้านเทคนิค: การสื่อสารวอลุ่ม 2*
- คำสั่ง chauthent ใน *ข้อมูลอ้างอิงคำสั่ง วอลุ่ม 1*
- คำสั่ง lsauthent ใน *ข้อมูลอ้างอิงคำสั่ง วอลุ่ม 3*

การพิสูจน์ตัวตนกับระบบปฏิบัติการ AIX โดยใช้ Network Authentication Service หรือเซอร์วิสที่ไม่ใช่ AIX

ก่อนหน้า AIX 6.1, โหลดโมดูล KRB5 ที่จัดการกับการพิสูจน์ตัวตน Kerberos กับสภาวะแวดล้อม Network Authentication Service (NAS) และโหลดโมดูล KRB5A ที่จัดการกับการพิสูจน์ตัวตน Kerberos กับสภาวะแวดล้อมของระบบที่ไม่ใช่ AIX เริ่มต้นด้วย AIX 6.1, โหลดโมดูล KRB5 จัดการกับการพิสูจน์ตัวตน Kerberos ของทั้งสภาวะแวดล้อม Network Authentication Service (NAS) และสภาวะแวดล้อมของระบบที่ไม่ใช่ AIX แอ็ตทริบิวต์ **is_kadmind_compat** ในไฟล์ `etc/`

security/methods.cfg ระบุสภาวะแวดล้อม KRB5 หรือสภาวะแวดล้อม KRB5A ตั้งแต่AIX 7.1 เป็นต้นไป ไม่มีโมดูลโหลด KRB5A แอ็ตทริบิวต์ **is_kadmind_compat** ต้องอยู่ในไฟล์ etc/security/methods.cfg เพื่อระบุสภาวะแวดล้อม KRB5 หรือ KRB5A อย่างใดอย่างหนึ่ง

เมื่อไคลเอ็นต์ Kerberos ถูกตั้งค่าให้พิสูจน์ตัวตนกับ NAS โหลดโมดูล KRB5 จะดำเนินการพิสูจน์ตัวตน Kerberos และการจัดการหลักการ Kerberos โมดูลเปิดใช้งานผู้ดูแลระบบ เพื่อจัดการกับ Kerberos โดยใช้คำสั่งการดูแลระบบผู้ใช้ AIX ในการใช้การจัดการหลัก เซิร์ฟเวอร์ Kerberos ต้องสนับสนุน โปรโตคอลการดูแล kadmin NAS จัดเตรียมส่วนสนับสนุนนี้ผ่าน **kadmind** daemon (เซิร์ฟเวอร์ Kerberos ที่รันบนระบบปฏิบัติการ AIX)

หมายเหตุ: เมื่อคุณกำหนดคอนฟิกไคลเอ็นต์ Kerberos, คุณต้องระบุการพิสูจน์ตัวตนกับ NAS; มิฉะนั้น, ไคลเอ็นต์จะถูกกำหนดคอนฟิกเพื่อพิสูจน์ตัวตน กับเซอร์วิสที่ไม่ใช่ AIX แทน และการจัดการหลักจะไม่พร้อมใช้งาน

เมื่อคุณใช้ Kerberos กับระบบที่ไม่ใช่ AIX, Kerberos principals จะถูกเก็บอยู่บนระบบที่ไม่ใช่ AIX และไม่สามารถจัดการได้จากระบบปฏิบัติการ AIX โดยใช้อินเตอร์เฟสฐานข้อมูล kadmin Kerberos ในกรณีนี้ การจัดการหลักต้องถูกดำเนินงาน แยกต่างหากโดยใช้เครื่องมือการจัดการหลัก Kerberos เครื่องมือ เหล่านี้อาจเป็นส่วนหนึ่งของผลิตภัณฑ์ Kerberos หรือถูกผนวกรวมไว้ใน OS (ตัวอย่างเช่น Windows 2000) เป้าหมายเดิมของการใช้ Kerberos กับระบบที่ไม่ใช่ AIX ได้จัดเตรียมการพิสูจน์ตัวตนกับ Windows 2000 Active Directory โดยที่การจัดการกับ Kerberos principal ถูกดำเนินการโดยใช้เครื่องมือการจัดการแอคเคาต์ Active Directory และ APIs อย่างไรก็ตาม, Kerberos กับระบบที่ไม่ใช่ AIX สามารถใช้กับ KDCs ที่เข้ากันได้โดยที่ไม่สนับสนุนอินเตอร์เฟสการดูแลระบบ Kerberos

การติดตั้งและการตั้งค่าระบบสำหรับการล็อกอินที่รวม Kerberos โดยใช้ IBM NAS:

การนำ IBM Kerberos ไปใช้ของ Network Authentication Services (NAS) มาพร้อมกับ expansion pack

ในการติดตั้งเซิร์ฟเวอร์แพ็กเกจ Kerberos Version 5 ให้ติดตั้ง ชุดไฟล์ krb5.server.rte โดยการรันคำสั่ง ต่อไปนี้:

```
installp -aqXYgd . krb5.server
```

ถ้า เครื่องที่ตั้งค่าเป็นเซิร์ฟเวอร์ Kerberos จะถูกใช้เป็นไคลเอ็นต์ Kerberos ด้วยเช่นกัน ให้ติดตั้งแพ็กเกจ Kerberos KRB5 ทั้งหมด

DCE ยังมีชุดของไคลเอ็นต์ยูทิลิตี Kerberos ที่มีชื่อเดียวกับ ยูทิลิตี Kerberos เพื่อหลีกเลี่ยงการใช้ namespace ชนกันระหว่างคำสั่ง DCE และ Kerberos (นั่นคือ ระหว่างคำสั่ง **klist, kinit** และ **kdestroy**) คำสั่ง Kerberos ถูก ติดตั้งในไดเร็กทอรี /usr/krb5/bin และ /usr/krb5/sbin

ในการ รันคำสั่ง Kerberos คุณต้องระบุชื่อพาธคำสั่ง แบบเต็มยกเว้นคุณจะเพิ่มไดเร็กทอรี Kerberos ลงในนิยาม PATH ของคุณดังนี้:

```
export PATH=$PATH:/usr/krb5/sbin:/usr/krb5/bin
```

หมายเหตุ: Java14 SDK ยังติดตั้งคำสั่ง **kinit** และ อาจมาก่อนคำสั่ง **kinit** อื่นๆ ในตัวแปรสภาวะแวดล้อม PATH ถ้าจำเป็นต้องใช้คำสั่ง Network Authentication Service แทนคำสั่ง ของโปรแกรม Java14 **kinit** ให้ย้ายโปรแกรม Java14 **kinit** ไปยังตำแหน่งอื่นในนิยาม PATH ของคุณ

เอกสารคู่มือ Network Authentication Services มีอยู่ในแพ็กเกจ krb5.doc.*lang*.pdf|html โดยที่ *lang* แทนภาษาที่สนับสนุน

ระบบปฏิบัติการ AIX มีสองโมดูลฐานข้อมูลที่พร้อมใช้งานเพื่อจัดรูปแบบโหลดโมดูลผสม : LDAP and BUILTIN โมดูล LDAP ถูกใช้เพื่อเข้าถึงข้อมูลที่เก็บอยู่บนรีจิสทรี LDAP (ไดเร็กทอรี) และโมดูล BUILTIN ถูกใช้เพื่อเข้าถึงข้อมูลที่เก็บอยู่บนไฟล์รีจิสทรี (ระบบไฟล์โลคัล) โหลดโมดูลผสมที่สร้างขึ้นโดยปกติชื่อ KRB5files หรือ KRB5LDAP ชื่อเหล่านี้บ่งชี้ว่า KRB5 ถูกใช้สำหรับการพิสูจน์ตัวตนและไฟล์โลคัล หรือสำหรับ LDAP อย่างใดอย่างหนึ่ง

Network Authentication Service ยังสนับสนุนการเก็บข้อมูล Kerberos ในระบบไฟล์โลคัล (Kerberos Legacy database) หรือ LDAP โดยมีการตั้งค่าได้สี่รูปแบบ:

- KRB5files ที่มีข้อมูลเซิร์ฟเวอร์ Kerberos เก็บในฐานข้อมูล Kerberos Legacy
- KRB5files ที่มีข้อมูลเซิร์ฟเวอร์ Kerberos เก็บในฐานข้อมูล Kerberos LDAP
- KRB5LDAP ที่มีข้อมูลเซิร์ฟเวอร์ Kerberos เก็บในฐานข้อมูล Kerberos Legacy
- KRB5LDAP ที่มีข้อมูลเซิร์ฟเวอร์ Kerberos เก็บในฐานข้อมูล Kerberos LDAP

เมื่อ LDAP คือกลไกหน่วยเก็บสำหรับเก็บ Kerberos หรือข้อมูลผู้ใช้และกลุ่ม AIX, กำหนดคอนฟิก LDAP ก่อนที่คุณจะเรียกทำงานคำสั่งคอนฟิกูเรชัน Kerberos หลังจากคุณตั้งค่า LDAP ให้ใช้คำสั่ง **mkkrb5srv** เพื่อตั้งค่าเซิร์ฟเวอร์ Kerberos

การตั้งค่าเซิร์ฟเวอร์ Network Authentication Service ที่มี สื่อบันทึกฐานข้อมูลเก่า:

คุณสามารถตั้งค่า Network Authentication Service KDC และเซิร์ฟเวอร์ การดูแลที่มีฐานข้อมูล Kerberos เก่าและตั้งค่าเซิร์ฟเวอร์ Network Authentication Service โดยใช้คำสั่ง **mkkrb5srv**

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการใช้คำสั่ง **mkkrb5srv** ดูที่คำสั่ง **mkkrb5srv**

หมายเหตุ: ห้าม ติดตั้งทั้งเซิร์ฟเวอร์ซอฟต์แวร์ DCE และ Kerberos บนระบบฟิสิคัล เดียวกัน ถ้าคุณต้องทำเช่นนั้น หมายเลขพอร์ตอินเตอร์เน็ตที่ดำเนินงานค่าดีฟอลต์ ต้องถูกเปลี่ยนสำหรับไคลเอ็นต์หรือเซิร์ฟเวอร์ DCE หรือสำหรับ ไคลเอ็นต์หรือเซิร์ฟเวอร์ Kerberos อย่างใดอย่างหนึ่ง ไม่ว่ากรณีใด การเปลี่ยนแปลงเช่นนั้นสามารถส่งผล ต่อการทำงานร่วมกันกับการนำใช้ DCE และ Kerberos ที่มีอยู่แล้วใน สภาวะแวดล้อมของคุณ สำหรับข้อมูลเกี่ยวกับการมีอยู่ร่วมกันของ DCE และ Kerberos โปรดอ้างอิงเอกสารคู่มือ Network Authentication Services

Kerberos Version 5 ถูกตั้งค่าเพื่อปฏิเสธการร้องขอตั๋วจากโฮสต์ใดๆ ที่ นาฬิกาไม่อยู่ภายในความเบี่ยงเบนของนาฬิกาสูงสุดที่ระบุของ KDC ค่าดีฟอลต์ สำหรับความเบี่ยงเบนนาฬิกาสูงสุดคือ 300 วินาที (ห้านาที) Kerberos จำเป็นต้องมีการตั้งค่าการซิงโครไนซ์เวลาในรูปแบบใดรูปแบบหนึ่ง ระหว่างเซิร์ฟเวอร์และไคลเอ็นต์ ขอแนะนำให้คุณใช้ **xntpd** หรือ **timed** daemons สำหรับ การซิงโครไนซ์เวลา ในการใช้ **timed** daemon ให้ทำต่อไปนี้:

1. ตั้งค่าเซิร์ฟเวอร์ KDC เป็นเซิร์ฟเวอร์เวลาโดยการเริ่มทำงาน **timed** daemon ดังนี้:

   ```
   timed -M
   ```

2. เริ่มทำงาน **timed** daemon บนแต่ละไคลเอ็นต์ Kerberos ดังนี้:

   ```
   timed -t
   ```

3. ในการตั้งค่าเซิร์ฟเวอร์ Kerberos KDC และ kadmin ให้รันคำสั่ง **mkkrb5srv** ตัวอย่าง ในการตั้งค่า Kerberos สำหรับ `MYREALM` realm, เซิร์ฟเวอร์ `sundial` และโดเมน `xyz.com` ให้รันคำสั่งต่อไปนี้:

   ```
   mkkrb5srv -r MYREALM -s sundial.xyz.com -d xyz.com -a admin/admin
   ```

   รอ สักครู่เพื่อให้คำสั่ง **kadmind** และ **krb5kdc** เริ่มทำงานจากไฟล์ `/etc/inittab`

Network Authentication Service ใช้พื้นที่ในระบบไฟล์ **/var** เพื่อเก็บข้อมูล ข้อมูลนี้ประกอบด้วยฐานข้อมูล บันทึกการทำงาน และ แคชไฟล์ credential ของผู้ใช้ที่ได้รับการพิสูจน์ตัวตน ขนาดของไฟล์ เหล่านี้สามารถเพิ่มขึ้นเมื่อเวลาผ่านไป ทำให้แน่ใจว่าระบบไฟล์ **/var** มีพื้นที่ว่างเพียงพอสำหรับเก็บข้อมูลนี้โดยการมอนิเตอร์ขนาดพื้นที่ว่าง เป็นประจำ

ต่อไปนี้คือคำสั่ง **mkkrb5srv** ทั่วไป:

```
mkkrb5srv -r Realm_Name -s KDC_Server -d Domain_Name -a Admin_Name
```

ค่า ตัวแปรใน ตารางที่ 15ถูกใช้ในตัวอย่างต่อไปนี้ที่แสดงวิธีตั้งค่าเซิร์ฟเวอร์ Network Authentication Service ที่มีฐานข้อมูลเก่า

ตารางที่ 15. ชื่อตัวแปรคำสั่ง **mkkrb5srv**

ชื่อตัวแปร	ค่าตัวแปร
Realm Name	MYREALM
KDC Server	kdcsrv.austin.ibm.com
Domain Name	austin.ibm.com
Administrator Name	admin/admin

ถ้ามีการตั้งค่าเซิร์ฟเวอร์ Kerberos อยู่แล้ว คุณสามารถลบออกได้โดยการใช้คำสั่ง **mkkrb5srv –U** หรือ **unconfig.krb5**

ข้อควรสนใจ: ถ้าคุณจำเป็นต้องเก็บการตั้งค่าเซิร์ฟเวอร์ Kerberos ที่มีอยู่แล้ว ไม่ต้องดำเนินขั้นตอนต่อไปนี้

โพรซีเดอร์ต่อไปนี้ เป็นตัวอย่างวิธีตั้งค่าเซิร์ฟเวอร์ Network Authentication Service ที่มีฐานข้อมูลเก่า

1. ป้อนคำสั่งต่อไปนี้:

   ```
   mkkrb5srv -r MYREALM -s kdcsrv.austin.ibm.com -d austin.ibm.com -a admin/admin
   ```

   หลังจาก ป้อนนี้ คุณจะได้รับพร้อมต์เพื่อป้อนรหัสผ่านฐานข้อมูลมาสเตอร์

   เนื่องจาก Network Authentication Service ไม่สนับสนุนการตั้งค่าที่ KDC และเซิร์ฟเวอร์การดูแลอยู่บนคนละโฮสต์ จึงใช้โฮสต์โลคัล สำหรับ KDC และเซิร์ฟเวอร์การดูแลทั้งสอง ข้าม ข้อความแสดงความผิดพลาดต่อไปนี้หากแสดงขึ้นมา: `The -s option is not supported.`
2. ป้อนรหัสผ่านฐานข้อมูลมาสเตอร์เมื่อคุณได้รับพร้อมต์
3. ป้อนรหัสผ่านหลักการการดูแลเมื่อคุณได้รับพร้อมต์

   หลังจาก คุณป้อนรหัสผ่านหลักการการจัดการแล้ว คำสั่ง **mkkrb5srv** จะเริ่มทำงาน **kadmind** และ **krb5kdc** daemons จากพาธไฟล์ `/etc/inittab` กระบวนการนี้ อาจใช้เวลาหลานาที
4. ตรวจสอบรายการในไฟล์ `/etc/inittab` โดยการรันคำสั่งต่อไปนี้:

   ```
   lsitab krb5kdc
   lsitab kadm
   ```
5. ตรวจสอบว่าเซิร์ฟเวอร์ KDC และ kadmind ได้เริ่มทำงานแล้วโดยการป้อน คำสั่งต่อไปนี้:

   ```
   ps -ef | grep -v grep | grep krb5
   ```

คำสั่ง **mkkrb5srv** สร้าง KDC มาสเตอร์และเซิร์ฟเวอร์การดูแล kadmind สำหรับ Kerberos realm (MYREALM) รวมทั้งสร้างไฟล์คอนฟิกูเรชัน เตรียมข้อมูลเบื้องต้น ของฐานข้อมูล principal และเริ่มทำงานเซิร์ฟเวอร์ KDC และ kadmind

การรัน คำสั่ง **mkkrb5srv** ส่งผลให้เกิดการดำเนินการ ต่อไปนี้:

1. สร้างไฟล์ `/etc/krb5/krb5.conf` ค่า สำหรับ realm name, Kerberos admin server และ domain name ถูกตั้งค่าตามที่ระบุบนบรรทัดคำสั่ง ไฟล์ `/etc/krb5/krb5.conf` ยังตั้งค่าพาธสำหรับล็อกไฟล์ default_keytab_name, kdc และ admin_server
2. สร้างไฟล์ `/var/krb5/krb5kdc/kdc.conf` ไฟล์ `/var/krb5/krb5kdc/kdc.conf` ตั้งค่า สำหรับตัวแปร *kdc_ports*, *kadmin_port*, *max_life*, *max_renewable_life*, *master_key_type* และ *supported_enctypes* ไฟล์นี้ยัง ตั้งค่าพาธสำหรับตัวแปร *database_name*, *admin_keytab*, *acl_file*, *dict_file* และ *key_stash_file*
3. สร้างไฟล์ `/var/krb5/krb5kdc/kadm5.acl` ตั้งค่าควบคุมการเข้าใช้สำหรับ admin, root และ host principals
4. สร้างฐานข้อมูลและหนึ่ง admin principal คุณจะถูกขอให้ตั้งค่ามาสเตอร์คีย์ Kerberos และเพื่อตั้งชื่อและตั้งค่า รหัสผ่านสำหรับ Kerberos administrative principal identity เพื่อวัตถุประสงค์ในการ กู้คืนจากความเสียหาย เป็นสิ่งสำคัญที่มาสเตอร์คีย์และ administrative principal identity และรหัสผ่านต้องถูกเก็บไว้อย่างปลอดภัย

สำหรับข้อมูลเพิ่มเติม โปรดดูที่ "ตัวอย่างการรัน" ในหน้า 327 และ "ข้อความแสดงความผิดพลาดและการดำเนินการแก้ไข" ในหน้า 326

การตั้งค่าเซิร์ฟเวอร์ Kerberos กับสื่อบันทึก LDAP:

คุณสามารถตั้งค่า Network Authentication Service kadmin และ เซิร์ฟเวอร์ KDC สำหรับการล็อกอินที่ผนวกรวม Kerberos โดยใช้คำสั่ง **mkkrb5srv**

ค่าตัวแปรใน ตารางที่ 16ถูก ใช้ในตัวอย่างต่อไปนี้เพื่อแสดงวิธีตั้งค่าคอมโพเนนต์เซิร์ฟเวอร์ Network Authentication Service กับสื่อบันทึก LDAP โดยใช้คำสั่ง **mkkrb5srv**

ตารางที่ 16. ชื่อตัวแปรคำสั่ง **mkrb5srv**

ชื่อตัวแปร	ค่าตัวแปร
Realm_Name	MYREALM
KDC_Server	kdcsrv.austin.ibm.com
Domain_Name	austin.ibm.com
Admin_Name	admin/admin
เซิร์ฟเวอร์ LDAP	kdcsrv.austin.ibm.com
ชื่อผู้ดูแลระบบ LDAP	cn=root
รหัสผ่านผู้ดูแลระบบ LDAP	secret

โพรซีเดอร์ต่อไปนี้เป็นตัวอย่างของวิธีตั้งค่า คอมโพเนนต์เซิร์ฟเวอร์ Network Authentication Service กับสื่อบันทึก LDAP โดยใช้คำสั่ง **mkkrb5srv**

1. รันคำสั่งต่อไปนี้:

```
mkkrb5srv -r MYREALM -s kdcsrv.austin.ibm.com -d austin.ibm.com\
-a admin/admin -l kdcsrv.austin.ibm.com -u cn=root -p secret
```

2. ตรวจสอบว่าเซิร์ฟเวอร์ KDC และ kadmind ได้เริ่มทำงานโดยการรัน คำสั่งต่อไปนี้:

```
ps -ef | grep -v grep | grep krb5
```

การรันคำสั่ง **mkkrb5srv** ด้วย LDAP สร้างผลลัพธ์ที่คล้ายกับการรันคำสั่งด้วย การตั้งค่าฐานข้อมูลเก่า อย่างไรก็ตาม เมื่อใช้ LDAP ฐานข้อมูล จะไม่ถูกสร้างขึ้นบนระบบไฟล์โลคัล แต่สร้างไฟล์ `.kdc_ldap_data` ขึ้นในไฟล์ `/var/krb5/krb5kdc` แทน เพื่อเก็บ ข้อมูลเกี่ยวกับ LDAP

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการใช้งาน ดูที่คำสั่ง **mkkrb5srv**

การตั้งค่าการล็อกอินที่รวม Kerberos:

หลังจากการติดตั้ง Kerberos เสร็จสมบูรณ์ คุณต้องตั้งค่า ระบบเพื่อใช้ Kerberos เป็นแนวทางหลักในการพิสูจน์ตัวตนผู้ใช้

ในการตั้งค่าระบบเพื่อใช้ Kerberos เป็นแนวทางหลัก ของการพิสูจน์ตัวตนผู้ใช้ให้รันคำสั่ง **mkkrb5clnt** ด้วยพารามิเตอร์ต่อไปนี้:

```
mkkrb5clnt -c KDC -r realm -a admin -s server -d domain -A -i database -K -T
```

ค่าตัวแปร ใน ตารางที่ 17ถูกใช้ในตัวอย่างต่อไปนี้สำหรับวิธีการคอนฟิกระบบสำหรับ Kerberos ที่รวมล็อกอินเข้ากับระบบไฟล์โลคัลเป็นที่เก็บ AIX user/group

ตารางที่ 17. ชื่อตัวแปรคำสั่ง **mkkrb5clnt**

ชื่อตัวแปร	ค่าตัวแปร
Realm Name	MYREALM
KDC Server	kdcsrv.austin.ibm.com
Domain Name	austin.ibm.com
Administration Server	kdcsrv.austin.ibm.com
Administrator Name	admin/admin
ฐานข้อมูล AIX User/Group	ไฟล์

คำสั่งต่อไปนี้คือตัวอย่างวิธีการกำหนดคอนฟิกระบบ สำหรับ Kerberos ที่รวมล็อกอินเข้ากับระบบไฟล์โลคัลเป็นที่เก็บผู้ใช้/กลุ่ม AIX

รัน คำสั่งต่อไปนี้:

```
mkkrb5clnt -r MYREALM -c kdcsrv.austin.ibm.com -s kdcsrv.austin.ibm.com\
-a admin/admin -d austin.ibm.com -A -i files -K -T
```

ตัวอย่าง ก่อนหน้านี้ให้ผลลัพธ์การดำเนินการต่อไปนี้:

1. คำสั่งสร้างไฟล์ `/etc/krb5/krb5.conf` ค่าสำหรับ realm name, Kerberos administration server และ domain name ถูกตั้งค่าดังที่ระบุบนบรรทัดคำสั่ง พาธสำหรับล็อกไฟล์ `default_keytab_name`, `kdc` และ `kadmin` ถูกอัพเดตเช่นกัน
2. แฟล็ก **-i** กำหนดคอนฟิกการล็อกอินที่รวมแบบสมบูรณ์ ฐานข้อมูล ที่ป้อนคือตำแหน่งที่ข้อมูล identification ผู้ใช้ AIX ถูกเก็บ ซึ่งต่างจากสื่อบันทึกหลักการ Kerberos สื่อบันทึกที่ใช้เก็บหลักการ Kerberos ถูกตั้งค่าระหว่าง การตั้งค่า Kerberos
3. แฟล็ก **-K** กำหนดคอนฟิก Kerberos เป็น scheme การพิสูจน์ตัวตน ดีฟอลต์ ค่านี้อนุญาตให้ผู้ใช้ได้รับการพิสูจน์ตัวตนด้วย Kerberos ในตอนล็อกอิน
4. แฟล็ก **-A** เพิ่มรายการใน Kerberos Database เพื่อให้ root เป็นผู้ใช้ระดับผู้ดูแลสำหรับ Kerberos

5. แฟล็ก **-T** ได้รับตั๋วการให้สิทธิตั๋วผู้ดูแลเซิร์ฟเวอร์

หมายเหตุ: ห้ามใช้อ็อพชัน **-D** ในคำสั่ง **mkkrb5clnt** เพื่อกำหนดคอนฟิกสภาวะแวดล้อมไคลเอ็นต์ Kerberos สำหรับการพิสูจน์ตัวตนกับ IBM Network Authentication Service (NAS) ถ้าคุณไม่ได้ระบุอ็อพชัน **-D** ในคำสั่ง **mkkrb5clnt,** แอ็ตทริบิวต์ **is_kadmind_compat** ไม่ได้รวมอยู่ในไฟล์ /usr/lib/security/methods.cfg และสภาวะแวดล้อมไคลเอ็นต์ Kerberos ถูกกำหนดคอฟนิกไว้สำหรับการพิสูจน์ตัวตน กับ IBM NAS

ตรวจสอบคอนฟิกูเรชันโดยการพิจารณาไฟล์ /etc/krb5/krb5.conf ต่อไปนี้คือตัวอย่างของไฟล์ /etc/krb5/krb5.conf บนเครื่องไคลเอ็นต์:

```
[libdefaults]
     default_realm = MYREALM
     default_keytab_name = FILE:/etc/krb5/krb5.keytab
     default_tkt_enctypes = des3-cbc-sha1 arcfour-hmac aes256-cts des-cbc-md5 des-cbc-crc
     default_tgs_enctypes = des3-cbc-sha1 arcfour-hmac aes256-cts des-cbc-md5 des-cbc-crc
[realms]
     MYREALM = {
        kdc = kdcsrv.austin.ibm.com:88
        admin_server = kdcsrv.austin.ibm.com:749
        default_domain = austin.ibm.com
     }
[domain_realm]
    .austin.ibm.com = MYREALM
    kdcsrv.austin.ibm.com = MYREALM
[logging]
     kdc = FILE:/var/krb5/log/krb5kdc.log
     admin_server = FILE:/var/krb5/log/kadmin.log
     default = FILE:/var/krb5/log/krb5lib.log
```

หมายเหตุ: ถ้า LDAP ถูกใช้เป็นสื่อบันทึกหลักการ Kerberos ดังนั้นไฟล์ krb5.conf จะมีบรรทัดต่อไปนี้ภายใต้ [realms] stanza:

```
vdb_plugin_lib = /usr/lib/libkrb5ldplug.a
```

ถ้า ระบบถูกติดตั้งในตำแหน่งในโดเมน DNS อื่นที่ต่างจาก KDC การดำเนินการเพิ่มต่อไปนี้ต้องได้รับการดำเนินการ:

1. แก้ไขไฟล์ /etc/krb5/krb5.conf และเพิ่ม อีกรายการหลัง [domain realm]
2. แม็พโดเมนที่แตกต่างกับ realm ของคุณ

ตัวอย่าง ถ้าคุณต้องการรวมไคลเอ็นต์ที่ อยู่ในโดเมน abc.xyz.com ใน MYREALM realm ของคุณ ให้แก้ไขไฟล์ /etc/krb5/krb5.conf ดังนี้:

```
[domain realm]
     .austin.ibm.com = MYREALM
     .raleigh.ibm.com = MYREALM
```

เมื่อการตั้งค่า Network Authentication Service เสร็จเรียบร้อย กระบวนการล็อกอินเข้าสู่ระบบปฏิบัติการยังคงไม่เปลี่ยนแปลง หลัง การล็อกอินสำเร็จ ผู้ใช้จะมีตั๋วการให้สิทธิตั๋ว Kerberos ที่เชื่อมโยงกับกระบวนการที่กำลังทำงานของตน ตัวแปรสภาวะแวดล้อม $KRB5CCNAME ของผู้ใช้ชี้ไปที่ตั๋วการให้สิทธิตั๋ว ในการตรวจสอบ ว่าล็อกอินสำเร็จ และผู้ใช้มีตั๋วการให้สิทธิตั๋ว ให้ใช้คำสั่ง **klist**

หมายเหตุ: เมื่อคุณรันคำสั่ง **mkkrb5clnt** stanza ต่อไปนี้ จะถูกเพิ่มในไฟล์ methods.cfg

```
KRB5:
        program = /usr/lib/security/KRB5
        program_64 = /usr/lib/security/KRB5_64
        options = is_kadmind_compat=yes

KRB5files:
        options = db=BUILTIN,auth=KRB5
```

สำหรับ ข้อมูลเพิ่มเติมเกี่ยวกับ:

- คำสั่ง **mkkrb5clnt** ดูที่คำสั่ง **mkkrb5clnt**
- ไฟล์ `methods.cfg` ดูที่ไฟล์ `methods.cfg`

ข้อความแสดงความผิดพลาดและการดำเนินการแก้ไข:

ข้อผิดพลาดที่สามารถเกิดขึ้นเมื่อใช้คำสั่ง **mkkrb5srv** มีดังต่อไปนี้:

- ถ้าไฟล์ `krb5.conf`, `kdc.conf` หรือ `kadm5.acl` มีอยู่แล้ว คำสั่ง **mkkrb5srv** จะไม่แก้ไขค่า คุณจะได้รับข้อความที่แจ้งว่าไฟล์ มีอยู่แล้ว ค่าใดๆ ของการตั้งค่าสามารถเปลี่ยนแปลงได้โดย การแก้ไขไฟล์ `krb5.conf`, `kdc.conf` หรือ `kadm5.acl`
- ถ้าคุณพิมพ์บางอย่างผิด และไม่มีฐานข้อมูลถูกสร้างขึ้น ให้ลบ ไฟล์คอนฟิกูเรชันที่ถูกสร้างและรันคำสั่งอีกครั้ง
- ถ้ามีความไม่สอดคล้องกันระหว่างฐานข้อมูลและค่าคอนฟิกูเรชัน ให้ลบฐานข้อมูลออกจากไดเร็กทอรี `/var/krb5/krb5kdc/*` และรันคำสั่งอีกครั้ง
- ทำให้แน่ใจว่า **kadmind** และ **krb5kdc** daemons เริ่มทำงานบนเครื่องของคุณ ใช้คำสั่ง **ps** เพื่อตรวจสอบว่า daemons กำลังทำงาน ถ้า daemons เหล่านี้ยังไม่ เริ่มทำงาน ให้ตรวจสอบล็อกไฟล์

ข้อผิดพลาดที่สามารถเกิดขึ้นเมื่อใช้คำสั่ง **mkkrb5clnt** มีดังต่อไปนี้:

- ค่าที่ไม่ถูกต้องสำหรับ `krb5.conf` สามารถแก้ไขได้โดยการแก้ไขไฟล์ `/etc/krb5/krb5.conf`
- ค่าที่ไม่ถูกต้องสำหรับแฟล็ก **-i** สามารถแก้ไขได้โดยการแก้ไข ไฟล์ `/usr/lib/security/methods.cfg`

การกำจัดการขึ้นต่อกันบน kadmind Daemon ระหว่างการพิสูจน์ตัวตนที่ไม่ใช่ KRB5: โมดูลโหลด KRB5 ทำให้เกิดความล่าช้าเมื่อไม่มี kadmind daemon และเมื่อใช้กลไกการตรวจสอบที่ไม่ใช่ KRB5 เช่น single sign-on (SSO) การขึ้นต่อกันนี้สามารถกำจัดได้โดยการตั้งค่าพารามิเตอร์ `kadmind_timeout` ในไฟล์ **methods.cfg**

ค่าที่เป็นไปได้คือ `kadmind_timeout=<seconds>`, โดยที่วินาทีต้องมากกว่า 0

เมื่อโมดูลโหลด KRB5 พยายามเชื่อมต่อกับเซิร์ฟเวอร์ kadmind ที่หยุดทำงานชั่วคราว จะเกิดไทม์เอาต์ transmission control protocol (TCP) พารามิเตอร์ `kadmind_timeout` จะป้องกันการล่าช้าเพิ่มเติม หลังจากไทม์เอาต์ TCP ครั้งแรก พารามิเตอร์ `kadmind_timeout` ระบุหน้าต่างเวลาสำหรับโมดูลโหลด KRB5 เพื่อพยายามเชื่อมต่อ kadmind อีกครั้งหลังจากไทม์เอาต์ tcp ครั้งแรก เมื่อเซิร์ฟเวอร์ kadmind กำลังทำงาน ลักษณะดีฟอลต์จะยังคงมีผลใช้อยู่

โดยค่าดีฟอลต์ `kadmind_timeout` ถูกปิดใช้งาน เพื่อเปิดใช้งานพารามิเตอร์ `kadmind_timeout` เปลี่ยนแปลงไฟล์`methods.cfg` ดังนี้:

```
KRB5:
        program = /usr/lib/security/KRB5
        options = kadmind_timeout=300
KRB5files:
        options = db=BUILTIN,auth=KRB5
```

ไฟล์ที่ถูกสร้าง:

คำสั่ง mkkrb5srv สร้างไฟล์ต่อไปนี้:

- /etc/krb5/krb5.conf
- /var/krb5/krb5kdc/kadm5.acl
- /var/krb5/krb5kdc/kdc.conf

คำสั่ง mkkrb5clnt สร้างไฟล์ ต่อไปนี้:

- /etc/krb5/krb5.conf

อ็อพชัน mkkrb5clnt -i *files* เพิ่ม stanza ต่อไปนี้ในไฟล์ /usr/lib/security/methods.cfg:

```
KRB5:
  program =
  options =
KRB5files:
  options =
```

ตัวอย่างการรัน:

ส่วนนี้แสดงตัวอย่างจากตัวอย่างการรัน

ต่อไปนี้คือตัวอย่างของคำสั่ง mkkrb5srv:

```
# mkkrb5srv -r MYREALM -s sundial.xyz.com -d xyz.com -a admin/admin
```

เอาต์พุตคล้ายกับที่แสดงต่อไปนี้:

```
  Fileset                      Level  State      Description
  ----------------------------------------------------------------------------
Path: /usr/lib/objrepos
  krb5.server.rte            1.3.0.0  COMMITTED  Network Authentication Service
                                                 Server

Path: /etc/objrepos
  krb5.server.rte            1.3.0.0  COMMITTED  Network Authentication Service
                                                 Server

The -s option is not supported.
The administration server will be the local host.
Initializing configuration...
Creating /etc/krb5/krb5.conf...
Creating /var/krb5/krb5kdc/kdc.conf...
Creating database files...
Initializing database '/var/krb5/krb5kdc/principal' for realm 'MYREALM'
master key name 'K/M@MYREALM'
You will be prompted for the database Master Password.
It is important that you NOT FORGET this password.
Enter database Master Password:
Re-enter database Master Password to verify:
WARNING: no policy specified for admin/admin@MYREALM;
  defaulting to no policy. Note that policy may be overridden by
  ACL restrictions.
```

```
Enter password for principal "admin/admin@MYREALM":
Re-enter password for principal "admin/admin@MYREALM":
Principal "admin/admin@MYREALM" created.
Creating keytable...
Creating /var/krb5/krb5kdc/kadm5.acl...
Starting krb5kdc...
krb5kdc was started successfully.
Starting kadmind...
kadmind was started successfully.
The command completed successfully.
Restarting kadmind and krb5kdc
```

ต่อไปนี้คือตัวอย่างของคำสั่ง mkkrb5clnt:

```
mkkrb5clnt -r MYREALM -c sundial.xyz.com -s sundial.xyz.com \
           -a admin/admin -d xyz.com -i files -K -T -A
```

เอาต์พุตคล้ายกับที่แสดงต่อไปนี้:

```
Initializing configuration...
Creating /etc/krb5/krb5.conf...
The command completed successfully.
Password for admin/admin@MYREALM:
Configuring fully integrated login
Authenticating as principal admin/admin with existing credentials.
WARNING: no policy specified for host/diana.xyz.com@MYREALM;
  defaulting to no policy. Note that policy may be overridden by
  ACL restrictions.
Principal "host/diana.xyz.com@MYREALM" created.

Administration credentials NOT DESTROYED.
Authenticating as principal admin/admin with existing credentials.

Administration credentials NOT DESTROYED.
Authenticating as principal admin/admin with existing credentials.
Principal "kadmin/admin@MYREALM" modified.

Administration credentials NOT DESTROYED.
Configuring Kerberos as the default authentication scheme
Making root a Kerberos administrator
Authenticating as principal admin/admin with existing credentials.
WARNING: no policy specified for root/diana.xyz.com@MYREALM;
  defaulting to no policy. Note that policy may be overridden by
  ACL restrictions.
Enter password for principal "root/diana.xyz.com@MYREALM":
Re-enter password for principal "root/diana.xyz.com@MYREALM":
Principal "root/diana.xyz.com@MYREALM" created.

Administration credentials NOT DESTROYED.
Cleaning administrator credentials and exiting.
```

การกำจัดการขึ้นต่อกันบน kadmind daemon ระหว่างการพิสูจน์ตัวตน:

โหลดโมดูล KRB5 อาจล้มเหลวในการพิสูจน์ตัวตนเมื่อ **kadmind** daemon ไม่มีอยู่ การขึ้นต่อกันนี้สามารถกำจัดได้โดยการตั้งค่าพารามิเตอร์ *kadmind* ในไฟล์ methods.cfg

ค่าที่เป็นไปได้คือ kadmind=no หรือ kadmind=false สำหรับ การปิดใช้งานการค้นหา **kadmind** และ kadmind=yes หรือ kadmind=true สำหรับ การปิดใช้งานการค้นหา **kadmind** (ค่าดีฟอลต์คือ yes) เมื่ออ็อพชันนี้ ถูกตั้งค่าเป็น no **kadmind** daemon จะไม่ถูกติดต่อระหว่าง การพิสูจน์ตัวตน ดังนั้น ผู้ใช้สามารถล็อกเข้าสู่ระบบได้ไม่ว่า จัดสถานะของ **kadmind** daemon เป็นค่าใด ซึ่งผู้ใช้ เพียงป้อนรหัสผ่านที่ถูกต้องเมื่อระบบพร้อมต์ อย่างไรก็ตาม, คำสั่งการดูแลระบบผู้ใช้ AIX เช่น **mkuser**, **chuser**, หรือ **rmuser** จะไม่ทำงานกับ Kerberos การดูแลระบบที่รวมผู้ใช้หาก daemon ไม่พร้อมใช้งาน (ตัวอย่างเช่น, daemon ไม่ทำงานหรือไม่สามารถเข้าถึงเครื่องได้)

ค่าดีฟอลต์สำหรับพารามิเตอร์ *kadmind* คือ yes นี้หมายความว่าการค้นหา **kadmind** ถูก ดำเนินการระหว่างะการพิสูจน์ตัวตน ในกรณีดีฟอลต์ ถ้า daemon ไม่พร้อมใช้งาน การพิสูจน์ตัวตนอาจต้องใช้เวลานานขึ้น

ในการ ปิดใช้งานการตรวจสอบ **kadmind** daemon ระหว่างการพิสูจน์ตัวตน ให้แก้ไข stanzas ในไฟล์ methods.cfg ดังนี้:

```
KRB5:
        program = /usr/lib/security/KRB5
        options = kadmind=no

KRB5files:
        options = db=BUILTIN,auth=KRB5
```

เมื่อ **kadmind** daemon ไม่พร้อมใช้งาน ผู้ใช้ root จะไม่สามารถเปลี่ยนรหัสผ่านผู้ใช้ในสถานการณ์เช่น ลืมรหัสผ่าน คุณต้องทำให้ **kadmind** daemon พร้อมใช้งาน และ, ถ้าผู้ใช้เลือกที่จะป้อนชื่อ Kerberos ที่พร้อมต์ ล็อกอิน, ชื่อหลักของชื่อจะถูกตัดปลายตามข้อจำกัดความยาวชื่อผู้ใช้ AIX ชื่อที่ถูกตัดปลายนี้จะถูกใช้สำหรับการดึงข้อมูล AIX user identification (ตัวอย่างเช่น, เพื่อดึงข้อมูลค่าโฮมไดเร็กทอรี ของคุณ)

ถ้า **kadmind** daemon ไม่พร้อมใช้งาน (daemon ไม่ทำงานหรือไม่สามารถเข้าถึงได้) คำสั่ง **mkuser** จะให้ข้อผิดพลาดต่อไปนี้:

```
3004-694 Error adding "krb5user": You do not have permission.
```

ถ้า พารามิเตอร์ *kadmind* ถูกตั้งเป็น no หรือ **kadmind** daemon ไม่สามารถเข้าถึงได้ ระบบไม่สามารถตรวจสอบความถูกต้องในการมีอยู่ของ principal ในฐานข้อมูล Kerberos ดังนั้นระบบจะไม่เรียกแอ็ตทริบิวต์ที่เกี่ยวข้องกับ Kerberos ออกมา สถานการณ์นี้ทำให้ผลลัพธ์ไม่สมบูรณ์หรือไม่ถูกต้อง ตัวอย่าง คำสั่ง **lsuser** อาจไม่รายงาน ผู้ใช้ใดๆ สำหรับเคียวรี ALL

นอกจากนั้น คำสั่ง **chuser** จะจัดการเฉพาะแอ็ตทริบิวต์ที่เกี่ยวกับ AIX และแอ็ตทริบิวต์ที่ไม่เกี่ยวกับ Kerberos คำสั่ง **rmuser** จะไม่ลบ Kerberos principal และคำสั่ง **passwd** จะล้มเหลว สำหรับผู้ใช้ที่ผ่านการพิสูจน์ตัวตน Kerberos

ถ้าเน็ตเวิร์กที่ **kadmind** daemon มีอยู่ไม่สามารถเข้าถึงได้ เวลาตอบสนองอาจหน่วงออกไป การตั้งค่าอ็อพชัน kadmind เป็น no ในไฟล์ methods.cfg เป็นการกำจัดเวลาหน่วงระหว่างการพิสูจน์ตัวตนเมื่อเครื่อง ไม่สามารถเข้าถึงได้

เมื่อ **kadmind** daemon ไม่ทำงาน ผู้ใช้ ที่มีรหัสผ่านหมดอายุจะสามารถล็อกอิน หรือเปลี่ยนรหัสผ่านของตน

เมื่อ คุณตั้งค่า kadmind=no แต่ **kadmind** daemon กำลังทำงานอยู่ คุณสามารถรันคำสั่งต่อไปนี้: **login**, **su**, **passwd**, **mkuser**, **chuser** และ **rmuser**

Kerberos เทียบกับ Network Authentication Service: ข้อมูล การแก้ปัญหา:

หัวข้อนี้จัดเตรียมข้อมูลการแก้ปัญหาเกี่ยวกับไคลเอ็นต์ Kerberos ที่กำลังใช้เซิร์ฟเวอร์ Kerberos บนระบบปฏิบัติการ AIX

โมดูล LDAP บันทึกข้อมูลข้อผิดพลาดและการดีบักลงใน ระบบย่อย syslog

IBM Network Authentication Service ใช้ล็อกไฟล์ของตนเพื่อบันทึกการร้องขอที่ส่ง ไปยัง KDC และ **kadmind** daemons ล็อกไฟล์ถูก ระบุใน `[logging]` stanza ของไฟล์ `krb5.conf` ตำแหน่งดีฟอลต์ของไฟล์เหล่านี้คือไฟล์ `/var/krb5/log/krb5kdc.log` และไฟล์ `/var/krb5/log/kadmin.log`

ถ้า ปัญหาเกี่ยวข้องกับ IBM Tivoli Directory Server ให้ตรวจสอบ ล็อกไฟล์ที่สร้างโดย IBM Tivoli Directory Server โดยดีฟอลต์ ล็อกไฟล์อยู่ในไฟล์ `/var/ldap/ibmslapd.log` และไฟล์ `/var/ldap/db2cli.log`

- ***ฉันจะสร้างผู้ใช้ที่พิสูจน์ตัวตน AIX Kerberos ได้อย่างไร?***

  ผู้ใช้ root ต้องให้ Kerberos credentials ที่ให้สิทธิพิเศษที่จำเป็นสำหรับดำเนินงาน การดูแล งานการดูแลถูกดำเนินการบนเซิร์ฟเวอร์ KDC ต่อไปนี้: kdcsrv.austin.ibm.com

  สร้างแอคเคาต์ผู้ใช้ AIX (foo) และ Kerberos principal (foo@MYREALM) บนฐานข้อมูล Kerberos โดยป้อน คำสั่งต่อไปนี้:

  ```
  kinit root/kdcsrv.austin.ibm.com
  mkuser -R KRB5files SYSTEM=KRB5files registry=KRB5files foo
  ```

  คำสั่ง เหล่านี้ยังพิสูจน์ตัวตนผู้ใช้ไปยังไฟล์ KRB5files

  ถ้าคุณกำหนดคอนฟิก LDAP โดยใช้คำสั่ง **mksecldap,** คุณสามารถสร้างผู้ใช้ที่พิสูจน์ตัวตน AIX Kerberos โดยป้อนคำสั่งต่อไปนี้:

  ```
  mkuser -R KRB5LDAP SYSTEM=KRB5LDAP registry=KRB5LDAP foo
  ```

- ***ฉันจะลบผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos ออกอย่างไร?***

  ในการ ลบผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos ให้ป้อนคำสั่งต่อไปนี้ :

  ```
  rmuser -R KRB5files foo
  ```

  ถ้าคุณตั้งค่า LDAP โดยใช้คำสั่ง **mksecldap** คุณสามารถลบ ผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos ออกโดยการป้อนคำสั่งต่อไปนี้:

  ```
  rmuser -R KRB5LDAP foo
  ```

- ***ฉันจะเปลี่ยนรหัสผ่านของผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos ได้อย่างไร?***

  ในการเปลี่ยนรหัสผ่านของผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos ป้อนคำสั่งต่อไปนี้:

  ```
  passwd -R KRB5files foo
  ```

- ***อะไรคือแอ็ตทริบิวต์ที่ขยายเพิ่มของ AIX Kerberos?***

  ข้อมูล Kerberos principal ถูกจัดการโดยใช้แอ็ตทริบิวต์ที่ขยายเพิ่ม AIX ผ่านคำสั่ง AIX **lsuser** และ **chuser** เฉพาะแอ็ตทริบิวต์ที่มีโหมดการเข้าถึง GET เท่านั้นที่สามารถแสดง แอ็ตทริบิวต์ที่มีโหมดการเข้าถึง SET สามารถกำหนดค่าโดยผู้ใช้ที่มีสิทธิพิเศษ (root บนระบบปฏิบัติการ AIX) ผู้ใช้ที่พิสูจน์ตัวตน AIX Kerberos สามารถแสดงแอ็ตทริบิวต์ที่ขยายเพิ่ม Kerberos ของตนเองและอนุญาตให้ใช้แอ็ตทริบิวต์ AIX อื่น เช่น id, pgrp, groups, gecos, home, และ shell

  ตารางที่ 18 ในหน้า 331 แสดงแอ็ตทริบิวต์ที่ขยายเพิ่ม AIX Kerberos และโหมดการเข้าถึง

ตารางที่ 18. แอ็ตทริบิวต์ที่ขยายเพิ่ม AIX Kerberos และโหมดการเข้าถึง

ชื่อแอ็ตทริบิวต์ที่ขยาย	คำอธิบาย	โหมดการเข้าถึง
krb5_principal_name	ชื่อ principal ที่เชื่อมโยงกับชื่อผู้ใช้ AIX	GET
krb5_principal	เหมือนกับแอ็ตทริบิวต์ krb5_principal_name	GET
krb5_realm	ชื่อ Kerberos realm name ที่ principal เป็นสมาชิก	GET
krb5_last_pwd_change	เวลาที่รหัสผ่านสำหรับ principal ถูกเปลี่ยนแปลงครั้งล่าสุด	GET
krb5_attributes	ชุดของแอ็ตทริบิวต์ที่ใช้โดย KDC	GET/SET
krb5_mod_name	ชื่อของ principal ผู้แก้ไข principal ครั้งล่าสุด	GET
krb5_mod_date	เวลาที่ principal ถูกแก้ไขล่าสุด	GET
krb5_kvno	เวอร์ชันของคีย์ปัจจุบันของ principal (รหัสผ่าน)	GET/SET
krb5_mkvno	หมายเลขเวอร์ชันมาสเตอร์คีย์ของฐานข้อมูล ค่านี้ใช้เพื่อความเข้ากันได้กับการนำไปใช้อื่นๆ ฟิลด์นี้เป็น 0	GET
krb5_max_renewable_life	ช่วงอายุสูงสุดที่สามารถต่ออายุได้ของตั๋วใดๆ ที่ออกเพื่อ principal	GET/SET
krb5_names	รายการคู่ name:hostname ฟิลด์นี้ สำหรับใช้ในอนาคต ห้ามแก้ไขแอ็ตทริบิวต์นี้	GET/SET

แอ็ตทริบิวต์ที่ขยาย krb5_attributes แทน ชุดของแอ็ตทริบิวต์ Kerberos principal ที่มีใช้โดย KDC ผู้ใช้ที่มีสิทธิพิเศษสามารถใช้คำสั่ง **chuser** เพื่อแก้ไขแอ็ตทริบิวต์ Kerberos เหล่านี้

```
chuser -R KRB5files krb5_attributes=+requires_preauth krb5user
```

ในการ ตั้งค่าแฟล็ก เพิ่มเครื่องหมายบวก (+) ข้างหน้าแฟล็ก ในการตั้งค่าแฟล็กใหม่ ให้เพิ่มเครื่องหมายลบ (-) ข้างหน้าแฟล็ก ตัวอย่าง:

+attribute_name ตั้งค่า แฟล็ก

-attribute_name ตั้งค่า แฟล็กใหม่

หมายเหตุ: เมื่อสร้างผู้ใช้ แอ็ตทริบิวต์ทั้งหมดยกเว้น ต่อไปนี้จะถูกตั้งค่า: requires_hwauth, needchange, password_changing_service และ support_desmd5

รายการต่อไปนี้มีแอ็ตทริบิวต์ สำหรับแอ็ตทริบิวต์ที่ขยาย krb5_attributes:

allow_postdated
ถ้าตั้งค่า สามารถออกตั๋ว postdated สำหรับ principal

allow_forwardable
ถ้าตั้งค่า สามารถออกตั๋ว forwardable สำหรับ principal

allow_tgs_req
ถ้าตั้งค่า ตั๋วสำหรับเซอร์วิสสำหรับ principal ถูกออกโดยใช้ตั๋วที่ใช้ในการให้สิทธิ ตั๋ว

allow_renewable
ถ้าตั้งค่า ตั๋วที่ต่ออายุใหม่สามารถออกสำหรับ principal

allow_proxiable
ถ้าตั้งค่า สามารถออกตั๋ว proxiable สำหรับ principal

allow_dup_skey
ถ้าตั้งค่า การพิสูจน์ตัวตน user-to-user จะถูกเปิดใช้งานสำหรับ principal

allow_tix
ถ้าตั้งค่า ตั๋วจะถูกออกสำหรับ principal

requires_preauth
ถ้าตั้งค่า จำเป็นต้องใช้การพิสูจน์ตัวตนล่วงหน้าโดยซอฟต์แวร์ก่อนออก ตั๋ว

requires_hwauth
ถ้าตั้งค่า การพิสูจน์ตัวตนฮาร์ดแวร์ล่วงหน้าโดยซอฟต์แวร์จำเป็นต้องใช้ ก่อนออกตั๋วสำหรับ principal

needchange
ถ้าตั้งค่า คีย์ (รหัสผ่าน) สำหรับ principal ต้องถูกเปลี่ยนก่อน ออกตั๋ว

หมายเหตุ: ถ้าแฟล็ก needchange ถูกตั้งค่า ผู้ใช้จะได้รับ พร้อมต์เพื่อเปลี่ยนรหัสผ่านระหว่างการพยายามล็อกอินครั้งถัดไป ใน กรณีนี้ ผู้ใช้ได้รับการพิสูจน์ตัวตน (โดยใช้ Kerberos) แต่ไม่มี ตั๋วในใช้ในการให้สิทธิตั๋ว ในการขอรับตั๋วที่ใช้ในการให้สิทธิตั๋ว ผู้ใช้ต้องร้องเรียกใช้คำสั่ง **kinit** แฟล็ก needchange ใช้กับ Kerberos ที่กำลังใช้โมดูล Network Authentication Services เท่านั้น

allow_svr
ถ้าตั้งค่า สามารถออกตั๋วสำหรับเซอร์วิสสำหรับ principal

password_changing_service
ถ้าตั้งค่า principal จะเป็น principal พิเศษสำหรับเซอร์วิสการเปลี่ยน รหัสผ่าน

support_desmd5
ถ้าตั้งค่า KDC อาจออกตั๋วที่ใช้อัลกอริทึมการผลรวมตรวจสอบ RSA MD5

หมายเหตุ: การตั้งค่าแอ็ตทริบิวต์นี้อาจทำให้เกิดปัญหา การทำงานร่วมกัน

- ***ฉันจะแสดงรายการแอ็ตทริบิวต์ที่ขยายเพิ่ม AIX Kerberos ได้อย่างไร?***

  ในการแสดงรายการแอ็ตทริบิวต์ที่ขยาย AIX Kerberos รันคำสั่ง ต่อไปนี้:

  ```
  lsuser -R KRB5files foo
  ```

  คุณยัง สามารถแสดงรายการแอ็ตทริบิวต์ที่ขยายที่เฉพาะเจาะจงได้โดยใช้อ็อพชัน –a ตัวอย่าง:

  ```
  lsuser -R KRB5files -f -a krb5_principal krb5_principal_name krb5_realm
  ```

- ***ฉันจะปรับเปลี่ยนแอ็ตทริบิวต์ที่ขยายเพิ่ม AIX Kerberos ได้อย่างไร?***

  ผู้ใช้ที่ได้รับสิทธิพิเศษเท่านั้นที่สามารถแก้ไข แอ็ตทริบิวต์ที่ขยายต่อไปนี้ที่มีโหมดการเข้าถึง SET: krb5_kvno, krb5_max_renewable_life, krb5_attributes และ krb5_names

  – ในการเปลี่ยนช่วงอายุที่ต่ออายุใหม่ได้สูงสุดเป็นห้าวันสำหรับ ตั๋วใดๆ ที่ออกให้แก่ foo ให้ป้อนคำสั่งต่อไปนี้:

    ```
    chuser -R KRB5files krb5_max_renewable_life=432000 foo
    ```

  – ในการเปลี่ยนหมายเลขเวอร์ชันคีย์ (รหัสผ่าน) ของ principal ที่เชื่อมโยง กับ foo ป้อนคำสั่งต่อไปนี้:

    ```
    chuser -R KRB5files krb5_kvno=4 foo
    ```

  – ในการตั้งค่าแอ็ตทริบิวต์ Kerberos principal ทั้งหมดที่แสดงรายการใน ตารางที่ 18 ในหน้า 331 ป้อนคำสั่งต่อไปนี้:

```
chuser -R KRB5files krb5_attributes=+allow_postdated,+allow_forwardable,\
+allow_tgs_req,+allow_renewble,+allow_proxiable,+allow_dup_skey,+allow_tix,\
+requires_preauth,+requires_hwath,+needchange,+allow_svr,\
+password_changing_service,+support_desmd5 foo

lsuser -R KRB5files -a krb5_attributes foo
```

– ในการตั้งค่าแอ็ตทริบิวต์ Kerberos principal ใหม่ทั้งหมดที่แสดงรายการใน ตารางที่ 18 ในหน้า 331 ป้อนคำสั่งต่อไปนี้:

```
chuser -R KRB5files krb5_attributes=-allow_postdated,-allow_forwardable,\
-allow_tgs_req,-allow_renewable,-allow_proxiable,-allow_dup_skey,\
-allow_tix,-requires_preauth,-requires_hwauth,-needchange,-allow_svr,\
-password_changing_service,-support_desmd5 foo

lsuser -R KRB5files -a krb5_attributes foo
```

– เมื่อต้องการเปลี่ยน krb5_names และเพิ่มคู่ชื่อผู้ใช้/ชื่อโฮสต์ AIX, ให้ป้อนคำสั่งต่อไปนี้:

```
lsuser -R KRB5files -a krb5_names foo

chuser -R KRB5files krb5_names=bar:greenjeans.austin.ibm.com foo

lsuser -R KRB5files -a krb5_names foo
```

- ***ฉันจะแสดงรายการผู้ใช้ทั้งหมดที่ถูกกำหนดในไฟล์ KRB5files ได้อย่างไร?***

  ในการ แสดงรายการผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos ทั้งหมด ป้อนคำสั่ง ต่อไปนี้:

  ```
  lsuser -R KRB5files -a registry ALL
  ```

- ***ฉันแปลงผู้ใช้ AIX เป็นผู้ใช้ที่พิสูจน์ตัวตน Kerberos ได้อย่างไร?***

  ใช้คำสั่ง **mkseckrb5** เพื่อแปลง ผู้ใช้ AIX เป็นผู้ใช้ Kerberos ที่พิสูจน์ตัวตน คำสั่ง **mkseckrb5** จะแปลงผู้ใช้ที่ไม่ใช่ผู้ดูแล (ผู้ใช้ที่มี ID ผู้ใช้ที่มากกว่า 201) เป็นผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos เมื่อคุณเรียกใช้คำสั่ง **mkseckrb5** คุณได้รับพร้อมต์สำหรับชื่อ principle การจัดการ Network Authentication Service และรหัสผ่าน ถ้าคุณไม่ใช้อ็อพชันแบบสุ่ม คุณจะไดรับพร้อมต์สำหรับผู้ใช้แต่ละคนที่คุณกำลังแปลง

  หมายเหตุ: คำสั่ง **mkseckrb5** แปลงเฉพาะผู้ใช้โลคัลเท่านั้น ผู้ใช้ในรีโมตโดเมน เช่น LDAP ไม่สามารถถูกแปลงโดยใช้คำสั่งนี้

  ตัวอย่างต่อไปนี้ *ไม่ได้* ใช้อ็อพชันการสุ่ม ในระหว่างการแปลงผู้ใช้ AIX เป็นผู้ใช้ที่พิสูจน์ตัวตน Kerberos

  1. ป้อนคำสั่งต่อไปนี้:

     ```
     mkseckrb5 foo
     ```

  2. ก่อนที่คุณจะล็อกอินผู้ใช้ที่มี Kerberos ให้ตั้งค่า SYSTEM ของผู้ใช้ และรีจิสทรีแอ็ตทริบิวต์ ดังนี้:

     ```
     chuser -R KRB5files SYSTEM=KRB5files registry=KRB5files foo
     ```

  ตัวอย่างต่อไปนี้ใช้อ็อพชันการสุ่ม ในระหว่างการแปลงผู้ใช้ AIX เป็นผู้ใช้ที่พิสูจน์ตัวตน Kerberos

  1. ป้อนคำสั่งต่อไปนี้:

     ```
     mkseckrb5 -r user1
     ```

  2. หลังการแปลงเสร็จเรียบร้อย ตั้งค่า SYSTEM ของผู้ใช้ รีจิสทรี แอ็ตทริบิวต์ และรหัสผ่านดังนี้:

     ```
     chuser -R KRB5files SYSTEM=KRB5files registry=KRB5files user1

     passwd -R KRB5files user1
     ```

- ***ฉันจะเปลี่ยนรหัสผ่านสำหรับ Kerberos principal ได้อย่างไร?***

  ผู้ใช้ root สามารถตั้งค่ารหัสผ่านของ Kerberos principal โดยการป้อน คำสั่ง **passwd** ต่อไปนี้:

```
passwd -R KRB5files foo
```

ข้อความ ต่อไปนี้แสดงหลังจากคุณป้อนคำสั่ง **passwd**:

```
Changing password for "foo"
foo's Old password:
foo's New password:
Enter the new password again:
```

เมื่อคุณป้อนคำสั่ง **passwd** เป็นผู้ใช้ root รหัสผ่านเก่าจะถูกข้าม คุณสามารถปิดใช้งานพร้อมต์ สำหรับรหัสผ่านเก่า โดยการใช้อ็อพชัน rootpwdrequired ในไฟล์ `methods.cfg` ในการปิดใช้งานพร้อมต์สำหรับรหัสผ่านเก่า แก้ไขไฟล์ `/usr/lib/security/methods.cfg` ดังนี้:

```
KRB5files:
   options = db=BUILTIN,auth=KRB5,rootrequiresopw=false
```

- ***ฉันจะได้รับตั๋วที่ใช้ในการให้สิทธิตั๋วหลังล็อกอินสำเร็จ เมื่อแอ็ตทริบิวต์ needchange ถูกตั้งค่าได้อย่างไร?***

  ในการขอรับตั๋วที่ใช้ ในการให้สิทธิตั๋วหลังจากล็อกอินสำเร็จเมื่อแฟล็ก needchange ถูกตั้งค่า ให้เรียกใช้ คำสั่ง **kinit** สำหรับข้อมูลเพิ่มเติมเกี่ยวกับเรื่องนี้ ดูที่แอ็ตทริบิวต์ **needhange**

- ***ทำไมรหัสผ่านของฉันไม่ยอมรับโดยระบบปฏิบัติการ AIX?***

  ถ้ารหัสผ่านของคุณไม่ได้รับการยอมรับ ดำดังนี้:

  - ตรวจสอบว่าเซิร์ฟเวอร์ KDC และ kadmind กำลังทำงาน
  - ตรวจสอบว่ารหัสผ่านตรงตามข้อกำหนดของระบบปฏิบัติการ AIX และ Network Authentication Service

- ***ฉันจะเปลี่ยนกฎรหัสผ่านได้อย่างไร?***

  คุณสามารถเปลี่ยนกฎของรหัสผ่าน บนระบบปฏิบัติการ AIX ได้โดยปรับเปลี่ยนแอ็ตทริบิวต์นโยบายรหัสผ่าน คุณสามารถใช้เครื่องมือ Network Authentication Server kadmin เพื่อเปลี่ยนนโยบายรหัสผ่านบน ฐานข้อมูล Kerberos

- ***ผู้ใช้ที่พิสูจน์ตัวตน Kerberos กลายเป็นการพิสูจน์ตัวตนโดยใช้การพิสูจน์ตัวตน AIX แบบมาตรฐานได้หรือไม่?***

  ผู้ใช้ที่พิสูจน์ตัวตน Kerberos (foo) สามารถกลายเป็นการพิสูจน์ตัวตนโดยใช้การพิสูจน์ตัวตน AIX **crypt( )** ดังต่อไปนี้:

  1. ตั้งค่ารหัสผ่าน AIX ของผู้ใช้ foo (`/etc/security/passwd`) โดยใช้คำสั่ง **passwd**
  2. เลือกรหัสผ่านอื่นเพื่อวัตถุประสงค์ในการทดสอบ ตัวอย่าง:

     ```
     passwd -R files foo
     ```

  3. เปลี่ยนแอ็ตทริบิวต์ SYSTEM ของผู้ใช้ ดังนี้:

     ```
     chuser -R KRB5files SYSTEM=compat foo
     ```

     การเปลี่ยน แอ็ตทริบิวต์ SYSTEM เป็นการเปลี่ยนวิธีการการพิสูจน์ตัวตนจาก Kerberos เป็น **crypt( )**

     หมายเหตุ: เนื่องจากผู้ใช้ในตัวอย่างนี้ ล็อกอินโดยใช้การพิสูจน์ตัวตนโลคัล ค่า AUTHSTATE compat และไม่มีการออกตั๋วที่ใช้ในการให้สิทธิตั๋ว ถ้าคุณต้องการใช้การพิสูจน์ตัวตน **crypt( )** เป็นวิธีสำรอง ให้ไปที่ขั้นตอน 4

  4. ในการใช้การพิสูจน์ตัวตน **crypt( )** เป็นวิธีสำรอง ให้เปลี่ยนแอ็ตทริบิวต์ SYSTEM ดังนี้:

     ```
     chuser -R KRB5files SYSTEM="KRB5files or compat" foo
     ```

- ***ฉันจะเปลี่ยนพอร์ต kadmind ไคลเอ็นต์ได้อย่างไร?***

  kadmind daemon ถูกใช้เพื่อดำเนินการจัดการ Kerberos principal บนระบบที่ได้รับการพิสูจน์ตัวตน Kerberos ที่กำลังใช้ NAS ตัวอย่างต่อไปนี้แสดงวิธี เปลี่ยนพอร์ต **kadmind** ไคลเอ็นต์ ในตัวอย่างนี้ **kadmind** daemon ทำงานบนเซิร์ฟเวอร์ `kdcsrv.austin.ibm.com` และใช้พอร์ต 812

  1. ใช้คำสั่ง **config.krb5** เพื่อตั้งค่า ไคลเอ็นต์:

```
config.krb5 -C -r MYREALM -c kdcsrv.austin.ibm.com -s \
kdcsrv.austin.ibm.com -d austin.ibm.com
```

2. แก้ไขไฟล์ krb5.conf และเปลี่ยนหมายเลขพอร์ต:

```
admin_server = kdcsrv.austin.ibm.com:812
```

- ***ฉันจะลบ Kerberos credentials ออกได้อย่าง?***

  แต่ละครั้ง ที่ผู้ใช้ล็อกอิน Kerberos credentials ก่อนหน้าจะถูกบันทึกทับ อย่างไรก็ตาม เมื่อผู้ใช้ล็อกเอาต์ credentials เหล่านี้จะไม่ถูกลบออกไป ในการลบ credentials เหล่านี้ออก ป้อนคำสั่ง NAS **kdestroy** ต่อไปนี้:

  ```
  /usr/krb5/bin/kdestroy
  ```

- ***ฉันจะเปลี่ยนช่วงอายุตั๋วบน KDC ได้อย่างไร?***

  ในการ เปลี่ยนช่วงอายุตั๋วบน KDC ให้ทำต่อไปนี้:

  1. เปลี่ยนแอ็ตทริบิวต์ max_life ในไฟล์ kdc.conf ตัวอย่าง :

     ```
     max_life = 8h 0m 0s
     ```

  2. หยุดทำงานจากนั้นเริ่มทำงาน **krb5kdc** และ **kadmind** daemons
  3. เปลี่ยนค่า max_life ของ krbtgt/MYREALM and kadmin/admin principals เป็นค่าที่คุณป้อนในขั้นตอน 1 ตัวอย่าง:

     ```
     kadmin.local
     kadmin.local: modify_principal -maxlife "8 hours" krbtgt/MYREALM
     ```

- ***ถ้า kadmind daemon ไม่พร้อมใช้งานจะเกิดอะไร?***

  ถ้า kadmind daemon ไม่พร้อมใช้งาน การพิสูจน์ตัวตนอาจใช้เวลานานขึ้น หรือล้มเหลว การพิสูจน์ตัวตนอาจล้มเหลวถ้าส่วนหนึ่งของเน็ตเวิร์ก ที่มี kadmind daemon อยู่ไม่สามารถเข้าถึงได้ หรือระบบ ที่กำลังโฮสต์เซิร์ฟเวอร์ kadmind ไม่ทำงาน เมื่อระบุไม่ สามารถเข้าถึงได้ การตั้งค่าอ็อพชัน kadmind ในไฟล์ methods.cfg เป็น no เพื่อกำจัดการหน่วงระหว่างการพิสูจน์ตัวตน

  เมื่อ kadmind daemon ไม่ทำงาน ผู้ใช้ไม่สามารถล็อกอินได้ถ้ารหัสผ่าน หมดอายุ ถ้า kadmind daemon ไม่พร้อมใช้งาน (daemon ไม่ทำงานหรือไม่สามารถเข้าถึงได้) และผู้ใช้ป้อนคำสั่ง **mkuser** ข้อผิดพลาดต่อไปนี้จะถูกแสดง:

  ```
  3004-694 Error adding "krb5user": You do not have permission
  ```

  นอกจากนั้น คำสั่ง **chuser** และ **lsuser** จัดการเฉพาะแอ็ตทริบิวต์ที่เกี่ยวข้องกับ AIX เท่านั้น ไม่จัดการแอ็ตทริบิวต์ที่เกี่ยวข้องกับ Kerberos คำสั่ง **rmuser** ไม่ลบ Kerberos principal และคำสั่ง **passwd** ล้มเหลวสำหรับผู้ใช้ ที่ได้รับการพิสูจน์ตัวตน Kerboros

  เมื่อ kadmind daemon ไม่พร้อมใช้งาน ผู้ใช้ root จะไม่สามารถเปลี่ยนรหัสผ่านผู้ใช้ในสถานการณ์เช่น ลืมรหัสผ่าน คุณต้องทำให้ kadmind daemon พร้อมใช้งาน และ, หากผู้ใช้เลือกเพื่อป้อนชื่อ Kerberos principal ที่พร้อมต์ล็อกอิน, ชื่อหลักของชื่อ principal จะถูกตัดปลาย (ตามข้อจำกัดความยาวชื่อผู้ใช้ AIX) ชื่อที่ถูกตัดปลายถูกใช้สำหรับการดึงข้อมูล indentification ของผู้ใช้ AIX (ตัวอย่างเช่น, ดึงข้อมูลค่า โฮมไดเร็กทอรี)

- ***ฉันจะกำหนดคอนฟิกระบบปฏิบัติการ AIX สำหรับ Kerberos ที่รวมล็อกอินกับการจัดการผู้ใช้และกลุ่ม LDAP AIX ได้อย่างไร?***

  ถ้าคุณวางแผนที่จะใช้ LDAP เพื่อเก็บข้อมูลผู้ใช้/กลุ่ม AIX, ให้ใช้คำสั่ง **mksecldap** เพื่อกำหนดคอนฟิกเซิร์ฟเวอร์ LDAP และไคลเอ็นต์ก่อนที่คุณจะรันคำสั่ง **mkkrb5srv** และ **mkkrb5clnt** ในการตั้งค่าเซิร์ฟเวอร์ Kerberos ใช้คำสั่ง **mkkrb5srv** ในการตั้งค่าไคลเอ็นต์ Kerberos ใช้คำสั่ง **mkkrb5clnt** ที่มีอ็อพชัน –i LDAP ตัวอย่าง:

  ```
  mkkrb5clnt -r MYREALM -c kdcsrv.ustin.ibm.com\
   -s kdcsrv.austin.ibm.com -a admin/admin -d austin.ibm.com -A -i LDAP -K -T
  ```

- ***ฉันจะใช้คำสั่งรีโมตที่เปิดใช้งาน Kerberos หลังล็อกอินสำเร็จ ได้อย่างไร?***

เมื่อผู้ใช้ AIX พิสูจน์ตัวตนกับระบบโดยใช้ Kerberos, ตั๋วการให้สิทธิ์ตั๋ว สามารถใช้สำหรับคำสั่งแบบรีโมตที่เปิดใช้งาน Kerberos

ในตัวอย่าง ต่อไปนี้ เซิร์ฟเวอร์ NAS ถูกตั้งค่าบน kdcsrv.austin.ibm.com โดย ใช้คำสั่ง **mkkrb5srv** ระบบนี้ยังถูก ตั้งค่าสำหรับการล็อกอินแบบ Kerberos โดยใช้คำสั่ง **mkkrb5clnt** ระบบที่สอง tx3d.austin.ibm.com ถูกตั้งค่าเป็นไคลเอ็นต์โดยใช้คำสั่ง **mkkrb5clnt**

1. บันทึกคีย์สำหรับ host principal host/tx3d.austin.ibm.com ลงในไฟล์ /etc/krb5/krb5.keytab บนระบบ tx3d
2. เนื่องจากคุณใช้ **mkkrb5clnt** เพื่อตั้งค่า เครื่องไคลเอ็นต์ คีย์เหล่านี้ถูกแยกไปยังไฟล์ /var/krb5/security/keytab/tx3d.austin.ibm.com.keytab ลิงก์ไฟล์นี้กับไฟล์ /etc/krb5/krb5.keytab ดังนี้:

   ```
   ln -s /var/krb5/security/keytab/tx3d.austin.ibm.com.keytab /etc/krb5/krb5.keytab
   ```

3. ถ้าระบบ tx3d.austin.ibm.com ถูกกำหนดคอนฟิกด้วยเซิร์ฟเวอร์ที่ไม่ใช่ AIX Kerberos, ให้สร้างโฮสต์ host และแตกคีย์ ตัวอย่าง:

   ```
   kadmin -p admin/admin

   kadmin: addprinc -randkey host/tx3d.austin.ibm.com
   kadmin: ktadd -k /etc/krb5/krb5.keytab host/tx3d.austin.ibm.com
   kadmin:
   ```

   เนื่องจากเครื่องมือ kadmin ถูกเรียกใช้จาก ระบบ tx3d.austin.ibm.com คีย์จะถูกยแกไปยังไฟล์ /etc/krb5/krb5.keytab บนระบบ tx3d.austin.ibm.com คุณยังสามารถทำขั้นตอนนี้บนเครื่อง ที่โฮสต์ Kerberos admin server (ตัวอย่าง kdcsrv) หลังจากคุณแยกคีย์ลงในไฟล์ keytab ไฟล์ถูกถ่ายโอน และผสมรวมกับไฟล์ /etc/krb5/krb5.keytab บน tx3d
4. เปิดใช้งานคำสั่งรีโมตเพื่อใช้การพิสูจน์ตัวตน Kerberos Version 5 บนระบบ tx3d.austin.ibm.com:

   ```
   lsauthent
   Standard Aix
   chauthent -k5 -std
   lsauthent
   Kerberos 5
   Standard Aix
   ```

5. เปิดใช้งานคำสั่งรีโมตเพื่อใช้การพิสูจน์ตัวตน Kerberos Version 5 บนระบบ kdcsrv.austin.ibm.com:

   ```
   chauthent -k5 -std
   lsauthent
   Kerberos 5
   Standard Aix
   ```

6. สร้างผู้ใช้ (foo) ที่ได้รับการพิสูจน์ตัวตน Kerberos บน kdcsrv และตั้งค่า รหัสผ่าน

   ```
   mkuser -R KRB5files SYSTEM=KRB5files registry=KRB5files foo
   passwd -R KRB5files foo
   ```

7. สร้างผู้ใช้ foo บน tx3d:

   ```
   mkuser -R files foo
   ```

8. Telnet ไปยังระบบ kdcsrv.austin.ibm.com โดยใช้การพิสูจน์ตัวตน Kerberos
9. เพื่อให้แน่ใจว่ามีการออกตั๋วที่ใช้ในการให้สิทธิตั๋ว ให้ป้อนคำสั่ง **klist**

   ```
   /usr/krb5/bin/klist
   ```

ต่อไปนี้คือตัวอย่างของคำสั่งรีโมตที่เปิดใช้งาน Kerberos

หมายเหตุ: ก่อนคุณรันคำสั่งในตัวอย่างต่อไปนี้ ให้ลบไฟล์ `.klogin`, `.rhost` หรือ `hosts.equiv`

- ป้อนคำสั่ง **date** บนระบบโฮสต์ remote tx3d.austin.ibm.com ด้วยคำสั่ง **rsh**:

```
rsh tx3d date
```

- ล็อกอินเข้าสู่ระบบ remote tx3d.austin.ibm.com ด้วยคำสั่ง **rlogin**:

```
hostname
kdcsrv.austin.ibm.com
rlogin tx3d -l foo
*******************************
* Welcome to AIX Version 6.1! *
*******************************
hostname
tx3d.austin.ibm.com
id
uid=234(foo) gid=1(staff)
```

- ถ่ายโอนไฟล์ไปยังระบบ tx3d.austin.ibm.com รีโมตด้วย คำสั่ง **rcp**:

```
rsh tx3d "ls -l /home/foo"
total 0
echo "Testing Kerberize-d rcp" >> xfile
rcp xfile tx3d:/home/foo
rsh tx3d "ls -l /home/foo"
total 0
-rw-r--r-- 1 foo staff 0 Apr 28 14:30 xfile
rsh tx3d "more /home/foo/xfile"
Testing Kerberize-d rcp
```

- Telnet ไปยังระบบ tx3d.austin.ibm.com รีโมตด้วย Kerberos credentials:

```
telnet tx3d
Trying...
Connected to tx3d.austin.ibm.com.
Escape character is '^]'.
[ Kerberos V5 accepts you as "foo@MYREALM" ]
```

- Telnet to the tx3d.austin.ibm.com system, and then enter the host name and ID when prompted:

```
hostname
tx3d.austin.ibm.com
id
uid=234(foo) gid=1(staff)
```

- ก่อนคุณสามารถใช้คำสั่ง **ftp** ที่เปิดใช้งาน Kerboros คุณต้องใช้คำสั่ง **kadmin** (จาก tx3d.austin.ibm.com) เพื่อสร้าง FTP service principal `ftp/tx3d.austin.ibm.com` และแยกออกมาไว้ในไฟล์ `/etc/krb5/krb5.keytab`:

```
kadmin: addprinc -randkey ftp/tx3d.austin.ibm.com@MYREALM
kadmin: ktadd -k /etc/krb5/krb5.keytab ftp/tx3d.austin.ibm.com@MYREALM
kadmin:
```

ต่อไปนี้คือตัวอย่างของวิธี FTP ไปยัง ระบบรีโมต tx3d.austin.ibm.com ด้วย Kerberos credentials

```
ftp tx3d
Name (tx3d:foo): foo
232 GSSAPI user foo@MYREALM is authorized as foo
230-Last login: Thu May 19 17:58:57 CDT 2005 on ftp from kdcsrv.austin.ibm.com
230 User foo logged in.
ftp> ftp> ls -la
```

การตั้งค่าไคลเอ็นต์ Kerberos กับเซิร์ฟเวอร์ Kerberos บน ระบบที่มิใช่ AIX:

ไคลเอ็นต์ AIX Kerberos สามารถกำหนดคอนฟิกกับเซิร์ฟเวอร์ Kerberos บนระบบ ที่ไม่ใช่AIX: Windows Active Directory, Solaris, และ HP

การตั้งค่า Kerberos กับ Windows Server Kerberos Service:

มีวิธีการที่ใช้ได้หลายวิธีสำหรับการตั้งค่า Kerberos กับ Windows Server Kerberos Service

โมดูลที่ทำการพิสูจน์ตัวตน Kerberos เท่านั้นใน KRB5 สามารถ ถูกใช้ในส่วนการพิสูจน์ตัวตนของโหลดโมดูลแบบผสม ระหว่าง การตั้งค่า ผู้ใช้ระบุสภาวะแวดล้อม Kerberos สำหรับ โหลดโมดูล โหลดโมดูล KRB5 ช่วยให้ Kerberos ใช้เป็นวิธีทางเลือก หนึ่งสำหรับการพิสูจน์ตัวตนกับ Windows 2000 หรือ Windows 2003 Server Kerberos Service โหลดโมดูล AIX BUILTIN pseudo จัดเตรียมการเข้าถึงฟังก์ชันไลบรารีความปลอดภัย โหลดโมดูล BUILTIN สามารถรวมเข้ากับโหลดโมดูลที่ทำการพิสูจน์ตัวตนเท่านั้น เพื่อจัดให้มีส่วนฐานข้อมูลของโหลดโมดูลผสม รวมทั้งจัดให้มี สื่อบันทึก legacy-user-and-group และการเข้าถึงระบบไฟล์โหลดโมดูล LDAP ยังสามารถใช้เป็นส่วนฐานข้อมูลของโหลดโมดูลผสม

ไม่เหมือนกับสภาวะแวดล้อม Kerberos อื่นๆ เมื่อเทียบกับ NAS บนระบบ AIX, สภาวะแวดล้อมนี้ไม่ได้จัดเตรียมการจัดการกับ Kerberos principal ไว้โหลดโมดูล KRB5 สามารถใช้ได้ในสภาวะแวดล้อมที่ Kerberos principals ถูกเก็บไว้บนระบบ ที่ไม่ใช่ AIX และไม่สามารถจัดการได้จากระบบปฏิบัติการ AIX โดยใช้อินเตอร์เฟส **kadmin** Kerberos-database การจัดการ Kerberos principal ถูกดำเนินการแยกต่างหากกับเครื่องมือการจัดการ Kerberos principal เครื่องมือเหล่านี้อาจเป็นส่วนหนึ่งของผลิตภัณฑ์ Kerberos ที่พัฒนาโดยผู้จำหน่ายซอฟต์แวร์หรือผนวกรวมกับ OS เช่น Windows 2000

การตั้งค่า Windows Server 2000 Kerberos Service:

Windows Server 2000 Kerberos Service และไคลเอ็นต์ NAS สามารถทำงานร่วมกันได้ที่ระดับโปรโตคอล Kerberos (RFC1510) เนื่องจาก Windows Server 2000 ไม่สนับสนุนอินเตอร์เฟส **kadmin**, ประกอบด้วยแฟล็ก –D ในคำสั่ง **mkkrb5clnt** ในระหว่างคอนฟิกูเรชันของไคลเอ็นต์ AIX ใช้เครื่องมือ Windows เพื่อจัดการ principals บนระบบ Windows

ใช้โพรซีเดอร์ต่อไปนี้เพื่อกำหนดคอนฟิกไคลเอ็นต์ AIX สำหรับการพิสูจน์ตัวตนแบบอิง Kerberos กับ Windows Server 2000 Kerberos Service

1. ตั้งค่า Windows Server 2000 อ้างอิงเอกสารคู่มือ Microsoft สำหรับการตั้งค่า Microsoft Active Directory Server
2. ถ้าไม่ได้ติดตั้งไคลเอ็นต์ NAS ไว้บนไคลเอ็นต์ AIX, ให้ติดตั้งชุดไฟล์ `krb5.client.rte` จากแพ็กเสริม AIX
3. ใช้คำสั่ง **mkkrb5clnt** ด้วยข้อมูลคอนฟิกูเรชันต่อไปนี้ เพื่อกำหนดคอนฟิกไคลเอ็นต์ AIX Kerberos:

   realm โดเมนเนม Windows Active Directory

   domain โดเมนเนมของเครื่องที่ทำหน้าที่โฮสต์เซิร์ฟเวอร์ Active Directory

   KDC ชื่อโฮสต์ของเซิร์ฟเวอร์ Windows

   server ชื่อโฮสต์ของเซิร์ฟเวอร์ Windows

   ต่อไปนี้คือตัวอย่างของคำสั่ง **mkkrb5clnt**:

   ```
   mkkrb5clnt -r MYREALM -d austin.ibm.com -c w2k.austin.ibm.com -s w2k.austin.ibm.com -D
   ```

อ็อพชัน **-D** ในคำสั่ง **mkkrb5clnt** สร้างอ็อพชัน **is_kadmind_compat=no** ในไฟล์ /etc/methods.cfg และกำหนดค่าสภาวะแวดล้อมไคลเอ็นต์ Kerberos สำหรับการพิสูจน์ตัวตน กับระบบที่ไม่ใช่ AIX ห้ามใช้อ็อพชัน **-D** ในคำสั่ง **mkkrb5clnt** เพื่อกำหนดคอนฟิก สภาวะแวดล้อมไคลเอ็นต์ Kerberos สำหรับการพิสูจน์ตัวตนกับ IBM Network Authentication Service (NAS)

หมายเหตุ: เมื่อคุณรันคำสั่ง **mkkrb5clnt** stanza ต่อไปนี้ จะถูกเพิ่มในไฟล์ methods.cfg

```
KRB5:
        program = /usr/lib/security/KRB5
        program_64 = /usr/lib/security/KRB5_64
        options = authonly,is_kadmind_compat=no

KRB5files:
        options = db=BUILTIN,auth=KRB5
```

สำหรับข้อมูล เพิ่มเติมเกี่ยวกับ:

- คำสั่ง **mkkrb5clnt** และแฟล็กที่สามารถใช้ได้ ดูที่คำสั่ง **mkkrb5clnt**
- ไฟล์ methods.cfg ดูที่ไฟล์ methods.cfg

4. เนื่องจาก Windows สนับสนุน ประเภทการเข้ารหัส DES-CBC-MD5 และ DES-CBC-CRC ให้เปลี่ยนข้อมูลไฟล์ krb5.conf ให้คล้ายตัวอย่างต่อไปนี้:

```
[libdefaults]
    default_realm = MYREALM
    default_keytab_name = FILE:/etc/krb5/krb5.keytab
    default_tkt_enctypes = des-cbc-md5 des-cbc-crc
    default_tgs_enctypes = des-cbc-md5 des-cbc-crc
```

5. สร้างหลักการโฮสต์

   เนื่องจากชื่อบัญชีผู้ใช้ Windows ไม่มีหลายส่วน เหมือนชื่อ NAS principal คุณไม่สามารถสร้างบัญชีผู้ใช้ได้โดยตรง โดยใช้ชื่อโฮสต์แบบเต็ม (host/<*fully_qualified_host_name*>) แต่ principal instance จะถูกสร้างขึ้นมาแทนผ่านการแม็พ service-principal-name ในกรณีนี้ บัญชีผู้ใช้ถูกสร้างโดยสัมพันธ์กับ หลักการโฮสต์ และการแม็พชื่อหลักการถูกเพิ่มเข้าไป

   บนเซิร์ฟเวอร์ Active Directory, ให้ใช้เครื่องมือ Active Directory Management เพื่อสร้างแอคเคาต์ผู้ใช้ใหม่ที่สอดคล้องกับไคลเอ็นต์ tx3d.austin.ibm.com AIX ดังต่อไปนี้:

   a. เลือกโฟลเดอร์ User
   b. คลิกขวาเพื่อเลือก New
   c. เลือก User
   d. ป้อน tx3d ในฟิลด์ First name จากนั้นคลิก Next
   e. สร้างรหัสผ่าน จากนั้นคลิก Next
   f. คลิก Finish เพื่อสร้างหลักการโฮสต์

6. บนเครื่อง Windows Server 2000, ให้ป้อนคำสั่ง **Ktpass** จากบรรทัดรับคำสั่ง เพื่อสร้างไฟล์ tx3d.keytab และตั้งค่าแอคเคาต์โฮสต์ AIX ดังต่อไปนี้:

```
Ktpass -princ host/tx3d.austin.ibm.com@MYREALM -mapuser tx3d -pass password -out tx3d.keytab
```

7. คัดลอกระบบโฮสต์ tx3d.keytab file to the AIX
8. ผสานไฟล์ tx3d.keytab ไปยังไฟล์ /etc/krb5/krb5.keytab บนระบบ AIX ดังต่อไปนี้:

```
ktutil
rkt tx3d.keytab
wkt /etc/krb5/krb5.keytab
q
```

9. สร้างบัญชีผู้ใช้โดเมน Windows โดยใช้เครื่องมือการจัดการผู้ใช้ Active Directory
10. เมื่อต้องการสร้างแอคเคาต์ AIX ที่สอดคล้องกับแอคเคาต์โดเมน Windows และใช้การพิสูจน์ตัวตน Kerberos, ให้รันคำสั่งต่อไปนี้:

    ```
    mkuser registry=KRB5files SYSTEM=KRB5files foo
    ```

11. เมื่อต้องการล็อกอินเข้าสู่ระบบ AIX และตรวจสอบคอนฟิกูเรชัน, ให้รันคำสั่ง **telnet**

    ต่อไปนี้ คือตัวอย่างของเซสชันการล็อกอินที่รวม Kerberos ที่ ใช้ KRB5 กับ Windows Active Directory:

    ```
    telnet tx3d

    Trying...
    Connected to tx3d.austin.ibm.com.
    Escape character is '^]'.

    telnet (tx3d.austin.ibm.com)
    login: foo
    foo's Password:
    ******************************************************************************
    * Welcome to AIX Version 6.1! *
    ******************************************************************************
    echo $AUTHSTATE
    KRB5files

    /usr/krb5/bin/klist
    Ticket cache: FILE:/var/krb5/security/creds/krb5cc_foo@AUSTIN.IBM.COM_203
    Default principal: foo@AUSTIN.IBM.COM

    Valid starting Expires Service principal
    04/29/05 14:37:28 04/30/05 00:39:22 krbtgt/AUSTIN.IBM.COM@AUSTIN.IBM.COM
         Renew until 04/30/05 14:37:28

    04/29/05 14:39:22 04/30/05 00:39:22 host/tx3d.austin.ibm.com@AUSTIN.IBM.COM
    ```

การตั้งค่า Windows Server 2003 Kerberos Service:

ไคลเอ็นต์ Kerberos สามารถตั้งค่าใช้กับ Windows Server 2003 Kerberos Service

เมื่อต้องการกำหนดคอนฟิกไคลเอ็นต์ AIX กับ Windows Server 2003 Kerberos Service, ให้ใช้ขั้นตอนใน “การตั้งค่า Windows Server 2000 Kerberos Service” ในหน้า 338

หมายเหตุ: ยูทิลิตี ไคลเอ็นต์ NAS **kpasswd** สามารถเปลี่ยนรหัสผ่าน ของ Kerberos principal บน Windows Server 2003 Kerberos Service ดังนั้น, หลังจากที่ล็อกอินเข้าสู่ระบบ AIX ที่กำลังใช้ Kerberos, ผู้ใช้ไม่สามารถเปลี่ยนรหัสผ่านบน Windows Server 2003

การตั้งค่า Kerberos กับ Sun Solaris และ HP-UX Kerberos Domain Controllers:

ไคลเอ็นต์ Kerberos สามารถตั้งค่าใช้กับ Sun Solaris และ HP-UX Kerberos Domain Controllers

ไม่เหมือนกับสภาวะแวดล้อม Kerberos เมื่อเทียบกับ NAS บนระบบ AIX, สภาวะแวดล้อมนี้ไม่ได้จัดเตรียมการจัดการกับ Kerberos โหลดโมดูล KRB5 สามารถใช้ได้ในสภาวะแวดล้อมที่ Kerberos ถูกเก็บอยู่บน ระบบที่ไม่ใช่ AIX และไม่สามารถจัดการได้จากระบบปฏิบัติการ AIX โดยใช้อินเตอร์เฟสฐานข้อมูล **kadmin** Kerberos การจัดการ Kerberos principal ถูกดำเนินการ แยกต่างหากโดยใช้เครื่องมือการจัดการ Kerberos principal เครื่องมือเหล่านี้ อาจเป็นส่วนหนึ่งของผลิตภัณฑ์ Kerberos ที่พัฒนาโดยผู้จำหน่ายซอฟต์แวร์ หรือรวมเข้ากับ OS

การตั้งค่า Sun Solaris:

ไคลเอ็นต์ Kerberos สามารถใช้ตั้งค่าให้กับ Sun Solaris

Sun Enterprise Authentication Mechanism (SEAM) และไคลเอ็นต์ AIX NAS สามารถทำงานร่วมกัน ที่ระดับโปรโตคอล Kerberos (RFC1510) เนื่องจากอินเตอร์เฟส Solaris **kadmind** daemon ทำงานร่วมกันไม่ได้กับไคลเอ็นต์ AIX NAS อินเตอร์เฟส **kadmin,** ประกอบด้วยแฟล็ก –D ในคำสั่ง **mkkrb5clnt** เมื่อคุณกำหนดคอนฟิกไคลเอ็นต์ AIX ใช้เครื่องมือ Solaris เพื่อ ทำการจัดการหลักการบนระบบ Solaris เนื่องจากโปรโตคอลสำหรับการเปลี่ยนรหัสผ่านแตกต่างกัน ระหว่างเซิร์ฟเวอร์ SEAM Kerberos และไคลเอ็นต์ AIX NAS, การเปลี่ยนรหัสผ่านของ principal จะทำให้คอนฟิกูเรชันล้มเหลว

Solaris ถูกใช้ ในตัวอย่างต่อไปนี้

ใช้โพรซีเดอร์ต่อไปนี้ เพื่อกำหนดคอนฟิกไคลเอ็นต์ AIX สำหรับการพิสูจน์ตัวตนแบบอิง Kerberos กับ SEAM

1. ตั้งค่า SEAM โดยการใช้เอกสารคู่มือ Sun
2. ถ้าไม่ได้ติดตั้งไคลเอ็นต์ NAS ไว้บนไคลเอ็นต์ AIX, ให้ติดตั้งชุดไฟล์ `krb5.client.rte` จากแพ็กเสริม AIX
3. เมื่อต้องการกำหนดคอนฟิกไคลเอ็นต์ AIX Kerberos, ให้ใช้คำสั่ง **mkkrb5clnt** พร้อมกับข้อมูลคอนฟิกูเรชันต่อไปนี้:

   realm ชื่อ realm ของ Solaris Kerberos: AUSTIN.IBM.COM

   domain โดเมนเนมของเครื่องที่ทำหน้าที่โฮสต์เซิร์ฟเวอร์ Kerberos: Austin.ibm.com

   KDC ชื่อโฮสต์ของระบบ Solaris ที่ทำหน้าที่โฮสต์ KDC: sunsys.austin.ibm.com

   server ชื่อโฮสต์ของระบบ Solaris ที่ทำหน้าที่โฮสต์ **kadmin** daemon (ปกติเหมือนกับ KDC): sunsys.austin.ibm.com

   หมายเหตุ: เนื่องจากไคลเอ็นต์ Solaris และ AIX NAS อินเตอร์เฟส **kadmin** แตกต่างกัน, ชื่อเซิร์ฟเวอร์ ไม่ถูกใช้โดยไคลเอ็นต์ NAS, และคุณต้องใช้แฟล็ก –D กับคำสั่ง **mkkrb5clnt**

   ต่อไปนี้ คือตัวอย่างของคำสั่ง **mkkrb5clnt** :

   ```
   mkkrb5clnt -r AUSTIN.IBM.COM -d austin.ibm.com\
    -c sunsys.austin.ibm.com -s sunsys.austin.ibm.com -D
   ```

   อ็อพชัน **–D** ในคำสั่ง **mkkrb5clnt** สร้างอ็อพชัน **is_kadmind_compat=no** ในไฟล์ `/etc/security/methods.cfg` และกำหนดคอนฟิกสภาวะแวดล้อมไคลเอ็นต์ Kerberos สำหรับการพิสูจน์ตัวตนกับระบบ ที่ไม่ใช่ AIX ห้ามใช้อ็อพชัน **–D** ในคำสั่ง **mkkrb5clnt** เพื่อกำหนดคอนฟิก สภาวะแวดล้อมไคลเอ็นต์ Kerberos สำหรับการพิสูจน์ตัวตนกับ IBM Network Authentication Service (NAS)

   หมายเหตุ: เมื่อคุณรันคำสั่ง **mkkrb5clnt** stanza ต่อไปนี้ จะถูกเพิ่มในไฟล์ `methods.cfg`

   ```
   KRB5:
           program = /usr/lib/security/KRB5
           program_64 = /usr/lib/security/KRB5_64
   ```

```
        options = authonly,is_kadmind_compat=no

KRB5files:
        options = db=BUILTIN,auth=KRB5
```

สำหรับข้อมูล เพิ่มเติมเกี่ยวกับ:

- คำสั่ง **mkkrb5clnt** และแฟล็กที่สามารถใช้ได้ ดูที่คำสั่ง **mkkrb5clnt**
- ไฟล์ `methods.cfg` ดูที่ไฟล์ `methods.cfg`

4. ใช้เครื่องมือ Solaris **kadmin** เพื่อสร้างหลักการโฮสต์ `host/tx3d.austin.ibm.com@MYREALM` และบันทึกลงไฟล์ คล้ายกับตัวอย่างต่อไปนี้:

```
kadmin: add_principal -randkey host/tx3d.austin.ibm.com
Principal "host/tx3d.austin.ibm.com@AUSTIN.IBM.COM" created.

kadmin:ktadd -k /tmp/tx3d.keytab host/tx3d.austin.ibm.com
Entry for principal host/tx3d.austin.ibm.com with kvno 3,
    encryption type DES-CBC-CRC added to keytab WRFILE:/tmp/tx3d.keytab.

kadmin: quit
```

5. คัดลอกระบบโฮสต์ `tx3d.keytab` file to the AIX
6. ผสานไฟล์ `tx3d.keytab` ไปยังไฟล์ `/etc/krb5/krb5.keytab` บนระบบ AIX ดังต่อไปนี้:

```
ktutil
rkt tx3d.keytab
l
slot KVNO Principal
wkt /etc/krb5/krb5.keytab
q
```

7. ในการสร้าง Kerberos principal ใช้เครื่องมือ Solaris **kadmin**

```
add_principal sunuser
```

8. เมื่อต้องการสร้างแอคเคาต์ AIX ที่สอดคล้องกับ Solaris Kerberos principal และใช้การพิสูจน์ตัวตน Kerberos, ให้ป้อนคำสั่งต่อไปนี้:

```
mkuser registry=KRB5files SYSTEM=KRB5files sunuser
```

9. ใช้คำสั่ง **telnet** เพื่อล็อกอินเข้าสู่ระบบ AIX ด้วยชื่อผู้ใช้และรหัสผ่าน sunuser, และตรวจสอบคอนฟิกูเรชัน ต่อไปนี้คือ ตัวอย่างของเซสชันการล็อกอินที่รวม Kerberos ที่ใช้ KRB5 กับ Solaris KDC:

```
telnet tx3d

echo $AUTHSTATE
KRB5files

echo $KRB5CCNAME
FILE:/var/krb5/security/creds/krb5cc_sunuser@AUSTIN.IBM.COM_207

View credentials:
/usr/krb5/bin/klist
```

การตั้งค่า HP-UX:

ไคลเอ็นต์ Kerberos สามารถใช้ตั้งค่าให้กับ HP-UX

ขั้นตอนในการพิสูจน์ตัวตนกับ HP-UX 11i เหมือนกับ ขั้นตอนใน “การตั้งค่า Sun Solaris” ในหน้า 341 ไคลเอ็นต์ HP-UX KDC และ AIX NAS สามารถทำงานร่วมกันได้ที่ระดับโปรโตคอล Kerberos (RFC1510) โปรโตคอล การเปลี่ยนรหัสผ่านยังเข้ากันได้เช่นกัน เนื่องจากอินเตอร์เฟส HP-UX **kadmind** daemon ทำงานร่วมกันไม่ได้กับไคเอ็นต์ AIX NAS อินเตอร์เฟส **kadmin**, คุณต้องใส่แฟล็ก –D ในคำสั่ง **mkkrb5clnt** เมื่อคุณกำหนดคอนฟิกไคลเอ็นต์ AIX

ใช้โพรซีเดอร์ต่อไปนี้ เพื่อกำหนดคอนฟิกไคลเอ็นต์ AIX สำหรับการพิสูจน์ตัวตนแบบอิง Kerberos กับ HP-UX 11i Kerberos เวอร์ชัน 2.1

1. ตั้งค่า HP-UX 11i Kerberos Version 2.1 โดยใช้เอกสารคู่มือ HP
2. ถ้าไม่ได้ติดตั้งไคลเอ็นต์ NAS ไว้บนไคลเอ็นต์ AIX, ให้ติดตั้งชุดไฟล์ `krb5.client.rte` จากแพ็กเสริม AIX
3. ใช้คำสั่ง **mkkrb5clnt** ด้วยข้อมูลคอนฟิกูเรชันต่อไปนี้ เพื่อกำหนดคอนฟิกไคลเอ็นต์ AIX Kerberos:

   realm ชื่อ HP Kerberos realm: HPSYS.AUSTIN.IBM.COM

   domain โดเมนเนมของเครื่องที่ทำหน้าที่โฮสต์เซิร์ฟเวอร์ HP-UX Kerberos: austin.ibm.com

   KDC ชื่อโฮสต์ของระบบ HP-UX ที่ทำหน้าที่โฮสต์ KDC: hpsys.austin.ibm.com

   server ชื่อโฮสต์ของเซิร์ฟเวอร์ HP-UX: hpsys.austin.ibm.com

   หมายเหตุ: เนื่องจาก HP-UX และไคลเอ็นต์ AIX NAS อินเตอร์เฟส **kadmin** แตกต่างกัน, ชื่อเซิร์ฟเวอร์ไม่ถูกใช้โดยไคลเอ็นต์ NAS, และแฟล็ก –D ต้องถูกใช้ในคำสั่ง **mkkrb5clnt**

   ต่อไปนี้ คือตัวอย่างของคำสั่ง **mkkrb5clnt** :

   ```
   mkkrb5clnt -r AUSTIN.IBM.COM -d austin.ibm.com\
    -c hpsys.austin.ibm.com -s hpsys.austin.ibm.com -D
   ```

   อ็อพชัน **–D** ในคำสั่ง **mkkrb5clnt** สร้างอ็อพชัน **is_kadmind_compat=no** ในไฟล์ `/etc/security/methods.cfg` และกำหนดคอนฟิกสภาวะแวดล้อมไคลเอ็นต์ Kerberos สำหรับการพิสูจน์ตัวตนกับระบบ ที่ไม่ใช่ AIX ห้ามใช้อ็อพชัน **–D** ในคำสั่ง **mkkrb5clnt** เพื่อกำหนดคอนฟิก สภาวะแวดล้อมไคลเอ็นต์ Kerberos สำหรับการพิสูจน์ตัวตนกับ IBM Network Authentication Service (NAS)

   หมายเหตุ: เมื่อคุณรันคำสั่ง **mkkrb5clnt** stanza ต่อไปนี้ จะถูกเพิ่มในไฟล์ `methods.cfg`

   ```
   KRB5:
           program = /usr/lib/security/KRB5
           program_64 = /usr/lib/security/KRB5_64
           options = authonly,is_kadmind_compat=no

   KRB5files:
           options = db=BUILTIN,auth=KRB5
   ```

   สำหรับข้อมูล เพิ่มเติมเกี่ยวกับ:

   - คำสั่ง **mkkrb5clnt** และแฟล็กที่สามารถใช้ได้ ดูที่คำสั่ง **mkkrb5clnt**
   - ไฟล์ `methods.cfg` โปรดดูที่ไฟล์ `methods.cfg`
4. แก้ไขไฟล์ `krb5.conf` เพื่อให้ประเภทการเข้ารหัส ตรงกับที่ที่ใช้ระหว่างการติดตั้ง HP-UX Kerberos (**krbsetup**) ถ้าใช้ค่า DES-CRC, ให้แก้ไข [libdefaults] stanza ในไฟล์ `krb5.conf` บนไคลเอ็นต์ AIX ดังต่อไปนี้:

   ```
   default_tkt_enctypes = des-cbc-crc

   default_tgs_enctypes = des-cbc-crc
   ```

5. ใช้เครื่องมือ HP-UX **kadmin_ui** เพื่อสร้างหลักการโฮสต์ host/tx3d.austin.ibm.com
6. แตกคีย์และบันทึกลงไฟล์ จากเมนู Edit ในหน้าต่าง Principal Information เลือก Extract Service Key เพื่อแตกคีย์ออก
7. คัดลอกระบบโฮสต์ tx3d.keytab file to the AIX
8. ผสานไฟล์ tx3d.keytab ไปยังไฟล์ /etc/krb5/krb5.keytab บนระบบ AIX ดังต่อไปนี้:

```
ktutil
rkt tx3d.keytab
l
slot KVNO Principal
wkt /etc/krb5/krb5.keytab
q
```

9. ใช้เครื่องมือ HP-UX **kadmin_ui** เพื่อสร้างหลักการ hpuser Kerberos จากนั้นคลิกแท็บ Edit/Attribute เพื่อลบค่าแฟล็ก pw_require
10. สร้างแอคเคาต์ AIX ที่สอดคล้องกับ Kerberos principal บน HP-UX, ดังต่อไปนี้:

```
mkuser registry=KRB5files SYSTEM=KRB5files hpuser
```

11. ใช้คำสั่ง **telnet** เพื่อล็อกอินเข้าสู่ระบบ AIX ด้วยชื่อผู้ใช้และรหัสผ่าน hpuser, และตรวจสอบคอนฟิกูเรชัน ต่อไปนี้คือ ตัวอย่างของเซสชันการล็อกอินที่รวม Kerberos ที่ใช้ KRB5 กับ HP-UX:

```
telnet tx3d

echo $AUTHSTATE
KRB5files

View credentials:
/usr/krb5/bin/klist
```

12. ใช้คำสั่ง **passwd** เพื่อเปลี่ยนรหัสผ่าน

    หมายเหตุ: นโยบาย รหัสผ่าน HP-UX ถูกบังคับใช้ขณะเปลี่ยนรหัสผ่าน อ้างอิง เอกสารคู่มือ HP-UX เพื่อพิจารณาวิธีตั้งค่านโยบายรหัสผ่าน

Kerberos เทียบกับระบบที่ไม่ใช่ AIX: คำถามและข้อมูล การแก้ปัญหา:

หัวข้อนี้ให้คำตอบของคำถามที่เกี่ยวกับไคลเอ็นต์ Kerberos ที่กำลังใช้เซิร์ฟเวอร์ Kerberos บนระบบที่ไม่ใช่ AIX

หมายเหตุ: Microsoft Active Directory Server ถูกใช้ในตัวอย่างต่อไปนี้ อย่างไรก็ตาม ตัวอย่าง เหล่านี้ยังถูกนำไปใช้กับระบบ Solaris และ HP

ซึ่ง ขั้นตอนแรกในการแก้ปัญหา คือทำให้แน่ใจว่าเซิร์ฟเวอร์และ deamons ทั้งหมด กำลังทำงาน

Kerberos เทียบกับระบบที่ไม่ใช่ AIX ใช้ ระบบย่อย syslog เพื่อเขียนข้อมูลเกี่ยวกับข้อผิดพลาดและการดีบัก ในการศึกษาเพิ่มเติมเกี่ยวกับการบันทึกการทำงาน syslog ดูที่ **syslogd** daemon

- ***ฉันจะสร้างผู้ใช้ AIX ได้อย่างไร?***

  สร้าง บัญชีผู้ใช้ AIX (foo) โดย รันคำสั่งต่อไปนี้:

```
mkuser registry=KRB5files SYSTEM=KRB5files foo
```

คำสั่ง **mkuser** สร้างบน AIX คุณต้อง สร้างบัญชีผู้ใช้สำหรับผู้ใช้บน Windows Server Active Directory ที่สอดคล้องกับบัญชีผู้ใช้ AIX ด้วย การสร้าง บัญชีผู้ใช้บน Windows Server Active Directory เป็นการสร้าง principals โดยนัย

- ***ฉันจะลบผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos ออกอย่างไร?***

  ในการ ลบผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos รันคำสั่งต่อไปนี้:

  ```
  rmuser -R KRB5files foo
  ```

  คำสั่ง **rmuser** ลบผู้ใช้ออกจาก AIX คุณต้อง ลบผู้ใช้ออกจาก Windows Server Active Directory ด้วยโดยการใช้เครื่องมือการจัดการผู้ใช้ Windows Server

- ***ฉันจะเปลี่ยนรหัสผ่านของผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos ได้อย่างไร?***

  ในการเปลี่ยนรหัสผ่านของผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos รันคำสั่งต่อไปนี้:

  ```
  passwd -R KRB5files foo
  ```

  ถ้า KDC สนับสนุนคำสั่ง **kpasswd** คำสั่ง **passwd** จะเปลี่ยนรหัสผ่านของ Kerberos principal foo@MYREALM บน เซิร์ฟเวอร์ Kerberos

- ***ฉันจะอนุญาตให้ผู้ใช้เปลี่ยนรหัสผ่านที่หมดอายุบน ไคลเอ็นต์ได้อย่างไร?***

  ในการอนุญาตให้ผู้ใช้เปลี่ยนรหัสผ่านที่หมดอายุบน ไคลเอ็นต์ ให้เพิ่มอ็อพชัน allow_expired_pwd=yes ในไฟล์ `methods.cfg` เมื่ออ็อพชันนี้ถูกตั้งค่าเป็น `yes` ผู้ใช้ที่มีรหัสผ่าน หมดอายุจะได้รับพร้อมต์ให้เปลี่ยนรหัสผ่านที่หมดอายุ ถ้าอ็อพชัน ถูกตั้งค่าเป็น `no` หรือ `not present` ผู้ใช้ไม่สามารถพิสูจน์ตัวตนได้

  ```
  KRB5:
     program = /usr/lib/security/KRB5
     options = authonly,allow_expired_pwd=yes
  ```

- ***ฉันจะแปลงผู้ใช้ AIX เป็นผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos ได้อย่างไร?***

  ในการแปลงผู้ใช้ AIX เป็นผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos ทำต่อไปนี้:

  1. ตรวจสอบว่าผู้ใช้มีบัญชีผู้ใช้บน Windows Server Active Directory โดยการรัน คำสั่งต่อไปนี้:

     ```
     chuser registry=KRB5files SYSTEM=KRB5files foo
     ```

  2. ถ้าผู้ใช้ไม่มีบัญชีผู้ใช้อยู่บน Active Directory ให้สร้าง บัญชีผู้ใช้บน Active Directory และตั้งค่าแอ็ตทริบิวต์ SYSTEM และรีจิสทรี โดยใช้คำสั่ง **chuser** บัญชีผู้ใช้ Active Directory อาจมีชื่อผู้ใช้ไม่เหมือนกับชื่อผู้ใช้ AIX ถ้ามีการใช้ชื่อที่ต่างกันสำหรับ ชื่อผู้ใช้ AIX ให้ใช้แอ็ตทริบิวต์ auth_name เพื่อแม็พกับชื่อ Active Directory

     ```
     chuser registry=KRB5files SYSTEM=KRB5files auth_name=Christopher chris
     ```

- ***ถ้าลืมรหัสผ่าน ฉันควรทำอย่างไร?***

  ถ้า ลืมรหัสผ่าน ต้องเปลี่ยนรหัผ่านโดยผู้ดูแลระบบ Active Directory ผู้ใช้ root ของ AIX ไม่สามารถตั้งค่ารหัสผ่านของ Active Directory Kerberos principal

- ***วัตถุประสงค์ของแอ็ตทริบิวต์ auth_name และ auth_domain คืออะไร?***

  หมายเหตุ: แอ็ตทริบิวต์ เหล่านี้เป็นทางเลือก ถ้าระบบ AIX สนับสนุนชื่อผู้ใช้ที่ยาวมากกว่าแปดอักขระ อาจ ไม่จำเป็นต้องใช้แอ็ตทริบิวต์ auth_name

  แอ็ตทริบิวต์ auth_name และ auth_domain แม็พชื่อผู้ใช้ AIX เป็นชื่อ Kerberos principal บน KDC ตัวอย่าง ถ้าผู้ใช้ AIX ชื่อ chris มีแอ็ตทริบิวต์ auth_name=christopher และ auth_domain=SOMEREALM ดังนั้นชื่อ Kerberos principal คือ christopher@SOMEREALM โดยใช้แอ็ตทริบิวต์ auth_domain การร้องขอจะถูกส่งไปยังชื่อ SOMEREALM realm แทนชื่อ realm ดีฟอลต์ ซึ่งจะอนุญาตให้ผู้ใช้ chris พิสูจน์ตัวตน ไปยัง SOMEREALM realm แทน MYREALM realm ในตัวอย่างนี้ ไฟล์ `krb5.conf` ยังต้องถูกแก้ไขเพื่อรวม ชื่อ SOMEREALM realm

- ***ผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos จะสามารถได้รับการพิสูจน์ตัวตนโดยใช้ การพิสูจน์ตัวตน AIX มาตรฐานได้หรือไม่?***

  ได้ ผู้ใช้ที่ได้รับการพิสูจน์ตัวตน Kerberos สามารถได้รับการพิสูจน์ตัวตนด้วยการพิสูจน์ตัวตน AIX มาตรฐานโดยการทำต่อไปนี้:

  1. ตั้งค่ารหัสผ่าน AIX (`/etc/security/passwd`) โดยใช้คำสั่ง **passwd**:

     ```
     passwd -R files foo
     ```

  2. เปลี่ยนแอ็ตทริบิวต์รีจิสทรีและ SYSTEM ของผู้ใช้ ดังนี้:

     ```
     chuser -R KRB5files registry=files SYSTEM=compat foo
     ```

     คำสั่งนี้ เปลี่ยนการพิสูจน์ตัวตนจาก Kerberos เป็น compat (ซึ่งใช้ รูทีนย่อย crypt) ครั้งหน้า ล็อกอินถูกพยายามดำเนินการโดยผู้ใช้ user foo รหัสผ่านโลคัลจากไฟล์ `/etc/security/passwd` จะถูกใช้

     คุณยังสามารถใช้การพิสูจน์ตัวตนลับเป็นวิธีสำรอง โดยการเปลี่ยนค่าแอ็ตทริบิวต์ SYSTEM เพื่ออนุญาตการพิสูจน์ตัวตนโลคัล เมื่อการพิสูจน์ตัวตน Kerberos ล้มเหลว ดังนี้:

     ```
     chuser -R KRB5files SYSTEM="KRB5files or compat" foo
     ```

- ***ฉันต้องตั้งค่าเซิร์ฟเวอร์ Kerberos บน AIX เมื่อใช้ Windows Server 2000 Kerberos Service หรือไม่?***

  ไม่ คุณไม่จำเป็นต้องตั้งค่า เซิร์ฟเวอร์ Kerberos (KDC) บนไคลเอ็นต์ AIX เนื่องจากผู้ใช้พิสูจน์ตัวตนกับ Active Directory KDC ถ้าคุณวางแผนใช้ AIX Network Authentication Service KDC เป็นเซิร์ฟเวอร์ Kerberos เพื่อวัตถุประสงค์บางประการต้องตั้งค่าเซิร์ฟเวอร์ Kerberos

- ***ฉันควรทำอย่างไรถ้า AIX ไม่ยอมรับรหัสผ่านของฉัน?***

  ถ้า AIX ไม่ยอมรับรหัสผ่าน ให้ทำดังต่อไปนี้:

  - ทำให้แน่ใจว่าไคลเอ็นต์กำลังสื่อสารกับ Windows 2000 Active Directory Server
  - ทำให้แน่ใจว่ารหัสผ่านตรงตามข้อกำหนดของทั้ง AIX และ Windows Server 2000 Active Directory โปรดดูที่ เปลี่ยนแปลงนโยบายการแสดงผล สำหรับข้อมูลกฎการเปลี่ยนแปลงนโยบายรหัสผ่าน ใน AIX

  หมายเหตุ: คุณไม่สามารถเปลี่ยนรหัสผ่านสำหรับ Windows Server 2003 Kerberos Service

- ***ฉันควรทำอย่างไรถ้าไม่สามารถล็อกอินเข้าสู่ระบบ?***

  ถ้า คุณไม่สามารถล็อกอินเข้าสู่ระบบ ให้ทำดังต่อไปนี้:

  - บนระบบ Windows ตรวจสอบว่า KDC กำลังทำงานโดยทำดังต่อไปนี้:
    1. ใน Control Panel เลือกไอคอน Administrative Tools
    2. เลือกไอคอน Services
    3. ตรวจสอบว่า Kerberos Key Distribution Center อยู่ในสถานะ ถูกเริ่มทำงาน
  - บนระบบ AIX ตรวจสอบ ว่าไฟล์ `/etc/krb5/krb5.conf` ชี้ไปที่ KDC ถูกต้อง และมีพารามิเตอร์ที่ถูกต้อง
  - บนระบบ AIX ตรวจสอบ ว่าไฟล์ client-keytab มีโฮสต์คีย์ ตัวอย่าง ถ้า ไฟล์ keytab ดีฟอลต์คือ `/etc/krb5/krb5.keytab` รันต่อไปนี้:

    ```
    ktutil
    rkt /etc/krb5/krb5.keytab
    l
    ```

  - ตรวจสอบว่าเอาต์พุตของคำสั่ง **kvno** ที่อยู่ในไฟล์ keytab ตรงกับเอาต์พุตจากคำสั่ง **Ktpass**

- ตรวจสอบว่าถ้าแอ็ตทริบิวต์ auth_name และ auth_domain ถูกตั้งค่า แอ็ตทริบิวต์เหล่านั้นจะอ้างถึงชื่อ principal ที่ถูกต้องบน Active Directory KDC
- ตรวจสอบว่าแอ็ตทริบิวต์ SYSTEM ถูกตั้งค่าสำหรับการล็อกอิน Kerberos
- ตรวจสอบว่ารหัสผ่านยังไม่หมดอายุ

- ***ฉันสามารถปิดใช้งานการตรวจสอบตั๋วที่ใช้ในการให้สิทธิตั๋ว ได้อย่างไร?***

  คุณสามารถปิดใช้งานการตรวจสอบตั๋วที่ใช้ในการให้สิทธิตั๋ว ได้โดยระบุอ็อพชันในไฟล์ /usr/lib/security/methods.cfg ภายใต้ KRB5 stanza ดังนี้:

```
KRB5:
   program = /usr/lib/security/KRB5
   options = tgt_verify=no
KRB5files:
   options = db=BUILTIN,auth=KRB5
```

  ค่าที่เป็นไปได้ สำหรับอ็อพชัน tgt_verify คือ no หรือ false สำหรับ การปิดใช้งานการตรวจสอบตั๋วที่ใช้ในการให้สิทธิตั๋ว และ yes หรือ true สำหรับ การเปิดใช้งานตั๋วที่ใช้ในการให้สิทธิตั๋ว โดยดีฟอลต์ การตรวจสอบ ตั๋วที่ใช้ในการให้สิทธิตั๋วจะถูกเปิดใช้งาน เมื่อคุณตั้งค่าอ็อพชัน tgt_verify เป็น no การตรวจสอบตั๋วที่ใช้ในการให้สิทธิตั๋ว จะถูกปิดใช้งาน และคุณไม่จำเป็นต้องถ่ายโอนคีย์ host-principal การเปลี่ยนแปลงนี้เป็นการกำจัดความต้องการใช้ไฟล์ keytab เพื่อวัตถุประสงค์ในการการพิสูจน์ตัวตนเท่านั้น แอ็พพลิเคชันที่เปิดใช้ Kerberos อื่นๆ อาจต้องการใช้ไฟล์ keytab เพื่อ principals โฮสต์และเซอร์วิส

- ***ฉันควรทำอย่างไรถ้าไม่สามารถล็อกอินได้ เนื่องจากไม่สามารถ ระบุชื่อโฮสต์ และชื่อโฮสต์แบบเต็มล้มเหลว?***

  การตรวจสอบ ตั๋วที่ใช้ในการให้สิทธิตั๋วจำเป็นต้องให้ host/<host_name> principal ถูกสร้างบน KDC ชื่อโฮสต์นี้เป็นชื่อแบบเต็ม ของไคลเอ็นต์ที่มีการดำเนินการพิสูจน์ตัวตน ระบบไคลเอ็นต์ ร้องขอตั๋วโดยใช้ชื่อ principal โฮสต์ host/<host_name> ในบางการตั้งค่า เครื่องไคลเอ็นต์ไม่สามารถจัดหา ชื่อโฮสต์แบบเต็มได้ จึงใช้ชื่อแบบสั้นแทน ในกรณี เช่นนั้น จะเกิดการไม่ตรงกัน การตรวจสอบตั๋วที่ใช้ในการให้สิทธิตั๋ว ล้มเหลว และล็อกอินล้มเหลว ตัวอย่าง ถ้า /etc/hosts มีแต่ชื่อแบบสั้น และไฟล์ /etc/netsvc.conf ระบุ hosts=local,bind, แล้ว ชื่อที่ได้กลับคืนมาจะเป็นชื่อ แบบสั้น

  ในการแก้ไขปัญหาการกำหนดชื่อ ให้ดำดังต่อไปนี้:

  - แก้ไขลำดับการกำหนดชื่อในไฟล์ /etc/netsvc.conf เพื่อให้ส่งค่าชื่อโฮสต์แบบเต็มกลับ ไฟล์ netsvc.conf ระบุการจัดเรียงลำดับของการกำหนดชื่อโฮสต์และ aliases

    ใน ตัวอย่างต่อไปนี้ ตัวกำหนดชื่อใช้เซอร์วิส BIND เพื่อกำหนด ชื่อโฮสต์ ถ้าเซอร์วิส BIND ล้มเหลว ตัวกำหนดชื่อจะใช้ไฟล์ /etc/hosts แทน ถ้าทั้งสองวิธีล้มเหลว ตัวกำหนดชื่อจะใช้ nis

```
hosts=bind,local,nis
```

    ถ้า ใช้วิธีแรกในลำดับการค้นหาต้องเป็น local เปลี่ยนชื่อแบบสั้น (myhost) ในไฟล์ /etc/hosts เป็นชื่อโฮสต์แบบเต็ม (myhost.austin.ibm.com)
  - ถ้าการตรวจสอบตั๋วที่ใช้ในการให้สิทธิตั๋วไม่ต้องใช้ คุณ สามารถดูคำแนะนำการปิดใช้งานการตรวจสอบตั๋วที่ใช้ในการให้สิทธิตั๋วได้ใน *ฉันปิด ใช้งานการตรวจสอบตั๋วที่ใช้ในการให้สิทธิตั๋วได้อย่างไร?*

- ***ทำไมรูทีนย่อย passwdexpired จึงคืนค่า 0 เมื่อรหัสผ่านของผู้ใช้ kerberos หมดอายุในเซิร์ฟเวอร์ kerberos ที่ไม่ใช่ระบบ AIX?***

  รูทีนย่อย **passwdexpired** คืนค่า 0 เนื่องจากข้อมูลการหมดอายุของรหัสผ่านไม่สามารถขอรับข้อมูลได้โดยตรงจากเซิร์ฟเวอร์ kerberos ที่ไม่ใช่ระบบ AIX เป็นผลมาจากความไม่เข้ากัน หรือไม่พร้อมใช้งานของอินเตอร์เฟส **kadmin**

  แฟล็ก **allow_expired_pwd** ในไฟล์ **methods.cfg** อนุญาตให้ AIX รับข้อมูลการหมดอายุของรหัสผ่านโดยใช้อินเตอร์เฟสการพิสูจน์ตัวตนของ kerberos สถานะจริงของข้อมูลการหมดอายุของรหัสผ่านได้รับ ระหว่างการล็อกอิน หรือโดยการเรียกรูทีนย่อย **authenticate** และรูทีนย่อย **passwdexpired**

โมดูล Kerberos

โมดูล Kerberos เป็นส่วนขยายเคอร์เนลที่ใช้โดยโค้ดไคลเอ็นต์ และเซิร์ฟเวอร์ NFS โมดูลอนุญาตให้โค้ดไคลเอ็นต์และเซิร์ฟเวอร์ NFS ประมวลผลฟังก์ชัน integrity และความสมบูรณ์ของข้อความ Kerberos โดยไม่ต้อง เรียกใช้ **gss** daemon

โมดูล Kerberos ถูกโหลดโดย **gss** daemon วิธีการ ที่ใช้จะยึดตาม Network Authentication Service เวอร์ชัน 1.2 ซึ่ง ในทางกลับกันก็ยึดตาม MIT Kerberos

ตำแหน่งโมดูล Kerberos คือ: `/usr/lib/drivers/krb5.ext`

สำหรับข้อมูลที่เกี่ยวข้อง ดูที่ **gss** daemon

ข้อมูลที่เกี่ยวข้อง:

IBM developerWorks Resources บน IBM Network Authentication Service และเทคโนโลยีที่เกี่ยวข้องสำหรับ AIX

Remote authentication dial-in user service server

IBM's Remote Authentication Dial-In User Service (RADIUS) คือโปรโตคอลเข้าถึงเครือข่ายที่ออกแบบมาสำหรับการพิสูจน์ตัวตน การอนุญาต และการจัดการบัญชีผู้ใช้ ถือเป็นโปรโตคอลที่อิงตามพอร์ตที่กำหนดการสื่อสาร ระหว่าง Network Access Servers (NAS) และเซิร์ฟเวอร์การพิสูจน์ตัวตนและการจัดการบัญชีผู้ใช้

NAS ทำหน้าที่เป็นไคลเอ็นต์ของ RADIUS ทรานแซกชันระหว่างไคลเอ็นต์และเซิร์ฟเวอร์ RADIUS ถูกพิสูจน์ตัวตนผ่าน การใช้ *ความลับที่แบ่งใช้* ซึ่งจะไม่ถูกส่งผ่านเน็ตเวิร์ก รหัสผ่าน ผู้ใช้ใดๆ ที่ส่งระหว่างไคลเอ็นต์และเซิร์ฟเวอร์ RADIUS จะถูกเข้ารหัส

ไคลเอ็นต์มีหน้าที่ในการส่งข้อมูลผู้ใช้ไปยังเซิร์ฟเวอร์ RADIUS ที่กำหนดจากนั้นดำเนินการ ตามการตอบกลับที่ส่งคืนมา เซิร์ฟเวอร์ RADIUS มีหน้าที่รับการร้องขอการเชื่อมต่อของผู้ใช้ ทำการพิสูจน์ตัวตน ผู้ใช้ และส่งกลับข้อมูลคอนฟิกูเรชันทั้งหมดที่จำเป็นสำหรับไคลเอ็นต์ เพื่อนำส่งเซอร์วิสให้แก่ผู้ใช้ เซิร์ฟเวอร์ RADIUS สามารถทำหน้าที่เป็น พร็อกซี ไคลเอ็นต์สำหรับเซิร์ฟเวอร์ RADIUS อื่นๆ เมื่อ ข้อมูลพร็อกซีระดับสูงถูกกำหนดคอนฟิก RADIUS ใช้ User Datagram Protocol (**UDP**) เป็นโปรโตคอลการส่งผ่าน

โปรโตคอลการพิสูจน์ตัวตนและการอนุญาต RADIUS เป็นไปตามมาตรฐาน IETF RFC 2865 เซิร์ฟเวอร์ ยังจัดให้มีโปรโตคอลการจัดการบัญชีผู้ใช้ที่กำหนดใน RFC 2866 มาตรฐานอื่นๆ ที่สนับสนุนคือ RFC 2284 (EAP) เป็นส่วนหนึ่งของ RFC 2869 ข้อความการหมดอายุของรหัสผ่าน ของ RFC 2882, MD5-Challenge และ TLS สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ RFCs เหล่านี้ โปรดดูที่ ลิงก์ต่อไปนี้:

IETF RFC 2865

http://www.ietf.org/rfc/rfc2865.txt

RFC 2866

http://www.ietf.org/rfc/rfc2866.txt

RFC 2284

http://www.ietf.org/rfc/rfc2284.txt

RFC 2869

http://www.ietf.org/rfc/rfc2869.txt

RFC 2882

http://www.ietf.org/rfc/rfc2882.txt

คุณยังสามารถดูมาตรฐาน RFC เหล่านี้ทั้งหมดได้ที่ http://www.ietf.org

การติดตั้งเซิร์ฟเวอร์ RADIUS

คุณสามารถติดตั้งเซิร์ฟเวอร์ RADIUS โดยใช้คำสั่ง **installp** หรือ SMIT ซอฟต์แวร์ RADIUS อยู่บนสื่อบันทึกหลัก AIX, และชื่ออิมเมจ คือ radius.base และ bos.msg.<lang>.rte

ถ้าคุณวางแผนที่จะใช้ LDAP directory เป็น ฐานข้อมูลสำหรับเก็บข้อมูลของคุณเพื่อเก็บชื่อผู้ใช้และรหัสผ่าน คุณต้อง ติดตั้ง ldap.server ซอฟต์แวร์ **installp** ต้องถูกติดตั้งบนแต่ละการติดตั้งเซิร์ฟเวอร์ RADIUS

ถ้าคุณ วางแผนที่จะใช้การพิสูจน์ตัวตน EAP-TLS (ตัวอย่าง สำหรับการพิสูจน์ตัวตน ใบรับรองดิจิตัลบนเน็ตเวิร์กไร้สาย) คุณต้องติดตั้ง OpenSSL 0.9.7 หรือใหม่กว่าด้วย และระบุพาธแบบเต็มไปยังไลบรารี libssl.a ในไฟล์คอนฟิกูเรชัน /etc/radius/radiusd.conf

RADIUS daemons สามารถเริ่มทำงานโดยใช้คำสั่ง **radiusctl** เมื่อเริ่มทำงาน จะมีการประมวลผล radiusd หลายตัวกำลังทำงานอยู่โดยแต่ละการประมวลผล สำหรับการทำงานต่อไปนี้:

- การอนุญาต
- การจัดการบัญชีผู้ใช้
- การมอนิเตอร์ daemons อื่นๆ

เมื่อบูตใหม่ daemons ถูกเริ่มทำงานโดยอัตโนมัติที่ การรันระดับ 2 ยกเว้น RADIUS ถูกตังค่าไว้สำหรับ EAP-TLS

ในการเปลี่ยน รูทีนนี้ ให้แก้ไขไฟล์ /etc/rc.d/rc2.d/Sradiusd

หมายเหตุ: ถ้า RADIUS ถูกกำหนดคอนฟิกเพื่อพิสูจน์ตัวตนใบรับรองดิจิตัลโดยใช้ EAP-TLS daemons จะไม่สามารถกำหนดคอนฟิกเพื่อเริ่มทำงานโดยอัตโนมัติได้ เนื่องจาก ผู้ดูแลระบบต้องป้อนวลีรหัสผ่านใบรับรอง ซึ่ง จำเป็นต้องมีการเริ่มทำงานด้วยตนเอง และรีสตาร์ท RADIUS โดยใช้คำสั่ง **radiusctl**

การหยุดทำงานและการรีสตาร์ท RADIUS

คุณต้องหยุดทำงานและรีสตาร์ท **radiusd** daemons เมื่อใดก็ตามที่มี การเปลี่ยนแปลงไฟล์คอนฟิกูเรชัน /etc/radius/radiusd.conf ของเซิร์ฟเวอร์ RADIUS หรือไฟล์การอนุญาตดีฟอลต์ /etc/radius/authorization/default.policy หรือ/etc/radius/authorization/default.auth โดยสามารถจัดการได้จาก SMIT หรือจากบรรทัดคำสั่ง

เมื่อต้องการเริ่มทำงาน รีสตาร์ท และหยุดทำงานเซิร์ฟเวอร์ RADIUS ใช้คำสั่ง ต่อไปนี้:

```
radiusctl start
radiusctl restart
radiusctl stop
```

การหยุดทำงานและการเริ่มทำงาน RADIUS เป็นสิ่งจำเป็นเนื่องจาก daemon ต้องสร้างตารางหน่วยความจำของแอ็ตทริบิวต์ดีฟอลต์ทั้งหมด ที่มีอยู่ในไฟล์คอนฟิกูเรชันข้างต้น หน่วยความจำที่แบ่งใช้ถูกใช้สำหรับผู้ใช้โลคัล แต่ละรายและตารางผู้ใช้โลคัลถูกสร้างขึ้นตอนทำการเตรียมข้อมูล daemon เพื่อเหตุผลด้านผลการทำงาน

คุณลักษณะตามต้องการ:

คุณสามารถเริ่ม daemon การพิสูจน์ตัวตน RADIUS และเซิร์ฟเวอร์การจัดการบัญชีผู้ใช้ หลาย daemons ได้เท่าที่ต้องการ

แต่ละเซิร์ฟเวอร์รอฟังจากพอร์ตต่างกัน ไฟล์ `radiusd.conf` มีมาพร้อมกับหมายเลขพอร์ตดีฟอลต์คือ 1812 สำหรับการพิสูจน์ตัวตนและ 1813 สำหรับการจัดการบัญชีผู้ใช้ เหล่านี้คือหมายเลขพอร์ตที่ IANA กำหนด โดยการอัพเดต `radiusd.conf` สามารถใช้หมายเลขพอร์ตเหล่านี้พร้อมกับพอร์ตอื่นๆ (หลายพอร์ต) ตามที่ต้องการ ขอให้แน่ใจว่าใช้หมายเลขพอร์ตที่ไม่ได้ถูกกำหนดให้แก่เซอร์วิสที่มีอยู่แล้ว เมื่อ มีหลายหมายเลขพอร์ตถูกป้อนเข้ามาในฟิลด์ **Authentication_Ports** และ **Accounting_Ports** ในไฟล์ `radiusd.conf` **radiusd** daemon จะถูกเริ่มทำงานสำหรับแต่ละพอร์ต daemons จะรอฟังจากหมายเลขพอร์ตของตนตามลำดับ

ไฟล์คอนฟิกูเรชัน RADIUS

RADIUS daemon ใช้หลายไฟล์คอนฟิกูเรชัน เวอร์ชันตัวอย่างของไฟล์เหล่านี้มีให้มา ในแพ็กเกจ RADIUS

ไฟล์คอนฟิกูเรชันทั้งหมดมีผู้ใช้ root และกลุ่ม `security` เป็นเจ้าของ คุณสามารถแก้ไขไฟล์คอนฟิกูเรชันทั้งหมด ยกเว้นไฟล์พจนานุกรม ด้วย System Management Interface Tool (SMIT) เซิร์ฟเวอร์ต้อง ถูกรีสตาร์ทก่อนที่การแก้ไขใดๆ ที่ทำกับไฟล์คอนฟิกูเรชันจะ มีผล

ไฟล์ `radiusd.conf`:

ไฟล์ `radiusd.conf` มี พารามิเตอร์การตั้งค่าสำหรับ RADIUS

โดยดีฟอลต์ RADIUS จะค้นหา ไฟล์ `radiusd.conf` ในไดเร็กทอรี `/etc/radius` รายการไฟล์คอนฟิกูเรชันต้องอยู่ในรูปแบบดังแสดงใน ไฟล์ RADIUS ยอมรับ เฉพาะคีย์เวิร์ดและค่าที่ถูกต้อง และใช้ค่าดีฟอลต์ถ้าไม่ใช้คีย์เวิร์ด หรือค่าที่ถูกต้อง เมื่อคุณเรียกทำงาน RADIUS daemons ตรวจสอบ เอาต์พุต SYSLOG สำหรับข้อผิดพลาดพารามิเตอร์คอนฟิกูเรชัน ข้อผิดพลาดคอนฟิกูเรชัน บางอย่างที่นำไปสู่การหยุดทำงานเซิร์ฟเวอร์

ไฟล์นี้ควรได้รับการป้องกันการอ่านและการเขียนอย่างเหมาะสม เนื่องจาก มีผลต่อลักษณะการทำงานของเซิร์ฟเวอร์การพิสูจน์ตัวตนและการจัดการบัญชีผู้ใช้ รวมทั้ง ข้อมูลลับที่อาจมีอยู่ในไฟล์

สำคัญ: ถ้าคุณแก้ไขไฟล์ `radiusd.conf` อย่าเปลี่ยนแปลงลำดับของรายการ แผง SMIT ขึ้นอยู่กับการเรียงลำดับ

ต่อไปนี้คือตัวอย่างของไฟล์ `radiusd.conf`:

```
#-------------------------------------------------------------------#
#              CONFIGURATION FILE                                   #
#                                                                   #
#  By default RADIUS will search for radiusd.conf in the            #
#  /etc/radius directory.                                           #
#                                                                   #
#  Configuration file entries need to be in the below               #
#  formats.  RADIUS will accept only valid "Keyword : value(s)",    #
#  and will use defaults, if "Keyword : value(s)" are not           #
#  present or are in error.                                         #
#                                                                   #
#  It is important to check the syslog output when launching        #
#  the radius daemons to check for configuration parameter          #
#  errors. Once again, not all configuration errors will lead to    #
#  the server stopping.                                             #
```

```
#                                                                     #
#  Lastly, this file should be appropriately read/write protected,    #
#  because it will affect the behavior of authentication and          #
#  accounting, and confidential or secretive material may             #
#  exist in this file.                                                #
#                                                                     #
#  IF YOU ARE EDITING THIS FILE, DO NOT CHANGE THE ORDER OF THE       #
#  ENTRIES IN THIS FILE. SMIT PANELS DEPEND ON THE ORDER.             #
#                                                                     #
#                                                                     #
#---------------------------------------------------------------------#

#---------------------------------------------------------------------#
#            Global Configuration                                     #
#                                                                     #
#   RADIUSdirectory  :  This is the base directory for the RADIUS     #
#                       daemon. The daemon will search this           #
#                       directory for further configuration files.    #
#                                                                     #
#   Database_location : This is the value of where the                #
#                       authentication (user ids & passwords)         #
#                       will be stored and retrieved.                 #
#                       Valid values: Local, LDAP, UNIX               #
#                       UNIX  - User defined in AIX system            #
#                       Local - Local AVL Database using raddbm       #
#                       LDAP  - Central Database                      #
#                                                                     #
#   Local_Database   :  This indicates the name of the local          #
#                       database file to be used.                     #
#                       This field must be completed if the           #
#                       Database location is Local.                   #
#                                                                     #
#   Debug_Level      :  This pair sets the debug level at which       #
#                       the RADIUS server will run.  Appropriate      #
#                       values are 0,3 or 9. The default is 3.        #
#                       Output is directed to location specified      #
#                       by *.debug stanza in /etc/syslog.conf         #
#                                                                     #
#                       Each level increases the amount of messages   #
#                       sent to syslog. For example "9" includes      #
#                       the new messages provided by "9" as well      #
#                       as all messages generated by level 0 and 3.   #
#                                                                     #
#                       0 : provides the minimal output to the        #
#                           syslogd log.  It sends start up           #
#                           and end messages for each RADIUS          #
#                           process.  It also logs error              #
#                           conditions.                               #
#                                                                     #
#                       3 : includes general ACCESS ACCEPT, REJECT    #
#                           and DISCARD messages for each packet.     #
#                           This level provides a general audit       #
#                           trail for authentication.                 #
#                                                                     #
#                       9 : Maximum amount of log data. Specific      #
```

```
#                          values of attributes while a          #
#                          transaction is passing thru           #
#                          processing and more.                  #
#                          [NOT advised under normal operations] #
#                                                                #
#----------------------------------------------------------------#
RADIUSdirectory   : /etc/radius
Database_location : UNIX
Local_Database    : dbdata.bin
Debug_Level       : 3
#----------------------------------------------------------------#
#                Accounting Configuration                        #
#                                                                #
#  Local_Accounting :  When this flag is set to ON or TRUE a file #
#                      will contain a record of ACCOUNTING START  #
#                      and STOP packets received from the Network #
#                      Access Server(NAS).  The default log file  #
#                      is:                                        #
#                      /var/radius/data/accounting                #
#                                                                 #
#  Local_accounting_loc  : /var/radius/data/accounting            #
#                        path and file name of the local          #
#                        accounting data file. Used only if Local_#
#                        Accounting=ON. If the default is         #
#                        changed, then the path and file need to  #
#                        to be created (with proper permissions)  #
#                        by the admin.                            #
#                                                                 #
#----------------------------------------------------------------#
Local_Accounting     : ON
Local_Accounting_loc : /var/radius/data/accounting
#----------------------------------------------------------------#
#     Reply Message Attributes                                   #
#                                                                #
#     Accept_Reply-Message : Sent when the RADIUS server         #
#                            replies with an Access-Accept packet#
#                                                                #
#     Reject_Reply-Message : Sent when the RADIUS server         #
#                            replies with an Access-Reject packet#
#                                                                #
#     Challenge_Reply-Message   : Sent when the RADIUS server    #
#                                 replies with an Access-Challenge#
#                                 packet                         #
#----------------------------------------------------------------#
Accept_Reply-Message :
Reject_Reply-Message :
Challenge_Reply-Message :
Password_Expired_Reply-Message :
#----------------------------------------------------------------#
#    Support Renewal of Expired Password                         #
#                                                                #
#    Allow_Password_Renewal: YES or NO                           #
#                            Setting this attribute to YES allows#
#                            users to update their expired password#
#                            via the RADIUS protocol. This requires#
```

```
#                           the hardware support of              #
#                           Access-Password-Request packets.     #
#----------------------------------------------------------------#
Allow_Password_Renewal  : NO
#----------------------------------------------------------------#
#    Require Message Authenticator in Access-Request             #
#                                                                #
#    Require_Message_Authenticator: YES or NO                    #
#                                   Setting this attribute to YES #
#                                   checks message authenticator #
#                                   in Access-Request packet.If not#
#                                   present, it will discard the #
#                                   packet.                      #
#----------------------------------------------------------------#
Require_Message_Authenticator : NO
#----------------------------------------------------------------#
#       Servers ( Authentication and Accounting )                #
#                                                                #
#   Authentication_Ports : This field indicates on which port(s) #
#                          the authentication server(s) will listen#
#                          on.  If the field is blank an         #
#                          authentication daemon will not be     #
#                          started.                              #
#                          The value field may contain more than #
#                          one value.  Each value is REQUIRED to #
#                          be separated by a comma ',' .         #
#                                                                #
#                          The value field must contain a numeric #
#                          value, like "6666".  In this case a   #
#                          server daemon will listen on "6666".  #
#                                                                #
#   Accounting_Ports     : The same as authentication_Ports.  See #
#                          above definitions.                    #
#                                                                #
# [NOTE] There is no check for port conflicts.  If a server is   #
#        currently running on the specified port the deamon will #
#        error and not run.  Be sure to check the syslog output  #
#        insure that all servers have started without incident.  #
#                                                                #
#                                                                #
# [Example]                                                      #
#   Authentication_Ports  : 1812,6666 (No Space between commas)  #
#                                                                #
#   In the above example a sever will be start for each port     #
#   specified.  In the case                                      #
#                                                                #
#           6666 : port 6666                                     #
#                                                                #
#----------------------------------------------------------------#
Authentication_Ports  : 1812
Accounting_Ports      : 1813
#----------------------------------------------------------------#
#       LDAP Directory User Information                          #
#                                                                #
#   Required if RADIUS is to connect to a LDAP Version 3 Directory #
```

```
#   and the Database_location field=LDAP                            #
#                                                                   #
#   LDAP_User     : User ID which has admin permission to connect   #
#                   to the remote (LDAP) database. This is the      #
#                   the LDAP administrator's DN.                    #
#                                                                   #
#   LDAP_User_Pwd : Password associated with the above User Id      #
#                   which is required to authenticate to the LDAP   #
#                   directory.                                      #
#                                                                   #
#-------------------------------------------------------------------#
LDAP_User        : cn=root
LDAP_User_Pwd    :
#-------------------------------------------------------------------#
#       LDAP Directory Information                                  #
#                                                                   #
#   If the Database_location field is set to "LDAP" then the        #
#   following fields need to be completed.                          #
#                                                                   #
#   LDAP_Server_name     : This field specifies the fully qualified#
#                          host name where the LDAP Version 3       #
#                          Server is located.                       #
#   LDAP_Server_Port     : The TCP port number for the LDAP server  #
#                          The standard LDAP port is 389.           #
#   LDP_Base_DN          : The distinguished name for search start  #
#   LDAP_Timeout         : # seconds to wait for a response from    #
#                          the LDAP server                          #
#   LDAP_Hoplimit        : maximum number of referrals to follow    #
#                          in a sequence                            #
#   LDAP_Sizelimit       : size limit (in entries) for search       #
#   LDAP_Debug_level     : 0=OFF 1=Trace ON                         #
#                                                                   #
#-------------------------------------------------------------------#
LDAP_Server_name          :
LDAP_Server_port          :  389
LDAP_Base_DN              :  cn=aixradius
LDAP_Timeout              :  10
LDAP_Hoplimit             :  0
LDAP_Sizelimit            :  0
LDAP_Debug_level          :  0
#-------------------------------------------------------------------#
#       PROXY RADIUS Information                                    #
#                                                                   #
#                                                                   #
#   Proxy_Allow               :  ON or OFF. If ON, then the server  #
#                                   can proxy packets to realms it  #
#                                   knows of and the following      #
#                                   fields must also be configured. #
#   Proxy_Use_Table           :  ON or OFF. If ON, then the server  #
#                                   can use table for faster        #
#                                   processing of duplicate requests#
#                                   Can be used without proxy ON, but#
#                                   it is required to be ON if      #
#                                   Proxy_Use_Table is set to ON.   #
#   Proxy_Realm_name          :  This field specifies the realm     #
```

```
#                               this server services.            #
#   Proxy_Prefix_delim        :  A list of separators for parsing  #
#                               realm names added as a prefix to  #
#                               the username.  This list must be  #
#                               mutually exclusive to the Suffix  #
#                               delimiters.                       #
#   Proxy_Suffix_delim        :  A list of separators for parsing  #
#                               realm names added as a suffix to  #
#                               the username.  This list must be  #
#                               mutually exclusive to the Prefix  #
#                               delimiters.                       #
#   Proxy_Remove_Hops         :  YES or NO.  If YES then the       #
#                               will remove its realm name, the   #
#                               realm names of any previous hops  #
#                               and the realm name of the next    #
#                               server the packet will proxy to.  #
#                                                                 #
#   Proxy_Retry_count         :  The number of times to attempt    #
#                               to send the request packet.       #
#                                                                 #
#   Proxy_Time_Out            :  The number of seconds to wait     #
#                               in between send attempts.         #
#                                                                 #
#-----------------------------------------------------------------#
Proxy_Allow               :   OFF
Proxy_Use_Table           :   OFF
Proxy_Realm_name          :
Proxy_Prefix_delim        :   $/
Proxy_Suffix_delim        :   @.
Proxy_Remove_Hops         :   NO
Proxy_Retry_count         :   2
Proxy_Time_Out            :   30
#-----------------------------------------------------------------#
#    Local Operating System Authentication Configuration          #
#                                                                 #
#  UNIX_Check_Login_Restrictions  : ON or OFF. If ON, during      #
#                                   local operating system authen- #
#                                   tication, a call to           #
#                                   loginrestrictions() will be   #
#                                   made to verify the user has   #
#                                   no local login restrictions.  #
#                                                                 #
#-----------------------------------------------------------------#
UNIX_Check_Login_Restrictions : OFF
#-----------------------------------------------------------------#
#    Global IP Pooling Flag                                       #
#                                                                 #
#  Enable_IP_Pool : ON or OFF. If ON, then RADIUS Server will do  #
#                   IP address assignment from a pool of addresses #
#                   defined to the RADIUS server.                 #
#                                                                 #
#-----------------------------------------------------------------#
Enable_IP_Pool            :   OFF
#-----------------------------------------------------------------#
# Send Accept MA: ON or OFF. Some NAS's dislike it if Message     #
```

```
#                 Authenticators (MA's) are present in an ACCEPT   #
#                 message. Use this option to disable sending MA   #
#                 when sending an ACCEPT.                          #
#                                                                  #
# NOTE: Sometimes these same NAS's do not like custom ACCEPT       #
# messages either.                                                 #
#                                                                  #
#------------------------------------------------------------------#
Send_Accept_MA : ON
#------------------------------------------------------------------#
#                                                                  #
# Maximum_Threads : The number of threads that will get            #
#                   spawned to handle authentication               #
#                   requests. If nothing is specified              #
#                   RADIUS defaults to 10.                         #
#                                                                  #
#------------------------------------------------------------------#
Maximum_Threads : 99
#------------------------------------------------------------------#
#                                                                  #
# EAP_Conversation_Timeout : The number of seconds to wait         #
#                            before a conversation becomes         #
#                            stale and gets deleted.               #
#                                                                  #
# NOTE: This prevents Denial-of-Service (DoS) attacks on the       #
#       RADIUS Authentication Server. You may need to increase     #
#       the value of this timeout if your network has high         #
#       latency.                                                   #
#                                                                  #
#------------------------------------------------------------------#
EAP_Conversation_Timeout : 30
#------------------------------------------------------------------#
# Global EAP-TLS (eap-tls) Configuration Settings:                 #
#                                                                  #
# Examples:                                                        #
#                                                                  #
# Enable_EAP-TLS : ON or OFF. If ON, then the server               #
#                  can use OpenSSL to authenticate users           #
#                  using EAP-TLS. These users must first           #
#                  have an EAP authentication type of 13           #
#                  (or EAP-TLS). This setting is found in          #
#                  smitty, using: 'smitty rad_conf_users'          #
#                                                                  #
# NOTE: The following attributes below are completely ignored      #
#       if the above 'Enable_EAP' attribute is not 'ON'.           #
#                                                                  #
# OpenSSL_Library : /opt/freeware/lib/libssl.a(libssl.so.0.9.7)    #
# OpenSSL_Ciphers : ALL:!ADH:RC4+RSA:+SSLv2:@STRENGTH              #
# RootCA_Dir          : /etc/radius/tls                            #
# RootCA_File         : /etc/radius/tls/cacert.pem                 #
# Server_Cert_File    : /etc/radius/tls/cert-srv.pem               #
# Server_PrivKey_File : /etc/radius/tls/cert-srv.pem               #
# Server_CRL_File     : /etc/radius/tls/crl.pem                    #
#                                                                  #
# NOTE: Server_Cert_File and Server_PrivKey_File can be the        #
```

```
#       same file if the file is of the following format (but      #
#       in any order):                                             #
#                                                                  #
#       -----BEGIN RSA PRIVATE KEY-----                            #
#       Proc-Type: 4,ENCRYPTED                                     #
#       <rsa private key data here>                                #
#       -----END RSA PRIVATE KEY-----                              #
#       -----BEGIN CERTIFICATE-----                                #
#       <certificate data here>                                    #
#       -----END CERTIFICATE-----                                  #
#                                                                  #
#------------------------------------------------------------------#
Enable_EAP-TLS      : ON
OpenSSL_Library     : /opt/freeware/lib/libssl.a(libssl.so.0.9.7)
OpenSSL_Ciphers     : ALL:!ADH:RC4+RSA:+SSLv2:@STRENGTH
RootCA_Dir          : /etc/radius/tls
RootCA_File         : /etc/radius/tls/radiusdcacert.pem
Server_Cert_File    : /etc/radius/tls/cert-srv.pem
Server_PrivKey_File : /etc/radius/tls/cert-srv.pem
Server_CRL_File     :
```

วิธีการพิสูจน์ตัวตน EAP สำหรับผู้ใช้แต่ละรายสามารถตั้งค่าด้วย SMIT ในการตั้งค่าวิธี EAP สำหรับผู้ใช้แต่ละ ราย ดำเนินขั้นตอนต่อไปนี้:

```
Radius Server
     -> Configure users
          -> Local Database
             LDAP Directory
                  -> Add a user
                     Change/Show Characteristics of a user
                          ->
                          Login User ID [ ]
                          EAP Type [0 2 4]
                          Password Max Age
```

เมื่อเลือก EAP Type จะมีตัวเลือกต่อไปนี้:

0 None

2 MD5 – Challenge

4 TLS

วิธี EAP ที่เลือกถูกเปรียบเทียบกับลำดับวิธีการพิสูจน์ตัวตน ที่ถูกตั้งค่าในไฟล์ `radiusd.conf` เพื่อดำเนินการพิสูจน์ตัวตน

ไฟล์ /etc/radius/clients:

ไฟล์ `clients` ประกอบด้วยรายชื่อไคลเอ็นต์ ที่ได้รับอนุญาตให้สร้างการร้องขอของเซิร์ฟเวอร์ RADIUS

โดยทั่วไป สำหรับแต่ละไคลเอ็นต์ NAS หรือ AP คุณต้องป้อน IP แอดเดรสของไคลเอ็นต์ ร่วมกับความลับที่แบ่งใช้ระหว่างเซิร์ฟเวอร์ RADIUS และไคลเอ็นต์ และ *poolname* ทางเลือกสำหรับ IP pooling

ไฟล์ประกอบด้วยรายการในรูปแบบต่อไปนี้:

```
<Client IP Address>    <Shared Secret>    <Pool Name>
```

รายการ ตัวอย่างจะแสดงดังนี้:

```
10.10.10.1      mysecret1      floor6
10.10.10.2      mysecret2      floor5
```

ความลับที่แบ่งใช้คือสตริงอักขระที่ถูกกำหนดคอนฟิกไว้บนทั้งไคลเอ็นต์ฮาร์ดแวร์ และบนเซิร์ฟเวอร์ RADIUS ความยาวสูงสุดของความลับที่แบ่งใช้ คือ 256 ไบต์ และเป็นแบบคำนึงถึงขนาดตัวพิมพ์ ความลับที่แบ่งใช้ไม่ถูกส่งในแพ็กเก็ต RADIUS ใดๆ และไม่ถูกส่งบนเน็ตเวิร์ก ผู้ดูแลระบบ ต้องตรวจสอบให้แน่ใจว่าความลับที่แน่นอนถูกกำหนดคอนฟิกไว้ทั้งสองฝั่ง (ไคลเอ็นต์และเซิร์ฟเวอร์ RADIUS) ความลับที่แบ่งใช้ถูกใช้เพื่อการเข้ารหัสข้อมูลรหัสผ่านผู้ใช้ และสามารถใช้ในการยืนยัน integrity ข้อความโดยการใช้แอ็ตทริบิวต์ Message Authentication

ความลับที่แบ่งใช้ของแต่ละไคลเอ็นต์ความเป็นค่าเฉพาะในไฟล์ `/etc/radius/clients` และ เช่นเดียวกับรหัสผ่านที่ดี คือวิธีดีที่สุดคือใช้การผสมกันของตัวอักษรพิมพ์ใหญ่/ตัวอักษรพิมพ์เล็ก, ตัวเลข และสัญลักษณ์ในความลับนั้น เมื่อต้องการเก็บรักษาความลับที่แบ่งใช้ให้ปลอดภัย ให้มีความยาวอย่างน้อย 16 อักขระ ไฟล์ `/etc/radius/clients` สามารถแก้ไขโดยใช้ SMIT ความลับที่แบ่งใช้ควรเปลี่ยนเป็นประจำเพื่อป้องกัน การโจมตีโดยใช้คำในพจนานุกรม

poolname คือชื่อของพูดที่โกลบอล IP แอดเดรส ถูกจัดสรรระหว่างการแปลแบบไดนามิก ผู้ดูแลระบบสร้าง *poolname* เมื่อติดตั้งเซิร์ฟเวอร์ RADIUS การใช้แผง SMIT *poolname* จะถูกเพิ่มจาก **Configure Proxy Rules** > **IP Pool** > **Create an IP Pool** โดยถูกใช้ระหว่าง IP pooling ด้านเซิร์ฟเวอร์

ไฟล์ /etc/radius/dictionary:

ไฟล์ `dictionary` มีคำอธิบายของ แอ็ตทริบิวต์ที่ถูกนิยามโดยโปรโตคอล RADIUS และสนับสนุนโดย AIX RADIUS Server

ไฟล์ถูกใช้โดย RADIUS daemon เมื่อทำการตรวจสอบความถูกต้องและการสร้างข้อมูลแพ็กเก็ต แอ็ตทริบิวต์ที่ผู้จำหน่ายระบุก็ควรถูกเพิ่มในที่นี้ ไฟล์พจนานุกรมสามารถแก้ไขโดยใช้ เอดิเตอร์ใดๆ ไม่มีอินเตอร์เฟส SMIT

ต่อไปนี้คือส่วนหนึ่งของตัวอย่างไฟล์พจนานุกรม:

```
#############################################################################
#                                                                           #
#   This file contains dictionary translations for parsing                  #
#   requests and generating responses.  All transactions are                #
#   composed of Attribute/Value Pairs.  The value of each attribute         #
#   is specified as one of 4 data types.  Valid data types are:             #
#                                                                           #
#   string - 0-253 octets                                                   #
#   ipaddr - 4 octets in network byte order                                 #
#   integer - 32 bit value in big endian order (high byte first)            #
#   date - 32 bit value in big endian order - seconds since                 #
#                                   00:00:00 GMT,  Jan.  1,  1970           #
#                                                                           #
#   Enumerated values are stored in the user file with dictionary           #
#   VALUE translations for easy administration.                             #
#                                                                           #
#   Example:                                                                #
#                                                                           #
#   ATTRIBUTE         VALUE                                                 #
#   ---------------   -----                                                 #
```

```
#   Framed-Protocol = PPP                                                  #
#   7               = 1     (integer encoding)                             #
#                                                                          #
############################################################################
ATTRIBUTE       User-Name                       1       string
ATTRIBUTE       User-Password                   2       string
ATTRIBUTE       CHAP-Password                   3       string
ATTRIBUTE       NAS-IP-Address                  4       ipaddr
ATTRIBUTE       NAS-Port                        5       integer
ATTRIBUTE       Service-Type                    6       integer
ATTRIBUTE       Framed-Protocol                 7       integer
ATTRIBUTE       Framed-IP-Address               8       ipaddr
ATTRIBUTE       Framed-IP-Netmask               9       ipaddr
ATTRIBUTE       Framed-Routing                  10      integer
ATTRIBUTE       Filter-Id                       11      string
.
.
.
```

หมายเหตุ: แอ็ตทริบิวต์ใดๆ ที่ถูกนิยามอยู่ในไฟล์ `default.policy` หรือไฟล์ `default.auth` (หรือสำหรับไฟล์ `user_id.policy` หรือ `user_id.auth`), ต้องเป็นแอ็ตทริบิวต์ RADIUS ตามที่นิยามไว้ใน ไฟล์คอนฟิกูเรชันพจนานุกรม AIX โลคัล ถ้าไม่พบแอ็ตทริบิวต์ในพจนานุกรม **radiusd** daemon จะไม่โหลดและข้อความแสดงความผิดพลาดถูกบันทึก

หมายเหตุ: ถ้า พจนานุกรม ไฟล์ `default.policy` และไฟล์ `default.auth` สำหรับระบบถูกแก้ไข คุณต้องรีสตาร์ท RADIUS daemons โดยการรัน คำสั่ง **stopsrc** และคำสั่ง **startsrc** หรือโดยใช้ SMIT

ไฟล์ /etc/radius/proxy:

ไฟล์ `/etc/radius/proxy` คือไฟล์คอนฟิกูเรชัน ที่สนับสนุนคุณลักษณะพร็อกซี ไฟล์นี้แม็พขอบเขตที่ทราบที่ พร็อกซีเซิร์ฟเวอร์สามารถส่งต่อแพ็กเก็ตไป

ไฟล์ `/etc/radius/proxy` ใช้ IP address ของ เซิร์ฟเวอร์ที่จัดการแพ็กเก็ตสำหรับขอบเขตนั้น และความลับที่แบ่งใช้ระหว่างสองเซิร์ฟเวอร์

ไฟล์มีฟิลด์ต่อไปนี้ที่คุณสามารถแก้ไขด้วย SMIT:

- **Realm Name**
- **Next Hop IP address**
- **Shared Secret**

ต่อไปนี้คือตัวอย่างของไฟล์ `/etc/radius/proxy`:

หมายเหตุ:

ความลับ ที่แบ่งใช้ควรยาว 16 อักขระ ความลับเดียวกันที่แบ่งใช้ต้อง ถูกกำหนดคอนฟิกบนเซิร์ฟเวอร์ RADIUS ฮ็อพถัดไป

```
# @(#)91 1.3  src/rad/usr/sbin/config_files/proxy, radconfig, radius530 1/23/04 13:11:14
############################################################################
#                                                                          #
#       This file contains a list of proxy realms which are                #
#       authorized to send/receive proxy requests/responses to/from        #
```

```
#       this RADIUS server and their Shared secret used in encryption.#
#                                                                      #
#       The first field is the name of the realm of the remote RADIUS #
#       Server.                                                        #
#                                                                      #
#       The second field is a valid IP address for the remote RADIUS  #
#       Server.                                                        #
#                                                                      #
#       The third column is the shared secret associated with this    #
#       realm.                                                         #
#                                                                      #
#       NOTE: This file contains sensative security information and   #
#             precautions should be taken to secure access to this     #
#             file.                                                    #
#                                                                      #
########################################################################
# REALM NAME               REALM IP               SHARED SECRET
#---------------       ------------------     ----------------------
#  myRealm                10.10.10.10              sharedsec
```

การพิสูจน์ตัวตน

โดยทั่วไปการพิสูจน์ตัวตนใช้ชื่อและรหัสผ่านที่คงที่ และโดยปกติเกิดขึ้นเมื่อผู้ใช้ล็อกอินเข้าสู่เครื่อง หรือมีการร้องขอเซอร์วิสเป็นครั้งแรก RADIUS ขึ้นอยู่กับฐานข้อมูลการพิสูจน์ตัวตน เพื่อเก็บ ID ผู้ใช้ รหัสผ่าน และข้อมูลอื่นๆ

สำหรับการพิสูจน์ตัวตนผู้ใช้ เซิร์ฟเวอร์สามารถใช้ฐานข้อมูลโลคัล รหัสผ่าน UNIX หรือ LDAP ตำแหน่งที่ตั้ง ฐานข้อมูลถูกตั้งค่าในไฟล์ `/etc/radius/radiusd.conf` ของเซิร์ฟเวอร์ ระหว่างการติดตั้ง หรือโดยการอัพเดตไฟล์ผ่าน SMIT โปรดดูที่ “ไฟล์คอนฟิกูเรชัน RADIUS” ในหน้า 350 สำหรับข้อมูลเพิ่มเติม เกี่ยวกับไฟล์คอนฟิกูเรชัน RADIUS

ฐานข้อมูลผู้ใช้:

ซอฟต์แวร์ RADIUS สามารถใช้ฐานข้อมูลอื่นเพื่อเก็บข้อมูลผู้ใช้

คุณสามารถใช้ฐานข้อมูลโลคัล UNIX หรือ LDAP เพื่อเก็บข้อมูลผู้ใช้

UNIX:

อ็อพชันการพิสูจน์ตัวตน UNIX อนุญาตให้ RADIUS ใช้วิธีการพิสูจน์ตัวตนของระบบโลคัล เพื่อพิสูจน์ตัวตนผู้ใช้

ในการใช้การพิสูจน์ตัวตน UNIX โลคัล แก้ไขฟิลด์ **database_location** ของไฟล์ `radiusd.conf` หรือเลือก `UNIX` ในฟิลด์ `Database Location` ของ SMIT วิธีการพิสูจน์ตัวตนจะเรียกใช้ UNIX **authenticate( )** application program interface (API) เพื่อพิสูจน์ตัวตน ID ผู้ใช้ และรหัสผ่าน รหัสผ่าน ถูกบันทึกในไฟล์ข้อมูลที่ UNIX ใช้ เช่น `/etc/passwords` ID ผู้ใช้และรหัสผ่านถูกสร้างโดยใช้คำสั่ง **mkuser** หรือผ่าน SMIT

ในการใช้ฐานข้อมูล UNIX เลือก `UNIX` ในฟิลด์ **Database Location** ดังแสดงด้านล่าง:

```
    Configure Server

RADIUS Directory             /etc/radius
*Database Location           [UNIX]
Local AVL Database File Name [dbdata.bin]
Local Accounting             [ON]

Debug Level                  [3]
.
.
.
```

Local:

ถ้าฟิลด์ **database_location** ของไฟล์ `radiusd.conf` หรือรายการ Database Location ของ SMIT มีคำว่า `Local` RADIUS Server จะใช้ `/etc/radius/dbdata.bin` เป็น ตำแหน่งสำหรับเก็บ ID ผู้ใช้และรหัสผ่านทั้งหมด

ฐานข้อมูลผู้ใช้โลคัลเป็นแฟล็ตไฟล์ที่มีข้อมูล ID ผู้ใช้ และ รหัสผ่าน รหัสผ่านถูกบันทึกในรูปแบบที่ถูกแฮช การแฮชเป็นเทคนิคการกำหนดแอดเดรส ที่เร็วที่สุดสำหรับการเข้าถึงข้อมูลโดยตรงในพื้นที่หน่วยความจำ ในการเพิ่ม ลบ หรือแก้ไขรหัสผ่านผู้ใช้ให้รันคำสั่ง **raddbm** หรือใช้ SMIT เมื่อ **radiusd** daemon เริ่มทำงาน จะอ่านไฟล์ `radiusd.conf` และโหลด ID ผู้ใช้และรหัสผ่านเข้ามาไว้ในหน่วยความจำ

หมายเหตุ: ความยาว ID ผู้ใช้สูงสุด คือ 253 อักขระ และความยาวรหัสผ่านสูงสุดคือ 128 อักขระ

ในการใช้ฐานข้อมูลผู้ใช้โลคัล เลือก `Local` ในฟิลด์ **Database Location** ดังแสดงด้านล่าง:

```
    Configure Server

RADIUS Directory             /etc/radius
*Database Location           [Local]
Local AVL Database File Name [dbdata.bin]
Local Accounting             [ON]

Debug Level                  [3]
.
.
.
```

LDAP:

RADIUS สามารถ ใช้ LDAP เวอร์ชัน 3 เพื่อเก็บข้อมูลผู้ใช้รีโมต

RADIUS จะใช้ การเรียกใช้ API ของ LDAP เวอร์ชัน 3 เพื่อเข้าถึงข้อมูลผู้ใช้แบบรีโมต การเข้าถึง LDAP เวอร์ชัน 3 จะเกิดขึ้นถ้าฟิลด์ **database_location** ในไฟล์ `/etc/radiusd.conf` ถูกตั้งค่าเป็น `LDAP` และ ชื่อเซิร์ฟเวอร์ ID ผู้ใช้ของผู้ดูแลระบบ LDAP และรหัสผ่านผู้ดูแลระบบ LDAP ถูกตั้งค่า

AIX ใช้ไลบรารีไคลเอ็นต์ LDAP เวอร์ชัน 3 ที่ได้รับการสนับสนุนและทำแพ็คเกจใน IBM Tivoli Directory Server LDAP คือโปรโตคอลที่ปรับขนาดได้ และข้อดีของการใช้ LDAP คือผู้ใช้และข้อมูลที่อยู่ระหว่างดำเนินการสามารถอยู่ในตำแหน่งที่เป็นศูนย์กลาง ช่วยให้ง่ายต่อการจัดการเซิร์ฟเวอร์ RADIUS คุณสามารถ ใช้ยูทิลิตีบรรทัดคำสั่ง **ldapsearch** เพื่อดู ข้อมูลใดๆ ของ RADIUS

รวมทั้ง LDAP ต้องได้รับการตั้งค่าและจัดการก่อนจึงจะสามารถ ใช้สำหรับ RADIUS

เซิร์ฟเวอร์ RADIUS มีไฟล์ LDAP `ldif` เพื่อเพิ่ม RADIUS schema รวมถึง คลาสอ็อบเจ็กต์และแอ็ตทริบิวต์ ในไดเร็กทอรี แต่คุณต้องติดตั้ง และตั้งค่า LDAP

คำต่อท้ายแยกที่สร้างขึ้นเป็นพิเศษสำหรับ RADIUS เพื่อใช้อ็อบเจ็กต์ RADIUS LDAP คำต่อท้ายนี้ คือคอนเทนเนอร์ที่มีชื่อ cn=aixradius และมี สองคลาสอ็อบเจ็กต์ดังอธิบายใน “การตั้งค่าเซิร์ฟเวอร์ RADIUS LDAP”คุณใช้ไฟล์ RADIUS-supplied `ldif` ที่สร้างคำต่อท้ายและ RADIUS schema

เมื่อคุณใช้ LDAP เป็นฐานข้อมูลการพิสูจน์ตัวตนคุณจะได้รับคุณลักษณะ ต่อไปนี้:

1. ฐานข้อมูลผู้ใช้ที่สามารถเห็นและเข้าถึงได้จากเซิร์ฟเวอร์ RADIUS ทั้งหมด
2. รายการผู้ใช้ทีผู้ใช้แอ็คทีฟ
3. คุณลักษณะในการอนุญาตจำนวนล็อกอินสูงสุดต่อหนึ่ง ID ผู้ใช้
4. ประเภท **EAP** ที่สามารถตั้งค่าต่อหนึ่ง ผู้ใช้
5. วันที่รหัสผ่านหมดอายุ

ในการใช้ฐานข้อมูล LDAP ให้เลือก `LDAP` ใน ฟิลด์ **Database Location** ดังแสดงด้านล่าง:

```
    Configure Server

RADIUS Directory             /etc/radius
*Database Location           [LDAP]
Local AVL Database File Name [dbdata.bin]
Local Accounting             [ON]

Debug Level                  [3]
.
.
.
```

ข้อมูลที่เกี่ยวข้อง:

IBM Directory Server

การตั้งค่าเซิร์ฟเวอร์ RADIUS LDAP:

เมื่อการพิสูจน์ตัวตนผู้ใช้ LDAP ถูกตั้งค่า schema ของเซิร์ฟเวอร์ LDAP ต้องถูกอัพเดต ผู้ดูแลระบบ LDAP ต้องเพิ่มแอ็ตทริบิวต์นิยาม AIX RADIUS และคลาสอ็อบเจ็กต์ไปยังไดเร็กทอรี LDAP ก่อนที่จะนิยามผู้ใช้ LDAP RADIUS

คุณต้องเพิ่มคำต่อท้ายเซิร์ฟเวอร์ LDAP คำต่อท้ายสำหรับ RADIUS ชื่อ `cn=aixradius` คำต่อท้ายเป็น distinguished name ที่ระบุรายการบนสุดในลำดับชั้นไดเร็กทอรี

เมื่อเพิ่มคำต่อท้าย ไดเร็กทอรี LDAP จะมีคอนเทนเนอร์ว่างเปล่า *คอนเทนเนอร์*คือรายการว่างที่สามารถใช้ แบ่งพาร์ติชัน namespace คอนเทนเนอร์คล้ายกับไดเร็กทอรี ระบบไฟล์โดยที่สามารถมีรายการไดเร็กทอรีอยู่ภายใต้คอนเทนเนอร์ได้ ข้อมูลโปรไฟล์ผู้ใช้สามารถเพิ่มในไดเร็กทอรี LDAP ผ่าน SMIT ID ผู้ดูแลระบบ LDAP และรหัสผ่านถูกเก็บในไฟล์ `/etc/radius/radiusd.conf` และสามารถถูกตั้งค่าผ่าน SMIT บนเซิร์ฟเวอร์ RADIUS

ในการจัดการข้อมูลที่เก็บในรายการไดเร็กทอรี LDAP schema จะกำหนดคลาสอ็อบเจ็กต์ คลาสอ็อบเจ็กต์ประกอบด้วยชุดของ แอ็ตทริบิวต์ที่จำเป็นและที่เป็นทางเลือก แอ็ตทริบิวต์อยู่ในรูปของคู่ `type=value` ซึ่งประเภทถูกกำหนดโดย identifier อ็อบเจ็กต์เฉพาะ (OID) และ ค่ามีไวยากรณ์ที่กำหนด ทุกรายการในไดเร็กทอรี LDAP คือ instance ของอ็อบเจ็กต์

หมายเหตุ: คลาสอ็อบเจ็กต์โดยตัวเองแล้วไม่ได้กำหนดแผนผังข้อมูลไดเร็กทอรี หรือ namespace นี้เกิดขึ้นต่อเมื่อรายการถูกสร้างขึ้นและ instance ที่ระบุ ของคลาสอ็อบเจ็กต์คือ distinguished name เฉพาะที่กำหนด ตัวอย่าง เมื่อคลาสอ็อบเจ็กต์คอนเทนเนอร์คือ DN เฉพาะที่กำหนด ดังนั้นจะสามารถ เชื่อมโยงกับรายการอื่นสองรายการซึ่งเป็น instances ของหน่วยระดับองค์กร คลาสอ็อบเจ็กต์ ผลที่ได้คือโครงสร้างที่มีลักษณะเป็นแบบแผนผัง หรือ namespace

คลาสอ็อบเจ็กต์เป็นคลาสเฉพาะสำหรับเซิร์ฟเวอร์ RADIUS และถูกนำมาใช้ จากไฟล์ `ldif` บางแอ็ตทริบิวต์เป็น แอ็ตทริบิวต์ LDAP schema ที่มีอยู่และบางแอ็ตทริบิวต์เป็นแอ็ตทริบิวต์เฉพาะสำหรับ RADIUS คลาส อ็อบเจ็กต์ RADIUS ใหม่มีลักษณะเป็นโครงสร้างและเป็นนามธรรม

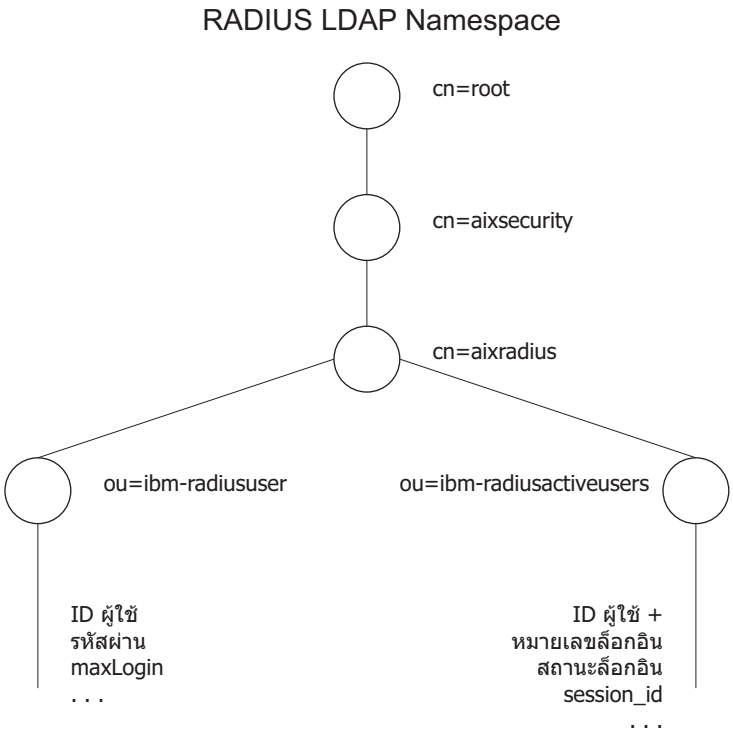
เพื่อวัตถุประสงค์ด้านความปลอดภัย การโยงไปยังเซิร์ฟเวอร์ LDAP ใช้การโยงแบบง่าย หรือการเรียกใช้ SASL API **ldap_bind_s** ซึ่ง จะรวม DN และ CRAM-MD5 ที่เป็นวิธีการพิสูจน์ตัวตน และ รหัสผ่านของผู้ดูแลระบบ LDAP ซึ่งจะส่งส่วนย่อยข้อความ แทนการส่งรหัสผ่านไปบนเน็ตเวิร์ก CRAM-MD5 เป็น กลไกการรักษาความปลอดภัยที่ไม่จำเป็นต้องมีการตั้งค่าพิเศษ ใดไม่ว่าบนด้านใด (ไคลเอ็นต์หรือเซิร์ฟเวอร์)

หมายเหตุ: แอ็ตทริบิวต์ทั้งหมดในคลาสอ็อบเจ็กต์ เป็นค่าเดี่ยว

RADIUS LDAP namespace:

RADIUS LDAP namespace มีคอนเทนเนอร์ `cn=aixradius` เป็นชั้นบนสุดของ ลำดับชั้น ด้านล่างของ `cn=aixradius` มีหน่วยระดับองค์กร (OUs) สองหน่วย OUs เหล่านี้คือคอนเทนเนอร์ที่ช่วยให้รายการเป็นรายการเฉพาะ

รูปต่อไปนี้แสดง RADIUS LDAP schema อย่างชัดเจน รูปนี้ แสดงคอนเทนเนอร์และหน่วยระดับองค์กรทั้งหมดแทนด้วยรูปวงกลม และเชื่อมต่อถึงกันโดยเส้นหรือก้านแยก คอนเทนเนอร์ `aixradius` ตรงกลาง แตกสาขาแยกออกเป็นสองหน่วยระดับองค์กร: `ibm-radiususer` และ `ibm-radiusactiveusers` ด้านล่างของคอนเทนเนอร์ `ibm-radiususer` คือคอนเทนเนอร์ `userid`, `password` และ `maxLogin` ที่แสดง ด้านล่างของคอนเทนเนอร์ `ibmradiusactiveusers` คือคอนเทนเนอร์ `userid` +, `login number`, `login status` และ `session_id` ที่แสดง เหนือคอนเทนเนอร์ `aixradius` คือคอนเทนเนอร์ `aixsecurity` และคอนเทนเนอร์ `root` อยู่ด้านบนสุด



รูปที่ 16. RADIUS LDAP Namespace

ไฟล์ LDAP namespace schema:

ไฟล์ LDAP schema กำหนดคลาสอ็อบเจ็กต์และแอ็ตทริบิวต์เฉพาะ RADIUS สำหรับ LDAP namespace

ไฟล์ LDAP schema ต่อไปนี้อยู่ในไดเร็กทอรี `/etc/radius/ldap`:

`IBM.V3.radiusbase.schema.ldif`

ไฟล์นี้กำหนดคลาสอ็อบเจ็กต์ระดับบนสุดสำหรับเซิร์ฟเวอร์ RADIUS (`cn=aixradius`) ไฟล์ยังสร้างสาขาต่อไปนี้ภายใต้คลาสอ็อบเจ็กต์ `cn=aixradius`:

```
ou=ibm-radiususer
ou=ibm-radiusactiveusers
```

คุณสามารถเพิ่มข้อมูลที่จำเป็น ได้โดยใช้คำสั่งต่อไปนี้:

```
ldapadd -D ldap_admin_id -w password -i /etc/radius/ldap/IBM.V3.radiusbase.schema.ldif
```

คุณ สามารถรันคำสั่งนี้บนระบบเซิร์ฟเวอร์ LDAP หรือคุณสามารถรันแบบ รีโมตด้วยอ็อพชัน **-h** (ชื่อระบบโฮสต์)

`IBM.V3.radius.schema.ldif`

ไฟล์นี้กำหนดแอ็ตทริบิวต์เฉพาะ RADIUS และคลาสอ็อบเจ็กต์

คุณสามารถเพิ่มแอ็ตทริบิวต์ RADIUS ใหม่ และคลาสอ็อบเจ็กต์โดยการพิมพ์คำสั่งต่อไปนี้:

```
ldapmodify -D ldap_admin_id -w password -i /etc/radius/ldap/IBM.V3.radius.schema.ldif
```

คุณ ยังต้องระบุ LDAP เป็นตำแหน่งฐานข้อมูลผ่าน SMIT และป้อนชื่อเซิร์ฟเวอร์ LDAP และรหัสผ่านผู้ดูแลระบบ หลังจาก ทำแล้ว คุณสามารถเพิ่มผู้ใช้ RADIUS LDAP ในไดเร็กทอรีผ่าน SMIT

คลาสอ็อบเจ็กต์โปรไฟล์ผู้ใช้:

โปรไฟล์ผู้ใช้ LDAP ต้องถูกป้อนในระบบก่อนที่เซิร์ฟเวอร์ RADIUS จะสามารถพิสูจน์ตัวตน ผู้ใช้กับระบบ โปรไฟล์มี ID ผู้ใช้ และรหัสผ่าน

อ็อบเจ็กต์โปรไฟล์ผู้ใช้มีข้อมูลเกี่ยวกับบุคคลเป็นรายเฉพาะที่ มีการเข้าถึงเน็ตเวิร์กและมีข้อมูลการพิสูจน์ตัวตน คลาสอ็อบเจ็กต์ **ibm-radiusUserInstance** ถูกเข้าถึงแบบซิงโครไนซ์ด้วยการเรียกใช้ LDAP API จาก daemon ฟิลด์ เฉพาะ ซึ่งเป็นจุดเริ่มต้นของ DN คือ ID ผู้ใช้ ฟิลด์ **MaxLoginCount** จำกัดจำนวนครั้งที่ผู้ใช้ LDAP สามารถล็อกอิน

คลาสอ็อบเจ็กต์รายการล็อกอินที่แอ็คทีฟ:

รายการล็อกอินที่แอ็คทีฟของ LDAP แสดงข้อมูลที่มีรายละเอียด เกี่ยวกับผู้ใช้ที่ล็อกอินในขณะนี้

มีหลายเร็กคอร์ดต่อหนึ่งผู้ใช้ที่เริ่มต้นเร็กคอร์ดด้วย `login_number = 1` จำนวน `MaxLoginCount` สูงสุดคือ 5 ID เซสชัน ถูกนำมาจากข้อความ RADIUS `start_accounting` เร็กคอร์ดที่สำเร็จบางส่วนถูกสร้างขึ้นเมื่ออ็อบเจ็กต์ **ibm-radiusUserInstance** ถูกสร้างขึ้น นี้หมายความว่าฟิลด์ส่วนใหญ่ว่างก่อนที่แพ็กเก็ตการจัดการบัญชีผู้ใช้ RADIUS จะได้รับ หลังจากได้รับข้อความ RADIUS `start_accounting` อ็อบเจ็กต์ **ibm-radiusactiveusers** อัพเดต เพื่อระบุว่าขณะนี้ผู้ใช้ที่ล็อกอินในขณะนี้ และข้อมูลเซสชันเฉพาะ ถูกเขียนลงในหมายเลขล็อกอินที่ถูกต้อง หลังจากได้รับข้อความ `stop_accounting` ข้อมูลในเร็กคอร์ดรายการล็อกอินที่แอ็คทีฟจะถูกลบออก เร็กคอร์ดล็อกอินที่แอ็คทีฟถูกอัพเดตเพื่อแสดงผู้ใช้ที่ล็อกออกจากระบบ ในขณะนี้ หมายเลขเซสชันในข้อความการจัดการบัญชีผู้ใช้เริ่มต้นและสิ้นสุด เป็นหมายเลขเฉพาะเดียวกัน คลาสอ็อบเจ็กต์จะถูกเข้าถึงแบบซิงโครไนซ์ใน การเรียกใช้ LDAP API

Password authentication protocol:

Password Authentication Protocol (**PAP**) จัดให้มีการรักษาความปลอดภัยโดยการโค้ดรหัสผ่านของผู้ใช้ด้วยอัลกอริทึมการแฮช MD5 ของค่าที่ใช้สร้างทั้งไคลเอ็นต์และเซิร์ฟเวอร์

โดยทำงานดังนี้:

1. ในแพ็กเก็ตที่มีรหัสผ่านผู้ใช้ ฟิลด์ Authentication จะมี หมายเลขสุ่มฐานแปด 16 ตัวที่เรียกว่า Request Authenticator
2. Request Authenticator และความลับที่แบ่งใช้ของไคลเอ็นต์ถูกเก็บไว้ใน MD5 hash ผลลัพธ์คือ hash ฐานแปด 16 ตัว
3. รหัสผ่านที่ผู้ใช้ระบุถูกเติมให้เป็นค่าฐานแปด 16 ตัวด้วยค่า Null
4. hash จากขั้นตอน 2 คือ XORed (Exclusive-OR) ที่มีรหัสผ่าน ที่ถูกเติม นี้คือข้อมูลที่ส่งในแพ็กเก็ตเป็นแอ็ตทริบิวต์ *user_password*
5. เซิร์ฟเวอร์ RADIUS คำนวณ hash เดียวกันกับในขั้นตอน 2
6. hash นี้ถูก XOR กับข้อมูลแพ็กเก็ตจากขั้นตอน 4 เพื่อทำการเรียกคืน รหัสผ่าน

Challenge handshake authentication protocol:

RADIUS ยังสนับสนุนการใช้ **CHAP** ของ PPP สำหรับ การปกป้องรหัสผ่าน

ด้วย CHAP รหัสผ่านของผู้ใช้ไม่ถูกส่งไปบนเน็ตเวิร์ก แต่ MD5 hash ของรหัสผ่านจะถูกส่งไปแทน และเซิร์ฟเวอร์ RADIUS จะสร้าง hash ขึ้นใหม่จากข้อมูลของผู้ใช้ ประกอบด้วยรหัสผ่านที่เก็บไว้ จากนั้นเปรียบเทียบ ค่านี้กับค่าที่ส่งโดยไคลเอ็นต์

Extensible authentication protocol:

Extensible Authentication Protocol (**EAP**) คือโปรโตคอลที่ออกแบบมาเพื่อสนับสนุนวิธีการพิสูจน์ตัวตนหลายวิธี

EAP ระบุโครงสร้างการสื่อสารเพื่อการพิสูจน์ตัวตน ระหว่างไคลเอ็นต์และเซิร์ฟเวอร์การพิสูจน์ตัวตน โดยไม่ต้องมีการกำหนด เนื้อหาของข้อมูลการพิสูจน์ตัวตน เนื้อหานี้ถูกกำหนดโดยวิธี **EAP** ที่เจาะจง ที่ใช้สำหรับการพิสูจน์ตัวตน วิธี **EAP** ทั่วไป ประกอบด้วย:

- MD5-challenge
- One-time password
- Generic token card
- Transport layer security (TLS)

RADIUS ใช้ประโยชน์ของ **EAP** โดยการระบุแอ็ตทริบิวต์ RADIUS ที่ใช้เพื่อถ่ายโอนข้อมูล **EAP** ระหว่าง เซิร์ฟเวอร์ RADIUS และไคลเอ็นต์ ข้อมูล **EAP** นี้ สามารถถูกส่งโดยตรงโดยเซิร์ฟเวอร์ RADIUS ไปยังเซิร์ฟเวอร์ back-end ที่นำใช้วิธีการพิสูจน์ตัวตน **EAP** ที่แตกต่างกัน

เซิร์ฟเวอร์ AIX RADIUS สนับสนุน วิธี EAP-TLS และ MD5-challenge EAP เท่านั้น

คุณสามารถตั้งค่าวิธี EAP ที่ใช้พิสูจน์ตัวตนผู้ใช้ ที่ระดับผู้ใช้ได้ โดยการตั้งค่าในรายการของผู้ใช้ในฐานข้อมูลโลคัล หรือ LDAP

โดยดีฟอลต์ EAP จะถูกปิดทำงานสำหรับผู้ใช้แต่ละราย

การอนุญาต

RADIUS อนุญาต ให้กำหนดแอ็ตทริบิวต์การอนุญาตรายผู้ใช้ในไฟล์นโยบายการอนุญาต `default.auth` และ `default.policy`

แอ็ตทริบิวต์การอนุญาตคือแอ็ตทริบิวต์โปรโตคอล RADIUS ที่ถูกต้องที่ถูกระบุใน RFC และกำหนดอยู่ในไฟล์ `/etc/radius/dictionary` การอนุญาตเป็นทางเลือกและขึ้นอยู่กับวิธีที่ NAS ฮาร์ดแวร์หรือ จุดการเข้าถึงถูกตั้งค่า คุณต้องตั้งค่าแอ็ตทริบิวต์การอนุญาต ถ้าจำเป็นต้องใช้ การอนุญาตจะเกิดขึ้นหลังการพิสูจน์ตัวตนสำเร็จ แล้วเท่านั้น

นโยบายคือคู่ของค่าแอ็ตทริบิวต์ผู้ใช้ที่สามารถตั้งค่าได้ ที่สามารถใช้เพื่อควบคุม วิธีที่ผู้ใช้เข้าถึงเน็ตเวิร์ก นโยบายสามารถถูกกำหนดเป็น โกลบอลสำหรับเซิร์ฟเวอร์ RADIUS หรือแบบเจาะจงผู้ใช้

ไฟล์คอนฟิกูเรชันการอนุญาตสองไฟล์ที่มีให้มา: `/etc/radius/authorization/default.auth` และ `default.policy` ไฟล์ `default.policy` ใช้เพื่อจับคู่ แพ็กเก็ตการร้องขอเพื่อเข้าถึงที่มีเข้ามา ไฟล์มีคู่ค่าแอ็ตทริบิวต์ ที่เริ่มแรกจะว่างเปล่าย และต้องถูกตั้งค่าเพื่อให้ได้ การตั้งค่าที่ต้องการ หลังการพิสูจน์ตัวตน นโยบายจะพิจารณาแพ็กเก็ต การยอมรับการเข้าถึง หรือการปฏิเสธการเข้าถึงที่จะถูกส่งกลับไปยังไคลเอ็นต์

ผู้ใช้แต่ละรายยังมีไฟล์ *user_id*`.policy` ถ้าผู้ใช้มีไฟล์นโยบายเฉพาะที่สร้างขึ้นสำหรับ ID ผู้ใช้ที่เจาะจง ดังนั้นแอ็ตทริบิวต์ของไฟล์นั้นจะถูกตรวจสอบเป็นอันดับแรก ถ้าคู่ค่าแอ็ตทริบิวต์ ในไฟล์ *user_id*`.policy` ไม่ตรงกันอย่างถูกต้อง ไฟล์ `default.`

policy จะถูกตรวจสอบ ถ้าคู่แอ็ตทริบิวต์จากแพ็กเก็ตการร้องขอเพื่อเข้าถึง ไม่ตรงกับในไฟล์ แพ็กเก็ตการปฏิเสธการเข้าถึงจะถูกส่งไป ถ้าพบที่ตรงกันในไฟล์ใดไฟล์หนึ่ง แพ็กเก็ตการยอมรับการเข้าถึง จะถูกส่งไปยังไคลเอ็นต์ ซึ่งจะสร้างนโยบายสองระดับอย่างมี ประสิทธิผล

ไฟล์ default.auth ถูกใช้เป็นรายการของ คู่ค่าแอ็ตทริบิวต์เพื่อส่งกลับไปยังไคลเอ็นต์ทันทีที่นโยบาย ได้รับการตรวจสอบ ไฟล์ default.auth ยังมี คู่ค่าแอ็ตทริบิวต์ที่เริ่มแรกจะว่างเปล่าและต้องถูกตั้งค่า เพื่อให้ได้การตั้งค่าที่ต้องการ คุณต้องแก้ไขไฟล์ default.auth หรือใช้ SMIT เพื่อตั้งค่าแอ็ตทริบิวต์การอนุญาตที่ต้องการ แต่ละแอ็ตทริบิวต์ที่มีค่าจะถูกส่งกลับไปยัง NAS โดยอัตโนมัติในแพ็กเก็ตที่ยอมรับการเข้าถึง

คุณยังสามารถกำหนดแอ็ตทริบิวต์การอนุญาตที่ส่งกลับโดยเจาะจงผู้ใช้ โดยการสร้างไฟล์ตามชื่อผู้ใช้เฉพาะที่มีส่วนขยาย .auth เช่น ***user_id*.auth** ไฟล์ แบบกำหนดเองนี้อยู่ในไดเร็กทอรี /etc/radius/authorization โดยมีแผง SMIT ที่อนุญาตให้คุณสร้างและแก้ไขไฟล์ผู้ใช้แต่ละไฟล์

แอ็ตทริบิวต์การอนุญาตของผู้ใช้แต่ละรายถูกส่งกลับในแพ็กเก็ตการยอมรับ การเข้าถึงพร้อมด้วยแอ็ตทริบิวต์การอนุญาตดีฟอลต์ใดๆ ที่พบในไฟล์ default.auth หรือไฟล์ global.auth

ถ้าค่าเป็นค่าทั่วไปในไฟล์ default.auth และไฟล์ *user_id*.auth ดังนั้นค่าของผู้ใช้จะแทนที่ค่าดีฟอลต์ นี้อนุญาตสำหรับแอ็ตทริบิวต์การอนุญาตโกลบอลบางแอ็ตทริบิวต์ (เซอร์วิสหรือรีซอร์ส) ไปยัง ผู้ใช้ทั้งหมด และจากนั้นสำหรับระบบของการอนุญาตที่เจาะจงมากขึ้นสำหรับแต่ละผู้ใช้

หมายเหตุ: ใช้ไฟล์ global.auth เพื่อรวมแอ็ตทริบิวต์การอนุญาต กับแอ็ตทริบิวต์การอนุญาตที่ผู้ใช้เจาะจง แทน การใช้ไฟล์ default.auth ยกเว้นต้องการให้มีลักษณะการทำงาน รวมกันบางอย่าง

เริ่มต้นด้วย AIX เวอร์ชัน 6.1 ด้วย 6100-02 Technology Level, RADIUS สนับสนุนไฟล์การพิสูจน์ตัวตน global.auth ไฟล์นี้แทนที่และขยายเพิ่มเจตนาเริ่มต้นของการรวม แอ็ตทริบิวต์การอนุญาตที่ผู้ใช้ระบุ (ที่ในไฟล์ user_id.auth) ด้วยชุดของแอ็ตทริบิวต์การอนุญาตโกลบอล

ไฟล์ user_id.auth ต่างจากไฟล์ default.auth ถูกแทนที่โดยแอ็ตทริบิวต์ที่ใกล้เคียงกันที่พบในไฟล์การอนุญาตที่ผู้ใช้ระบุ แต่จะรวมกันโดยมีความยืดหยุ่นมากยิ่งขึ้นแทน ในการดูแลรักษาการอนุญาตสำหรับผู้ใช้

ถ้าแอ็ตทริบิวต์เป็นค่าทั่วไปในไฟล์ default.auth และไฟล์ user_id.auth ค่าของผู้ใช้ะแทนที่ค่าดีฟอลต์ ซึ่งค่านี้ทำการแทนที่ค่าดีฟอลต์ที่อนุญาต สำหรับแอ็ตทริบิวต์การอนุญาตดีฟอลต์บางค่า (เซอร์วิสหรือรีซอร์ส) ไปยังผู้ใช้ทั้งหมด และจากนั้นสำหรับระดับของการอนุญาตที่เจาะจงเฉพาะผู้ใช้มากขึ้น

เช่นเดียวกับเป็นจริงสำหรับแอ็ตทริบิวต์ในไฟล์ global.auth ยกเว้นว่าไม่ถูกแทนที่โดยแอ็ตทริบิวต์ user_id.auth แอ็ตทริบิวต์ในไฟล์สองไฟล์จะถูกรวมกันแทน นี้เป็นประโยชน์ เมื่อคุณกำลังระบุ vender-specific attributes (VSA)

กระบวนการอนุญาตมีดังนี้:

1. เมื่อเริ่มทำงาน daemon นโยบายและการอนุญาตดีฟอลต์ที่แสดงรายการ จากไฟล์ /etc/radius/authorization/default.policy ไฟล์ default.auth และ default.auth ถูกอ่านเข้าสู่หน่วยความจำ
2. พิสูจน์ตัวตน ID ผู้ใช้และรหัสผ่าน
3. แพ็กเก็ตขาเข้าถูกตรวจสอบคู่ค่าแอ็ตทริบิวต์
   a. ตรวจสอบไฟล์ ***user_id*.auth** แบบกำหนดเอง
   b. ถ้าไม่พบที่ตรงกัน ให้ตรวจสอบไฟล์ default.policy

c. ถ้าไม่พบที่ตรงกัน ให้ส่งแพ็กเก็ตปฏิเสธการเข้าถึง

4. นำใช้แอ็ตทริบิวต์การอนุญาตของผู้ใช้ถ้ามี
   a. อ่านไฟล์ /etc/radius/authorization/***user_id***.auth และไฟล์ default.auth และเปรียบเทียบสอง รายการ
   b. ใช้รายการที่อยู่ในไฟล์ของผู้ใช้ด้านบนเป็นรายการดีฟอลต์
   c. รวมแอ็ตทริบิวต์ที่เป็นผลลัพธ์กับแอ็ตทริบิวต์ที่ พบในไฟล์ global.auth
5. ส่งกลับแอ็ตทริบิวต์การอนุญาตในแพ็กเก็ตยอมรับการเข้าถึง

การจัดการบัญชีผู้ใช้

เซิร์ฟเวอร์การจัดการบัญชีผู้ใช้ RADIUS มีหน้าที่ในการรับการร้องขอเพื่อการจัดการบัญชีผู้ใช้จากไคลเอ็นต์และ การส่งการตอบกลับไปยังไคลเอ็นต์เพื่อบ่งชี้ว่าเซิร์ฟเวอร์ได้รับ การร้องขอและเขียนข้อมูลการจัดการบัญชีผู้ใช้

คุณสามารถเปิดใช้งานการจัดการบัญชีผู้ใช้โลคัลในไฟล์ radiusd.conf

เมื่อไคลเอ็นต์ถูกตั้งค่าเพื่อใช้การจัดการบัญชีผู้ใช้ RADIUS ซึ่งจะสร้างแพ็กเก็ต ACCOUNTING_START ที่อธิบายประเภทของเซอร์วิส ที่จะถูกนำส่ง และผู้ใช้ที่ประเภทจะถูกนำส่งในตอนเริ่มต้นของ การนำส่งเซอร์วิส ไคลเอ็นต์จะส่งแพ็กเก็ตไปยังเซิร์ฟเวอร์การจัดการบัญชีผู้ใช้ RADIUS ซึ่งส่งกลับการตอบรับที่แสดงว่าได้รับแพ็กเก็ตแล้วกลับมา เมื่อสิ้นสุดการนำส่งเซอร์วิส ไคลเอ็นต์จะสร้างแพ็กเก็ต ACCOUNTING_STOP อธิบายประเภทของเซอร์วิสที่ถูกนำส่ง และเป็นทางเลือกที่จะแสดงสถิติ เช่นเวลาที่ผ่านไป ค่าฐานแปดของอินพุตและเอาต์พุต หรือจำนวนแพ็กเก็ตอินพุต และเอาต์พุต เมื่อเซิร์ฟเวอร์การจัดการบัญชีผู้ใช้ RADIUS ได้รับ ACCOUNTING_STOP จะส่งการตอบรับกลับไปยังไคลเอ็นต์การจัดการบัญชีผู้ใช้เพื่อแจ้งกว่าได้รับแพ็กเก็ตแล้ว

ค่า ACCOUNTING_REQUEST ไม่ว่าสำหรับ START หรือ STOP ถูกส่งไปยัง เซิร์ฟเวอร์การจัดการบัญชีผู้ใช้ RADIUS ผ่านเน็ตเวิร์ก ขอแนะนำให้ไคลเอ็นต์ พยายามส่งแพ็กเก็ต ACCOUNTING_REQUEST ต่อไปจนกว่าจะได้รับ การตอบรับ ไคลเอ็นต์ยังสามารถส่งต่อการร้องขอไปยังเซิร์ฟเวอร์อื่น ในเหตุการณ์ที่เซิร์ฟเวอร์หลักไม่ทำงาน หรือไม่สามารถเข้าถึงได้ผ่านการใช้การตั้งค่าพร็อกซี สำหรับข้อมูลเพิ่มเติมเกี่ยวกับพร็อกซีเซอร์วิส ดูที่ “พร็อกซีเซอร์วิส” ในหน้า 369

ข้อมูลการจัดการบัญชีผู้ใช้ถูกเขียนในรูปแบบ RADIUS มาตรฐาน ของ *attribute=value* ลงในไฟล์ /etc/var/radius/data/accounting โลคัล ข้อมูลที่เขียนคือข้อมูลการจัดการบัญชีผู้ใช้ที่อยู่ในแพ็กเก็ต พร้อมการประทับเวลา ถ้าเซิร์ฟเวอร์การจัดการบัญชีผู้ใช้ RADIUS ไม่สามารถบันทึกแพ็กเก็ตการจัดการบัญชีผู้ใช้ ได้สำเร็จ เซิร์ฟเวอร์จะไม่ส่งการตอบรับ **Accounting_Response** ไปให้ไคลเอ็นต์ และข้อมูลข้อผิดพลาดจะถูกบันทึกไว้ในไฟล์ syslog

ไฟล์/var/radius/data/accounting:

/var/radius/data/accounting ดักจับสิ่งที่ ไคลเอ็นต์ส่งมาในแพ็กเก็ต ACCOUNTING START และ ACCOUNTING STOP

ไฟล์ /var/radius/data/accounting ว่างเปล่าเมื่อ ติดตั้งเป็นครั้งแรก ข้อมูลถูกเขียนลงไฟล์โดยยึดตามสิ่งที่ไคลเอ็นต์ส่งมาในแพ็กเก็ต ACCOUNTING START และ ACCOUNTING STOP

ต่อไปนี้คือตัวอย่างของประเภทของข้อมูลที่เซิร์ฟเวอร์ AIX RADIUS เขียนลงไฟล์ /var/radius/data/accounting ข้อมูลของคุณจะแตกต่างกันขึ้นอยู่กับวิธีติดตั้งระบบของคุณ

หมายเหตุ:

- ขอให้แน่ใจว่าระบบไฟล์ /var มีขนาดใหญ่เพียงพอที่จะจัดการ ข้อมูลการจัดการบัญชีผู้ใช้ทั้งหมด
- สคริปต์ Perl ของบุคคลที่สามสามารถนำมาใช้เพื่อแยกวิเคราะห์ข้อมูลในไฟล์นี้ ตัวอย่าง ของสคริปต์ที่สร้างรายงานจากข้อมูลการจัดการบัญชีผู้ใช้สามารถพบได้ที่ http://www.pgregg.com/projects/radiusreport
- แพ็กเก็ตการจัดการบัญชีผู้ใช้ยังสามารถถูกพร็อกซี

```
Thu May 27 14:43:19 2004
 NAS-IP-Address = 10.10.10.1
 NAS-Port = 1
 NAS-Port-Type = Async
 User-Name = "rod"
 Acct-Status-Type = Start
 Acct-Authentic = RADIUS
 Service-Type = Framed-User
 Acct-Session-Id = "0000000C"
 Framed-Protocol = PPP
 Acct-Delay-Time = 0
 Timestamp = 1085686999

Thu May 27 14:45:19 2004
 NAS-IP-Address = 10.10.10.1
 NAS-Port = 1    <-- rod was physically connected to port #1 on the hardware
 NAS-Port-Type = Async
 User-Name = "rod"
 Acct-Status-Type = Stop
 Acct-Authentic = RADIUS
 Service-Type = Framed-User
 Acct-Session-Id = "0000000C"   <-- note the session id's are the same so can match up start with stops
 Framed-Protocol = PPP
 Framed-IP-Address = 10.10.10.2   <-- IP address of user rod
 Acct-Terminate-Cause = User-Request  <-- user cancelled the session
 Acct-Input-Octets = 4016
 Acct-Output-Octets = 142
 Acct-Input-Packets = 35
 Acct-Output-Packets = 7
 Acct-Session-Time = 120  <--- seconds
 Acct-Delay-Time = 0
 Timestamp = 1085687119  <--- note "rod" was only logged on for 120 seconds (2 minutes)
```

พร็อกซีเซอร์วิส

พร็อกซีเซอร์วิสอนุญาตให้เซิร์ฟเวอร์ RADIUS ส่งต่อการร้องขอจาก NAS ไปยังเซิร์ฟเวอร์ RADIUS อีกเซิร์ฟเวอร์ และจากนั้นส่งข้อความตอบกลับไปยัง NAS พร็อกซีเซอร์วิสจะยึดตาม ชื่อขอบเขต

เซิร์ฟเวอร์ RADIUS สามารถทำหน้าที่ เป็นทั้งพร็อกซีเซิร์ฟเวอร์และเซิร์ฟเวอร์ด้านหลังพร้อมกัน กลไกนี้ใช้ได้กับทั้งแพ็กเก็ตการจัดการบัญชีผู้ใช้และการพิสูจน์ตัวตน พร็อกซีถูกปิดใช้งาน เป็นค่าดีฟอลต์ในไฟล์ radiusd.conf

ขอบเขต:

ขอบเขตคือ identifiers ที่อยู่ก่อนหน้าหรือหลังค่า โดยทั่วไปที่อยู่ในแอ็ตทริบิวต์ User-Name ที่เซิร์ฟเวอร์ RADIUS สามารถใช้เพื่อระบุเซิร์ฟเวอร์ที่จะติดต่อเพื่อเริ่มต้นกระบวนการพิสูจน์ตัวตนและ การจัดการบัญชีผู้ใช้

ตัวอย่างต่อไปนี้แสดงวิธีใช้ขอบเขตกับ RADIUS:

User, *Joe*, is employed by company *XYZ* in Sacramento. ขอบเขตสำหรับพื้นที่นี้ คือ *SAC* อย่างไรก็ตาม *Joe* ขณธนี้อยู่ใน นิวยอร์คซิตี้บนการมอบหมาย รีโมต ขอบเขตสำหรับนิวยอร์คซิตี้คือ *NYC* เมื่อ *Joe* ต่อเลขหมาย ในขอบเขต *NYC* User-Name ที่ส่งคือ *SAC/Joe* นี้เป็นการแจ้ง เซิร์ฟเวอร์ขอบเขต *NYC* RADIUS ว่าแพ็กเก็ตนี้จำเป็นต้องถูกส่งต่อไปยังเซิร์ฟเวอร์ที่ทำการพิสูจน์ตัวตน และการจัดการบัญชีผู้ใช้สำหรับผู้ใช้ที่มีขอบเขต *SAC*

แอ็ตทริบิวต์ Realm user-name:

แพ็กเก็ตการพิสูจน์ตัวตนและการจัดการบัญชีผู้ใช้จัดเส้นทางผ่านขอบเขต ตามค่าแอ็ตทริบิวต์ **User-Name** แอ็ตทริบิวต์นี้กำหนดลำดับ ของขอบเขตที่แพ็กเก็ตจะผ่านไปเพื่อจัดเส้นทางแพ็กเก็ตไปยังเซิร์ฟเวอร์สุดท้าย ที่ทำการพิสูจน์ตัวตนหรือการจัดการบัญชีผู้ใช้

แพ็กเก็ตถูกจัดเส้นทางโดยการเชื่อมสตริงขอบเขตเข้าด้วยกันในแอ็ตทริบิวต์ **User-Name** ขอบเขตที่แท้จริงที่ถูกแทรกเข้ามาในแอ็ตทริบิวต์ **User-Name** ซึ่งท้ายที่สุดจะพิจารณาพาธของแพ็กเก็ต คือการตัดสินใจที่ขึ้น อยู่กับผู้ดูแลระบบในการนำใช้ผัง RADIUS โดยสามารถวาง ชื่อของฮ็อพขอบเขตไว้ด้านหน้าของแอ็ตทริบิวต์ **User-Name** เช่นเดียวกับ ด้านหลัง อักขระที่ใช้มากสุดในการวางรูปแบบขอบเขตคือ เครื่องหมายทับ (/) เป็นตัวคั่นด้านหน้าของแอ็ตทริบิวต์ **User-Name** และแอมเปอร์แซนด์ (&) เป็นตัวคั่นต่อท้ายแอ็ตทริบิวต์ **User-Name** ตัวคั่นถูกตั้งค่าในไฟล์ `radiusd.conf` แอ็ตทริบิวต์ **User-Name** ถูกแยกวิเคราะห์จากซ้ายไปขวา

ตัวอย่าง ของแอ็ตทริบิวต์ **User-Name** ที่ใช้วิธีใส่คำนำหน้าเท่านั้นเป็นดังนี้:

```
USA/TEXAS/AUSTIN/joe
```

ตัวอย่าง ของแอ็ตทริบิวต์ **User-Name** ที่ใช้วิธีต่อท้ายเท่านั้นเป็น ดังนี้:

```
joe@USA@TEXAS@AUSTIN
```

โดยสามารถใช้ได้ทั้งสองวิธีคือวิธีนำหน้าและต่อท้าย สิ่งสำคัญคือต้องจำว่าเมื่อระบุ ฮ็อพขอบเขต แพ็กเก็ตจะไปตามลำดับฮ็อพที่ถูกแยกวิเคราะห์จากซ้าย ไปขวา และฮ็อพนำหน้าทั้งหมดถูกประมวลผลก่อนการประมวลผลฮ็อพต่อท้าย ผู้ใช้ต้องได้รับการพิสูจน์ตัวตน หรือข้อมูลการจัดการบัญชีผู้ใช้ถูกเขียนที่โหนดเดียว

ตัวอย่างต่อไปนี้ใช้ทั้งสองวิธี จะให้ผลลัพธ์เหมือนกับ ตัวอย่างข้างต้น:

```
USA/joe@TEXAS@AUSTIN
```

การตั้งค่าพร็อกซีเซอร์วิส:

ข้อมูลการตั้งค่าพร็อกซี RADIUS อยู่ในไฟล์ `proxy` ในไดเร็กทอรี `/etc/radius`

ไฟล์ `proxy` เริ่มต้นมี รายการตัวอย่าง มีฟิลด์สามฟิลด์ในไฟล์ proxy: **Realm Name, Next Hop IP address** และ **Shared Secret**

ในการ ตั้งค่ากฎพร็อกซี ให้เลือกจากต่อไปนี้:

```
    Configure Proxy Rules

List all Proxy
Add a Proxy
Change / Show Characteristics of a Proxy
Remove a Proxy
```

เลือกอ็อพชัน **List all Proxy** เพื่ออ่านไฟล์ /etc/radius/proxy และแสดงทั้งสามฟิลด์ในรูปแบบคอลัมน์ ต่อไปนี้คือส่วนหัวคอลัมน์:

```
realm_name   next_hop_address  shared_secret
```

เลือก **Add a Proxy** เพื่อแสดงหน้าจอต่อไปนี้ ข้อมูลถูกเรียกออกมา จากแผงและข้อมูลถูกผนวกกับด้านล่างของไฟล์ /etc/radius/proxy

แต่ละ ฮ็อพของห่วงโซ่พร็อกซีจะใช้ความลับที่แบ่งใช้ระหว่างสองเซิร์ฟเวอร์ RADIUS ความลับที่แบ่งใช้มีอยู่ใน /etc/radius/proxy_file ความลับที่แบ่งใช้ควรเป็นค่าเฉพาะสำหรับพร็อกซีฮ็อพในห่วงโซ่

สำหรับข้อมูล เพิ่มเติมเกี่ยวกับการสร้างความลับที่แบ่งใช้ โปรดดูที่ “ไฟล์ /etc/radius/clients” ในหน้า 357

เมื่อต้องการ เพิ่มพร็อกซี ให้เลือกจากฟิลด์ที่แสดงด้านล่าง:

```
            Add a Proxy
*Realm Name                              []  (max 64 chars)
*Next Hop IP address (dotted decimal)    [xx.xx.xx.xx]
*Shared Secret                           []  (minimum 6, maximum 256 chars)
```

การเลือกอ็อพชัน **Change/Show** จะแสดง รายการของชื่อขอบเขต รายการถูกแสดงในหน้าจอป๊อบอัพและคุณ ต้องเลือกชื่อขอบเขต

อ็อพชัน **Remove a Proxy** แสดงรายกรของชื่อขอบเขต รายการถูกแสดงในหน้าต่างป๊อบอัพ และผู้ใช้ต้องเลือกชื่อขอบเขต หลังจากชื่อถูกเลือก หน้าจอป๊อบอัพ การตรวจสอบความถูกต้องจะถูกแสดงก่อนที่ขอบเขตจะถูกลบออก

ตัวอย่างต่อไปนี้ คือส่วนข้อมูลคอนฟิกูเรชันพร็อกซีของไฟล์ radiusd.conf:

```
#-----------------------------------------------------------------#
#       PROXY RADIUS Information                                  #
#                                                                 #
#                                                                 #
#   Proxy_Allow               :  ON or OFF. If ON, then the server #
#                                can proxy packets to realms it   #
#                                knows of and the following       #
#                                fields must also be configured.  #
#   Proxy_Use_Table           :  ON or OFF. If ON, then the server #
#                                can use table for faster         #
#                                processing of duplicate requests #
#                                Can be used without proxy ON, but #
#                                it is required to be ON if       #
#                                Proxy_Use_Table is set to ON.    #
#   Proxy_Realm_name          :  This field specifies the realm   #
#                                this server services.            #
#   Proxy_Prefix_delim        :  A list of separators for parsing #
#                                realm names added as a prefix to #
#                                the username.  This list must be #
#                                mutually exclusive to the Suffix #
#                                delimiters.                      #
#   Proxy_Suffix_delim        :  A list of separators for parsing #
#                                realm names added as a suffix to #
#                                the username.  This list must be #
```

```
#                               mutually exclusive to the Prefix  #
#                               delimiters.                       #
#   Proxy_Remove_Hops         : YES or NO.  If YES then the       #
#                               will remove its realm name, the   #
#                               realm names of any previous hops  #
#                               and the realm name of the next    #
#                               server the packet will proxy to.  #
#                                                                 #
#   Proxy_Retry_count         : The number of times to attempt    #
#                               to send the request packet.       #
#                                                                 #
#   Proxy_Time_Out            : The number of seconds to wait     #
#                               in between send attempts.         #
#                                                                 #
#-----------------------------------------------------------------#
Proxy_Allow               :   OFF
Proxy_Use_Table           :   OFF
Proxy_Realm_name          :
Proxy_Prefix_delim        :   $/
Proxy_Suffix_delim        :   @.
Proxy_Remove_Hops         :   NO
Proxy_Retry_count         :   2
Proxy_Time_Out            :   3
```

การตั้งค่าเซิร์ฟเวอร์ RADIUS:

daemon เซิร์ฟเวอร์ RADIUS ใช้ไฟล์คอนฟิกูเรชันหลายไฟล์ ข้อมูลคอนฟิกูเรชันเซิร์ฟเวอร์ถูกบันทึกในไฟล์ `/etc/radius/radiusd.conf` ไฟล์คอนฟิกูเรชันเซิร์ฟเวอร์แพ็กเกจมีมาพร้อมค่าดีฟอลต์

หมายเหตุ: ต่อไปนี้คือแผง RADIUS Configure Server SMIT ตัวอย่าง:

```
    Configure Server

RADIUS Directory            /etc/radius
*Database Location           [UNIX]
Local AVL Database File Name [dbdata.bin]
Local Accounting             [ON]
Local Accounting Directory   []

Debug Level                  [3]
Accept Reply-Message         []
Reject Reply-Message         []
Challenge Reply-Message      []
Password Expired Reply Message []
Support Renewal of Expired Passwords [NO]
Require Message Authenticator  [NO]

*Authentication Port Number  [1812]
*Accounting Port Number      [1813]

LDAP Server Name             []
LDAP Server Port Number      [389]
LDAP Server Admin Distinguished Name     []
LDAP Server Admin Password         []
LDAP Base Distinguished Name       [cn=aixradius]
LDAP Size Limit                    [0]
LDAP Hop Limit                     [0]
LDAP wait time limit               [10]
LDAP debug level                   [ 0]

Proxy Allowed                      [OFF]
Proxy Use table                    [OFF]
Proxy Realm Name                   []
Proxy Prefix Delimiters            [$/]
Proxy Suffix Delimiters            [@.]
     NOTE: prefix & suffix are mutually exclusive
Proxy Remove Hops                  [NO]
Proxy Retry Count                  [2]
Proxy Timeout                      [30]
UNIX Check Login Restrictions      [OFF]
Enable IP Pool                     [ON]
Authentication Method Sequence     [TLS, MD5]
OpenSSL Configuration File         []
```

ยูทิลิตีการบันทึกการทำงาน

เซิร์ฟเวอร์ RADIUS ใช้ SYSLOG เพื่อบันทึกกิจกรรมและข้อมูล ข้อผิดพลาด

มีสามระดับของการบันทึกข้อมูล:

0 เฉพาะปัญหาหรือข้อผิดพลาดเท่านั้น และการเริ่มทำงานของ daemons ถูกบันทึก

3 บันทึกการติดตามการตรวจสอบของข้อความ `access_accept`, `access_reject*`, `discard` และ `error`

หมายเหตุ: ข้อความ `discard` ถูกบันทึกเมื่อแพ็กเก็ตขาเข้าไม่ถูกต้อง และไม่มีการสร้างแพ็กเก็ต การตอบกลับ

9 รวมข้อมูลการบันทึกระดับ 0 และ 3 และอื่นๆ โดยรันการบันทึกการทำงานระดับ 9 เพื่อดีบักเท่านั้น

ระดับดีฟอลต์ของการบันทึกการทำงานคือระดับ 3 การบันทึกการทำงานที่ระดับ 3 ใช้เพื่อปรับปรุงระดับการตรวจสอบของเซิร์ฟเวอร์ RADIUS ทั้งนี้ขึ้นอยู่กับ ระดับที่เซิร์ฟเวอร์กำลังใช้บันทึกการทำงาน คุณสามารถใช้กิจกรรมที่เก็บ ในบันทึกเพื่อตรวจสอบหารูปแบบที่น่าสงสัยของกิจกรรม ถ้ามีการ ละเมิด เอาต์พุต SYSLOG สามารถใช้เพื่อพิจารณาว่าการละเมิดเกิดขึ้นอย่างไร และเมื่อใด และบางครั้งอาจรวมถึงระดับการเข้าถึงที่ได้รับ ข้อมูลนี้ เป็นประโยชน์สำหรับใช้ในการพัฒนาการวัดการรักษาความปลอดภัยให้ดีขึ้น เพื่อป้องกันปัญหาที่เกิดขึ้นในอนาคต

ข้อมูลที่เกี่ยวข้อง:

IBM Directory Server

การตั้งค่า RADIUS เพื่อใช้ syslogd daemon:

ในการใช้ SYSLOG เพื่อดูกิจกรรมและข้อมูลข้อผิดพลาด คุณต้องเปิดใช้งาน syslogd daemon

ในการเปิดใช้งาน syslogd daemon ดำเนินขั้นตอนต่อไปนี้

1. แก้ไขไฟล์ `/etc/syslog.conf` เพื่อเพิ่ม รายการต่อไปนี้:`local4.debug var/adm/ipsec.log` ใช้สิ่งอำนวยความสะดวก *local4* เพื่อบันทึกปริมาณการรับส่งข้อมูลและเหตุการณ์ IP Security โดยใช้ระดับความสำคัญของระบบปฏิบัติการมาตรฐาน คุณควรตั้งค่า sระดับความสำคัญของ *debug* จนกว่าการรับส่งผ่าน IP Security tunnels และตัวกรองจะแสดงความเสถียร์และการเคลื่อนย้ายที่เหมาะสม

   หมายเหตุ: การบันทึกการทำงานของเหตุการณ์ตัวกรองอาจสร้างกิจกรรมจำนวนมากที่โฮสต์ IP Security และอาจใช้พื้นที่สื่อบันทึกเป็นจำนวนมาก
2. บันทึก `/etc/syslog.conf file`
3. ไปที่ไดเร็กทอรีที่คุณระบุสำหรับล็อกไฟล์และสร้าง ไฟล์ว่างที่มีชื่อเดียวกัน ในกรณีข้างต้น คุณควร เปลี่ยนเป็นไดเร็กทอรี `/var/adm` และรันคำสั่ง **touch** ดังนี้:

   ```
   touch ipsec.log
   ```
4. รันคำสั่ง **refresh** กับระบบย่อย `syslogd` ดังนี้:

   ```
   refresh -s syslogd
   ```

การตั้งค่าเอาต์พุต SYSLOG:

คุณสามารถตั้งค่า Debug_Level เป็น 0, 3 หรือ 9 ที่ตั้งค่าในไฟล์is set in the `radiusd.conf` file, depending on how much debugging information you want included in the SYSLOG output.

ค่ากำหนดดีฟอลต์คือ 3 ส่วนการดีบักของไฟล์ `radiusd.conf` จะคล้ายกับที่แสดงต่อไปนี้:

```
#.
#.
#.
#   Debug_Level      :  This pair sets the debug level at which    #
#                       the RADIUS server will run.  Appropriate   #
#                       values are 0,3 or 9. The default is 3.     #
#                       Output is directed to location specified   #
#                       by *.debug stanza in /etc/syslog.conf      #
#                                                                  #
#                       Each level increases the amount of messages#
#                       sent to syslog. For example "9" includes   #
#                       the new messages provided by "9" as well   #
```

```
#                       as all messages generated by level 0 and 3.#
#                                                                  #
#                       0 : provides the minimal output to the     #
#                           syslogd log.  It sends start up        #
#                           and end messages for each RADIUS       #
#                           process.  It also logs error           #
#                           conditions.                            #
#                                                                  #
#                       3 : includes general ACCESS ACCEPT, REJECT #
#                           and DISCARD messages for each packet.  #
#                           This level provides a general audit    #
#                           trail for authentication.              #
#                                                                  #
#                       9 : Maximum amount of log data. Specific   #
#                           values of attributes while a           #
#                           transaction is passing thru            #
#                           processing and more.                   #
#                           [NOT advised under normal operations]  #
#                                                                  #
#------------------------------------------------------------------#
```

ตัวอย่างต่อไปนี้แสดงเอาต์พุตตัวอย่างสำหรับระดับการดีบักที่ต่างกัน

แพ็กเก็ตการจัดการบัญชีผู้ใช้ที่มีการดีบักระดับ 3

```
Aug 18 10:23:57 server1 syslog: [0]:Monitor process [389288] has started
Aug 18 10:23:57 server1 radiusd[389288]: [0]:Local database (AVL) built.
Aug 18 10:23:57 server1 radiusd[389288]: [0]:Authentication process started : Pid= 549082 Port = 1812
Aug 18 10:23:57 server1 radiusd[389288]: [0]:Accounting process started : Pid= 643188 Port = 1813
Aug 18 10:23:57 server1 radiusd[643188]: [0]:Socket created [15]
Aug 18 10:23:57 server1 radiusd[643188]: [0]:Bound Accounting socket [15]
Aug 18 10:23:57 server1 radiusd[549082]: [0]:Socket created [15]
Aug 18 10:23:57 server1 radiusd[549082]: [0]:Bound Authentication socket [15]
Aug 18 10:24:07 server1 radiusd[643188]: [1]:*** Start Process_Packet() ***
Aug 18 10:24:07 server1 radiusd[643188]: [1]:Code 4, ID = 96, Port = 41639 Host = 10.10.10.10
Aug 18 10:24:07 server1 radiusd[643188]: [1]:ACCOUNTING-START - sending Accounting Ack to User [ user_id1 ]
Aug 18 10:24:07 server1 radiusd[643188]: [1]:Sending Accounting Ack of id 96 to 10.10.10.10 (client1.ibm.com)
Aug 18 10:24:07 server1 radiusd[643188]: [1]:send_acct_reply() Outgoing Packet:
Aug 18 10:24:07 server1 radiusd[643188]: [1]: Code = 5, Id = 96, Length = 20
Aug 18 10:24:07 server1 radiusd[643188]: [1]:*** Leave Process_Packet() ***
Aug 18 10:24:13 server1 radiusd[643188]: [2]:*** Start Process_Packet() ***
Aug 18 10:24:13 server1 radiusd[643188]: [2]:Code 4, ID = 97, Port = 41639 Host = 10.10.10.10
Aug 18 10:24:13 server1 radiusd[643188]: [2]:ACCOUNTING-STOP - sending Accounting Ack to User [ user_id1 ]
Aug 18 10:24:14 server1 radiusd[643188]: [2]:Sending Accounting Ack of id 97 to 10.10.10.10 (client1.ibm.com)
Aug 18 10:24:14 server1 radiusd[643188]: [2]:send_acct_reply() Outgoing Packet:
Aug 18 10:24:14 server1 radiusd[643188]: [2]: Code = 5, Id = 97, Length = 20
Aug 18 10:24:14 server1 radiusd[643188]: [2]:*** Leave Process_Packet() **
```

แพ็กเก็ตการจัดการบัญชีผู้ใช้ที่ระดับ 9

```
Aug 18 10:21:18 server1 syslog: [0]:Monitor process [643170] has started
Aug 18 10:21:18 server1 radiusd[643170]: [0]:Local database (AVL) built.
Aug 18 10:21:18 server1 radiusd[643170]: [0]:Authentication process started : Pid= 389284 Port = 1812
Aug 18 10:21:18 server1 radiusd[643170]: [0]:Accounting process started : Pid= 549078 Port = 1813
Aug 18 10:22:03 server1 radiusd[643170]: [0]:PID = [389284] dead
Aug 18 10:22:03 server1 radiusd[643170]: [0]:PID = [549078] dead
Aug 18 10:22:03 server1 radiusd[643170]: [0]:All child processes stopped. radiusd parent stopping
```

```
Aug 18 10:22:09 server1 syslog: [0]:Monitor process [1081472] has started
Aug 18 10:22:09 server1 radiusd[1081472]: [0]:Local database (AVL) built.
Aug 18 10:22:09 server1 radiusd[1081472]: [0]:Inside client_init()
Aug 18 10:22:09 server1 radiusd[1081472]: [0]:Number of client entries read: 1
Aug 18 10:22:09 server1 radiusd[1081472]: [0]:Inside read_authorize_policy routine for file:
  /etc/radius/authorization/default.policy.
Aug 18 10:22:09 server1 radiusd[1081472]: [0]:Inside read_authorize_file routine for file:
  /etc/radius/authorization/default.policy.
Aug 18 10:22:09 server1 radiusd[1081472]: [0]:read_authorize_file() routine complete.
Aug 18 10:22:09 server1 radiusd[1081472]: [0]:Inside read_authorize_file routine for file:
  /etc/radius/authorization/default.auth.
Aug 18 10:22:09 server1 radiusd[1081472]: [0]:read_authorize_file() routine complete.
Aug 18 10:22:09 server1 radiusd[549080]: [0]:connect_to_LDAP_server:Database Location (where the data
  resides)=LDAP.
Aug 18 10:22:09 server1 radiusd[549080]: [0]:connect_to_LDAP_server:LDAP Server name= server1.austin.ibm.com.
Aug 18 10:22:09 server1 radiusd[549080]: [0]:connect_to_LDAP_server:LDAP Server port= 389.
Aug 18 10:22:09 server1 radiusd[1081472]: [0]:Authentication process started : Pid= 549080 Port = 1812
Aug 18 10:22:09 server1 radiusd[389286]: [0]:connect_to_LDAP_server:Database Location (where the data
 resides)=LDAP.
Aug 18 10:22:09 server1 radiusd[389286]: [0]:connect_to_LDAP_server:LDAP Server name= server1.austin.ibm.com.
Aug 18 10:22:09 server1 radiusd[389286]: [0]:connect_to_LDAP_server:LDAP Server port= 389.
Aug 18 10:22:09 server1 radiusd[1081472]: [0]:Accounting process started : Pid= 389286 Port = 1813
Aug 18 10:22:10 server1 radiusd[549080]: [0]:Socket created [15]
Aug 18 10:22:10 server1 radiusd[549080]: [0]:Bound Authentication socket [15]
Aug 18 10:22:10 server1 radiusd[389286]: [0]:Socket created [15]
Aug 18 10:22:10 server1 radiusd[389286]: [0]:Bound Accounting socket [15]
Aug 18 10:22:15 server1 radiusd[389286]: [1]:*** Start Process_Packet() ***
Aug 18 10:22:15 server1 radiusd[389286]: [1]:Incoming Packet:
Aug 18 10:22:15 server1 radiusd[389286]: [1]:  Code = 4, Id = 94, Length = 80
Aug 18 10:22:15 server1 radiusd[389286]: [1]:  Authenticator = 0xC5DBDDFE6EFFFDBD6AE64CA35947DD0F
Aug 18 10:22:15 server1 radiusd[389286]: [1]:    Type =  40, Length =   6, Value = 0x00000001
Aug 18 10:22:15 server1 radiusd[389286]: [1]:    Type =   1, Length =   8, Value = 0x67656E747931
Aug 18 10:22:15 server1 radiusd[389286]: [1]:    Type =   4, Length =   6, Value = 0x00000000
Aug 18 10:22:15 server1 radiusd[389286]: [1]:    Type =   8, Length =   6, Value = 0x0A0A0A01
Aug 18 10:22:15 server1 radiusd[389286]: [1]:    Type =  44, Length =   8, Value = 0x303030303062
Aug 18 10:22:15 server1 radiusd[389286]: [1]:    Type =  30, Length =  10, Value = 0x3132332D34353638
Aug 18 10:22:15 server1 radiusd[389286]: [1]:    Type =  31, Length =  10, Value = 0x3435362D31323335
Aug 18 10:22:15 server1 radiusd[389286]: [1]:    Type =  85, Length =   6, Value = 0x00000259
Aug 18 10:22:15 server1 radiusd[389286]: [1]:Starting parse_packet()
Aug 18 10:22:15 server1 radiusd[389286]: [1]:Code 4, ID = 94, Port = 41639 Host = 10.10.10.10
Aug 18 10:22:15 server1 radiusd[389286]: [1]:Acct-Status-Type = Sta
```

แพ็กเก็ตการพิสูจน์ตัวตนระดับ 0

```
Aug 18 10:06:11 server1 syslog: [0]:Monitor process [1081460] has started
Aug 18 10:06:11 server1 radiusd[1081460]: [0]:Local database (AVL) built.
Aug 18 10:06:11 server1 radiusd[1081460]: [0]:Authentication process started : Pid= 549076 Port = 1812
Aug 18 10:06:11 server1 radiusd[1081460]: [0]:Accounting process started : Pid= 389282 Port = 18
```

Level 3 authentication packet

```
Aug 18 10:01:32 server2 radiusd[389276]: [3]:*** Start Process_Packet() ***
Aug 18 10:01:32 server2 radiusd[389276]: [3]:Code 1, ID = 72, Port = 41638 Host = 10.10.10.10
Aug 18 10:01:32 server2 radiusd[389276]: [3]:authenticate_password_PAP: Passwords do not match, user is rejected
Aug 18 10:01:32 server2 radiusd[389276]: [3]:Authentication failed for user [user_id1] using IP [10.10.10.10]
Aug 18 10:01:32 server2 radiusd[389276]: [3]:ACCESS-REJECT - sending reject for id 72 to 10.10.10.10
  (client1.ibm.com)
Aug 18 10:01:32 server2 radiusd[389276]: [3]:send_reject() Outgoing Packet:
Aug 18 10:01:32 server2 radiusd[389276]: [3]: Code = 3, Id = 72, Length = 30
```

```
Aug 18 10:01:32 server2 radiusd[389276]: [3]:*** Leave Process_Packet() ***
Aug 18 10:01:53 server2 radiusd[389276]: [4]:*** Start Process_Packet() ***
Aug 18 10:01:53 server2 radiusd[389276]: [4]:Code 1, ID = 74, Port = 41638 Host = 10.10.10.10
Aug 18 10:01:53 server2 radiusd[389276]: [4]:authenticate_password_PAP: Passwords Match, user is authenticated
Aug 18 10:01:53 server2 radiusd[389276]: [4]:Authentication successful for user [user_id1] using IP [10.10.10.10]
Aug 18 10:01:53 server2 radiusd[389276]: [4]:Authorization successful for user [user_id1] using IP [10.10.10.10]
Aug 18 10:01:53 server2 radiusd[389276]: [4]:ACCESS-ACCEPT - sending accept for id 74 to 10.10.10.10
  (client1.ibm.com)
Aug 18 10:01:53 server2 radiusd[389276]: [4]:send_accept() Outgoing Packet:
Aug 18 10:01:53 server2 radiusd[389276]: [4]: Code = 2, Id = 74, Length = 31
Aug 18 10:01:53 server2 radiusd[389276]: [4]:*** Leave Process_Packet() **
```

แพ็กเก็ตการพิสูจน์ตัวตนระดับ 9

```
Aug 18 10:03:56 server1 radiusd[389278]: [1]:*** Start Process_Packet() ***
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Incoming Packet:
Aug 18 10:03:56 server1 radiusd[389278]: [1]:  Code = 1, Id = 77, Length = 58
Aug 18 10:03:56 server1 radiusd[389278]: [1]:  Authenticator = 0xE6CB0F9C22BB4E799854E734104FB2D5
Aug 18 10:03:56 server1 radiusd[389278]: [1]:    Type =   1, Length =   8, Value = 0x67656E747931
Aug 18 10:03:56 server1 radiusd[389278]: [1]:    Type =   4, Length =   6, Value = 0x00000000
Aug 18 10:03:56 server1 radiusd[389278]: [1]:    Type =   2, Length =  18, Value = 0x**********
**********************
Aug 18 10:03:56 server1 radiusd[389278]: [1]:    Type =   7, Length =   6, Value = 0x00000001
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Starting parse_packet()
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Code 1, ID = 77, Port = 41638 Host = 10.10.10.10
Aug 18 10:03:56 server1 radiusd[389278]: [1]:User-Name = "user_id1"
Aug 18 10:03:56 server1 radiusd[389278]: [1]:NAS-IP-Address = 10.10.10.10
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Framed-Protocol = PPP
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Leaving parse_packet()
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Verifying Message-Authenticator
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Message-Authenticator successfully verified
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Inside proxy_request_needed() function
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Proxy is not turned on
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Username = [user_id1]
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Client IP = [10.10.10.10]
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Inside parse_for_login( user_id1 )
Aug 18 10:03:56 server1 radiusd[389278]: [1]:User_id remaining after prefix removal = [user_id1]
Aug 18 10:03:56 server1 radiusd[389278]: [1]:User_id remaining after suffix removal = [user_id1]
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Inside rad_authenticate() function
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Authentication request received for [client1.austin.ibm.com]
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Calling get_ldap_user() to get LDAP user data
Aug 18 10:03:56 server1 radiusd[389278]: [1]:get_ldap_user:LDAP user id: user_id1.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:get_ldap_user:LDAP max_login_cnt:2.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:get_ldap_user:LDAP EAP_type: 4.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:get_ldap_user:LDAP passwordexpiredweeks: 9.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:get_ldap_active_sessions:number of free entries= 2.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:get_ldap_active_session:dn retrieved=
radiusuniqueidentifier=user_id11,ou=radiusActiveUsers,cn=aixradius.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Inside get_client_secret routine for ip:10.10.10.10
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Found NAS-IP = [10.10.10.10]
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Found shared secret.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:authenticate_password_PAP: Passwords Match, user is authenticated
Aug 18 10:03:56 server1 radiusd[389278]: [1]:is_ldap_pw:password for user has NOT expired
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Authentication successful for user [user_id1] using IP [10.10.10.10]
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Inside rad_authorize() routine.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Inside read_authorize_policy routine for file:
  /etc/radius/authorization/user_id1.policy.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Inside read_authorize_file routine for file:
  /etc/radius/authorization/user_id1.policy.
```

```
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Did not open /etc/radius/authorization/user_id1.policy file.
  File may not be found.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Error reading policy file: /etc/radius/authorization/user_id1.policy.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:rad_authorize:default policy list and userpolicy list were empty.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:In create_def_copy() routine.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Successfully made a copy of the master authorization list.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Inside read_authorize_file routine for file:
  /etc/radius/authorization/user_id1.auth.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Did not open /etc/radius/authorization/user_id1.auth file.
  File may not be found.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:rad_authorize:copy authorization list and user list were empty.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Authorization successful for user [user_id1] using IP [10.10.10.10]
Aug 18 10:03:56 server1 radiusd[389278]: [1]:ACCESS-ACCEPT - sending accept for id 77 to 10.10.10.10
  (client1.austin.ibm.com)
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Inside proxy_response_needed() function
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Proxy is not turned on
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Inside get_client_secret routine for ip:10.10.10.10
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Found NAS-IP = [10.10.10.10]
Aug 18 10:03:56 server1 radiusd[389278]: [1]:Found shared secret.
Aug 18 10:03:56 server1 radiusd[389278]: [1]:send_accept() Outgoing Packet:
Aug 18 10:03:56 server1 radiusd[389278]: [1]: Code = 2, Id = 77, Length = 31
Aug 18 10:03:56 server1 radiusd[389278]: [1]:send_accept() Outgoing Packet:
Aug 18 10:03:56 server1 radiusd[389278]: [1]:  Code = 2, Id = 77, Length = 31
Aug 18 10:03:56 server1 radiusd[389278]: [1]:  Authenticator = 0xCCB2B645BBEE86F5E4FC5BE24E904B2A
Aug 18 10:03:56 server1 radiusd[389278]: [1]:    Type =  18, Length =  11, Value = 0x476F6F646E65737321
Aug 18 10:03:56 server1 radiusd[389278]: [1]:*** Leave Process_Packet() ***
Aug 18 10:04:18 server1 radiusd[389278]: [2]:*** Start Process_Packet() ***
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Incoming Packet:
Aug 18 10:04:18 server1 radiusd[389278]: [2]:  Code = 1, Id = 79, Length = 58
Aug 18 10:04:18 server1 radiusd[389278]: [2]:  Authenticator = 0x774298A2B6DD90D7C33B3C10C4787D41
Aug 18 10:04:18 server1 radiusd[389278]: [2]:    Type =   1, Length =   8, Value = 0x67656E747931
Aug 18 10:04:18 server1 radiusd[389278]: [2]:    Type =   4, Length =   6, Value = 0x00000000
Aug 18 10:04:18 server1 radiusd[389278]: [2]:    Type =   2, Length =  18, Value = 0x*******
*************************
Aug 18 10:04:18 server1 radiusd[389278]: [2]:    Type =   7, Length =   6, Value = 0x00000001
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Starting parse_packet()
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Code 1, ID = 79, Port = 41638 Host = 10.10.10.10
Aug 18 10:04:18 server1 radiusd[389278]: [2]:User-Name = "user_id1"
Aug 18 10:04:18 server1 radiusd[389278]: [2]:NAS-IP-Address = 10.10.10.10
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Framed-Protocol = PPP
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Leaving parse_packet()
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Verifying Message-Authenticator
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Message-Authenticator successfully verified
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Inside proxy_request_needed() function
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Proxy is not turned on
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Username = [user_id1]
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Client IP = [10.10.10.10]
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Inside parse_for_login( user_id1 )
Aug 18 10:04:18 server1 radiusd[389278]: [2]:User_id remaining after prefix removal = [user_id1]
Aug 18 10:04:18 server1 radiusd[389278]: [2]:User_id remaining after suffix removal = [user_id1]
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Inside rad_authenticate() function
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Authentication request received for [client1.austin.ibm.com]
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Calling get_ldap_user() to get LDAP user data
Aug 18 10:04:18 server1 radiusd[389278]: [2]:get_ldap_user:LDAP user id: user_id1.
Aug 18 10:04:18 server1 radiusd[389278]: [2]:get_ldap_user:LDAP max_login_cnt:2.
Aug 18 10:04:18 server1 radiusd[389278]: [2]:get_ldap_user:LDAP EAP_type: 4.
Aug 18 10:04:18 server1 radiusd[389278]: [2]:get_ldap_user:LDAP passwordexpiredweeks: 9.
Aug 18 10:04:18 server1 radiusd[389278]: [2]:get_ldap_active_sessions:number of free entries= 2.
Aug 18 10:04:18 server1 radiusd[389278]: [2]:get_ldap_active_session:dn retrieved=
```

```
  radiusuniqueidentifier=user_id11, ou=radiusActiveUsers, cn=aixradius.
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Inside get_client_secret routine for ip:10.10.10.10
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Found NAS-IP = [10.10.10.10]
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Found shared secret.
Aug 18 10:04:18 server1 radiusd[389278]: [2]:authenticate_password_PAP: Passwords do not match, user is rejected
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Authentication failed for user [user_id1] using IP [10.10.10.10]
Aug 18 10:04:18 server1 radiusd[389278]: [2]:ACCESS-REJECT - sending reject for id 79 to 10.10.10.10
 (client1.austin.ibm.com)
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Inside proxy_response_needed() function
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Proxy is not turned on
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Inside get_client_secret routine for ip:10.10.10.10
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Found NAS-IP = [10.10.10.10]
Aug 18 10:04:18 server1 radiusd[389278]: [2]:Found shared secret.
Aug 18 10:04:18 server1 radiusd[389278]: [2]:send_reject() Outgoing Packet:
Aug 18 10:04:18 server1 radiusd[389278]: [2]: Code = 3, Id = 79, Length = 30
Aug 18 10:04:18 server1 radiusd[389278]: [2]:send_reject() Outgoing Packet:
Aug 18 10:04:18 server1 radiusd[389278]: [2]:  Code = 3, Id = 79, Length = 30
Aug 18 10:04:18 server1 radiusd[389278]: [2]:  Authenticator = 0x05D4865C6EBEFC1A9300D2DC66F3DBE9
Aug 18 10:04:18 server1 radiusd[389278]: [2]:    Type =  18, Length =  10, Value = 0x4261646E65737321
Aug 18 10:04:18 server1 radiusd[389278]: [2]:*** Leave Process_Packet() **
```

การหมดอายุของรหัสผ่าน

การหมดอายุของรหัสผ่านอนุญาตให้ไคลเอ็นต์ RADIUS ได้รับการแจ้งเตือนเมื่อ รหัสผ่านของผู้ใช้หมดอายุ และอัพเดตรหัสผ่านของผู้ใช้ผ่าน โปรโตคอล RADIUS

การหมดอายุของรหัสผ่านเกี่ยวกับการสนับสนุนประเภทแพ็กเก็ตเพิ่มเติมสี่ประเภทและหนึ่งแอ็ตทริบิวต์ใหม่ ประเภทแพ็กเก็ตใหม่มีมาในพจนานุกรม AIX และต้องเปิดใช้คุณลักษณะการหมดอายุของรหัสผ่าน

อาจไม่เป็นที่ต้องการในการติดตั้ง RADIUS ทุกครั้งที่จะอนุญาตให้มีการอัพเดต รหัสผ่านที่หมดอายุผ่าน RADIUS รายการไฟล์ `radiusd.conf` มีอ็อพชันให้คุณเลือกที่จะอนุญาตหรือไม่อนุญาตการสนับสนุนการเปลี่ยนรหัสผ่านที่หมดอายุ ผ่าน RADIUS ค่าดีฟอลต์สำหรับอ็อพชันนี้คือ ไม่อนุญาต คุณสามารถเพิ่มข้อความตอบกลับผู้ใช้ `Password_Expired_Reply_Message` และค่านี้จะถูกส่งกลับในแพ็กเก็ตรหัสผ่านที่หมดอายุ แอ็ตทริบิวต์รหัสผ่าน ทั้งใหม่และเก่า ต้องถูกเข้ารหัสและถอดรหัสด้วย วิธี PAP

แอ็ตทริบิวต์ Vendor-specific

แอ็ตทริบิวต์ Vendor-specific (VSA) ถูกกำหนดโดยผู้จำหมายเซิร์ฟเวอร์ ที่เข้าถึงแบบรีโมต โดยทั่วไปเป็นผู้จำหน่ายฮาร์ดแวร์ เพื่อกำหนดวิธีที่ RADIUS ทำงานบนเซิร์ฟเวอร์เอง

แอ็ตทริบิวต์ vendor-specific จำเป็นต้องใช้ถ้าคุณต้องการ ให้สิทธิแก่ผู้ใช้สำหรับการเข้าถึงมากกว่าหนึ่งประเภท VSAs อาจถูกใช้ร่วมกับแอ็ตทริบิวต์ที่ RADIUS กำหนด

VSAs เป็นทางเลือก แต่ถ้าฮาร์ดแวร์ NAS จำเป็นต้องใช้แอ็ตทริบิวต์เพิ่มเติม ถูกตั้งค่าเพื่อให้ทำงานได้อย่างเหมาะสม คุณต้องเพิ่ม VSAs ในไฟล์พจนานุกรม

VSAs ยังสามารถใช้สำหรับการอนุญาตอื่นๆ โดยใช้ **User-Name** และ **Password**, ร่วมกัน คุณสามารถใช้ VSAs สำหรับการอนุญาต บนด้านเซิร์ฟเวอร์ ไฟล์นโยบายการอนุญาต ผู้ใช้จะมีรายการแอ็ตทริบิวต์ที่จะตรวจสอบในแพ็กเก็ต Access-Request สำหรับผู้ใช้เฉพาะ ถ้าแพ็กเก็ตไม่มีแอ็ตทริบิวต์ ที่แสดงรายการในไฟล์ผู้ใช้ ทำให้ access_reject ถูกส่งกลับไปยัง NAS VSAs ยังสามารถใช้เป็นรายการคู่ attribute=value ในไฟล์ ***user_id*`.policy`

ต่อไปนี้ คือตัวอย่างส่วน VSA ที่มาจากพจนานุกรม:

```
#############################################################################
#                                                                           #
#   This section contains examples of dictionary translations for           #
#   parsing vendor specific attributes (vsa). The example below is for      #
#   "Cisco." Before defining an Attribute/Value pair for a    #
#   vendor a "VENDOR" definition is needed.                                 #
#                                                                           #
#   Example:                                                                #
#                                                                           #
#   VENDOR          Cisco           9                                       #
#                                                                           #
#   VENDOR:  This specifies that the Attributes after this entry are        #
#            specific to Cisco.                                             #
#   Cisco :  Denotes the Vendor name                                        #
#   9     :  Vendor Id defined in the "Assigned Numbers" RFC                #
#                                                                           #
#############################################################################

#VENDOR          Cisco           9

#ATTRIBUTE       Cisco-AVPair              1       string
#ATTRIBUTE       Cisco-NAS-Port            2       string
#ATTRIBUTE       Cisco-Disconnect-Cause    195     integer
#
#----------------Cisco-Disconnect-Cause--------------------------------#
#
#VALUE           Cisco-Disconnect-Cause    Unknown                 2
#VALUE           Cisco-Disconnect-Cause    CLID-Authentication-Failure 4
#VALUE           Cisco-Disconnect-Cause    No-Carrier              10
#VALUE           Cisco-Disconnect-Cause    Lost-Carrier            11
#VALUE           Cisco-Disconnect-Cause    No-Detected-Result-Codes    12
#VALUE           Cisco-Disconnect-Cause    User-Ends-Session       20
#VALUE           Cisco-Disconnect-Cause    Idle-Timeout            21
#VALUE           Cisco-Disconnect-Cause    Exit-Telnet-Session     22
#VALUE           Cisco-Disconnect-Cause    No-Remote-IP-Addr       23
```

การสนับสนุนข้อความตอบกลับ RADIUS

ข้อความตอบกลับคือข้อความที่คุณสร้างขึ้นและตั้งค่าในไฟล์ radiusd.conf

มีจุดมุ่งหมายสำหรับ NAS หรือ AP เพื่อส่งกลับเป็นสตริงไปยังผู้ใช้ซึ่ง สามารถเป็นข้อความแสดงความสำเร็จ ล้มเหลว หรืออุปสรรค โดยเป็นฟิลด์ข้อความที่อ่านได้ และเนื้อหาของฟิลด์จะถูกนำไปใช้แบบมีเงื่อนไข และตั้งค่าในตอนทำการตั้งค่า เซิร์ฟเวอร์ ดีฟอลต์สำหรับแอ็ตทริบิวต์เหล่านี้คือไม่มีข้อความ คุณอาจตั้งค่าแอ็ตทริบิวต์ทั้งหมด ไม่ตั้งค่า หรือ หนึ่ง สอง หรือสามแอ็ตทริบิวต์

RADIUS สนับสนุนแอ็ตทริบิวต์ Reply-Message ต่อไปนี้:

- Accept Reply-Message
- Reject Reply-Message
- CHAP Reply-Message
- Password Expired Reply-Message

แอ็ตทริบิวต์เหล่านี้ถูกเพิ่มในไฟล์คอนฟิกูเรชัน radiusd.conf และอ่านไว้ในโครงสร้างโกลบอลคอนฟิกูเรชันตอนเริ่มทำงาน daemon ตั้งค่าเหล่านี้โดยใช้ SMIT RADIUS Panels เป็นส่วนหนึ่งของอ็อพชัน **Configure Server** จำนวนอักขระสูงสุดในแต่ละสตริง คือ 256 ไบต์

ฟังก์ชันถูกนำไปใช้ดังนี้:

1. เมื่อ **radiusd** daemon เริ่มทำงาน จะอ่านไฟล์ radiusd.conf และตั้งค่าแอ็ตทริบิวต์ Reply-Message
2. เมื่อได้รับแพ็กเก็ตการร้องขอการเข้าถึง ผู้ใช้จะถูกพิสูจน์ตัวตน
3. ถ้าการตอบกลับการพิสูจน์ตัวตนเป็นยอมรับการเข้าถึง ข้อความ Accept Reply-Message จะถูกทำเครื่องหมาย ถ้ามีข้อความแสดง สตริงถูกส่งกลับในแพ็กเก็ต ยอมรับการเข้าถึง
4. ถ้าการพิสูจน์ตัวตนถูกปฏิเสธ ข้อมู Reject Reply-Message ถูกทำเครื่องหมายและส่งกลับในแพ็กเก็ตการปฏิเสธการเข้าถึง
5. ถ้าการพิสูจน์ตัวตนมีอุปสรรค แอ็ตทริบิวต์ CHAP Reply-Message จะถูกทำเครื่องหมายและส่งเป็นส่วนหนึ่งของแพ็กเก็ต Access-Challenge

การตั้งค่า IP pool ของเซิร์ฟเวอร์ RADIUS

ด้วยเซิร์ฟเวอร์ RADIUS คุณสามารถกำหนด IP address แบบไดนามิก จาก IP address pool

การจัดสรร IP address เป็นส่วนหนึ่งของกระบวนการอนุญาตและ ถูกดำเนินการภายหังการพิสูจน์ตัวตน ผู้ดูแลระบบต้องกำหนด IP เฉพาะต่อหนึ่งผู้ใช้ในการกำหนด IP address ให้แก่ผู้ใช้แบบไดนามิก เซิร์ฟเวอร์ RADIUS จัดให้มีสามอ็อพชัน:

- แอ็ตทริบิวต์ Framed Pool
- การใช้แอ็ตทริบิวต์ Vendor Specific
- IP pooling ด้านเซิร์ฟเวอร์ RADIUS

แอ็ตทริบิวต์ Framed Pool

IP pool *poolname* ต้อง ถูกกำหนดบน Network Access Server (NAS) NAS ต้องเป็นไปตาม RFC2869 สำหรับเซิร์ฟเวอร์ RADIUS เพื่อส่งแอ็ตทริบิวต์ **Framed-Pool** ใน แพ็ก Access-Accept (แอ็ตทริบิวต์ประเภท 88) ผู้ดูแลระบบต้อง ตั้งค่า NAS และอัพเดตแอ็ตทริบิวต์การอนุญาตสำหรับ ผู้ใช้โดยการรวมแอ็ตทริบิวต์ **Framed-Pool** ในไฟล์ default.auth โกลบอล หรือไฟล์ user.auth บนเซิร์ฟเวอร์ RADIUS อย่างใดอย่างหนึ่ง ไฟล์ พจนานุกรมในเซิร์ฟเวอร์ RADIUS มีแอ็ตทริบิวต์นี้:

```
ATTRIBUTE    Framed-Pool    88       string
```

ถ้า NAS ไม่สามารถใช้พูลหลายแอดเดรส NAS จะข้ามแอ็ตทริบิวต์นี้ พูลแอดเดรสบน NAS มีรายการของ IP addresses NAS เลือกหนึ่งใน IP addresses ที่กำหนดในอยู่ในพูลที่ระบุ แบบไดนามิกและกำหนดให้แก่ผู้ใช้

แอ็ตทริบิวต์ Vendor Specific

ผู้จำหน่ายซอฟต์แวร์ อิสระ (ISV) บางรายไม่สามารถใช้แอ็ตทริบิวต์ **Framed-Pool** ได้ แต่มีความสามารถในการกำหนดพูล IP address เซิร์ฟเวอร์ RADIUS สามารถใช้ประโยชน์พูลแอดเดรสเหล่านี้ได้โดยใช้โมเดล Vendor-Specific Attribute (VSA) ตัวอย่าง Cisco NAS จัดให้มีแอ็ตทริบิวต์ชื่อ Cisco-AVPair ไฟล์พจนานุกรมในเซิร์ฟเวอร์ RADIUS มีแอ็ตทริบิวต์ นี้:

```
VENDOR     Cisco          9
ATTRIBUTE  Cisco-AVPair   1       string
```

เมื่อ NAS ส่งแพ็กเก็ต Access-Request จะรวมแอดเดรสนี้กับ `Cisco-AVPair="ip:addr-pool=`*poolname*`"` โดยที่ *poolname* คือชื่อของพูลแอดเดรสที่กำหนดอยู่บน NAS หลังจากการร้องขอได้รับการพิสูจน์ตัวตน และได้รับอนุญาต เซิร์ฟเวอร์ RADIUS จะส่งแอ็ตทริบิวต์กลับในแพ็กเก็ต Access-Accept จากนั้น NAS สามารถใช้พูดที่กำหนดเพื่อจัดสรร IP address ให้แก่ผู้ใช้ ผู้ดูแลระบบต้องตั้งค่า NAS และอัพเดตแอ็ตทริบิวต์ การอนุญาตสำหรับผู้ใช้โดยการรวมแอ็ตทริบิวต์ VSA ในไฟล์ `default.auth` โกลบอล หรือไฟล์ `user.auth` อย่างใดอย่างหนึ่งบนเซิร์ฟเวอร์ RADIUS

IP Pooling ด้านเซิร์ฟเวอร์ RADIUS

เซิร์ฟเวอร์ RADIUS สามารถถูกตั้งค่าเพื่อสร้าง IP address จากพูลของ IP addresses IP address ถูกส่งกลับในแอ็ตทริบิวต์ Framed-IP-Address ของแพ็กเก็ต Access-Accept

ผู้ดูแลระบบ สามารถกำหนดพูลของ IP addresses โดยใช้อินเตอร์เฟส SMIT แอดเดรส ถูกเก็บรักษาในไฟล์ `/etc/radius/ippool_def` *Poolnames* ถูก กำหนดในไฟล์ `etc/radius/clients` ผู้ดูแลระบบ ยังต้องตั้งค่าหมายเลขพอร์ต NAS daemon เซิร์ฟเวอร์ RADIUS ใช้ข้อมูลจากไฟล์ `etc/radius/clients` และ `/etc/radius/ippool_def` เพื่อสร้างไฟล์ข้อมูล เมื่อ daemon เริ่มทำงาน ผู้ดูแลระบบ ไม่สามารถเปลี่ยนแปลงหรือเพิ่ม *poolnames* หรือช่วง IP address ได้จนกว่า เซิร์ฟเวอร์ RADIUS จะหยุดทำงาน เมื่อ daemon เซิร์ฟเวอร์ RADIUS เริ่มทำงาน จะอ่านไฟล์คอนฟิกูเรชัน (`/etc/radius/radius.conf`) และถ้าเปิดใช้งาน IP Allocation (`Enable_IP_Pooling=YES`) จะตั้งค่าแฟล็กการจัดสรร IP โกลบอล (`IP_pool_flag`) เป็น On จากนั้น daemon ตรวจสอบเพื่อดูว่าไฟล์ `poolname.data` มีอยู่หรือไม่ ถ้ามี จะอาจไฟล์และเก็บข้อมูลนั้น ในหน่วยความจำที่แบ่งใช้ จากนั้นอัพเดตไฟล์และหน่วยความจำที่แบ่งใช้โดยยึดตาม การร้องขอที่มาจากไคลเอ็นต์ ถ้าไฟล์ไม่มีอยู่ daemon จะสร้างไฟล์ใหม่โดยใช้ข้อมูลจากไฟล์ `etc/radius/clients` และ `/etc/radius/ippool_def` ไฟล์ `poolname.data` มีการจำกัดขนาดสูงสุด 256 MB (ขีดจำกัดขนาดเซ็กเมนต์ AIX) ถ้าไฟล์ `poolname.data` มีขนาดใหญ่กว่า 256 MB เซิร์ฟเวอร์ RADIUS บันทึกข้อความแสดงความผิดพลาดและออกจากการทำงาน

daemon รับละเอียด IP-pool จากไฟล์ `/etc/radius/ippool_def` และเก็บรักษาตารางของ IP แอดเดรสสำหรับชื่อพูลแต่ละชื่อใน หน่วยความจำที่แบ่งใช้ ตารางมีรายการสำหรับแฟล็ก NAS-IP-address, NAS-port และ IN USE daemon เก็บรักษาตาราง hash ที่ถูกคีย์โดย NAS-IP NAS-port เมื่อมีการร้องขอมาจากหลายผู้ใช้ UDP จะจัดคิวการร้องขอ และ daemon เรียกข้อมูล NAS-IP และ NAS-port จากการร้องขอ การใช้ข้อมูลนั้น จะตรวจสอบเพื่อดูว่า *poolname* ถูกกำหนดสำหรับ NAS นั้นหรือไม่โดยการตรวจสอบข้อมูล ที่อ่านจากไฟล์ `etc/radius/clients`

daemon พยายามรับค่าแอดเดรสที่ไม่ใช่จากพูล ถ้ามีแอดเดรส ที่ไม่ใช้อยู่ จะถูกทำเครื่องหมายเป็น “in use” โดยแฟล็ก NAS-IP และ NAS-port และถูกส่งกลับไปยังเซิร์ฟเวอร์ RADIUS IP address ถูกนำไปไว้ในแอ็ตทริบิวต์ **Framed-IP-Address** โดย daemon และส่งกลับ ไปยัง NAS ในแพ็กเก็ตการยอมรับ ไฟล์ `poolname.data` ยังถูกอัพเดตเพื่อให้ตรงกับข้อมูลในหน่วยความจำที่แบ่งใช้

ถ้า พูลไม่มีอยู่ หรือมีอยู่แต่ไม่มีแอดเดรสที่ไม่ใช้งานใดๆ ข้อผิดพลาดจะถูกส่งกลับไปยังเซิร์ฟเวอร์ RADIUS ข้อผิดพลาด `Could not allocate IP address` ถูกบันทึกลงในล็อกไฟล์และแพ็กเก็ต Access-Reject ถูกส่งไปยัง NAS โดยเซิร์ฟเวอร์ RADIUS

โค้ดระบุความผิดพลาดคือ:

- NOT_POOLED – ไม่มีพูลถูกกำหนดสำหรับ **nas_ip**
- POOL_EXHAUSTED – พูลถูกกำหนดสำหรับ **nas_ip** แต่แอดเดรสทั้งหมดในพูลขณะนี้ใช้งานอยู่

เมื่อมีการร้องขอการพิสูจน์ตัวตนจากการรวม NAS และ NAS-port ที่มี IP แอดเดรสถูกจัดสรรไว้แล้ว daemon จะส่งกลับ การจัดสรรก่อนหน้าไปที่พูล โดยการทำเครื่องหมายแฟล็ก IN USE เป็น Off และลบรายการ NAS-IP-address และ NAS-port ในตาราง จากนั้นจัดสรร IP address ใหม่จากพูล

IP address ยังถูกส่งกลับไปที่พูลเมื่อเซิร์ฟเวอร์ RADIUS ได้รับแพ็กเก็ต Accounting-Stop จาก NAS แพ็กเก็ต Accounting-Stop ต้องมีรายการ NAS-IP-address และ NAS-port daemon เข้าถึงไฟล์ `ippool_mem` สำหรับกรณีต่อไปนี้:

- การร้องขอมีเข้ามาเพื่อขอ IP address ใหม่ ตั้งค่าแฟล็ก IN USE เป็นจริง
- แพ็กเก็ต Accounting-Stop ได้รับ โดยจะรีลีส IP แอดเดรส โดยการตั้งค่าแฟล็ก “in use” เป็นเท็จ

ในแต่ละกรณี การเรียกใช้ระบบหน่วยความจำที่แบ่งใช้เพื่อให้มั่นใจว่า ข้อมูลในหน่วยความจำที่แบ่งใช้ และไฟล์ `poolname.data` ตรงกัน ผู้ดูแลระบบสามารถกำหนดให้การจัดสรร IP เป็น `ON` หรือ `OFF` โดยใช้ พารามิเตอร์ *Enable_IP_Pooling* ในไฟล์คอนฟิกูเรชันเซิร์ฟเวอร์ RADIUS (`radiusd.conf`) นี้ มีประโยชน์ในกรณีที่ผู้ดูแลระบบมี IP แอดเดรสที่กำหนด อยู่ในไฟล์ `default.auth` หรือ `user.auth` โกลบอล ในการใช้ IP แอดเดรสที่กำหนด ผู้ดูแลระบบต้องตั้งค่า `Enable_IP_Pool = NO`

ตัวอย่างของไฟล์ `/etc/radius/ippool_def` ที่สร้างผ่าน SMIT:

ชื่อพูล	ช่วงเริ่ม	ช่วงสิ้นสุด
Floor5	192.165.1.1	192.165.1.125
Floor6	192.165.1.200	192.165.1.253

ต่อไปนี้คือตัวอย่างของไฟล์ `/etc/radiusclients` ที่สร้างผ่าน SMIT:

NAS-IP	ความลับที่แบ่งใช้	ชื่อพูล
1.2.3.4	Secret1	Floor5
1.2.3.5	Secret2	Floor6
1.2.3.6	Secret3	Floor5
1.2.3.7	Secret4	

ในตัวอย่างด้านบนสำหรับ NAS-IP-Address 1.2.3.7 ชื่อพูล เป็นค่าว่าง ในกรณีนี้ ไม่มีการทำ IP pooling สำหรับ NAS นี้ (แม้ว่า โกลบอล `IP_pool_flag = True`) เมื่อแพ็กเก็ต Access-Request เข้ามา เซิร์ฟเวอร์ RADIUS ทำการพิสูจน์ตัวตนและการอนุญาต ถ้าสำเร็จ เซิร์ฟเวอร์ส่ง IP address สแตติกที่กำหนดในการร้องขอ หรือจากไฟล์ `default.auth` หรือไฟล์ `user.auth` โกลบอล ในแพ็กเก็ต Access-Accept ในกรณีนี้ ไม่จำเป็นต้องใช้แอ็ตทริบิวต์ NAS-Port

ถ้า IP pooling เป็น `True` ผู้ดูแลระบบยังได้กำหนด IP แอดเดรสสแตติกเป็นส่วนหนึ่ง ของ `default.auth` หรือ `user.auth` โกลบอล หรือเป็นส่วนหนึ่งของแพ็กเก็ต Access-Request เซิร์ฟเวอร์ RADIUS แทน IP แอดเดรสนั้นด้วย IP แอดเดรสที่จัดสรรจากชื่อพูล ที่กำหนดสำหรับ NAS นั้น ถ้า IP แอดเดรสทั้งหมดในพูลถูกใช้งานอยู่ เซิร์ฟเวอร์จะบันทึกข้อผิดพลาด (พูลเต็ม) และส่งแพ็กเก็ต Access-Reject เซิร์ฟเวอร์ไม่สนใจ IP แอดเดรสสแตติกใดๆ ที่กำหนดในไฟล์ `auth`

ถ้า IP pooling เป็น `True` และชื่อพูลที่ถูกต้องถูกำหนดไว้ สำหรับ NAS เมื่อมีแพ็กเก็ต Access-Request เข้ามาจาก NAS-IP นั้น และไม่มี NAS-Port ถูกกำหนด เซิร์ฟเวอร์จะส่งแพ็กเก็ต Access-Reject

ต่อไปนี้คือตัวอย่างของไฟล์ `Floor5.data` ที่สร้างโดย daemon:

IP Address	NAS-IP	NAS-Port	In Use
192.165.1.1	1.2.3.4	2	1
192.165.1.2	1.2.3.4	3	0
.....	.....	....	....
192.165.1.124	1.2.3.6	1	1
192.165.1.125	1.2.3.6	6	1

ต่อไปนี้คือตัวอย่างของไฟล์ `Floor6.data` ที่สร้างโดย daemon:

IP Address	NAS-IP	NAS-Port	In Use
192.165.200	1.2.3.4	1	1
192.165.201	1.2.3.4	4	1
.....	.....	....	....
192.165.1.252	1.2.3.4	5	0
192.165.1.253	1.2.3.4	6	1

เมื่อจำเป็นต้องรีลีส IP addresses ที่จัดสรรไว้ทั้งหมด สำหรับ NAS ที่ระบุ (ตัวอย่าง เมื่อ NAS หยุดทำงาน) อาจจำเป็น ต้องรีลีส IP addresses ทั้งหมดจากพูลทั้งหมดเพื่อเตรียมข้อมูลเบื้องต้นไฟล์ *poolname*`.data` ผู้ดูแลระบบสามารถทำการดำเนินการเมนูต่อไปนี้โดยใช้ SMIT:

- Clear IP Pool for a Client
- Clear entire IP Pool

แผง SMIT สำหรับ IP Pool

ใน Client Configuration, **Add a Client** คุณสามารถป้อน **Pool Name** ที่เป็นทางเลือก ชื่อสามารถยาวสูงสุด 64 อักขระ เมื่อ **Pool Name** เป็นค่าว่าง IP pooling จะไม่ถูกทำ และเซิร์ฟเวอร์ RADIUS กำหนด IP address ที่กำหนดโดยผู้ดูแลระบบผ่าน แอ็ตทริบิวต์การอนุญาต **Framed-IP-Address**

เมื่อ **IP Pool** ถูกเลือก อ็อพชันต่อไปนี้จะแสดง:

- List all IP Pools
- Create an IP Pool
- Change/Show Characteristics of an IP Pool
- Delete an IP Pool
- Clear IP Pool for a Client
- Clear entire IP Pool

List all IP Pools: ใช้อ็อพชันนี้ เพื่อแสดงรายการ **Pool Name, Start Range IP address** และ **Stop Range IP address**

Create an IP Pool: ใช้อ็อพชันนี้เพื่อเพิ่มชื่อพูล ช่วงเริ่มต้น และช่วงสิ้นสุด ข้อมูลนี้ถูกต่อท้ายด้านล่างของไฟล์ `ippool_def` การตรวจสอบถูกทำเพื่อให้แน่ใจว่าไม่มีชื่อพูลซ้ำ และ ช่วง IP address ไม่ต่อเนื่องกัน การดำเนินการนี้สามารถดำเนินการ เมื่อ daemon เซิร์ฟเวอร์ RADIUS ไม่ได้ทำงานอยู่เท่านั้น

Change/Show Characteristics of an IP Pool: อ็อพชันนี้แสดงรายการ ชื่อพูลในแผงป๊อบอัพ จากแผงนี้ คุณต้องเลือก ชื่อพูลที่เจาะจง เมื่อคุณเลือกชื่อพูล แผงที่มีชื่อที่เลือก จะแสดง เมื่อคุณกด Enter ข้อมูลสำหรับชื่อพูลนั้น จะถูกอัพเดตในไฟล์ `ippool_def` การดำเนินการนี้สามารถดำเนินการเมื่อ daemon เซิร์ฟเวอร์ RADIUS ไม่ได้ทำงานอยู่ เท่านั้น

Delete an IP Pool: การเลือก อ็อพชันนี้แสดงรายการของชื่อพูลที่คณสามารถเลือกได้ เมื่อ คุณเลือกชื่อพูล แผงป๊อบอัพ **Are You Sure** จะแสดงขึ้นเพื่อให้คำยืนยันก่อนที่ชื่อพูลที่เลือก จะถูกลบ สคริปต์ **rmippool** ถูกเรียกใช้ เพื่อลบชื่อพูลที่เลือกออกจากไฟล์ `ippool_def` การดำเนินการนี้สามารถดำเนินการเมื่อ daemon เซิร์ฟเวอร์ RADIUS ไม่ได้ทำงานอยู่ เท่านั้น

Clear IP Pool for a Client: อ็อพชันนี้ทำเครื่องหมายรายการ **IN-USE** เป็น 0 สำหรับ IP addresses ที่เป็นของ NAS ซึ่งหมายความว่า IP addresses ทั้งหมด สำหรับ NAS นี้จะพร้อมใช้งานได้ในตอนนี้ การดำเนินการนี้สามารถดำเนินการเมื่อ daemon เซิร์ฟเวอร์ RADIUS ไม่ได้ทำงานอยู่เท่านั้น

Clear Entire IP Pool: เมื่อเลือกอ็อพชันนี้ แผงป๊อบอัพ **Are You Sure** จะแสดงเพื่อให้คำยืนยัน ก่อนที่ทั้งไฟล์ `ippool_mem` จะถูกทำให้ว่าง การดำเนินการนี้สามารถดำเนินการเมื่อ daemon เซิร์ฟเวอร์ RADIUS ไม่ได้ทำงานอยู่ เท่านั้น

พาเนล RADIUS SMIT

เมื่อใช้ SMIT เพื่อตั้งค่าเซิร์ฟเวอร์ RADIUS ฟิลด์ ที่ทำเครื่องหมายด้วยเครื่องหมายดอกจัน (*) เป็นฟิลด์ที่จำเป็นต้องมี

พาธด่วน SMIT คือ:

```
smitty radius
```

เมนูหลักของ RADIUS เป็นดังนี้:

```
       RADIUS Server

Configure Server
Configure Clients
Configure Users
Configure Proxy Rules
Advanced Server Configuration
Start RADIUS Server daemons
Stop RADIUS Server daemons
```

การแสดงภาพหน้าจอต่อไปนี้จะแสดงตัวอย่างพาเนล RADIUS Configure Server SMIT:

```
Configure Server
  RADIUS Directory                          /etc/radius
* Database Location                         [Local]                  +
  Local AVL Database File Name              [dbdata.bin]
  Debug Level                               [9]                      +#
  Local Accounting                          [ON]                     +
  Local Accounting Directory                [/var/radius/data/accou>
  Accept Reply-Message                      []
  Reject Reply-Message                      []
  Challenge Reply-Mesage                    []
  Password Expired Reply-Message            []
  Support Renewal of Expired Password       [NO]                     +
  Require Message Authenticator             [NO]                     +
* Authentication Port Number                [1812]
* Accounting Port Number                    [1813]
  LDAP Server Name                          []
  LDAP Server Port Number                   [389]                     #
  LDAP Server Admin Distinguished Name      [cn=root]
  LDAP Server Admin Password                []
  LDAP Base Distinguished Name              [cn=aixradius]
  LDAP Size Limit                           [0]                       #
  LDAP Hop Limit                            [0]                       #
  LDAP wait time limit                      [10]                      #
  LDAP debug level                          [0]                      +#
  Proxy Allowed                             [OFF]                    +
  Proxy Use Table                           [OFF]                    +
  Proxy Realm Name                          []
  Proxy Prefix Delimiters                   [$/]
  Proxy Suffix Delimiters                   [@.]
  Proxy Remove Hops                         [NO]                     +
  Proxy Retry Count                         [2]                       #
  Proxy Timeout                             [30]                      #
  UNIX Check Login Restrictions             [OFF]                    +
  Enable IP Pool                            [OFF]                    +
  Send Message Authenticator for ACCEPT     [ON]                     +
  Maximum RADIUS Server Threads             [15]                      #
  EAP Conversation Timeout (Seconds)        [30]                      #
  Enable EAP-TLS                            [ON]                     +
  Required Options for EAP-TLS
       Path to OpenSSL Library              [/opt/freeware/lib/libs>
       OpenSSL Cipher List                  [ALL:!ADH:RC4+RSA:+SSLv>
       Root CA Directory (Full Path)        [/etc/radius/tls]
       Root CA Certificate (Full Path)      [/etc/radius/tls/radius>
       RADIUS Server Certificate (Full Path) [/etc/radius/tls/cert-s>
       RADIUS Server Private Key (Full Path) [/etc/radius/tls/cert-s>
       RADIUS Server CRL (Full Path)        []
```

ข้อมูลวิธีใช้ SMIT โดยละเอียดมีให้สำหรับทุกฟิลด์และ เมนูอ็อพชันโดยการกดคีย์ **F1**

ตัวสร้างเลขสุ่ม

เลขสุ่มจำเป็นต้องใช้เมื่อสร้างฟิลด์ Authenticator ของแพ็กเก็ต RADIUS

เป็นสิ่งสำคัญที่ต้องมีตัวสร้างค่าทีเป็นไปได้ที่ดีที่สุดเนื่องจากผู้บุกรุก สามารถพยายามปลอมเป็นเซิร์ฟเวอร์ RADIUS ในการตอบกลับการร้องขอที่คาดคะเน จากนั้นใช้การตอบกลับเพื่อปลอมเป็นเซิร์ฟเวอร์ RADIUS ในการร้องขอ เพื่อการเข้าถึงใน

อนาคต เซิร์ฟเวอร์ AIX RADIUS ใช้ส่วนขยายเคอร์เนล **/dev/urandom** เพื่อ สร้างเลขสุ่มเทียม ส่วนขยายเคอร์เนลนี้รวบรวมตัวอย่างเอ็นโทรฟี จากแหล่งที่มาฮาร์ดแวร์เป็นตัวอย่างไดรเวอร์อุปกรณ์เทียม อุปกรณ์นี้ ผ่านการทดสอบ NIST เพื่อให้มั่นใจว่ามีการสุ่มอย่างเหมาะสม

การเปิดใช้งาน Globalization

คำสั่ง RADIUS **raddbm** และพาเนล SMIT เปิดใช้งาน globalization และแต่ละคำสั่งใช้การเรียกใช้ AIX globalization API มาตรฐาน เพื่อจัดเตรียมฟังก์ชันนี้

ข้อมูลที่เกี่ยวข้อง

คำสั่ง: **installp**,**mkuser** และ **raddbm**

การขัดขวางการบุกรุก AIX

การขัดขวางการบุกรุก AIX ตรวจหาความไม่เหมาะสม ที่ไม่ได้รับอนุญาต หรือข้อมูลอื่นที่อาจ ถูกพิจารณาว่าเป็นอันตรายต่อระบบ

ส่วนต่อไปนี้อธิบายการตรวจหาการบุกรุกประเภทต่างๆ ที่หลากหลาย ที่จัดให้มีโดย AIX

ข้อมูลที่เกี่ยวข้อง

คำสั่ง: **chfilt, ckfilt, expfilt, genfilt, impfilt, lsfilt, mkfilt, mvfilt, rmfilt**

การตรวจหาการบุกรุก

การตรวจหาการบุกรุกเป็นเหตุการณ์การดำเนินการของระบบการมอนิเตอร์และวิเคราะห์ เพื่อเข้าขัดขวางและปฏิเสธความพยายามใดๆ เพื่อเข้าถึงระบบโดยไม่ได้รับ อนุญาต ใน AIX การตรวจหา การเข้าถึงที่ไม่ได้รับอนุญาต หรือการพยายามเข้าถึงที่ไม่ได้รับอนุญาตนี้กระทำโดยการตรวจดู การดำเนินการบางอย่าง จากนั้นบังคับใช้กฎตัวกรองกับการดำเนินการเหล่านี้

หมายเหตุ: คุณต้องติดตั้งชุดไฟล์ `bos.net.ipsec` บนระบบโฮสต์เพื่อเปิดใช้การตรวจหาการบุกรุก เทคโนโลยีการตรวจหาสร้างขึ้นบนคุณลักษณะ AIX Internet Protocol Security (IPsec) ที่มีอยู่แล้ว

กฎตัวกรองโดยการจับคู่รูปแบบ:

การจับคู่รูปแบบเป็นการใช้กฎตัวกรอง IPsec เพื่อกรอง แพ็กเก็ตเน็ตเวิร์ก รูปแบบการกรองสามารถเป็นสตริงข้อความ สตริงฐานสิบหก หรือไฟล์ที่มีมากกว่าหนึ่งรูปแบบ หลังจากสร้างกฎตัวกรอง รูปแบบแล้ว และตรวจสอบรูปแบบนั้นในเนื้อหาของแพ็กเก็ตเน็ตเวิร์ก การดำเนินการที่กำหนดไว้แล้วของกฎตัวกรองจะแสดงผล

กฎตัวกรองโดยการจับคู่รูปแบบมีผลใช้กับแพ็กเก็ตเน็ตเวิร์กขาเข้าเท่านั้น ใช้คำสั่ง **genfilt** เพื่อเพิ่มกฎตัวกรองลงในตารางกฎตัวกรอง กฎตัวกรองที่สร้างขึ้น โดยคำสั่งนี้ถูกเรียกเป็นกฎตัวกรองด้วยตนเอง ใช้คำสั่ง **mkfilt** เพื่อเรียกทำงาน หรือเลิกทำงานกฎตัวกรอง คำสั่ง **mkfilt** ยังสามารถใช้ควบคุมฟังก์ชันบันทึกการทำงานการควบคุม

ไฟล์รูปแบบ สามารถมีรายการรูปแบบข้อความ หรือรูปแบบฐานสิบหกหนึ่งรายการต่อหนึ่งบรรทัด กฎตัวกรองโดยการจับคู่รูปแบบสามารถใช้เพื่อป้องกันไวรัส บัฟเฟอร์ โอเวอร์โฟลว์ และการโจมตีเน็ตเวิร์กความปลอดภัยอื่นๆ

กฎตัวกรองโดย การจับคู่รูปแบบอาจส่งผลด้านลบต่อผลการทำงานระบบถ้าถูกใช้แพร่หลาย เกินไป และถ้ามีจำนวนรูปแบบมากเกินไป ทางที่ดีที่สุดคือให้ขอบเขต การนำใช้วิธีนี้เฉพาะที่จำเป็นเท่าที่เป็นไปได้ ตัวอย่าง ถ้ารูปแบบไวรัส ที่รู้จักใช้กับ **sendmail** ให้ระบุปลายทาง **sendmail** SMTP พอร์ต 25 ในกฎตัวกรอง นี้อนุญาตให้การรับส่งอื่นๆ ทั้งหมดผ่านได้โดยไม่ก่อให้เกิดผลกระทบต่อผลการทำงานจากการจับคู่รูปแบบ

คำสั่ง **genfilt** ทราบและเข้าใจรูปแบบที่ใช้ในบาง เวอร์ชันของ ClamAV

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง genfilt

คำสั่ง mkfilt

เว็บไซต์ ClamAV

ประเภทของรูปแบบ:

มีประเภทระดับต้นของรูปแบบสามประเภท: ข้อความ เลขฐานสิบหก และ ไฟล์ กฎตัวกรองการจับคู่รูปแบบใช้กับแพ็กเก็ตขาเข้าเท่านั้น

รูปแบบข้อความ

รูปแบบตัวกรองข้อความคือสตริง ASCII ที่คล้ายกับตัวอย่างต่อไปนี้:

```
GET /../../../../../../../../
```

รูปแบบเลขฐานสิบหก

รูปแบบเลขฐานสิบหกคล้ายกับ ตัวอย่างต่อไปนี้:

```
0x33c0b805e0cd16b807e0cd1650558becc7460200f05d0733ffb8c800b9fffff3abb00150
e670e47132c0e67158fec03c8075f033c033c9b002fa99cd26fb4183f90575f5c3
```

หมายเหตุ: รูปแบบเลขฐานสิบหกแตกต่างจากรูปแบบข้อความตรงที่มี 0x นำหน้า

ไฟล์ที่มีรูปแบบข้อความ

ไฟล์สามารถมี รายการของรูปแบบข้อความ หนึ่งรายการต่อหนึ่งบรรทัดของรูปแบบข้อความหรือรูปแบบเลขฐานสิบหก ไฟล์รูปแบบตัวอย่าง สามารถดูได้ที่ http://www.clamav.net

Shun port และ shun host filter rules:

โดยการตั้งค่า shun filter rule คุณสามารถกันโฮสต์รีโมตหรือ คู่โฮสต์รีโมตและพอร์ตจากการเข้าถึงเครื่องโลคัล

กฎตัวกรอง shun สร้างกฎที่มีผลที่ปฏิเสธ โฮสต์รีโมตหรือโฮสต์รีโมตและคู่พอร์ตมิให้เข้าถึงเครื่องโลคัล เมื่อตรงตามเกณฑ์ที่ระบุของกฎ

เนื่องจากเป็นเรื่องปกติสำหรับการโจมตีที่เกิดขึ้นก่อนโดยการสแกนพอร์ต กฎ การกรอง shun พอร์ตจะมีประโยชน์อย่างมากในการป้องกันการบุกรุกโดยการตรวจหา พฤติกรรมการโจมตีนี้

ตัวอย่าง ถ้าโฮสต์โลคัลไม่ได้ใช้เซิร์ฟเวอร์พอร์ต 37 ซึ่งเป็น เซิร์ฟเวอร์เวลา ดังนั้นโฮสต์รีโมตไม่ควรเข้าถึงพอร์ต 37 ยกเว้นกำลังทำการสแกนพอร์ต เพิ่ม shun port filter rule บนพอร์ต 37 เพื่อที่ว่า ถ้าโฮสต์รีโมตพยายามเข้าถึงพอร์ตนั้น shun filter rule จะสร้าง กฎที่มีผลที่บล็อกโฮสต์มิให้เข้าถึงได้ในช่วงเวลา หนึ่งตามที่ระบุใน shun rule ในฟิลด์ **expiration time**

ถ้าฟิลด์ **expiration time** ของ shun rule ถูกตั้งค่าเป็น 0 shun rule ที่มีผลซึ่งสร้างขึ้นแบบไดนามิกจะไม่มีการหมดอายุ

หมายเหตุ:

1. เวลาหมดอายุที่ระบุโดย shun port filter rule บังคับใช้ กับกฎที่มีผลซึ่งสร้างขึ้นแบบไดนามิกเท่านั้น
2. กฎที่มีผลซึ่งสร้างขึ้นแบบไดนามิกสามารถดูได้ด้วยคำสั่ง **lsfilt -a** เท่านั้น

กฎตัวกรองโฮสต์ Shun

เมื่อตรงตามเกณฑ์ของกฎ ตัวกรองโฮสต์ shun กฎที่มีผลซึ่งสร้างขึ้นแบบไดนามิกจะ บล็อกหรือ shun การรับส่งในเน็ตเวิร์กทั้งหมดจากโฮสต์รีโมตตามช่วงเวลาการหมดอายุ ที่ระบุ

กฎตัวกรองพอร์ต Shun

เมื่อตรงตามเกณฑ์ของกฎตัวกรอง พอร์ต shun กฎที่มีผลที่สร้างขึ้นแบบไดนามิกจะ บล็อกหรือ shun การรับส่งในเน็ตเวิร์กจากพอร์ตเฉพาะของโฮสต์รีโมตนี้เท่านั้น จนกว่าจะพ้นเวลาหมดอายุ

กฎตัวกรอง Stateful:

ตัวกรอง Stateful ตรวจสอบข้อมูลเช่น แอดเดรสต้นทางและปลายทาง หมายเลขพอร์ต และสถานะ จากนั้น โดยการนำใช้กฎตัวกรอง IF, ELSE และ ENDIF กับแฟล็กส่วนหัวเหล่านี้ ระบบ stateful สามารถทำการตัดสินใจเกี่ยวกับการกรอง ในบริบทของทั้งเซสชันแทนการทำกับแต่ละแพ็กเก็ต และข้อมูลส่วนหัว

การตรวจหาแบบ Stateful ตรวจสอบแพ็กเก็ตการสื่อสารขาเข้าและขาออก เมื่อกฎตัวกรอง stateful ถูกเรียกทำงานด้วยคำสั่ง **mkfilt -u** กฎในบล็อก ELSE จะถูกตรวจสอบเสมอจนกว่าจะต้องกับกฎ IF หลังจากเป็นไปตามกฎหรือเงื่อนไข IF กฎในบล็อก IF ถูกใช้ จนกว่ากฎตัวกรองจะถูกเรียกทำงานอีกครั้งด้วยคำสั่ง **mkfilt -u**

คำสั่ง **ckfilt** จะตรวจสอบไวยากรณ์ของกฎตัวกรอง stateful และแสดงในจอแสดงผล ในลักษณะอธิบายให้เห็นชัดเจนดังตัวอย่างต่อไปนี้:

```
%ckfilt -v4
Beginning of IPv4 filter rules.
Rule 2
IF Rule 3
    IF Rule 4
        Rule 5
    ELSE Rule 6
        Rule 7
    ENDIF Rule 8
ELSE Rule 9
    Rule 10
ENDIF Rule 11
Rule 0
```

กฎที่ตั้งเวลา:

กฎที่ตั้งเวลาระบุระยะเวลาเป็นวินาทีที่กฎตัวกรอง ถูกนำใช้หลังถูกทำให้มีผลใช้ด้วยคำสั่ง **mkfilt -v [4|6] -u**

เวลาหมดอายุถูกระบุด้วยคำสั่ง **genfilt -e** สำหรับข้อมูลเพิ่มเติม ดูที่คำสั่ง **mkfilt** และ **genfilt**

หมายเหตุ: ตัวจับเวลาไม่มีผลต่อกฎ IF, ELSE หรือ ENDIF ถ้าเวลา หมดอายุถูกระบุในกฎการสับเปลี่ยนโฮสต์สับเปลี่ยนพอร์ต เวลามีผลกับ กฎที่มีผลที่ถูกเริ่มต้นโดยกฎการสับเปลี่ยนเท่านั้น กฎการสับเปลี่ยน ไม่มีเวลาหมดอายุ

การเข้าถึงกฎตัวกรองจาก SMIT

คุณสามารถตั้งค่ากฎจาก SMIT

ในการตั้งค่ากฎตัวกรองจาก SMIT ดำเนินขั้นตอนต่อไปนี้

1. จากบรรทัดคำสั่ง ป้อนคำสั่งต่อไปนี้:`smitty ipsec4`
2. เลือก **Advanced IP Security Configuration**
3. เลือก **Configure IP Security Filter Rules**
4. เลือก **Add an IP Security Filter Rule**

```
                        เพิ่ม IP Security Filter Rule

พิมพ์หรือเลือกค่าในฟิลด์ที่ต้องป้อนข้อมูล
กด Enter หลังทำการเปลี่ยนแปลงที่ต้องการทั้งหมดเสร็จ

[TOP]                                                [Entry Fields]
* Rule Action                                       [permit]              +
* IP Source Address                                 []
* IP Source Mask                                    []
  IP Destination Address                            []
  IP Destination Mask                               []
* Apply to Source Routing? (PERMIT/inbound only)    [yes]                 +
* Protocol                                          [all]                 +
* Source Port / ICMP Type Operation                 [any]                 +
* Source Port Number / ICMP Type                    [0]                    #
* Destination Port / ICMP Code Operation            [any]                 +
* Destination Port Number / ICMP Type               [0]                    #
* Routing                                           [both]                +
* Direction                                         [both]                +
* Log Control                                       [no]                  +
* Fragmentation Control                             [0]                   +
* Interface                                         []                    +
  Expiration Time (sec)                             []                     #
  Pattern Type                                      [none]                +
  Pattern / Pattern File                            []
  Description                                       []


โดยที่ "Pattern Type" เป็นค่าใดค่าหนึ่งต่อไปนี้
  x   none                                                                x#
  x   pattern                                                             x
  x   file                                                                x
  x   Anti-Virus patterns
```

ตัวเลือกสำหรับฟิลด์ action **คือ:** `permit, deny, shun_host, shun_port, if, else, endif`

ถ้าไฟล์รูปแบบ ถูกระบุ ไฟล์นั้นต้องอ่านได้เมื่อกฎตัวกรองถูกเรียกทำงาน ด้วยคำสั่ง **mkfilt –a** กฎตัวกรองถูกเก็บ ในฐานข้อมูล /etc/security/ipsec_filter

---

AIX Security Expert

AIX Security Expert จัดให้มี ศูนย์กลางสำหรับค่าติดตั้งความปลอดภัยทั้งหมด (TCP, NET, IPSEC, ระบบ และการตรวจสอบ)

AIX Security Expert คือ เครื่องมือที่ช่วยให้ระบบมีความปลอดภัยมากขึ้น คำสั่งนี้เป็นส่วนหนึ่งของชุดไฟล์ **bos.aixpert** AIX Security Expert มี ค่าติดตั้งเมนูแบบง่ายสำหรับ High Level Security, Medium Level Security, Low Level Security และความปลอดภัย AIX Standard Settings ที่รวมการตั้งค่าความปลอดภัยมากกว่า 300 การตั้งค่า ขณะที่ยังคงมีการควบคุมองค์ประกอบความปลอดภัยแต่ละองค์ประกอบสำหรับผู้ดูแลระบบ ระดับสูง AIX Security Expert สามารถ ใช้เพื่อประยุกต์ใช้ระดับความปลอดภัยที่เหมาะสม โดยไม่จำเป็น ต้องอ่านเอกสารจำนวนมากเกี่ยวกับการทำให้มีความปลอดภัยมากขึ้น จากนั้นทำการประยุกต์ใช้องค์ประกอบความปลอดภัยแต่ละองค์ประกอบเฉพาะตัว

AIX Security Expert สามารถ ใช้เพื่อสแน็ปช็อตการตั้งค่าความปลอดภัย สแน็ปช็อตนี้สามารถใช้เพื่อตั้งค่า คอนฟิกูเรชันความปลอดภัยเหมือนกับระบบอื่น วิธีนี้ช่วยทั้งประหยัดเวลาและทำให้แน่ใจว่าทุกระบบ มีการตั้งค่าความปลอดภัยที่เหมาะกับสภาวะแวดล้อมองค์กร

AIX Security Expert สามารถรันจาก SMIT, หรือคุณสามารถใช้คำสั่ง **aixpert**

ค่าติดตั้ง AIX Security Expert

ค่าติดตั้ง ความปลอดภัยคร่าวๆ มีดังตอไปนี้:

High Level Security
: ความปลอดภัยระดับสูง

Medium Level Security
: ความปลอดภัยระดับกลาง

Low Level Security
: ความปลอดภัยระดับต่ำ

Advanced Security
: ควาปลอดภัยที่ผู้ใช้กำหนดเอง

AIX ค่าติดตั้งมาตรฐาน
: ความปลอดภัยดีฟอลต์ของระบบดีฟอลต์

Undo Security
: การตั้งค่า AIX Security Expert บางอย่างสามารถเลิกทำ

Check Security
: ให้รายงานแสดงรายละเอียดของค่าติดตั้งความปลอดภัยปัจจุบัน

การทำให้ AIX Security Expert มี ความปลอดภัยมากขึ้น

การทำให้มีความปลอดภัยมากขึ้นช่วยป้องกันองค์ประกอบทั้งหมดของระบบโดย การเพิ่มความปลอดภัยหรือการนำระดับการรักษาความปลอดภัยที่สูงยิ่งขึ้นมาใช้

การทำให้มีความปลอดภัยมากขึ้นช่วยทำให้มั่นใจว่าการตัดสินใจและการตั้งค่า การรักษาความปลอดภัยทั้งหมดมีความถูกต้องและเหมาะสม ค่าติดตั้งการรักษาความปลอดภัย หลายร้อยค่าอาจต้องถูกเปลี่ยนแปลงเพื่อให้ระบบ AIX มีความปลอดภัยมากขึ้น

AIX Security Expert จัดให้มี เมนูที่รวมศูนย์ค่าติดตั้งการรักษาความปลอดภัยทั่วไปที่มีประสิทธิภาพ ค่าติดตั้งเหล่านี้ยึดตามการวิจัยเพิ่มเติมเกี่ยวกับการรักษาความปลอดภัยระบบ UNIX อย่างเหมาะสม โดยจัดให้มีค่าติดตั้งการรักษาความปลอดภัยดีฟอลต์ สำหรับความต้องการของสภาวะแวดล้อมที่มีความปลอดภัย (High Level Security, Medium Level Security และ Low Level Security) และผู้ดูแลระบบ ระดับสูงสามารถตั้งค่าการรักษาความปลอดภัยแต่ละระดับได้โดยอิสระ

การตั้งค่าระบบให้มีระดับการรักษาความปลอดภัยที่สูงเกินไปอาจทำให้มีการปฏิเสธ เซอร์วิสที่จำเป็นต้องใช้ ตัวอย่าง **telnet** และ **rlogin** ถูก ปิดใช้งานสำหรับ High Level Security เนื่องจากรหัสผ่านการล็อกอินถูกส่ง บนเน็ตเวิร์กโดยไม่มีการเข้ารหัส ถ้าระบบถูกตั้งค่าให้มีระดับการรักษาความปลอดภัย ที่ต่ำเกินไป ระบบสามารถมีจุดอ่อนที่ทำให้เกิดการคุกคามด้านความปลอดภัยได้ เนื่องจากแต่ละองค์กรมีความต้องการการรักษาความปลอดภัยเฉพาะของตนเอง ค่าติดตั้งสำหรับ High Level Security, Medium Level Security และ Low Level Security ที่กำหนดไว้แล้วจะเหมาะสมที่สุดสำหรับการใช้เป็นจุดเริ่มการทำงาน สำหรับการตั้งค่าการรักษาความปลอดภัยมากกว่าการตั้งค่าที่ตรงตาม ข้อกำหนดการรักษาความปลอดภัยขององค์กรโดยเฉพาะ

วิธีปฏิบัติในการใช้ AIX Security Expert คือสร้าง ระบบทดสอบ (ในสภาวะแวดล้อมของการทดสอบการใช้งานจริง) ที่คล้ายกับสภาวะแวดล้อม การทำงานจริงที่จะถูกนำไปใช้งาน ติดตั้งแอ็พพลิเคชันทางธุรกิจ ที่จำเป็น และรัน AIX Security Expert ผ่าน GUI AIX Security Expert จะ วิเคราะห์ระบบที่กำลังทำงานนี้ในสถานะที่ได้รับการไว้วางใจนี้ ทั้งนี้ขึ้นอยู่กับ อ็อพชันการรักษาความปลอดภัยที่คุณเลือก AIX Security Expert จะเปิดใช้งาน การป้องกันการสแกนพอร์ต เปิดใช้การตรวจสอบ บล็อกพอร์ต เน็ตเวิร์กที่ไม่ได้ใช้งานโดยแอ็พพลิเคชันทางธุรกิจ หรือเซอร์วิสอื่นๆ ร่วมกับค่าติดตั้งการรักษาความปลอดภัยอื่นอีกมากมาย หลังจากทดสอบกับการตั้งค่า การรักษาความปลอดภัยเหล่านี้อีกครั้ง ระบบก็พร้อมที่จะนำไปใช้ในสภาวะแวดล้อมการทำงานจริง รวมทั้ง ไฟล์ AIX Security Expert XML ที่กำหนดนโยบายการรักษาความปลอดภัยหรือการตั้งค่าของระบบนี้สามารถนำไป ประยุกต์ใช้การตั้งค่าที่เหมือนกันได้โดยง่ายบน ระบบที่คล้ายกันในองค์กรของคุณ

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการทำให้มีความปลอดภัยมากขึ้น ดูที่ NIST Special Publication 800-70, NIST Security Configurations Checklist Program for IT Products

การรักษาความปลอดภัยค่าดีฟอลต์

Secure By Default (SbD) คือแนวคิดในการติดตั้ง ชุดซอฟต์แวร์น้อยที่สุดในการตั้งค่าที่มีความปลอดภัย

อ็อพชันการติดตั้ง AIX Secure by Default (SbD) จะติดตั้งเวอร์ชันไลท์ของ ชุดไฟล์ไคลเอ็นต์และเซิร์ฟเวอร์ TCP ที่ไม่รวมคำสั่งและไฟล์ ที่มีจุดอ่อน ชุดไฟล์ **bos.net.tcp.client** และ **bos.net.tcp.server** เป็นส่วนหนึ่งของการติดตั้ง SbD และมีคำสั่งและไฟล์ทั้งหมด ยกเว้นสำหรับแอ็พพลิเคชันใดๆ ที่อนุญาตให้มีการส่ง รหัสผ่านบนเน็ตเวิร์กในรูปแบบข้อความธรรมดา เช่น **telnet** และ **ftp** นอกจากนั้น แอ็พพลิเคชัน ที่อาจถูกใช้ เช่น **rsh, rcp** และ **sendmail** ไม่รวมใน SbD filesets

กระบวนการอัตโนมัติขั้นสุดท้ายของการติดตั้ง SbD คือการกำหนดค่าติดตั้ง AIX Security Expert การรักษาความปลอดภัยระดับสูง คุณสามารถทำได้โดยการรันคำสั่ง **aixpert** จากสคริปต์ /etc/firstboot: `/usr/sbin/aixpert -f /etc/security/aixpert/core/SbD.xml -p 2>/etc/security/aixpert/log/firstboot.log`

เป็นไปได้ที่จะย้ายเครื่องออกจากโหมด SbD โดยการเปลี่ยน ตัวแปร ODM *SbD_STATE* เป็น sbd_disable การติดตั้ง ชุดไฟล์ **bos.net.tcp.client** และ **bos.net.tcp.server** อีกครั้ง และใช้ AIX Security Expert เพื่อทำให้ ระบบมีระดับการรักษาความปลอดภัยเป็นค่าดีฟอลต์

ไม่สามารถใช้การโอนย้ายระบบการติดตั้งหรือการคงการติดตั้ง เพื่อดำเนินการระบบที่ติดตั้ง SbD สำเร็จ SbD เป็นพาธเมนูติดตั้ง แยก

หมายเหตุ: เมื่อคุณอัพเดตระบบที่อยู่ในโหมด SbD ด้วยเซอร์วิส แพ็ก ระบบที่อัพเดตจะไม่อยู่ในโหมด SbD หลังการอัพเกรด

สามารถมีระบบที่ตั้งค่าอย่างปลอดภัยได้โดยไม่ต้องใช้ อ็อพชันการติดตั้ง SbD ตัวอย่างเช่นอ็อพชันการรักษาความปลอดภัย AIX Security Expert ระดับสูง กลาง หรือต่ำ สามารถกำหนดค่าในการติดตั้งปกติ

ความแตกต่างระหว่างระบบที่ติดตั้ง SbD และการติดตั้งปกติ ที่มีการกำหนดค่าการรักษาความปลอดภัย AIX Security Expert สูงที่แสดงให้เห็นได้ชัดเจนที่สุดคือการตรวจสอบ คำสั่ง **telnet** ในกรณีทั้งสอง คำสั่ง **telnet** ถูกปิดใช้งาน ในการติดตั้ง SbD ไบนารีหรือแอ็พพลิเคชัน **telnet** ไม่คอยถูกติดตั้งบนระบบ

เมื่อใช้การติดตั้ง SbD, เซอร์วิสต่อไปนี้ ไม่ได้ติดตั้งบนระบบในเวลาติดตั้ง หรือ ถูกปิดใช้งาน ด้วยมีบางเซอร์วิสเหล่านี้ที่ไม่ถูกติดตั้งบนระบบ จึงไม่ สามารถเข้าถึง หรือรันคำสั่งเหล่านี้ได้จากระบบ ถ้าคำสั่งและโปรแกรมเหล่านี้จำเป็นต้องใช้ อย่าใช้ อ็อพชันการติดตั้ง SbD นอกจากนั้น ถ้าสคริปต์ รีโมตโปรแกรม หรือชุดไฟล์อิสระใดๆ ต้องการใช้คำสั่งและโปรแกรมเหล่านี้ อย่าใช้อ็อพชันการติดตั้ง SbD

เซอร์วิส	**โปรแกรม**	**อากิวเมนต์**
bootps	/usr/sbin/bootpd	bootpd /etc/bootp
comsat	/usr/sbin/comsat	comsat
exec	/usr/sbin/rexecd	rexecd
finger	/usr/sbin/fingerd	fingerd
ftp	/usr/sbin/ftpd	ftpd
instsrv	/u/netinst/bin/instsrv	instsrv -r /tmp/netinstalllog /u/netinst/scripts
login	/usr/sbin/rlogind	rlogind
netstat	/usr/bin/netstat	netstat -f inet
ntalk	/usr/sbin/talkd	talkd
pcnfsd	/usr/sbin/rpc.pcnfsd	pcnfsd
rexd	/usr/sbin/rpc.rexd	rexd

เซอร์วิส	โปรแกรม	อากิวเมนต์
rquotad	/usr/sbin/rpc.rquotad	rquotad
rstatd	/usr/sbin/rpc.rstatd	rstatd
rusersd	/usr/lib/netsvc/rusers/rpc.rusersd	rusersd
rwalld	/usr/lib/netsvc/rwall/rpc.rwalld	rwalld
shell	/usr/sbin/rshd	rshd
sprayd	/usr/lib/netsvc/spray/rpc.sprayd	sprayd
systat	/usr/bin/ps	ps -ef
talk	/usr/sbin/talkd	talkd
telnet	/usr/sbin/telnetd	telnetd -a
tftp	/usr/sbin/tftpd	tftpd -n
uucp	/usr/sbin/uucpd	uucpd

มีบางฟังก์ชันใน IBM Systems Director Console for AIX, ซึ่งประกอบด้วยพอร์ตเล็ต HealthMetrics, ซึ่งไม่มีอยู่เมื่อคุณกำลังรันระบบปฏิบัติการ AIX ในโหมด SbD คุณสามารถ เปิดใช้งานฟังก์ชันเหล่านั้นโดยติดตั้งชุดไฟล์ที่ต้องการ เพื่อรันฟังก์ชัน

การแจกจ่ายนโยบายด้านความปลอดภัยทาง LDAP

LDAP สามารถใช้เพื่อแจกจ่ายไฟล์คอนฟิกูเรชัน AIX Security Expert XML คุณสามารถใช้ AIX Security Expert เพื่อ ทำสำเนาคอนฟิกูเรชันความปลอดภัยจากระบบหนึ่งไปยังอีกระบบหนึ่ง ซึ่งอนุญาตให้ทำได้สำหรับ ระบบที่คล้ายคลึงกันซึ่งมีคอนฟิกูเรชันด้านความปลอดภัยเหมือนกัน ความสอดคล้องกันนี้ ช่วยลดจุดอ่อนด้านความปลอดภัย

แนวปฏิบัติที่แนะนำคือให้ใช้ AIX Security Expert เพื่อ ตั้งค่าระบบเดียวและตั้งค่าระดับความปลอดภัยให้สอดคล้องกับนโยบายด้านความปลอดภัย ขององค์กร รวมถึงสภาวะแวดล้อมที่ระบบจะต้องดำเนินงาน คอนฟิกูเรชันนี้ ถูกบันทึกค่าขณะนั้นไว้ในไฟล์ `/etc/security/aixpert/core/appliedaixpert.xml` ไฟล์นี้สามารถย้ายไปยังเซิร์ฟเวอร์ LDAP ที่กำหนดคอนฟิกและไว้วางใจ ระบบอื่นๆ ที่มีภาวะเชื่อมต่อกับเซิร์ฟเวอร์ LDAP นี้จะค้นหาไฟล์คอนฟิกูเรชัน XML นี้ผ่านคำสั่ง **aixpertldap**

เซิร์ฟเวอร์ LDAP ที่มีอยู่แล้วเซิร์ฟเวอร์ใดๆ สามารถถูกอัพเดตด้วยสกีมา aixpert นี้เพื่อแจกจ่าย ไฟล์ XML กำหนดคอนฟิก aixpert ไปยังแต่ละไคลเอ็นต์ที่เชื่อมต่อถึง ถ้าเซิร์ฟเวอร์ LDAP ไม่มีสกีมา aixpert ที่อัพเดต ให้อัพเดต aixpert schema ไปยัง LDAP ด้วยคำสั่งต่อไปนี้: `ldapmodify -c -D <bindDN> -w <bindPwd> -i /etc/security/ldap/sec.ldif` เมื่อเซิร์ฟเวอร์ LDAP ถูกอัพเดตด้วยสกีมา aixpert ไคลเอ็นต์สามารถนำไฟล์คอนฟิกูเรชัน XML ไปไว้บน LDAP โดยใช้อ็อพชัน -u ของคำสั่ง **aixpertldap** ไฟล์คอนฟิกูเรชันเหล่านี้ ต้องอัพเดตแบบแมนวล

หมายเหตุ: คุณลักษณะ นี้จะขึ้นอยู่กับ LDAP โมเดลการไว้วางใจที่มีอยู่ ผู้ใช้ที่มีสิทธิพิเศษ ในการเขียนใน LDAP สามารถแก้ไขข้อมูลที่อัพโหลดโดยผู้ใช้ของเครื่องอื่นได้ เช่นเดียวกัน ถ้าไคลเอ็นต์ LDAP มีจุดอ่อนด้านความปลอดภัย ดังนั้นสิ่งนี้สามารถ นำมาใช้เพื่ออ่าน และเข้าใจในสถานะความปลอดภัยของไคลเอ็นต์ LDAP อื่นๆ โดยการอ่านไฟล์คอนฟิกูเรชัน AIX Security Expert XML ที่เชื่อมโยงกับไคลเอ็นต์นั้น

ตัวอย่าง ไฟล์ appliedaixpert.xml สามารถถูกบันทึก บนเซิร์ฟเวอร์ LDAP ภายใต้ชื่อ **BranchOfficeSecurityProfile** หรือ ไฟล์ appliedaixpert.xml ที่กำหนดคอนฟิกต่างกันอาจ ถูกบันทึกภายใต้ชื่อ **InternetDirectAttachedSystemsProfile** เนื่องจากระบบ อื่นๆ ที่มีการเชื่อมต่อกับ LDAP ถูกกำหนดคอนฟิกด้วย AIX Security Expert โปรไฟล์การรักษาความปลอดภัย เหล่านี้จะถูกแสดงเป็นอ็อพชันเมนูโดยอัตโนมัติ ซึ่ง อนุญาตให้ผู้ดูแลระบบสามารถเลือกโปรไฟล์การรักษาความปลอดภัยซึ่ง เหมาะสม กับสภาวะแวดล้อมของระบบตนดีที่สุดภายในแนวทางนโยบายด้านความปลอดภัยขององค์กร

จากนั้น AIX Security Expert จะถูกใช้ เพื่อรักษาความปลอดภัยของระบบ รายการทั้งหมดของค่าติดตั้งความปลอดภัยที่นำใช้บนระบบจะถูกบันทึกไว้ในไฟล์ /etc/security/aixpert/core/appliedaixpert.xml ไฟล์นี้ใช้เป็นนโยบายด้านความปลอดภัยสำหรับระบบนี้ นโยบายด้านความปลอดภัยถูกเปรียบเทียบ เมื่อมีการใช้อ็อพชัน AIX Security Expert Check Security นโยบายด้านความปลอดภัยนี้ยังสามารถทำสำเนาและนำไปใช้ กับระบบอื่นๆ ซึ่งช่วยให้เกิดความสอดคล้องกันด้านความปลอดภัยของระบบทั่วทั้ง สภาวะแวดล้อมด้านไอทีของคุณ มีสองวิธีในการทำสำเนานโยบายด้านความปลอดภัยไปยังระบบอื่นๆ ด้วยตนเอง หรือผ่าน LDAP

การทำสำเนานโยบายการรักษาความปลอดภัย AIX Security Expert

คุณสามารถใช้ AIX Security Expert เพื่อทำสำเนา นโยบายการรักษาความปลอดภัยจากระบบหนึ่งไปอีกระบบหนึ่ง

คุณสามารถรัน AIX Security Expert บน ระบบหนึ่งและใช้นโยบายการรักษาความปลอดภัยเดียวกันนั้นบนอีกระบบหนึ่ง ตัวอย่าง บ๊อบต้องการใช้ AIX Security Expert บนระบบ AIX ทั้งหกระบบของเขา เขาใช้ ค่าติดตั้งการรักษาความปลอดภัยบนระบบหนึ่ง (Alpha) ที่มีการรักษาความปลอดภัย High, Medium, Low, Advanced หรือ AIX Standard Settings เขาทดสอบระบบนี้เพื่อดูปัญหาความเข้ากันได้ภายใน สภาวะแวดล้อมของเขา ถ้าเขาพอใจกับค่าติดตั้งเหล่านี้ เขาสามารถนำ ค่าติดตั้งเดียวกันนี้ไปใช้บน AIX ระบบอื่นๆ ได้โดยใช้ชื่อ เขา ทำสำเนาค่าติดตั้งจากระบบ Alpha ไปยังระบบที่เขาต้องการ ใช้ค่าติดตั้งการรักษาความปลอดภัยเดียวกันโดยการทำสำเนาไฟล์ /etc/security/aixpert/core/appliedaixpert.xml จาก Alpha ไปยังระบบอื่น

หมายเหตุ: อย่าทำสำเนาไฟล์นี้ไปยัง ไดเร็กทอรีและชื่อไฟล์เดียวกันบนระบบอื่น เนื่องจากคำสั่ง **aixpert** จะเขียนทับ /etc/security/aixpert/core/appliedaixpert.xml ขณะ ที่นำใช้นโยบายการรักษาความปลอดภัย

แต่ให้ ทำสำเนานโยบายการรักษาความปลอดภัยของ Alpha ไปยังไดเร็กทอรี /etc/security/aixpert/custom/ แทน วิธีนี้ช่วยให้ระบบอื่นสามารถดูและนำนโยบายการรักษาความปลอดภัยของ Alpha ไปใช้ ผ่าน GUI การจัดการระบบ AIX Security Expert หรือใช้โดยตรงโดยคำสั่ง **aixpert**

ตัวอย่าง ถ้านโยบายการรักษาความปลอดภัย appliedaixpert.xml ของ Alpha ถูกนำไปไว้บนระบบอื่นเป็น /etc/security/aixpert/custom/*AlphaPolicy* ดังนั้นคำสั่ง aixpert -f /etc/security/aixpert/custom/*AlphaPolicy* จะ ใช้นโยบายการรักษาความปลอดภัยนี้ทันทีและระบบนี้จะมี การตั้งค่าการรักษาความปลอดภัยเหมือนกับเครื่อง Alpha นอกจากนั้น เมื่อ นโยบายการรักษาความปลอดภัยของ Alpha อยู่ในไดเร็กทอรีนี้ จะสามารถเห็นนโยบายได้และสามารถ ถูกนำไปใช้ยังระบบอื่นๆ ผ่านคอนโซลการจัดการระบบทาง พาธของ Aix Security Expert -> Overview and Tasks -> Customized Options -> AlphaPolicy

นโยบายการรักษาความปลอดภัยที่กำหนดเองได้ด้วยกฎ AIX Security Expert XML ที่ผู้ใช้กำหนดเอง

คุณสามารถใช้ไฟล์ XML เพื่อตั้งค่านโยบายการรักษาความปลอดภัยเฉพาะ

AIX Security Expert จดจำไฟล์ XML เหล่านี้แบบไดนามิก ไฟล์นโยบาย XMLsecurity ที่กำหนดเอง ใดๆ ที่สร้างควรอยู่ในไดเร็กทอรี /etc/security/aixpert/custom/ ที่มี ไฟล์อธิบาย ดังนั้น เมื่อ AIX Security Expert ถูกเข้าถึง ผ่านส่วนการติดต่อแบบกราฟิกคอนโซล ชุดคุณลักษณะ XML แบบกราฟิก ใน aixpert DTD จะได้รับการยอมรับทั้งหมด

DTD เป็นดังนี้:

```
<?xml version='1.0'?>

<!--START-->

<!ELEMENT AIXPertSecurityHardening (AIXPertEntry+)>

<!-- AIXPertEntry ควรมีหนึ่ง instance ขององค์ประกอบต่อไปนี้เท่านั้น -->

<!ELEMENT AIXPertEntry (AIXPertRuleType,
  AIXPertDescription, AIXPertPrereqList, AIXPertCommand,
  AIXPertArgs,AIXPertGroup)>

<!-- ชื่อของ AIXPertEntry ควรเป็นชื่อเฉพาะ -->

<!ATTLIST AIXPertEntry
 name ID #REQUIRED
 function CDATA ""
>

<!ELEMENT AIXPertRuleType EMPTY>
<!ATTLIST AIXPertRuleType    type (LLS|MLS|HLS|DLS|SCBPS|Prereq) "DLS">
<!ELEMENT AIXPertDescription (#PCDATA)>
<!ELEMENT AIXPertPrereqList (#PCDATA)>
<!ELEMENT AIXPertCommand (#PCDATA)>
<!ELEMENT AIXPertArgs (#PCDATA)*>
<!ELEMENT AIXPertGroup (#PCDATA)*>
```

ชื่อ AIXPertEntry เป็นชื่อเฉพาะภายใน XMLfile ชื่อ นี้จะเป็นชื่อของปุ่มกราฟิกที่เลือกได้เมื่อดูไฟล์นี้ ผ่านคอนโซลระบบทางพาธ Aix Security Expert -> Overview and Tasks -> Customized Options -> <xml file=""></xml>

<!ELEMENT AIXPertRuleType EMPTY>
ไฟล์ XML นี้ควรระบุเป็นกำหนดเอง

<!ATTLIST AIXPertRuleType type (LLS|MLS|HLS|DLS|SCBPS|Prereq|Custom) "DLS"
ไฟล์ XML นี้ควรระบุเป็นกำหนดเอง

<!ELEMENT AIXPertDescription (#PCDATA)>
เมื่อดูผ่านส่วนการติดต่อแบบกราฟิกที่กล่าวด้านบน ข้อความอธิบาย จะแสดงบนหน้าต่างป๊อปอัพเมื่อวางเมาส์บนปุ่มนี้

<!ELEMENT AIXPertPrereqList (#PCDATA)>
คุณสามารถเลือกกฎที่เป็นเงื่อนไขสำหรับกฎนี้ กฎ ที่เป็นเงื่อนไขต้องคืนค่า 0 ก่อน aixpert จึงจะประยุกต์ใช้ กฎนี้ ถ้าไฟล์ XML นี้ดูผ่านส่วนการติดต่อแบบกราฟิก กฎนี้ จะไม่สามารถใช้ได้ ถ้าไม่ตรงตามกฎที่เป็นเงื่อนไข ถ้า คุณกำลังสร้างกฎที่เป็นเงื่อนไข AIXPertRuleType ต้องเป็น 'Prereq'

ฟิลด์ AIXPertDescription ของกฎที่เป็นเงื่อนไข ควรอธิบายสิ่งที่ควรทำเพื่อให้ตรงตามกฎที่เป็นเงื่อนไข ถ้า กฎ Custom ไม่สามารถเลือกได้ เนื่องจากไม่ตรงกับกฎที่เป็นเงื่อนไข กฎใดกฎหนึ่ง ดังนั้นผู้ใช้จะพบหน้าต่างป๊อปอัพที่อธิบาย กฎที่เป็นเงื่อนไข ซึ่งอธิบายสิ่งที่ผู้ใช้ต้องทำเพื่อดำเนิน เงื่อนไขที่จำเป็นต้องมีให้ถูกต้อง

<!ELEMENT AIXPertCommand (#PCDATA)>

องค์ประกอบนี้ต้องเป็นพาธและคำสั่งแบบเต็ม ซึ่ง aixpert จะ ทำงานสำหรับกฎการรักษาความปลอดภัยนี้ เช่น `/usr/bin/ls`

<!ELEMENT AIXPertArgs (#PCDATA)*>

องค์ประกอบนี้ต้องมีอากิวเมนต์ใดๆ สำหรับคำสั่งด้านบน เช่น -l

<!ELEMENT AIXPertGroup (#PCDATA)*>

คุณสามารถจัดกลุ่มชุดของกฎ aixpert เมื่อแสดง ผ่านส่วนการติดต่อแบบกราฟิก ตัวอย่าง ชุดกฎทั่วไปอาจ ระบุชื่อ AIXPertGroup ทั้งหมดเป็น "Network Security"

การกวดขันการตรวจหารหัสผ่านที่คาดเดาง่าย

คุณลักษณะ AIX นี้ จะตรวจหารหัสผ่านที่คาดเดาง่ายเมื่อมีการเปลี่ยนรหัสผ่าน ถ้าเลือกใช้อ็อพชันนี้ โดย AIX Security Expert การตรวจสอบ รหัสผ่านเพิ่มเติมนี้จะถูกดำเนินการเมื่อผู้ใช้เลือกหรือเปลี่ยนรหัสผ่าน การตรวจสอบนี้ช่วยป้องกันการใช้คำในพจนานุกรมภาษาอังกฤษและชื่อของ ประชากรสหรัฐที่ใช้กันมากที่สุด 1000 ชื่อตามรายงาน US Census ล่าสุด

อ็อบเจ็กต์คอนโทรล COBIT ที่สนับสนุนโดย AIX Security Expert

AIX Security Expert สนับสนุน ระบบ SOB-COBIT Best Practices Security นอกเหนือจาก High, Medium, Low, ค่าติดตั้ง AIX Default และ Advanced Security

สภาคองเกรสของสหรัฐออกพระราชบัญญัติ 'Sarbanes-Oxley Act of 2002' เพื่อปกป้องนักลงทุนโดยการช่วยปรับปรุงความถูกต้องและความเชื่อถือได้ของ ข้อมูลการเงินที่องค์กรเปิดเผย คุณลักษณะวัตถุประสงค์การควบคุม COBIT จะช่วยระบบในการตั้งค่า ดูแลรักษา และตรวจสอบระบบไอทีของตนเพื่อให้เป็นไปตามกฎหมายนี้ SOX Configuration Assistant เข้าถึงได้ผ่านบรรทัดรับคำสั่ง aixpert คุณลักษณะนี้ยังช่วย SOX ส่วนที่ 404 ของ Sarbanes-Oxley Act, แต่ AIX Security Expert SOX Configuration Assistant ใช้ค่าติดตั้งความปลอดภัย ที่เชื่อมโยงกับแนวปฏิบัติที่เหมาะสม COBIT สำหรับ SOX ส่วนที่ 404, Internal Controls นอกจากนั้น AIX Security Expert ยังมีคุณลักษณะการตรวจสอบ SOX ซึ่งจะรายการให้ผู้ตรวจสอบทราบว่าขณะนี้ระบบถูกตั้งค่าให้เป็นไปตามแนวทางนี้แล้ว คุณลักษณะยังให้มี การตั้งค่าระบบโดยอัตโนมัติเพื่อช่วยในการกำกับดูแล IT SOX และในกระบวนการตรวจสอบให้เป็นอัตโนมัติ

เนื่องจาก SOX ไม่มีนำแนะนำเกี่ยวกับวิธีที่ IT ต้องปฏิเบัติตาม section 404 อุตสาหกรรม IT เน้นที่การกำกับดูแลที่มีอยู่แล้ว ซึ่งกำหนดรายละเอียด โดย www.isaca.org/ ที่เฉพาะเจาะจงยิ่งกว่านั้น คือการกำกับดูแล IT ครอบคลุมโดย Control Objectives for Information and related Technology (COBIT)

AIX Security Expert สนับสนุน วัตถุประสงค์การควบคุมต่อไปนี้:

- การบังคับใช้นโยบายรหัสผ่าน
- รายการการฝ่าฝืนและกิจกรรมด้านการรักษาความปลอดภัย
- การป้องกัน การตรวจหา และการแก้ไขซอฟต์แวร์ที่เป็นอันตราย รวมถึงซอฟต์แวร์ ที่ไม่ได้รับอนุญาต
- สถาปัตยกรรมไฟร์วอลล์และการเชื่อมต่อกับพับลิกเน็ตเวิร์ก

AIX Security Expert ไม่ ได้สนับสนุนแอ็ตทริบิวต์ทั้งหมดที่ระบุภายใต้วัตถุประสงค์การควบคุมแต่ละอย่าง แอ็ตทริบิวต์ที่สนับสนุนและวัตถุประสงค์การควบคุมที่เกี่ยวข้อง จะสรุปไว้ในตารางต่อไปนี้:

การบังคับใช้นโยบายรหัสผ่าน

คำอธิบาย	ค่าติดตั้งการรักษาความปลอดภัย
อายุรหัสผ่านสูงสุด	maxage=13
ประวัติการบังคับใช้รหัสผ่าน	histsize=20
อายุรหัสผ่านต่ำสุด	minage=1
ความยาวรหัสผ่านต่ำสุด	minlen=8
ต้องมีอักขระอย่างน้อย 6 ตัว	Minalpha=6
ความคล้ายคลึงกับรหัสผ่านเก่า	mindiff=4
จำนวนวันที่เตือนการหมดอายุรหัสผ่าน	pwdwarntime=14

รายงานการฝ่าฝืนและกิจกรรมด้านความปลอดภัย

คำอธิบาย	ค่าติดตั้งการรักษาความปลอดภัย	หมายเหตุ
การตรวจสอบถูกเปิดใช้งาน	ใช่	
ไม่มีการล็อกอินเป็น root โดยตรง	ใช่	
เปิดใช้การตรวจสอบการเพิ่มสิทธิพิเศษ	ใช่	AIXpert ยอมรับเหตุการณ์การตรวจสอบ USER_SU โปรดตรวจสอบให้แน่ใจ ว่าเปิดใช้เหตุการณ์นี้

การตรวจหาและแก้ไขซอฟต์แวร์ที่เป็นอันตราย

AIX Security Expert ยอมรับ คุณลักษณะการทำงานของซอฟต์แวร์ที่ได้รับความไว้วางใจ AIX เพื่อให้แน่ใจว่าซอฟต์แวร์ไม่ถูกเปลี่ยนโดย บุคคลใด คำสั่ง **trustchk** จะตรวจสอบความสอดคล้องกัน ของอ็อบเจ็กต์ที่ลงทะเบียนในฐานข้อมูล Trusted Software

การตั้งค่าไฟร์วอลล์

AIX Security Expert เปิดใช้ IPSec และเปิดใช้กฎตัวกรองเพื่อหลีกเลี่ยงการสแกนพอร์ต พอร์ตที่ เลี่ยง จะแสดงในตารางต่อไปนี้:

เซอร์วิส	คำอธิบาย
Tcp/11, udp/11	Systat
Tcp/13, udp/13	Daytime
(RFC 867) Tcp/19, udp/19	Character Generator

เซอร์วิส	คำอธิบาย
Tcp/25	Simple Mail Transfer (SMTP)
Tcp/43, udp/43	Who Is (nickname)
Tcp/63, udp/63	Whois++
Tcp/67, udp/67	Bootstrap protocol server (bootps)
Tcp/68, udp/68	Bootstrap protocol client (bootpc)
Tcp/69, udp/69	Trivial file transfer
(tftp) Tcp/79, udp/79	Finger
Tcp/87	Private Terminal Link
Tcp/110	Post office protocol – version 3 (POP3)
Udp/111	SUN Remote Procedure Call
Tcp/113	Authentication Service (auth)
Udp/123	Network Time Protocol
Udp/161	SNMP
Udp/162	SNMPTRAP
Tcp/194	Internet Relay chat Protocol
Tcp/443	http protocol over TLS/SSL
Tcp/511	PassGo
Tcp/514	Cmd (shell)
Tcp/520	Extended file name server (efs)
Tcp/540	Uucpd (uucp)
Tcp/546	DHCPv6 Client
Tcp/547	DHCPv6 Server
Tcp/555	Dsf
tcp/559	TEEDTAP
tcp/593	HTTP RPC Ep Map
udp/635	RLS Dbase
tcp/666	Mdqs
tcp/777	Multiling HTTP
tcp/901	SNMPNSMERES

เซอร์วิส	คำอธิบาย
tcp/902	IDEAFARM-CHAT
tcp/903	IDEAFARM-CATCH
tcp/1024	ถูกสำรอง

การนำใช้วัตถุประสงค์การควบคุม COBIT โดยใช้ AIX Security Expert

คุณสามารถใช้คำสั่ง **aixpert -l s** เพื่อนำใช้ระดับ SCBPS กับระบบ บันทึกการตรวจสอบสำหรับคำสั่งนี้ สามารถสร้างโดยการปรับแต่งที่เหตุการณ์ **AIXpert_apply** ความล้มเหลวใดๆ (ไม่ว่าเป็นความล้มเหลวสำหรับสิ่งที่จำเป็นต้องมีหรือความล้มเหลวในการนำไปใช้) ถูกรายงานไปยัง **stderr** และ ระบบย่อยการตรวจสอบถ้าถูกเปิดใช้งาน

การตรวจสอบการปฏิบัติตาม SOX-COBIT การตรวจ และคุณลักษณะก่อนการตรวจ

คุณสามารถใช้คำสั่ง **aixpert -c -l s** เพื่อตรวจสอบการปฏิบัติตาม SOX-COBIT ของระบบ AIX Security Expert ตรวจสอบเฉพาะการปฏิบัติตามวัตถุประสงค์การควบคุมที่สนับสนุนเท่านั้น การฝ่าฝืนใดๆ ที่พบ ระหว่างการตรวจสอบต้องถูกรายงานโดยค่าดีฟอลต์ การฝ่าฝืนใดๆ จะถูกส่ง ไปยัง **stderr**

คุณยังสามารถใช้คำสั่งเดียวกัน (**aixpert -c -l s**) เพื่อสร้างรายงานตรวจสอบการปฏิบัติ SOX-COBIT ในการสร้างรายงานการตรวจสอบ ให้ตั้งค่าและเปิดใช้งานระบบย่อยการตรวจสอบ ทำให้แน่ใจว่าเปิดใช้งานเหตุการณ์การตรวจสอบ **AIXpert_check** หลัง การตั้งค่าระบบย่อยการตรวจสอบแล้ว ให้รันคำสั่ง **aixpert -c -l s** อีกครั้ง คำสั่งจะสร้างบันทึกการตรวจสอบในทุกครั้ง ที่วัตถุประสงค์การควบคุมล้มเหลว สถานะ **Status** ของบันทึกการตรวจสอบ จะทำเครื่องหมายเป็น **failed** บันทึกการทำงานยังมีเหตุผล ที่เกิดความล้มเหลว ซึ่งสามารดูได้โดยใช้อ็อพชัน **-v** ของคำสั่ง **auditpr**

การเพิ่มอ็อพชัน **-p** ในคำสั่ง **aixpert -c -l s** ยังรวมวัตถุประสงค์การควบคุมที่สำเร็จ ไว้ในรายงานการตรวจสอบด้วย รายการบันทึกเหล่านั้นมี **Ok** ใน ฟิลด์สถานะ

คำสั่ง **aixpert -c -l s -p** สามารถใช้สร้างรายงานการตรวจสอบการปฏิบัติตาม SOX-COBIT โดยละเอียด

ไม่ว่าระบุอ็อพชัน **-p** หรือไม่ ก็จะมี เร็กคอร์ดสรุป เร็กคอร์ดสรุปประกอบด้วยข้อมูล เกี่ยวกับจำนวนกฎที่ถูกประมวลผล จำนวนกฎที่ล้มเหลว (instances ของการไม่ปฏิบัติตามที่ตรวจพบ) และระดับความปลอดภัยที่ระบบ จะถูกตรวจสอบ (ใน instance นี้ นี้อาจเป็น SCBPS)

กลุ่ม AIX Security Expert Password Policy Rules

AIX Security Expert จัดให้มี กฎเฉพาะสำหรับนโยบายรหัสผ่าน

นโยบายรหัสผ่านที่คาดเดาได้ยากเป็นกลวิธีการป้องกันวิธีหนึ่งเพื่อรักษาความปลอดภัย ระบบให้สำเร็จ นโยบายรหัสผ่านช่วยให้แน่ใจว่ารหัสผ่านนั้นคาดเดา ได้ยาก (รหัสผ่านมีการผสมกันของอักขระแบบตัวอักษรผสมตัวเลข ตัวเลข และอักขระพิเศษอย่างเหมาะสม) โดยกำหนดให้มีหมดอายุเป็นระยะ และไม่สามารถนำกลับมาใช้ ได้อีกหลังจากหมดอายุแล้ว ตารางต่อไปนี้แสดงกฎสำหรับนโยบายรหัสผ่าน สำหรับการตั้งค่าการรักษาความปลอดภัยแต่ละค่า

ตารางที่ 19. AIX Security Expert Password Policy Rules

ชื่อปุ่มการดำเนินการ	นิยาม	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
จำนวนอักขระต่ำสุด	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ **mindiff** ของ /etc/security/user ซึ่งระบุจำนวนอักขระ ต่ำสุดที่ต้องการสำหรับรหัสผ่านใหม่ที่ไม่มีใน รหัสผ่านเก่า	**High Level Security** 4 **Medium Level Security** 3 **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่จำกัด	ใช่
อายุต่ำสุดสำหรับรหัสผ่าน	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ **minage** ของ /etc/security/user ซึ่งระบุจำนวนสัปดาห์ ต่ำสุดก่อนที่รหัสผ่านจะสามารถเปลี่ยนได้	**High Level Security** 1 **Medium Level Security** 4 **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่จำกัด	ใช่
อายุสูงสุดสำหรับรหัสผ่าน	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ **maxage** ของ /etc/security/user ซึ่งระบุจำนวนสัปดาห์ สูงสุดก่อนที่รหัสผ่านจะสามารถเปลี่ยนได้	**High Level Security** 13 **Medium Level Security** 13 **Low Level Security** 52 **AIX Standard Settings** ไม่จำกัด	ใช่
ความยาวต่ำสุดสำหรับรหัสผ่าน	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ **minlen** ของ /etc/security/user ซึ่งระบุความยาว ต่ำสุดของรหัสผ่าน	**High Level Security** 8 **Medium Level Security** 8 **Low Level Security** 8 **AIX Standard Settings** ไม่จำกัด	ใช่

ตารางที่ 19. AIX Security Expert Password Policy Rules (ต่อ)

ชื่อปุ่มการดำเนินการ	นิยาม	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
จำนวนอักขระแบบตัวอักษรต่ำสุด	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ `minalpha` ของ `/etc/security/user` ซึ่งระบุจำนวน อักขระแบบตัวอักษรต่ำสุดในรหัสผ่าน	**High Level Security** 2 **Medium Level Security** 2 **Low Level Security** 2 **AIX Standard Settings** ไม่จำกัด	ใช่
เวลาที่จะตั้งรหัสผ่านใหม่	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ `histexpire` ของ `/etc/security/user` ซึ่งระบุจำนวนสัปดาห์ ต่ำสุดก่อนที่รหัสผ่านจะสามารถตั้งค่าใหม่ได้	**High Level Security** 13 **Medium Level Security** 13 **Low Level Security** 26 **AIX Standard Settings** ไม่จำกัด	ใช่
จำนวนครั้งสูงสุดที่หนึ่งอักขระสามารถแสดงในรหัสผ่าน	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ `maxrepeats` ของ `/etc/security/user` ซึ่งระบุจำนวนครั้ง สูงสุดที่หนึ่งอักขระสามารถแสดงในรหัสผ่าน	**High Level Security** 2 **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** 8	ใช่
เวลาการนำรหัสผ่านไปใช้ใหม่	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ **histsize** ของ `/etc/security/user` ซึ่งระบุจำนวนรหัสผ่าน ที่ใช้ก่อนหน้านี้ที่ผู้ใช้ไม่สามารถนำไปใช้ใหม่	**High Level Security** 20 **Medium Level Security** 4 **Low Level Security** 4 **AIX Standard Settings** ไม่จำกัด	ใช่

ตารางที่ 19. AIX Security Expert Password Policy Rules (ต่อ)

ชื่อปุ่มการดำเนินการ	นิยาม	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
เวลาที่จะเปลี่ยนรหัสผ่านหลังหมดอายุ	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ **maxexpired** ของ /etc/security/user ซึ่งระบุจำนวนสัปดาห์ สูงสุดหลัง **maxage** ที่ผู้ใช้ยังสามารถเปลี่ยนรหัสผ่านซึ่งหมดอายุแล้วได้	**High Level Security** 2 **Medium Level Security** 4 **Low Level Security** 8 **AIX Standard Settings** -1	ใช่
จำนวนต่ำสุดของอักขระที่มิใช่แบบตัวอักษร	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ **minother** ของ /etc/security/user ซึ่งระบุจำนวนอักขระที่มิใช่ แบบตัวอักษรต่ำสุดที่มีได้ในรหัสผ่าน	**High Level Security** 2 **Medium Level Security** 2 **Low Level Security** 2 **AIX Standard Settings** ไม่จำกัด	ใช่
เวลาเตือนรหัสผ่านหมดอายุ	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ **pwdwarntime** ของ /etc/security/user ซึ่งระบุจำนวน วันก่อนที่ระบบจะออกการเตือนให้ทราบว่าต้องทำการเปลี่ยน รหัสผ่าน	**High Level Security** 5 **Medium Level Security** 14 **Low Level Security** 5 **AIX Standard Settings** ไม่จำกัด	ใช่

กลุ่มนิยาม AIX Security Expert User Group System and Password

AIX Security Expert ดำเนิน การที่เฉพาะเจาะจงสำหรับนิยามผู้ใช้ กลุ่ม และรหัสผ่าน

ตารางที่ 20. AIX Security Expert User Group System and Password Definitions

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ตรวจสอบนิยามกลุ่ม	ตรวจสอบความถูกต้องของนิยามกลุ่ม รันคำสั่งต่อไปนี้เพื่อแก้ไขและรายงานข้อผิดพลาด: `% grpck -y ALL`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ใช่ **AIX Standard Settings** ไม่มีผล	ไม่ใช่
การอัพเดต TCB	ใช้คำสั่ง **tcbck** เพื่อ ตรวจสอบและอัพเดต TCB รันคำสั่งต่อไปนี้: `% tcbck -y ALL` **หมายเหตุ:** ถ้าขำเป็นต้องใช้ TCB ในระบบของคุณ กฎนี้จะล้มเหลว ถ้าไม่เปิดใช้งาน TCB ไว้ กฎที่เป็นเงื่อนไข (prereqtcb) จะล้มเหลวพร้อมคำเตือน **เงื่อนไข**: TCB ต้องถูกเลือกไว้ เมื่อระบบถูกติดตั้งไว้	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ใช่ **AIX Standard Settings** ใช่	ไม่ใช่

ตรวจสอบนิยามไฟล์	ใช้คำสั่ง **sysck** เพื่อ ตรวจสอบและแก้ไขไฟล์ของ `/etc/objrepos/inventory`:
	`% sysck -i -f \`
	`/etc/security/sysck.cfg.rte`
		High Level Security
		ใช่
		Medium Level Security
		ใช่
		Low Level Security
		ใช่
		AIX Standard Settings
		ไม่มีผล
			ไม่ใช่

ตรวจสอบนิยามรหัสผ่าน	ตรวจสอบความถูกต้องของนิยามรหัสผ่าน รันคำสั่งต่อไปนี้เพื่อแก้ไขและรายงานข้อผิดพลาด: `% pwdck -y ALL`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ใช่ **AIX Standard Settings** ไม่มีผล	ไม่ใช่

ตารางที่ 20. AIX Security Expert User Group System and Password Definitions (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ตรวจสอบนิยามผู้ใช้	ตรวจสอบความถูกต้องของนิยามผู้ใช้ รัน คำสั่งต่อไปนี้เพื่อแก้ไขและรายงานข้อผิดพลาด: `% usrck -y ALL`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ใช่ **AIX Standard Settings** ไม่มีผล	ไม่ใช่

กลุ่ม AIX Security Expert Login Policy Recommendations

AIX Security Expert มี ค่าติดตั้งเฉพาะเจาะจงสำหรับนโยบายการล็อกอิน

หมายเหตุ: เพื่อให้แน่ใจว่ามีการทำงานที่ดีขึ้นของกิจกรรมที่เกี่ยวข้องกับการรักษาความปลอดภัย ที่ดำเนินการโดย root อันดับแรกขอแนะนำให้ผู้ใช้ล็อกอิน โดยใช้ ID ผู้ใช้ปกติ จากนั้นรัน *คำสั่ง* **su** เพื่อ รันคำสั่งในฐานะ root แทนการล็อกอินเป็น root จากนั้นระบบสามารถ เชื่อมโยงผู้ใช้ต่างๆ เข้ากับกิจกรรมที่ดำเนินการโดยใช้บัญชีผู้ใช้ root เมื่อผู้ใช้หลายคนทราบและใช้รหัสผ่าน root

ตารางที่ 21. AIX Security Expert Login Policy Recommendations

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ช่วงเวลาระหว่างการล็อกอินที่ไม่สำเร็จ	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ `logininterval` ของ `/etc/security/login.cfg` ซึ่งระบุ ช่วงเวลา (เป็นวินาที) ระหว่างการพยายามล็อกอินที่ไม่สำเร็จ สำหรับพอร์ตที่ต้องเกิดขึ้นก่อนพอร์ตถูกปิดใช้งาน ตัวอย่าง ถ้า `logininterval` ถูก ตั้งค่าเป็น 60 และ `logindisable` ถูกตั้งค่าเป็น 4 บัญชีผู้ใช้ จะถูกปิดใช้งานถ้ามีจำนวนการพยายามล็อกอินไม่สำเร็จสี่ครั้งภายใน 1 นาที	**High Level Security** 300 **Medium Level Security** 60 **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่จำกัด	ใช่
จำนวนการพยายามล็อกอินก่อนทำการล็อก บัญชีผู้ใช้	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ `loginretries` ของ `/etc/security/user` ซึ่งระบุจำนวน ครั้งการพยายามล็อกอินติดต่อกันต่อหนึ่งบัญชีผู้ใช้ก่อนที่บัญชีผู้ใช้จะถูกปิดใช้งาน อย่าตั้งค่าที่ root	**High Level Security** 3 **Medium Level Security** 4 **Low Level Security** 5 **AIX Standard Settings** ไม่จำกัด	ใช่

ตารางที่ 21. AIX Security Expert Login Policy Recommendations (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ล็อกอิน root รีโมต	เปลี่ยนค่าของแอ็ตทริบิวต์ `rlogin` ของ `/etc/security/user` ซึ่งระบุว่า อนุญาตให้ล็อกอินแบบรีโมตบนระบบหรือไม่สำหรับบัญชีผู้ใช้ root	**High Level Security** เท็จ **Medium Level Security** เท็จ **Low Level Security** ไม่มีผล **AIX Standard Settings** จริง	ใช่
เปิดใช้การล็อกอินอีกครั้งหลังการล็อก	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ `loginreenable` ของ `/etc/security/login.cfg` ซึ่งระบุ ช่วงเวลา (เป็นวินาที) หลังจากพอร์ตถูกปลดล็อกหลังจาก พอร์ตถูกปิดใช้งานโดย `logindisable`	**High Level Security** 360 **Medium Level Security** 30 **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่จำกัด	ใช่
ปิดใช้การล็อกอินหลังการพยายามล็อกอินไม่สำเร็จ	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ `logindisable` ของ `/etc/security/login.cfg` ซึ่งระบุ จำนวนครั้งการพยายามล็อกอินที่ไม่สำเร็จบนพอร์ตก่อนพอร์ต ถูกล็อก	**High Level Security** 10 **Medium Level Security** 10 **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่จำกัด	ใช่
หมดเวลาใช้งานล็อกอิน	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ `logintimeout` ของ `/etc/security/login.cfg` ซึ่งระบุ ช่วงเวลาที่อนุญาตให้พิมพ์รหัสผ่าน	**High Level Security** 30 **Medium Level Security** 60 **Low Level Security** 60 **AIX Standard Settings** 60	ใช่

ตารางที่ 21. AIX Security Expert Login Policy Recommendations (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
เวลาหน่วงระหว่างการล็อกอินที่ไม่สำเร็จ	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ `logindelay` ของ `/etc/security/login.cfg` ซึ่งระบุ การหน่วง (เป็นวินาที) ระหว่างการล็อกอินที่ไม่สำเร็จ ระยะการหน่วงเวลา เพิ่มจะถูกเพิ่มหลังการล็อกอินที่ล้มเหลวแต่ละครั้ง ตัวอย่าง ถ้า `logindelay` ถูกตั้งค่าเป็น 5 เทอร์มินัลจะรอห้าวินาทีหลังจากล็อกอินครั้งแรก ที่ล้มเหลวจนกระทั่งมีการร้องขอครั้งถัดไป หลังการล็อกอินล้มเหลวครั้งที่สอง เทอร์มินัล จะรอ 10 วินาที (2*5) และหลังการล็อกอินล้มเหลวครั้งที่สาม เทอร์มินัล จะรอ 15 วินาที (3*5)	**High Level Security** 10 **Medium Level Security** 4 **Low Level Security** 5 **AIX Standard Settings** ไม่จำกัด	ใช่
ล็อกอินโลคัล	เปลี่ยนค่าของแอ็ตทริบิวต์ `login` ของ `/etc/security/user` ซึ่งระบุว่าล็อกอินที่คอนโซลอนุญาตให้ใช้บัญชีผู้ใช้ root บนระบบหรือไม่	**High Level Security** เท็จ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** จริง	ใช่

กลุ่ม AIX Security Expert Audit Policy Recommendations

AIX Security Expert มี ค่าติดตั้งนโยบายการตรวจสอบเฉพาะ

เช่นเดียวกับค่าติดตั้งความปลอดภัยอื่นๆ การตรวจสอบ bin จำเป็นต้องตรวจสอบ ให้ตามกฎการวิเคราะห์ (สิ่งที่จำเป็นต้องมี) เช่นกันก่อนที่จะบังคับใช้กฎการตรวจสอบ ใดๆ สำหรับ High, Medium หรือ Low Level Security ต้องเป็นไปตามกฎการวิเคราะห์ ต่อไปนี้จึงจะทำการตรวจสอบ bin:

1. กฎที่เป็นเงื่อนไขที่จะตรวจสอบต้องตรวจดูว่าการตรวจสอบนั้น ไม่ได้ทำงานอยู่ขณะนี้ ถ้าการตรวจสอบกำลังทำงาน ดังนั้นการตรวจสอบ ที่ตั้งค่าก่อนหน้านี้ และ AIX Security Expert ต้องไม่ เปลี่ยนการตั้งค่าการตรวจสอบและโพรซีเดอร์ที่มีอยู่
2. ต้องมีพื้นที่ว่างอย่างน้อย 100 เมกะไบต์ในกลุ่ม วอลุ่มที่จะแปรผันตาม หรือระบบไฟล์ `/audit` ต้องมีอยู่โดยมีขนาดอย่างน้อย 100 เมกะไบต์ในขณะนี้

ถ้าตรงตามเงื่อนไขที่ต้องมีด้านบน และอ็อพชันการตรวจสอบ ถูกเลือกภายใน AIX Security Expert ดังนั้น AIX Security Expert จะตั้งค่า และเปิดใช้งานการตรวจสอบบนระบบในแนวทางต่อไปนี้ ปุ่มการดำเนินการ AIX Security Expert **Enable binaudit** ตั้งค่านโยบายการตรวจสอบ การตรวจสอบต้องถูกเปิดใช้งาน บนระบบ

1. ระบบไฟล์ `/audit` JFS ต้องถูกสร้าง และใส่เข้าก่อนเริ่มการตรวจสอบ ระบบไฟล์ต้องมีขนาด อย่างน้อย 100 เมกะไบต์
2. การตรวจสอบต้องรันในโหมด bin ไฟล์ `/etc/security/audit/config` ต้องถูกตั้งค่าดังนี้:

```
start:
            binmode = on
            streammode = off
```

```
bin:
            trail = /audit/trail
            bin1 = /audit/bin1
            bin2 = /audit/bin2
            binsize = 10240
            cmds
= /etc/security/audit/bincmds
     .
     .
     etc
```

3. เพิ่มรายการการตรวจสอบสำหรับผู้ใช้ root และผู้ใช้ทั่วไปสำหรับ High, Medium, และ Low Level Security
4. การตรวจสอบต้องถูกเปิดใช้งานเมื่อบูตใหม่สำหรับ High, Medium และ Low Level Security
5. ผู้ใช้ใหม่ที่สร้างต้องมีการเปิดใช้งานการตรวจสอบสำหรับ High, Medium และ Low Level Security นี้สามารถทำได้โดยการเพิ่มรายการ `auditclasses` ใน stanza ผู้ใช้ในไฟล์ `/usr/lib/security/mkuser.default`
6. **cronjob** ต้องถูกเพิ่มเพื่อเลี่ยงการเติมค่าใน ระบบไฟล์ `/audit`

กฎการเลิกทำการตรวจสอบต้องปิดการตรวจสอบ และลบการเปิดใช้งานออก เมื่อบูตใหม่

ตารางต่อไปนี้แสดงรายการค่าที่ตั้งค่าโดย AIX Security Expert เพื่อ **Enable binaudit**:

ตารางที่ 22. ค่าที่ตั้งค่าโดย AIX Security Expert เพื่อ Enable binaudit

High Level Security	Medium Level Security	Low Level Security	AIX Standard Settings

เพิ่มรายการตรวจสอบต่อไปนี้สำหรับผู้ใช้ root และผู้ใช้ทั่วไป:
`Root:`
`General`
`Src`
`Mail`
`Cron`
`Tcpip`
`Ipsec`
`Lvm`
`User:`
`General`
`Src`
`Cron`
`Tcpip`
 	
เพิ่มรายการต่อไปนี้ใน stanza ผู้ใช้ของไฟล์ `/usr/lib/security/mkuser.default` สำหรับการเปิดใช้งานการตรวจสอบผู้ใช้ที่สร้างใหม่:
`auditclasses=general,SRC,\`
`cron,tcpip`
	เพิ่มรายการตรวจสอบต่อไปนี้สำหรับผู้ใช้ root และผู้ใช้ทั่วไป:
	`Root:`
	`General`
	`Src`
	`Tcpip`
	`User:`
	`General`
	`Tcpip`
	
	เพิ่มรายการต่อไปนี้ใน stanza ผู้ใช้ของไฟล์ `/usr/lib/security/mkuser.default` สำหรับการเปิดใช้งานการตรวจสอบผู้ใช้ที่สร้างใหม่:
	`auditclasses=general,`
	`tcpip`
		เพิ่มรายการตรวจสอบต่อไปนี้สำหรับผู้ใช้ root และผู้ใช้ทั่วไป:
		`Root:`
		`General`
		`Tcpip`
		`User:`
		`General`
		
		เพิ่มรายการต่อไปนี้ใน stanza ผู้ใช้ของไฟล์ `/usr/lib/security/mkuser.default` สำหรับการเปิดใช้งานการตรวจสอบผู้ใช้ที่สร้างใหม่:
		`auditclasses=general`
			ไฟล์ `/etc/security/audit/config` มี รายการต่อไปนี้:
			`default=login`
			
			การล๊อกอินคลาสการตรวจสอบถูกำหนดดังนี้:
			`login = USER_SU,`
			`USER_Login,`
			`USER_Logout,`
			`TERM_Logout,`
			`USER_Exit`
			
			หมายเหตุ: คุณลักษณะการตั้งค่ามาตรฐานปิดใช้งาน การตรวจสอบ

ตารางที่ 22. ค่าที่ตั้งค่าโดย AIX Security Expert เพื่อ Enable binaudit (ต่อ)

High Level Security	Medium Level Security	Low Level Security	AIX Standard Settings
เพิ่มรายการตรวจสอบต่อไปนี้สำหรับผู้ใช้ root และผู้ใช้ทั่วไป: **root:** general src mail cron tcpip ipsec lvm aixpert *User:* general src cron tcpip เพิ่มรายการต่อไปนี้ใน stanza ผู้ใช้ของไฟล์ /usr/lib/security/mkuser.default สำหรับการเปิดใช้งานการตรวจสอบผู้ใช้ที่สร้างใหม่: `auditclasses=general,SRC, cron,tcpip`	เพิ่มรายการตรวจสอบต่อไปนี้สำหรับผู้ใช้ root และผู้ใช้ทั่วไป: **root:** general src tcpip aixpert *User:* general tcpip เพิ่มรายการต่อไปนี้ใน stanza ผู้ใช้ของไฟล์ /usr/lib/security/mkuser.default สำหรับการเปิดใช้งานการตรวจสอบผู้ใช้ที่สร้างใหม่: `auditclasses=general, tcpip`	เพิ่มรายการตรวจสอบต่อไปนี้สำหรับผู้ใช้ root และผู้ใช้ทั่วไป: **root:** general tcpip aixpert *User:* general เพิ่มรายการต่อไปนี้ใน stanza ผู้ใช้ของไฟล์ /usr/lib/security/mkuser.default สำหรับการเปิดใช้งานการตรวจสอบผู้ใช้ที่สร้างใหม่: `auditclasses=general`	ใช่

cronjob ต้องทำงานทุกชั่วโมงและตรวจสอบขนาดของ /audit ถ้า Audit Freespace Equation เป็นจริง Audit Trail Copy Actions ต้องถูกดำเนินการ Audit Freespace Equation ถูกกำหนด เพื่อให้แน่ใจว่าระบบไฟล์ /audit ไม่เต็ม ถ้าระบบไฟล์ /audit เต็ม Audit Trail Copy Actions ทำงานเสร็จ (ปิดใช้งานการตรวจสอบ ทำการสำรองข้อมูล /audit/trail ไปยัง /audit/trailOneLevelBack และเปิดใช้งานการตรวจสอบอีกครั้ง)

กลุ่ม AIX Security Expert /etc/inittab Entries

AIX Security Expert ใส่เครื่องหมาย ความคิดเห็นรายการที่เจาะจงใน /etc/inittab เพื่อ ไม่ให้เริ่มทำงานระบบเปิดเครื่องใหม่

ตารางที่ 23. รายการ AIX Security Expert /etc/inittab

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งาน **qdaemon**/เปิดใช้งาน **qdaemon**	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inittab.conf: `qdaemon:2:wait:/usr/bin/startsrc -sqdaemon`	**High Level Security** ความคิดเห็น **Medium Level Security** ความคิดเห็น **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่
ปิดใช้งาน **lpd** daemon/เปิดใช้งาน **lpd** daemon	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inittab.conf: `lpd:2:once:/usr/bin/startsrc -s lpd`	**High Level Security** ความคิดเห็น **Medium Level Security** ความคิดเห็น **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่
ปิดใช้งาน CDE/เปิดใช้งาน CDE	ถ้าระบบไม่มี LFT ที่ถูกกำหนดคอนฟิก ให้ใส่เครื่องหมายความคิดเห็น หรือเอาออกจากรายการต่อไปนี้ใน /etc/inittab: `dt:2:wait:/etc/rc.dt`	**High Level Security** ความคิดเห็น **Medium Level Security** ความคิดเห็น **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่
ปิดใช้งาน **piobe** daemon/เปิดใช้งาน **piobe** daemon	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inittab.conf: `piobe:2:wait:/usr/lib/lpd/pio/etc/pioinit >/dev/null 2>&1`	**High Level Security** ความคิดเห็น **Medium Level Security** ความคิดเห็น **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่

กลุ่ม AIX Security Expert /etc/rc.tcpip Settings

AIX Security Expert ใส่เครื่องหมายความคิดเห็นรายการที่เจาะจงใน /etc/rc.tcpip เพื่อ ไม่ให้เริ่มทำงานระบบเปิดเครื่องใหม่

ตารางต่อไปนี้แสดงรายการที่ใส่เครื่องหมายความคิดเห็นใน /etc/rc.tcpip เพื่อ ไม่ให้เริ่มทำงานเมื่อระบบเปิดเครื่องใหม่

ตารางที่ 24. ค่าติดตั้ง AIX Security Expert /etc/rc.tcpip

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งาน mail client/เปิดใช้งาน mail client	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/lib/sendmail "$src_running"`	**High Level Security** ความคิดเห็น **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่
ปิดใช้งาน routing daemon	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/routed "$src_running" -q`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งาน **mrouted** daemon	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/mrouted "$src_running"`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่

ตารางที่ 24. ค่าติดตั้ง AIX Security Expert /etc/rc.tcpip (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งาน **timed** daemon	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/timed`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ใช่ **AIX Standard Settings** ใช่	ใช่
ปิดใช้งาน **rwhod** daemon	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/rwhod "$src_running"`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งาน print daemon	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/lpd "$src_running"`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งาน SNMP daemon/เปิดใช้งาน SNMP daemon	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/snmpd "$src_running"`	**High Level Security** ความคิดเห็น **Medium Level Security** ความคิดเห็น **Low Level Security** ปิดใช้งาน SNMP daemon **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่

ตารางที่ 24. ค่าติดตั้ง AIX Security Expert /etc/rc.tcpip (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
หยุดทำงาน DHCP Agent	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/dhcprd "$src_running"`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
หยุดทำงาน DHCP Server	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/dhcpsd "$src_running"`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
หยุดทำงาน autoconf6	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/autoconf6 ″"`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งาน DNS daemon	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/named "$src_running"`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่

ตารางที่ 24. ค่าติดตั้ง AIX Security Expert /etc/rc.tcpip (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งาน **gated** daemon	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/gated "$src_running"`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ใช่ **AIX Standard Settings** ใช่	ใช่
หยุดทำงาน DHCP Client	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/dhcpd "$src_running"`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งาน DPID2 daemon	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/dpid2 "$src_running"`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งาน NTP daemon	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/rc.tcpip: `start /usr/sbin/xntpd "$src_running"`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่

กลุ่ม AIX Security Expert /etc/inetd.conf Settings

AIX Security Expert ใส่ เครื่องหมายความคิดเห็นในรายการเฉพาะใน /etc/inetd.conf

การติดตั้งดีฟอลต์ของ AIX ทำให้ เน็ตเวิร์กเซอร์วิสจำนวนหนึ่งสามารถตรวจหาช่องโหว่ด้านความปลอดภัย ของระบบได้ AIX Security Expert ปิดใช้งาน เซอร์วิสที่ไม่จำเป็นและไม่ปลอดภัยโดยการใส่เครื่องหมายความคิดเห็นบนรายการ ที่เกี่ยวข้องในไฟล์ /etc/inetd.conf สำหรับ AIX Standard Settings รายการ เหล่านี้จะไม่ถูกใส่เครื่องหมายความคิดเห็น ตารางต่อไปนี้แสดงรายการที่ ถูกใส่เครื่องหมายความคิดเห็นหรือเอาเครื่องหมายออกใน /etc/inetd.conf

ตารางที่ 25. ค่าติดตั้ง AIX Security Expert /etc/inetd.conf

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งาน **sprayd** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `sprayd sunrpc_udp udp wait root \ /usr/lib/netsvc/`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งานเซอร์วิส UDP chargen ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `chargen dgram udp wait root internal`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งาน telnet / เปิดใช้งาน telnet	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf: `telnet stream tcp6 nowait root \ /usr/sbin/telnetd telnetd`	**High Level Security** ความคิดเห็น **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่

ตารางที่ 25. ค่าติดตั้ง AIX Security Expert /etc/inetd.conf (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งานเซอร์วิส UDP Echo ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `echo dgram udp wait root internal`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่

ปิดใช้งาน **tftp** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`tftp dgram udp6 SRC nobody \`
	`/usr/sbin/tftpd tftpd -n`
		High Level Security
		ใช่
		Medium Level Security
		ใช่
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ใช่
			ใช่

ปิดใช้งาน **krshd** daemon	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`kshell stream tcp nowait root \`
	`/usr/sbin/krshd krshd`
		High Level Security
		ใช่
		Medium Level Security
		ไม่มีผล
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ใช่
			ใช่

ปิดใช้งาน **rusersd** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`rusersd sunrpc_udp udp wait root \`
	`/usr/lib/netsvc/`
		High Level Security
		ใช่
		Medium Level Security
		ใช่
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ใช่
			ใช่

ตารางที่ 25. ค่าติดตั้ง AIX Security Expert /etc/inetd.conf (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ

ปิดใช้งาน **rexecd** ใน /etc/inetd.conf / เปิดใช้งาน **rexecd** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`exec stream tcp6 nowait root \`
	`/usr/sbin/rexecd rexecd`
		High Level Security
		ความคิดเห็น
		Medium Level Security
		ความคิดเห็น
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ไม่มีความคิดเห็น
			ใช่

ปิดใช้งาน POP3D	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`pop3 stream tcp nowait root \`
	`/usr/sbin/pop3d pop3d`
		High Level Security
		ใช่
		Medium Level Security
		ไม่มีผล
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ใช่
			ใช่

ปิดใช้งาน **pcnfsd** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`pcnfsd sunrpc_udp udp wait root \`
	`/usr/sbin/rpc.pcnfsd pcnfsd`
		High Level Security
		ใช่
		Medium Level Security
		ไม่มีผล
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ใช่
			ใช่

ปิดใช้งาน **bootpd** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`bootps dgram udp wait root \`
	`/usr/sbin/bootpd`
		High Level Security
		ใช่
		Medium Level Security
		ใช่
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ใช่
			ใช่

ตารางที่ 25. ค่าติดตั้ง AIX Security Expert /etc/inetd.conf (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งาน **rwalld** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `rwalld sunrpc_udp udp wait root \ /usr/lib/netsvc/`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งานเซอร์วิส UDP discard ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `discard dgram udp wait root \ internal`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งานเซอร์วิส TCP daytime ใน /etc/inetd.conf / เปิดใช้งานเซอร์วิส TCP daytime ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf: `daytime stream tcp nowait root \ internal`	**High Level Security** ความคิดเห็น **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่
ปิดใช้งาน **netstat** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `netstat stream tcp nowait nobody \ /usr/bin/netstat`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่

ตารางที่ 25. ค่าติดตั้ง AIX Security Expert /etc/inetd.conf (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งาน **rshd** daemon/เปิดใช้งาน **rshd** daemon	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf: `shell stream tcp6 nowait root \ /usr/sbin/rshd rshd rshd`	**High Level Security** ความคิดเห็น **Medium Level Security** ความคิดเห็น **Low Level Security** ความคิดเห็น **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่
ปิดใช้งานเซอร์วิส **cmsd** ใน /etc/inetd.conf / เปิดใช้งานเซอร์วิส **cmsd** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf: `cmsd sunrpc_udp udp wait root \ /usr/dt/bin/rpc.cms cmsd`	**High Level Security** ความคิดเห็น **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่
ปิดใช้งานเซอร์วิส **ttdbserver** ใน /etc/inetd.conf / เปิดใช้งานเซอร์วิส **ttdbserver** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf: `ttdbserver sunrpc_tcp tcp wait \ root /usr/dt/bin/`	**High Level Security** ความคิดเห็น **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่
ปิดใช้งาน **uucpd** ใน /etc/inetd.conf / เปิดใช้งาน **uucpd** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf: `uucp stream tcp nowait root \ /usr/sbin/uucpd uucpd`	**High Level Security** ความคิดเห็น **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่

ตารางที่ 25. ค่าติดตั้ง AIX Security Expert /etc/inetd.conf (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งานเซอร์วิส UDP time ใน /etc/inetd.conf / เปิดใช้งานเซอร์วิส UDP time ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf: `time dgram udp wait root internal`	**High Level Security** ความคิดเห็น **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่

ปิดใช้งานเซอร์วิส TCP time ใน /etc/inetd.conf / เปิดใช้งานเซอร์วิส TCP time ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf:
	`time stream tcp nowait root \`
	`internal`
		High Level Security
		ความคิดเห็น
		Medium Level Security
		ไม่มีผล
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ไม่มีความคิดเห็น
			ใช่

ปิดใช้งาน **rexd** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`rexd sunrpc_tcp tcp wait root \`
	`/usr/sbin/tpc.rexd.rexd rexd`
		High Level Security
		ใช่
		Medium Level Security
		ใช่
		Low Level Security
		ใช่
		AIX Standard Settings
		ใช่
			ใช่

ปิดใช้งานเซอร์วิส TCP chargen ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`chargen stream tcp nowait root \`
	`internal`
		High Level Security
		ใช่
		Medium Level Security
		ไม่มีผล
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ใช่
			ใช่

ตารางที่ 25. ค่าติดตั้ง AIX Security Expert /etc/inetd.conf (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ

ปิดใช้งาน **rlogin** ใน /etc/inetd.conf / เปิดใช้งาน **rlogin** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf:
	`login stream tcp6 nowait root \`
	`/usr/sbin/rlogind rlogind`
		High Level Security
		ความคิดเห็น
		Medium Level Security
		ความคิดเห็น
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ไม่มีความคิดเห็น
			ใช่

ปิดใช้งาน talk ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf:
	`talk dgram udp wait root \`
	`/usr/sbin/talkd talkd`
		High Level Security
		ความคิดเห็น
		Medium Level Security
		ความคิดเห็น
		Low Level Security
		ความคิดเห็น
		AIX Standard Settings
		ไม่มีความคิดเห็น
			ใช่

ปิดใช้งาน **fingerd** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`finger stream tcp nowait nobody \`
	`/usr/sbin/fingerd fingerd`
		High Level Security
		ใช่
		Medium Level Security
		ใช่
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ใช่
			ใช่

ปิดใช้งาน FTP / เปิดใช้งาน FTP	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf:
	`ftp stream tcp6 nowait root \`
	`/usr/sbin/ftpd ftpd`
		High Level Security
		ความคิดเห็น
		Medium Level Security
		ไม่มีผล
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ไม่มีความคิดเห็น
			ใช่

ตารางที่ 25. ค่าติดตั้ง AIX Security Expert /etc/inetd.conf (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ

ปิดใช้งาน IMAPD	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`imap2 stream tcp nowait root \`
	`/usr/sbin/imapd imapd`
		High Level Security
		ใช่
		Medium Level Security
		ไม่มีผล
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ใช่
			ใช่

ปิดใช้งาน **comsat** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`comsat dgram udp wait root \`
	`/usr/sbin/comsat comsat`
		High Level Security
		ใช่
		Medium Level Security
		ไม่มีผล
		Low Level Security
		ไม่มีผล
		AIX Standard Settings
		ใช่
			ใช่

ปิดใช้งาน **rquotad** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf:
	`rquotad sunrpc_udp udp wait root \`
	`/usr/sbin/rpc.rquotad`
		High Level Security
		ใช่
		Medium Level Security
		ใช่
		Low Level Security
		ใช่
		AIX Standard Settings
		ใช่
			ใช่

ปิดใช้งานเซอร์วิส UDP daytime ใน /etc/inetd.conf / เปิดใช้งานเซอร์วิส UDP daytime ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นหรือเอาออกรายการต่อไปนี้ใน /etc/inetd.conf: `daytime dgram udp wait root internal`	**High Level Security** ความคิดเห็น **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีความคิดเห็น	ใช่

ตารางที่ 25. ค่าติดตั้ง AIX Security Expert /etc/inetd.conf (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งาน **krlogind** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `klogin stream tcp nowait root \ /usr/sbin/krlogind krlogind`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งานเซอร์วิส TCP Discard ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `discard stream tcp nowait root \ internal`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งานเซอร์วิส TCP echo ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `echo stream tcp nowait root internal`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งาน **sysstat** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `systat stream tcp nowait nodby \ /usr/bin/ps ps -ef`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่

ตารางที่ 25. ค่าติดตั้ง AIX Security Expert /etc/inetd.conf (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งาน **rstatd** ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `rstatd sunrpc_udp udp wait root \ /usr/sbin/rpc.rstatd rstatd`	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ปิดใช้งาน dtspc ใน /etc/inetd.conf	ใส่เครื่องหมายความคิดเห็นรายการต่อไปนี้ใน /etc/inetd.conf: `dtspc stream tcp nowait root \ /usr/dt/bin/dtspcd`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่

กลุ่ม AIX Security Expert Disable SUID of Commands

โดยดีฟอลต์ คำสั่งต่อไปนี้ถูกติดตั้งด้วย ชุดบิต SUID สำหรับการรักษาความปลอดภัย High, Medium และ Low บิตนี้จะไม่ถูกตั้งค่า สำหรับ AIX Standard Settings บิต SUID ถูกเรียกคืนบนคำสั่งเหล่านี้

ตารางที่ 26. AIX Security Expert Disable SUID of Commands

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
hls_filepermgr	ตัวจัดการสิทธิในไฟล์: รันคำสั่ง **fpm** ด้วยอ็อพชัน high เพื่อลบ setuid, setgid ออกจากคำสั่งที่มีสิทธิพิเศษ	High Level Security	ใช่
mls_filepermgr	ตัวจัดการสิทธิในไฟล์: รันคำสั่ง **fpm** ด้วยอ็อพชัน medium เพื่อลบ setuid, setgid ออกจากคำสั่งที่มีสิทธิพิเศษ	Medium Level Security	ใช่
lls_filepermgr	ตัวจัดการสิทธิในไฟล์: รันคำสั่ง **fpm** ด้วยอ็อพชัน low เพื่อลบ setuid, setgid ออกจากคำสั่งที่มีสิทธิพิเศษ	Low Level Security	ใช่

กลุ่ม AIX Security Expert Disable Remote Services

AIX Security Expert ปิดใช้งาน คำสั่งที่ไม่ปลอดภัยสำหรับ High Level Security และ Medium Level Security

คำสั่งและ daemon ต่อไปนี้ถูกนำไปใช้ประโยชน์บ่อยครั้งเพื่อ ค้นหาช่องโหว่ด้านความปลอดภัย สำหรับ High Level Security และ Medium Level Security ความปลอดภัยที่ปลอดภัยเหล่านี้จะถูกปฏิเสธสิทธิการทำงาน และ daemons ถูกปิดใช้งาน สำหรับ Low Level Security คำสั่งและ daemons เหล่านี้ไม่ได้รับผลกระทบ สำหรับ AIX Standard Settings คำสั่งและ daemons เหล่านี้ถูกเปิดเพื่อใช้งาน

- **rcp**
- **rlogin**
- **rsh**
- **tftp**
- **rlogind**
- **rshd**
- **tftpd**

ตารางที่ 27. ปิดใช้งานรีโมตเซอร์วิส AIX Security Expert

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
เปิดใช้งาน daemon ที่ไม่ปลอดภัย	ถ้า TCB ถูกเปิดใช้งาน ให้ตั้งค่าสิทธิการทำงาน ของ **rlogind, rshd** และ **tftpd** daemons, อัพเดตฐานข้อมูล sysck ด้วยการเปลี่ยนบิตโหมดสำหรับ daemons เหล่านี้ ถ้า TCB ไม่เปิดใช้งาน สิทธิการทำงานบน **rlogind, rshd** และ **tftpd** daemons จะถูกตั้งค่า	**High Level Security** ไม่มีผล **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีผล	ใช่
ปิดใช้งานคำสั่งที่ไม่ปลอดภัย	1. ถ้า TCB ถูกเปิดใช้งาน ให้ลบสิทธิการทำงานของคำสั่ง **rcp, rlogin, rsh** และ **tftp** และอัพเดตฐานข้อมูล **sysck** ด้วยการเปลี่ยนบิตโหมดสำหรับคำสั่งเหล่านี้ ถ้า TCB ไม่เปิดใช้งาน ให้ลบสิทธิการทำงานบนคำสั่ง **rcp, rlogin** และ **rsh** 2. หมุดทำงาน instances ปัจจุบันของ คำสั่ง**rcp, rlogin, rsh, tftp** และ **uftp** ยกเว้นว่าหนึ่งในคำสั่งเหล่านี้ เป็นกระบวนการพาเรนต์ของ AIX Security Expert 3. เพิ่ม `tcpip:` stanza ใน `/etc/security/config` เพื่อ จำกัดการใช้งาน `.netrc` ใน **ftp** และ **rexec**	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีผล	ใช่
เปิดใช้งานคำสั่งที่ไม่ปลอดภัย	1. ถ้า TCB ถูกเปิดใช้งาน ให้ตั้งค่าสิทธิการทำงานของคำสั่ง **rcp, rlogin, rsh** และ **tftp** และอัพเดตฐานข้อมูล **sysck** ด้วยการเปลี่ยนบิตโหมดของคำสั่งเหล่านี้ ถ้า Tไม่เปิดใช้งาน ให้ตั้งค่าสิทธิการทำงานบนคำสั่ง **rcp, rlogin** และ **rsh** 2. ลบไฟล์ `/etc/security/config` ออก	**High Level Security** ไม่มีผล **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่

ตารางที่ 27. ปิดใช้งานรีโมตเซอร์วิส AIX Security Expert (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ปิดใช้งาน daemon ที่ไม่ปลอดภัย	1. ถ้า TCB ถูกเปิดใช้งาน ให้ลบสิทธิการทำงานของ **rlogind, rshd** และ **tftpd** daemons และอัพเดตฐานข้อมูล **sysck** ด้วยการเปลี่ยนบิตโหมดของ daemons เหล่านี้ ถ้า TCB ไม่เปิดใช้งาน ให้ลบสิทธิการทำงานของ **rlogind, rshd** และ **tftpd** daemons 2. หยุดทำงาน instances ปัจจุบันของ **rlogind, rshd** และ **tftpd** daemons ยกเว้นว่าหนึ่งใน daemons เหล่านี้ เป็นกระบวนการพาเรนต์ของ AIX Security Expert	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีผล	ใช่
หยุดทำงาน NFS daemon	• ลบการเมาท์ NFS ทั้งหมด • ปิดใช้งาน NFS • ลบสคริปต์การเริ่มทำงาน NFS ออกจาก `/etc/inittab`	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีผล	ใช่
ปิดใช้งาน NFS daemon	• เอ็กซ์พอร์ตรายการทั้งหมดที่แสดงใน `/etc/exports` • เพิ่มรายการใน `/etc/inittab` เพื่อรัน `/etc/rc.nfs` เมื่อ ระบบเริ่มทำงานต่อ • รัน `/etc/rc.nfs` ทันที	**High Level Security** ไม่มีผล **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่

กลุ่มการเข้าถึง AIX Security Expert Remove ที่ไม่จำเป็นต้องใช้การพิสูจน์ตัวตน

AIX สนับสนุน เซอร์วิสบางเซอร์วิสที่ไม่จำเป็นต้องใช้การพิสูจน์ตัวตนผู้ใช้เพื่อล็อกอินเข้าสู่ เน็ตเวิร์ก

ไฟล์ `/etc/hosts.equiv` และไฟล์ `$HOME/.rhosts` โลคัลใดๆ กำหนดโฮสต์และบัญชีผู้ใช้ที่สามารถรันคำสั่งรีโมตบนโฮสต์โลคัล ได้โดยไม่ต้องใช้รหัสผ่าน เว้นแต่ว่าจะจำเป็นต้องใช้ความสามารถนี้อย่างชัดเจน มิเช่นนั้นแล้วควรลบไฟล์เหล่านี้ออก

ตารางที่ 28. AIX Security Expert ลบ การเข้าถึงที่ไม่จำเป็นต้องใช้การพิสูจน์ตัวตน

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ลบเซอร์วิส rhosts และ netrc	ไฟล์ .rhosts และ .netrc เก็บชื่อผู้ใช้และรหัสผ่านในรูปแบบข้อความธรรมดา ซึ่งอาจถูกนำไปใช้ได้	**High Level Security** ลบไฟล์ .rhosts และ .netrc ออกจากโฮมไดเร็กทอรีของผู้ใช้ทั้งหมด รวมถึง root **Medium Level Security** ลบไฟล์ .rhosts และ .netrc ออกจากโฮมไดเร็กทอรีของผู้ใช้ทั้งหมด รวมถึง root **Low Level Security** ลบไฟล์ .rhosts และ .netrc ออกจากโฮมไดเร็กทอรีของ root **AIX Standard Settings** ลบไฟล์ .rhosts และ .netrc ออกจากโฮมไดเร็กทอรีของผู้ใช้ทั้งหมด รวมถึง root	ใช่
ลบรายการออกจากไฟล์ /etc/hosts.equiv	ไฟล์ /etc/hosts.equiv พร้อมด้วยไฟล์ $HOME/.rhosts ของผู้ใช้โลคัล กำหนดว่าผู้ใช้ใดบนโฮสต์เพื่อนบ้านที่จะได้รับอนุญาตให้รันคำสั่งแบบรีโมต บนโฮสต์โลคัล ถ้ามีบุคคลใดบนโฮสต์เพื่อนบ้านทราบ รายละเอียดผู้ใช้และชื่อโฮสต์ บุคคลเหล่านั้นสามารถหาวิธีรัน คำสั่งแบบรีโมตบนโฮสต์โลคัลได้โดยไม่ต้องใช้การพิสูจน์ตัวตน	**High Level Security** ลบรายการทั้งหมดออกจาก /etc/hosts.equiv **Medium Level Security** ลบรายการทั้งหมดออกจาก /etc/hosts.equiv **Low Level Security** ลบรายการทั้งหมดออกจาก /etc/hosts.equiv **AIX Standard Settings** ลบรายการทั้งหมดออกจาก /etc/hosts.equiv	ใช่

กลุ่ม AIX Security Expert Tuning Network Options

การปรับอ็อพชันเน็ตเวิร์กเป็นค่าที่เหมาะสมถือเป็น งานส่วนใหญ่ของการรักษาความปลอดภัย การตั้งค่าแอ็ตทริบิวต์เน็ตเวิร์กเป็น 0 เพื่อปิดใช้งานอ็อพชัน และการตั้งค่าแอ็ตทริบิวต์เน็ตเวิร์กเป็น 1 เพื่อเปิดใช้งานอ็อพชัน

ตารางต่อไปนี้แสดงรายการค่าติดตั้งแอ็ตทริบิวต์เน็ตเวิร์กสำหรับ High, Medium และ Low Level Security ตารางนี้ยังมีรายละเอียด ว่าค่าที่แนะนำของอ็อพชันเน็ตเวิร์กเฉพาะใดๆ นั้นช่วยทำให้แน่ใจ ในความปลอดภัยของระบบได้อย่างไร

ตารางที่ 29. AIX Security Expert Tuning Network Options สำหรับความปลอดภัยบนเน็ตเวิร์ก

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
อ็อพชันเน็ตเวิร์ก ipsrcrouteforward	ระบุว่าระบบจะส่งต่อ แพ็กเก็ตที่กำหนดเส้นทางโดยซอร์สหรือไม่ การปิดใช้งาน ipsrcrouteforward ป้องกันการเข้าถึง โดยการโจมตีเส้นทางซอร์ส	**High Level Security** 0 **Medium Level Security** 0 **Low Level Security** ไม่มีผล **AIX Standard Settings** 1	ใช่
อ็อพชันเน็ตเวิร์ก ipignoreredirects	ระบุว่าจะประมวลผลการเปลี่ยนเส้นทาง ที่ได้รับหรือไม่	**High Level Security** 1 **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่จำกัด	ใช่
อ็อพชันเน็ตเวิร์ก clean_partial_conns	ระบุว่าจะเลี่ยงการโจมตีโดยใช้อักขระการซิงโครไนซ์ (SYN) หรือไม่	**High Level Security** 1 **Medium Level Security** 1 **Low Level Security** 1 **AIX Standard Settings** ไม่จำกัด	ใช่
อ็อพชันเน็ตเวิร์ก ipsrcrouterecv	ระบุว่าระบบจะยอมรับ แพ็กเก็ตที่กำหนดเส้นทางโดยซอร์สหรือไม่ การปิดใช้งาน ipsrcrouterecv ป้องกันการเข้าถึง โดยการโจมตีเส้นทางซอร์ส	**High Level Security** 0 **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่จำกัด	ใช่

ตารางที่ 29. AIX Security Expert Tuning Network Options สำหรับความปลอดภัยบนเน็ตเวิร์ก (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
อ็อพชันเน็ตเวิร์ก ipforwarding	ระบุว่าเคอร์เนลควรส่งต่อ แพ็กเก็ตหรือไม่ การปิดใช้งาน ipforwarding ป้องกันแพ็กเก็ตที่ถูกเปลี่ยนเส้นทางมิให้ส่งไปยัง เน็ตเวิร์กรีโมต	**High Level Security** 0 **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่จำกัด	ใช่
อ็อพชันเน็ตเวิร์ก ipsendredirects	ระบุว่าเคอร์เนลควรส่งสัญญาณ แจ้งการเปลี่ยนเส้นทางหรือไม่ การปิดใช้งาน ipsendredirects ป้องกันแพ็กเก็ตที่ถูกเปลี่ยนเส้นทางมิให้ส่งไปยัง เน็ตเวิร์กรีโมต	**High Level Security** 0 **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** 1	ใช่
อ็อพชันเน็ตเวิร์ก ip6srcrouteforward	ระบุว่าระบบส่งต่อ แพ็กเก็ต IPv6 ที่กำหนดเส้นทางโดยซอร์สหรือไม่ การปิดใช้งาน ip6srcrouteforward ป้องกันการเข้าถึง โดยการโจมตีเส้นทางซอร์ส	**High Level Security** 0 **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** 1	ใช่
อ็อพชันเน็ตเวิร์ก directed_broadcast	ระบุว่าจะอนุญาตการกระจายโดยตรง ไปยังเกตเวย์หรือไม่ การปิดใช้งาน directed_broadcast ช่วยป้องกัน แพ็กเก็ตที่ถูกส่งตรงมิให้ส่งไปยังเน็ตเวิร์กรีโมต	**High Level Security** 0 **Medium Level Security** 0 **Low Level Security** 0 **AIX Standard Settings** ไม่จำกัด	ใช่

ตารางที่ 29. AIX Security Expert Tuning Network Options สำหรับความปลอดภัยบนเน็ตเวิร์ก (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
อ็อพชันเน็ตเวิร์ก tcp_pmtu_discover	เปิดใช้งานหรือปิดใช้งานการค้นหาพาธ MTU สำหรับแอ็พพลิเคชัน TCP การปิดใช้งาน tcp_pmtu_discover ป้องกันการเข้าถึงโดย การโจมตีเส้นทางซอร์ส	**High Level Security** 0 **Medium Level Security** 0 **Low Level Security** 0 **AIX Standard Settings** 1	ใช่
อ็อพชันเน็ตเวิร์ก bcastping	อนุญาตให้ตอบกลับแพ็กเก็ต ICMP echo ที่ส่ง ไปยังแอดเดรสการกระจาย การปิดใช้งาน bcastping ป้องกันการโจมตีโดย smurf	**High Level Security** 0 **Medium Level Security** 0 **Low Level Security** 0 **AIX Standard Settings** ไม่จำกัด	ใช่
อ็อพชันเน็ตเวิร์ก icmpaddressmask	ระบุว่าระบบตอบกลับ การร้องขอ ICMP address mask หรือไม่ การปิดใช้งาน icmpaddressmask ป้องกัน การเข้าถึงผ่านการโจมตีเส้นทางซอร์ส	**High Level Security** 0 **Medium Level Security** 0 **Low Level Security** 0 **AIX Standard Settings** ไม่จำกัด	ใช่
อ็อพชันเน็ตเวิร์ก udp_pmtu_discover	เปิดใช้งานหรือปิดใช้งานการค้นหาพาธ maximum transfer unit (MTU) สำหรับแอ็พพลิเคชัน UDP การปิดใช้งาน udp_pmtu_discover ป้องกันการเข้าถึงผ่านการโจมตีเส้นทางซอร์ส	**High Level Security** 0 **Medium Level Security** 0 **Low Level Security** 0 **AIX Standard Settings** 1	ใช่

ตารางที่ 29. AIX Security Expert Tuning Network Options สำหรับความปลอดภัยบนเน็ตเวิร์ก (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
อ็อพชันเน็ตเวิร์ก ipsrcroutesend	ระบุว่าแอ็พพลิเคชันสามารถส่ง แพ็กเก็ตที่กำหนดเส้นทางโดยซอร์สหรือไม่ การปิดใช้งาน ipsrcroutesend ป้องกันการเข้าถึง โดยการโจมตีเส้นทางซอร์ส	**High Level Security** 0 **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** 1	ใช่
อ็อพชันเน็ตเวิร์ก nonlocsrcroute	ระบุ Internet Protocol ว่า แพ็กเก็ตที่กำหนดเส้นทางโดยซอร์ตแบบเข้มงวดสามารถกำหนดแอดเดรสไปยังโฮสต์ภายนอก เน็ตเวิร์กโลคัลได้หรือไม่ การปิดใช้งาน nonlocsrcroute ป้องกัน การเข้าถึงผ่านการโจมตีเส้นทางซอร์ส	**High Level Security** 0 **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่จำกัด	ใช่
อ็อพชันเน็ตเวิร์ก tcp_tcpsecure	ป้องกันการเชื่อมต่อ TCP มิให้มีจุดอ่อน ค่า: • 0 = ไม่มีการป้องกัน • 1 = การส่ง SYN ปลอมไปยังการเชื่อมต่อที่สร้าง • 2 = การส่ง RST ปลอมไปยังการเชื่อมต่อที่สร้าง • 3 = การป้องกันข้อมูลในการเชื่อมต่อ TCP ที่สร้าง • 5–7 = การรวมกันของจุดอ่อนที่กล่าวด้านบน	**High Level Security** 7 **Medium Level Security** 7 **Low Level Security** 5 **AIX Standard Settings** ไม่จำกัด	ใช่
อ็อพชันเน็ตเวิร์ก sockthresh	ระบุขีดจำกัดการใช้งานหน่วยความจำเน็ตเวิร์ก ไม่อนุญาต ให้มีการเชื่อมต่อซ็อกเก็ตใหม่เกินค่าที่ sockthresh tunable ระบุจำนวนหน่วยความจำเน็ตเวิร์กสูงสุดที่สามารถ จัดสรรสำหรับซ็อกเก็ต	**High Level Security** 60 **Medium Level Security** 70 **Low Level Security** 85 **AIX Standard Settings** ไม่จำกัด	ใช่

อ็อพชันเน็ตเวิร์กต่อไปนี้สัมพันธ์กับผลการทงานของเน็ตเวิร์ก มากกว่าความปลอดภัยของเน็ตเวิร์ก

ตารางที่ 30. AIX Security Expert Tuning Network Options สำหรับผลการทำงานของเน็ตเวิร์ก

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
อ็อพชันเน็ตเวิร์ก rfc1323	rfc1323 tunable เปิดใช้งานอ็อพชันการปรับสเกล หน้าต่าง TCP	**High Level Security** 1 **Medium Level Security** 1 **Low Level Security** 1 **AIX Standard Settings** ไม่จำกัด	ใช่
อ็อพชันเน็ตเวิร์ก tcp_sendspace	tcp_sendspace tunable ระบุปริมาณ ข้อมูลที่แอ็พพลิเคชันการส่งสามารถทำบัฟเฟอร์ไว้ในเคอร์เนลก่อนแอ็พพลิเคชัน ถูกบล็อกเมื่อมีการเรียกใช้การส่ง	**High Level Security** 262144 **Medium Level Security** 262144 **Low Level Security** 262144 **AIX Standard Settings** 16384	ใช่
อ็อพชันเน็ตเวิร์ก tcp_mssdflt	ขนาดเซ็กเมนต์สูงสุดค่าดีฟอลต์ที่ใช้ในการสื่อสาร กับเน็ตเวิร์กรีโมต	**High Level Security** 1448 **Medium Level Security** 1448 **Low Level Security** 1448 **AIX Standard Settings** 1460	ใช่
อ็อพชันเน็ตเวิร์ก extendednetstats	เปิดใช้งานสถิติเพิ่มมากขึ้นสำหรับเซอร์วิส หน่วยความจำเน็ตเวิร์ก	**High Level Security** 1 **Medium Level Security** 1 **Low Level Security** 1 **AIX Standard Settings** ไม่จำกัด	ใช่

ตารางที่ 30. AIX Security Expert Tuning Network Options สำหรับผลการทำงานของเน็ตเวิร์ก (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
อ็อพชันเน็ตเวิร์ก tcp_recvspace	tcp_recvspace tunable ระบุจำนวนไบต์ ข้อมูลที่ระบบรับสามารถบัฟเฟอร์ในเคอร์เนลบน คิวซ็อกเก็ตการรับ	**High Level Security** 262144 **Medium Level Security** 262144 **Low Level Security** 262144 **AIX Standard Settings** 16384	ใช่
อ็อพชันเน็ตเวิร์ก sb_max	sb_max tunable ตั้งค่าขีดจำกัดบนของ จำนวนบัฟเฟอร์ซ็อกเก็ตที่เข้าคิวไปยังแต่ละซ็อกเก็ต ซึ่งควบคุม จำนวนพื้นที่บัฟเฟอร์ที่ใช้โดยบัฟเฟอร์ที่เข้าคิวไปยัง ซ็อกเก็ตของผู้ส่ง หรือซ็อกเก็ตของผู้รับ	**High Level Security** 1048576 **Medium Level Security** 1048576 **Low Level Security** 1048576 **AIX Standard Settings** 1048576	ใช่

กลุ่มกฎตัวกรอง AIX Security Expert IPsec

AIX Security Expert มี ตัวกรอง IPsec ต่อไปนี้

ตารางที่ 31. กฎตัวกรอง AIX Security Expert IPsec

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
เลี่ยงโฮสต์เป็นเวลา 5 นาที	เลี่ยงหรือบล็อกแพ็กเก็ตที่ต้องการสำหรับ พอร์ต tcp และ udp หลายๆ พอร์ตที่มีช่องโหว่ที่ทราบบนโฮสต์เป็นเวลา ห้านาที โฮสต์จะไม่ยอมรับแพ็กเก็ตใดๆ ที่กำหนดปลายทางไปยังพอร์ตเหล่านั้น เป็นเวลาห้านาที	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีผล	ใช่

ตารางที่ 31. กฎตัวกรอง AIX Security Expert IPsec (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ป้องกันโฮสต์มิให้สแกนพอร์ต	ป้องกันการสแกนพอร์ต โฮสต์รีโมตใดๆ ที่ทำหน้าที่ สแกนพอร์ตจะถูกเลี่ยงหรือบล็อกเป็นเวลาห้านาที แพ็กเก็ตทั้งหมดจาก โฮสต์รีโมตนี้จะไม่ถูกยอมรับเป็นเวลาห้านาที	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีผล	ใช่

กลุ่ม AIX Security Expert Miscellaneous

AIX Security Expert จัดให้มี การตั้งค่าความปลอดภัยต่างๆ สำหรับ High, Medium และ Low Level Security

ตารางที่ 32. กลุ่ม AIX Security Expert Miscellaneous

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ลบจุดออกจากพาธ root	ตรวจสอบไฟล์ `$HOME/.profile`, `$HOME/.kshrc`, `$HOME/.cshrc` และ `$HOME/.login` เพื่อหาจุด (.) ใน ตัวแปรสภาวะแวดล้อม PATH และลบจุดออก ถ้ามีอยู่ **หมายเหตุ:** การลบจุดเกิดขึ้นเมื่อรายการในไฟล์ขึ้นต้นด้วยตัวแปรสภาวะแวดล้อม PATH และมีจุด (.). ไฟล์ไม่มีการเปลี่ยนแปลงหากตัวแปรแสภาวะแวดล้อม PATH มีตัวแปรอื่น หรือถูกตั้งค่าที่ส่งคืนจากโปรแกรมที่เรียกจาก สคริปต์ ตัวอย่างของพาธที่จะไม่เปลี่ยนแปลงมีดังต่อไปนี้, โดยที่ *pathprog* คือโปรแกรมที่ส่งคืน สตริงพาธ: `PATH="$(pathprog)"` ในพาธนี้, จุดถูกลบออกจากพาธก่อนที่เนื้อหา ของตัวแปร *pathprog* ถูกแก้ปัญหา, ดังนั้น จุดใดๆ ที่มีอยู่ในพาธที่ส่งคืนไม่ได้ถูกลบทิ้ง	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ใช่ **AIX Standard Settings** ใช่	ใช่
จำกัดการเข้าถึงระบบ	ทำให้แน่ใจว่าผู้ใช้ root เท่านั้นที่ได้รับอนุญาต ให้รันงาน **cron**	**High Level Security** กำหนดให้ผู้ใช้ root เท่านั้นในไฟล์ `cron.allow` และลบไฟล์ `cron.deny` ออก **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ลบไฟล์ `cron.allow` และลบรายการทั้งหมดในไฟล์ `cron.deny`	ใช่

ตารางที่ 32. กลุ่ม AIX Security Expert Miscellaneous (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
ลบจุดออกจาก /etc/environment	ลบจุด (.) ออกจากตัวแปรสภาวะแวดล้อม PATH ในไฟล์ /etc/environment	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ใช่ **AIX Standard Settings** ใช่	ใช่
ลบจุดออกจากพาธที่ไม่ใช่ root	ลบจุด (.) ออกจากตัวแปรสภาวะแวดล้อม PATH ออกจากไฟล์ **$HOME**/.profile, **$HOME**/.kshrc, **$HOME**/.cshrc และ **$HOME**/.login ของผู้ใช้ที่ไม่ใช่ root ทั้งหมด **หมายเหตุ:** การลบจุดเกิดขึ้นเมื่อรายการในไฟล์ขึ้นต้นด้วยตัวแปรสภาวะแวดล้อม PATH และมีจุด (.). ไฟล์ไม่ถูกเปลี่ยนหากตัวแปรสภาวะแวดล้อม PATH มีตัวแปรอื่น หรือถูกตั้งค่าที่ส่งคืนจากโปรแกรม ที่ถูกเรียกจากสคริปต์ ตัวอย่างของพาธที่จะไม่ถูกเปลี่ยนมีดังต่อไปนี้, โดยที่ *pathprog* คือโปรแกรม ที่ส่งคืนสตริงพาธ: PATH="$(*pathprog*)" ในพาธนี้, จุดถูกลบออกจากพาธก่อนที่เนื้อหา ของตัวแปร *pathprog* จะถูกแก้ไข, ดังนั้น จุดใดๆ ที่มีอยู่ในพาธที่ส่งคืนไม่ถูกลบทิ้ง	**High Level Security** ใช่ **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีผล	ไม่ใช่
เพิ่มผู้ใช้ root ในไฟล์ /etc/ftpusers	เพิ่มชื่อผู้ใช้ root ในไฟล์ /etc/ftpusers เพื่อปิดใช้งาน **ftp** โดย root แบบรีโมต	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่
ลบผู้ใช้ root ในไฟล์ /etc/ftpusers	ลบรายการ root ออกจาก /etc/ftpusers เพื่อ เปิดใช้งาน root **ftp** แบบรีโมต	**High Level Security** ไม่มีผล **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ใช่	ใช่

ตารางที่ 32. กลุ่ม AIX Security Expert Miscellaneous (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ

ตั้งค่าการประกาศล็อกอิน	ตรวจสอบ /etc/security/login.cfg เพื่อ ให้แน่ใจว่าค่าการประกาศไม่ถูกระบุ ถ้าค่าการประกาศดีฟอลต์ กำลังถูกใช้งาน ควรเปลี่ยนแปลงค่าการประกาศ ผู้ประกาศอาจถูกเปลี่ยนได้ หากโลแคลของระบบคือ **en_US** หรือโลแคล ภาษาอังกฤษ อื่นๆ ถ้าตรงตามเกณฑ์นี้ ค่าของแอ็ตทริบิวต์การประกาศใน stanza ดีฟอลต์ของไฟล์ /etc/security/login.cfg จะถูกตั้งค่า เป็นดังนี้:
	`Unauthorized use of this \`
	`system is prohibited.\nlogin:`
	
	หมายเหตุ: The security setting takes effect only for new sessions. The security setting does not take effect in the session where the configuration was set.
		High Level Security
		herald="ห้ามมิให้ผู้ใช้ที่ไม่ได้รับอนุญาตเข้าใช้ระบบนี้\nล็อกอิน:"
		Medium Level Security
		herald="ห้ามมิให้ผู้ใช้ที่ไม่ได้รับอนุญาตเข้าใช้ระบบนี้\nล็อกอิน:"
		Low Level Security
		herald="ห้ามมิให้ผู้ใช้ที่ไม่ได้รับอนุญาตเข้าใช้ระบบนี้\nล็อกอิน:"
		AIX Standard Settings
		herald=
			ใช่

ลบบัญชีผู้ใช้ guest ออก	สำหรับระดับความปลอดภัยสูง, กลาง, และต่ำ, ลบแอคเคาต์เกสต์เช่นเดียวกับข้อมูลของเกสต์บนเครื่อง สำหรับ AIX Standard Settings บัญชีผู้ใช้ guest ถูกสร้างบนระบบ **หมายเหตุ:** ผู้ดูแลระบบ ต้องตั้งค่ารหัสผ่านสำหรับบัญชีผู้ใช้นี้โดยชัดแจ้ง เนื่องจาก AIX Security Expert ไม่ได้รับการออกแบบ ให้จัดการงานแบบมีการโต้ตอบกับผู้ใช้	**High Level Security** ลบบัญชีผู้ใช้ guest และข้อมูล **Medium Level Security** ลบบัญชีผู้ใช้ guest และข้อมูล **Low Level Security** ลบบัญชีผู้ใช้ guest และข้อมูล **AIX Standard Settings** เพิ่มบัญชีผู้ใช้ guest บนเครื่อง	ใช่
สิทธิ Crontab	ทำให้แน่ใจว่างาน **crontab** ของ root เป็นเจ้าของและสามารถเขียนได้เฉพาะ root เท่านั้น	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ใช่ **AIX Standard Settings** ไม่มีผล	ใช่
เปิดใช้การเข้าถึง X-Server	ดำเนินการพิสูจน์ตัวตนเพื่อเข้าถึง X-Server	**High Level Security** การพิสูจน์ตัวตนจำเป็น **Medium Level Security** การพิสูจน์ตัวตนจำเป็น **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่จำเป็น	ไม่ใช่

ตารางที่ 32. กลุ่ม AIX Security Expert Miscellaneous (ต่อ)

ชื่อปุ่มการดำเนินการ	คำอธิบาย	ค่ากำหนดโดย AIX Security Expert	เลิกทำ
สิทธิการสร้างอ็อบเจ็กต์	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ **umask** ของ /etc/security/user ซึ่งระบุสิทธิ การสร้างอ็อบเจ็กต์	**High Level Security** 077 **Medium Level Security** 027 **Low Level Security** ไม่มีผล **AIX Standard Settings** 022	ใช่
ตั้งค่าขนาดไฟล์หลัก	ตั้งค่าที่เหมาะสมให้แก่แอ็ตทริบิวต์ **core** ของ /etc/security/limits ซึ่งระบุ ขนาดไฟล์ระบุหลักสำหรับ root **หมายเหตุ:** The security setting takes effect only for new sessions. The security setting does not take effect in the session where the configuration was set.	**High Level Security** 0 **Medium Level Security** 0 **Low Level Security** 0 **AIX Standard Settings** 2097151	ใช่
เปิดใช้งานคุณลักษณะ SED	เปิดใช้งานคุณลักษณะ Stack Execution Disable และรันคำสั่ง **sedmgr** บนไฟล์ ที่ระบุ **หมายเหตุ:** จำเป็นต้องบูตระบบใหม่เพื่อให้กฎมีผล	**High Level Security** setidfiles **Medium Level Security** ไม่มีผล **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีผล	
Root Password Integrity Check	ทำให้แน่ใจว่ารหัสผ่าน root ไม่ถูกคาดเดาได้ง่าย แอ็ตทริบิวต์ dictionlist ของ root ถูกตั้งค่าเป็น /etc/security/aixpert/dictionary/English เพื่อให้คำสั่ง **passwd** ช่วยให้แน่ใจว่ารหัสผ่าน root ที่กำลังถูกตั้งค่าไม่ถูกคาดเดาได้ง่าย	**High Level Security** ใช่ **Medium Level Security** ใช่ **Low Level Security** ไม่มีผล **AIX Standard Settings** ไม่มีผล	ใช่

AIX Security Expert Undo Security

คุณสามารถเลิกทำค่าติดตั้งการรักษาความปลอดภัยและกฎ AIX Security Expert บางอย่าง

การตั้งค่าการรักษาความปลอดภัย AIX Security Expert และกฎต่อไปนี้ไม่สามารถเลิกทำ:

- ตรวจสอบข้อกำหนดรหัสผ่านสำหรับ High Level Security, Medium Level Security และ Low Level Security
- ตรวจสอบข้อกำหนดผู้ใช้สำหรับ High Level Security, Medium Level Security และ Low Level Security
- ตรวจสอบข้อกำหนดกลุ่มสำหรับ High Level Security, Medium Level Security และ Low Level Security
- การอัพเดต TCB สำหรับ High Level Security, Medium Level Security และ Low Level Security
- เปิดใช้งานการเข้าถึง X-Server สำหรับ High Level Security, Medium Level Security และ Low Level Security
- ลบ dot ออกจากพาธที่ไม่ใช่ root สำหรับ High Level Security และ AIX Standard Settings
- ลบบัญชีผู้ใช้ guest สำหรับ High Level Security, Medium Level Security และ Low Level Security

AIX Security Expert Check Security

AIX Security Expert สามารถ สร้างรายงานค่าติดตั้งความปลอดภัยของระบบและเน็ตเวิร์กปัจจุบัน

หลังจาก AIX Security Expert (คำสั่ง `aixpert`) ถูกใช้เพื่อกำหนดค่าระบบ สามารถใช้อ็อพชัน Check Security เพื่อรายงานการตั้งค่าการกำหนดคอนฟิกต่างๆ ถ้ามีค่าติดตั้งใดในค่าติดตั้งเหล่านี้ ถูกเปลี่ยนแปลงภายนอกการควบคุมของ AIX Security Expert อ็อพชัน AIX Security Expert Check Security จะบันทึกความแตกต่างเหล่านี้ไว้ในไฟล์ `/etc/security/aixpert/check_report.txt`

ตัวอย่าง **talkd** daemon ถูกปิดใช้งานใน `/etc/inetd.conf` เมื่อ คุณใช Low Level Security ถ้า **talkd** daemon ถูกเปิดใช้งานภายหลัง และ Check Security ถูกเรียกให้ทำงาน ข้อมูลนี้ จะถูกบันทึกในไฟล์ `check_report.txt` ดังนี้:

```
coninetdconf.ksh: Service talk using protocol udp should be disabled, however it is enabled now.
```

ถ้า ค่าติดตั้งการรักษาความปลอดภัยที่ใช้ไม่มีการเปลี่ยนแปลง ไฟล์ `check_report.txt` จะว่างเปล่า

อ็อพชัน Check Security ควรรันเป็นระยะและรายงานผลลัพธ์ ควรได้รับการตรวจเพื่อดูว่ามีค่าติดตั้งใดเปลี่ยนแปลง ตั้งแต่ที่ค่าติดตั้งการรักษาความปลอดภัย AIX Security Expert ถูกนำไปใช้ อ็อพชัน Check Security ยังควรรัน เป็นส่วนหนึ่งของการเปลี่ยนแปลงระบบหลักใดๆ เช่นการติดตั้งหรือการอัพเดต ซอฟต์แวร์

ข้อมูลที่เกี่ยวข้อง:

คำสั่ง aixpert

ไฟล์ AIX Security Expert

AIX Security Expert สร้าง และใช้ไฟล์หลายไฟล์

`/etc/security/aixpert/core/aixpertall.xml`
: ประกอบด้วยรายการ XML ของค่าติดตั้งความปลอดภัยที่เป็นไปได้ทั้งหมด

`/etc/security/aixpert/core/appliedaixpert.xml`
: ประกอบด้วยรายการ XML ของค่าติดตั้งความปลอดภัยที่นำไปใช้

/etc/security/aixpert/core/secaixpert.xml
ประกอบด้วยรายการ XML ของค่าติดตั้งความปลอดภัยที่เลือกเมื่อประมวลผล โดย AIX Security Expert GUI

/etc/security/aixpert/log/aixpert.log
ประกอบด้วยบันทึกการติดตามของค่าติดตั้งความปลอดภัยที่นำไปใช้ AIX Security Expert ไม่ใช้ syslog AIX Security Expert เขียน ลง /etc/security/aixpert/log/aixpert.log โดยตรง

หมายเหตุ: ไฟล์ AIX Security Expert XML และบันทึกการทำงานถูกสร้างโดยมีสิทธิดังนี้:

/etc/security/aixpert/
drwx------

/etc/security/aixpert/core/
drwx------

/etc/security/aixpert/core/aixpertall.xml
r--------

/etc/security/aixpert/core/appliedaixpert.xml

/etc/security/aixpert/core/secaixpert.xml

/etc/security/aixpert/log
drwx------

/etc/security/aixpert/log/aixpert.log
-rw-------

/etc/security/aixpert/core/secundoaixpert.xml
rw-------

/etc/security/aixpert/check_report.txt
rw-------

สถานการณ์การรักษาความปลอดภัยระดับสูง AIX Security Expert

นี้คือสถานการณ์แสดงการรักษาความปลอดภัยระดับสูง AIX Security Expert

มุมมอง AIX Security Expert ของระดับความปลอดภัยที่นำมาจากส่วนเอกสาร National Institute of Standards and Technology *Security Configuration Checklists Progarm for IT Pruducts – Guidance for CheckLists Users and Developers* (ค้นหา ชื่อเอกสารจัดพิมพ์บนเว็บไซต์ NIS: http://www.nist.gov/index.html) อย่างไรก็ตาม High, Medium และ Low level security เป็น สิ่งที่แตกต่างกันสำหรับบุคคลที่ต่างกัน เป็นเรื่องสำคัญที่ต้องทำความเข้าใจ สภาวะแวดล้อมที่ระบบของคุณทำงานอยู่ ถ้าคุณเลือกระดับ ความปลอดภัยที่สูงเกินไป คุณอาจล็อกไม่ให้ตัวคุณสามารถเข้าใช้คอมพิวเตอร์ของคุณ ถ้าคุณเลือกระดับความปลอดภัยที่ต่ำเกินไป คอมพิวเตอร์ของคุณ อาจมีช่องโหว่หรือจุดอ่อนให้ถูกโจมตีบนโลกไซเบอร์ได้

ต่อไปนี้เป็นตัวอย่างของสภาวะแวดล้อมที่อาจจำเป็นต้องใช้ High Level Security บ็อบกำลังย้ายระบบของเขาไปไว้ที่รวมที่ผู้ให้บริการอินเตอร์เน็ต ระบบจะถูกเชื่อมต่อโดยตรงกับอินเตอร์เน็ต และจะรันเป็น เซิร์ฟเวอร์ HTTP จะมีข้อมูลผู้ใช้ที่มีความอ่อนไหว และจำเป็นต้องได้รับการดูแล แบบรีโมตโดยบ็อบ ระบบควรได้รับการตั้งค่าและทดสอบบนโลคัลเน็ตเวิร์ก เดี่ยวก่อนที่จะนำระบบมาออนไลน์ร่วมกับ ISP

การรักษาความปลอดภัยระดับสูงเป็นระดับความปลอดภัยที่ถูกต้องสำหรับสภาวะแวดล้อมนี้ แต่บ็อบจำเป็นต้องมีการเข้าถึงระบบแบบรีโมต การรักษาความปลอดภัยระดับสูง ไม่อนุญาตให้**telnet, rlogin, ftp** และการเชื่อมต่อทั่วไปอื่นๆ ที่ส่งรหัสผ่านบนเน็ตเวิร์ก โดยไม่มีการปกปิด รหัสผ่านเหล่านี้อาจถูกแอบดูได้โดยง่าย บนอินเตอร์เน็ต บ็อบจำเป็นต้องหาวิธีการที่ปลอดภัยเพื่อล็อกอินแบบรีโมต เช่น **openssh** บ็อบสามารถอ่านเอกสารคู่มือ AIX Security Expert ฉบับสมบูรณ์ เพื่อดูว่ามีเรื่องใดที่เหมาะกับสภาวะแวดล้อมของของเขาบ้างที่อาจ ถูกขัดขวางโดยการรักษาความปลอดภัยระดับสูง ถ้าเป็นเช่นนั้น เขาสามารถไม่เลือกการทำงานนี้ได้เมื่อ แสดงแผงการรักษาความปลอดภัยระดับสูงโดยละเอียด บ็อบยังควร ตั้งค่าและเริ่มทำงานเซิร์ฟเวอร์ HTTP หรือเซิร์ฟเวอร์อื่นใดที่เขาต้องการ ให้มีบนระบบของเขา

จากนั้นเมื่อบ็อบเลือกการรักษาความปลอดภัยระดับสูง AIX Security Expert จะรับรู้ ว่าเซอร์วิสที่กำลังทำงานนั้นจำเป็นต้องใช้งานและจะไม่บล็อกการเข้าถึง พอร์ตของเซอร์วิสเหล่านั้น เข้าถึงพอร์ตอื่นทั้งหมดอาจเป็นจุดอ่อนและ การรักษาความปลอดภัยระดับสูงจะบล็อกพอร์ตเหล่านี้ หลังการทดสอบการตั้งค่านี้ ในตอนนี้เครื่องของบ็อบก็พร้อมออนไลน์บนอินเตอร์เน็ต

สถานการณ์การรักษาความปลอดภัยระดับกลาง AIX Security Expert

นี้คือสถานการณ์แสดงการรักษาความปลอดภัยระดับกลาง AIX Security Expert

อลิซต้องการให้ระบบมีความปลอดภัยมากขึ้นเพื่อจะเชื่อมต่อกับ เน็ตเวิร์กขององค์กร ซึ่งอยู่ภายใต้การป้องกันโดยไฟร์วอลล์ขององค์กร เน็ตเวิร์กมีความปลอดภัยและได้รับการดูแลอย่างดี ระบบนี้จะใช้โดยผู้ใช้จำนวนมากที่ต้องการเข้าถึงระบบโดยใช้ **telnet** และ**ftp** อลิซต้องการการตั้งค่าความปลอดภัยแบบทั่วไป เช่นการป้องกันการสแกนพอร์ต และการหมดอายุของรหัสผ่าน แต่ระบบก็ยังต้องเปิดให้ใช้วิธีเข้าถึงแบบรีโมตส่วนใหญ่ได้ ในสถานการณ์นี้ การรักษาความปลอดภัยระดับกลาง ถือเป็นการตั้งค่าความปลอดภัยที่เหมาะสมที่สุดสำหรับระบบของอลิซ

สถานการณ์การรักษาความปลอดภัยระดับต่ำ AIX Security Expert

นี้คือสถานการณ์แสดงการรักษาความปลอดภัยระดับต่ำ AIX Security Expert

บรูซทำหน้าที่ดูแลระบบช่วงเวลาหนึ่ง ระบบ ตั้งอยู่บนเน็ตเวิร์กโลคัลที่มีความปลอดภัยแยกต่างหาก ระบบนี้ใช้สำหรับ บุคคลและบริการที่หลากหลาย บรูซต้องการเปลี่ยนระบบ ให้มีความปลอดภัยมากขึ้นจากระดับความปลอดภัยระดับต่ำสุด แต่ไม่สามารถขัดจังหวะการเข้าถึง ระบบไม่ว่ารูปแบบใด การรักษาความปลอดภัยระดับต่ำถือเป็นระดับการรักษาความปลอดภัยที่เหมาะสมสำหรับเครื่องของบรูซ

รายการตรวจสอบความปลอดภัย

ต่อไปนี้คือรายการตรวจสอบการดำเนินการรักษาความปลอดภัยเพื่อ ดำเนินการบนระบบที่ติดตั้งใหม่หรือมีอยู่แล้ว

แม้ว่ารายการนี้จะไม่ใช่รายการตรวจสอบความปลอดภัยที่สมบูรณ์ แต่สามารถใช้เป็นรายการเบื้องต้นเพื่อสร้างรายการตรวจสอบความปลอดภัยสำหรับ สภาวะแวดล้อมของคุณเอง

- เมื่อติดตั้งระบบใหม่ ให้ติดตั้ง AIX จากสื่อบันทึกที่มีความปลอดภัย ดำเนินตามขั้นตอนต่อไปนี้ในตอนการการติดตั้ง:
  - อย่างติดตั้งเดสก์ท็อปซอฟต์แวร์ เช่น CDE, GNOME หรือ KDE บน เซิร์ฟเวอร์

  - ติดตั้งโปรแกรมแก้ไขด้านความปลอดภัยที่จำเป็น และการบำรุงรักษาที่แนะนำใดๆ รวมถึงโปรแกรมแก้ไขระดับเทคโนโลยี ดูที่ IBM System p eServer™ เว็บไซต์ Support Fixes (http://www.ibm.com/support/fixcentral) สำหรับกระดานข่าวเซอร์วิสล่าสุด คำแนะนำด้านการรักษาความปลอดภัย และข้อมูลโปรแกรมแก้ไข
  - สำรองข้อมูลระบบหลังการติดตั้งเริ่มต้น และเก็บ การสำรองระบบไว้ในที่ที่ปลอดภัย
- สร้างรายการควบคุมการเข้าถึงสำหรับไฟล์และไดเร็กทอรีที่จำกัด
- ปิดใช้งานบัญชีผู้ใช้ของผู้ใช้และบัญชีผู้ใช้ของระบบที่ไม่จำเป็น เช่น daemon, bin, sys, adm, lp และ uucp การลบบัญชีผู้ใช้ไม่แนะนำให้ทำ เนื่องจากจะลบข้อมูลบัญชีผู้ใช้ เช่น ID ผู้ใช้ และชื่อผู้ใช้ ซึ่งอาจยังคงมีความเชื่อมโยงกับข้อมูลบนการสำรองข้อมูลระบบ ถ้าผู้ใช้ถูกสร้างโดย ID ผู้ใช้ที่ถูกลบก่อนหน้านี้ และการสำรองข้อมูลระบบ ถูกเรียกคืนบนระบบ ผู้ใช้ใหม่อาจมีการเข้าถึงที่ไม่คาดคิด ไปยังระบบที่เรียกคืน
- ตรวจทานไฟล์ `/etc/inetd.conf`, `/etc/inittab`, `/etc/rc.nfs` และ `/etc/rc.tcpip` เป็นประจำ และ ลบ daemons และเซอร์วิสที่ไม่จำเป็นทั้งหมดออก
- ตรวจสอบว่าสิทธิสำหรับไฟล์ต่อไปนี้ถูกตั้งค่าอย่างถูกต้อง:

```
-rw-rw-r-- root     system  /etc/filesystems
-rw-rw-r-- root     system  /etc/hosts
-rw------- root     system  /etc/inittab
-rw-r--r-- root     system  /etc/vfs
-rw-r--r-- root     system  /etc/security/failedlogin
-rw-rw---- root     audit   /etc/security/audit/hosts
```

- ปิดใช้งานบัญชีผู้ใช้ root มิให้สามารถล็อกอินแบบรีโมต บัญชีผู้ใช้ root ควรสามารถล็อกอินจากคอนโซลระบบเท่านั้น
- เปิดใช้งานการตรวจสอบระบบ สำหรับข้อมูลเพิ่มเติม ดูที่ “ภาพรวมการตรวจสอบ” ในหน้า 150
- เปิดใช้งานนโยบายควบคุมการล็อกอิน สำหรับข้อมูลเพิ่มเติม ดูที่ “การควบคุมล็อกอิน” ในหน้า 39
- ปิดใช้งานสิทธิผู้ใช้ในการรันคำสั่ง `xhost` สำหรับข้อมูลเพิ่มเติม ดูที่ “การจัดการข้อควรพิจารณาของ X11 และ CDE” ในหน้า 45
- ป้องกันการเปลี่ยนแปลงที่ไม่ได้รับอนุญาตในตัวแปรสภาวะแวดล้อม **PATH** สำหรับข้อมูลเพิ่มเติม ดูที่ “ตัวแปรสภาวะแวดล้อม PATH” ในหน้า 63
- ปิดใช้งาน telnet, rlogin และ rsh สำหรับข้อมูลเพิ่มเติม ดูที่ “ความปลอดภัย TCP/IP” ในหน้า 224
- สร้างการควบคุมบัญชีผู้ใช้ของผู้ใช้ สำหรับข้อมูลเพิ่มเติม ดูที่ “การควบคุมบัญชีผู้ใช้” ในหน้า 60
- บังคับใช้นโยบายจำกัดรหัสผ่าน สำหรับข้อมูลเพิ่มเติม ดูที่ “รหัสผ่าน” ในหน้า 72
- สร้างดิสก์โควต้าสำหรับบัญชีผู้ใช้ของผู้ใช้ สำหรับข้อมูลเพิ่มเติม ดูที่ “การกู้คืนจากสภาวะใช้เกินโควต้า” ในหน้า 86
- อนุญาตให้เฉพาะบัญชีผู้ใช้การดูแลจัดการเท่านั้นที่ใช้ **su** มอนิเตอร์ล็อกอินของคำสั่ง **su** ในไฟล์ `/var/adm/sulog`
- เปิดใช้งานการล็อกหน้าจอเมื่อใช้ X-Windows
- จำกัดการเข้าถึงคำสั่ง **cron** และ **at** เฉพาะบัญชีผู้ใช้ที่จำเป็นต้องเข้าถึงเท่านั้น
- ใช้ alias สำหรับคำสั่ง **ls** เพื่แสดงไฟล์และ อักขระที่ซ่อนในชื่อไฟล์
- ใช้ alias สำหรับคำสั่ง **rm** เพื่อเลี่ยงการลบไฟล์ ออกจากระบบโดยบังเอิญ
- เซอร์วิสเน็ตเวิร์กเซอร์วิสที่ไม่จำเป็น สำหรับข้อมูลเพิ่มเติม ดูที่ “เน็ตเวิร์กเซอร์วิส” ในหน้า 233
- ทำการสำรองข้อมูลระบบบ่อยๆ และตรวจสอบ integrity ของ integrity ของการสำรองข้อมูล
- สมัครรับรายการการแจกจ่ายอีเมล์แจ้งเกี่ยวกับความปลอดภัย

สรุปเซอร์วิสระบบ AIX ทั่วไป

ตารางต่อไปนี้แสดงรายการเซอร์วิสระบบทั่วไปเพิ่มเติม ภายใน AIX ใช้ตาราง นี้เพื่อทราบจุดเริ่มต้นสำหรับการให้ความปลอดภัยระบบของคุณ

ก่อนที่คุณให้ความปลอดภัยระบบของคุณ ให้สำรองไฟล์คอนฟิกูเรชันต้นฉบับ ของคุณทั้งหมดก่อน โดยเฉพาะไฟล์ต่อไปนี้:

- `/etc/inetd.conf`
- `/etc/inittab`
- `/etc/rc.nfs`
- `/etc/rc.tcpip`

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
inetd/bootps	inetd	/etc/inetd.conf	เซอร์วิส bootp สำหรับไคลเอ็นต์ที่ไม่มีดิสก์	• จำเป็นสำหรับ Network Installation Management (NIM) และ การบูตระบบรีโมต • ทำงานควบคู่กับ tftp • ปิดใช้งานเป็นส่วนใหญ่
inetd/chargen	inetd	/etc/inetd.conf	ตัวสร้างอักขระ (การทดสอบเท่านั้น)	• มีใช้เป็นเซอร์วิส TCP และ UDP • เปิดโอกาสสำหรับการโจมตี Denial of Service • ปิดใช้งานยกเว้นคุณกำลังทดสอบเน็ตเวิร์กของคุณ
inetd/cmsd	inetd	/etc/inetd.conf	เซอร์วิสปฏิทิน (ดังที่ใช้โดย CDE)	• รันเป็น root ดังนั้นจึงต้องกังวลเรื่องความปลอดภัย • ปิดใช้งานยกเว้นคุณจำเป็นต้องใช้เซอร์วิสนี้กับ CDE • ปิดใช้งานเซิร์ฟเวอร์ฐานข้อมูล back room
inetd/comsat	inetd	/etc/inetd.conf	แจ้งเมื่อมีจดหมายอิเล็กทรอนิกส์ขาเข้า	• รันเป็น root ดังนั้นจึงต้องกังวลเรื่องความปลอดภัย • ไม่ค่อยจำเป็น • ปิดใช้งาน
inetd/daytime	inetd	/etc/inetd.conf	เซอร์วิสเวลาเก่า (การทดสอบเท่านั้น)	• รันเป็น root • มีใช้เป็นเซอร์วิส TCP และ UDP • เปิดโอกาสสำหรับการโจมตี Denial of Service PING • เซอร์วิสเก่าและใช้เพื่อทดสอบเท่านั้น • ปิดใช้งาน

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
inetd/discard	inetd	/etc/inetd.conf	/dev/null service (การทดสอบเท่านั้น)	• มีใช้เป็นเซอร์วิส TCP และ UDP • ใช้ในการโจมตี Denial of Service • เซอร์วิสเก่าและใช้เพื่อทดสอบเท่านั้น • ปิดใช้งาน
inetd/dtspc	inetd	/etc/inetd.conf	CDE Subprocess Control	• เซอร์วิสนี้ถูกเริ่มทำงานโดยอัตโนมัติโดย **inetd** daemon ตามที่ไคลเอ็นต์ CDE ร้องขอให้กระบวนการเริ่มทำงาน บนโฮสต์ของ daemon นี้ทำให้เกิดจุดอ่อนสำหรับการโจมตี • ปิดใช้งานเซิร์ฟเวอร์ back room ที่ไม่มี CDE • CDE อาจสามารถทำงานโดยไม่มีเซอร์วิสนี้ • ปิดใช้งานยกเว้นจำเป็นต้องใช้อย่างแท้จริง
inetd/echo	inetd	etc/inetd.conf	เซอร์วิส echo (การทดสอบเท่านั้น)	• มีใช้เป็นเซอร์วิส UDP และ TCP • สามารถใช้ในการโจมตี Denial of Service หรือ Smurf • ใช้เพื่อ echo ที่บุคคลบางคนเพื่อผ่านไฟร์วอลล์หรือเริ่มทำงาน datastorm • ปิดใช้งาน
inetd/exec	inetd	/etc/inetd.conf	เซอร์วิสการทำงานรีโมต	• รันเป็นผู้ใช้ root • จำเป็นต้องให้คุณป้อน ID ผู้ใช้ และรหัสผ่าน ซึ่งถูกส่งแบบ ไม่มีการป้องกัน • เซอร์วิสนี้เป็นไปได้สูงที่จถูกพิจารณาว่ากำลังถูกสอดแนม • ปิดใช้งาน
inetd/finger	inetd	/etc/inetd.conf	finger peeking ที่ผู้ใช้	• รันเป็นผู้ใช้ root • ให้ข้อมูลเกี่ยวกับระบบและผู้ใช้ของคุณ • ปิดใช้งาน
inetd/ftp	inetd	/etc/inetd.conf	file transfer protocol	• รันเป็นผู้ใช้ root • ID ผู้ใช้และรหัสผ่านถูกถ่ายโอนแบบไม่มีการป้องกัน ดังนั้นอาจทำให้ ถูกสอดแนม • ปิดใช้งานเซอร์วิสนี้และใช้ชุดเชลล์การป้องกันความปลอดภัยพับลิกโดเมน

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
inetd/imap2	inetd	/etc/inetd.conf	Internet Mail Access Protocol	• ทำให้แน่ใจว่าคุณกำลังใช้เวอร์ชันล่าสุดของเซิร์ฟเวอร์นี้ • จำเป็นต่อเมื่อคุณกำลังทำงานเมล์เซิร์ฟเวอร์ มิฉะนั้น ให้ปิดใช้งาน • ID ผู้ใช้และรหัสผ่านถูกส่งแบบไม่มีการป้องกัน
inetd/klogin	inetd	/etc/inetd.conf	ล็อกอิน Kerberos	• ถูกเปิดใช้งานถ้าไซต์ของคุณใช้การพิสูจน์ตัวตน Kerberos
inetd/kshell	inetd	/etc/inetd.conf	เชลล์ Kerberos	• ถูกเปิดใช้งานถ้าไซต์ของคุณใช้การพิสูจน์ตัวตน Kerberos
inetd/login	inetd	/etc/inetd.conf	เซอร์วิส rlogin	• อาจถูกสงสัยว่ามีการปลอมแปลง IP spoofing, การปลอมแปลง DNS • ข้อมูล รวมถึง ID ผู้ใช้และรหัสผ่านถูกส่งแบบไม่มีการป้องกัน • รันเป็นผู้ใช้ root • ใช้เชลล์แบบปลอดภัยแทนเซอร์วิสนี้
inetd/netstat	inetd	/etc/inetd.conf	การรายงานสถานะเน็ตเวิร์กปัจจุบัน	• อาจให้ข้อมูลเน็ตเวิร์กที่สำคัญแก่แฮกเกอร์ถ้ารันบน ระบบของคุณ • ปิดใช้งาน
inetd/ntalk	inetd	/etc/inetd.conf	อนุญาตให้ผู้ใช้คุยกับบุคคลอื่น	• รันเป็นผู้ใช้ root • ไม่จำเป็นบนเซิร์ฟเวอร์ใช้งานจริง หรือ back room • ปิดใช้งานยกเว้นจำเป็นต้องใช้อย่างแท้จริง
inetd/pcnfsd	inetd	/etc/inetd.conf	เซอร์วิสไฟล์ PC NFS	• ปิดใช้งานเซอร์วิสถ้าไม่ได้ใช้งานในขณะนี้ • ถ้าคุณต้องการใช้เซอร์วิสที่คล้ายกับเซอร์วิสนี้ ให้พิจารณาใช้ Samba เป็น pcnfsd daemon มีวันที่ล่วงหน้ารีลีสข้อกำหนดคุณลักษณะ SMB ของ Microsoft

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
inetd/pop3	inetd	/etc/linetd.conf	Post Office Protocol	• ID ผู้ใช้และรหัสผ่านถูกส่งแบบไม่มีการป้องกัน • จำเป็นต่อเมื่อระบบของคุณเป็นเมลเซิร์ฟเวอร์ และคุณมีไคลเอ็นต์ ที่กำลังใช้แอ็พพลิเคชันที่สนับสนุน POP3 เท่านั้น • ถ้าไคลเอ็นต์ของคุณใช้ IMAP ให้ใช้เซอร์วิสนั้นแทน หรือใช้เซอร์วิส POP3s เซอร์วิสนี้มี Secure Socket Layer (SSL) tunnel • ปิดใช้งานถ้าคุณไม่ได้ทำงานเมลเซิร์ฟเวอร์ หรือไม่มีไคลเอ็นต์ที่จำเป็นต้อง ใช้เซอร์วิส POP
inetd/rexd	inetd	/etc/inetd.conf	การทำงานรีโมต	• รันเป็นผู้ใช้ root • เพียร์กับคำสั่ง **on** • ปิดใช้งานเซอร์วิส • ใช้ **rsh** และ **rshd** แทน
inetd/quotad	inetd	/etc/inetd.conf	รายงานโควต้าไฟล์ (สำหรับไคลเอ็นต์ NFS)	• จำเป็นต่อเมื่อคุณกำลังทำงานเซอร์วิสไฟล์ NFS • ปิดใช้งานเซอร์วิสนี้ยกเว้นจำเป็นต้องให้คำตอบ คำสั่ง **quota** • ถ้าคุณจำเป็นต้องใช้เซอร์วิสนี้ ให้ใช้แพ็ตช์และโปรแกรมแก้ไขทั้งหมด ล่าสุดสำหรับเซอร์วิสนี้
inetd/rstatd	inetd	/etc/inetd.conf	Kernel Statistics Server	• ถ้าคุณจำเป็นต้องมอนิเตอร์ระบบ ใช้ SNMP และปิดใช้งานเซอร์วิสนี้ • จำเป็นสำหรับการใช้คำสั่ง **rup**
inetd/rusersd	inetd	/etc/inetd.conf	ข้อมูลเกี่ยวกับการล็อกอินของผู้ใช้	• นี้ไม่ใช่เซอร์วิสที่จำเป็น ปิดใช้งาน • รันเป็นผู้ใช้ root • ให้รายการผู้ใช้ปัจจุบันบนระบบของคุณ และเพียร์กับ rusers
inetd/rwalld	inetd	/etc/inetd.conf	เขียนไปยังผู้ใช้ทั้งหมด	• รันเป็นผู้ใช้ root • ถ้าระบบของคุณให้บริการแบบโต้ตอบ คุณอาจต้องคง เซอร์วิสนี้ไว้ • ถ้าระบบของคุณเป็นระบบใช้งานจริง หรือเซิร์ฟเวอร์ฐานข้อมูล ไม่จำเป็นต้องใช้เซอร์วิสนี้ • ปิดใช้งาน

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
inetd/shell	inetd	/etc/inetd.conf	เซอร์วิส rsh	• ปิดใช้งานเซอร์วิสนี้ถ้าทำได้ใช้ Secure Shell แทน • ถ้าคุณต้องใช้เซอร์วิสนี้ ใช้ TCP Wrapper เพื่อหยุดการปลอมแปลง และจำกัดการเปิดเผย • จำเป็นสำหรับโปรแกรมการแจกจ่ายซอฟต์แวร์**Xhier**
inetd/sprayd	inetd	/etc/inetd.conf	การทดสอบ RPC spray	• รันเป็นผู้ใช้ root • อาจจำเป็นสำหรับการวินิจฉัยปัญหาเน็ตเวิร์ก NFS • ปิดใช้งานถ้าคุณไม่ได้ทำงาน NFS
inetd/systat	inetd	/etc/inted.conf	รายงานสถานะ "ps -ef"	• อนุญาตให้ไซต์รีโมตเห็นสถานะของกระบวนการบนระบบของคุณ • เซอร์วิสนี้ถูกปิดใช้งานโดยดีฟอลต์ คุณต้องตรวจสอบเป็นระยะ เพื่อให้แน่ใจว่าเซอร์วิสไม่ถูกเปิดใช้งาน
inetd/talk	inetd	/etc/inetd.conf	สร้างหน้าจอแยกระหว่างผู้ใช้ 2 คนบนเน็ต	• ไม่ใช่เซอร์วิสที่จำเป็น • ใช้กับคำสั่ง **talk** • จัดให้มีเซอร์วิส UDP ที่พอร์ต 517 • ปิดใช้งานยกเว้นคุณต้องการเซสชันการคุยแบบโต้ตอบหลายเซสชันสำหรับผู้ใช้ UNIX
inetd/ntalk	inetd	/etc/inetd.conf	"new talk" สร้างหน้าจอแยกระหว่างผู้ใช้ 2 คน บนเน็ต	• ไม่ใช่เซอร์วิสที่จำเป็น • ใช้กับคำสั่ง **talk** • จัดให้มีเซอร์วิส UDP ที่พอร์ต 517 • ปิดใช้งานยกเว้นคุณต้องการเซสชันการคุยแบบโต้ตอบหลายเซสชันสำหรับผู้ใช้ UNIX
inetd/telnet	inetd	/etc/inetd.conf	เทลเน็ต telnet	• สนับสนุนเซสชันล็อกอินรีโมต แต่รหัสผ่านและ ID ถูกส่ง แบบไม่มีการป้องกัน • ถ้าเป็นไปได้ ให้ปิดใช้งานเซอร์วิสนี้และใช้ Secure Shell สำหรับ การเข้าถึงรีโมตแทน
inetd/tftp	inetd	/etc/inetd.conf	การถ่ายโอนไฟล์ trivial	• จัดให้มีเซอร์วิส UDP ที่พอร์ต 69 • รันเป็นผู้ใช้ root และอาจมีช่องโหว่ • ใช้โดย NIM • ปิดใช้งานยกเว้นคนกำลังใช้ NIM หรือต้องบูตเวิร์กสเตชันแบบไร้ดิสก์

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
inetd/time	inetd	/etc/inetd.conf	เซอร์วิสเวลาเก่า	• ฟังก์ชันภายในของ **inetd** ที่ใช้โดยคำสั่ง **rdate** • มีใช้เป็นเซอร์วิส TCP และ UDP • บางครั้งใช้เพื่อซิงโครไนซ์นาฬิกาตอนบูตเครื่อง • เซอร์วิสล้าสมัย ใช้ ntpdate แทน • ปิดใช้งานนี้ต่อหลังจากคุณได้ทดสอบระบบของคุณ (บูต/รีบูต) โดยที่เซอร์วิสนี้ปิดใช้งนและไม่พบปัญหาใดๆ
inetd/ttdbserver	inetd	/etc/inetd.conf	เซิร์ฟเวอร์ฐานข้อมูล tool-talk (สำหรับ CDE)	• **rpc.ttdbserverd** รันเป็นผู้ใช้ root และอาจมีช่องโหว่ • แจ้งเป็นเซอร์วิสที่จำเป็นสำหรับ CDE แต่ CDE สามารถทำงานได้โดยไม่ต้องใช้เซอร์วิสนี้ • ไม่ควรรันบนเซิร์ฟเวอร์ back room หรือระบบใดๆ ที่ต้องคำนึงเรื่อง ความปลอดภัย
inetd/uucp	inetd	/etc/inetd.conf	เน็ตเวิร์ก UUCP	• ปิดใช้งานยกเว้นคุณมีแอ็พพลิเคชันที่ใช้ UUCP
inittab/dt	init	/etc/rc.dt script in the /etc/inittab	เดสก์ท็อปล็อกอินไปยังสภาวะแวดล้อม CDE	• เริ่มทำงานเซิร์ฟเวอร์ X11 บนคอนโซล • สนับสนุน X11 Display Manager Control Protocol (xdcmp) เพื่อที่ สเตชัน X11 อื่นสามารถล็อกเข้าสู่เครื่องเดียวกัน • เซอร์วิสควรใช้บนเวิร์กสเตชันส่วนบุคคลเท่านั้น หลีกเลี่ยงการใช้ สำหรับระบบ back room
inittab/dt_nogb	init	/etc/inittab	เดสก์ท็อปล็อกอินไปยังสภาวะแวดล้อม CDE (ไม่มีกราฟิกบูต)	• ไม่แสดงกราฟิกจนกว่าระบบจะทำงานสมบูรณ์ • ข้อควรคำนึงเหมือน inittab/dt
inittab/httpdlite	init	/etc/inittab	เว็บเซิร์ฟเวอร์สำหรับคำสั่ง **docsearch**	• ดีฟอลต์เว็บเซิร์ฟเวอร์สำหรับเอ็นจิน docsearch • ปิดใช้งานยกเว้นเครื่องของคุณเป็นเซิร์ฟเวอร์เอกสารคู่มือ
inittab/i4ls	init	/etc/inittab	เซิร์ฟเวอร์ตัวจัดการไลเซนส์	• เปิดใช้งานสำหรับเครื่องที่ใช้ในการพัฒนา • ปิดใช้งานสำหรับเครื่องใช้งานจริง • เปิดใช้งานสำหรับคอมพิวเตอร์ฐานข้อมูล back room ที่จำเป็นต้องใช้ไลเซนส์ • ให้การสนับสนุนคอมไพลเลอร์ ซอฟต์แวร์ฐานข้อมูล หรือผลิตภัณฑ์ที่มีไลเซนส์ อื่นใด

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
inittab/imqss	init	/etc/inittab	กลไกค้นหาสำหรับ ″docsearch″	• ส่วนของดีฟอลต์เว็บเซิร์ฟเวอร์สำหรับเอ็นจิน docsearch • ปิดใช้งานยกเว้นเครื่องของคุณเป็นเซิร์ฟเวอร์เอกสารคู่มือ
inittab/lpd	init	/etc/inittab	ส่วนการติดต่อพรินเตอร์รายบรรทัด BSD	• ยอมรับงานพิมพ์จากระบบอื่น • คุณสามารถปิดใช้งานเซอร์วิสนี้และยังคงส่งงานไปยังพรินต์เซิร์ฟเวอร์ได้ • ปิดใช้งานหลังจากคุณยืนยันว่าไม่ส่งผลต่อการพิมพ์
inittab/nfs	init	/etc/inittab	Network File System/Net Information Services	• เซอร์วิส NFS และ NIS ที่ขึ้นกับสิ่งที่เกิดขึ้นบน UDP/RPC • การพิสูจน์ตัวตนน้อยที่สุด • ปิดใช้งานสำหรับเครื่อง back room
inittab/piobe	init	/etc/inittab	พรินเตอร์ IO Back End (สำหรับการพิมพ์)	• จัดการการจัดตารางนัดหมาย การสพูล และการพิมพ์งานที่ส่ง โดย **qdaemon** daemon • ปิดใช้งานถ้าคุณไม่พิมพ์จากระบบของคุณ เนื่องจากคุณกำลังส่ง งานพิมพ์ไปยังเซิร์ฟเวอร์
inittab/qdaemon	init	/etc/inittab	daemon คิว (สำหรับการพิมพ์	• ส่งงานพิมพ์ไปยัง **piobe** daemon • ถ้าคุณไม่ได้พิมพ์จากระบบของคุณ ให้ปิดใช้งาน
inittab/uprintfd	init	/etc/inittab	ข้อความเคอร์เนล	• โดยทั่วไปไม่จำเป็น • ปิดใช้งาน
inittab/writesrv	init	/etc/inittab	เขียนบันทึกย่อยไปยัง ttys	• ใช้โดยผู้ใช้เวิร์กสเตชัน UNIX แบบโต้ตอบเท่านั้น • ปิดใช้งานเซอร์วิสนี้สำหรับเซิร์ฟเวอร์ฐานข้อมูล back room และเครื่องที่ใช้ เพื่อการพัฒนา • เปิดใช้งานเซอร์วิสนี้สำหรับเวิร์กสเตชัน
inittab/xdm	init	/etc/inittab	traditional X11 Display Management	• ไม่รันบนเครื่อง back room ใช้งานจริง หรือเซิร์ฟเวอร์ฐานข้อมูล • ไม่รันบนระบบที่ใช้พัฒนา ยกเว้นจำเป็นต้องใช้การจัดการแสดงผล X11 • ยอมรับให้รันบนเวิร์กสเตชันได้ถ้าจำเป็นต้องใช้กราฟิก

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
rc.nfs/automountd		/etc/rc.nfs	ระบบไฟล์อัตโนมัติ	• ถ้าคุณใช้ NFS เปิดใช้งานเซอร์วิสนี้สำหรับเวิร์กสเตชัน • ไม่ใช้ automounter สำหรับเซิร์ฟเวอร์การพัฒนาหรือ back room
rc.nfs/biod		/etc/rc.nfs	Block IO Daemon (จำเป็นสำหรับเซิร์ฟเวอร์ NFS)	• ถูกเปิดใช้งานสำหรับเซิร์ฟเวอร์ NFS เท่านั้น • ถ้าไม่ใช่เซิร์ฟเวอร์ NFS ให้ปิดใช้งานนี้พร้อมกับ nfsd และ rpc.mountd
rc.nfs/keyserv		/etc/rc.nfs	เซิร์ฟเวอร์ Secure RPC Key	• จัดการคีย์ที่จำเป็นสำหรับ secure RPC • ปิดใช้งานถ้าคุณ *ไม่* ใช้ NFS และ NIS
rc.nfs/nfsd		/etc/rc.nfs	เซอร์วิส NFS (จำเป็นสำหรับเซิร์ฟเวอร์ NFS)	• การพิสูจน์ตัวตนอ่อนแอ • อาจส่งผลให้การหยุดทำงานสแต็กเฟรม • เปิดใช้งานถ้าเป็นบนไฟล์เซิร์ฟเวอร์ NFS • ถ้าคุณปิดใช้งานเซอร์วิสนี้ ให้ปิดใช้งาน **biod, nfsd** และ **rpc.mountd** เช่นกัน
rc.nfs/rpc.lockd		/etc/rc.nfs	การล็อกไฟล์ NFS	• ปิดใช้งานถ้าคุณไม่ได้ใช้ NFS • ปิดใช้งานถ้าคุณไม่ได้ใช้การล็อกไฟล์ในเน็ตเวิร์ก • **lockd** daemon ถูกกล่าวถึงใน SANS Top Ten Security Threats
rc.nfs/rpc.mountd		/etc/rc.nfs	การเมาต์ไฟล์ NFS (จำเป็นสำหรับเซิร์ฟเวอร์ NFS)	• การพิสูจน์ตัวตนอ่อนแอ • อาจส่งผลให้การหยุดทำงานสแต็กเฟรม • ควรถูกเปิดใช้งานบนไฟล์เซิร์ฟเวอร์ NFS เท่านั้น • ถ้าคุณปิดใช้งานเซอร์วิสนี้ ให้ปิดใช้งาน **biod** และ **nfsd** เช่นกัน
rc.nfs/rpc.statd		/etc/rc.nfs	การล็อกไฟล์ NFS (เพื่อกู้คืน)	• นำการล็อกไฟล์ไปใช้ใน NFS • ปิดใช้งานยกเว้นคุณกำลังใช้ NFS
rc.nfs/rpc.yppasswdd		/etc/rc.nfs	NIS password daemon (สำหรับ NIS มาสเตอร์)	• ใช้เพื่อดำเนินการไฟล์รหัสผ่านโลคัล • จำเป็นต่อเมื่อเครื่องที่สงสัยเป็น NIS มาสเตอร์ ในกรณีอื่นทั้งหมดให้ปิดใช้งาน
rc.nfs/ypupdated		/etc/rc.nfs	NIS Update daemon (สำหรับ NIS slave)	• รับแม็พฐานข้อมูล NIS ที่ส่งจาก NIS Master • จำเป็นต้องใช้เมื่อเครื่องที่สงสัยเป็น NIS slave สำหรับ Master NIS Server
rc.tcpip/autoconf6		/etc/rc.tcpip	อินเตอร์เฟส IPv6	• ปิดใช้งานยกเว้นคุณกำลังรัน IP Version 6

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
rc.tcpip/dhcpcd		/etc/rc.tcpip	Dynamic Host Configure Protocol (ไคลเอ็นต์)	• เซิร์ฟเวอร์ Back room ไม่ควรขึ้นกับ DHCP ปิดใช้งานเซอร์วิสนี้ • ถ้าโฮสต์ของคุณไม่ได้ใช้ DHCP ให้ปิดใช้งาน
rc.tcpip/dhcprd		/etc/rc.tcpip	Dynamic Host Configure Protocol (relay	• จับ DHCP broadcasts และส่งไปยังเซิร์ฟเวอร์บนเน็ตเวิร์กอื่น • ทำสำเนาเซอร์วิสที่พบบนเราเตอร์ • ปิดใช้งานถ้าคุณไม่ได้ใช้ DHCP หรือขึ้นกับการส่งข้อมูล ระหว่างเน็ตเวิร์ก
rc.tcpip/dhcpsd		/etc/rc.tcpip	Dynamic Host Configure Protocol (เซิร์ฟเวอร์	• ตอบการร้องขอ DHCP จากไคลเอ็นต์ตอนบูตเครื่อง ให้ข้อมูล ไคลเอ็นต์ เช่น IP name หมายเลข netmask เราเตอร์ และ broadcast address • ปิดใช้งานถ้าคุณไม่ได้ใช้ DHCP • ถูกปิดใช้งานบนเซิร์ฟเวอร์ใช้งานจริง และ back room และโฮสต์ ไม่ได้ใช้ DHCP
rc.tcpip/dpid2		/etc/rc.tcpip	เซอร์วิส SNMP ที่ล้าสมัย	• ปิดใช้งานยกเว้นคุณต้องใช้ SNMP
rc.tcpip/gated		/etc.rc.tcpip	gated routing ระหว่างอินเตอร์เฟส	• จำลองฟังก์ชันเราเตอร์ • ปิดใช้งานเซอร์วิสนี้และใช้ RIP หรือเราเตอร์แทน
rc.tcpip/inetd		/etc/rc.tcpip	เซอร์วิส inetd	• ระบบที่มีความปลอดภัยอย่างเต็มที่ควรปิดใช้งานเซอร์วิสนี้ แต่ มักไม่ค่อยทำในเชิงปฏิบัติ • การปิดใช้งานเซอร์วิสนี้จะปิดใช้งานเซอร์วิสรีโมตเชลล์ซึ่งจำเป็นสำหรับ บางเมล์เซิร์ฟเวอร์และเว็บเซิร์ฟเวอร์
rc.tcpip/mrouted		/etc/rc.tcpip	การจัดเส้นทางมัลติคาสต์	• อีมูเลตฟังก์ชันเราเตอร์ของการส่งแพ็กเก็ตมัลติคาสต์ระหว่าง เน็ตเวิร์กเซ็กเมนต์ • ปิดใช้งานเซอร์วิสนี้ ใช้เราเตอร์แทน
rc.tcpip/names		/etc/rc.tcpip	เซิร์ฟเวอร์ชื่อ DNS	• ใช้เซอร์วิสนี้ต่อเมื่อเครื่องของคุณเป็นเซิร์ฟเวอร์ชื่อ DNS • ปิดใช้งานสำหรับเวิร์กสเตชัน เครื่องสำหรับการพัฒนาและเครื่องใช้งานจริง
rc.tcpip/ndp-host		/etc/rc.tcpip	โฮสต์ IPv6	• ปิดใช้งานยกเว้นคุณใช้ IP Version 6
rc.tcpip/ndp-router		/etc/rc.tcpip	การจัดเส้นทาง IPv6	• ปิดใช้งานนี้ยกเว้นคุณใช้ IP Version 6 พิจารณาใช้เราเตอร์ แทน IP Version 6

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
rc.tcpip/portmap		/etc/rc.tcpip	เซอร์วิส RPC	• เซอร์วิสที่จำเป็น • เซิร์ฟเวอร์ RPC เรจิสเตอร์กับ **portmap** daemon ไคลเอ็นต์ที่ต้อง ค้นหาเซอร์วิส RPC จะขอให้ **portmap** daemon บอกให้ทราบ ว่าเซอร์วิสที่ต้องการอยู่ที่ใด • ปิดใช้งานต่อเมื่อคุณต้องการลดเซอร์วิส RPC เพื่อ ให้คงเหลือเฉพาะ **portmap**
rc.tcpip/routed		/etc/rc.tcpip	การจัดเส้นทาง RIP ระหว่างอินเตอร์เฟส	• จำลองฟังก์ชันเราเตอร์ • ปิดใช้งานถ้าคุณมีเราเตอร์สำหรับแพ็กเก็ตระหว่างเน็ตเวิร์ก
rc.tcpip/rwhod		/etc/rc.tcpip	″who″ daemon รีโมต	• รวบรวมและกระจายข้อมูลไปยังเพียร์เซิร์ฟเวอร์บนเน็ตเวิร์กเดียวกัน • ปิดใช้งานเซอร์วิสนี้
rc.tcpip/sendmail		/etc/rc.tcpip	เมล์เซอร์วิส	• รันเป็นผู้ใช้ root • ปิดใช้งานเซอร์วิสนี้ยกเว้นเครื่องของใช้เป็นเมล์เซิร์ฟเวอร์ • ถ้าปิดใช้งาน ให้ทำอย่างหนึ่งอย่างใดต่อไปนี้: – Place รายการใน crontab เพื่อลบคิวให้ว่าง ใช้คำสั่ง **/usr/lib/sendmail -q** – ตั้งค่าเซอร์วิส DNS เพื่อที่เมล์สำหรับเซิร์ฟเวอร์ของคุณถูกส่ง ไปยังระบบอื่นได้
rc.tcpip/snmpd		/etc/rc.tcpip	Simple Network Management Protocol	• ปิดใช้งานถ้าคุณไม่ได้มอนิเตอร์ระบบผ่านเครื่องมือ SNMP • SNMP จำเป็นต้องใช้บนเซิร์ฟเวอร์ที่วิกฤต
rc.tcpip/syslogd		/etc/rc.tcpip	บันทึกการทำงานระบบของเหตุการณ์	• การปิดใช้งานเซอร์วิสนี้ *ไม่* แนะนำ • มีแนวโน้มเป็นการโจมตี denial of service • จำเป็นในบางระบบ
rc.tcpip/timed		/etc/rc.tcpip	Old Time Daemon	• ปิดใช้งานเซอร์วิสนี้และใช้ xntp แทน
rc.tcpip/xntpd		/etc/rc.tcpip	New Time Daemon	• คงนาฬิกาบนระบบให้ซิงค์กัน • ปิดใช้งานเซอร์วิสนี้ • ตั้งค่าระบบอื่นเป็นเซิร์ฟเวอร์เวลา และให้ระบบอื่นๆ ซิงโครไนซ์กับเซิร์ฟเวอร์เวลาด้วยงาน cron ที่เรียกใช้ ntpdate
dt login		/usr/dt/config/Xaccess	CDE ที่ไม่จำกัด	• ถ้าคุณไม่ได้ให้ล็อกอิน CDE แก่กลุ่มของ X11 stations คุณสามารถจำกัด dtlogin ไปที่คอนโซล

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
เซอร์วิส FTP แบบไม่ระบุชื่อ		user rmuser -p <username>	ftp แบบไม่ระบุชื่อ	• ความสามารถ FTP แบบไม่ระบุชื่อป้องกันไม่ให้คุณติดตั้งการใช้งาน FTP สำหรับผู้ใช้เฉพาะ • ลบ ftp ผู้ใช้ถ้ามีบัญชีผู้ใช้นั้นอยู่ ดังนี้: **rmuser -p ftp** • ความปลอดภัยอื่นๆ สามารถได้จากการใช้ค่าจากไฟล์ `/etc/ftpusers` ที่มีรายชื่อบุคคลที่ไม่ควรสามารถ ftp ระบบของคุณ
การเขียน FTP แบบไม่ระบุชื่อ			การอัพโหด ftp แบบไม่ระบุชื่อ	• ไม่มีไฟล์ใดเป็นของ ftp • การอัพโหลดแบบไม่ระบุชื่อของ FTP อนุญาตให้โค้ดที่มีการทำงานที่ไม่ถูกต้องถูกใส่ในระบบของคุณ • ใส่ชื่อของผู้ใช้ที่คุณไม่อนุญาตลงในไฟล์ `/etc/ftpusers` • บางตัวอย่างของผู้ใช้ที่ระบบสร้างที่คุณอาจไม่อนุญาต ให้ใช้การอัพโหลดแบบไม่ระบุชื่อทาง FTP ไปยังระบบของคุณ: root, daemon, bin.sys, admin.uucp, guest, nobody, lpd, nuucp, ladp • เปลี่ยนสิทธิเจ้าของและกลุ่มในไฟล์ `ftpusers` ดังนี้: `chown root:system /etc/ftpusers` • เปลี่ยนสิทธิในไฟล์ `ftpusers` ให้มีค่าติดตั้งที่เข้มงวดมากขึ้นดังนี้: `chmod 644 /etc/ftpusers`
ftp.restrict			ftp ไปยังบัญชีผู้ใช้ระบบ	• ห้ามผู้ใช้จากภายนอกได้รับอนุญาตให้แทนที่ไฟล์ root โดยใช้ไฟล์ `ftpusers`
root.access		`/etc/security/user`	rlogin/telnet ไปยังบัญชีผู้ใช้ root	• ตั้งค่าอ็อพชัน rlogin ในไฟล์ `etc/security/user` เป็นเท็จ • บุคคลใดทีล็อกอินเป็น root อันดับแรกควรล็อกอินโดยใช้ชื่อตนเอง จากนั้น **su** เป็น root วิธีนี้จะช่วยให้มีหลักฐานการตรวจสอบ!
snmpd.readWrite		`/etc/snmpd.conf`	SNMP readWrite communities	• ถ้าคุณ *ไม่* ใช้ SNMP ให้ปิดใช้งาน SNMP daemon • ปิดใช้งาน community ไพรเวตและ community ระบบในไฟล์ `/etc/snmpd.conf` • จำกัด 'public' community แก่ IP แอดเดรสเหล่านั้นที่กำลังมอนิเตอร์ ระบบของคุณ

เซอร์วิส	Daemon	เริ่มทำงานโดย	ฟังก์ชัน	หมายเหตุ
syslog.conf			กำหนดคอนฟิก syslogd	• ถ้าคุณไม่ได้ตั้งค่า /etc/syslog.conf ให้ปิดใช้งาน daemon นี้ • ถ้าคุณกำลังใช้ syslog.conf เพื่อบันทึกข้อความระบบให้คงเปิดใช้งานไว้

ข้อสรุปของอ็อพชันเน็ตเวิร์กเซอร์วิส

ในการประสบความสำเร็จด้านการรักษาความปลอดภัยระบบในระดับที่สูงขึ้นไป มีอ็อพชันเน็ตเวิร์กมากมายที่คุณสามารถเปลี่ยนแปลงโดยใช้ 0 เพื่อปิดใช้งานและ 1 เพื่อ เปิดใช้งาน รายการต่อไปนี้ระบุพารามิเตอร์ต่างๆ ที่คุณสามารถใช้กับคำสั่ง **no**

พารามิเตอร์	คำสั่ง	วัตถุประสงค์
bcastping	/usr/sbin/no -o bcastping=0	อนุญาตให้ตอบกลับแพ็กเก็ต ICMP echo ไปยังแอดเดรส การกระจาย การปิดใช้งานสิ่งนี้จะป้องกันการโจมตี Smurf
clean_partial_conns	/usr/sbin/no -o clean_partial_conns=1	ระบุว่าการโจมตี SYN (ซิงโครไนซ์หมายเลขลำดับ) จะถูกหลีกเลี่ยงหรือไม่
directed_broadcast	/usr/sbin/no -o directed_broadcast=0	ระบุว่าจะอนุญาตการกระจายโดยตรงไปยังเกตเวย์หรือไม่ การตั้งค่า 0 ช่วยป้องกันแพ็กเก็ตโดยตรงมิให้ไปถึงรีโมต เน็ตเวิร์ก
icmpaddressmask	/usr/sbin/no -o icmpaddressmask=0	ระบุว่าะรบบตอบกลับการร้องขอมาสก์ ICMP address หรือไม่ การปิดใช้งานนี้ป้องกันการเข้าถึงผ่านการโจมตีโดยการกำหนดเส้นทางซอร์ส
ipforwarding	/usr/sbin/no -o ipforwarding=0	ระบุว่าเคอร์เนลควรส่งต่อแพ็กเก็ตหรือไม่ การปิดใช้งานนี้ป้องกันมิให้เน็ตเวิร์กที่ถูกกำหนดเส้นทางมิให้ไปยังรีโมตเน็ตเวิร์กที่ไปถึง
ipignoreredirects	/usr/sbin/no -o ipignoreredirects=1	ระบุว่าประมวลผลการเปลี่ยนเส้นทางที่ได้รับหรือไม่
ipsendredirects	/usr/sbin/no -o ipsendredirects=0	ระบุว่าเคอร์เนลควรส่งสัญญาณการเปลี่ยนเส้นทาง การปิดใช้งานนี้ป้องกันมิให้เน็ตเวิร์กที่ถูกกำหนดเส้นทางมิให้ไปยังรีโมตเน็ตเวิร์กที่ไปถึง
ip6srcrouteforward	/usr/sbin/no -o ip6srcrouteforward=0	ระบุว่าระบบส่งต่อแพ็กเก็ต IPv6 ที่กำหนดเส้นทาง โดยต้นทางหรือไม่ การปิดใช้งานนี้ป้องกันการเข้าถึงผ่านการโจมตีโดยการกำหนดเส้นทางซอร์ส
ipsrcrouteforward	/usr/sbin/no -o ipsrcrouteforward=0	ระบุว่าระบบส่งต่อแพ็กเก็ตที่กำหนดเส้นทาง โดยต้นทางหรือไม่ การปิดใช้งานนี้ป้องกันการเข้าถึงผ่านการโจมตีโดยการกำหนดเส้นทางซอร์ส
ipsrcrouterecv	/usr/sbin/no -o ipsrcrouterecv=0	ระบุว่าระบบจะยอมรับแพ็กเก็ตที่กำหนดเส้นทางโดยซอร์สหรือไม่ การปิดใช้งานนี้ป้องกันการเข้าถึงผ่านการโจมตีโดยการกำหนดเส้นทางซอร์ส

พารามิเตอร์	คำสั่ง	วัตถุประสงค์
ipsrcroutesend	/usr/sbin/no -o ipsrcroutesend=0	ระบุว่าแอ็พพลิเคชันสามารถส่งแพ็กเก็ตที่กำหนดเส้นทางโดย ซอร์ส การปิดใช้งานนี้ป้องกันการเข้าถึงผ่านการโจมตีโดยการกำหนดเส้นทางซอร์ส
nonlocsroute	/usr/sbin/no -o nonlocsrcroute=0	แจ้ง Internet Protocol ที่แพ็กเก็ตที่กำหนดเส้นทางโดยซอร์ส โดยจำกัดอาจถูกกำหนดแอดเดรสไปยังโฮสต์ภายในโลคัลเน็ตเวิร์ก การปิดใช้งานนี้ ป้องกันการเข้าถึงผ่านการโจมตีโดยการกำหนดเส้นทางซอร์ส
tcp_icmpsecure	/usr/sbin/no -o tcp_icmpsecurer=1	ป้องกันการโจมตีการเชื่อมต่อ TCP กับ ICMP (Internet Control Message Protocol) source quench และ PMTUD (Path MTU Discovery) ตรวจสอบ ความหนาแน่นของข้อความ ICMP เพื่อทดสอบหมายเลขลำดับของส่วนหัว TCP ว่าอยู่ภายในช่วงของหมายเลขลำดับที่ยอมรับได้หรือไม่ ค่า: 0=ปิด (ดีฟอลต์); 1=เปิด
ip_nfrag	/usr/sbin/no -o ip_nfrag=200	ระบุจำนวนแฟรกเมนต์สูงสุดของแพ็กเก็ต IP ที่สามารถเก็บบนคิวที่รวม IP ขึ้นใหม่ณ เวลาหนึ่ง (ค่าดีฟอลต์ 200 สามารถเก็บได้สูงสุด 200 แฟรกเมนต์ของแพ็กเก็ต IP ในคิวที่รวม IP ขึ้นใหม่)
tcp_pmtu_discover	/usr/sbin/no -o tcp_pmtu_discover=0	การปิดใช้งานนี้ป้องกันการเข้าถึงผ่านการโจมตีโดยการกำหนด เส้นทางซอร์ส
tcp_tcpsecure	/usr/sbin/no -o tcp_tcpsecure=7	ป้องกันการเชื่อมต่อ TCP มิให้มีจุดอ่อน ค่า: 0=ไม่มีการป้องกัน; 1=กาส่ง SYN ลวงไปยังการเชื่อมต่อที่สร้างขึ้น; 2=การส่ง RST ลวงไปยังการเชื่อมต่อที่สร้างขึ้น; 3=การอัดข้อมูลใน การเชื่อมต่อ TCP ที่สร้างขึ้น; 5–7=การรวมของจุดอ่อนด้านบน
udp_pmtu_discover	/usr/sbin/no -o udp_pmtu_discover=0	เปิดใช้งานหรือปิดใช้งานการค้นหาพาธ MTU สำหรับแอ็พพลิเคชัน TCP การปิดใช้งานนี้ป้องกันการเข้าถึงผ่านการโจมตีโดยการกำหนดเส้นทางซอร์ส

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับอ็อพชันที่ปรับค่าเน็ตเวิร์กได้ ดูที่ *การจัดการประสิทธิภาพ*

Trusted AIX

Trusted AIX เปิดใช้งานความสามารถ Multi Level Security (MLS) ใน AIX

หมายเหตุ: MLS ยังหมายถึงการรักษาความปลอดภัยโดยใช้เลเบล

เทียบกับ AIX ปกติ การรักษาความปลอดภัยโดยใช้เลเบล Trusted AIX ใช้เลเบลสำหรับเรื่องและอ็อบเจ็กต์ทั้งหมดในระบบ

หมายเหตุ: ตัวเลือก Trusted AIX install เปิดใช้งานสภาวะแวดล้อม Labeled Security AIX ค่าควบคุมการเข้าใช้ในระบบใช้เลเบลที่จัดเตรียมสำหรับสภาวะแวดล้อม Multi Level Security (MLS) และมีการสนับสนุนดังต่อไปนี้:

- อ็อบเจ็กต์ที่เลเบล: ไฟล์ อ็อบเจ็กต์ IPC แพ็กเก็ตเน็ตเวิร์ก และ อ็อบเจ็กต์ที่เลเบลอื่น

- เลเบลพรินเตอร์
- เน็ตเวิร์กที่ไว้วางใจ: สนับสนุน RIPSO และ CIPSO ใน IPv4 และ IPv6

โปรดหมายเหตุว่าเมื่อคุณเลือกโหมดการติดตั้งนี้ คุณจะไม่สามารถกลับไปที่สภาวะแวดล้อม AIX ปกติโดยไม่ต้อง ทำการแทนที่การติดตั้งของ AIX ปกติ ประเมินความต้องการ ของคุณสำหรับสภาวะแวดล้อม Trusted AIX ก่อนเลือกโหมดการติดตั้งนี้ รายละเอียดเพิ่มเติมเกี่ยวกับ Trusted AIX อยู่ใน เอกสารที่เข้าถึงได้แบบพับลิก AIX

AIX มาตรฐานจัดเตรียม ชุดของคุณลักษณะความปลอดภัยเพื่อให้ผู้จัดการข้อมูลและผู้ดูแลระบบ ได้รับการรักษาความปลอดภัยระบบและเน็ตเวิร์กระดับพื้นฐาน คุณลักษณะการรักษาความปลอดภัย AIX มี ดังต่อไปนี้:

- ล็อกอินและรหัสผ่านที่ควบคุมการเข้าถึงระบบและเน็ตเวิร์ก
- สิทธิการเข้าถึง ผู้ใช้ กลุ่ม และไฟล์ world
- access control lists (ACLs)
- ระบบย่อยการตรวจสอบ
- Role Based Access Control (RBAC)

Trusted AIXสร้างจาก คุณลักษณะระบบปฏิบัติการ AIX หลักเพื่อเพิ่มและขยายการรักษาความปลอดภัย AIX เข้าสู่ระบบย่อยเน็ตเวิร์ก

Trusted AIXทำงานได้กับ AIX application programming interface (API) แอ็พพลิเคชันที่รันบน AIX สามารถรันบน Trusted AIX อย่างไรก็ตาม เนื่องจาก ข้อจำกัดความปลอดภัยเพิ่มเติม, แอ็พพลิเคชัน MLS-unaware อาจจำเป็นต้อง ใช้ privileges ในการดำเนินการในสภาวะแวดล้อม Trusted AIX คำสั่ง **tracepriv** สามารถถูกใช้เพื่อทำโปรไฟล์แอ็พพลิเคชันในสถานการณ์ดังกล่าว

Trusted AIX ขยาย AIX API เพื่อสนับสนุน การทำงานด้านความปลอดภัยเพิ่มเติม ซึ่งช่วยให้ผู้ใช้สามารถพัฒนา แอ็พพลิเคชันที่ปลอดภัยของตัวเอง สามารถถูกพัฒนาโดยใช้ AIX API และส่วนขยาย Trusted AIX ใหม่

Trusted AIX เปิดใช้ระบบ AIX ในการประมวลผลข้อมูล ที่หลายระดับความปลอดภัย ถูกออกแบบให้ตรงตามเกณฑ์ US Department of Defense (DoD) TCSEC และ European ITSEC สำหรับ enhanced B1 security

ดูที่ การรักษาความปลอดภัย ระบบปฏิบัติการฐาน และ การรักษาความปลอดภัย เน็ตเวิร์ก สำหรับข้อมูลการรักษาความปลอดภัย AIX มาตรฐาน

บทนำ Trusted AIX

Trusted AIX เพิ่ม ความปลอดภัยของระบบปฏิบัติการ AIX มาตรฐาน โดยจัดเตรียมความสามารถ label-based-security ภายใน ระบบปฏิบัติการ

สภาวะแวดล้อม Trusted AIX label-based สามารถถูกติดตั้งโดยเลือกตัวเลือกเวลาติดตั้ง ถ้าคุณติดตั้ง Trusted AIX คุณจะไม่สามารถกลับไปที่สภาวะแวดล้อม AIX ปกติโดยไม่ต้อง ทำการแทนที่การติดตั้งของ AIX ปกติ เมื่อติดตั้ง แล้ว สภาวะแวดล้อม Trusted AIX จะใช้กับระบบ AIX ทั้งหมด รวมถึง WPARs ที่สร้างภายในสภาวะแวดล้อม AIX ขณะที่การรักษาความปลอดภัยที่ใช้เลเบล (หรือเรียกว่า Multi Level Security หรือ MLS) ถูกใช้บ่อยครั้งในกิจการด้านการทหารและข่าวกรอง ยัง สามารถถูกใช้ในกิจการทางการค้าได้เช่นกัน ซึ่งทำได้โดย กำหนดเลเบลที่มีอยู่ใน Trusted AIX เอง การติดตั้ง Trusted AIX ใหม่จัดเตรียมสำหรับเลเบลที่ยึดตามมาตรฐาน MLS

สภาวะแวดล้อม Trusted AIX ประกอบด้วย AIX ปกติกับ แพ็กเกจและชุดไฟล์เพิ่มเติมบางส่วน นอกจากนี้ เคอร์เนลสวิตช์จะบังคับเคอร์เนลให้ทำงานในโหมด Trusted AIX เมื่อบูต ผ่าน ซีดีหรือดีวีดี ระบบบูตในสภาวะแวดล้อม AIX ปกติ เมื่อเมนูติดตั้งถูกแสดง โปรแกรมติดตั้งสามารถเลือกตัวเลือก Trusted AIX และเริ่ม การติดตั้งไฟล์ MLS-related เมื่อการติดตั้งสมบูรณ์โปรแกรมติดตั้ง ต้องเริ่มลำดับการบูตใหม่ครั้งแรก ระหว่างระดับการบูตใหม่ ครั้งแรก, Config Assistant จัดเตรียมเมนูสำหรับผู้ใช้ต่างๆ และผู้ใช้ ISSO, SA และ SO ถูกตั้งค่า จากนั้นระบบเสร็จสิ้น การดำเนินการบูตและ MLS ถูกสร้างขึ้น

Trusted AIX เพื่อ การรักษาความปลอดภัยระบบผ่านสี่องค์ประกอบหลักของการรักษาความปลอดภัยข้อมูล:

- การรักษาความลับ
- ความซื่อสัตย์
- สภาพพร้อมใช้งาน
- การตรวจสอบได้

นอกจากคุณลักษณะการรักษาความปลอดภัยที่จัดเตรียมโดย AIX, Trusted AIX เพิ่มความสามารถ ดังต่อไปนี้:

Sensitivity labels (SLs)
ทั้งหมดและไฟล์ทั้งหมดถูกเลเบลตามระดับ การรักษาความปลอดภัย กระบวนการสามารถเข้าถึงอ็อบเจ็กต์ที่อยู่ภายในขอบเขตการรักษาความปลอดภัย ของกระบวนการ

Integrity labels (TLs)
ทั้งหมดและไฟล์ทั้งหมดถูกเลเบลตามระดับ integrity ไฟล์ไม่สามารถถูกเขียนโดยกระบวนการที่มีเลเบลระดับ integrity ต่ำกว่าไฟล์ กระบวนการไม่สามารถอ่านจากไฟล์ที่มีเลเบล ระดับ integrity ต่ำกว่าของกระบวนการได้

แฟล็กการรักษาความปลอดภัยของไฟล์
แต่ละไฟล์สามารถมีแฟล็กเพิ่มเติมเพื่อควบคุมการรักษาความปลอดภัย ที่เกี่ยวข้องกับการดำเนินการ

แฟล็กการรักษาความปลอดภัยเคอร์เนล
ระบบทั้งหมดมีคุณลักษณะการรักษาความปลอดภัยที่เปิดใช้งานหรือปิดใช้งาน ต่างกันได้

Privileges
คำสั่งและการเรียกของระบบจำนวนมากมีเฉพาะในกระบวนการ ที่มี privileges จำเพาะ

การอนุญาต
ผู้ใช้แต่ละคนได้รับชุดของการอนุญาตเฉพาะได้ แต่ละ การอนุญาต อนุญาตให้ผู้ใช้เรียกใช้ฟังก์ชัน security-related จำเพาะได้ การอนุญาตถูกกำหนดให้ผู้ใช้ผ่านบทบาท

บทบาท
ฟังก์ชัน Role Based Access Control เป็นส่วนหนึ่งของ Trusted AIX มีไว้สำหรับตัวแทน ที่เลือก ซึ่งไม่ใช้ผู้ใช้ root ให้ทำหน้าที่ดูแล การกำหนดตัวแทนนี้ ทำได้โดยรวมการอนุญาตที่สัมพันธ์กัน ไว้ใน Role แล้วกำหนดบทบาทให้กับผู้ใช้ที่ไม่ใช่ root

การรักษาความลับ

การคุกคาม มุ่งเป้าไปที่การเปิดเผยข้อมูลแก่กลุ่มที่ไม่ได้รับอนุญาต เป็นปัญหาด้านการรักษาความลับ

Trusted AIX จัดเตรียมการนำ อ็อบเจ็กต์มาใช้ใหม่และกลไกควบคุมการเข้าใช้สำหรับการป้องกันรีซอร์สข้อมูลทั้งหมด ระบบปฏิบัติการประกันว่ารีซอร์สข้อมูลที่ป้องกันสามารถถูกเข้าถึงได้ เฉพาะผู้ใช้ที่ได้รับอนุญาตพิเศษและผู้ใช้ดังกล่าว ไม่สามารถทำให้รีซอร์สที่ป้องกันถูกเข้าถึงได้โดยผู้ใช้ที่ไม่ได้รับอนุญาต ไม่ว่าโดยเจตนาหรือไม่เจตนาก็ตาม

ผู้ดูแลระบบสามารถป้องกัน ไฟล์สำคัญไม่ให้ถูกเขียนไปที่ฟล็อปปีดิสก์หรือสื่อบันทึกที่ถอดได้อื่น การพิมพ์ไปที่พรินเตอร์ที่ไม่มีการป้องกัน หรือถูกถ่ายโอน ผ่านเน็ตเวิร์กไปยังระบบรีโมตที่ไม่ได้รับอนุญาต การปกป้องความปลอดภัยนี้ ถูกบังคับใช้โดยระบบปฏิบัติการและไม่สามารถหลบเลี่ยงโดยผู้ใช้ ที่ประสงค์ร้ายหรือกระบวนการที่ไม่ปลอดภัย

ความซื่อสัตย์

การคุกคามมุ่งเป้า ไปที่การดัดแปลงข้อมูลโดยกลุ่มที่ไม่ได้รับอนุญาตเป็น ปัญหาด้านความซื่อสัตย์

Trusted AIX นำเสนอ กลไกการรักษาความปลอดภัยหลายประเภทซึ่งประกันความซื่อสัตย์ของ trusted computing base และข้อมูลที่ปกป้อง ไม่ว่าข้อมูลถูกสร้างบน ระบบหรือถูกนำเข้าผ่านเน็ตเวิร์กรีซอร์ส กลไกการรักษาความปลอดภัย การควบคุมการเข้าใช้ประกันว่าเฉพาะผู้ที่ได้รับอนุญาตเท่านั้นที่สามารถแก้ไข ข้อมูลได้ เพื่อป้องกันผู้ใช้ที่ประสงค์ร้ายหรือกระบวนการที่ไม่ปลอดภัย เข้ายึด หรือปิดการทำงานรีซอร์สระบบ Trusted AIX เอา root privilege ออก การอนุญาตการดูแลระบบพิเศษและบทบาทอนุญาต การแยกหน้าที่การดูแลระบบ แทนการให้ root privileges แก่ผู้ใช้

สภาพพร้อมใช้งาน

การคุกคามมุ่งเป้า ไปที่ความสามารถในการเข้าใช้งานเซอร์วิสบนเครื่องโฮสต์ เป็น ปัญหาสภาพพร้อมใช้งาน ตัวอย่าง ถ้าโปรแกรมประสงค์ร้ายสร้างพื้นที่ไฟล์จนเต็ม เพื่อที่ จะไม่สามารถสร้างไฟล์ใหม่เพิ่มได้ นั่นคือยังคงเข้าถึงได้แต่ใช้งานไม่ได้

Trusted AIX ปกป้องระบบ จากการโจมตีโดยผู้ใช้ที่ไม่ได้รับอนุญาตและกระบวนการที่สร้าง การปฏิเสธการเข้าใช้งานเซอร์วิส (denial of service) กระบวนการที่ไม่มี privilege ไม่ได้รับอนุญาต ให้อ่านหรือเขียนไฟล์หรือไดเร็กทอรีที่มีการป้องกัน

การตรวจสอบได้

การคุกคาม มุ่งเป้าไปที่การไม่สามารถทราบได้ว่ากระบวนการใดทำงานใดอยู่ บนระบบ เป็นปัญหาการตรวจสอบได้ ตัวอย่าง เช่น ถ้าผู้ใช้หรือ กระบวนการที่แก้ไขระบบไม่สามารถถูกติดตามได้ คุณไม่สามารถ ระบุวิธีในการหยุดการดำเนินการดังกล่าวได้ในอนาคต

คุณลักษณะการรักษาความปลอดภัย ที่เพิ่มความสามารถนี้ประกัน identification และการพิสูจน์ตัวตนของ ผู้ใช้ทั้งหมดก่อนการอนุญาตให้ผู้ใช้เข้าถึงระบบ เซอร์วิสการตรวจสอบ จัดเตรียมชุดของเหตุการณ์ที่ตรวจสอบได้และการติดตามตรวจสอบเหตุการณ์ระบบที่เกี่ยวกับความปลอดภัย แก่ผู้ดูแลระบบ

คุณสมบัติของ Trusted AIX

- Trusted AIX ถูกติดตั้ง ผ่านเมนูติดตั้ง AIX ตัวเลือกเพิ่มเติมสามารถเลือกได้ระหว่างการติดตั้ง Trusted AIX
- สภาวะแวดล้อม Trusted AIX ไม่สามารถกลับเป็นสภาวะแวดล้อม AIX ปกติ โดยไม่มีการเขียนทับการติดตั้ง AIX ปกติ
- Root ถูกปิดใช้งานจากการล็อกอินในสภาวะแวดล้อม Trusted AIX
- ในสภาวะแวดล้อม Trusted AIX WPARs ที่สร้างจะทำงานในสภาวะแวดล้อม Labeled Security เช่นกัน
- Trusted AIX สนับสนุน ทั้ง MAC (Mandatory Access Control) และ MIC (Mandatory Integrity Control) ลูกค้าสามารถกำหนดชุดแยกของเลเบลสำหรับ MAC และ MIC
- ไฟล์ Label Encodings อยู่ในไดเร็กทอรี `/etc/security/enc` และดักจับข้อมูลการแปล label-to-binary ไฟล์ดีฟอลต์ Label Encodings ยึดตามข้อกำหนดการตั้งชื่อ Compartmented Mode Workstations (CMW) labels-related
- มีการสนับสนุนการติดตั้ง NIM เมื่อมีการเริ่มต้นจากไคลเอ็นต์ การติดตั้ง NIM จาก Server ทำไม่ได้เนื่องจาก root ถูกปิดใช้งานในการล็อกอิน บนระบบ MLS

- ระบบไฟล์ JFS2 (J2) (ใช้ Extended Attributes เวอร์ชัน 2) ถูกเปิดใช้งานสำหรับการเก็บ Labels ใน AIX ระบบไฟล์อื่น (เช่น J1 หรือ NFS) สามารถถูกประกอบเข้าในสภาวะแวดล้อม Trusted AIX เป็นระบบไฟล์ระดับเดียว เท่านั้น (เลเบลที่กำหนดให้กับจุดประกอบเข้า)
- สภาวะแวดล้อม X ถูกปิดใช้งานสำหรับ Trusted AIX
- Trusted AIX สนับสนุนโปรโตคอล CIPSO และ RIPSO สำหรับการสื่อสาร netowork-based label-based โปรโตคอลเหล่านี้ได้รับการสนับสนุนทั้ง IPv4 และ IPv6
- บางกลไกการรักษาความปลอดภัย AIX เป็นสิ่งสามัญระหว่าง AIX ปกติและ Trusted AIX กลไกการรักษาความปลอดภัยสามัญ สองกลไกนี้คือ Role Based Access Control (RBAC) และ Trusted Execution สำหรับการตรวจสอบ integrity
- เนื่องจาก root ถูกปิดใช้งานเมื่อ Trusted AIX ถูกติดตั้ง โปรแกรมติดตั้งตั้งตั้งค่ารหัสผ่านสำหรับผู้ใช้ ISSO, SA และ SO ระหว่าง การบูตครั้งแรกหลังจากการติดตั้ง ระบบยังคงใช้ไม่ได้จนกว่า จะมีการสร้างรหัสผ่าน
- เอกสารเผยแพร่ AIX 6 security features Redbooks® มีการใช้เคสและตัวอย่างสำหรับ Trusted AIX

ความปลอดภัยหลายระดับ

เป้าหมายหลักของระบบความปลอดภัยคือการบังคับใช้นโยบาย ความปลอดภัยของไซต์เพื่อจัดให้มีความเชื่อถือได้และสภาพพร้อมใช้งาน

นโยบายความปลอดภัย Trusted AIX จัดเตรียมชุดของคำสั่งที่กำหนด ซึ่งระบุชนิดของการเข้าถึงระบบที่ ทำได้ นี่รวมถึงการพักความสามารถของผู้ใช้ในการดำเนินการ และการป้องการ การเปลี่ยนแปลงกับระบบปฏิบัติการ

Trusted AIX ใช้ค่าควบคุมการเข้าใช้ และเงื่อนไข need-to-know จำเพาะเพื่อควบคุมการเข้าถึงไฟล์ ไดเร็กทอรี กระบวนการ และอุปกรณ์

Trusted AIX ดูแลหลักฐานการตรวจสอบ ของเหตุการณ์ที่สัมพันธ์กับความปลอดภัยทั้งหมด หลักฐานการตรวจสอบนี้อนุญาตสำหรับ แต่ละหน้าที่ แม้กับโปรแกรมซึ่งแก้ไข effective ID ผู้ใช้และ ID ผู้ใช้จริง เช่นคำสั่ง **su** Trusted AIX ยังจำกัด ฟังก์ชันการดูแลระบบกับบุคคลจำเพาะ ที่มีการอนุญาต และ privilege ขั้นต่ำ (การให้ชุดของ privileges เท่าที่จำเป็นจริงๆ แก่ผู้ใช้ หรือกระบวนการเพื่อดำเนินการ)

Identification และ authentication

กลไกความปลอดภัย Identification และ authentication (I&A) มีหน้าที่ในการประกันว่าแต่ละการเข้าถึงการร้องข้อไปที่ ระบบถูกระบุต้องและถูกพิสูจน์ตัวตน การระบุต้องการ ชื่อผู้ใช้และการพิสูจน์ตัวตนต้องการรหัสผ่าน

บัญชีผู้ใช้ Trusted AIX ทั้งหมด ป้องกันด้วยรหัสผ่าน ISSO (Information Systems Security Officer) สามารถตั้งค่า ระบบเพื่ออนุญาตให้ผู้ใช้เลือกรหัสผ่าน เขา/เธอ ซับเจ็คต์กันความยาวรหัสผ่าน และความซับซ้อนของข้อบังคับ ISSO สามารถระบุพารามิเตอร์การกำหนดอายุต่ำสุดและสุดงสุด (ระยะเวลาหมดอายุ) ในแบบต่อผู้ใช้ รวมถึง ระยะเวลาการเตือนก่อนที่รหัสผ่านจะหมดอายุ

กลไกความปลอดภัย identification และการพิสูจน์ตัวตนต้องการ ให้ชื่อผู้ใช้และ ID ผู้ใช้เป็นค่าเฉพาะ บัญชีผู้ใช้ที่มีรหัสผ่านไม่ถูกต้องไม่สามารถ ใช้ล็อกอินได้ ผู้ใช้ที่มีบทบาม ISSO ต้องเพิ่มรหัสผ่านเริ่มต้น สำหรับผู้ใช้ใหม่ แต่ละผู้ใช้ถูกกำหนด identifier เฉพาะเพิ่มเติม ที่ถูกใช้สำหรับจุดประสงค์การตรวจสอบ

เฉพาะฟอร์มที่เข้ารหัสของรหัสผ่านที่ถูกเก็บ รหัสผ่านไม่ได้ถูกเก็บ บนระบบในแบบข้อความธรรมดา รหัสผ่านที่เข้ารหัสถูกเก็บในไฟล์ รหัสผ่าน shadow ซึ่งถูกป้องกันการเข้าถึงยกเว้นโดยกระบวนการ ที่มี privilege สำหรับข้อมูลเพิ่มเติม ดูที่คำสั่ง **passwd**

ระบบ Trusted AIX รู้จัก บัญชีผู้ใช้สองชนิด: บัญชีผู้ใช้ระบบและผู้ใช้ บัญชีผู้ใช้ระบบ คือบัญชีที่มี ID ผู้ใช้น้อยกว่า 128 แม้ว่าบัญชีผู้ใช้ระบบอาจมีที่เชื่อมโยง แต่ไม่สามารถใช้เพื่อล็อกอินเข้าสู่ระบบได้

Discretionary access control

Discretionary access controls (DAC) เป็นเง่มุมความปลอดภัย ที่อยู่ภายใต้การควบคุมของเจ้าของไฟล์หรือไดเร็กทอรี

สิทธิ UNIX

ผู้ใช้ ที่มีการเข้าถึง เจ้าของ กับรีซอร์สสามารถทำดังต่อไปนี้:

- ให้การเข้าถึงกับผู้ใช้อื่นโดยตรง
- ให้การเข้าถึงเพื่อคัดลอกแก่ผู้ใช้อื่น
- จัดเตรียมโปรแกรมเพื่ออนุญาตการเข้าถึงรีซอร์สต้นฉบับ ตัวอย่างเช่น การใช้โปรแกรม SUID)

เมธอดบิตสิทธิ UNIX แบบดั้งเดิม (owner/group/other และ read/write/execute) เป็นตัวอย่างของการทำงาน DAC นี้

บิตสิทธิช่วยให้ผู้ใช้ให้หรือปฏิเสธการเข้าถึง ข้อมูลในไฟล์แก่ผู้ใช้และกลุ่ม (ขึ้นกับเกณฑ์ need-to-know) ชนิดของการเข้าถึงนี้ขึ้นกับ ID ผู้ใช้และกลุ่มซึ่งผู้ใช้ เป็นสมาชิก อ็อบเจ็กต์ระบบไฟล์ทั้งหมดมีสิทธิที่เชื่อมโยงเพื่ออธิบายการเข้าถึง สำหรับ owner, group และ world

เจ้าของไฟล์ยังสามารถให้ privileges การเข้าถึงกับผู้ใช้อื่นโดยเปลี่ยนความเป็นเจ้าของหรือกลุ่มของ ไฟล์ด้วยคำสั่ง **chown** และ **chgrp**

umask

เมื่อไฟล์ถูกสร้าง บิตสิทธิทั้งหมด ถูกเปิดเป็นค่าเริ่มต้น จากนั้นไฟล์ถูกเอาบิตสิทธิออก โดยกระบวนการ umask ซึ่งถูกเซ็ตระหว่างกระบวนการล็อกอิน ดีฟอลต์ umask ใช้กับทุกไฟล์ที่สร้างโดยเชลล์ผู้ใช้และทุกคำสั่งที่ รันจากเชลล์ผู้ใช้

โดยดีฟอลต์การตั้งค่า umask สำหรับรายการเคอร์เนล คือ 000 (ซึ่งคือให้สิทธิทั้งหมดแก่ผู้ใช้ทั้งหมด) AIX เซ็ต เคอร์เนล umask เป็น 022 (ซึ่งปิดบิตสิทธิ group และ world write) อย่างไรก็ตาม ผู้ใช้อาจแทนที่การตั้งค่านี้ได้ถ้าจำเป็น

หมายเหตุ: โปรดระวัง เกี่ยวกับการเปลี่ยน umask เพื่อตั้งค่าการอนุญาตมากกว่า 022 ถ้ามีการให้สิทธิ แก่ไฟล์และกระบวนการมากขึ้นเท่าใด ระบบทั้งหมดยิ่งมีความปลอดภัยลดลงเท่านั้น
มีสอง วิธีในการแทนที่ค่าติดตั้งดีฟอลต์ umask:

- คุณสามารถเปลี่ยนค่า umask ในไฟล์ `.profile`, `.login`, หรือ `.chsrc` ของคุณ การเปลี่ยนแปลงนี้จะมีผลกับไฟล์ ที่ถูกสร้างระหว่างเซสชันล็อกอินของคุณ
- คุณสามารถเซ็ตระดับ umask สำหรับแต่ละกระบวนการด้วยคำสั่ง **umask** หลังจากรันคำสั่ง **umask** ไฟล์ใหม่ทั้งหมดที่ถูกสร้าง จะได้รับผล ตามค่า umask ใหม่จนกว่าหนึ่งในสองเหตุการณ์ ดังต่อไปนี้เกิดขึ้น:
  - คุณรันคำสั่ง **umask** อีกครั้ง

    หรือ
  - คุณออกจากเชลล์ซึ่งคำสั่ง **umask** ถูกเรียก

ถ้าคุณรันคำสั่ง **umask** โดยไม่มีอากิวเมนต์คำสั่ง **umask** ส่งกลับค่า umask ปัจจุบันสำหรับเซสชันของคุณ

คุณควรอนุญาตให้เซสชันล็อกอิน สืบทอดค่า 022 umask ของเคอร์เนลโดยไม่ต้องระบุ umask ในโปรไฟล์ของคุณ ค่า Umask ที่น้อยกว่า 022 ควรถูกใช้ ด้วยความระมัดระวังอย่างยิ่งเท่านั้น

ถ้าจำเป็นต้องใช้สิทธิเพิ่มเติมสำหรับไฟล์ สิทธิเหล่านี้ควรถูกเซ็ตด้วยการใช้คำสั่ง **chmod** ด้วยความรอบคอบ หลังจากไฟล์ถูกสร้าง

Access Control Lists

นอกจากบิตสิทธิ UNIX และค่า umask มาตรฐาน, AIX ยังสนับสนุน access control lists (ACL)

บิตสิทธิ UNIX ควบคุมการเข้าถึงสำหรับเจ้าของไฟล์ หนึ่งกลุ่ม และทุกคนบนระบบ เท่านั้น ด้วย ACL เจ้าของไฟล์สามารถระบุสิทธิการเข้าถึงสำหรับ ผู้ใช้และกลุ่มเพิ่มเติม เหมือนกับบิตสิทธิ ACLs ถูกเชื่อมโยงกับ แต่ละอ็อบเจ็กต์ระบบ เช่นไฟล์หรือไดเร็กทอรี

บิตสิทธิ setuid และ setgid

บิตสิทธิ setuid setgid (set user ID และ set group ID) อนุญาตให้ไฟล์โปรแกรม รับด้วย ID ผู้ใช้หรือ ID กลุ่มของเจ้าของไฟล์ แทนที่จะเป็น ID ผู้ใช้ หรือ ID กลุ่มของผู้ที่รันโปรแกรมอยู่ ซึ่งทำได้โดยการตั้งค่า บิต setuid และ setgid ที่เชื่อมโยงกับไฟล์ ซึ่งอนุญาตให้มีการพัฒนาระบบย่อยที่มีการป้องกัน ซึ่งผู้ใช้สามารถเข้าถึงและ รันไฟล์โดยไม่ต้องเป็นเจ้าของไฟล์

ถ้าบิต setgid ถูกเซ็ตในไดเร็กทอรีพาเรนท์ เมื่ออ็อบเจ็กต์ถูกสร้าง อ็อบเจ็กต์ใหม่จะ มีกลุ่มเดียวกับไดเร็กทอรีพาเรนท์ ไม่ใช่กลุ่มของ ผู้สร้างอ็อบเจ็กต์ อย่างไรก็ตาม อ็อบเจ็กต์ที่สร้างในไดเร็กทอรีที่มีการเซ็ตบิต setuid เป็นของผู้สร้างอ็อบเจ็กต์ ไม่เจ้าเจ้าของไดเร็กทอรี บิต setuid/setgid ของไดเร็กทอรีพาเรนท์ได้รับการสืบทอดโดยไดเร็กทอรีย่อยเมื่อไดเร็กทอรีย่อย ถูกสร้าง

บิตสิทธิ setuid และ setgid แสดงถึงความเสี่ยง ของการรักษาความปลอดภัย โปรแกรมที่ถูกเซ็ตให้รันกับ root ในฐานะเจ้าของ ควรมี การเข้าถึงแบบไม่จำกัดกับระบบ บนระบบ Trusted AIX อย่างไรก็ตาม การใช้ privileges และการควบคุมการเข้าถึงอื่นช่วยลดความเสี่ยง การรักษาความปลอดภัยได้อย่างมาก

องค์ประกอบ Role Based Access Control

Trusted AIX สนับสนุน Role Based Access Control (RBAC) RBAC เป็นกลไกระบบปฏิบัติการ ซึ่งฟังก์ชันระบบจำเพาะผู้ใช้ root/system super user สามารถใช้งานได้โดยผู้ใช้ปกติโดยใช้บทบาทที่ถูกกำหนด ให้

องค์ประกอบหลักของ AIX RBAC คือ:

การอนุญาต
สตริงเหล่านี้แสดงการดำเนินการ privilege ที่สตริงเป็นตัวแทน และควบคุมตามชื่อโดยตรง ตัวอย่างเช่น สตริงการพิสูจน์ตัวตน `aix.network.manage` นิยามฟังก์ชันการจัดการเครือข่าย ในระบบปฏิบัติการ AIX

Privileges
privilege คือแอ็ตทริบิวต์ของกระบวนการที่อนุญาตให้กระบวนการ เลี่ยงข้อห้ามและข้อจำกัดของระบบ Privileges ถูกเชื่อมโยงกับกระบวนการและโดยปกติได้รับผ่าน การเรียกใช้คำสั่ง privileged

บทบาท
องค์ประกอบบทบาทใน AIX RBAC อนุญาตให้ผู้ใช้รวมชุดฟังก์ชันการจัดการในระบบ และกำหนดฟังก์ชันเหล่านี้

เพื่อให้ถูกจัดการโดยผู้ใช้ปกติ บทบาท ใน AIX ประกอบด้วย คอลเล็กชันของการอนุญาต (ซึ่งเป็นได้ทั้งการอนุญาตระบบ และการอนุญาตแบบกำหนดเอง) และบทบาทอื่น (เป็นบทบาทย่อย)

ดูที่ RBAC สำหรับข้อมูลเพิ่มเติมเกี่ยวกับ Role Based Access Control

Mandatory Access Control

Mandatory access control คือวิธีการบังคับใช้กับระบบในการจำกัด การเข้าถึงอ็อบเจ็กต์โดยขึ้นกับระดับความลับของอ็อบเจ็กต์และ clearance ของผู้ใช้ในทางตรงข้าม Discretionary Access Control ถูกบังคับใช้โดย เจ้าของไฟล์แต่ละคนไม่ใช่โดยระบบ

การใช้เลเบลสำหรับ MAC

Trusted AIX ใช้ ระบบเลเบลในการบังคับใช้ MAC บนระบบ Trusted AIX อ็อบเจ็กต์ที่มีชื่อทั้งหมดมี sensitivity labels (SL) เพื่อระบุระดับความลับของ อ็อบเจ็กต์ กระบวนการก็มี SL เช่นกัน SL ของกระบวนการระบุระดับของข้อมูลสำคัญ ที่กระบวนการได้รับอนุญาตให้เข้าถึง โดยทั่วไป กระบวนการต้องมีระดับความลับ เท่ากับหรือมากกว่าอ็อบเจ็กต์เพื่อ เข้าถึงอ็อบเจ็กต์ SLs สามารถใช้เพื่อทำให้ไฟล์เข้าถึงได้แบบอ่านอย่างเดียว หรือป้องกันไฟล์โดยสมบูรณ์จากการเข้าถึงโดยผู้ใช้ทั่วไป

อ็อบเจ็กต์ระบบ ทั้งหมดเช่นไฟล์ อ็อบเจ็กต์ IPC การเชื่อมต่อเน็ตเวิร์ก และกระบวนการ มี SL SL จะอยู่ในอ็อบเจ็กต์โดยอัตโนมัติเมื่ออ็อบเจ็กต์ถูกสร้าง core dumps ทั้งหมดถือว่าเป็นอ็อบเจ็กต์และจะถูกเลเบลโดยอัตโนมัติโดย ระบบ

อ็อบเจ็กต์ที่มีอยู่ก่อนหน้าการติดตั้ง Trusted AIX ได้รับ ดีฟอลต์ `SYSTEM_LOW` SL (SLSL) เมื่ออ็อบเจ็กต์เหล่านี้ ถูกเข้าถึงหลังจากการติดตั้ง Trusted AIX SL ไม่ถูกเซ็ตแบบถาวรบนอ็อบเจ็กต์เหล่านี้ คำสั่ง **settxattr** command ต้องถูกรันบนอ็อบเจ็กต์เพื่อเซ็ต SL สำหรับอ็อบเจ็กต์ที่ถูกสร้างหลังการติดตั้ง Trusted AIX SL ของอ็อบเจ็กต์ถูกเซ็ตเป็น SL ของกระบวนการที่สร้าง

ผู้ใช้และเลเบล

ระบบกำหนดขอบเขตของ SL ที่ใช้ได้ ให้กับแต่ละบัญชีผู้ใช้ ไม่ว่าโดยระบบดีฟอลต์หรือโดยการตั้งค่าจำเพาะผู้ใช้ และผู้ใช้สามารถดำเนินงานได้ภายในขอบเขตนี้เท่านั้น กระบวนการหรือผู้ใช้สามารถสร้างไฟล์หรือ ไดเร็กทอรีที่เลเบลระดับความลับปัจจุบันของกระบวนการได้เท่านั้น หรือผู้ใช้สามารถอ่านและเขียนไฟล์ซับเจ็คต์ไปที่ข้อจำกัด MAC ที่ระบบกำหนด

การบังคับใช้ MAC

Mandatory Access Control ถูกบังคับใช้ตลอดเวลา ที่กระบวนการพยายามเปิดอ็อบเจ็กต์ระบบไฟล์ เรียกข้อมูลแอ็ตทริบิวต์ของอ็อบเจ็กต์ระบบไฟล์ ส่งสัญญาณไปที่กระบวนการ ถ่ายโอนข้อมูลผ่าน STREAM หรือส่งหรือรับแพ็กเก็ตผ่านเน็ตเวิร์กอินเตอร์เฟส การเข้าถึง อ็อบเจ็กต์ระบบเข้าถึงเกิดขึ้นได้ ถ้าตรงกับเกณฑ์ MAC และ DAC เท่านั้น เมื่อผู้ใช้พยายามเข้าถึงไฟล์ ข้อจำกัด MAC จะถูกใช้ ก่อนข้อจำกัด DAC เช่นบิตสิทธิหรือ ACL ถูกตรวจสอบ

การเข้าถึง อ็อบเจ็กต์ระบบไฟล์ไม่ถูกจำกัดเฉพาะโดย SL ของอ็อบเจ็กต์ แต่โดย SL ของไดเร็กทอรีที่อ็อบเจ็กต์อยู่ด้วย ดังนั้น อ็อบเจ็กต์ระบบ ไฟล์สามารถถูกป้องกันที่ระดับความลับต่างกัน (SL ของไดเร็กทอรี) ไม่ใช่เพียง SL ของตัวอ็อบเจ็กต์เอง อ็อบเจ็กต์ระบบไฟล์สามารถมีหลายชื่อ (ลิงก์) อยู่ในไดเร็กทอรีหนึ่งหรือหลายไดเร็กทอรี แม้ว่าแต่ละชื่อ (ลิงก์) ถูกป้องกันที่ SL เดียวกับไฟล์ซึ่งลิงก์ไป การปกป้องที่มีผล ของลิงก์ต่างๆ อาจไม่เหมือนกันเนื่องจากลิงก์อยู่ในไดเร็กทอรี ที่มีระดับการป้องกันต่างกัน

ชื่อของอ็อบเจ็กต์ถูกเก็บอยู่ในไดเร็กทอรีที่อ็อบเจ็กต์อยู่ ดังนั้นกระบวนการที่มีการเข้าถึงกับ ไดเร็กทอรีนั้นสามารถดูชื่อของอ็อบเจ็กต์ทั้งหมดในไดเร็กทอรี อย่างไรก็ตาม เฉพาะกระบวนการที่มีการเข้าถึงที่เหมาะสมที่ได้รับอนุญาตให้อ่านหรือเขียนอ็อบเจ็กต์

การแสดงและการเปลี่ยน SL

SL ของอ็อบเจ็กต์และกระบวนการบนระบบสามารถดูได้ด้วยคำสั่ง **lstxattr** และสามารถแก้ไขได้โดยใช้คำสั่ง **settxattr**

มีเพียง ผู้ใช้ที่ได้รับการอนุญาตที่ถูกต้องและกระบวนการที่มี privileges ที่ถูกต้อง เท่านั้นที่สามารถเปลี่ยน SL ของไฟล์หรือกระบวนการ

ด้วยคำสั่ง **settxattr** เมื่อต้องการเปลี่ยนอ็อบเจ็กต์ระบบไฟล์ SL ไปเป็น SL ที่ระดับต่ำกว่าผู้ใช้ ควรมีการอนุญาต `aix.mls.label.sl.downgrade` เมื่อต้องการอัพเกรด อ็อบเจ็กต์ระบบไฟล์ SL ผู้ใช้ควรมีการอนุญาต `aix.mls.label.sl.upgrade` เพื่อเปลี่ยนแปลง SL ของกระบวนการ เพื่ออัพเกรด ผู้ใช้ควรมีการอนุญาต `aix.mls.proc.sl.upgrade` และเพื่อดาวน์เกรด ผู้ใช้ควรมีการอนุญาต `aix.mls.proc.sl.downgrade`

MAC บนไฟล์ descriptors ที่เปิด

เมื่อต้องการ อ่าน/เขียน และ เข้าถึงไฟล์แบบปกติ การตรวจสอบ MAC ถูกดำเนินการเมื่อกระบวนการเข้าถึงไฟล์ เมื่อกระบวนการมีไฟล์ descriptor สำหรับไฟล์ กระบวนการสามารถอ่านและเขียน ไฟล์แม้ว่า SL ของกระบวนการเปลี่ยนเป็นระดับที่ต่ำกว่า SL ของไฟล์ อย่างไรก็ตาม บางการดำเนินการเช่นการเซ็ต เจ้าของ สิทธิ เลเบล และ privileges ของไฟล์ ทำการตรวจสอบการเข้าถึงหลังจากกระบวนการได้รับไฟล์ descriptor

ซึ่งหมายความว่าการตรวจสอบ MAC และการหาพาธไดเร็กทอรีที่พาร์ติชัน ไม่ถูกดำเนินการเมื่อกระบวนการเข้าถึงไฟล์โดยใช้ไฟล์ descriptor SL ของไฟล์ และ/หรือ กระบวนการอาจเปลี่ยนแปลงและการเข้าถึงยังคง ทำได้

Mandatory Integrity Control

Mandatory Integrity Control คือวิธีการบังคับใช้กับระบบในการจำกัด การเข้าถึงและการแก้ไข อ็อบเจ็กต์โดยขึ้นกับ integrity ของอ็อบเจ็กต์และ clearance ของผู้ใช้ ขณะ MAC ถูกพิจารณากับระดับความลับ ของอ็อบเจ็กต์, MIC ถูกพิจารณากับความน่าเชื่อถือของอ็อบเจ็กต์

การใช้เลเบลสำหรับ MIC

Trusted AIX ใช้ ระบบเลเบลในการบังคับใช้ MIC บนระบบ Trusted AIX อ็อบเจ็กต์ที่มีชื่อทั้งหมดมี integrity labels (TL) เพื่อระบุระดับ integrity ของ อ็อบเจ็กต์ กระบวนการก็มี TL เช่นกัน TL ของระบุระดับของ integrity ข้อมูลที่กระบวนการได้รับอนุญาตให้เข้าถึง ยิ่ง TL มีค่าสูง ความน่าเชื่อถือของ อ็อบเจ็กต์หรือกระบวนการยิ่งสูงเท่านั้น

กระบวนการต้องมีความน่าเชื่อถืออย่างน้อย เท่ากับอ็อบเจ็กต์ ในการแก้ไขอ็อบเจ็กต์ ดังนั้น กระบวนการต้องมี TL เท่ากับหรือมากกว่า TL ของอ็อบเจ็กต์ ดังนั้นเลเบล integrity สามารถถูกใช้เพื่อทำให้ไฟล์เข้าถึงได้แบบอ่านอย่างเดียว

นอกจากนี้ กระบวนการ ไม่สามารถใช้ข้อมูลจากอ็อบเจ็กต์ที่มีความน่าเชื่อถือน้อยกว่าตัวกระบวนการเอง ดังนั้นอ็อบเจ็กต์ต้องมี TL เท่ากับหรือมากกว่ากระบวนการนั้น

อ็อบเจ็กต์ระบบ ทั้งหมด เช่นไฟล์และกระบวนการมี TL TL จะอยู่ในอ็อบเจ็กต์โดย อัตโนมัติเมื่ออ็อบเจ็กต์ถูกสร้าง core dumps ทั้งหมดถือว่าเป็นอ็อบเจ็กต์และจะถูกเลเบลโดยอัตโนมัติโดย ระบบ

อ็อบเจ็กต์ที่มีอยู่บนระบบก่อนหน้าการติดตั้ง Trusted AIX ได้รับ ดีฟอลต์ SYSTEM_LOW TL (SLTL) เมื่ออ็อบเจ็กต์เหล่านี้ถูกเข้าถึงหลังจากการติดตั้ง Trusted AIX SL ไม่ถูกเซ็ตแบบถาวรบนอ็อบเจ็กต์เหล่านี้ คำสั่ง **settxattr** command ต้องถูกรันบนอ็อบเจ็กต์เหล่านี้เพื่อเซ็ต TL สำหรับอ็อบเจ็กต์ที่ถูกสร้างหลังการติดตั้ง Trusted AIX TL ของอ็อบเจ็กต์เหล่านี้ถูกเซ็ตเป็นระดับ integrity ของกระบวนการที่ สร้างอ็อบเจ็กต์

ผู้ใช้และเลเบล

ระบบกำหนดขอบเขตของ SL ที่ใช้ได้ ให้กับแต่ละบัญชีผู้ใช้ ไม่ว่าโดยระบบดีฟอลต์หรือโดยการตั้งค่าจำเพาะผู้ใช้ และผู้ใช้สามารถดำเนินงานได้ภายในขอบเขตนี้เท่านั้น กระบวนการหรือผู้ใช้สามารถสร้างไฟล์หรือ ไดเร็กทอรีที่ TL ปัจจุบันของกระบวนการได้เท่านั้น หรือผู้ใช้สามารถอ่านและเขียนไฟล์ซับเจ็คต์ไปที่ข้อจำกัด MIC ที่ระบบกำหนด

การบังคับใช้ MIC

Mandatory Integrity Control ถูกบังคับใช้ เมื่อใดก็ตามที่ MAC ถูกบังคับใช้ นอกจากนี้ MIC ถูกบังคับใช้เมื่อไฟล์หรือไดเร็กทอรี ถูกลบหรือเปลี่ยนชื่อ

การเปลี่ยน TL

TL ของอ็อบเจ็กต์และกระบวนการสามารถดูได้ด้วยคำสั่ง **lstxattr** และแก้ไขได้ด้วยคำสั่ง **settxattr**

มีเพียง ผู้ใช้ที่ได้รับการอนุญาตที่ถูกต้องและกระบวนการที่มี privileges ที่ถูกต้อง เท่านั้นที่สามารถเปลี่ยน TL ของไฟล์หรือกระบวนการ ด้วยคำสั่ง **settxattr** เมื่อต้องการเปลี่ยนอ็อบเจ็กต์ระบบไฟล์ TL ไปเป็น TL ที่ระดับต่ำกว่าผู้ใช้ ควรมีการอนุญาต `aix.mls.label.tl.downgrade` เมื่อต้องการอัพเกรด อ็อบเจ็กต์ระบบไฟล์ TL ผู้ใช้ควรมีการอนุญาต `aix.mls.label.tl.upgrade` เพื่อเปลี่ยนแปลง TL ของกระบวนการ เพื่ออัพเกรด ผู้ใช้ควรมีการอนุญาต `aix.mls.proc.tl.upgrade` และเพื่อดาวน์เกรดผู้ใช้ควรมีการอนุญาต `aix.mls.proc.tl.downgrade`

NOTL

มี TL พิเศษ `NOTL` ที่สามารถถูกใช้กับระบบไฟล์ อ็อบเจ็กต์ ipc หรือกระบวนการ เมื่ออ็อบเจ็กต์ หรือกระบวนการมี `NOTL` TL ไม่มีการตรวจสอบ MIC บน อ็อบเจ็กต์หรือกระบวนการ เฉพาะผู้ใช้ที่มี privilege สามารถเซ็ต TL กับ `NOTL` หรือ เปลี่ยน TL ถ้า TL มีค่าเป็น `NOTL` ในขณะนี้

MIC บนไฟล์ descriptors ที่เปิด

เมื่อต้องการ อ่าน/เขียน และ เข้าถึงไฟล์แบบปกติ การตรวจสอบ MIC ถูกดำเนินการเมื่อกระบวนการเข้าถึงไฟล์ เมื่อกระบวนการมีไฟล์ descriptor สำหรับไฟล์ กระบวนการสามารถอ่านและเขียน ไฟล์แม้ว่า TL ของกระบวนการเปลี่ยนเป็นระดับที่ต่ำกว่า TL ของไฟล์ อย่างไรก็ตาม บางการดำเนินการเช่นการเซ็ต เจ้าของ สิทธิ เลเบล และ privileges ของไฟล์ ทำการตรวจสอบการเข้าถึงหลังจากกระบวนการได้รับไฟล์ descriptor ซึ่ง หมายความว่าการตรวจสอบ MIC ไม่ถูกดำเนินการ เมื่อกระบวนการเข้าถึงไฟล์โดยใช้ไฟล์ descriptor TL ของไฟล์ และ/หรือ กระบวนการอาจเปลี่ยนแปลงและการเข้าถึงยังคง ทำได้

เลเบล

เลเบลถูกใช้เพื่อแสดงระดับความปลอดภัยสำหรับซับเจ็คต์และอ็อบเจ็กต์ บนระบบ Trusted AIX เลเบล ที่จะถูกใช้ในระบบและความสัมพันธ์ระหว่างเลเบลถูกกำหนดโดย ISSO

Sensitivity labels (SLs):

SL ที่เชื่อมโยงกับแต่ละซับเจ็คต์และอ็อบเจ็กต์ถูกใช้เพื่อบังคับใช้ นโยบายการควบคุมการเข้าถึงข้อบังคับ จาก Bell-LaPadula Model ของ ค่าควบคุมการเข้าใช้

SL ประกอบด้วยสองส่วน:

- การจัดประเภทลำดับชั้น
- ชุดของการจัดแบ่งหนึ่งชุดหรือมากกว่านั้น

แต่ละไซต์การติดตั้งสามารถกำหนดชื่อและความสัมพันธ์ของเลเบล บนระบบนั้น ผู้ดูแลระบบสามารถตั้งค่าชื่อและความสัมพันธ์เหล่านี้ ตามที่ต้องการโดยนโยบายไซต์ในไฟล์การเข้ารหัสเลเบล

การจัดประเภท SL:

การจัดประเภทมีลำดับชั้นและแสดงระดับ ของระดับความลับ

ตัวอย่างเช่น ถ้า `Top Secret`, `Secret` และ `Unclassified` เป็น การจัดประเภทที่ใช้ได้ที่ไซต์ `Top Secret` สำคัญกว่า `Secret` และ `Secret` สำคัญกว่า `Unclassified` Trusted AIX สนับสนุน การจัดประเภทลำดับชั้นถึง 32,000

การจัดแบ่ง SL:

การจัดแบ่งแสดงหัวข้อหรือกลุ่มงาน แต่ละการจัดแบ่ง มีชื่อเช่น `NATO` หรือ `CRYPTO`

การจัดแบ่งไม่มีการจำลำดับแท้จริง แต่ ISSO สามารถกำหนด ข้อจำกัดซึ่งการจัดแบ่งและการจัดประเภทสามารถถูกรวมได้ Trusted AIX สนับสนุน 1,024 การจัดแบ่ง

คอมโพเนนต์ SL:

ในฟอร์ม human-readable, SL ถูกแสดงโดยสตริงขององค์ประกอบ องค์ประกอบแรกงแสดงการจัดประเภท องค์ประกอบที่เหลือแสดง การจัดแบ่ง องค์ประกอบถูกคั่นด้วยช่องว่าง

ตัวอย่างเช่น ถ้าไฟล์มีข้อมูลลับสุดยอดเกี่ยวกับ Brazilian economy การจัดประเภทลำดับชั้นของไฟล์ควรเป็นลับสุดยอด (`TS`) และการจัดแบ่งอาจเป็น Brazil (`B`) และ economy (`e`) ฟอร์ม human-readable ของ SL จะเป็น `TS B e` หรือ `Top Secret Brazil economy`

ความสัมพันธ์ SL:

ในฐานะผู้ใช้ระบบ เป็นสิ่งสำคัญที่จะต้องเข้าใจความสัมพันธ์ ระหว่างเลเบลและวิธีใช้เลเบล

มีความสัมพันธ์สามประเภทระหว่างเลเบล MAC:

- Dominance
- Equality
- Non-Comparable

Dominance

หนึ่ง SL (L1) ควบคุม SL อื่น (L2) เฉพาะถ้าทั้งสองเงื่อนไขดังต่อไปนี้ เป็นจริง:

- การจัดประเภทใน L1 เท่ากับหรือมากกว่าการจัดประเภท ใน L2
- ชุดของการจัดแบ่งใน L1 มี ชุดของการจัดแบ่งใน L2

ตัวอย่างเช่น ถ้าเราสมมุติว่าหนึ่ง SL L1 ของข้อมูลลับสุดยอด บนการจัดแบ่ง A และ B (TS A B) และอีก SL L2 ของข้อมูลลับบนการจัดแบ่ง A แต่ไม่ใช่ B (S A) ดังนั้น TS A B จะควบคุม S A เนื่องจาก การจัดประเภท TS ควบคุมการจัดประเภท S และ ชุดของการจัดแบ่งใน L1 มีชุด ของการจัดแบ่ง L2 L2 จะไม่ควบคุม L1 ใน ตัวอย่างนี้

ตารางที่ 33. การควบคุม SL

L1		L2		Dominance
Label	**Compartment**	**Label**	**Compartment**	
TOP SECRET	A,B	SECRET	A	L1 > L2

ความเท่าเทียม

หนึ่ง SL (L1) เท่ากับ SL อื่น (L2) เฉพาะถ้าทั้งสองเงื่อนไขดังต่อไปนี้ เป็นจริง:

- การจัดประเภทใน L1 เท่ากับการจัดประเภทใน L2
- ชุดของการจัดแบ่งใน L1 เท่ากับ ชุดของการจัดแบ่งใน L2

ถ้าสองเลเบลเท่ากัน แต่ละเลเบลจะควบคุมอีกเลเบล ตัวอย่างเช่น ถ้าเราสมมุติว่า SL สำหรับไฟล์ข้อมูลลับสุดยอดบนการจัดแบ่ง A (TS A) และไฟล์อื่นที่มีข้อมูลลับสุดยอดบนการจัดแบ่ง A (และ TS A), ดังนั้น SL จะเท่ากันและจะควบคุมกันและกัน

ตารางที่ 34. ความเท่ากันของ SL

L1		L2		Dominance
Label	**Compartment**	**Label**	**Compartment**	
TOP SECRET	A	TOP SECRET	A	L1 = L2

Non-comparable

สอง SL สามารถถูกแยก (L1 ไม่ เท่ากับ L2, L1 ไม่ควบคุม L2 และ L2 ไม่ควบคุม L1) หนึ่ง SL (L1) ไม่สามารถเข้ากันได้กับ (L2) อีกอันต่อเมื่อ เงื่อนไขต่อไปนี้เป็นจริง:

- ชุดของการจัดแบ่งใน L1 ไม่มีชุด ใน L2 และ L2 ไม่มี ชุดใน L1 โดยสมบูรณ์ ดังนั้น L1 และ L2 ถือว่า แยกกัน

ตัวอย่างเช่น ถ้าเราสมมุติว่าไฟล์ที่มีเลเบล L1 มีข้อมูล ลับสุดยอดบนการจัดแบ่ง A และ B (TS A B) และ L2 เป็นเลเบลสำหรับไฟล์ ที่มีข้อมูลลับ บนการจัดแบ่ง C (C C) ดังนั้น L1 เทียบไม่ได้กับ L2

ตารางที่ 35. Non-comparable SL

L1		L2		Dominance
Label	**Compartment**	**Label**	**Compartment**	
TOP SECRET	A, B	CLASSIFIED	C	-

Integrity labels (TLs):

TLs แสดงระดับของการไว้วางใจในอ็อบเจ็กต์หรือกระบวนการระบบ โครงสร้างของ TL เหมือนกับ SL ยกเว้น TL มีเฉพาะการจัดประเภทลำดับชั้นและไม่มีการจัดแบ่ง

กระบวนการขั้นตอนแก้ไขหรือลบอ็อบเจ็กต์เฉพาะถ้า TL ของกระบวนการ ควบคุม TL ของอ็อบเจ็กต์ กระบวนการสามารถลบหรือเปลี่ยนชื่ออ็อบเจ็กต์เฉพาะถ้า TL ของกระบวนการควบคุมทั้ง TL ของอ็อบเจ็กต์และ TL ของ ไดเร็กทอรีซึ่งมีอ็อบเจ็กต์อยู่ กระบวนการขั้นตอนเข้าถึงอ็อบเจ็กต์เฉพาะ ถ้า TL ของอ็อบเจ็กต์ควบคุม TL ของกระบวนการ

เมื่อต้องการระบุ TL ของอ็อบเจ็กต์หรือกระบวนการ ใช้คำสั่ง **lstxattr** เมื่อต้องการเปลี่ยน TL ของอ็อบเจ็กต์หรือกระบวนการ ใช้คำสั่ง **settxattr**

เลเบลบนซับเจ็คต์และอ็อบเจ็กต์:

ใน Trusted AIX กระบวนการ ถูกระบุเป็นซับเจ็คต์และแต่ละกระบวนการมี SL

SL ที่ใช้สำหรับการตรวจสอบ MAC เรียกว่า Effective SL (ESL) ESL ต้องอยู่ ภายในขอบเขต clearance ของกระบวนการ ขอบเขต clearance มีขีดจำกัดบน และขีดจำกัดล่าง ขีดจำกัดบนเรียกว่า Maximum clearance (Max CL) และขีดจำกัดล่างเรียกว่า Minimum clearance (Min CL) ESL, Max CL และ Min CL ถูกเก็บในโครงสร้าง credential ของกระบวนการและถูกกำหนดค่าระหว่างการสร้างกระบวนการ Max CL ต้องควบคุม Min CL และ ESL และ ESL ต้องควบคุม Min CL คำสั่ง **settxattr** และ **lstxattr** สามารถ ถูกใช้เพื่อแสดงรายการและเซ็ตกระบวนการ SL

การเข้าถึงกับอ็อบเจ็กต์ต่างๆ ในระบบจำเป็นต้องถูกควบคุม อ็อบเจ็กต์ สามารถเป็นหนึ่งในอ็อบเจ็กต์ดังต่อไปนี้:

- กระบวนการ
- ไฟล์ (ไฟล์ข้อมูลหรือไบนารี)
- อ็อบเจ็กต์ IPC เน็ตเวิร์กแพ็กเก็ต และอื่นๆ

อ็อบเจ็กต์ทั้งหมดและซับเจ็คต์บนระบบ MLS ถูกเลเบล

ไดเร็กทอรี
ไดเร็กทอรีถูกเชื่อมโยงกับขอบเขต SL; minimum SL และ maximum SL maximum SL ควรควบคุมหรือเท่าเทียมกับ minimum SL ไฟล์ทั้งหมดในไดเร็กทอรี อยู่ภายในขอบเขตนี้

ไฟล์ ไฟล์ธรรมดาถูกเชื่อมโยงกับสอง SL แต่ค่าเหมือน กันเสมอ ดังนั้นเพื่อประสิทธิภาพไฟล์มีเพียงหนึ่ง SL เท่านั้น ลิงก์สัญลักษณ์อาจมีค่า ต่างกันสำหรับ SL

ไฟล์พิเศษ
ไฟล์พิเศษเช่น อุปกรณ์ ttys และ fifos ถูกเชื่อมโยงกับ maximum และ minimum SL ไดเร็กทอรี ไฟล์ และไฟล์พิเศษมีเพียงหนึ่ง integrity label (TL) โดยที่กระบวนการถูกเชื่อมโยงกับ minimum และ maximum TL

กระบวนการ

กระบวนการทั้งหมดถูกเชื่อมโยงกับขอบเขต maximum และ minimum sensitivity cleareance เช่นเดียวกับขอบเขต maximum และ minimum integrity clearance ค่าเหล่านี้ ถูกสืบทอดจากค่า clearance ของผู้ใช้ ระดับ sensitivity และ integrity ซึ่งกระบวนการถูกเรียกใช้งานเรียกว่าระดับ effective sensitivity และ effective integrity

เลเบล clearance ของผู้ใช้:

ผู้ใช้มี maximum และ minimum sensitivity clearance labels (SCL) และ maximum และ minimum integrity clearance labels (TCL)

Maximum และ minimum sensitivity clearance labels

ผู้ใช้แต่ละคน มี maximum sensitivity clearance label (max SCL) effective SL ของผู้ใช้ต้องถูกควบคุมโดย max SCL max SCL ถูกใช้เพื่อจำกัดผู้ใช้ไม่ให้เห็นข้อมูลที่มีความสำคัญสูงกว่า min SCL ถูกใช้เพื่อป้องกันผู้ใช้ ที่ระดับการรักษาความปลอดภัยสูง จากการส่งข้อมูลไปให้ผู้ใช้ที่ระดับ การรักษาความปลอดภัยต่ำกว่า

ตัวอย่างเช่น สมมุติว่าผู้ใช้ A มีa max SCL และ min SCL ทั้งสองเป็น PUBLIC_A และผู้ใช้ B มี max SCL และ min SCL ของ PUBLIC_B โดยที่ไม่มี min SCL ผู้ใช้ A สามารถสื่อสารข้อมูลกับผู้ใช้ B โดย ล็อกอินด้วย effective SL ของ IMPL_LO และเขียนไปที่ไฟล์ที่ผู้ใช้ B สามารถอ่านได้ภายหลัง ด้วย min SCL ผู้ใช้ A ต้องล็อกอินที่ PUBLIC_A และสามารถเขียน ไฟล์ไปที่ PUBLIC_A เท่านั้น ไฟล์ที่เขียนไปที่ PUBLIC_A อ่านไม่ได้โดยผู้ใช้ B

เลเบล Maximum และ minimum integrity clearance

ผู้ใช้แต่ละคน มี maximum integrity clearance label (max TCL) effective TL ของผู้ใช้ต้องถูกควบคุมโดย max TCL max TCL ถูกใช้เพื่อจำกัดผู้ใช้ไม่ให้เห็นข้อมูลที่มีความสำคัญสูงกว่า min TCL ยังถูกใช้เพื่อป้องกันผู้ใช้ ที่ระดับการรักษาความปลอดภัยสูง จากการส่งข้อมูลไปให้ผู้ใช้ที่ระดับ การรักษาความปลอดภัยต่ำกว่า

เลเบลบนอ็อบเจ็กต์ระบบไฟล์:

ไฟล์ทั้งหมดรวมข้อมูลความปลอดภัยจำเพาะ เมื่อไฟล์ใหม่ถูกสร้าง จะมี SL เหมือนกับกระบวนการที่สร้างไฟล์ SL ของข้อมูลในไฟล์สามารถถูกอัพเกรดหรือดาวน์เกรดโดยการเพิ่มหรือลด SL ของไฟล์

ไดเร็กทอรีถูกกำหนด minimum และ maximum SL เมื่อไดเร็กทอรี ถูกสร้าง เมื่อสร้าง ทั้งสองค่าถูกเซ็ตเท่ากับ effective SL ของกระบวนการ ที่สร้าง โดยเฉพาะการสร้างไดเร็กทอรีระดับเดียว เฉพาะผู้ใช้ที่มี privileges และการอนุญาตที่เหมาะสมที่สามารถเปลี่ยน SL เหล่านี้ อ็อบเจ็กต์ใหม่สามารถถูกสร้าง ในไดเร็กทอรีนี้เฉพาะถ้า effective SL ของกระบวนการที่สร้าง อ็อบเจ็กต์ใหม่อยู่ในขอบเขตของ SL ของไดเร็กทอรี

หน้าต่างโดยปกติถูกสร้างเป็นกระบวนการไซลด์แยกโดยมี SL เท่ากับ effective SL ของผู้ใช้ อุปกรณ์ (ตัวอย่างเช่น pseudo-terminals ที่เชื่อมโยง กับหน้าต่าง) มี SL เชื่อมโยงเช่นกัน pipe ที่มีชื่อ ซึ่งคือ อุปกรณ์ที่ใช้สำหรับการสื่อสารระหว่างกระบวนการสืบทอด effective SL ของ กระบวนการที่สร้าง pipe ที่มีชื่อ stream ซึ่งคืออุปกรณ์ที่ใช้เพื่อจัดเตรียม ช่องสัญญาณข้อมูลสองทิศทางสำหรับสืบทอดระหว่างกระบวนการ สืบทอด effective SL ของกระบวนการที่สร้าง stream เช่นกัน

อุปกรณ์ทั้งหมดมี minimum SL และ maximum SL maximum SL ต้องควบคุม minimum SL โดยดีฟอลต์ minimum SL และ maximum SL ถูกเซ็ตเท่ากัน กระบวนการ สามารถเข้าถึงอุปกรณ์ดังกล่าวในโหมดอ่านเท่านั้น ถ้า SL ของกระบวนการ ควบคุม

minimum SL ของอุปกรณ์หรือไดเร็กทอรี กระบวนการอาจเข้าถึงเพียงอุปกรณ์ ดังกล่าวในโหมดเขียน ถ้า SL ของกระบวนการอยู่ภายในขอบเขตที่กำหนดโดย minimum และ maximum SL ของอุปกรณ์หรือไดเร็กทอรี

แฟล็กการรักษาความปลอดภัยของไฟล์

อ็อบเจ็กต์ สามารถ ถูกทำเครื่องหมายด้วย file security flags (FSFs) ซึ่งมีผลกับวิธีที่กระบวนการ จัดการอ็อบเจ็กต์ ดูที่ File Security Flags สำหรับรายการของ FSF และ privileges ที่จำเป็นในการเซ็ต แต่ละ FSF กระบวนการไม่มี file security flags

การลบไฟล์:

คุณสามารถลบอ็อบเจ็กต์จากระบบไฟล์เฉพาะถ้าเงื่อนไข ดังต่อไปนี้เป็นจริง:

- กระบวนการที่จะลบอ็อบเจ็กต์ต้องสามารถดูชื่อไฟล์ใน ไดเร็กทอรีที่มีไฟล์ นั่นคือ กระบวนการต้องมีการเข้าถึง การค้นหาในแต่ละไดเร็กทอรีในพาธใต้ไดเร็กทอรีซึ่งอ็อบเจ็กต์จะถูกลบ ออก และกระบวนการต้องมี SL ที่มีผลซึ่งควบคุม แต่ละไดเร็กทอรีเหล่านี้ ใช้คำสั่ง **ls** เพื่อดูชื่อ ไฟล์
- กระบวนการต้องมีการเข้าถึงเพื่อเขียน ในไดเร็กทอรีซึ่งอ็อบเจ็กต์ จะถูกลบออก

การพิมพ์ไฟล์:

ระบบย่อยพรินเตอร์เลเบลเอาต์พุตทั้งหมดโดยอัตโนมัติด้วย เลเบลระดับความลับที่เหมาะสม แต่ละงานพิมพ์จัดเตรียม หน้าแถบป้ายและหน้าเทรเลอร์โดยอัตโนมัติที่แสดงเลเบลที่สัมพันธ์กับความปลอดภัย และการทำเครื่องหมายทั้งหมด

การสำรองข้อมูลและการเรียกคืนไฟล์:

เมื่อเขียนข้อมูลไปที่ดิสก์หรือเทปบน AIX ด้วย คำสั่ง **backup**, SL ถูกรวมไว้ในข้อมูล

การอนุญาต SO จำเป็นในการใช้คำสั่ง **backup** หรือ **restore** เพื่ออิมพอร์ตหรือเอ็กซ์พอร์ตข้อมูลที่ไม่ได้เลเบลจากเทปหรือดิสก์ เมื่อข้อมูลที่ไม่ได้เลเบล ถูกเขียน ข้อมูลถูกกำหนดค่าดีฟอลต์ SL ของ `SYSTEM_LOW` สำหรับ ไฟล์และขอบเขต SL ของ `SYSTEM_LOW` ถึง `SYSTEM_HIGH` สำหรับ ไดเร็กทอรี

เลเบลบนอ็อบเจ็กต์ IPC:

ส่วนช่วยเหลือ IPC AIX ทั้งหมด เกี่ยวข้องในการสร้างและการเข้าถึงอ็อบเจ็กต์สื่อสาร

มีสามส่วนช่วยงาน IPC ที่ต่างกันที่กำหนดใน AIX:

- คิวข้อความ
- Semaphores
- หน่วยความจำที่แบ่งใช้

ทั้งหมดนี้เกี่ยวข้องกับการสร้างและการเข้าถึงอ็อบเจ็กต์สื่อสาร ที่ เรียกว่า IPC สำหรับการสื่อสารระหว่างกระบวนการ แต่อ็อบเจ็กต์ IPC ถูกป้องกัน โดยชุดของแอ็ตทริบิวต์เหมือนกับแอ็ตทริบิวต์ที่ป้องกันไฟล์ แอ็ตทริบิวต์ เหล่าคือ:

- ID ผู้ใช้ และ ID กลุ่มของเจ้าของอ็อบเจ็กต์
- ID ผู้ใช้และ ID กลุ่มของผู้สร้างอ็อบเจ็กต์
- โหมดการเข้าถึงรีซอร์ส ซึ่งเหมือนกับบิตสิทธิการเข้าถึง ไฟล์ แต่ละอ็อบเจ็กต์มีการเข้าถึง read, write และ execute สำหรับ world, group และ เจ้าของอ็อบเจ็กต์

- หมายเลขลำดับเพื่อติดตามการใช้รีซอร์ส
- คีย์เพื่อระบุรีซอร์ส

เหมือนกับแอ็ตทริบิวต์ระบบอื่น Trusted AIX ขยาย แอ็ตทริบิวต์เหล่านี้ด้วยแอ็ตทริบิวต์ความปลอดภัยเพิ่มเติม บนระบบ Trusted AIX อ็อบเจ็กต์ IPC ทั้งหมดมีแอ็ตทริบิวต์ดังต่อไปนี้:

- sensitivity label (SL)
- integrity label (TL)

คุณสามารถใช้คำสั่ง **settxattr** เพื่อดูแอ็ตทริบิวต์ ความปลอดภัยทั้งหมดของอ็อบเจ็กต์ IPC การอ่านแอ็ตทริบิวต์ของอ็อบเจ็กต์ IPC ต้องการการเข้าถึง DAC READ และ MAC READ กับอ็อบเจ็กต์

การเข้าถึงอ็อบเจ็กต์ IPC:

อ็อบเจ็กต์ IPC ถูกสร้าง, ลบ, และเข้าถึง ผ่านการเรียกระบบต่างๆ ที่กล่าวถึงในหัวข้อ Trusted AIX Programming ผู้ใช้ปกติจะไม่ดำเนินการดำเนินการเหล่านี้ หัวข้อนี้แสดงภาพรวมทั่วไปของ กฎสำหรับ การสร้าง การลบ และการเข้าถึงอ็อบเจ็กต์ IPC

เมื่อต้องการเข้าถึงอ็อบเจ็กต์ IPC กระบวนการต้องผ่านการตรวจสอบการเข้าถึง DAC, MIC และ MAC

การตรวจสอบการเข้าถึง DAC อยู่บนพื้นฐานของโหมด (owner, group หรือ world) ของอ็อบเจ็กต์และ ID ผู้ใช้และ ID กลุ่มของกระบวนการ กระบวนการ มีการเข้าถึงแบบ DAC owner กับอ็อบเจ็กต์ IPC ถ้า effective UID ของกระบวนการเหมือนกับ object owner UID หรือ object creator UID ซึ่งใช้กับการเข้าถึงกลุ่ม DAC เช่นกัน

การเข้าถึง MAC ขึ้นกับ SL ของกระบวนการและอ็อบเจ็กต์ การเข้าถึง MIC ขึ้นกับ TL ของกระบวนการและอ็อบเจ็กต์

กฎการเข้าถึงสำหรับเนื้อหาอ็อบเจ็กต์ IPC เหมือนกับแอ็ตทริบิวต์อ็อบเจ็กต์ IPC เมื่อต้องการอ่านเนื้อหาหรือแอ็ตทริบิวต์ของอ็อบเจ็กต์ IPC ต้องมีสิทธิการเข้าถึง DAC READ, MIC READ และ MAC READ เมื่อต้องการเขียนอ็อบเจ็กต์ IPC ต้องมีการเข้าถึง DAC WRITE, MIC WRITE และ MAC WRITE

แอ็ตทริบิวต์อ็อบเจ็กต์ IPC มีข้อจำกัดเคร่งครัดกว่าเนื้อหาอ็อบเจ็กต์ IPC การเปลี่ยนแอ็ตทริบิวต์อ็อบเจ็กต์ IPC จึงต้องการ privilege ที่สูงกว่า เมื่อต้องการแก้ไขแอ็ตทริบิวต์ AIX มาตรฐาน เช่นโหมด กระบวนการต้องการการเข้าถึง DAC OWNER และ MAC WRITE กับอ็อบเจ็กต์ เมื่อต้องการเปลี่ยน SL ของอ็อบเจ็กต์ IPC กระบวนการต้องมี:

- PV_SL_PROC privilege
- DAC OWNER (ดาวน์เกรดเท่านั้น)
- DAC WRITE
- MAC WRITE
- PV_SL_UG privilege เพื่ออัพเกรด SL หรือ PV_SL_DG privilege เพื่อดาวน์เกรด SL
- PV_MAC_CL ถ้ามีหรือ SL ใหม่ภายนอก clearance ของกระบวนการ
- MIC WRITE

เมื่อต้องการเปลี่ยน TL ของอ็อบเจ็กต์ IPC กระบวนการต้องมี:

- PV_TL privilege
- DAC OWNER

- MAC WRITE
- MIC WRITE

นอกจากนี้ เพื่อล็อกหรือปลดล็อกเซ็กเมนต์หน่วยความจำที่แบ่งใช้ในหน่วยความจำ กระบวนการต้องมี PV_KER_IPC_O privilege กระบวนการยังต้องการ PV_KER_IPC privilege เพื่อเปลี่ยน msg qbytes ของคิวข้อความในรูทีนย่อย msgctl

หลักการที่เกี่ยวข้อง:

"โปรแกรมมิง Trusted AIX" ในหน้า 507
การรักษาความปลอดภัยระบบขึ้นอยู่กับซอฟต์แวร์ trusted computing base (TCB) ฮาร์ดแวร์ และเฟิร์มแวร์ ซึ่งรวมถึงเคอร์เนลระบบปฏิบัติการทั้งหมด ไดรเวอร์อุปกรณ์ทั้งหมดและโมดูล System V STREAMS ส่วนขยาย เคอร์เนล และโปรแกรมที่ไว้วางใจทั้งหมด ไฟล์ทั้งหมดที่ใช้โดยโปรแกรมเหล่านี้ในการสร้างการตัดสินใจด้านความปลอดภัย ถือว่าเป็นส่วนหนึ่งของ TCB

การสร้างและการลบอ็อบเจ็กต์ IPC:

ไม่มีข้อจำกัดในการสร้างอ็อบเจ็กต์ IPC เมื่อกระบวนการ สร้างอ็อบเจ็กต์ IPC อ็อบเจ็กต์จะสืบทอด SL และ TL ของกระบวนการ

โหมดการเข้าถึงของอ็อบเจ็กต์ IPC ต้องถูกระบุโดยการเรียกระบบ ที่สร้างอ็อบเจ็กต์

เมื่อต้องการลบอ็อบเจ็กต์ IPC กระบวนการต้องมีการเข้าถึง DAC OWNER, MIC WRITE และ MAC WRITE กับอ็อบเจ็กต์

เน็ตเวิร์กที่ไว้วางใจ:

ชุดของข้อกำหนดเน็ตเวิร์กที่ปลอดภัยจำเป็นสำหรับแอ็ตทริบิวต์ความปลอดภัยส่วนขยาย ของระบบความปลอดภัยที่เพิ่มประสิทธิภาพ AIX Trusted Network สนับสนุนมาตรฐานความปลอดภัยเครือข่ายที่รู้จักกันดีรวมถึง U.S. DoD RFC1108 Revised Internet Protocol Security Option (RIPSO) และ Commercial Internet Protocol Security Option (CIPSO)

AIX Trusted Network สนับสนุนทั้ง IPv4 และ IPv6 เมื่อสื่อสารกับระบบที่ไว้วางใจอื่น SL ถูก encapsulate ในตัวเลือก IP ตามมาตรฐาน CIPSO/RIPSO การตรวจ MAC ถูกบังคับใช้ที่เลเยอร์ IP สำหรับ SL ที่ถูกส่งหรือ รับบนแพ็กเก็ต ขอบเขตเลเบลที่อนุญาตถูกกำหนดค่าด้วย กฎเน็ตเวิร์ก กฎเน็ตเวิร์กประกอบด้วยกฎโฮสต์และกฎอินเตอร์เฟส AIX Trusted Network ติดตั้งเฉพาะกฎดีฟอลต์อินเตอร์เฟส (หนึ่งกฎต่ออินเตอร์เฟสที่กำหนดค่า) คุณสามารถตั้งค่ากฎโฮสต์เพื่ออนุญาตการกรอง granular เพิ่มเติม คุณสามารถใช้คำสั่ง netrule เพื่อตั้งค่าทั้งกฎ โฮสต์และ อินเตอร์เฟส การดำเนินการที่สนับสนุนโดยคำสั่ง netrule รวมถึงกฎ การเพิ่ม การลบ การแสดง และการเคียวรี

คุณสามารถใช้คำสั่ง tninit เพื่อเตรียมข้อมูลเบื้องต้นระบบย่อย Trusted Network และดูและฐานข้อมูลกฎ Trusted Network

การปิดใช้งาน Root:

บัญชีผู้ใช้ root ถูกปิดใช้งานบนระบบ Trusted AIX นี่เป็นเรื่องหลักเพื่อลดความเสียหายที่สามารถเกิดขึ้นกับระบบ จากผู้ใช้คนเดียวที่มี privileges ทั้งหมด

ชนิดของล็อกอินระบบทั้งหมดที่เป็นผู้ใช้ root ถูกปิดใช้งาน มีเพียงคำสั่ง **su** ที่อนุญาตการล็อกอินผู้ใช้ root กระบวนการที่เป็นของ root จะไม่ถูกกำหนด privileges พิเศษใด root-owned setuid และโปรแกรม non-setuid ทำงานเหมือนเดิมเมื่อ มีการร้องขอโดยผู้ใช้ที่ได้รับอนุญาต สำหรับผู้ใช้ที่ไม่ได้รับอนุญาต โปรแกรมจะรัน ถ้า DAC modebits หรือ ACL อนุญาตการดำเนิน

การa แต่โปรแกรมจะไม่ถูกกำหนด privileges ใด ดังนั้นโปรแกรมอาจไม่สามารถดำเนินการดำเนินการ privileged ได้เมื่อรันโดยผู้ใช้ที่ไม่ได้รับอนุญาต ดังนั้นเป็นสิ่งจำเป็นในการกำหนด privileges ที่ถูกต้องให้กับแอ็พพลิเคชันที่ติดตั้งใหม่ ถ้าแอ็พพลิเคชันควร ดำเนินการปฏิบัติการ privileged

งานผู้ดูแลระบบสามารถกระทำโดยผู้ใช้ซึ่งได้รับรับบทบาท Information System Security Officer (ISSO), System Administrator (SA) หรือ System Officer (SO) บทบาทเหล่านี้อนุญาตให้ผู้ใช้ทำงานการดูแล administration ระบบได้

หมายเหตุ: ระหว่างการติดตั้ง Trusted AIX แอ็ตทริบิวต์ **su** ของบัญชีผู้ใช้ root ถูกเซ็ตเป็น `false` เพื่ออนุญาตการเข้าถึงบัญชีผู้ใช้ root กับผู้ใช้ที่ทำหน้าที่ดูแลอื่น ผู้ใช้ที่ได้รับ ISSO จะ จำเป็นต้องรีเซ็ตแอ็ตทริบิวต์นี้เป็น `true` โดยใช้คำสั่ง **chuser** และกำหนดรหัสผ่านให้กับบัญชีผู้ใช้นี้

การสนับสนุนเลเบลในการตรวจสอบ:

จุดประสงค์หลักของระบบย่อยการตรวจสอบคือการมอนิเตอร์และบันทึก เหตุการณ์ที่เกี่ยวกับความปลอดภัย

ข้อมูลที่จัดเตรียมโดยระบบย่อยการตรวจสอบเปิดใช้งานชนิดของ ข้อมูลที่จะถูกบันทึกดังต่อไปนี้:

- ความพยายามละเมิดนโยบายการรักษาความปลอดภัย
- การดำเนินการที่เกี่ยวกับความปลอดภัยสำเร็จ

ระบบย่อยการตรวจสอบมีความสามารถดังต่อไปนี้:

- กำหนดเหตุการณ์ที่จะตรวจสอบ
- เปิดและปิดการตรวจสอบขณะที่ระบบรันอยู่
- สลับไฟล์ร่องรอยการตรวจสอบได้อย่างเรียบร้อย (โดยไม่มีข้อมูลสูญหาย)
- แปลงข้อมูลการตรวจสอบเป็นรูปแบบที่อ่านได้
- เลือกและประมวลผลเซ็ตย่อยของข้อมูลการตรวจสอบ

เมื่อตั้งค่าระบบย่อยการตรวจสอบ ISSO ควรเข้าใจสิ่งที่ จะถูกตรวจสอบ เงื่อนไขภายที่เกิดขึ้นเมื่อมีการตรวจสอบ และวิธีเริ่มและหยุด การตรวจสอบ ดูที่ ภาพรวม การตรวจสอบ สำหรับข้อมูลรายละเอียดในการตั้งค่า เริ่มและหยุดการทำงาน การดูแล และการตรวจทานการตรวจสอบ

ระบบย่อยการตรวจสอบคงสถานะปัจจุบันและถูกทำต่อโดยอัตโนมัติ ในภาวะนั้นหลังจาก ไฟดับ ระบบขัดข้อง ระบบไฟฟ้าล้มเหลว หรือการขัดข้องอื่นๆ ระบบย่อยการตรวจสอบสามารถปิดตัวเองโดยอัตโนมัติ ปิดระบบ หรือเปลี่ยนไฟล์การตรวจสอบ ถ้าเงื่อนไขเกิดขึ้นโดยที่ไม่สามารถเก็บเร็กคอร์ดการตรวจสอบ ได้อีกต่อไปในไฟล์การตรวจสอบที่มีอยู่ ไฟล์การตรวจสอบสามารถถูกสลับโดยอัตโนมัติ เมื่อระบบไฟล์มีระดับถึงที่ระบุ อย่างไรก็ตาม ในเหตุการณ์ ที่ระบบไฟฟ้าเสียหายอย่างมาก เร็กคอร์ดการตรวจสอบบางส่วนอาจสูญหาย

ไดเร็กทอรีหลายระดับและที่พาร์ติชัน:

ไดเร็กทอรีหลายระดับเป็นไดเร็กทอรีมาตรฐานที่ถูกกำหนด ขอบเขต SL แทน SL เดียว ไดเร็กทอรีที่พาร์ติชันแสดง เป็นไดเร็กทอรีเดียวต่อผู้ใช้ อย่างไรก็ตาม ไฟล์ที่แสดงแก่ผู้ใช้จริงๆ แล้วอยู่ในไดเร็กทอรีย่อยที่ซ่อนอยู่ของไดเร็กทอรีที่พาร์ติชัน

ไดเร็กทอรีหลายระดับ:

ไดเร็กทอรีหลายระดับเป็นไดเร็กทอรีมาตรฐานที่ถูกกำหนด ขอบเขต SL แทน SL เดียว

เมื่อต้องการดูชื่อไฟล์ในไดเร็กทอรีหลายระดับ กระบวนการต้องถูกดำเนินการ ที่ระดับความปลอดภัยที่สูงกว่า minimum SL ของไดเร็กทอรี เมื่อ ต้องการสร้างหรือลบไฟล์จริง กระบวนการต้องถูกดำเนินการภายในขอบเขต SL ของไดเร็กทอรีหลายระดับ

แต่ละไฟล์ในไดเร็กทอรีหลายระดับมี SL ของตัวเองและถูกป้องกันโดย ข้อจำกัด MAC มาตรฐาน อย่างไรก็ตามกระบวนการที่มีการเข้าถึงกับ ไดเร็กทอรีสามารถดูชื่อของอ็อบเจ็กต์ทั้งหมดในไดเร็กทอรี ดังนั้น กระบวนการ อาจมีความสามารถในการอ่านและเขียน MAC ในไดเร็กทอรี แต่ไม่สามารถ อ่าน และ/หรือเขียนบางไฟล์ในไดเร็กทอรี แม้ว่ากระบวนการสามารถดู ชื่อของไฟล์ทั้งหมดในไดเร็กทอรี

ไดเร็กทอรีที่พาร์ติชัน:

ไดเร็กทอรีที่พาร์ติชันแสดงเป็นไดเร็กทอรีเดียวต่อผู้ใช้ อย่างไรก็ตาม ไฟล์ที่แสดงแก่ผู้ใช้จริงๆ แล้วอยู่ในไดเร็กทอรีย่อยที่ซ่อนอยู่ ของไดเร็กทอรีที่พาร์ติชัน

ไดเร็กทอรีหลายระดับมีความเสี่ยงด้านความปลอดภัย กระบวนการทำงานที่ระดับความปลอดภัย สูงสามารถอ่านไฟล์ที่ระดับความปลอดภัยต่ำกว่า แล้วสร้างไฟล์ ที่ระดับความปลอดภัยสูงเท่ากัน ขณะที่คุณลักษณะ MAC ป้องกันไม่ให้กระบวนการที่มีความปลอดภัยต่ำกว่า อ่านไฟล์ใหม่ กระบวนการที่มีความปลอดภัยต่ำกว่ายังคงสามารถเห็น ชื่อของไฟล์ใหม่ ถ้ากระบวนการที่มีความปลอดภัยสูงกำหนดชื่อไฟล์ใหม่ จากเนื้อหาของไฟล์ความปลอดภัยสูงต้นฉบับ กระบวนการความปลอดภัยต่ำกว่าสามารถเข้าถึงข้อมูลความปลอดภัยสูงกว่าโดยการอ่าน ชื่อไฟล์ใหม่

เมื่อไดเร็กทอรีที่พาร์ติชันถูกสร้างและกระบวนการแอดเดรสไดเร็กทอรี, ระบบสร้างไดเร็กทอรีย่อยที่ซ่อนด้วย SL เดียวกันกับการแอดเดรส กระบวนการ ถ้าจากนั้นกระบวนการสร้างไฟล์ ไฟล์ถูกสร้างจริงๆ ในไดเร็กทอรีย่อยที่ซ่อนอยู่ ไดเร็กทอรีที่พาร์ติชันอาจมีหลายไดเร็กทอรีย่อยที่ซ่อนอยู่ หลายไดเร็กทอรี แต่กระบวนการการกำหนดแอดเดรสไดเร็กทอรีที่พาร์ติชัน จะเห็นเพียงไฟล์ในไดเร็กทอรีย่อยที่ซ่อนอยู่ที่มี SL เหมือนกับ กระบวนการการกำหนดแอดเดรสนั้น เมื่อกระบวนการสร้างไดเร็กทอรีไซลด์ของไดเร็กทอรีย่อยที่พาร์ติชัน ไดเร็กทอรีไซลด์นั้นเป็น sub-subdirectory ที่พาร์ติชัน

ไดเร็กทอรีที่พาร์ติชันถูกกำหนดขอบเขต SL จาก SYSTEM_LOW ถึง SYSTEM_HIGH ดังนั้นกระบวนการใดๆ สามารถเข้าถึงไดเร็กทอรีที่พาร์ติชัน

ผู้ใช้ที่มีการอนุญาต **aix.mls.pdir.mkdir** สามารถสร้างไดเร็กทอรี ที่พาร์ติชันด้วยคำสั่ง **pdmkdir** ไดเร็กทอรีที่ พาร์ติชันว่างเปล่าสามารถลบออกด้วยคำสั่ง **pdrmdir** คำสั่ง **pdset** สามารถถูกใช้เพื่อเปลี่ยนไดเร็กทอรีธรรมดา ไปเป็นชนิดไดเร็กทอรีที่พาร์ติชัน ไม่มีคำสั่งในการเปลี่ยนไดเร็กทอรีที่พาร์ติชัน ไปเป็นไดเร็กทอรีธรรมดา

ภายในไดเร็กทอรีที่พาร์ติชัน คุณสามารถลิงก์ไฟล์ในหนึ่งไดเร็กทอรีย่อยที่พาร์ติชัน ไปที่ไดเร็กทอรีย่อยที่พาร์ติชันที่อยู่อื่นทั้งหมดด้วย SL สูงกว่าในไดเร็กทอรีที่พาร์ติชันเดียวกัน นี่อนุญาตการเข้าถึงไฟล์ภายในไดเร็กทอรี ที่พาร์ติชันโดยกระบวนการทั้งหมดที่มีการเข้าถึงไดเร็กทอรีย่อยที่พาร์ติชันนั้น หรือกับไดเร็กทอรีย่อยที่พาร์ติชันระดับสูงกว่า ในไดเร็กทอรีที่พาร์ติชันเดียวกัน คุณสามารถใช้คำสั่ง **pdlink** สำหรับการลิงก์ไฟล์นี้

โหมดการเข้าถึงไดเร็กทอรีที่พาร์ติชัน:

กระบวนการถูกกำหนดหนึ่งในสองโหมดเมื่อมีการสร้าง โหมดจริงหรือ โหมดเสมือน โหมดจะกำหนดวิธีที่กระบวนการดูไดเร็กทอรีพาร์ติชัน

กระบวนการโหมดจริงปฏิบัติกับไดเร็กทอรีที่พาร์ติชันแบบไดเร็กทอรีหลายระดับ มาตรฐาน ไดเร็กทอรีย่อยที่พาร์ติชันทั้งหมดสามารถถูกเข้าถึงแบบไดเร็กทอรีมาตรฐาน ซับเจ็คต์กับ DAC, MIC ปกติและข้อจำกัด MAC กระบวนการโหมดจริงสามารถ เข้าสู่ไดเร็กทอรีที่พาร์ติชันและดูไดเร็กทอรีย่อยทั้งหมด ซับเจ็คต์กับ DAC, MIC และข้อจำกัด MAC

กระบวนการโหมดเสมือนจะไม่เข้าสู่ไดเร็กทอรีที่พาร์ติชัน แต่ถูกเปลี่ยนทิศทาง ไปที่ไดเร็กทอรีย่อยที่พาร์ติชัน ซึ่ง maximum และ minimum SL มีค่า เท่ากันทั้งคู่กับ effective SL ของกระบวนการ

กระบวนการโหมดจริงสามารถรันคำสั่งในโหมดเสมือนด้วยคำสั่ง **pdmode** (ตัวอย่างเช่น `pdmode ls`) เช่นเดียวกัน กระบวนการโหมดเสมือน สามารถรันคำสั่งในโหมดจริง ด้วยคำสั่ง **pdmode** (ตัวอย่างเช่น `pdmode -r ls`) อย่างไรก็ตาม นี่ต้องการการอนุญาต `aix.mls.pdir.mode` ด้วยการอนุญาตนี้ คุณสามารถสลับจากเชลล์ที่รันในโหมดเสมือน ไปเป็นเชลล์ที่รันในโหมดจริงโดยรัน `pdmode -r sh` ไม่จำเป็นต้องใช้การอนุญาตเพื่อเรียกทำงานโปรแกรมในโหมดเสมือนขณะรันใน โหมดจริง

การดูและการเปลี่ยนชนิดไดเร็กทอรี:

คุณสามารถใช้คำสั่ง **lstxattr** เพื่อแสดงชนิดไดเร็กทอรีเป็นส่วนหนึ่งของแอ็ตทริบิวต์ `secflags` `FSF_PDIR` ระบุ ไดเร็กทอรีที่พาร์ติชัน, `FSF_PSDIR` ระบุไดเร็กทอรีย่อย ที่พาร์ติชัน และ `FSF_PSSDIR` ระบุ sub-subdirectory ที่พาร์ติชัน เมื่อต้องการเปลี่ยนชนิดไดเร็กทอรีธรรมดาเป็นชนิดไดเร็กทอรีที่พาร์ติชัน ให้ใช้คำสั่ง **pdset**

การดูแลระบบ Trusted AIX

การจัดการระบบ Trusted AIX เกี่ยวข้องกับจำนวนของปัจจัยที่จำเพาะกับ Trusted AIX

การติดตั้ง Trusted AIX

Trusted AIX สามารถ ถูกเปิดใช้งานเฉพาะระหว่างการติดตั้งระบบปฏิบัติการฐานโดยใช้ตัวเลือก `Security Model` จากเมนูติดตั้ง

ตัวเลือกการโอนย้ายระบบสำหรับ Trusted AIX ไม่สนับสนุน สำหรับการติดตั้ง preservation ระบบไฟล์ต้องเป็น JFS2 สำหรับการติดตั้งเน็ตเวิร์กแบบไม่มีพร้อมท์ ดูที่ ตารางที่ 36สำหรับรหัสผ่านที่เชื่อมโยง กับผู้ใช้การดูแลระบบดีฟอลต์

ตารางที่ 36. รหัสผ่านสำหรับผู้ใช้การดูแลระบบ ดีฟอลต์

User	Password
isso	isso
sa	sa
so	so

รันโหมด

สองรันโหมด configuration และ operational พร้อมใช้เพื่ออนุญาตการตั้งค่าระบบและดูแล และ สำหรับการดำเนินการรายวัน

เมื่อระบบบูต จะรันในโหมดคอนฟิกูเรชัน หลังจาก การเตรียมข้อมูลเสร็จสมบูรณ์ รันโหมดจะเปลี่ยนเป็น operational

โหมดคอนฟิกูเรชันถูกใช้เพื่อดูแลและกู้คืนระบบ เมื่อ ระบบถูกบูตในโหมดผู้ใช้เดียว ระบบถูกกำหนดค่าเท่าที่จำเป็นและเน็ตเวิร์กถูกปิดใช้งาน โหมดคอนฟิกูเรชันถูกใช้สำหรับการดูและส่วนสำคัญ เรื่องเกี่ยวกับความปลอดภัย ของระบบ

โหมด Operational เป็นโหมดการดำเนินงานระบบมาตรฐาน ระบบเปลี่ยนเป็น โหมดนี้หลังจากงานทั้งหมดที่จำเป็นในการเข้าสู่ระดับรันดีฟอลต์ สมบูรณ์

รันโหมดระบบสามารถถูกแสดงด้วยคำสั่ง **getrunmode** และสามารถถูกแก้ไขด้วยคำสั่ง **setrunmode**

แฟล็กการรักษาความปลอดภัยเคอร์เนล

แฟล็กความปลอดภัยเคอร์เนลถูกใช้เพื่อเปิดใช้งาน/ปิดใช้งาน คุณลักษณะการรักษาความปลอดภัย เช่นการบังคับใช้การตรวจสอบเลเบล การตรวจสอบเลเบล integrity ระหว่าง การอ่าน และวัตถุประสงค์อื่น

เคอร์เนลตรวจสอบแฟล็กการรักษาความปลอดภัยเคอร์เนลก่อนบังคับใช้การตรวจสอบด้านความปลอดภัย แฟล็กเหล่านี้ได้รับการสนับสนุนเฉพาะเมื่ Trusted AIX ถูก เปิดใช้งาน ในพื้นที่ผู้ใช้ แฟล็กเหล่านี้ถูกเก็บในฐานข้อมูล ODM ขึ้นอยู่กับ รันโหมดของระบบ เคอร์เนลตรวจสอบแฟล็กความปลอดภัยเคอร์เนล ที่เกี่ยวข้อง

ตารางที่ 37. แฟล็กความปลอดภัยและค่าดีฟอลต์เคอร์เนล

แฟล็กความปลอดภัยเคอร์เนล	เปิดใช้งาน	ปิดใช้งาน	โหมด Operational ดีฟอลต์	Configuration mode default
tnet_enabled	การทำงานของเน็ตเวิร์กที่ไว้วางใจที่มี	การทำงานของเน็ตเวิร์กที่ไว้วางใจไม่สามารถถูกกำหนดค่าหรือ ใช้ได้	ปิดใช้งาน	ปิดใช้งาน
tl_write_enforced	MIC บังคับใช้การดำเนินการเขียน ลบและเปลี่ยนชื่อ	คอนฟิกูเรชันเซ็ตเพื่อที่ TL จะไม่ถูกใช้สำหรับการตรวจสอบ การเขียน	เปิดใช้งาน	เปิดใช้งาน
tl_read_enforced	MIC ที่บังคับใช้บนการดำเนินการอ่าน	คอนฟิกูเรชันเซ็ตเพื่อที่ TL จะไม่ถูกใช้สำหรับการตรวจสอบ การอ่าน	ปิดใช้งาน	ปิดใช้งาน
sl_enforced	MAC ที่บังคับใช้	คอนฟิกูเรชันเซ็ตเพื่อที่ SL จะไม่ถูกใช้สำหรับการควบคุม การเข้าถึง	เปิดใช้งาน	ปิดใช้งาน
trustedlib_enabled	แฟล็ก FSF_TLIB flag บนอ็อบเจ็กต์ระบบไฟล์ honored	แฟล็ก FSF_TLIB ไม่ honored	ปิดใช้งาน	ปิดใช้งาน

การตั้งค่าพารามิเตอร์เคอร์เนล

เคอร์เนล Trusted AIX สามารถถูกตั้งค่าเพื่อบังคับเกณฑ์ความปลอดภัยที่จำเป็นตามนโยบายไซต์

คอนฟิกูเรชันความปลอดภัยที่ดูโดยใช้คำสั่ง **getsecconf** และสามารถถูกเปลี่ยนโดยใช้คำสั่ง **setsecconf** พารามิเตอร์เคอร์เนลที่กำหนดค่าได้มีดังนี้:

- การบังคับใช้เลเบลระดับความลับ
- การบังคับใช้การอ่าน Integrity
- การบังคับใช้การเขียน Integrity
- Trusted Network
- Trusted library

พารามิเตอร์เหล่านี้สามารถถูกกำหนดค่าขณะที่ระบบอยู่ในรันโหมด configuration

การกำหนดไฟล์ /etc/security/enc/LabelEncodings เอง

เลเบลสำหรับระบบถูกกำหนดในไฟล์ /etc/security/enc/LabelEncodings และสามารถถูกกำหนดสำหรับแต่ละไซต์

เลเบลสามารถถูกกำหนดเองหลังจาก Trusted AIX ถูกติดตั้ง

ระบบ Trusted AIX ได้กำหนด SYSTEM LOW SL (SLSL) ที่ถูกควบคุมโดยเลเบลระดับความลับอื่นบนระบบและ SYSTEM HIGH SL (SHSL) ที่กำหนด ซึ่งควบคุมเลเบลระดับความลับอื่นทั้งหมด เช่นเดียวกัน SYSTEM LOW TL (SLTL) ถูกควบคุมโดยเลเบล integrity อื่นทั้งหมดบนระบบและ SYSTEM HIGH TL (SHTL) ควบคุมเลเบล integrity อื่นทั้งหมด นิยามเหล่านี้รับค่า สูงสุดและต่ำสุด SL และ TL ตามที่กำหนดไว้ในไฟล์ /etc/security/enc/LabelEncodings

เมื่อระบบ Trusted AIX ถูกบูต เลเบลระบบจากไฟล์ /etc/security/enc/LabelEncodings ถูกดาวน์โหลดไปที่เคอร์เนล เลเบลยังสามารถถูกดาวน์โหลดไปที่ เคอร์เนลด้วยคำสั่ง **setsyslab** เลเบลระบบ ตามที่กำหนดในเคอร์เนลสามารถถูกแสดงด้วยคำสั่ง **getsyslab** ขอแนะนำให้ระบบถูกรีบูตหลังจากแก้ไขไฟล์ /etc/security/enc/LabelEncodings

ความคิดเห็นสามารถถูกใส่ไว้ในไฟล์ /etc/security/enc/LabelEncodings ได้ทุกที่ที่สามารถเริ่มไฟล์ได้ ความคิดเห็นเริ่มต้นด้วย * และ ต่อไปจนจบบรรทัด

ไฟล์ /etc/security/enc/LabelEncodings มีข้อมูลเวอร์ชันและส่วนจำเป็นดังต่อไปนี้ แต่ละส่วนควรเริ่มด้วยหนึ่งในคีย์เวิร์ดส่วนเหล่านี้ตามด้วย โคลอน (:)

- classifications
- information labels
- sensitivity labels
- clearances
- channels
- printer banners
- accreditation range

ไฟล์ /etc/security/enc/LabelEncodings เริ่มด้วยรายการ VERSION รายการนี้เป็นลำดับ ของอักขระและมีช่องว่างได้

แต่ละคีย์เวิร์ดดังต่อไปนี้มีอยู่ในส่วนได้ คีย์เวิร์ด เหล่านี้ปิดท้ายด้วยเซมิโคลอน (;):

name=*name*
: คีย์เวิร์ดเพื่อกำหนดชื่อเต็มของการจัดประเภทหรือการจัดแบ่ง

sname=*name*
: คีย์เวิร์ดเพื่อกำหนดชื่อย่อ เป็นทางเลือก

aname=*name*
: คีย์เวิร์ดทางเลือกสำหรับการจัดประเภท เป็นทางเลือก

value=*value*
: คีย์เวิร์ดเพื่อระบุค่าจำนวนเต็มภายในของการจัดประเภท หรือการจัดแบ่ง

compartments=*bit*
: คีย์เวิร์ดเพื่อว่าระบุบิตการจัดแบ่งใดต้องเป็น 0 หรือ 1 ถ้ามี คำในเลเบล

การเพิ่มประสิทธิภาพ Trusted AIX กับรูปแบบการเข้ารหัสเลเบล

การเข้ารหัสเลเบลตามที่แนะนำโดย Defense Intelligence Agency Document DDS-2600-6216-93 ไม่ สนับสนุนเลเบล integrity

โดยดีฟอลต์ เลเบลระดับความลับ ถูกใช้เป็นเลเบล integrity Trusted AIX จัดเตรียมการสนับสนุน สำหรับส่วนเลเบล integrity ทางเลือก ซึ่งต่างไปจาก ส่วนเลเบลระดับความลับได้ ซึ่งให้ความยืดหยุ่นในการมี ชื่อการจัดประเภทและค่าสำหรับระดับความลับและเลเบล integrity ต่างกันได้ ตัวอย่างเช่น เลเบลระดับความลับนำหน้าด้วย SL ได้ และเลเบลระดับความลับนำหน้าด้วย TL ตามลำดับ:

ตารางที่ 38. ชื่อและค่าการจัดประเภทเลเบลระดับความลับ

name	sname	value
name= SL IMPLEMENTATION LOW	sname= SL_IMPL_LO	value= 0
name= SL UNCLASSIFIED	sname= SL_U	value= 20
name= SL PUBLIC	sname= SL_PUB	value= 40
name= SL SENSITIVE	sname= SL_SEN	value= 60
name= SL RESTRICTED	sname= SL_RES	value= 80
name= SL CONFIDENTIAL	sname= SL_CON	value= 100
name= SL SECRET	sname= SL_SEC	value= 120
name= SL TOP SECRET	sname= SL_TS	value= 140

ตารางที่ 39. ชื่อและค่าการจัดประเภทเลเบล Integrity

name	sname	value
name= TL IMPLEMENTATION LOW	sname= TL_IMPL_LO	value= 0
name= TL UNCLASSIFIED	sname= TL_U	value= 20
name= TL PUBLIC	sname= TL_PUB	value= 40
name= TL SENSITIVE	sname= TL_SEN	value= 60
name= TL RESTRICTED	sname= TL_RES	value= 80
name= TL CONFIDENTIAL	sname= TL_CON	value= 100
name= TL SECRET	sname= TL_SEC	value= 120
name= TL TOP SECRET	sname= TL_TS	value= 140

กฎดังต่อไปนี้ใช้กับส่วนเลเบล integrity:

- ส่วน "INTEGRITY LABELS" ควรถูกเพิ่มหลังจากส่วน "NAME INFORMATION LABELS" เท่านั้น ในกรณีที่ผู้ดูแลระบบ ไม่ได้กำหนดส่วน "NAME INFORMATION LABELS" ทางเลือก ส่วน "INTEGRITY LABELS" ควรถูกเพิ่มต่อจากส่วน "ACCREDITATION RANGE"
- ควรมีเพียงหนึ่งส่วน "INTEGRITY LABELS" ในไฟล์การเข้ารหัส เลเบล ส่วนเดียวกันใช้กับทั้งอ็อบเจ็กต์และซับเจ็คต์

- ส่วน ″INTEGRITY LABELS″ ใหม่เป็นส่วนทางเลือก ในกรณี ที่ส่วนนี้ไม่มีอยู่ การจัดประเภทตามที่กำหนดใน ส่วน ″CLASSIFICATIONS″ ที่จำเป็นควรถูกใช้
- ส่วน ″INTEGRITY LABELS″ ควรเหมือนกับส่วน ″CLASSIFICATIONS″ จะมีคีย์เวิร์ดดังต่อไปนี้: ″**name=**″, ″**sname=**″, ″**aname=**″ และ ″**value=**″ คีย์เวิร์ด ″**initial compartments=**″ และ″**initial markings=**″ ซึ่งเป็นส่วนหนึ่งของส่วน ″CLASSIFICATIONS″ จะใช้ไม่ได้ในส่วน ″INTEGRITY LABELS″
- ขอบเขตข้อมูลสำหรับ ″**value=**″ จะเหมือนกับ ส่วน ″CLASSIFICATIONS″ – ค่าต่ำสุดคือ 0 ถึงสูงสุด คือ 32,000

การเริ่มระบบ

การรักษาความปลอดภัยของระบบจะถูกเรียกโดยอัตโนมัติระหว่าง การเริ่มระบบ คุณควรตรวจสอบว่าพารามิเตอร์ความปลอดภัยแสดงระหว่าง การเริ่มต้นถูกต้องสำหรับระบบ

โหมดเริ่มต้นคอนฟิกูเรชัน:

โหมด Configuration ถูกใช้เพื่อดูแลและกู้คืนระบบ

เมื่อระบบถูกบูตในโหมดผู้ใช้เดียว ระบบถูกกำหนดค่าเท่าที่จำเป็นและ เน็ตเวิร์กถูกปิดใช้งาน

โหมดเริ่มต้นการดำเนินการ:

โหมดการดำเนินการถูกใช้สำหรับงานประจำวัน

โดยปกติ ระบบควรถูกบูตโดยตรงในโหมด หลายผู้ใช้ ถ้าโปรแกรมการอนุญาตบูตได้รับชื่อผู้ใช้และรหัสผ่านที่ถูกต้อง ระบบเข้าสู่โหมดการทำงาน หน้าจอการพิสูจน์ตัวตนล็อกอินคอนโซล ถูกแสดง และผู้ใช้ที่ถูกต้องสามารถล็อกอินได้

กลไกความปลอดภัยเช่นเลเบลระดับความลับ ค่าควบคุมการเข้าใช้ discretionary ค่าควบคุมการเข้าใช้ mandatory การตรวจสอบ privilege การระบุแลการพิสูจน์ตัวตน และการอนุญาต การอนุญาตทั้งในโหมด configuration และโหมด operational ตามที่ควบคุมโดยแฟล็ก configuration ความปลอดภัยที่เกี่ยวข้อง สำหรับข้อมูลเพิ่มเติม ดูที่คำสั่ง **getsecconf**

ขอแนะนำให้ระบบถูกดำเนินการเฉพาะในโหมด operational เพื่อประกันว่าการทำงานของระบบที่ต้องการทั้งหมดพร้อมใช้งาน

กระบวนการบูต:

สคริปต์บูตใหม่ที่เพิ่มให้กับไฟล์ `/etc/inittab` บนระบบ Trusted AIX สคริปต์บูต ใหม่คือ `rc.mls.boot`, `rc.mls.net`, และ `rc.mls` และถูกเรียกใช้งานตามลำดับนี้

ขั้นตอนที่เรียกใช้งานในสคริปต์ `rc.mls.boot` คือ:

1. การตรวจสอบ integrity แบบโต้ตอบถูกรันเพื่อพร้อมท์ผู้ใช้สำหรับข้อมูล วิธีจัดการแต่ละความแตกต่าง (ใช้คำสั่ง **trustchk**)
2. เซ็ตแฟล็กความปลอดภัยเคอร์เนลโหมดคอนฟิกูเรชัน (โดยใช้คำสั่ง **setsecconf**)
3. เซ็ตเลเบลระบบ (Minimum และ Maximum Sensitivity Labels และ Integrity Labels)
4. แฟล็กความปลอดภัยเคอร์เนลโหมดคอนฟิกูเรชันถูกแสดงบนจอแสดงผล

ขั้นตอนที่เรียกใช้งานในสคริปต์ `rc.mls.net` คือ:

1. เตรียมข้อมูลเบื้องต้น Trusted AIX sub-system
2. ถ้าไฟล์ /etc/security/rules.int มีอยู่ไฟล์จะโหลดฐานข้อมูลกฎลงในเคอร์เนล

ขั้นตอนที่เรียกใช้งานในสคริปต์ rc.mls คือ:

1. เตรียมข้อมูลเบื้องต้น Trusted AIX sub-system
2. ถ้าไฟล์ /etc/security/rules.int มีอยู่ไฟล์จะโหลดฐานข้อมูลกฎลงในเคอร์เนล

หมายเหตุ: การเปลี่ยนแปลงกับสคริปต์บูตอาจมีผลให้ระบบทำงานผิดปกติได้

การกำหนดการเริ่มทำงานระบบเอง:

แม้ว่าจะไม่แนะนำ การพิสูจน์ตัวตนเมื่อบูตและการตรวจสอบ integrity ระบบ เมื่อระบบเริ่มทำงานสามารถถูกปิดใช้งานได้

โอเปอเรเตอร์ต้องอยู่ที่หน้าคอนโซลระบบเพื่อเริ่มต้น ระบบนอกจาการพิสูจน์ตัวตนเมื่อบูตและการตรวจสอบ integrity ระบบ ถูกปิดใช้งาน

การปิดใช้งานการพิสูจน์ตัวตน BOOT:

การพิสูจน์ตัวตน BOOT สามารถถูกปิดใช้งานโดยการรันคำสั่ง **rmitab bootauth** หรือการใช้เมนู SMIT

การปิดใช้งานการตรวจสอบ integrity ระบบ:

คุณสามารถปิดใช้งานการตรวจสอบ integrity บูตระบบอัตโนมัติโดยการลบ บรรทัด trustchk จากสคริปต์ **rc.mls.boot**

การปิดระบบ

การปิดระบบเป็นการดำเนินการ privileged และถูกป้องกันโดยการอนุญาต aix.system.boot.shutdown

ผู้ใช้ที่มีบทบาท SO หรือบทบาทอื่นที่มีการอนุญาตนี้ สามารถปิดระบบได้

การกู้คืนการไว้วางใจ

อาจมีบางครั้งที่เครื่องปิดในสถานะ unclean ซึ่งอาจเป็นผลจากไฟฟ้าดับ เครื่องปิดโดยอุบัติเหตุ หรือความขัดข้องของฮาร์ดแวร์ Trusted AIX สามารถกู้คืน จากเหตุการณ์เหล่านี้โดยไม่ต้องมีขั้นตอนการบูตใหม่พิเศษ

เมื่อระบบรีบูต กลไกการปกป้องแอ็คทีฟทั้งหมด ไม่ว่า ระบบจะถูกปิดลงอย่างไร ระหว่างกระบวนการเริ่มต้นระบบ ระบบไฟล์ทั้งหมดถูกตรวจสอบความเสียหายโดยอัตโนมัติ ก่อนที่ผู้ใช้จะสามารถล็อกอินได้ สคริปต์เริ่มทำงานรันคำสั่ง **fsck** เพื่อรักษาความปลอดภัยหรือ ทำให้เข้าถึงไม่ได้จากผู้ใช้ที่ไม่ได้รับอนุญาต ไฟล์ที่เสียหายหรือไฟล์ที่ถูกทำให้มีช่องโหว่

คำสั่ง **trustchk** รายงานความไม่สอดคล้องกันทั้งหมดใน แอ็ตทริบิวต์การรักษาความปลอดภัยของไฟล์หรือไดเร็กทอรี และพร้อมต์ แบบมีการโต้ตอบกับผู้ใช้เพื่อแก้ไขแอ็ตทริบิวต์เหล่านี้ คำสั่ง **trustchk** ควรถูก รันเมื่อใดก็ตามที่เป็นไปได้ ที่ integrity ของระบบไฟล์ อาจถูกทำให้มีช่องโหว่ ดูที่คำสั่ง **trustchk** สำหรับข้อมูลเพิ่มเติม

ล็อกอิน

ผู้ใช้ Trusted AIX ทุกคน ควรได้รับการกำหนด ระดับความลับ และ integrity clearances เพื่อที่ จะสามารถล็อกอินเข้าสู่ระบบ

clearances ของผู้ใช้ถูกกำหนดเป็นแอ็ตทริบิวต์ผู้ใช้ในไฟล์ /etc/security/user แอ็ตทริบิวต์ minsl และ maxsl กำหนด clearance ระดับความลับของผู้ใช้ แอ็ตทริบิวต์ mintl and maxtl กำหนด integrity clearance สำหรับผู้ใช้ แอ็ตทริบิวต์ defsl และ deftl กำหนดระดับ effective sensitivity และ integrity ของผู้ใช้เมื่อล็อกอิน

แอ็ตทริบิวต์ clearance ของผู้ใช้สามารถถูกแก้ไขด้วยคำสั่ง chuser และ chsec แสดงสามารถถูกแสดงด้วยคำสั่ง lsuser และ lssec

ผู้ใช้สามารถแสดงเลเบลของตัวเองแต่ไม่สามารถเปลี่ยนแปลงได้ เมื่อต้องการแสดงระดับ clearance ของผู้ใช้อื่น ผู้ใช้ต้องมีการอนุญาต aix.mls.clear.read เมื่อต้องการแก้ไข clearances ผู้ใช้ต้องมีการอนุญาต aix.mls.clear.write

เมื่อต้องการล็อกอิน กฎการควบคุมดังต่อไปนี้ทั้งหมดต้องเป็นจริง:

- ค่า minsl ต้องถูกควบคุมโดยค่า defsl
- ค่า defs ต้องถูกควบคุมโดยค่า maxsl
- ค่า mintl ต้องถูกควบคุมโดยค่า deftl
- ค่า deftl ต้องถูกควบคุมโดยค่า maxtl

คุณสามารถระบุระดับ effective sensitivity และ integrity ที่ต้องการ ระหว่างล็อกอินโดยใช้ตัวเลือก **–e** และ **–t** ของคำสั่ง **login** ดูที่คำสั่ง **login** สำหรับข้อมูลเพิ่มเติม

เมื่อต้องการล็อกอินที่ระดับ sensitivity ที่ไม่อยู่ในขอบเขตการกำหนดค่า ของระบบ คุณต้องมีการอนุญาต aix.mls.label.outsideaccred

Trusted AIX ไม่อนุญาตให้ผู้ใช้ระบบ (ผู้ใช้ที่มี uid น้อยกว่า 128) ให้ล็อกอิน

เหตุผลความล้มเหลวของล็อกอิน

การล็อกอินล้มเหลวได้จากหลายสาเหตุ

การล็อกอินจะล้มเหลวถ้าเงื่อนไขดังต่อไปนี้เป็นจริง:

- มีการป้อนล็อกอิน ID ไม่ถูกต้อง
- มีการป้อนรหัสผ่านไม่ถูกต้อง
- บัญชีผู้ใช้ถูกทำเครื่องหมายเป็นล็อกเนื่องจากจำนวนการล็อกอิน ที่ไม่ถูกต้องก่อนหน้านี้ เกินที่ระบบจำกัด
- ล็อกอินพอร์ตถูกทำเครื่องหมายเป็นล็อกเนื่องจากจำนวนการล็อกอิน ที่ไม่ถูกต้องสำหรับพอร์ต เกินที่ระบบจำกัด
- ล็อกอิน ID มี clearance ที่ไม่ถูกต้อง
- เลเบลที่ระบุ (หรือดีฟอลต์เลเบล sensitivity หรือ integrity สำหรับ ล็อกอิน ID ถ้าไม่มีการระบุเลเบล) ไม่ถูกต้อง ไม่ได้อยู่ใน clearance สำหรับล็อกอิน ID ไม่ได้อยู่ใน clearance สำหรับอุปกรณ์ล็อกอิน หรือไม่ได้ อยู่ในขอบเขตข้อบังคับของระบบ
- ผู้ใช้ไม่มีการเข้าถึง DAC กับชื่อพาธของล็อกอินเชลล์โปรแกรม หรือบัญชีผู้ใช้ไม่มีการเข้าถึง DAC exec กับล็อกอินเชลล์โปรแกรม
- ผู้ใช้ไม่มีการเข้าถึงเพื่ออ่าน MAC หรือ MIC กับชื่อพาธของล็อกอิน เชลล์โปรแกรม หรือไม่มีการเข้าถึงเพื่ออ่าน MAC หรือ MIC กับล็อกอินเชลล์โปรแกรม
- uid ของล็อกอิน ID น้อยกว่า 128

การสลับผู้ใช้ด้วยคำสั่ง su

บนระบบ Trusted AIX เมื่อคำสั่ง **su** พร้อมกับตัวเลือก – ถูกเรียก clearances ของผู้ใช้ปัจจุบันต้องควบคุมระดับ clearance ของผู้ใช้ใหม่

เงื่อนไขดังต่อไปนี้ต้องตรงกันสำหรับทั้งเลเบล sensitivity และ integrity :

- clearance สูงสุดของผู้ใช้ปัจจุบันต้องควบคุม clearance สูงสุดของผู้ใช้ใหม่
- clearance ต่ำสุดของผู้ใช้ใหม่ต้องควบคุม clearance ต่ำสุดของผู้ใช้ปัจจุบัน
- clearance ที่มีผลของผู้ใช้ปัจจุบันต้องถูกควบคุมโดย clearance สูงสุดของผู้ใช้ใหม่ และต้องควบคุม clearance ต่ำสุดของผู้ใช้ใหม่

ความรับผิดชอบการรักษาความปลอดภัยของผู้ใช้

มีความรับผิดชอบที่ผู้ใช้ต้องรับทราบ เข้าใจ และปฏิบัติตาม ผู้ใช้ต้องเก็บรหัสผ่านเป็นส่วนตัว รายการการเปลี่ยนแปลง ในสถานะของผู้ใช้ รายงานการละเมิดการรักษาความปลอดภัยที่สงสัย และอื่นๆ

รหัสผ่าน

รหัสผ่านควรถูกจดจำไปและไม่ควรถูก เขียนลงในสื่อบันทึกใด ถ้าผู้ใช้อื่นได้รหัสผ่านไป สามารถทำให้เกิดช่องโหว่การรักษาความปลอดภัยของข้อมูลบนระบบ

การคุกคาม ที่ชัดเจนที่สุดกับการรักษาความปลอดภัยรหัสผ่าน คือรหัสผ่านมีช่องโหว่ วิธีที่ง่ายที่สุด ในการป้องกันบัญชีผู้ใช้จากการโจมตีโดยผู้ใช้ที่อาจพบ รหัสผ่านคือการเปลี่ยนรหัสผ่านเป็นระยะ รหัสผ่านควรถูกเปลี่ยน บ่อยครั้งเพียงพอที่จะลดความเป็นไปได้ในการสร้างช่องโหว่ ระหว่างอายุการใช้งานของรหัสผ่าน ยิ่งการใช้รหัสผ่านเดียวนานเท่าไหร่ โอกาสที่จะมีช่องโหว่ก็มากขึ้นเท่านั้น

ถ้าผู้ใช้ ได้รับอนุญาตให้เลือกรหัสผ่านเอง รหัสผ่านใหม่ต้องมีความยาวอย่างน้อยหก อักขระและต้องมีอักขระแบบตัวอักษรอย่างน้อยสองตัวและหนึ่ง อักขระแบบตัวเลข รหัสผ่านไม่ควรแสดงถึงแง่มุมส่วนตัวหรืออาชีพของผู้ใช้ (ตัวอย่างเช่น เพื่อน ชื่อผู้ใช้ ชื่อสัตว์เลี้ยง หรือตำแหน่ง) และไม่ควรเป็นคำธรรมที่พบได้ในพจนานุกรม รูปแบบการเดารหัสผ่านส่วนมากทำการสแกนพจนานุกรมและรายการ สิ่งของส่วนตัว เช่นชื่อผู้ใช้ ชื่อบุตรหรือสัตว์เลี้ยง และวันเกิด

รหัสผ่านสามารถมีระยะใช้งานที่จำกัด ซึ่งกำหนด โดย ISSO ถ้ารหัสผ่านหมดอายุและผู้ใช้พยายามล็อกอิน ผู้ใช้ จะได้รับแจ้งว่าต้องเปลี่ยนรหัสผ่านและผู้ใช้ได้รับอนุญาตให้ล็อกอิน นอกจากรหัสผ่านถูกเปลี่ยนแปลง ขอแนะนำให้เปลี่ยนรหัสผ่านผู้ใช้บ่อยครั้งกว่า อายุของรหัสผ่านที่กำหนด ถ้ามี ข้อสงสัยว่ารหัสผ่านผู้ใช้อาจมีช่องโหว่ ควรเปลี่ยน รหัสผ่านทันที

การเปิดเครื่องทิ้งไว้

คุณไม่ควรปล่อยเครื่อง เปิดทิ้งไว้ ขณะที่มีผู้ใช้ล็อกอินในแอ็คทีฟเซสชัน ถ้าคุณต้องลุกไปจากหน้าเครื่องแม้เป็นระยะเวลาสั้นๆ ขอแนะนำ ให้คุณล็อกออฟจากระบบก่อน

การจัดการระบบที่ปลอดภัย

การจัดการระบบคอมพิวเตอร์ที่ปลอดภัย เกี่ยวข้องกับการสร้างและ การบังคับใช้นโยบายความปลอดภัยและการมอนิเตอร์ระบบเป็นประจำ

รายการดังต่อไปนี้ควรใช้เป็นจุดเริ่มต้นสำหรับการพัฒนา นโยบายการจัดการการทำงานเรื่องความปลอดภัยสำหรับไซต์ของคุณ:

- ระดับความปลอดภัยสูงสุดในขอบเขตการใช้ในระบบไม่ควรมากกว่า ระดับความปลอดภัยสูงสุดสำหรับไซต์ซึ่ง ระบบตั้งอยู่
- ฮาร์ดแวร์ระบบควรอยู่ในสถานที่ปลอดภัย สถานที่ปลอดภัยที่สุด โดยทั่วไปคือห้องชั้นในที่ไม่ได้อยู่ชั้นล่าง
- การเข้าถึงระบบฮาร์ดแวร์ควรถูกจำกัด เฝ้าระวัง และจัดทำเอกสาร
- การสำรองข้อมูลระบบและสื่อเก็บถาวรควรถูกเก็บในสถานที่ปลอดภัย แยกจากไซต์ฮาร์ดแวร์ระบบ การเข้าถึงสถานที่นี้ควรถูกจำกัด ในแบบเดียวกับการเข้าถึงฮาร์ดแวร์ระบบ
- การเข้าถึงคู่มือปฏิบัติการและเอกสารการดูแลระบบ ควรถูกจำกัด สำหรับผู้ที่เกี่ยวข้อง
- การรีบูตระบบ ไฟฟ้าดับ และการปิดระบบควรถูกบันทึก ระบบไฟล์ เสียหายควรถูกทำเอกสารและไฟล์ที่ได้รับผลทั้งหมดควรถูกวิเคราะห์ เพื่อหาการละเมิดนโยบายการรักษาความปลอดภัยที่เป็นไปได้
- การติดตั้งโปรแกรมใหม่ ไม่ว่าจากการอิมพอร์ตหรือสร้าง ควรถูกจำกัด และเฝ้าระวัง โปรแกรมใหม่ควรถูกวิเคราะห์และทดสอบอย่างระมัดระวังก่อน ถูกรัน
- การทำงานที่ไม่ปกติหรือไม่คาดคิดของซอฟต์แวร์ระบบควรถูกบันทึกเป็นเอกสาร และรายงาน และสาเหตุของการทำงานที่พบ
- เมื่อใดที่เป็นไปได้ อย่างน้อยควรมีสองคนดูแลระบบ คนหนึ่ง ควรมีบทบาท `isso` และอีกคนหนึ่งควรมี บทบาท `sa`
- ไม่ควรใช้ `PV_ROOT` privilege สำหรับผู้ดูแล ระบบ การใช้งานโปรแกรม privileged โดยผู้ใช้ ISSO, SA, or SO ควรเพียงพอ
- ข้อมูลการตรวจสอบควรถูกเก็บในล็อกและตรวจทานเป็นระยะ เหตุการณ์ที่ไม่ธรรมดาหรือไม่ปกติควรถูกบันทึกและตรวจหาสาเหตุ
- จำนวนการล็อกอินด้วยบทบาท `isso`, `sa`, และ `so` ควรถูกลดลงให้น้อยที่สุด
- จำนวนของโปรแกรม setuid และ setgid ควรถูกลดลงและควรถูก ใช้ในระบบย่อยที่มีการป้องกัน
- Privileges ที่กำหนดให้กับโปรแกรมใหม่ควรถูกกำหนดและลดให้น้อยที่สุด โดยการตรวจทาน Privileges ที่กำหนดให้กับโปรแกรมที่มีอยู่
- แอ็ตทริบิวต์การรักษาความปลอดภัยของไฟล์และไดเร็กทอรีควรถูกตรวจสอบเป็นระยะ ด้วยคำสั่ง **trustchk**
- รหัสผ่านทั้งหมดควรมีอย่างน้อย 8 อักขระ ซึ่งควรถูกตรวจสอบ โดยผู้ใช้ ISSO เป็นระยะ
- ผู้ใช้ทั้งหมดควรมีดีฟอลต์ล็อกอินเชลล์ที่ถูกต้อง ซึ่งควรถูกตรวจสอบ โดยผู้ใช้ SA เป็นระยะ
- ID ผู้ใช้ของผู้ใช้ปกติไม่ควรเป็น ID ระบบ ซึ่งควรถูกตรวจสอบ โดยผู้ใช้ SA เป็นระยะ ID ระบบคือที่มี uid น้อยกว่า 128

การตั้งค่าระบบ:

บางขั้นตอนต้องทำโดย ISSO และ SA เพื่อตั้งค่าระบบ อย่างถูกต้อง ISSO มีหน้าที่หลักในการจัดการการรักษาความปลอดภัย ขณะ ที่ SA มีหน้าที่หลักในการดูแลการทำงานรายวัน

ISSO ดำเนินงานดังต่อไปนี้:

- ติดตั้งและตั้งค่าการทำงานการรักษาความปลอดภัยพื้นฐานรวมถึง การตรวจสอบ ระบบ การจัดการบัญชีผู้ใช้ และการรักษาความปลอดภัยสำหรับอุปกรณ์ที่จัดสรรได้
- แก้ไขสคริปต์เริ่มต้นระบบในไฟล์ `/etc/rc.mls` และ `/etc/rc.mls.boot` เพื่อให้ตรงกับนโยบายความปลอดภัยไซต์

  หมายเหตุ: การเปลี่ยนแปลงที่ทำกับสคริปต์เริ่มต้นระบบ ไม่ได้เป็นส่วนหนึ่งของคอนฟิกูเรชันที่ประเมินค่าและต้องถูกระบุ ก่อนการรับรองระบบ
- ตั้งค่าพารามิเตอร์ล็อกอินทั้งระบบ
- ตั้งค่าพารามิเตอร์รหัสผ่านทั้งระบบ

- ตั้งค่าขอบเขต SL สำหรับอุปกรณ์ tty ที่อนุญาตให้ผู้ใช้ล็อกอินไปที่ ขอบเขต SL ที่ระบุสำหรับพอร์ต tty ดูที่คำสั่ง **chsec** สำหรับข้อมูลเพิ่มเติม
- ตั้งค่า SL อุปกรณ์ระบบสำหรับเทปไดร์ฟและฟล็อปปีดิสก์ไดร์ฟ ดูที่คำสั่ง **setsecattr** สำหรับข้อมูลเพิ่มเติม
- ตั้งค่าคุณลักษณะความปลอดภัย site-configurable ของระบบ

  หมายเหตุ: การเปลี่ยนแปลงที่ทำกับ คุณลักษณะความปลอดภัยที่กำหนดค่าได้ไม่ได้เป็นส่วนหนึ่งของคอนฟิกูเรชันที่ประเมินค่าและต้องถูกระบุ ก่อนการรับรองระบบ การเปลี่ยน การตั้งค่าคอนฟิกูเรชันดีฟอลต์สามารถทำให้ระบบทำงานในโหมดที่มีความปลอดภัยน้อยลง
- ตั้งค่าฐานข้อมูลความปลอดภัยที่วางใจสำหรับ การบูตที่ไว้วางใจและการกู้คืนที่ไว้วางใจ ดูที่คำสั่ง **trustchk** สำหรับข้อมูลเพิ่มเติม
- ตั้งค่ากลุ่มผู้ใช้บนระบบ

ISSO และ SA ทำงานร่วมกันเพื่อตั้งค่าพรินเตอร์ SA ตั้งค่า พรินเตอร์สำหรับระบบและ ISSO ตั้งค่าขอบเขต SL สำหรับพรินเตอร์

เน็ตเวิร์กคอนฟิกูเรชัน:

ISSO มีหน้าที่หลักสำหรับการรักษาความปลอดภัยเน็ตเวิร์ก ขณะ ที่ SA มีหน้าที่หลักในการดูแลเน็ตเวิร์กรายวัน ISSO และ SA ทำงานร่วมกันเพื่อตั้งค่าเน็ตเวิร์กอย่างถูกต้อง

การรักษาความปลอดภัยเน็ตเวิร์กถูกตั้งค่าด้วยค่ากำหนดดีฟอลต์ระหว่างการติดตั้ง Trusted AIX และยังสามารถส่ง เลเบลระดับความลับไปที่โฮสต์ Trusted AIX อื่นบนเน็ตเวิร์ก ISSO ติดตั้งและตั้งค่าการทำงานของเน็ตเวิร์กระดับต้นที่จัดเตรียมมากับ ระบบ ISSO ตั้งค่าตารางเน็ตเวิร์กแล้วรันคำสั่ง **tninit** เพื่อบันทึกฐานข้อมูล

การเข้าถึงเน็ตเวิร์ก:

เมื่อเชื่อมต่อกับระบบ non-Trusted AIX ผ่านเน็ตเวิร์กหรือกับระบบ Trusted AIX ที่ไม่ใช้คุณลักษณะ Trusted Networking บางแอ็ตทริบิวต์ความปลอดภัยอาจไม่ ถูกส่งข้อมูลโดยระบบ non-Trusted AIX ในกรณีนี้ระบบ Trusted AIX ใช้กับกลไกความปลอดภัยดีฟอลต์ กลไกความปลอดภัยดีฟอลต์ ถูกสร้าง โดยผู้ดูแลระบบ

คอนฟิกูเรชันบัญชีผู้ใช้:

ISSO และ SA ทำงานร่วมกันในการตั้งค่าบัญชีผู้ใช้บน ระบบ ISSO มีหน้าที่หลักในการจัดการแอ็ตทริบิวต์ผู้ใช้ที่เกี่ยวข้องด้านความปลอดภัย และ SA มีหน้าที่หลักกับแอ็ตทริบิวต์ผู้ใช้อื่น

ISSO ดำเนินงานดังต่อไปนี้สำหรับแต่ละผู้ใช้:
- ตั้งค่า clearance ดูที่คำสั่ง **chsec** และ **chuser** commands สำหรับข้อมูลเพิ่มเติม
- การตั้งค่าบทบาทและการอนุญาต
- การตั้งค่ากลุ่มผู้ใช้
- เซ็ตระดับ clearance ไดเร็กทอรี home ดูที่คำสั่ง **settxattr** สำหรับข้อมูลเพิ่มเติม
- เซ็ตรหัสผ่าน
- เซ็ตมาสก์การตรวจสอบ

SA ดำเนินงานดังต่อไปนี้:

- การตั้งค่าบัญชีผู้ใช้
- แจ้ง ISSO ถึงบัญชีผู้ใช้ใหม่ที่ต้องการแอ็ตทริบิวต์ความปลอดภัย

คอนฟิกูเรชันระบบไฟล์:

ระบบ ไฟล์ส่วนใหญ่ถูกสนับสนุนบน Trusted AIX อย่างไรก็ตาม การสนับสนุนความปลอดภัย Trusted AIX ที่เกี่ยวกับแอ็ตทริบิวต์ส่วนขยายบนอ็อบเจ็กต์ระบบไฟล์มีอยู่เฉพาะบน JFS2 ที่มี EAv2

ระบบไฟล์ JFS2 ที่มี EAv1 ถูกแปลงเป็น EAv2 เมื่อถูกประกอบเข้ากับ ระบบ Trusted AIX ไฟล์บน ระบบไฟล์ JFS2 เหล่านี้ไม่มีแอ็ตทริบิวต์ความปลอดภัย ระบบใช้ แอ็ตทริบิวต์ `SYSTEM_LOW` ดีฟอลต์ในการเข้าถึงไฟล์เหล่านี้ แอ็ตทริบิวต์ ความปลอดภัยสามารถถูกเซ็ตบนไฟล์โดยคำสั่ง **settxattr**

ในสภาวะแวดล้อมเน็ตเวิร์ก ไดเร็กทอรีบนหนึ่งระบบสามารถถูกทำเครื่องหมายเป็น shared หมายถึงไดเร็กทอรีนั้นสามารถถูกประกอบเข้าและเข้าถึงบนระบบอื่นใน เน็ตเวิร์ก เหมือนกับเป็นไดเร็กทอรี root ของระบบไฟล์บนโลคัลดิสก์ พาร์ติชัน

ระบบไฟล์เป็นได้ทั้ง multilevel filesystem (MLFS) หรือ single-level filesystem (SLFS) แต่ละไฟล์อ็อบเจ็กต์ใน MLFS มีเลเบลของตัวเอง ซึ่งอ็อบเจ็กต์ทั้งหมด ใน SLFS มีเลเบลเหมือนกับจุดประกอบเข้า SLFS ไม่สนับสนุน ไดเร็กทอรีหลายระดับและไดเร็กทอรีที่พาร์ติชัน

การเข้าถึงระบบไฟล์:

เมื่อกระบวนการพยายามเข้าถึงอ็อบเจ็กต์ระบบไฟล์ระบบ ตรวจสอบการเข้าถึงกับแต่ละคอมโพเนนต์ชื่อพาธ

ถ้ากระบวนการไม่มีการเข้าถึง search กับไดเร็กทอรีทั้งหมดในชื่อพาธ กระบวนการนี้ไม่สามารถเข้าถึงอ็อบเจ็กต์ได้ เมื่อชื่อพาธสัมพันธ์ถูกใช้ การเข้าถึงไดเร็กทอรีปัจจุบันถูกตรวจสอบไม่ว่าไดเร็กทอรีปัจจุบัน ถูกอ้างอิงอย่างชัดแจ้งโดยใช้เครื่องหมายจุด (`.`) ที่ ตอนต้นของชื่อพาธ

การจัดการเน็ตเวิร์กที่ไว้วางใจ:

มีข้อควรพิจารณาในการจัดการ Trusted Network รวมถึงคอนฟิกูเรชันฐานข้อมูล ไวยากรณ์ netrule และ ข้อกำหนดของกฎแฟล็ก Trusted Network และตัวเลือก RIPSO/CIPSO

คำเตือนดีฟอลต์คอนฟิกูเรชัน:

ความสามารถทางเน็ตเวิร์กของ AIX Trusted Network ได้ถูกออกแบบมาอย่างระมัดระวัง เพื่ออนุญาตการคอนฟิกูเรชันได้ตามต้องการ อย่างไรก็ตาม การเปลี่ยน คอนฟิกูเรชันจากค่าดีฟอลต์โดยไม่มีความเข้าใจ AIX Trusted Network เป็น สิ่งอันตรายได้

เป็นไปได้โดยการตั้งค่าเครื่องอย่างไม่ถูกต้อง ดาวน์เกรด โดยอัตโนมัติ อัพเกรด หรือลบข้อมูลความปลอดภัยทั้งหมด ดังนั้นคุณไม่ควรเปลี่ยนค่าดีฟอลต์ในตารางเน็ตเวิร์ก นอกจากคุณคุ้นเคยกับ AIX Trusted Network

ฐานข้อมูลคอนฟิกูเรชัน AIX Trusted Network:

เน็ตเวิร์กคอนฟิกูเรชันขณะบูตถูกสร้างโดย ไฟล์ `rules.host` และ `rules.int`

หลังจากการติดตั้ง Trusted AIX ดีฟอลต์, ไม่มีกฎของโฮสต์หรือไฟล์ของกฎ คำสั่ง **netrule** สามารถใช้กับแฟล็ก **–u** เพื่อบันทึก หรือ อัพเดตกฎกับไฟล์ ไฟล์เป็นฐานข้อมูลไบนารีที่สามารถถูก จัดการด้วยคำสั่ง **tninit** ผู้ใช้ต้องมี การอนุญาต `aix.mls.network.init` เพื่อใช้ คำสั่ง **tninit**

การแสดงฐานข้อมูลกฎ AIX Trusted Network:

เนื้อหาของชุดฐานข้อมูลกฎ AIX Trusted Network สามารถถูกแสดงด้วยการดำเนินการ **disp** ของคำสั่ง **tninit**

ป้อนคำสั่งดังต่อไปนี้เพื่อผนวกส่วนขยาย **.host** และ **.int** กับ *filename* เพื่อ สร้างชื่อไฟล์ของฐานข้อมูลกฎโฮสต์และฐานข้อมูลกฎ อินเตอร์เฟส เนื้อหาของทั้งสองไฟล์จะถูกส่งไปที่ out stream มาตรฐานในฟอร์มที่อ่านได้

```
tninit disp filename
```

ป้อนคำสั่งดังต่อไปนี้เพื่อแสดงบูตดีฟอลต์คอนฟิกูเรชัน:

```
tninit disp /etc/security/rules
```

การโหลดฐานข้อมูลกฎ AIX Trusted Network:

คำสั่ง **tninit** อ่านชุดของฐานข้อมูลกฎ AIX Trusted Network และโหลดคำสั่งมาไว้ในเคอร์เนลเพื่อให้กลายเป็นแอ็คทีฟเซ็ต ชื่อไฟล์ของตารางการกำหนดโฮสต์และอินเตอร์เฟส ถูกระบุ ในวิธีเดียวกับการดำเนินการ **tninit disp**

แฟล็กทางเลือก **–m** ระบุที่ระบบควรดูแล กฎโฮสต์ที่มีอยู่ ถ้าแฟล็ก **–m** ไม่ถูกระบุ กฎโฮสต์ที่มีอยู่ทั้งหมดถูกลบออกก่อน ชุดแอ็คทีฟเซ็ตใหม่ถูกโหลด ถ้าแฟล็ก **–m** ถูกระบุ ชุดกฎโฮสต์ใหม่และที่มีอยู่ ถูกรวบรวม กฎใหม่แทนที่กฎที่มีอยู่แล้ว ถ้ามีความขัดแย้งเกิดขึ้น กฎอินเตอร์เฟสทั้งหมดถูกแทนที่ไม่ว่าจะ มีการระบุแฟล็ก **–m** หรือไม่

คำสั่งดังต่อไปนี้โหลดกฎใหม่ขณะดูและชุดกฎ เก่า:

```
tninit -m load /dir/dir/filename
```

คำสั่ง นี้ใช้ไฟล์ที่ระบุโดยพารามิเตอร์ *filename* และ ผนวกส่วนขยาย **.host** และ **.int** เพื่อสร้างสอง ไฟล์ที่รวมถึงฐานข้อมูล

การบันทึกฐานข้อมูลกฎ AIX Trusted Network:

ซีแมนทิกส์ที่เหมือนกันถูกใช้สำหรับการโหลดและการบันทึก ฐานข้อมูลกฎ

ชื่อไฟล์ที่ระบถูกผนวกกับ **.int** และ **.host** เพื่อ สร้างสองไฟล์ที่ใช้เก็บฐานข้อมูล การดำเนินการบันทึกของคำสั่ง **tninit** เก็บกฎทั้งหมดที่แอ็คทีฟอยู่ใน เคอร์เนล

เมื่อต้องการสร้างชุดกฎดีฟอลต์ คุณต้องใช้คำสั่ง **netrule** เพื่อปรับกฎเคอร์เนลให้เหมาะกับนโยบายความปลอดภัยไซต์ที่ต้องการ แล้วรันคำสั่ง **tninit** คำสั่งดังต่อไปนี้ สร้างไฟล์ `/etc/security/rules.int` และ `/etc/security/rules.host`:

```
tninit save /etc/security/rules
```

AIX Trusted Network เคอร์เนลคอนฟิกูเรชัน:

คุณสามารถใช้คำสั่ง **netrule** เพื่อตั้งค่า ชุดกฎ AIX Trusted Network ของเคอร์เนลโดยสมบูรณ์เพื่อให้เหมาะสมกับนโยบายความปลอดภัยของไซต์ถ้าคุณมีการอนุญาต `aix.mls.network.config`

คำสั่ง **netrule** สามารถถูกใช้เพื่อจัดการ ทั้งกฎโฮสต์และเน็ตเวิร์กในเคอร์เนล ดูที่คำสั่ง **netrule** สำหรับข้อมูลเพิ่มเติม

แต่ละอินเตอร์เฟสในระบบต้องมีกฎเชื่อมโยงด้วย ถ้าคุณจะลบกฎอินเตอร์เฟส กฎกลับค่าไปอยู่ในสถานะ ดีฟอลต์ ถ้าคุณเพิ่มกฎอินเตอร์เฟสอื่น กฎอินเตอร์เฟสใหม่เขียนทับ กฎปัจจุบัน กฎอินเตอร์เฟสดีฟอลต์สามารถดูได้โดยการเคียวรี กฎอินเตอร์เฟสที่มีชื่ออินเตอร์เฟสเป็น “default” ตัวอย่างเช่น : # `netrule iq default`

ไวยากรณ์ netrule:

มีกฎไวยากรณ์โฮสต์และอินเตอร์เฟส สำหรับคำสั่ง **netrule**

คำสั่ง **netrule** มีกฎไวยากรณ์ดังต่อไปนี้เมื่อใช้ สำหรับโฮสต์:

netrule h l [**i** | **o** | **io**]

netrule h q { **i** | **o** } *src_host_rule_specification dst_host_rule_rule_specification*

netrule h – [{ **i** | **o** } [**u**] [*src_host_rule_specification dst_host_rule_specification*]

netrule h + { **i** | **o** } [**u**] *src_host_rule_specification dst_host_rule_specification* [*flags*] [*RIPSO/CIPSO_options*] *security*

คำสั่ง **netrule** มีกฎไวยากรณ์ดังต่อไปนี้เมื่อใช้ สำหรับอินเตอร์เฟส:

netrule i l

netrule i q *interface*

netrule i + [**u**] *interface* [*flags*] [*RIPSO/CIPSO_options*] *security*

องค์ประกอบแรก `h` หรือ `i` บ่งชี้ การดำเนินการโฮสต์หรือเน็ตเวิร์กอินเตอร์เฟส

การดำเนินการที่ต้องการแสดงถัดไป มีสี่การดำเนินการ:

l แสดงกฎทั้งหมด

q เคียวรีกฎ

– ลบกฎโฮสต์หรือกลับกฎอินเตอร์เฟส สู่สถานะดีฟอลต์

+ การเพิ่มหรือแทนที่กฎ

องค์ประกอบที่สามในกฎโฮสต์ระบุชนิดกฎ สำหรับกฎโฮสต์ มีความแตกต่างระหว่างกฎกฎและขาออก กฎ in ใช้กับ แพ็กเก็ตขาเข้าทั้งหมด ขณะที่กฎ out ใช้กับแพ็กเก็ตขาออกทั้งหมด; `i` แสดงถึง กฎ in, `o` แสดงถึงกฎ out และเมื่อสามารถใช้ได้ `io` หรือว่างเปล่า หมายถึงทั้งกฎ in และ out ถ้ามีการระบุองค์ประกอบสุดท้าย `u` เมื่อเพิ่มหรือลบกฎโฮสต์หรืออินเตอร์เฟสไฟล์ `/etc/security/rules.host` และ `/etc/security/rules.int` ถูก อัพเดตหลังจากกฎโฮสต์หรืออินเตอร์เฟส ถูกเพิ่มหรือลบเสร็จสมบูรณ์

ค่ากำหนดกฎ AIX Trusted Network:

กฎอินเตอร์เฟสต้องการให้คุณป้อนชื่อของ อินเตอร์เฟสเน็ตเวิร์ก กฎโฮสต์มีความยืดหยุ่นมากกว่าดังนั้น ต้องการค่ากำหนดกฎที่ซับซ้อนกว่า

เมื่อต้องการระบุอินเตอร์เฟส ป้อนชื่อของอินเตอร์เฟสเน็ตเวิร์ก ที่จะใช้กฎ ชื่อเน็ตเวิร์กอินเตอร์เฟสมีชื่อเช่น en0 คุณสามารถใช้คำสั่ง **ifconfig -a** เพื่อดู ชื่ออินเตอร์เฟสเน็ตเวิร์ก คุณต้องระบุอินเตอร์เฟสตามชื่อ เท่านั้น คุณไม่สามารถระบุพอร์ต โปรโตคอลหรือ subnet mask

กฎโฮสต์ต้องการค่ากำหนดกฎที่ซับซ้อนกว่า ระบบ AIX Trusted Network ใช้กฎที่เจาะจงที่สุด ตัวอย่างเช่น นโยบายไซต์สามารถถูกตั้งค่าเพื่อให้กฎโฮสต์ที่มีมาสก์ 24 ใช้กับโฮสต์ ทั้งหมดบน subnet แต่กฎที่เจาะจงมากกว่าสามารถใช้กับโฮสต์เดี่ยวบนเน็ต และโฮสต์นี้ใช้กฎที่เจาะจงมากกว่า กฎ ที่เจาะจงมากกว่ายังสามารถใช้กับหนึ่งพอร์ต TCP เจาะจงบน โฮสต์นี้ ความยืดหยุ่นของ AIX Trusted Network คอนฟิกูเรชัน ให้ความสามารถแก่คุณในการสร้างนโยบายความปลอดภัย ไซต์อะไรก็ตามที่จำเป็นสำหรับแอ็พพลิเคชัน ไวยากรณ์ คือ:

source_host [/*mask*] [=*proto*] [:*start_port_range* [:*end_port_range*]]

destination_host [/*mask*] [=*proto*] [:*start_port_range* [:*end_port_range*]]

source_host
ชื่อโฮสต์ IPv4 แอดเดรสหรือ IPv6 แอดเดรสของซอร์สโฮสต์

destination_host
ชื่อโฮสต์ IPv4 แอดเดรสหรือ IPv6 แอดเดรสของ โฮสต์ปลายทาง

mask subnet mask หมายเลขระบุจำนวนบิตจาก MSB สัมพันธ์กัน เมื่อคู่ IPv4 address/subnet ถูกเขียน *a.b.c.d/e*, *e* คือตัวเลขจาก 0 ถึง 32 ตัวเลขนี้ระบุจำนวนของตัวเลขที่จุดเริ่มต้น ของ subnet mask ตัวอย่างเช่น สำหรับ IPv4 แอดเดรส /24 ระบุ netmask ของ 255.255.255.0, ซึ่งเมื่อดูในแบบ 32 บิตจะเป็น 11111111.11111111.11111111.00000000 นี่คือตัวเลข 24 ตัวตามด้วยศูนย์แปดตัว

proto หมายเลขโปรโตคอลหรือชื่อตามที่บันทึกในไฟล์ /etc/protocols (ตัวอย่างเช่น, =tcp)

start_port_range
พอร์ต TCP หรือ UDP ซึ่งใช้กฎ หรือจุดเริ่มต้นของ ขอบเขต ถ้ากฎใช้กับขอบเขตของพอร์ต ซึ่งสามารถเป็น หมายเลขพอร์ตหรือชื่อของเซอร์วิส UDP หรือ TCP ตามที่บันทึก ในไฟล์ /etc/services

end_port_range
ขอบเขตบนของช่วงพอร์ต

AIX รายละเอียดแฟล็ก Trusted Network:

ระบบ AIX Trusted Network มีสองแฟล็กคลัสเตอร์ ถ้าไม่มีการระบุข้อมูลเหล่านี้ จะใช้ค่าดีฟอลต์

แฟล็ก **-d** และ **-r** ถูกใช้ดังนี้:

-d *drop*

drop AIX Trusted Network สามารถถูกตั้งค่าเพื่อลบแพ็กเก็ตทั้งหมด

r ลบแพ็กเก็ตทั้งหมดบนอินเตอร์เฟสนี้

n ไม่ลบแพ็กเก็ตทั้งหมดบนอินเตอร์เฟสนี้โดยอัตโนมัติ (อินเตอร์เฟส ดีฟอลต์)

i ใช้อินเตอร์เฟสดีฟอลต์ (โฮสต์ดีฟอลต์ โฮสต์เท่านั้น)

-f*rflag*:*tflag*

rflag ข้อกำหนดตัวเลือกความปลอดภัยในแพ็กเก็ตขาเข้า (ได้รับ)

r RIPSO เท่านั้น

c CIPSO เท่านั้น

e CIPSO หรือ RIPSO

n ไม่ใช่ทั้ง CIPSO หรือ RIPSO (ดีฟอลต์ระบบ)

a ไม่มีข้อจำกัด

i ใช้ อินเทอร์เฟส/ระบบ ดีฟอลต์ (ดีฟอลต์)

tflag การจัดการตัวเลือกความปลอดภัยบนแพ็กเก็ตขาออก (ส่ง)

r RIPSO ที่กำหนดตำแหน่งบนส่วนหัว IP แพ็กเก็ตขาออกทั้งหมด

c CIPSO ที่กำหนดตำแหน่งบนส่วนหัว IP แพ็กเก็ตขาออกทั้งหมด

i ใช้อินเตอร์เฟสดีฟอลต์ (โฮสต์ดีฟอลต์โฮสต์เท่านั้น)

ตัวเลือก RIPSO/CIPSO:

ระบบย่อย AIX Trusted Network สนับสนุนตัวเลือกสำหรับ configuration ของการเลเบลแพ็กเก็ต CIPSO และ RIPSO

-rpafs=*PAF_field* [, *PAF_field* ...]

ระบุแต่ละ *PAF_field* ที่ยอมรับเมื่อได้รับแพ็กเก็ต IPSO ฟิลด์นี้มีได้ถึง 256 ฟิลด์

-epaf=*PAF_field*

ระบุ *PAF_field* ที่ถูกแนบกับการตอบกลับข้อผิดพลาด เมื่อ แพ็กเก็ตข้อผิดพลาดถูกส่งโดยใช้ IPSO บนแพ็กเก็ตที่ส่ง

-tpaf=*PAF_field*

ระบุ *PAF_field* ที่ถูกใช้กับแพ็กเก็ตขาออกเมื่อ IPSO ถูกใช้ในแพ็กเก็ตที่ส่งผ่าน

PAF_field:**NONE** | *PAF* [+ *PAF* ...]

PAF_field เป็นคอลเล็กชันของ *PAF* มีห้า *PAF* ที่สามารถถูกรวมไว้ใน *PAF_field* เดียว มี **GENSER**, **SIOP-ESI**, **SCI**, **NSA**, และ **DOE** *PAF_field* คือการรวมกันของค่าเหล่านี้คั่นด้วย เครื่องหมายบวก (+) ตัวอย่างเช่น *PAF_field* มีทั้ง **GENSER** และ **SCI** ถูกแสดง เป็น **GENSER+SCI** *PAF_field* **NONE** พิเศษสามารถ ใช้ได้; ซึ่งระบุ *PAF_field* โดยไม่เซ็ต *PAF*

-DOI=*doi*

ระบุโดเมนของการแปลสำหรับแพ็กเก็ต CIPSO แพ็กเก็ต CIPSO ขาเข้าต้องมี **DOI** นี้และแพ็กเก็ต CIPSO ขาออกจะถูกเลเบลด้วย **DOI** นี้

-tags=*tag*[,*tag* ...]

tag=**1** | **2** | **5**

ระบุชุดของแท็กที่ได้รับและพร้อมที่จะถูกส่งโดย ตัวเลือก CIPSO นี่เป็นการรวมกันของ **1**, **2** และ **5** คั่น ด้วยคอมมา ตัวอย่าง เช่น **1**,**2** จะเปิดแท็ก **1** และ **2**

นโยบายความปลอดภัย AIX Trusted Network:

minimum SL และ maximum SL ใช้ได้ และ ดีฟอลต์ SL ต้องถูกระบุ

ค่าโดยนัยหรือดีฟอลต์ SL ถูกใช้กับแพ็กเก็ตทั้งหมดที่ไม่มี ข้อมูลเกี่ยวกับ SL ของตัวเอง ระดับถูกป้อนใน ไวยากรณ์ดังต่อไปนี้:

+min +max +default

เลเบลที่ใช้ได้ตามไฟล์การเข้ารหัสเลเบลสามารถ ใช้ได้ ไม่จำเป็นต้องมีเครื่องหมายคำพูดสำหรับเลเบลที่มีช่องว่าง

ตัวอย่าง netrule:

ต่อไปนี้เป็นตัวอย่างของคำสั่ง **netrule**

ป้อนข้อมูลดังต่อไปนี้เพื่อตั้งค่า **en0** เพื่อส่งตัวเลือกที่ไม่มีการรักษาความปลอดภัย และอนุญาตให้แพ็กเก็ตทั้งหมดผ่าน:

```
netrule i+ en0 +impl_lo +ts all +impl_lo
```

ป้อนข้อมูลดังต่อไปนี้เพื่อตั้งค่าโฮสต์ **185.0.0.62** ให้ยอมรับเฉพาะแพ็กเก็ต CIPSO ภายในขอบเขต **CONFIDENTIAL A** ถึง **TOP SECRET ALL**:

```
netrule h+i 192.168.0.0 /24 185.0.0.62 -fc:c +confidential a +top secret all +confidential a
```

ป้อนข้อมูลดังต่อไปนี้เพื่อลบแพ็กเก็ตเทลเน็ตทั้งหมดจาก subnet:

```
netrule h+i 192.168.0.0 /24 =tcp :telnet 192.0.0.5 -dr +impl_lo +impl_lo +impl_lo
```

ดูที่คำสั่ง **netrule** สำหรับข้อมูลเพิ่มเติมและตัวอย่าง

การจัดการบัญชีผู้ใช้:

ข้อมูล Identification และ authentication (I&A) เกี่ยวกับแต่ละผู้ใช้ ถูกป้องกันและถูกใช้เพื่อเป็นค่าเฉพาะในการระบุที่ใช้และตรวจสอบ สิทธิการเข้าถึงของผู้ใช้ภายในระบบ

ข้อมูลการระบุผู้ใช้ประกอบด้วย ชื่อผู้ใช้ ชื่อข้อความล็อกอิน ID ID ผู้ใช้ ID กลุ่ม ไดเร็กทอรี home รหัสผ่าน พารามิเตอร์การกำหนดอายุรหัสผ่าน เชลล์ clearances การอนุญาต และมาสก์การตรวจสอบ ข้อมูลเกี่ยวกับผู้ใช้ส่วนใหญ่ ถูกเก็บในไฟล์ดังต่อไปนี้:

`/etc/passwd`
: ชื่อผู้ใช้ ID ผู้ใช้ การกำหนดกลุ่มหลัก และไดเร็กทอรี home

`/etc/group`
: การกำหนดกลุ่มรองและไดเร็กทอรี home

`/etc/security/passwd`
: รหัสผ่านผู้ใช้ในกฟอร์มที่เข้ารหัส

`/etc/security/user`
: ข้อจำกัดล็อกอิน พารามิเตอร์รหัสผ่าน (เช่นความยาวต่ำสุด) umask และอื่นๆ

ไฟล์ `/etc/security/passwd` และ `/etc/security/user` ไม่สามารถอ่านได้โดยผู้ใช้ปกติ ไฟล์ `/etc/security/passwd` ถูกป้องกันด้วยการไม่เปิดบิตการเข้าถึงที่ระมัดระวัง และ SL ของ `SYSTEM_HIGH` การป้องกันผู้ใช้ปกติจากการอ่านรหัสผ่านที่เข้ารหัสขจัดรูทีน การเข้ารหัส/การเปรียบเทียบ ลำดับที่พยายามจับคู่รหัสผ่านที่เข้ารหัส

ผู้ใช้ที่ได้รับอนุญาตสามารถแก้ไขไฟล์เหล่านี้ได้โดยตรง แต่บ่อยครั้งที่สะดวกกว่าที่จะใช้ คำสั่ง **smit** เพื่อแก้ไขพารามิเตอร์ผู้ใช้ คำสั่ง **smit** ร้องขอ System Management Interface Tool (SMIT) ซึ่งแสดงเมนู พร้อมกับตัวเลือกสำหรับงานการจัดการระบบ เช่นการดูแลรักษาผู้ใช้

ID ผู้ใช้และ ID กลุ่ม:

มี ID ผู้ใช้สองคลาส : ID ระบบและ ID ผู้ใช้ปกติ ID ระบบถูกสำรองไว้สำหรับความเป็นเจ้าของของระบบย่อยที่ป้องกันและฟังก์ชัน การดูแลระบบ ID ผู้ใช้ปกติถูกกำหนดให้กับผู้ใช้ซึ่ง ใช้ระบบแบบตอบโต้

ผู้ใช้แต่ละคนมี ID ผู้ใช้เฉพาะที่ใช้เพื่อระบุผู้ใช้ในระบบ ผู้ใช้แต่ละคนยังสามารถถูกกำหนด ID กลุ่มหนึ่งกลุ่มหรือมากกว่านั้น ID กลุ่มถูกแบ่งใช้โดยผู้ใช้ในกลุ่มเดียวกันและไม่จำเป็นต้องเป็นค่าเฉพาะ มีการจำกัดขอบเขตในค่าตัวเลข ที่ใช้สำหรับ ID ตารางดังต่อไปนี้กำหนดการจำกัด ขอบเขต ID ค่าได้ถูกกำหนดให้อนุญาตสำหรับจำนวนที่เพียงพอ ของผู้ใช้ ระบบและปกติ และ ID กลุ่ม

ID ผู้ใช้ระบบ
: 0 ถึง 127

ID ผู้ใช้ปกติ
: 128 ถึง MAXUID

ID กลุ่มปกติ
: 0 ถึง MAXUID-1

ค่า MAXUID ถูกกำหนดในไฟล์ `/usr/include/sys/param.h`

ควรระวังเมื่อทำการกำหนดค่า ID ผู้ใช้สำหรับผู้ใช้ใหม่ ถ้าผู้ใช้ปกติ ถูกกำหนดค่า ID ผู้ใช้น้อยกว่า 128 ผู้ใช้ จะไม่สามารถล็อกอินเข้าระบบได้

ค่า ID ผู้ใช้ไม่ควรนำกลับมาใช้ใหม่ เมื่อผู้ใช้ถูกลบ ขอแนะนะให้ ล็อกรายการที่เหลือในไฟล์ `/etc/passwd` and `/etc/security/passwd` และบัญชีผู้ใช้ ทำได้โดยใช้คำสั่ง **smit** นี่ป้องกันผู้ใช้จากการล็อกอินและการนำ ID มาใช้ใหม่ การไม่นำ ID มาใช้ใหม่ป้องกันผู้ใช้ใหม่จากการเข้าถึงไฟล์ที่เป็นของผู้ใช้ก่อนหน้า และยังไม่ได้ถูกเอาออก ซึ่งยังอนุญาตการติดตามตรวจสอบ ให้ถูกสร้างไม่อย่างชัดเจน

ไฟล์ `/etc/passwd`, `/etc/security/passwd`, และ `/etc/group` สามารถถูกจัดการด้วยคำสั่ง **mkuser**, **chuser**, **rmuser**, **pwdadm** และ **passwd** คำสั่งเหล่านี้ บังคับข้อควรระวังด้านบนและข้อควรพิจารณาความปลอดภัยระบบ อื่นๆ คำสั่ง **mkuser** สามารถเพิ่มได้เพียงผู้ใช้ปกติ เข้าสู่ระบบ

หมายเหตุ: ระมัดระวังในการบังคับใช้มาตรฐานดังต่อไปนี้:

- อย่ากำหนด ID ผู้ใช้ที่ใช้แล้วก่อนหน้า ให้กับผู้ใช้ใหม่
- อย่ากำหนด ID ผู้ใช้ซ้ำซ้อน
- อย่ากำหนด ID ระบบให้กับผู้ใช้ปกติ
- อย่ากำหนด MAXUID เป็น ID ผู้ใช้หรือ ID กลุ่ม

รหัสผ่าน:

รหัสผ่านคือสตริงอักขระที่สัมพันธ์กับผู้ใช้ และถูกใช้เพื่อพิสูจน์ตัวตนผู้ใช้ขณะเริ่มเซสชัน

รหัสผ่านถูกเก็บในฟอร์มเข้ารหัสในไฟล์ shadow รหัสผ่านที่ไม่เข้ารหัสไม่ถูกเก็บในระบบ

หมายเหตุ: รหัสผ่านสำหรับผู้ใช้บทบาทสำคัญมากต่อความปลอดภัย ของระบบและควรถูกป้องกันตลอดเวลา

การกำหนดอายุรหัสผ่าน:

ผู้ใช้สามารถเปลี่ยนรหัสผ่านตราบที่เป็นไปตามเกณฑ์ การกำหนดอายุรหัสผ่าน

การกำหนดอายุรหัสผ่านต้องการให้ผู้ใช้เปลี่ยนรหัสผ่าน ถ้ารหัสผ่าน มีอยู่ในระบบตามระยะเวลาที่กำหนด การกำหนดอายุรหัสผ่านประกอบด้วย ระยะเวลาต่ำสุดและระยะเวลาสูงสุด รหัสผ่านไม่สามารถถูกเปลี่ยนแปลงก่อน ผ่านช่วงเวลาต่ำสุดนี้ รหัสผ่านต้องถูกเปลี่ยน หลังจากระยะเวลาสูงสุด

พารามิเตอร์การกำหนดอาจุรหัสผ่านสามารถถูกเซ็ตในไฟล์ `/etc/security/user` พารามิเตอร์ดังต่อไปนี้สัมพันธ์กับการกำหนดอายุรหัสผ่าน:

maxage
จำนวนสัปดาห์สูงสุดที่รหัสผ่านใช้ได้

maxexpired
จำนวนสัปดาห์สูงสุดหลังจาก maxage ที่รหัสผ่านที่หมดอายุสามารถถูกเปลี่ยน โดยผู้ใช้

minage จำนวนสัปดาห์ต่ำสุดระหว่างการเปลี่ยนรหัสผ่าน

minlen ความยาวต่ำสุดของรหัสผ่าน

พารามิเตอร์อื่นสามารถถูกเซ็ตเพื่อระบุอักขระที่ใช้ได้ใน รหัสผ่าน ดูที่คำสั่ง **passwd** สำหรับรายการสมบูรณ์ ของพารามิเตอร์รหัสผ่าน

เชลล์:

ขณะทำงานในแอ็พพลิเคชัน เช่นเวิร์ดโปรเซสเซอร์หรือ สเปร็ดชีท โดยปกติผู้ใช้จะไม่จำเป็นต้องติดต่อโดยตรงกับระบบปฏิบัติการ เนื่องจากแอ็พพลิเคชันจัดการการติดต่อนั้น อย่างไรก็ตาม ผู้ใช้บางคน จำเป็นต้องติดต่อโดยตรงกับระบบปฏิบัติการ โดยไม่มีอินเตอร์เฟสของ แอ็พพลิเคชัน

เมื่อจำเป็นต้องมีการโต้ตอบโดยตรงกับระบบปฏิบัติการ ผู้ใช้ต้องใช้ เชลล์โปรแกรม เชลล์โปรแกรมอนุญาตให้ผู้ใช้ป้อนคำสั่ง AIX และเข้าถึงไฟล์และไดเร็กทอรีได้โดยตรงและดำเนินการดำเนินการอื่น ผู้ใช้ทุกคน ต้องมีดีฟอลต์เชลล์โปรแกรมระบุไว้ในไฟล์ /etc/passwd ดีฟอลต์เชลล์โปรแกรมของผู้ใช้ (เช่น /bin/sh, /bin/csh, หรือ /bin/ksh) ถูกรันโดยคำสั่ง **login** หรือ **xterm** เมื่อผู้ใช้จำเป็นต้องใช้เชลล์

ล็อกอิน effective SL และ TL:

ผู้ใช้ถูกกำหนดดีฟอลต์ล็อกอิน SL และ TL ดีฟอลต์ล็อกอิน SL และ TL เป็น effective SL และ effective TL ของกระบวนการของผู้ใช้หลังจาก การล็อกอินสำเร็จ

ถ้าผู้ใช้ไม่ต้องการล็อกอินด้วยดีฟอลต์ล็อกอิน SL ผู้ใช้สามารถเลือก SL อื่นขณะล็อกอินโดยใช้ตัวเลือก **-e** ของคำสั่ง **login** SL ที่ระบุโดยผู้ใช้ต้องถูกควบคุมโดย clearance ของผู้ใช้ ที่มีในขอบเขตการแต่งตั้งของระบบ TL สามารถถูกระบุโดยผู้ใช้ ขณะล็อกอินโดยใช้ตัวเลือก **-t** ของคำสั่ง **login**

ดีฟอลต์ล็อกอิน SL และ TL ถูกกำหนดในไฟล์ /etc/security/user ตามด้วยชื่อผู้ใช้และ clearance สำหรับแต่ละผู้ใช้ effective SL ของ ผู้ใช้ต้องอยู่ระหว่างขอบเขต tty SL ตามที่ระบุในไฟล์ /etc/security/login.cfg effective SL ของผู้ใช้ต้องถูกควบคุมโดย maximum SL ของ tty และควบคุม minimum SL effective TL ของผู้ใช้ต้องเหมือนกับ TL tty

Clearances:

เชลล์กระบวนการผู้ใช้ถูกกำหนดเลเบลหกเลเบลระหว่างล็อกอิน

effective SL ถูกใช้โดยระบบในการตรวจสอบ MAC minimum SL clearance และ maximum SL clearance จำกัด effective SL; effective SL ไม่สามารถควบคุม maximum SL clearance และต้องควบคุม minimum SL effective TL ถูกใช้โดยระบบในการตรวจสอบ MIC minimum TL clearance และ maximum TL clearance จำกัด effective TL; effective TL ไม่สามารถควบคุม maximum TL clearance และต้องควบคุม minimum TL

ผู้ใช้ ISSO-authorized สามารถแก้ไข SL clearance ของผู้ใช้ TL clearance ดีฟอลต์ล็อกอิน SL และดีฟอลต์ล็อกอิน TL ค่าเหล่านี้ถูกกำหนดในไฟล์ /etc/security/user

การแบ่งหน้าที่สำหรับข้อมูลผู้ใช้:

ผู้ใช้เดียวไม่สามารถเพิ่มผู้ใช้ให้กับระบบ ผู้ใช้ถูกเพิ่ม เข้าระบบโดยการดำเนินการร่วมของผู้ใช้ SA- และ ISSO-authorized

ผู้ใช้ SA-authorized สามารถเพิ่มข้อมูลผู้ใช้ที่ไม่เกี่ยวกับความปลอดภัย ซึ่งประกอบด้วย ชื่อผู้ใช้ ID ผู้ใช้ ID กลุ่ม ชื่อข้อความล็อกอิน ID เชลล์ และไดเร็กทอรี home ผู้ใช้ ISSO-authorized สามารถเพิ่มข้อมูลผู้ใช้ที่เกี่ยวกับความปลอดภัย ซึ่งประกอบด้วย รหัสผ่านของผู้ใช้ clearance มาสก์การตรวจสอบ และบทบาท ความต้องการ บุคคลสองคนในการเพิ่มผู้ใช้ป้องกันผู้ใช้เดียวที่มีการอนุญาต ไม่ให้มอบการอนุญาต system-wide ให้แก่ผู้ใช้อื่น

ปรับปรุงที่เพิ่มประสิทธิภาพ:

Trusted AIX ได้ เพิ่มประสิทธิภาพระบบย่อยการตรวจสอบเพื่อตรวจจับรายละเอียดความปลอดภัยเพิ่มเติม

ฟิลด์เร็กคอร์ดการตรวจสอบใหม่:

ฟิลด์ดังต่อไปนี้ได้ถูกเพิ่มให้กับเร็กคอร์ดการตรวจสอบ AIX ทั้งหมดสำหรับ Trusted AIX ฟิลด์ใหม่เหล่านี้ถูกใช้กับคำสั่ง **auditselect** ตามเกณฑ์การเลือก

- บทบาทของกระบวนการที่ตรวจสอบ
- Effective TL ของกระบวนการหรืออ็อบเจ็กต์ที่ตรวจสอบ
- Effective SL ของกระบวนการหรืออ็อบเจ็กต์ที่ตรวจสอบ
- Effective privileges ของกระบวนการที่ตรวจสอบ

Trusted AIX ตรวจสอบ แอ็ตทริบิวต์ความปลอดภัยดังต่อไปนี้ในบางหลักฐานการตรวจสอบเช่นกัน:

- TL ของกระบวนการหรืออ็อบเจ็กต์ที่ตรวจสอบ
- SL ของกระบวนการหรืออ็อบเจ็กต์ที่ตรวจสอบ
- แฟล็กความปลอดภัย Trusted AIX-related

คุณสามารถแสดงแอ็ตทริบิวต์ความปลอดภัยใหม่เหล่านี้ด้วยคำสั่ง **auditpr -v**

ขอบเขตการตรวจสอบ:

Trusted AIX รวม กลไกที่อนุญาตให้ผู้ดูแลระบบระบุชุดของการตรวจสอบขอบเขต จาก TL และ/หรือ SL ของกระบวนการหรืออ็อบเจ็กต์ที่ตรวจสอบ อ็อบเจ็กต์ทั้งหมด และซับเจ็คต์ซึ่ง TL หรือ SL อยู่นอกขอบเขตการตรวจสอบจะถูกละเว้น

เมื่อต้องการเซ็ตขอบเขตการตรวจสอบสำหรับกระบวนการและอ็อบเจ็กต์ เพิ่ม **war** stanza ในไฟล์ `/etc/security/audit/config`:

```
war:
        obj_min_sl = "impl_lo a,b"
        obj_max_sl = "TS a,c"
        sub_min_sl = "impl_lo a,b"
        sub_max_sl = "TS a,c"
        obj_min_tl = impl_lo
        obj_max_tl = TS
        sub_min_tl = impl_lo
        sub_max_tl = TS
```

obj_min_sl และ **obj_max_sl** กำหนดขอบเขตการตรวจสอบ SL สำหรับ อ็อบเจ็กต์ **sub_min_sl** และ **sub_max_sl** กำหนดขอบเขตการตรวจสอบ SL สำหรับซับเจ็คต์ (กระบวนการ) **obj_min_tl** และ **obj_max_tl** กำหนดขอบเขตการตรวจสอบ TL สำหรับ อ็อบเจ็กต์ **sub_min_tl** และ **sub_max_tl** กำหนดขอบเขตการตรวจสอบ TL สำหรับซับเจ็คต์ (กระบวนการ)

war stanza พร้อมด้วยคำสั่ง **audit start** และถูกอัพโหลดไปที่เคอร์เนลก่อนระบบย่อยการตรวจสอบเริ่มต้น ถ้า **war** stanza ถูกข้าม ขอบเขตการตรวจสอบปัจจุบันในเคอร์เนลจะถูกลบออก เคอร์เนล ไม่ได้ทำการตรวจสอบขอบเขตการตรวจสอบ TL หรือ SL ถ้าไม่มีขอบเขตการตรวจสอบ TL SL ในเคอร์เนล

แฟล็กเคอร์เนล Trusted AIX:

เมื่อระบบถูกกำหนดค่าเป็นระบบ Trusted AIX เมื่อติดตั้ง โกลบอลเคอร์เนลแฟล็กถูกเปิดใช้งานในตัวแปร **_system_configuration** แมโคร **__MLS_KERNEL( )** ถูกจัดเตรียมในเคอร์เนลเพื่อกำหนดว่า ระบบถูกตั้งค่าเป็นระบบ

Trusted AIX หรือไม่ แมโครนี้ สามารถถูกเรียกโดยแอ็พพลิเคชัน user-space หรือเคอร์เนลรูทีน ค่าส่งกลับ **1** จากแมโคร **__MLS_KERNEL( )** แสดงว่า ระบบถูกกำหนดค่าเป็น Trusted AIX ค่าส่งกลับอื่น แสดงว่าระบบไม่ได้ถูกตั้งค่าเป็นระบบ Trusted AIX

การอัพเดตโปรแกรมที่มีอยู่:

โปรแกรมที่มี privilege หรือได้การไว้วางใจ โดยทั่วไปทำงานอย่างถูกต้อง บนระบบที่ไว้วางใจโดยไม่มีการเปลี่ยนแปลง

อย่างไรก็ตาม การเปลี่ยนแปลงสามารถทำได้เพื่อเพิ่มระดับการไว้วางใจ และ/หรือ ความเข้ากันได้แบบรุดหน้าของโปรแกรมเหล่านี้ คำแนะนำจำนวนมากสำหรับการสร้าง โปรแกรมใหม่ใช้ได้กับการอัพเดตโปรแกรมที่มีอยู่ ข้อแนะนำดังต่อไปนี้ มีการนำมาใช้:

- โปรแกรมที่ทดสอบเพื่อระบุว่าโปรแกรมเป็นกระบวนการที่มี privilege หรือไม่ (นั่นคือ effective user ID เป็น 0) ควรถูกแก้ไขตาม แนวทางใน Direct Privilege Checking
- โค้ดที่จัดการบิตสิทธิระบบ UNIX มาตรฐาน (บิตโหมด) ควรถูกเปลี่ยนแปลงเพื่อสะท้อนการมีอยู่ที่เป็นไปได้ ของ ACL
- โค้ดที่ใช้เพื่อรันแบบ setuid-to-root ควรถูกตรวจสอบสำหรับการใช้ privilege และควรมี privilege ที่เหมาะสมกำหนดให้

การสำรองข้อมูลและเรียกคืน:

การอิมพอร์ตและปฏิบัติการข้อมูลบนระบบ Trusted AIX ใช้เวอร์ชันที่ไว้วางใจของคำสั่ง **backup** และ **restore**

คำสั่ง **backup** และ **restore** ถูกขยาย เพื่อจัดการเลเบล ส่วนขยายเหล่านี้ผู้ใช้สามารถเห็นได้ และนอกจากส่วนขยายการเลเบล ฟังก์ชันคำสั่งเหล่านี้เทียบเท่า กับคำสั่ง AIX **backup** และ **restore** เมื่อปิดใช้งานการสำรองข้อมูลหรือการเรียกคืน ข้อมูลไม่ให้ทำส่วนขยาย แฟล็ก **-O** สามารถ ใช้ได้

ระบบ อิมพอร์ต/เอ็กพอร์ต ถูกป้องกันโดยการรวมกันของกลไก privilege และ การอนุญาต

ข้อจำกัด cron:

คำสั่ง **cron** ถูกปิดใช้งานและจะไม่รัน งานใด เมื่อระบบอยู่ในโหมด configuration ถ้าระบบอยู่ในโหมด operational คำสั่ง **cron** รันงานที่เลเบลระดับความลับ ซึ่งงานถูกส่งและเลเบล integrity ดีฟอลต์ของผู้ใช้

มีข้อจำกัดเช่น clearance ต่ำสุดและ clearance สูงสุดของผู้ใช้ขึ้นกับข้อมูลใดใหม่กว่า clearance ถูกนำมาจาก การตั้งค่าเวลาที่งานถูกส่งหรือเวลาล่าสุด ที่คำสั่ง **cron** เริ่มทำต่อ เฉพาะผู้ใช้ SA สามารถดูแล คำสั่ง **cron**

การ Mount และ unmount ระบบไฟล์:

Trusted AIX สนับสนุน การเลเบล (SL และ TL) บน JFS2 ที่มีระบบไฟล์ EAv2 SA หรือ SO สามารถ mount ระบบไฟล์ที่ไม่สนับสนุนการเลเบล (CDFS หรือ HSFS) ถ้าจำเป็น ในกรณีนี้ ไฟล์ทั้งหมดบนระบบไฟล์ที่ mount ไม่มี SL, TL หรือ FSF แยกแต่สืบทอด แอ็ตทริบิวต์ความปลอดภัยของจุด mount

การจัดการระบบ Trusted AIX

คำแนะนำสำหรับการจัดการที่ถูกต้องของระบบ Trusted AIX ต้องได้รับการปฏิบัติ ตามเพื่อประกันความปลอดภัยของระบบ

การจัดการระบบ Trusted AIX ถูกดำเนินการโดยผู้ใช้ซึ่งบัญชีผู้ใช้สัมพันธ์กับบทบาท การดูแลระบบ ผู้ใช้เหล่านี้เรียกว่า Information System Security Officer (ISSO), System Administrator (SA) และ System Officer (SO) และผู้ใช้เหล่านี้แต่ละ

คนมีการอนุญาตที่อนุญาตให้พวกเขาปฏิบัติ งานที่เป็นเซ็ตย่อยของการดูแลระบบ ผู้ใช้เหล่านี้ถูกเชื่อมโยง กับระบบที่กำหนดบทบาท isso, sa, และ so ตามลำดับ คำว่า ISSO, SA และ SO ถูกใช้เพื่ออ้างถึงที่ใช้ที่มีบทบาท isso, sa, and so ตามลำดับ หน้าที่ดูแลระบบบางหน้าที่ สามารถทำได้เฉพาะสองในสามของผู้จัดการระบบที่ทำงานร่วมกัน เนื่องจากผู้จัดการคนเดียวมีการอนุญาต เพื่อดำเนินการเหล่านี้ไม่เพียงพอ ตัวอย่างเช่น เมื่อเพิ่ม ผู้ใช้ใหม่ให้กับระบบ มีเพียง SA ที่สามารถเพิ่มบัญชีผู้ใช้ใหม่ และมีเพียง ISSO ที่สามารถสร้างรหัสผ่าน clearance และมาสก์ การตรวจสอบของผู้ใช้ การแบ่งหน้าที่นี้เรียกว่ากฎ two-man

หมายเหตุ: ประสิทธิผลของกฎ two-man ขึ้นกับการอนุญาต ที่ถูกกำหนดให้กับบทบาทการดูแลระบบ การเพิ่มการอนุญาตเพิ่มเติม ให้กับบทบาทการดูแลระบบเกินความจำเป็นสามารถทำให้ระบบมีความเปราะบาง ต่อการโจมตีจากภายใน ดูที่ RBAC สำหรับข้อมูลเพิ่มเติม เกี่ยวกับการเชื่อมโยงการอนุญาตกับบทบาท

ระบบที่กำหนดบทบาท isso, sa, และ so ถูกเชื่อมโยงกับการอนุญาต Trusted AIX ดังต่อไปนี้โดยดีฟอลต์ ควรใช้ความระมัดระวังถ้าการเชื่อมโยงเหล่านี้ ถูกเปลี่ยนแปลง เนื่องจากอาจทำให้ระบบเปราะบาง

ตารางที่ 40. การตั้งค่า บทบาทและการอนุญาต

isso	**sa**	**so**
		aix.mls.login
	aix.mls.printer	
aix.mls.network.config		
aix.mls.network.init		
aix.mls.network.config		
aix.mls.login		
aix.mls.pdir		
aix.mls.system.label		
aix.mls.tpath		
aix.mls.label		
aix.mls.system.config		
aix.mls.proc		
aix.mls.clear		
aix.mls.lef		
aix.mls.stat		
aix.mls.printer		

การจัดการระบบสำหรับ Information System Security Officers:

ระบบ Trusted AIX ถูกจัดการโดยกิจกรรมที่ทำงานร่วมกันของผู้ใช้ ISSO, SA และ SO

ระหว่างการติดตั้ง Trusted AIX สามดีฟอลต์บัญชีผู้ใช้ **isso, sa** และ **so** ถูกสร้าง (ถ้าบัญชีผู้ใช้เหล่านี้ยังไม่มีในกรณีการย้ายระบบจาก AIX ธรรมดาไปเป็น Trusted AIX) ผู้ใช้เหล่านี้ถูกเชื่อมโยงกับ isso, sa และ so ตามลำดับ

หมายเหตุ: บัญชีผู้ใช้ดีฟอลต์มีเพื่อการเซ็ตอัพและคอนฟิกูเรชันเริ่มต้น ของระบบ Trusted AIX เท่านั้น ขอแนะนำ ให้บทบาทเหล่านี้ถูกกำหนดให้กับผู้ใช้ปกติอื่น หลังจากบทบาท เหล่านี้ถูกกำหนดให้กับผู้ใช้อื่น สามารถเอาบัญชีผู้ใช้ดีฟอลต์ออกได้ ดูที่ *การติดตั้งและการย้าย* สำหรับข้อมูลเพิ่มเติม เกี่ยวกับการติดตั้ง Trusted AIX

กิจกรรม ISSO

หน้าที่หลักของ Information System Security Officer (ISSO) คือผู้ดูแลความปลอดภัยของ ระบบ เฉพาะผู้ใช้ที่ได้รับการอนุญาต ISSO ที่สามารถดำเนินกิจกรรม ISSO กิจกรรมเหล่านี้ ประกอบด้วย:

- การวางแผน การสร้าง และการบังคับใช้ นโยบายการรักษาความปลอดภัยของไซต์
- สร้างค่าดีฟอลต์ทั้งระบบสำหรับ clearance ผู้ใช้ การอนุญาต privileges การควบคุมล็อกอิน และพารามิเตอร์รหัสผ่าน
- การเซ็ตอัพโปรไฟล์การพิสูจน์ตัวตนผู้ใช้แสดงถึงระดับของการไว้วางใจ ที่กำหนดให้กับผู้ใช้ เมื่อบัญชีผู้ใช้ถูกสร้างโดยผู้ดูแลระบบ
- การกำหนด แอ็ตทริบิวต์ความปลอดภัย SL และ TL ให้กับอุปกรณ์เช่น เทอร์มินัล พรินเตอร์ ดิสก์ไดร์ฟที่ถอดได้ และไดร์ฟเทปแม่เหล็ก
- การกำหนดแท็กความปลอดภัย เลเบล privileges และชุดการอนุญาตกับ ไฟล์
- การกู้คืนระบบเป็นสถานะที่ไว้วางใจในเหตุการณ์ของระบบล้มเหลว

การจัดการระบบการตรวจสอบ:

การเข้าถึงคำสั่งการตรวจสอบถูกจำกัดกับผู้ใช้ด้วยการอนุญาต **AUDITSYS** สำหรับข้อมูลเพิ่มเติม อ้างอิงคำสั่ง **audit, auditselect,** และ **auditpr**

ตัวอย่างดังต่อไปนี้แสดง:

1. วิธีสร้างระบบไฟล์ที่จะใช้สำหรับไฟล์หลักฐานการตรวจสอบ
2. วิธีสตาร์ทระบบการตรวจสอบ
3. วิธีทำให้เร็กคอร์ดถูกสร้าง
4. วิธีวิเคราะห์คำหลักฐานการตรวจสอบเพื่อค้นหาชนิดของเร็กคอร์ดต่างๆ

รันคำสั่งดังต่อไปนี้ในฐานะผู้ใช้ที่มีการอนุญาต **FSADMIN**:

```
/usr/sbin/crfs -v jfs -g rootvg -m /audit -a size=32M -A yes
mount /audit
```

ใช้คำสั่ง **/tbin/auctlmod -e** เพื่อเพิ่มรายการดังต่อไปนี้ให้กับส่วนผู้ใช้ของไฟล์ /etc/security/audit/config:

```
username = ALL
```

แทนที่ *username* ด้วยชื่อจริงของผู้ใช้ซึ่งสามารถล็อกอินเข้าสู่ ระบบ

ในฐานะผู้ใช้ ISSO สร้างไฟล์ชื่อ /tmp/top_secret และ เปลี่ยน SL ของไฟล์เป็น **TS ALL**

```
touch /tmp/top_secret
/usr/sbin/settxattr -f sl= "TS ALL" /tmp/top_secret
```

รันคำสั่งดังต่อไปนี้ในฐานะผู้ใช้ที่มีการอนุญาต **AUDITSYS**:

```
/usr/sbin/audit start
```

ระบบการตรวจสอบขณะนี้ได้ถูกตั้งค่าและเริ่มทำงาน เพื่อบันทึกการดำเนินการ ของผู้ใช้ที่ระบุโดย *username* เมื่อผู้ใช้นี้ล็อกอินเข้าสู่ ระบบ

ล็อกอินเข้าสู่ระบบด้วยผู้ใช้ที่ระบุโดย *username* ในไฟล์ /etc/security/audit/config และรันคำสั่งดังต่อไปนี้:

```
ls -l /tmp/top_secret
exit
```

ในฐานะผู้ใช้ที่มีการอนุญาต **AUDITSYS** รันคำสั่งดังต่อไปนี้:

```
audit shutdown
$ /usr/sbin/auditselect -e "mac_fail==WILDCARD" /audit/trail | \
/usr/sbin/auditpr -v -APSV > /tmp/audit_trail-mac_failure
```

ตรวจสอบหลักฐานการตรวจสอบว่าได้ถูกเปลี่ยนทิศทางไปที่ไฟล์ /tmp/audit_trail-mac_failure และค้นหา **mac_fail** auditselect ได้ถูกแก้ไขเพื่อรับ ตัวเลือกดังต่อไปนี้:

- **subj_sl**
- **obj_sl**
- **mac_fail**
- **mac_pass**
- **mic_fail**
- **mic_pass**
- **priv_fail**
- **priv_pass**
- **auth_pass**
- **fsf_fail**
- **fsf_pass**

ตัวเลือกเหล่านี้ทั้งหมดใช้คำ **WILDCARD** เป็นค่าที่ตรงกัน

การจัดการเลเบลอ็อบเจ็กต์และกระบวนการ:

ทุกอ็อบเจ็กต์ระบบไฟล์และกระบวนการระบบมีเลเบลที่เชื่อมโยง

อ็อบเจ็กต์ระบบไฟล์ทั้งหมดที่ไม่ใช่ไฟล์ปกติมีขอบเขตของเลเบลระดับความลับ และเลเบล integrity กระบวนการมีขอบเขตของทั้งเลเบลระดับความลับ และ integrity นอกจากขอบเขต กระบวนการมี effective SL และ effective TL เลเบลนี้แสดง SL หรือ TL ปัจจุบันซึ่ง กระบวนการรันอยู่ คุณสามารถดูตารางด้วยคำสั่ง **lstxattr** คุณสามารถเซ็ตเลเบลของอ็อบเจ็กต์ระบบไฟล์และกระบวนการด้วยคำสั่ง **settxattr**

การจัดการความปลอดภัยเน็ตเวิร์ก:

AIX Trusted Network ต้องการตารางหลายตารางที่ถูกกำหนดโดย ISSO ตารางเหล่านี้ถูกเก็บใน ในไดเร็กทอรี /etc/security คำสั่ง **tninit** ถูกใช้ เพื่อสร้างไบนารีเวอร์ชันและโหลดลงในเคอร์เนล

กฎโฮสต์และเน็ตเวิร์กอินเตอร์เฟส กำหนดวิธีที่ระบบจัดการกับแพ็กเก็ตเน็ตเวิร์ก ขาเข้าและขาออก กฎโฮสต์ใช้กับโฮสต์จำเพาะ กฎเน็ตเวิร์ก อินเตอร์เฟสใช้กับอินเตอร์เฟสผ่านโฮสต์ซึ่งเชื่อมต่อกับเน็ตเวิร์ก ถ้ามีความขัดแย้งระหว่างกฎโฮสต์และกฎอินเตอร์เฟส กฎ โฮสต์มีความสำคัญกว่า

ใช้คำสั่ง **netrule** เพื่อเพิ่ม แก้ไข และเคียวรี กฎ โดยทั่วไป กฎเกี่ยวข้องกับโปรโตคอลที่ใช้ ช่วงแอดเดรส (ทั้ง โฮสต์และพอร์ต) ซึ่งใช้กับกฎ และ SL ที่กำหนดให้กับ แพ็กเก็ต ดูที่คำสั่ง **netrule** สำหรับข้อมูลเพิ่มเติม

ใช้คำสั่ง **tninit** เพื่อเตรียมข้อมูลเบื้องต้นระบบย่อย AIX Trusted Network เพื่อบันทึกกฎลงในรูปแบบไบนารี และเพื่อแสดงกฎในรูปแบบข้อความ

คุณลักษณะความปลอดภัยที่ตั้งค่าได้:

ค่าติดตั้งคุณลักษณะที่ตั้งค่าได้ ถูกแสดงระหว่างลำดับ การบูต

ค่าติดตั้งที่กำหนดค่าได้ถูกเก็บใน ODM ค่าติดตั้งเหล่านี้ สามารถถูกแสดงด้วยคำสั่ง **getsecconf** และสามารถถูกแก้ไข โดยผู้ใช้ ISSO ด้วยคำสั่ง **setsecconf**

การจัดการเลเบล:

ผู้ใช้ ISSO สามารถ เพิ่ม แก้ไข หรือลบ การเข้ารหัสเลเบลโดยการแก้ไข ไฟล์ `/etc/security/enc/LabelEncodings` ไฟล์ `/etc/security/enc/LabelEncodings` กำหนดวิธีที่ชื่อที่อ่านได้ จะถูกแม็พกับการแสดงแบบไบนารีของ เลเบลระดับความลับระบบ

หมายเหตุ: การแก้ไขไฟล์การเข้ารหัสเลเบลระดับความลับ บนระบบที่รันอยู่อาจ มีผลทำให้เลเบลไม่ถูกต้องได้ นอกจากมีความระมัดระวังอย่างสูง เนื่องจากอ็อบเจ็กต์สามารถถูกเลเบล ด้วยคำเดียวหรือเป็นการรวมกันของคำ การเปลี่ยน การเพิ่ม หรือการลบข้อบังคับการรวมคำอย่างไม่ระวัง สามารถทำให้ เลเบลไม่สามารถใช้งานได้

ไฟล์ `/etc/security/enc/LabelEncodings` ถูกแปล เป็นแบบไบนารีโดยไลบรารีรูทีน **l_init** และเก็บไว้ใน ตาราง ตารางเหล่านี้ถูกใช้เพื่อแปลง SL แถบป้ายพรินเตอร์ และ clearances ไปเป็นและจาก การเข้ารหัสไบนารีภายในของตาราง

Trusted AIX ใช้ซอฟต์แวร์ MITRE Compartmented Mode Workstation Labeling เป็นฐานสำหรับการนำการเลเบลมาใช้ เอกสาร Compartmented Mode Workstation Labeling: Encodings Format, DDS-2600-6216-93 (MTR 10649 revision 1), September 1993 อธิบายรูปแบบการเข้ารหัสเลเบล มาตรฐาน

รูปแบบการเข้ารหัสเลเบลมาตรฐานทำงานกับเลเบล integrity และเลเบล ระดับความลับ เหมือนกับที่กำหนดในส่วน **Sensitivity Labels** ของไฟล์ `/etc/security/enc/LabelEncodings`

Trusted AIX เป็นทางเลือก สนับสนุน ส่วน **Integrity Labels** ซึ่งอนุญาตให้เลเบล integrity ต่างจาก เลเบลระดับความลับได้

การจัดการไดเร็กทอรีที่พาร์ติชัน:

ต่อกระบวนการผู้ใช้ปกติ ไดเร็กทอรีที่พาร์ติชันแสดงและทำงาน เช่นเดียวกับไดเร็กทอรีธรรมดา แต่กับไดเร็กทอรีที่พาร์ติชันกระบวนการ ต่างกันที่มี SL ต่างกันเห็นเนื้อหาของไดเร็กทอรีเดียวกันต่างกัน

ตัวอย่างเช่น ถ้ากระบวนการรันที่เลเบลความปลอดภัย **SECRET** สร้าง ไฟล์ชื่อ **foo** ในไดเร็กทอรีที่พาร์ติชัน กระบวนการที่สอง รัน ที่เลเบลความปลอดภัย TOP SECRET ไม่สามารถเห็นหรือเข้าถึงไฟล์ **foo** ใน ไดเร็กทอรีนั้นได้ นอกจากนี้ กระบวนการที่สองสามารถสร้างไฟล์ **foo** ของตัวเอง โดยไม่รบกวนไฟล์ **foo** แรก

นี่ทำได้โดยการใช้ไดเร็กทอรีย่อยที่ซ่อนอยู่ สำหรับแต่ละ SL เฉพาะที่ กระบวนการเข้าถึงไดเร็กทอรีที่พาร์ติชัน มีไดเร็กทอรีย่อย ที่พาร์ติชัน เมื่อกระบวนการเข้าถึงไดเร็กทอรีที่พาร์ติชัน ระบบจะเปลี่ยนทิศทาง กระบวนการโดยอัตโนมัติไปที่ไดเร็กทอรีย่อยที่ซ่อนอยู่ ในตัวอย่างด้านบน สองไฟล์ **foo** จริงๆ แล้วอยู่ในไดเร็กทอรีย่อยต่างกัน แม้ว่าไฟล์จะแสดงต่อผู้ใช้ว่าอยู่ในไดเร็กทอรีเดียวกัน

ดูที่ “ไดเร็กทอรีที่พาร์ติชัน” ในหน้า 472 สำหรับข้อมูลเพิ่มเติมเกี่ยวกับไดเร็กทอรี ที่พาร์ติชัน

ไดเร็กทอรีที่พาร์ติชันได้รับการสนับสนุนใน JFS2 กับ EAv2

การสร้างไดเร็กทอรีที่พาร์ติชัน:

เมื่อไดเร็กทอรีที่พาร์ติชันถูกสร้าง ขอบเขต SL ดีฟอลต์คือ System Low SL ถึง System High SL เมื่อไดเร็กทอรีที่พาร์ติชัน ถูกเข้าถึง เคอร์เนลสร้างไซลด์ไดเร็กทอรีจำเพาะเลเบลโดยอัตโนมัติ (ถ้ายังไม่มีอยู่) และเปลี่ยนทิศทางกระบวนการผู้ใช้ไปที่ไซลด์ไดเร็กทอรีนี้

ใช้คำสั่ง **pdmkdir** เพื่อสร้างไดเร็กทอรี ที่พาร์ติชัน คำสั่ง **pdmkdir** ต้องการการอนุญาต **aix.mls.pdir.create** เพื่อแทนที่ข้อจำกัด DAC, MAC และ MIC ใช้คำสั่ง **pdrmdir** เพื่อลบไดเร็กทอรีที่พาร์ติชันซึ่งว่างเปล่า

ไดเร็กทอรีย่อยและ sub-subdirectories ที่พาร์ติชัน

ไดเร็กทอรี ไซลด์จำเพาะเลเบลของไดเร็กทอรีที่พาร์ติชันคือไดเร็กทอรีย่อย ที่พาร์ติชัน เมื่อกระบวนการสร้างไดเร็กทอรีไซลด์ภายใต้ไดเร็กทอรีย่อย ที่พาร์ติชัน (ด้วยคำสั่ง **mkdir**) ไดเร็กทอรี ไซลด์เป็น sub-subdirectory ที่พาร์ติชัน

เมื่อไดเร็กทอรีย่อยที่พาร์ติชัน ถูกสร้าง จะสืบทอดแอ็ตทริบิวต์ความปลอดภัยของ ไดเร็กทอรีที่พาร์ติชันพาเรนท์ ยกเว้น สำหรับ minimum SL และ maximum SL minimum และ maximum SL ถูกเซ็ตให้กับ effective SL ของ กระบวนการโหมดเสมือนที่เข้าถึงไดเร็กทอรีย่อยที่พาร์ติชันครั้งแรก

Trusted AIX จำแนกเป็นสี่ชนิดที่แตกต่างกันของไดเร็กทอรี:

- ไดเร็กทอรีปกติ (dir)
- ไดเร็กทอรีที่พาร์ติชัน (pdir)
- ไดเร็กทอรีย่อยที่พาร์ติชัน (psdir)
- sub-subdirectory ที่พาร์ติชัน (pssdir)

โหมดเสมือนและโหมดจริง:

มีสองโหมดการเข้าถึงไดเร็กทอรีที่พาร์ติชันต่างกัน: โหมดเสมือนและโหมดจริง

ในโหมดเสมือน กระบวนการเข้าถึงไดเร็กทอรีที่พาร์ติชัน สามารถเห็น เนื้อหาของไดเร็กทอรีย่อยที่พาร์ติชันจำเพาะเลเบล ไดเร็กทอรีที่พาร์ติชัน ไม่สามารถเห็นได้จากกระบวนการที่รันในโหมดเสมือน ไดเร็กทอรีพาร์ติชันเห็นได้จากกระบวนการที่รันในโหมดจริง กระบวนการ ที่รันในโหมดจริงสามารถเห็นเนื้อจริงทั้งหมดของไดเร็กทอรี และไดเร็กทอรีย่อย ที่พาร์ติชัน สำหรับกระบวนการโหมดจริง ระบบ จะไม่ดำเนินการเปลี่ยนทิศทาง

โดยดีฟอลต์ กระบวนการรันในโหมดเสมือน โหมดจริงมีไว้สำหรับ การดูระบบระบบไฟล์ ใช้คำสั่ง **pdmode** เพื่อ รันคำสั่งใน โหมดแทนเชลล์กระบวนการปัจจุบันหรือ เพื่อสลับไปที่เชลล์ในโหมดอื่น

แม้ว่ากระบวนการผู้ใช้โหมดจริงสามารถเห็นและจัดการ ไดเร็กทอรีและไดเร็กทอรีย่อย ที่พาร์ติชัน ชนิดของการเข้าถึงและการ จัดการนี้ควรถูกดำเนินการ ด้วยความระมัดระวัง ตัวอย่างเช่น ถ้าไดเร็กทอรีปกติถูกสร้างหรือย้ายไปที่ไดเร็กทอรี ที่พาร์ติชัน โดยกระบวนการโหมดจริง ไดเร็กทอรีจะเห็นไม่ได้จาก กระบวนการที่รันในโหมดเสมือน

แม้ว่าไดเร็กทอรีที่พาร์ติชันจะดูเหมือนไดเร็กทอรีปกติต่อกระบวนการโหมดเสมือน ยังมีข้อจำกัดบางประการในไดเร็กทอรีที่ พาร์ติชัน

ลำดับชั้น:

มีลำดับชั้นของไดเร็กทอรีและไดเร็กทอรีย่อยที่พาร์ติชัน

กฎดังต่อไปนี้วางระเบียบลำดับชั้นของไดเร็กทอรีที่พาร์ติชันและ ไดเร็กทอรีย่อย:

- ไดเร็กทอรีต้องเป็นหนึ่งในสี่ชนิด:
  - ไดเร็กทอรีปกติ
  - ไดเร็กทอรีที่พาร์ติชัน
  - ไดเร็กทอรีย่อยที่พาร์ติชัน
  - sub-subdirectory ที่พาร์ติชัน
- ไดเร็กทอรีต้องมีชนิดไม่เกินหนึ่งชนิด
- พาเรนท์ของไดเร็กทอรีย่อยที่พาร์ติชันต้องเป็นไดเร็กทอรีที่พาร์ติชัน
- ไดเร็กทอรีไซลด์ของไดเร็กทอรีย่อยที่พาร์ติชันต้องเป็น sub-subdirectory ที่พาร์ติชัน
- พาเรนท์ของ sub-subdirectory ที่พาร์ติชันต้องเป็นไดเร็กทอรีย่อยที่พาร์ติชัน

การละเมิดกฎเหล่านี้มีผลให้แผนผังไดเร็กทอรีที่พาร์ติชันไม่ถูกต้อง และระบบไฟล์ที่ต้องตรงกันซึ่งมีการทำงานที่กำหนดไม่ ได้

การ Mount ระบบไฟล์:

ไดเร็กทอรีหรือไดเร็กทอรีย่อยที่พาร์ติชันสามารถเป็นจุด mount แต่ ไดเร็กทอรีย่อยที่พาร์ติชันไม่สามารถเป็นจุด mount ได้ เช่นเดียวกัน root ของระบบไฟล์ที่กำลังถูก mount เป็นไดเร็กทอรีหรือไดเร็กทอรีย่อยที่พาร์ติชันได้ แต่ไม่สามารถเป็น sub-subdirectory ที่พาร์ติชัน

การสร้างและการลบไดเร็กทอรี:

เมื่อกระบวนการโหมดเสมือนที่รันอยู่ใน sub-subdirectory ที่คำสั่งพาร์ติชัน, คำสั่ง **mkdir** สร้างไดเร็กทอรีธรรมดา ถ้ากระบวน การ เดียวกันอยู่ในไดเร็กทอรีย่อยที่พาร์ติชันและเรียกใช้งานคำสั่ง **mkdir** sub-subdirectory ที่พาร์ติชันจะถูกสร้างโดย อัตโนมัติ ไดเร็กทอรีว่าง สามารถลบได้ ซับเจ็คต์กับข้อบังคับ MAC, MIC และ DAC

การย้ายไดเร็กทอรี:

ข้อบังคับ MAC, MIC และ DAC นำมาใช้เมื่อไดเร็กทอรีถูกย้าย

ไดเร็กทอรีธรรมดาสามารถถูกย้ายไปได้ทุกที่ ถ้าไดเร็กทอรีพาเรนท์ใหม่ คือไดเร็กทอรีย่อยที่พาร์ติชัน ไดเร็กทอรีธรรมดาที่ถูกย้ายจะกลายเป็น sub-subdirectory ที่พาร์ติชัน หรือไม่แล้ว จะยังคงเป็นไดเร็กทอรีธรรมดา ถ้าพาเรนท์ใหม่ของไดเร็กทอรีคือไดเร็กทอรีที่พาร์ติชันและชื่อขัดแย้งกับ ชื่อของไดเร็กทอรีย่อยที่พาร์ติชัน การเปลี่ยนทิศทางกระบวนการโหมด เสมือนภายหลังไปที่ไดเร็กทอรีย่อยที่พาร์ติชันจะล้มเหลว

ไดเร็กทอรีที่พาร์ติชันสามารถถูกย้ายไปที่ไดเร็กทอรีธรรมดาอื่นและจะยังคง เป็นไดเร็กทอรีที่พาร์ติชันหลังจากถูกย้าย ไดเร็กทอรีที่พาร์ติชัน แบบซ้อนไม่ถูกสนับสนุนใน Trusted AIX เนื่องจาก ไม่ได้ให้ประโยชน์เพิ่มขึ้น

ไดเร็กทอรีย่อยที่พาร์ติชันสามารถถูกย้ายไปที่ไดเร็กทอรีที่พาร์ติชันเท่านั้น และยังคงเป็นไดเร็กทอรีย่อยที่พาร์ติชันหลังจากการย้าย การย้ายไดเร็กทอรีย่อยที่พาร์ติชัน ไปที่ไดเร็กทอรีธรรมดา ไดเร็กทอรีย่อยที่พาร์ติชัน หรือ sub-subdirectory ที่พาร์ติชันทำไม่ได้

sub-subdirectory ที่พาร์ติชันสามารถถูกย้ายไปได้ทุกที่ ถ้าพาเรนท์ใหม่เป็น ไดเร็กทอรีธรรมดา ไดเร็กทอรีที่พาร์ติชัน หรือ sub-subdirectory ที่พาร์ติชัน จะกลายเป็นไดเร็กทอรีธรรมดา มิฉะนั้น จะยังคงเป็น sub-subdirectory ที่พาร์ติชัน

ตารางที่ 41. ข้อสรุปการย้ายไดเร็กทอรี

ชนิดการย้ายไดเร็กทอรี	ไปที่ไดเร็กทอรีธรรมดา	ไปที่ไดเร็กทอรีที่พาร์ติชัน	ไปที่ไดเร็กทอรีย่อยที่พาร์ติชัน	ไปที่ sub-subdirectory ที่พาร์ติชัน่
ธรรมดา	ทำได้ ยังคงเป็นไดเร็กทอรีธรรมดา	ทำได้[1] ยังคงเป็นไดเร็กทอรีธรรมดา	ทำได้[1] กลายเป็น sub-subdirectory ที่พาร์ติชัน	ทำได้ ยังคงเป็นไดเร็กทอรีธรรมดา
ที่พาร์ติชัน	ทำได้ ยังคงเป็นไดเร็กทอรีที่พาร์ติชัน	ทำได้[1] ยังคงเป็นไดเร็กทอรีที่พาร์ติชัน	ทำไม่ได้	ทำได้ ยังคงเป็นไดเร็กทอรีที่พาร์ติชัน
ไดเร็กทอรีย่อยที่พาร์ติชัน	ทำไม่ได้	ทำได้ ยังคงเป็นไดเร็กทอรีย่อยที่พาร์ติชัน	ทำไม่ได้	ทำไม่ได้
sub-subdirectory ที่พาร์ติชัน	ทำได้ กลายเป็นไดเร็กทอรีธรรมดา	ทำได้ กลายเป็นไดเร็กทอรีธรรมดา	ทำได้ ยังคงเป็น sub-subdirectory	ทำได้ กลายเป็นไดเร็กทอรีธรรมดา

[1] ถ้าชื่อขัดแย้งกับชื่อของ (ขณะนี้ ยังไม่มีอยู่) ไดเร็กทอรีย่อยที่พาร์ติชัน การเปลี่ยนทิศทางกระบวนการโหมด เสมือนไปที่ไดเร็กทอรีย่อยที่พาร์ติชันจะล้มเหลว

การเปลี่ยนชนิดไดเร็กทอรี:

คำสั่ง **pdset** สามารถถูกใช้เพื่อเปลี่ยนไดเร็กทอรีธรรมดา ไปเป็นชนิดไดเร็กทอรีที่พาร์ติชัน ไม่มีคำสั่งในการเปลี่ยนไดเร็กทอรีที่พาร์ติชัน ไปเป็นไดเร็กทอรีธรรมดา

การแทนที่ตัวเลข inode:

เมื่อไดเร็กทอรีย่อยที่ทำพาร์ติชันถูกเข้าถึง และหมายเลข inode ของไดเร็กทอรีย่อย หรือหมายเลข inode ของไดเร็กทอรีที่ทำพาร์ติชันที่เป็นพาเรนต์ (..) จำเป็นต้องใช้ หมายเลข inode ของไดเร็กทอรีที่ทำพาร์ติชันทีเป็นพาเรนต์ หรือหมายเลข inode ของ พาเรนต์ของไดเร็กทอรีที่ทำพาร์ติชันที่เป็นพาเรนต์ถูกส่งคืน ตามลำดับ เมื่อ sub-subdirectory ที่พาร์ติชันถูกเข้าถึงและตัวเลข inode ของพาเรนท์ ของ sub-subdirectory ที่พาร์ติชัน(..) มีความจำเป็น, ตัวเลข inode ของ ไดเร็กทอรีที่พาร์ติชัน grandparent ถูกส่งกลับ

คำสั่งไดเร็กทอรีที่พาร์ติชัน:

คำสั่งเหล่านี้ใช้กับไดเร็กทอรีที่พาร์ติชัน

pdmkdir
สร้างไดเร็กทอรีที่พาร์ติชัน

pdrmdir
ลบไดเร็กทอรีและไดเร็กทอรีย่อยที่พาร์ติชัน

pdlink ลิงก์ไฟล์ขามไดเร็กทอรีย่อยที่พาร์ติชัน

pdset เซ็ตไดเร็กทอรีกับไดเร็กทอรีที่พาร์ติชัน

pdmode
ส่งกลับโหมดการเข้าถึงไดเร็กทอรีปัจจุบัน

รันคำสั่งด้วยโหมดการเข้าถึงไดเร็กทอรีที่ระบุ

ไดเร็กทอรีปกติที่ถูกแปลงไปเป็นไดเร็กทอรีที่พาร์ติชัน สามารถถูกแปลงกับไปเป็นไดเร็กทอรีปกติ

การตรวจทานความปลอดภัยระบบ:

เป็นหน้าที่ของ ISSO ในการตรวจทานสถานะความปลอดภัย ของระบบ การตรวจทานความปลอดภัยระบบจำเป็นต้องถูกดำเนินการทันทีหลังจาก การติดตั้ง และในเวลาที่ system integrity อาจถูกสร้าง ช่องโหว่ และการตรวจทานความปลอดภัยระบบควรดำเนินการเป็นระยะๆ

ไดเร็กทอรีฐานข้อมูล system integrity ซึ่งถูกเก็บในไฟล์ `/etc/security/tsd/tsd.dat` มีข้อมูลเกี่ยวกับความปลอดภัยของอ็อบเจ็กต์ filesystem เช่นคำสั่งสำคัญ และอุปกรณ์ระบบ ฐานข้อมูลนี้ต้องถูกอัพเดตเมื่ออุปกรณ์ใหม่ถูกเพิ่ม หรือข้อมูลความปลอดภัยของไฟล์ถูกแก้ไข ดูที่คำสั่ง **trustchk** สำหรับข้อมูลเพิ่มเติม

คำสั่ง **trustchk** เปรียบเทียบการตั้งค่าความปลอดภัยปัจจุบัน ของ ไฟล์ ไดเร็กทอรีหรืออุปกรณ์ กับรายการที่ตรงกันในฐานข้อมูล system integrity และซ่อมแซมแอ็ตทริบิวต์ความปลอดภัยที่ไม่ตรงกัน คำสั่ง **trustchk** สามารถ รันโดยผู้ใช้ ISSO-authorized เท่านั้น

การจัดการ TTY:

SL ต่ำสุด SL สูงสุด และ TL สำหรับอุปกรณ์ tty ถูกกำหนดใน ฐานข้อมูล tty ในไฟล์ `/etc/login.cfg` อ้างอิง ถึงคำสั่ง **chsec** สำหรับข้อมูลเพิ่มเติม

SL ที่มีผลของการล็อกอินของผู้ใช้ผ่านพอร์ต TTY ควรอยู่ภายใน ช่วงที่กำหนดสำหรับพอร์ตนี้ในไฟล์นี้ ถ้า TL อื่นที่ไม่ใช่ NOTL ถูกระบุ สำหรับพอร์ต TTY ดังนั้น TL ที่มีผลของผู้ใช้ต้องเหมือนกับ TL ที่ระบุ

การจัดการ clearances ผู้ใช้:

ผู้ใช้แต่ละคนรวมถึงผู้ใช้ ISSO, SA และ SO ต้องมีเลเบล เพื่อล็อกอินเข้าสู่ระบบ clearance ผู้ใช้สามารถถูกระบในไฟล์ `/etc/security/user` เป็นส่วนหนึ่งของ stanza ของผู้ใช้ แอ็ตทริบิวต์ **minsl, maxsl, defsl, mintl, maxtl,** และ **deftl** ระบุ minimum SL, maximum SL, default SL, minimum TL, maximum TL, และ default TL ตามลำดับสำหรับผู้ใช้ ถ้าแอ็ตทริบิวต์เหล่านี้ถูกระบุใน stanza ของผู้ใช้ ค่าที่ระบุใน stanza ดีฟอลต์ของไฟล์ถูกกำหนดให้กับผู้ใช้

เฉพาะผู้ใช้ ISSO ที่สามารถแก้ไขฐานข้อมูล clearance การรักษาความปลอดภัย clearance ของผู้ใช้ สามารถถูกแสดงด้วยคำสั่ง **lsuser** and **lssec** และสามารถถูกแก้ไขโดยใช้คำสั่ง **chuser** and **chsec**

ค่า default SL ต้องถูกควบคุมโดยค่า maximum SL และต้องควบคุม minimum SL เช่นเดียวกัน ค่า default TL ต้องถูกควบคุมโดย ค่า maximum TL และต้องควบคุม minimum TL

หมายเหตุ: สำหรับผู้ใช้ เพื่อให้ล็อกอินเข้าสู่ระบบสำเร็จ ความสัมพันธ์ทางด้านบนต้องเป็นจริง

การจัดการระบบสำหรับผู้ดูแลระบบ:

ผู้ใช้ SA มีหน้าที่หลักในแง่มุมเกี่ยวกับการดูแลระบบ ที่ไม่เกี่ยวข้องกับการรักษาความปลอดภัย

ความรับผิดชอบของผู้ใช้ SA มีดังนี้:

- เพิ่ม ลบ และดูแล บัญชีผู้ใช้
- แบ่งงานกับผู้ใช้ ISSO ในการประกับ integrity ภายใน ของซอฟต์แวร์ระบบและระบบไฟล์
- สร้างและดูแลระบบไฟล์ รวมถึงการวางแผนโครงร่างดิสก์ การทำพาร์ติชันดิสก์ และการเปลี่ยนขนาดพาร์ติชันดิสก์ การจัดสรรพื้นที่สวอป และพื้นที่สำหรับไดเร็กทอรี ระบบและผู้ใช้ การมอนิเตอร์การใช้ระบบไฟล์ การตรวจจับ และการจัดการบล็อกดิสก์ที่ bad และจัดการพื้นที่ระบบไฟล์โดยการย้าย ลบ เก็บถาวร หรือบีบอัดไฟล์และระบบไฟล์
- ระบุและรายงานปัญหาของระบบโดยการวิเคราะห์ข้อมูลข้อผิดพลาด และ ทดสอบคอมโพเนนต์ระบบ เช่น ระบบไฟล์ หน่วยความจำระบบ และอุปกรณ์

การจัดการบัญชีผู้ใช้:

ผู้ใช้ SA มีหน้าที่เพิ่มผู้ใช้ใหม่ให้กับระบบ ผู้ใช้ ISSO มีหน้าที่เปิดทางให้ผู้ใช้ใหม่ล็อกอินและเรียกใช้ คำสั่งบนระบบ

ดูที่ การจัดการ ระบบสำหรับ Information System Security Officers สำหรับ ข้อมูลเกี่ยวกับการเพิ่มการอนุญาตให้กับบัญชีผู้ใช้

เมื่อผู้ใช้ SA ถูกเพิ่ม ผู้ใช้ได้ถูกเพิ่มให้กับระบบ ผู้ใช้ ISSO ต้องได้รับการแจ้งเตือน เพื่อให้สามารถตั้งค่ารหัสผ่านเริ่มต้นเพื่อเปิดให้ ผู้ใช้ใหม่เข้าถึงระบบ

เมื่อมีการกำหนดว่าผู้ใช้ไม่ควรมีสิทธิเข้าถึงระบบอีกต่อไป ผู้ใช้ควรถูกเอาออกทันที การเอาผู้ใช้ออกสามารถทำได้โดย ผู้ใช้ SA เท่านั้น ID ผู้ใช้ของผู้ใช้ที่เอาออกจากระบบควรถูกนำกลับมาใช้ นอกจากได้ถูกมอบคืนให้กับผู้ใช้ดั้งเดิม และเฉพาะเมื่อแต่งตั้งผู้ใช้นี้ กลับสู่ระบบอีกครั้ง

ดูที่คำสั่ง **mkuser, rmuser, chuser,** และ **pwadm** สำหรับข้อมูลในการสร้างและ แก้ไขบัญชีผู้ใช้

การจัดการพรินเตอร์:

เมื่อพรินเตอร์ถูกติดตั้งอย่างถูกต้อง พรินเตอร์ถูกเพิ่ม ให้กับระบบโดยการดำเนินการที่รวมกันของผู้ใช้ SA และ SO ผู้ใช้ SO เพิ่มพรินเตอร์ให้กับระบบและผู้ใช้ SA สร้างขอบเขต SL ของพรินเตอร์ ผู้ใช้ ISSO มีสิทธิในการดำเนินงาน ทั้งสองนี้

ขอบเขต SL ของพรินเตอร์ต้องไม่ถูกสร้างจนกว่าพรินเตอร์ได้ ถูกเพิ่มให้กับระบบ ใช้คำสั่ง **smit** เพื่อจัดกัดพรินเตอร์

หมายเหตุ: การพิมพ์ที่เลเบลของไฟล์ PostScript และ ASCII สนับสนุนเฉพาะบนพรินเตอร์ PostScript

การเข้าถึง MAC กับพรินเตอร์ถูกกำหนดโดย SL ของกระบวนการ ที่กำลังพิมพ์ไฟล์ SL นี้แสดงบนแถบป้าย ส่วนหัว/ส่วนท้าย และหน้าเทรลเลอร์ กระบวนการใช้คำสั่ง **lp** ต้องมีการเข้าถึง MAC, MIC และ DAC กับไฟล์ที่กำลังถูกพิมพ์ มิฉะนั้นคำสั่ง **lp** ไม่สร้างการร้องขอ การพิมพ์

เมื่อพรินเตอร์ถูกเอาออกจากระบบ โปรไฟล์พรินเตอร์ควรถูก ลบออกจากระบบทันที ซึ่งสามารถทำได้เฉพาะโดย ผู้ใช้ที่มีการอนุญาต SO

การจัดการ filesystems:

ระบบไฟล์ประกอบด้วย ไดเร็กทอรี ไฟล์ข้อมูล ไฟล์เรียกทำงาน และไฟล์พิเศษ ระบบไฟล์สามารถอยู่บนอุปกรณ์สื่อบันทึกความจุสูง เช่นฮาร์ดดิสก์และฟล็อปปีดิสเก็ต

แม้ว่ามีเพียงผู้ใช้ SA ที่สามารถสร้างและดูแลระบบไฟล์ ทั้งผู้ใช้ SA และ SO สามารถ mount และ unmount ระบบไฟล์

การตรวจสอบระบบไฟล์ด้วยคำสั่ง fsck:

integrity ภายในของระบบไฟล์ควรถูกตรวจสอบเป็นระยะ ด้วยคำสั่ง **fsck** คำสั่ง **fsck** ต้องถูกรันบนระบบไฟล์ที่ unmount. คำสั่ง **fsck** สามารถถูกเรียกใช้งานได้โดยผู้ใช้ SA เท่านั้น

โดยดีฟอลต์คำสั่ง **fsck** รันแบบโต้ตอบ พร้อมต์ ผู้ใช้สำหรับการดำเนินการที่จะกระทำเมื่อพบไฟล์หรือไดเร็กทอรี orphaned ผู้ใช้มีตัวเลือกที่จะลบไฟล์หรือพยายามกู้คืนไฟล์ ถ้าผู้ใช้ ระบุว่าไฟล์ควรถูกกู้คืน คำสั่ง **fsck** พยายามเก็บไฟล์ในไดเร็กทอรี `/lost+found`

หลังจากคำสั่ง **fsck** สมบูรณ์และไฟล์ที่กู้คืน ถูกเก็บในไดเร็กทอรี `/lost+found` ผู้ใช้ ISSO ควรตรวจไฟล์เพื่อกำหนดระดับความปลอดภัย ขอแนะนำ ให้ไดเร็กทอรี `/lost+found` ถูกกำหนดให้กับ **SYSTEM_HIGH** SL เพื่อป้องกันผู้ใช้ปกติไม่ให้เข้าถึงไฟล์ที่กู้คืน

ดูที่คำสั่ง **fsck** สำหรับข้อมูลเพิ่มเติม

การจัดการระบบสำหรับ System Officers:

ผู้ใช้ SO มีหน้าที่หลักในแง่มุมเกี่ยวกับความปลอดภัย ของการดูแลระบบ

การจัดการ filesystems:

System Officers มีหน้าที่ในการจัดการ filesystem

ระบบไฟล์ที่สนับสนุน:

Trusted AIX สนับสนุน ระบบไฟล์ disk-based ทั้งหมด

ระบบไฟล์ทั้งหมดยกเว้น JFS2 ได้รับการสนับสนุนบน Trusted AIX เป็น ระบบไฟล์ single-level ระบบไฟล์เหล่านี้สามารถถูกประกอบเข้ากับระบบ Trusted AIX จะได้รับเลเบลและแอ็ตทริบิวต์ความปลอดภัยอื่นโดยอัตโนมัติ และจะ ถูกซับเจ็คต์กับกลไกความปลอดภัยที่บังคับใช้โดย Trusted AIX ไฟล์อ็อบเจ็กต์ทั้งหมดในระบบไฟล์ single-level มีแอ็ตทริบิวต์ความปลอดภัยเหมือนกัน แอ็ตทริบิวต์ความปลอดภัยเหล่านี้ถูกสืบทอดมาจากจุดประกอบเข้า

JFS2 ถูกสร้างบน Trusted AIX เป็น ระบบไฟล์หลายระดับ แต่ละไฟล์อ็อบเจ็กต์ในระบบไฟล์หลายระดับ มีแอ็ตทริบิวต์ความปลอดภัยเป็นของตัวเอง (เลเบลความปลอดภัย) ตัวอย่างเช่น ไดเร็กทอรี JFS2 มี minimum และ maximum SL อิสระ

ในระบบไฟล์ single-level, minimum และ maximum SL ของจุดประกอบเข้า เหมือนกัน และไดเร็กทอรีและไฟล์ทั้งหมดภายใต้จุดประกอบเข้าต้องเท่ากับ SL เหล่านี้เช่นกัน

การ Mount และ unmount filesystems:

ผู้ใช้ SO (ที่ได้รับการอนุญาต **aix.fs.manage.mount**) ได้รับอนุญาตให้ mount หรือ unmount filesystem คำสั่ง **mount** ใช้ชื่อไฟล์พิเศษของอุปกรณ์ และไดเร็กทอรีที่ mount เป็นตัวเลือก

เมื่อ multilevel JFS2 filesystems ถูกเชื่อมต่อ ไดเร็กทอรีการเชื่อมต่อถูกกำหนด เลเบลของ root ของระบบไฟล์ บน multilevel filesystem แต่ละ ไฟล์มีเลเบล sensitivity และ integrity ของตัวเอง ถ้าไฟล์ ถูกแก้ไข เลเบลของไฟล์จะถูกอัพเดต

การจัดการพรินเตอร์:

ผู้ใช้ SO สามารถใช้คำสั่ง **lpadmin** เพื่อเพิ่มและ เอาพรินเตอร์ออก แก้ไขพรินเตอร์ และใช้การควบคุมประเภทอื่นกับ ระบบย่อยพรินเตอร์ ผู้ใช้ SA สามารถใช้คำสั่ง **lpadmin** เพื่อเพิ่มหรือแก้ไข Sensitivity Labels (SL) สำหรับพรินเตอร์และ สามารถคำสั่ง enable และ disable เพื่อเปิดหรือปิดใช้งานพรินเตอร์

ระบบย่อยพรินเตอร์:

ระบบย่อยพรินเตอร์ดำเนินงานหลายประเภทที่เกี่ยวกับการดำเนินการพรินเตอร์

งานระบบย่อยพรินเตอร์มีดังต่อไปนี้:

- ดูแลพรินเตอร์และแอ็ตทริบิวต์
- รับ เก็บ และจัดตาราง งานพิมพ์ของผู้ใช้
- จัดตารางงานพิมพ์สำหรับหลายพรินเตอร์
- เริ่มโปรแกรมที่ติดต่อกับพรินเตอร์
- ติดตามสถานะของพรินเตอร์และงานพิมพ์
- รายงานเมื่อมีปัญหาเกิดขึ้น
- จำกัดงานพิมพ์ของผู้ใช้ให้อยู่ภายในขอบเขต SL ของ พรินเตอร์
- จำกัดการเข้าถึงแก่งานพิมพ์ของผู้ใช้เมื่อมีการส่งงาน
- จำกัดการเข้าถึง ไฟล์และไดเร็กทอรีสนับสนุนพรินเตอร์
- เลเบลพรินเตอร์เอาต์พุตอย่างถูกต้อง

คุณลักษณะความปลอดภัยพรินเตอร์:

ระบบย่อยพรินเตอร์ถูกแก้ไขใน Trusted AIX เพื่อรวม คุณลักษณะความปลอดภัย

ระบบย่อยพรินเตอร์คือระบบย่อยที่ป้องกันโดย ที่เป็นของ system ID **lp** นี่เป็นการป้องกันผู้ใช้ปกติจากการเข้าถึงไฟล์และไดเร็กทอรีสนับสนุน พรินเตอร์ นอกเหนือจากผู้ใช้ที่เป็นเจ้าของงานพิมพ์ที่ส่ง และไฟล์พิเศษของอุปกรณ์การพิมพ์

ระบบย่อยพรินเตอร์ตรวจสอบงานพิมพ์ที่ส่งของผู้ใช้ที่อยู่ ภายในขอบเขต SL ของพรินเตอร์ การตรวจสอบนี้กระทำเมื่อ ผู้ใช้ส่งงานพิมพ์ด้วยคำสั่ง **lp** และก่อนที่งานที่ส่งถูกพิมพ์โดย **lpsched** daemon ผู้ดูแลระบบควรทำการตรวจสอบความปลอดภัยระบบย่อยพรินเตอร์ ในกรณีที่งานพิมพ์ของผู้ใช้ถูกปฏิเสธ

แถบป้ายเพจถูกพิมพ์สำหรับงานพิมพ์ทั้งหมด แถบป้ายเพจรวม human-readable SL ของงานพิมพ์ แถบป้ายเพจแสดงที่ด้านหน้า และหลังของงานพิมพ์ทั้งหมด ผู้ใช้สามารถพิมพ์โดยไม่มีแถบป้าย แต่ นี่เป็นการกระทำที่ตรวจสอบได้ คุณควรตรวจสอบเสมอว่าเลเบลส่วนหัว และส่วนท้ายบนแต่ละหน้าถูกต้องและถูกควบคุมโดย เลเบลบนแถบป้ายเพจ

หมายเหตุ: ผู้ดูแลระบบพรินเตอร์รายบรรทัดต้องสร้างขอบเขตเลเบล สำหรับแต่ละพรินเตอร์ เมื่อต้องการกำหนดเลเบลเดี่ยวให้กับพรินเตอร์ให้รันคำสั่ง ดังต่อไปนี้:

lpadmin -d *printer_name* **-J***label* **-L***label* นี่ เป็นการประกันว่าเฉพาะข้อมูลที่มี *เลเบล* ที่กำหนด สามารถถูกพิมพ์บนพรินเตอร์

สรุปคำสั่งพรินเตอร์:

บางคำสั่งระบบย่อยพรินเตอร์สามารถถูกรันได้โดยผู้ใช้ทั้งหมด อย่างไรก็ตาม, บางคำสั่งระบบย่อยพรินเตอร์สามารถรันได้โดยผู้ใช้ SO, SA หรือ ISSO เท่านั้น

ตารางดังต่อไปนี้แสดงคำสั่งระบบย่อยของพรินเตอร์ที่สามารถรันได้โดยผู้ใช้ ทั้งหมด:

lp ส่งไฟล์ไปที่พรินเตอร์

lpstat ให้รายงานสถานะของระบบย่อยพรินเตอร์

คำสั่งการดูแลระบบย่อยพรินเตอร์ต้องการการอนุญาต SO ยกเว้น ผู้ใช้นั้นมีการอนุญาต SA หรือ ISSO สามารถรันคำสั่ง **lpadmin** เพื่อระบุขอบเขตเลเบลของพรินเตอร์และรันคำสั่ง **lpstat** เพื่อแสดงพรินเตอร์และการร้องของงาน SL ตารางดังต่อไปนี้แสดงคำสั่ง การดูแลระบบย่อยพรินเตอร์:

accept อนุญาตงานบนพรินเตอร์

cancel ยกเลิกการร้องขอพิมพ์ไฟล์

disable หยุดการแอคทีฟพรินเตอร์

enable เรียกทำงานพรินเตอร์

lpadmin
ตั้งค่าหรือเปลี่ยนพรินเตอร์คอนฟิกูเรชัน

lpfilter ตั้งค่าหรือเปลี่ยนตัวกรองพรินเตอร์

lpforms
ตั้งค่าหรือเปลี่ยนพรินเตอร์ฟอร์ม

lpmove ย้ายการร้องขอการพิมพ์

lpsched พิมพ์การร้องขอ

lpshut หยุดเซอร์วิสการพิมพ์

lpusers ตั้งค่าหรือเปลี่ยนระดับความสำคัญการพิมพ์

reject ป้องกันงานบนพรินเตอร์

การจัดการพรินเตอร์บรรทัดคำสั่ง:

คุณสามารถใช้คำสั่ง **accept, enable, disable, lpstat,** และ **lp** เพื่อจัดการพรินเตอร์จากบรรทัดคำสั่ง

คุณสามารถใช้คำสั่ง **accept** เพื่ออนุญาตให้งานถูกส่งไปที่ พรินเตอร์ รันคำสั่งดังต่อไปนี้เพื่ออนุญาตให้พรินเตอร์ *เลเซอร์* รับงานพิมพ์:

```
/usr/sbin/accept laser
```

พรินเตอร์ที่ระบุ *laser* ขณะนี้สามารถรับการร้องของงานพิมพ์ อย่างไรก็ตามงานพิมพ์จะไม่ถูกพิมพ์จนกว่าพรินเตอร์ถูกเปิดใช้งาน รันคำสั่ง enable เพื่อเปิดใช้งานพรินเตอร์:

```
/usr/bin/enable laser
```

คำสั่ง **enable** และ **disable** เป็นคำสั่ง การดูแลระบบและสามารถถูกรันได้เฉพาะโดยผู้ใช้ที่มีการอนุญาต ISSO หรือ SA

เพื่อยืนยันว่าพรินเตอร์ได้ถูกตั้งค่าอย่างถูกต้อง รันคำสั่ง **lpstat** ดังต่อไปนี้:

```
lpstat -p laser -l
```

คำสั่ง นี้แสดงรายงานสถานะแบบยาวของพรินเตอร์ *เลเซอร์* ถ้าคุณรัน คำสั่ง **lpstat** โดยไม่มีตัวเลือก **–l** รายงาน สถานะที่สั้นกว่าจะถูกแสดง ถ้าผู้ใช้ได้รับอนุญาต SA หรือ ISSO และมีการใช้ ตัวเลือก **–l,** ขอบเขต SL ของพรินเตอร์ถูกรายงานด้วยเช่นกัน

เมื่อต้องการระบุสถานะของการร้องขอการพิมพ์ ให้รันคำสั่ง **lpstat** ดังต่อไปนี้:

```
lpstat -o
```

คำสั่งนี้ แสดงการร้องขอการพิมพ์ **lp** ทั้งหมด ถ้าผู้ใช้ได้รับอนุญาต SA หรือ ISSO, SL และ clearance ที่มีผลของแต่ละการร้องขอจะถูกรายงาน

เมื่อต้องการพิมพ์ชื่อไฟล์ รันคำสั่ง **lp** ดังต่อไปนี้:

```
lp -d laser filename
```

มิฉะนั้น คุณต้องระบปลายทางงานการพิมพ์ เมื่อคุณรันคำสั่ง **lp**

ถ้าพรินเตอร์ปลายทางดีฟอลต์ได้ถูกเซ็ตโดยผู้ดูแลระบบ ตัวเลือก **–d** *destination_ptr* ไม่จำเป็น ตัวอย่างเช่น เพื่อพิมพ์ชื่อไฟล์ของไฟล์บนเครื่องพิมพ์เลเซอร์ ให้ป้อนคำสั่ง **lp** ดังต่อไปนี้:

```
lp filename
```

การจัดการการปิดระบบ:

ผู้ใช้ SO สามารถปิดระบบโดยการรีบูตระบบ หรือการหยุดระบบโดยสมบูรณ์

คำสั่งดังต่อไปนี้สามารถถูกรันโดยผู้ใช้ SO เพื่อรีบูตหรือหยุดระบบ หรือเปลี่ยนภาวะ init ของระบบ:

reboot รีบูตระบบโดยอัตโนมัติ

halt หยุดการดำเนินการของระบบทั้งหมด

shutdown
หยุดการดำเนินการของระบบทั้งหมด

init เปลี่ยนภาวะ init ของระบบ

การสำรองและเรียกคืนไฟล์:

การสำรองข้อมูลช่วยป้องกันการสูญเสียข้อมูลในเหตุการณ์ความขัดข้องของฮาร์ดแวร์ หรือการลบไฟล์โดยไม่ตั้งใจ การสำรองข้อมูลควรทำเป็นประจำ พร้อมกับการสำรองเพิ่มค่าที่ทำระหว่างทำสำเนาสำรองให้สมบูรณ์

คำสั่ง **backup** และ **restore** มี ตัวเลือกเพื่อระบุ ชื่อการสำรองไฟล์ ตำแหน่ง ชนิด และตัวเลือก อื่น คุณสามารถใช้คำสั่ง **mksysb** เพื่อสร้างอิมเมจที่ติดตั้งได้ Trusted AIX ของกลุ่มวอลุ่ม root ในไฟล์หรือบนเทปที่บูตได้ คุณสามารถรันคำสั่งเหล่านี้โดยใช้คำสั่ง **smit** การสำรองระบบไฟล์ควรถูกเลเบล อย่างถูกต้องและเก็บไว้ในที่ปลอดภัย

โปรแกรมมิง Trusted AIX

การรักษาความปลอดภัยระบบขึ้นอยู่กับซอฟต์แวร์ trusted computing base (TCB) ฮาร์ดแวร์ และเฟิร์มแวร์ ซึ่งรวมถึง เคอร์เนลระบบปฏิบัติการทั้งหมด ไดรเวอร์อุปกรณ์ทั้งหมดและโมดูล System V STREAMS ส่วนขยาย เคอร์เนล และโปรแกรมที่ไว้วางใจทั้งหมด ไฟล์ทั้งหมดที่ใช้โดยโปรแกรมเหล่านี้ ในการสร้างการตัดสินใจด้านความปลอดภัย ถือว่าเป็นส่วนหนึ่งของ TCB

การสร้างซอฟต์แวร์ที่ไว้วางใจ ต้องการความเข้าใจที่ชัดเจน ในเรื่องหลักการและคุณลักษณะการรักษาความปลอดภัยระบบพื้นฐาน ข้อบกพร่องด้าน ความปลอดภัยเกือบทั้งหมดในระบบ UNIX-based เนื่องจากซอฟต์แวร์ที่ไว้วางใจที่เขียนขึ้นมาไม่ดี อย่างไรก็ตามด้วยการตรวจสอบความปลอดภัยเคอร์เนล Trusted AIX คุณสามารถเขียนแอ็พพลิเคชันที่ใช้คุณลักษณะความปลอดภัยที่เพิ่มประสิทธิภาพ แอ็พพลิเคชันที่เขียนสำหรับ Trusted AIX อาจตอบสนองต่อ ไฟล์และกระบวนการที่ระดับต่างกัน และสามารถมีการทำงาน ต่างกันขึ้นกับระดับของกระบวนการหรือไฟล์ที่แอ็พพลิเคชัน ใช้อยู่ แอ็พพลิเคชันดังกล่าวเรียกว่าแอ็พพลิเคชัน multilevel-aware (MLS)

โปรแกรมเมอร์ระบบที่ไว้วางใจต้องมีประสบการณ์เป็นอย่างดีในคุณลักษณะความปลอดภัยของ Trusted AIX และต้องเข้าใจการเรียกระบบ Trusted AIX ใหม่ทั้งหมด และ คำสั่ง ที่เกี่ยวข้องกับความปลอดภัยและไลบรารี ข้อมูลนี้มีเพื่อโปรแกรมเมอร์ที่สร้างหรือแก้ไขซอฟต์แวร์ที่ไว้วางใจ มี คำแนะนำ หลักการ และข้อควรระวังสำหรับการแก้ไขและการสร้าง ซอฟต์แวร์ที่ไว้วางใจ ขณะนี้ข้อมูลแนะนำการอธิบาย หลักการและเมธอดด้านความปลอดภัย บางส่วน ขอแนะนำให้โปรแกรมเมอร์ระบบที่ไว้วางใจ อ่านข้อมูลอื่นเกี่ยวกับระบบการรักษาความปลอดภัย

หลักการของซอฟต์แวร์ที่ไว้วางใจ

มีหลักการสำคัญที่เกี่ยวข้องในการสร้างและแก้ไขซอฟต์แวร์ ที่ไว้วางใจ รวมถึงการไว้วางใจและ privileges การออกแบบซอฟต์แวร์ที่ไว้วางใจ privilege ขั้นต่ำ ระเบียบโปรแกรมมิง และการการปกป้อง TCB

ความไว้วางใจและ privilege:

กระบวนการสามารถหลีกเลี่ยงข้อจำกัดการรักษาความปลอดภัยระดับต้น (MAC, MIC, DAC, และการดำเนินการที่จำกัดอื่นๆ) เฉพาะถ้ากระบวนการมี privilege เหมาะสม กระบวนการที่รันอยู่โดยมี privilege เรียกว่ากระบวนการ privileged และโปรแกรมที่กระบวนการกำลังรันอยู่เรียกว่าโปรแกรม privileged (trusted)

privilege หมายถึงแอ็ตทริบิวต์ที่อนุญาตให้กระบวนการ ดำเนินการที่เกี่ยวข้องกับความปลอดภัย Trusted AIX ระบุ และจัดกลุ่มการดำเนินการด้านความปลอดภัย และเชื่อมโยง privilege กับแต่ละการดำเนินการ ซึ่งเป็นการเอา superuser (หรือ root) privilege ออกจาระบบฐานอย่างมีประสิทธิภาพ Privileges สัมพันธ์กับกระบวนการและ ไฟล์เรียกทำงาน

โปรแกรมต้องได้รับการไว้วางใจภายใต้สภาวะดังต่อไปนี้:

- โปรแกรมถูกตั้งค่าหรือกำหนดให้รันเป็นกระบวนการ privileged ซึ่งใช้กับโปรแกรมที่ถูกกำหนดให้รับโดยกระบวนการ privileged
- โปรแกรมขึ้นกับโปรแกรมที่ไว้วางใจอื่นในการสร้าง การตัดสินใจด้านความปลอดภัย ตัวอย่างเช่น โปรแกรมที่เปลี่ยนฐานข้อมูลสำคัญต้องได้รับการไว้วางใจ ถ้าโปรแกรมอื่นขึ้นกับข้อมูลในฐานข้อมูลในการทำการตัดสินใจด้าน ความปลอดภัย

เป็นสิ่งสำคัญในการประกันว่าโปรแกรมที่ไม่ไว้วางใจจะไม่สามารถรันเป็นกระบวนการ privileged มีหลายวิธีในการป้องกันโปรแกรมที่ไม่ไว้วางใจจากการรัน เป็นกระบวนการ privileged:

- โดยปกติอย่าอนุญาตให้กระบวนการ privileged เรียกใช้งานโปรแกรมที่ไม่ไว้วางใจ ตัวอย่างเช่น เตือนผู้ใช้ที่รันโปรแกรม privileged shell-like ไม่ให้รัน โปรแกรมที่ไม่ไว้วางใจในโปรแกรม privileged shell-like
- อย่าอนุญาต privileges เริ่มต้น ที่สืบทอด หรือได้รับอนุญาต กับไฟล์เรียกทำงาน ที่ไม่ไว้วางใจ

ส่วนของเคอร์เนลระบบปฏิบัติการทั้งหมด รวมถึง ไดรเวอร์อุปกรณ์โมดูล STREAMS และส่วนขยายเคอร์เนล ต้องได้รับการไว้วางใจ อ็อบเจ็กต์ข้อมูลเช่น ไฟล์และอุปกรณ์ฟิสิคัล ถือว่าได้รับการไว้วางใจถ้ามีข้อมูล ที่ขึ้นกับโปรแกรมที่ไว้วางใจเพื่อสร้างการตัดสินใจด้านความปลอดภัย

การออกแบบซอฟต์แวร์ Trusted:

กระบวนการของการสร้างซอฟต์แวร์ที่ไว้วางใจเหมือนกับ คอมโพเนนต์ซอฟต์แวร์ที่สำคัญ การสร้างซอฟต์แวร์ที่ไว้วางใจควรเป็นไปตาม ความเข้าใจเป็นอย่างดีและเอกสารข้อกำหนด การออกแบบ การนำไปปฏิบัติ การทดสอบ และวงจรการควบคุมคอนฟิกูเรชัน

แง่มุมที่สำคัญที่สุดของการออกแบบซอฟต์แวร์ที่ไว้วางใจคือ identification ของประเด็นและอ็อบเจ็กต์ และนิยามของการดำเนินการการรักษาความปลอดภัย ที่แม่นยำที่ระดับของ abstraction ที่ถูกต้อง นโยบายการรักษาความปลอดภัยส่วนใหญ่จำกัดอยู่กับ ซับเจ็คต์ อ็อบเจ็กต์ และการดำเนินการ เมื่อซับเจ็คต์ร้องขอสิทธิในการ อ่าน เปลี่ยนแปลง หรือสร้างอ็อบเจ็กต์ นโยบายการรักษาความปลอดภัยมอนิเตอร์การร้องขอเหล่านั้น และรับรองหรือปฏิเสธการร้องขอ

ซับเจ็คต์

โดยปกติซับเจ็คต์แสดงโดย ID ผู้ใช้ และ ID กลุ่ม โดยปกติ ผู้ใช้ และ/หรือ ID กลุ่ม ที่มีผลของกระบวนการถูกใช้สำหรับจุดประสงค์นี้ แม้ว่าอาจเหมาะสมในบางกรณี เพื่อใช้ ผู้ใช้ และ/หรือ ID กลุ่มจริง

อ็อบเจ็กต์

อ็อบเจ็กต์เป็นคอลเล็กชันของข้อมูลซึ่งการเข้าถึง ควรถูกควบคุม ในกรณีส่วนใหญ่ อ็อบเจ็กต์จะเป็นไฟล์ ถึงแม้ว่า เป็นเรื่องปกติสำหรับโปรแกรมที่ไว้วางใจในการควบคุมการเข้าถึงอ็อบเจ็กต์ที่ต่างกัน ทางโลจิคัลภายในไฟล์เดียวกัน เป็นการดีกว่าที่จะแม็พอ็อบเจ็กต์แบบหนึ่งต่อหนึ่ง ไปยังไฟล์

ในบางกรณี ซับเจ็คต์สามารถถูกพิจารณาเป็นอ็อบเจ็กต์ ตัวอย่างเช่น กระบวนการโดยปกติจะถือว่าเป็นซับเจ็คต์ อย่างไรก็ตาม กระบวนการหนึ่ง พยายามมีผลต่อกระบวนการที่สอง กระบวนการที่สองโดยปกติจะถูกพิจารณา เป็นอ็อบเจ็กต์ตามการดำเนินการนี้

การร้องขอ

การร้องขอคือชุดของการดำเนินการที่โมดูลที่ไว้วางใจ กระทำในลักษณะของซับเจ็คต์ แต่ละการร้องขอต้องถูกระบุอย่างชัดเจน ในรูปของ อินพุต เอาต์พุตที่เป็นไปได้ และผลลัพธ์ รวมทั้ง ผลข้างเคียง ของการร้องขอ identification ที่แม่นยำของการร้องขอทั้งหมดเป็น การเริ่มต้นที่สำคัญต่อนิยามของนโยบายความปลอดภัย

นโยบายการรักษาความปลอดภัย

นโยบายการรักษาความปลอดภัยรวมถึงข้อความ ทั่วไปที่ระบุเวลาที่การร้องขอเกี่ยวข้องกับอ็อบเจ็กต์ที่ระบุจะถูกดำเนินการ ในลักษณะ ของซับเจ็คต์ที่ระบุ ซับเจ็คต์ อ็อบเจ็กต์ และการร้องขอควรถูกกำหนดอย่างระมัดระวัง และนโยบายการรักษาความปลอดภัยควรถูกต้องและชัดเจน เป็นเรื่องที่สำคัญในการระบุการจำแนกของซับเจ็คต์การร้องขอและ อ็อบเจ็กต์ที่เกี่ยวข้องสำหรับจุดประสงค์ของการตรวจสอบ

privilege ขั้นต่ำ:

หลักการของ privilege ขั้นต่ำกำหนดว่าโมดูลซอฟต์แวร์ควรถูก กำหนดความสามารถขั้นต่ำที่จำเป็นในการทำงานที่กำหนดให้สำเร็จ

privilege ขั้นต่ำรวมถึงหลักที่ว่าโปรแกรมที่ไว้วางใจควร จำกัดความสามารถ ที่สำคัญของตัวเองให้ถูกใช้ในพื้นที่ของ โปรแกรมให้น้อยที่สุดเท่าที่เป็นไปได้ privilege ขั้นต่ำลดความเสียหายจากข้อผิดพลาดซอฟต์แวร์ หรือจากผลข้างเคียงที่ไม่คาดคิดซอฟต์แวร์ที่ไว้วางใจทั้งหมดควรถูกออกแบบ ตามหลักของ privilege ขั้นต่ำ

การกำหนดและการเอา privilege ออก:

หนึ่งเทคนิคซอฟต์แวร์ที่ไว้วางใจคือสำหรับโปรแกรมที่ดำเนิน ปฏิบัติการทั้งหมดซึ่งต้องใช้ privilege ในตอนต้น ในการเรียกใช้งาน แล้ว ยกเลิก privilege สำหรับระยะเวลาที่เหลือของการดำเนินการ เรียกว่า privilege bracketing

โปรดจำไว้ว่าข้อควรพิจารณาดังต่อไปนี้เกี่ยวกับการใช้ privileges:

- แต่ละกระบวนการของผู้ใช้ถูกกำหนดชุดของ privileges สูงสุดเมื่อดำเนินการ กระบวนการ ชุดของ privileges นี้สามารถถูกลดลงได้เสมอ แต่จะไม่เพิ่มขึ้น โดยผู้ใช้ unprivileged
- เป็นหน้าที่ของกระบวนการที่เรียกใช้ ในการเพิ่มหรือลด privileges ของชุดสูงสุดเข้าหรือออก effective set เมื่อดำเนินปฏิบัติการ privileged
- privileges กระบวนการถูกแก้ไขเมื่อกระบวนการรันไฟล์เรียกทำงานซึ่ง มีชุด privilege การสืบทอดที่ไม่ว่างเปล่า ดูที่คำสั่ง **exec** สำหรับข้อมูลเพิ่มเติม
- กระบวนการยังถูกจำกัดชุด privilege เมื่อกระบวนการ ถูกรัน ด้วย privileges ที่เหมาะสม กระบวนการสามารถเพิ่ม privileges ในชุดสูงสุด ได้ถึงจำนวนในชุดที่จำกัด

การเปลี่ยนแปลงเลเบล Short-lived MAC:

เมื่อกระบวนการต้องเปลี่ยนแปลงเลเบล MAC จากเลเบลปฏิบัติปกติ ระยะเวลาของการเปลี่ยนแปลงเลเบลควรสั้นที่สุดเท่าที่เป็นไปได้ ซึ่งทำ ให้สำเร็จได้ด้วยการใช้ไลบรารีรูทีน

ดูที่ “การเรียกระบบ Trusted AIX” ในหน้า 547 สำหรับข้อมูลเพิ่มเติมเกี่ยวกับไลบรารีรูทีนเหล่านี้

การเปิดไฟล์สำคัญ Short-lived:

ไฟล์สำคัญคือไฟล์ เช่นไฟล์รหัสผ่าน shadow ที่มี ข้อมูลที่อาจทำให้การรักษาความปลอดภัยระบบมีช่องโหว่ เมื่อไฟล์ที่สำคัญถูกเปิดเพื่ออ่านหรือเขียน ไฟล์ควรถูกเปิดไว้ เท่าที่จำเป็นเท่านั้น

แอ็ตทริบิวต์ **close-on-exec** ของไฟล์ descriptor ควรถูกเซ็ต โดยใช้การเรียกระบบ **fcntl** นี่เป็นการป้องกันกระบวนการที่ไม่ได้รับอนุญาตไม่ให้สืบทอด ไฟล์ descriptors ของไฟล์ที่เปิดอยู่ผ่านการเรียกระบบ **exec**

การรวมศูนย์ของการดำเนินการสำคัญ:

การดำเนินการสำคัญคือการดำเนินการที่ต้องการ privilege ถ้าการดำเนินการที่สำคัญถูกดำเนินการโดยกระบวนการที่ไม่มี privilege อาจทำให้การรักษาความปลอดภัย ของระบบมีช่องโหว่ได้

การดำเนินการสำคัญควรถูกจำกัดกับเฉพาะโมดูล (รูทีนย่อย หรือโปรแกรมแยก) โดยการแยกโปรแกรมขนาดใหญ่ออกเป็นหลายโปรแกรม บางโปรแกรมจะต้องการ privilege น้อยลงหรือไม่ต้องการ privilege ซึ่งช่วยลดความเป็นไปได้ ของการทำให้การรักษาความปลอดภัยของระบบมีช่องโหว่โดยไม่เจตนา

การใช้ไดเร็กทอรี root ที่มีผล:

โปรแกรมสามารถถูกจำกัดให้ดับแผนผังไดเร็กทอรีเฉพาะโดยการตั้งค่า ไดเร็กทอรี root ที่มีผลของโปรแกรม กับไดเร็กทอรีฐานของแผนผัง (พร้อมกับ การเรียกระบบ **chroot**) และการตั้งค่าไดเร็กทอรีทำงานของโปรแกรม ภายในแผนผังเดียวกันนี้ ในทางปฏิบัต นี่เป็นกลไก least-privilege เนื่องจากมีการ จำกัดไฟล์ที่แม้แต่กระบวนการที่มี privilege สามารถเข้าถึงได้ภายในแผนผัง ซึ่งจะมีประสิทธิภาพโดยเฉพาะอย่างยิ่งเมื่อกระบวนการพาเรนท์ (ที่วางใจ) จำกัดกระบวนการไชลด์ ที่ไว้วางใจหรือไม่ไว้วางใจ

เมื่อการเปลี่ยนไดเร็กทอรี root จัดเตรียมการปกป้องไฟล์ภายนอก แผนผัง root ใหม่ จะเป็นต้นเหตุของปัญหาด้านความปลอดภัยที่อาจเกิดขึ้น การเปลี่ยน ไดเร็กทอรี root สามารถสร้างวิธีการสร้างช่องโหว่การรักษาความปลอดภัยของแผนผัง root ใหม่ ถ้าการดำเนินการนี้กระทำอย่างไม่ระมัดระวัง ซึ่งเกิดขึ้นเมื่อ runtime linker และ อ็อบเจ็กต์ที่แบ่งใช้ในแผนผัง root ใหม่สามารถถูกปลอมแปลง ขั้นตอนนี้ควรถูกใช้ อย่างระมัดระวังและใช้เมื่อจำเป็น

การใช้ระบบย่อยที่มีการปกป้อง:

ระบบย่อยที่มีการปกป้องจัดเตรียมการปกป้อง integrity สำหรับระบบย่อยพิเศษ ระบบย่อยคือคอลเล็กชันของโปรแกรม และ/หรือ ไฟล์ข้อมูล ที่ครอบครองโดย ID ผู้ใช้ และ/หรือ ID กลุ่มเดียวกัน ที่ถูกใช้เพื่อสร้างฟังก์ชันใน ระบบ

ระบบย่อยมีโปรแกรม setuid หรือ setgid ได้ ระบบย่อยที่ปกป้อง คือระบบย่อยที่มี ID ผู้ใช้ที่เป็น ID ผู้ใช้ระบบ

ID ผู้ใช้ระบบคือ ID ผู้ใช้ที่มีค่าน้อยกว่าหรือเท่ากับ 127 ผู้ใช้ไม่ สามารถล็อกอินด้วย ID ผู้ใช้ระบบ การใช้ระบบย่อยที่ปกป้องสามารถลด จำนวนของกระบวนการ privilege ได้อย่างชัดเจน

โหมดการเข้าถึงต่ำสุด:

โปรแกรมที่ไว้วางใจ (จริงๆแล้วคือโปรแกรมทั้งหมด) ควรเปิดอ็อบเจ็กต์ ในโหมดการเข้าถึง read/write เท่านั้นตามที่จำเป็น โดยทั่วไป นี่หมาย ถึงอย่าเปิดอ็อบเจ็กต์เพื่อ เขียน–และ–อ่าน เมื่อการเปิดเพื่ออ่าน ก็เพียงพอแล้ว สำหรับสถานการณ์ที่สำคัญ กระบวนการควรเปิด เพื่อเขียนอย่างเดียวในตำแหน่งที่เจาะจง เมื่อจำเป็นต้องเขียน

เทคนิคเหล่านี้มีความสำคัญมาก เมื่อโปรแกรมสร้างกระบวนการอื่น เนื่องจากการส่งผ่าน privilege และความสามารถทั่วไป อื่นๆ (เช่น เปิดการเชื่อมต่อไปที่ไฟล์ที่มีความสำคัญ) เป็นแง่มุมที่สำคัญของ การออกแบบซอฟต์แวร์ที่ไว้วางใจ Privileges สามารถแทนที่ข้อจำกัดทั้งหมด ควร ออกแบบอย่างระมัดระวัง และข้อควรพิจารณาควรถูกนำมาใช้ เมื่อสร้างคำสั่งใหม่ที่จะมี privileges

ระเบียบโปรแกรมมิงที่ไว้วางใจอื่น:

Trusted AIX ใช้ ระเบียบโปรแกรมมิงที่ไว้วางใจหลายข้อ

ความซ้ำซ้อน:

ความซ้ำซ้อนเป็นเทคนิคที่มีประโยชน์สำหรับระบบความปลอดภัย การรักษาความปลอดภัย ไม่มีกฎตายตัว ส่วนใหญ่เป็นเรื่อง ของการกำหนดอุปกสรรคที่มีประสิทธิภาพ เพื่อป้องกันผู้ที่พยายามเข้าถึงระบบ อย่างไม่ถูกต้อง

ประโยชน์ของการตรวจสอบความปลอดภัยซ้ำซ้อนคือถ้าการตรวจสอบหนึ่งล้มเหลวหรือถูกทำให้มี ช่องโหว่ การตรวจสอบอื่น อาจสามารถป้องกันได้ ข้อเสียของการตรวจสอบซ้ำซ้อนคือ การตรวจสอบความปลอดภัยโดยรวมถูกแยกหรือกระจายทั่ว ระบบ ดังนั้น ขณะที่การตรวจสอบซ้ำซ้อนมีประโยชน์อย่างมาก แต่ต้องถูก ออกแบบ ทำเอกสาร และดูแลอย่างระมัดระวัง

การตรวจสอบเคอร์เนลไม่ซ้ำ:

ไม่บ่อยครั้งที่มีการแนะนำให้ใช้กระบวนการทำการตรวจสอบที่ เคอร์เนลสามารถทำได้ ตัวอย่างเช่น กระบวนการไม่ควร อ่านเลเบล MAC ของไฟล์และดำเนินการตรวจสอบการเข้าถึงด้วยตัวเอง เมื่อใดก็ตามที่เป็นไปได้ การตรวจสอบเคอร์เนลควร ดำเนินการตรวจสอบ

มีสองเหตุผลหลักที่เคอร์เนลควรทำการตรวจสอบ

- การดำเนินการเคอร์เนล atomic กับกระบวนการอื่น โดยที่ การตรวจสอบกระบวนการเกิดขึ้นได้อย่างมีประสิทธิภาพพร้อม กับกระบวนการอื่น
- ที่สำคัญกว่านั้น อัลกอริทึมที่ใช้สามารถเปลี่ยนแปลงได้ตามเวอร์ชัน เคอร์เนลที่ใหม่กว่า เป็นเรื่องยากในการติดตามการ เปลี่ยนแปลงของอัลกอริทึมที่เป็น ส่วนหนึ่งของซอฟต์แวร์ผู้ใช้ขั้นปลาย

การตรวจสอบ privilege โดยตรง:

โปรแกรมไม่ควรจะกำหนดว่าถูกเรียกในแบบ กระบวนการที่มี privilege (ตัวอย่างเช่น โดยการตรวจสอบเวกเตอร์ effective หรือ maximum privilege) หรือไม่ แต่ โปรแกรมควรยึดว่าถูกเรียกเป็นแบบมี privilege เมื่อเหมาะสม

ถ้าโปรแกรมไม่ได้เป็นกระบวนการ privilege การเรียกระบบ privileged จะล้มเหลวและโปรแกรมสามารถดำเนินการตาม ความเหมาะสมได้ โดยปกติไม่ใช่การวัดผลความปลอดภัยที่มีประสิทธิภาพสำหรับโปรแกรมเอง ในการปฏิเสธที่จะดำเนินการ นอกว่าโปรแกรมมี privilege ถ้าโปรแกรมมี privilege การตรวจ สอบจะไร้ความหมาย ถ้าโปรแกรมไม่มี privilege โปรแกรมไม่ สามารถ ทำอันตรายใดมากไปกว่ากระบวนการที่ไม่ได้รับ privilege อื่น

อย่างไรก็ตาม การตรวจสอบนี้สามารถถูกใช้อย่างมีผล ในฐานะการช่วยเหลือการใช้งานไม่ถูกต้องโดยไม่เจตนา ข้อความแสดงความผิดพลาดที่มีความหมายสามารถถูกกำหนดเพื่อแจ้งว่าโปรแกรม ควรมี privilege แต่ไม่มี

การกระจายของความสามารถที่มีความละเอียดอ่อน:

ความสามารถที่มีความละเอียดอ่อนคือความสามารถของโปรแกรมที่ไว้วางใจที่ สามารถก่อให้เกิดช่องโหว่ด้านความปลอดภัยต่อระบบถ้ามอบให้แก่โปรแกรมที่ไม่ได้รับความไว้วางใจ

ควรใช้ความระมัดระวังเมื่อโปรแกรมสิทธิ์พิเศษกระจายสิทธิ์พิเศษ หรือความสามารถทั่วไปไปยังโปรแกรมอื่นโดยใช้กลุ่ม **fork** และ **exec** ของการเรียกใช้ระบบ การเรียกใช้ระบบ **exec** มีความสำคัญที่สุดเนื่องจาก เป็นการผ่านสิทธิ์พิเศษจากโปรแกรมหนึ่งไปอีกฌหนึ่ง การเรียกใช้ระบบ **fork** จะสร้างกระบวนการใหม่ แต่สิทธิ์พิเศษของกระบวนการใหม่จะเหมือน กับของพาเรนต์ อันตรายที่สำคัญที่สุดคือไฟล์โปรแกรมเรียกทำงานนั้น อาจไม่สามารถไว้วางใจได้ หรืออาจถูกเปลี่ยนแปลงโดยโปรแกรมที่ไม่ไว้วางใจ ข้อควรระวังต่อไปนี้ควรนำมาใช้พิจารณา:

- โปรแกรมที่ไว้วางใจควรระมัดระวังที่จะไม่ส่งการเชื่อมต่อที่เปิดไปยังอ็อบเจ็กต์ (ไฟล์สำคัญ) ไปยังกระบวนการชายด์ยกเว้นชายด์และกระบวนการที่สืบทอด สามารถไว้วางใจโดยใช้การเข้าถึงไฟล์ที่เหมาะสมในโหมดที่ไฟล์ถูก เปิดใช้ อาจเป็นวิธีดีที่สุดสำหรับกระบวนการที่จะส่งการเชื่อมต่อใหม่ไปยังอ็อบเจ็กต์ ที่มีโหมดมีจำกัดมากกว่าอ็อบเจ็กต์ที่ไม่มี
- กระบวนการที่ไว้วางใจที่รันด้วยไดเร็กทอรี root ที่ใช้งาน นอกเหนือจาก root สัมบูรณ์ควรได้รับความเชื่อมั่นว่ากระบวนการชายด์จะไม่ถูกทำให้สับสน ตัวอย่าง เมื่อโปรแกรมชายด์เปิดไฟล์ที่ไว้วางใจ เช่นไฟล์รหัสผ่านที่ซ่อน โปรแกรมสามารถใช้ชื่อพาธสัมบูรณ์ภายใต้สมมติฐานที่ว่า root ที่ใช้งานเป็นค่าสัมบูรณ์
- อาจมีบางกรณีที่โปรแกรมที่ไว้วางใจจำเป็นต้องบังคับให้มี umask ที่จำกัดมากขึ้นบนชายด์
- แอ็ตทริบิวต์กระบวนการหลายแอ็ตทริบิวต์ได้รับสืบทอดโดยกระบวนการชายด์ ถ้าโปรแกรม ที่ไว้วางใจทราบว่ากระบวนการชายด์ไม่ได้รับการไว้วางใจ และมีเลเบล MAC ที่ไม่ ครอบคลุมของกระบวนการที่ไว้วางใจนั้น และแอ็ตทริบิวต์เหล่านี้ถูกสืบทอด โดยโปรแกรมที่ไว้วางใจจาก ancestor ที่ไม่ไว้วางใจ ดังนั้นแอ็ตทริบิวต์เหล่านี้สามารถ เป็นแหล่งของช่องทางที่ใช้ปิดบังที่เป็นไปได้
- โปรดระวังกฎของการกระจายสิทธิ์พิเศษสำหรับการเรียกใช้ระบบ **fork** และ **exec** สิทธิ์พิเศษของกระบวนการพาเรนต์กลายเป็นสิทธิ์พิเศษของกระบวนการ ชายด์เมื่อมีการเรียกใช้ระบบ **fork** เกิดขึ้น สิทธิ์พิเศษถูกแก้ไขระหว่าง การเรียกใช้ระบบ **exec**

ในสถานการณ์ที่มีความละเอียดอ่อนอย่างยิ่ง โปรแกรมที่ไว้วางใจสามารถตรวจสอบการควบคุม การเข้าบนไฟล์ที่ไว้วางใจเพื่อช่วยให้มั่นใจว่าไฟล์ได้รับการปกป้องอย่างเหมาะสม จากการแก้ไขโดยโปรแกรมที่ไม่ได้รับการไว้วางใจ ตัวอย่าง ไฟล์สามารถถูกร้องขอ ให้ root เป็นเจ้าของด้วยสิทธิ์การเขียน DAC ส่วนใหญ่ที่อนุญาตสำหรับเจ้าของ ไฟล์

สภาวะแวดล้อม Effective root:

โปรแกรมที่ไว้วางใจบ่อยครั้งขึ้นอยู่กับชื่อพาธสัมบูรณ์ที่ถูกต้อง ตัวอย่างเช่น โปรแกรม **login** ขึ้นกับไฟล์ `/etc/security/passwd` เพื่อใช้เป็นไฟล์รหัสผ่าน shadow ที่ถูกต้อง

นี่ไม่เพียงแค่ไฟล์ข้อมูล แต่รวมถึงไฟล์เรียกทำงานสำหรับ โปรแกรมที่ไว้วางใจ ขณะที่โปรแกรมที่ไม่ไว้วางใจไม่สามารถใช้การเรียกระบบ **chroot** เพื่อ เปลี่ยนไดเร็กทอรี effective root ของโปรแกรมได้โดยตรง อาจมีสถานการณ์ซึ่ง TCB อนุญาตให้โปรแกรมที่ไม่ไว้วางใจรันภายใต้ effective root มีปัญหาการรักษาความปลอดภัยที่เกิดขึ้นได้ ถ้าโปรแกรมที่ไม่ไว้วางใจเหล่านี้สามารถเรียกใช้โปรแกรมที่ไว้วางใจที่ขึ้นกับชื่อพาธสัมบูรณ์

การพิสูจน์ตัวตนด้วย ID จริงและ effective ID:

โปรแกรมที่ไว้วางใจอาจจำเป็นต้องใช้ ID เชื่อมโยงและ ID กลุ่มที่ ถูกเชื่อมโยงกับกระบวนการ เป็นสิ่งสำคัญที่ต้องเข้าใจความแตกต่างระหว่าง ID เหล่านี้และการใช้งานที่เหมาะสม

ID ผู้ใช้จริงและ ID กลุ่ม

ID ผู้ใช้จริงและ ID กลุ่มโดยปกติ แสดงล็อกอิน identity ของล็อกอินเซสชันซึ่งกระบวนการถูกสร้าง ในบางกรณี ID จริง (โดยเฉพาะ ID ผู้ใช้จริง) สามารถถูกใช้สำหรับการตัดสินใจ ด้านความปลอดภัย ตัวอย่างเช่นการตรวจสอบการอนุญาต ID ผู้ใช้จริงถูกใช้โดยคำสั่งเป็นรูปแบบของการตรวจสอบ identity ซึ่งมีประโยชน์ ในการขัดขวางการประสงค์ร้ายหรือการใช้ที่ไม่ระวังของบิตควบคุม **setuid-on-exec** หรือ **setgid-on-exec** อย่างไรก็ตาม การตรวจสอบ ID จริงแยกจากแนวปฏิบัติ UNIX มาตรฐานและควรกระทำเมื่อจำเป็นเท่านั้น หลักการโดยรวมในระบบ UNIX คือ effective ID ถูกใช้สำหรับเข้าถึงและการตรวจสอบเกี่ยวกับความปลอดภัยอื่น นอกจากแนวปฏิบัติที่ยอมรับนี้ ไม่ควรถูกกระทำโดยปราศจากการพิจารณาอย่างรอบคอบและ การจัดทำเอกสาร

ID ผู้ใช้ Effective และ ID กลุ่ม

ID ผู้ใช้ Effective และ ID กลุ่มควรถูกใช้ในการตัดสินการควบคุมการเข้าถึงทั้งหมด (DAC และ MAC) ผู้ใช้ระบบ มีค่า ID ผู้ใช้ระหว่าง 0 และ 127 ผู้ใช้ปกติมีค่า ID 128 และสูงกว่า

ชื่อพาธสัมบูรณ์สำหรับคำสั่งที่ไว้วางใจ:

ความพยายามที่มีรูปแบบการแทรกซึมด้านความปลอดภัยบางอย่างจะสร้างโปรแกรม ที่ไว้วางใจปลอม และวางไว้ในพาธการค้นหาของโปรแกรมที่คล้ายเชลล์ที่ถูกใช้โดยผู้ใช้ด้านการดูแลจัดการหรือแม้แต่ผู้ใช้ปกติ ตัวอย่าง สำเนาปลอม ของคำสั่ง **passwd** สามารถถูกใช้เพื่อดักจับรหัสผ่านผู้ใช้ที่มี หรือรหัสผ่านใหม่

แนวทางปฏิบัติด้านการดูแลจัดการที่เหมาะสมสำหรับไดเร็กทอรีที่กำลังทำงาน ปัจจุบันถูกลบออกจากพาธการค้นหาเพื่อป้องกันในเรื่องนี้ อย่างไรก็ตาม มีหลาย พาธการค้นหาที่ไม่ได้รับการป้องกันอย่างเข้มงวดเท่าที่จำเป็น และผู้ใช้ปกติ ต้องได้รับอนุญาตให้ใส่ไดเร็กทอรีที่กำลังทำงานปัจจุบันในพาธการค้นหา ของตน ตัววัดการนับที่ใช้งานอยู่สำหรับโปรแกรมที่ไว้วางใจจะถูกเรียกใช้เสมอ โดยใช้ชื่อพาธสัมบูรณ์ (ตัวอย่าง `/usr/bin/passwd`) โปรแกรมที่ไว้วางใจเองจะตรวจสอบอากิวเมนต์การร้องขอแรก และชื่อการร้องขอ ของตน ถ้าไม่ได้ใช้ชื่อพาธสัมบูรณ์ที่เหมาะสม โปรแกรมที่ไว้วางใจ จะปฏิเสธการทำงาน โปรแกรมที่ไว้วางใจควรตรวจสอบให้แน่ใจอยู่เสมอว่าไม่มี ไดเร็กทอรี root ที่ใช้งานอยู่ที่แตกต่างจาก root สัมบูรณ์

หมายเหตุ: นี้มีผลกับขอบเขตที่ผู้ใช้ได้รับการอบรมเพื่อเรียกใช้ชื่อพาธสัมบูรณ์เท่านั้น ถ้าผู้ใช้ใช้ชื่อพาธสัมพัทธ์โดยไม่เจตนา แทน และโปรแกรมปลอมถูกร้องขอ รูปแบบของการแทรกซึมด้านความปลอดภัย ไม่ถูกป้องกัน

การจัดโครงสร้างแผนผังไดเร็กทอรี:

แผนผังไดเร็กทอรีควรถูกจัดโครงสร้างอย่างระมัดระวังเพื่อเพิ่มการป้องกัน ไฟล์สำคัญ คำแนะนำพื้นฐานคือการเข้าถึงไดเร็กทอรีเพื่อค้นหาควรถูก จำกัดเท่าที่เป็นไปได้ (ตัวอย่างเช่น การกำหนดไฟล์ที่เข้าถึงได้แบบพับลิก กับไดเร็กทอรีที่ใกล้กับ root ของระบบไฟล์)

และยังเป็นความคิดที่ดีในการกำหนดไดเร็กทอรีที่สำคัญมาใกล้กับ root สัมบูรณ์เท่าที่เป็นไปได้ เนื่องจากเป็นการลดจำนวนไดเร็กทอรี ตัวกลางที่จำเป็นต้องถูกป้องกัน

ระบบไฟล์อ่านอย่างเดียว:

บางทีพื้นฐานในการจัดโครงสร้างแผนผังไดเร็กทอรีคือ ไฟล์ที่ไว้วางใจ ที่มีการเปลี่ยนแปลงน้อยมากถูกเก็บไว้บนระบบไฟล์ของตัวเองและกำหนดเป็น อ่านอย่างเดียว นี่เป็นการทำให้แน่ใจอย่างแท้จริงว่าเนื้อหาจะไม่สามารถถูกแก้ไข ระหว่างการดำเนินการระบบปกติ เทคนิคนี้ใช้บ่อยครั้งสำหรับคอลเล็กชันขนาดใหญ่ ของไฟล์เรียกทำงานสำหรับโปรแกรมที่ไว้วางใจ

ถ้าจำเป็นต้องแก้ไขไฟล์ ระบบไฟล์สามารถถูกแยก เป็นเขียนได้ ในบริบทที่มีการป้องกันเพิ่มเติม (เช่นในโหมดผู้ใช้คนเดียว หรือ แยก เครื่องที่มีการป้องกันมากขึ้น) ขอแนะนำให้ใช้โปรแกรมในการ สแกนระบบไฟล์เพื่อ configuration ที่ถูกต้อง (ตัวอย่างเช่นเลเบล DAC, MIC และ MAC ที่ถูกต้อง) หลังจากการอัพเดต

นอกจากนี้ข้อมูล DAC, MIC และ MAC ไม่สามารถถูกแก้ไขในระบบไฟล์ อ่านอย่างเดียว เมื่อระบบไฟล์ถูกตั้งค่าอย่างถูกต้อง นี่ควรป้องกันรูปแบบการเจาะ การรักษาความปลอดภัย ที่พยายามเปลี่ยนข้อมูลเลเบล DAC และ/หรือ MIC และ MAC

การจัดการรหัสผ่าน:

เป็นแนวทางปฏิบัติที่ไม่ดีในการที่โปรแกรมที่ไม่ใช่ยูทิลิตีระบบมาตรฐาน ทำการสอบถามผู้ใช้ถึงรหัสผ่านล็อกอิน รหัสผ่านเป็นข้อมูลสำคัญมาก และการจัดการควรมีข้อบังคับเคร่งครัด กับยูทิลิตีระบบ well-trusted ที่มีอยู่ไม่มาก

อาจเป็นสิ่งที่ทำได้ในบางระบบย่อยที่ไว้วางใจในการสร้าง รหัสผ่านจำเพาะขึ้นเอง อย่างไรก็ตาม เป็นเรื่องอันตรายที่จะเชื่อถือรูปแบบรหัสผ่านไพรเวต เนื่องจากไม่ปลอดภัยเท่ากับกลไก system-enforced

การปกป้อง Trusted Computing Base (TCB):

ไฟล์ที่มีองค์ประกอบของ TCB ต้องถูกป้องกันจากการแก้ไข และในบางกรณีการเปิดเผย (การอ่าน) โดยโปรแกรม ที่ไม่ไว้วางใจ

การปกป้องจากการแก้ไขเป็นสิ่งสำคัญ และการปกป้องจากการเปิดเผย ก็เช่นกัน ไฟล์ต้องถูกป้องกันรวมถึง:

- ไฟล์ทั้งหมดที่มีข้อมูลที่ใช้โดยโปรแกรมที่ไว้วางใจในการสร้าง การตัดสินใจด้านความปลอดภัย (ตัวอย่างเช่นไฟล์รหัสผ่านเงา)
- ไฟล์เรียกทำงานทั้งหมดสำหรับโปรแกรมที่ไว้วางใจ
- Pseudofiles ที่อนุญาตการเข้าถึงส่วนของ TCB (ตัวอย่างเช่น `/dev/kmem`)

หมายเหตุ: ไฟล์การเตรียมข้อมูลระบบ (ไฟล์ rc) ต้องถูกป้องกันเป็นพิเศษ เนื่องจากเป็นส่วนหนึ่งของ TCB

การปกป้องจากการแก้ไขข้อมูล:

การปกป้องจากการแก้ไขที่ไม่ได้รับอนุญาตโดยหลักแล้วทำได้โดย การตั้งค่าข้อมูล DAC เป็นค่าที่เหมาะสม โดยปกติ ไฟล์เหล่านี้ จะเป็นของ ID ผู้ใช้ระบบโดยที่การเข้าถึงเพื่อเขียนได้รับอนุญาตเฉพาะ เจ้าของไฟล์เท่านั้น

MIC ถูกออกแบบเพื่อป้องกันการแก้ไขโดยการป้องกัน integrity ของอ็อบเจ็กต์โดยการกำหนดเลเบล MIC มีค่าสูงบนไฟล์กระบวนการที่มีเลเบล MIC ค่าต่ำกว่า ถูกป้องกันไม่ให้ แก้ไข ลบ หรือเปลี่ยนชื่อไฟล์ นี่เป็น เมธอดที่ดีที่สุดในการป้องกันการแก้ไขไฟล์ที่ไม่ต้องการ

ในบางกรณี MAC สามารถถูกใช้เพื่อป้องกันการแก้ไขข้อมูลที่ไม่อนุญาต อย่างไรก็ตาม MAC ถูกออกแบบมาเพื่อป้องกันเฉพาะการเปิดเผยข้อมูล (ก่อนอ่าน) และ ไม่เหมาะนักสำหรับการป้องกันการแก้ไข นโยบาย MAC ระดับต้นไม่ ยับยั้งซับเจ็คต์

จากการแก้ไขอ็อบเจ็กต์ higher-label แม้ว่าไม่อนุญาต ให้เขียนไฟล์โดยตรง บางระบบย่อยที่ไว้วางใจอาจอนุญาตให้ทำได้ นอกจากนี้ ไฟล์ที่ไว้วางใจ จำนวนมากเช่นไฟล์โปรแกรมเรียกทำงาน จำเป็นต้องถูกเก็บไว้ที่เลเบล MAC ระดับต่ำเพื่อที่สามารถถูกเข้าถึงได้โดยทั่วไป ดังนั้น การตั้งค่าเลเบล MAC ให้มีค่าสูงในไฟล์ไม่เหมาะเสมอไป

แฟล็กความปลอดภัยของไฟล์ ป้องกันไฟล์จากการถูกแก้ไขเช่นกัน แฟล็กความปลอดภัยของไฟล์ บางแฟล็ก ป้องกันการแก้ไขอ็อบเจ็กต์ แม้แต่อ็อบเจ็กต์ privileged ถ้าแฟล็กความปลอดภัยของไฟล์ **FSF_TLIB** ถูกเซ็ตให้กับไฟล์ ไฟล์จะถูกเปลี่ยนแปลงได้เฉพาะเมื่อระบบอยู่ใน โหมด configuration โดยถือว่าแฟล็กความปลอดภัยเคอร์เนล **trustedlib_enabled** เปิดอยู่ เมื่อต้องการเซ็ต **FSF_TLIB** สำหรับไฟล์ กระบวนการต้องมี **PV_TCB** privilege ใน EPS แฟล็กความปลอดภัยที่เกี่ยวข้องอื่นคือแฟล็ก **FSF_APPEND** ซึ่งป้องกันการแก้ไขข้อมูล ที่เขียนไปก่อนหน้านี้ ไฟล์ที่มีแฟล็ก **FSF_APPEND** เซ็ตไว้ถูกเพิ่มข้อมูลได้ เท่านั้น ซึ่งมีประโยชน์สำหรับแอ็พพลิเคชันที่บันทึกการทำงานลง ไฟล์

แฟล็กเหล่านี้โดยปกติเซ็ตให้กับไฟล์โดย integrators แทนที่จะอยู่ภายใต้ การควบคุมของโปรแกรม โปรแกรมเมอร์ควรตระหนักถึงแฟล็กเหล่านี้และการทำงานของแฟล็ก

การปกป้องจากการเปิดเผยข้อมูล:

DAC และ MAC สามารถถูกใช้เพื่อป้องกันไฟล์ TCB จากการเข้าถึงเพื่ออ่านข้อมูล เลเบล MAC บนไฟล์เหล่านี้ต้องสะท้อนอย่างแม่นยำถึงความสำคัญของ ข้อมูลในไฟล์เหล่านี้ ตัวอย่างเช่น ถ้าอัลกอริทึมถูกจัดประเภท ดังนั้นเลเบล MAC บนไฟล์เรียกทำงานของโปรแกรมที่ใช้อัลกอริทึม ต้องถูกเซ็ตอย่างเหมาะสม

เป็นการปฏิบัติที่ยอมรับได้ในการเซ็ตเลเบล MAC ให้มีค่าสูง (นั้นคือ สูงกว่าการจัดประเภทจริงของข้อมูลในไฟล์) เพื่อป้องกัน การเปิดเผยข้อมูล อย่างไรก็ตามการจัดประเภทที่เพิ่มขึ้นควรถูก ใช้เท่าที่จำเป็น

ในกรณีส่วนใหญ่ ลูกโซ่ไดเร็กทอรีทั้งหมดจาก root สัมบูรณ์ต้องถูก ป้องกันเพื่อให้ตัวไฟล์ได้รับการป้องกันที่เพียงพอ มิฉะนั้นโปรแกรมที่ประสงค์ร้ายอาจสามารถยกเลิกการลิงก์ส่วนของลูกโซ่ไดเร็กทอรี และสร้าง subtree ใหม่ที่มีสำเนาปลอดของไฟล์

ตัวอย่างเช่น สมมุติว่าไฟล์ที่ไว้วางใจถูกเก็บที่ `/A/B/foo` ขณะที่ **foo** ถูกป้องกันจากการแก้ไข แต่ไดเร็กทอรี **B** ไม่ได้ ถูกป้องกัน โปรแกรมที่ประสงค์ร้ายอาจลบลิงก์ใน **B** ไปที่ **foo** ออกและ สร้างไฟล์ **foo** ใหม่ด้วยสำเนาไม่ถูกต้องของไฟล์ **foo** เก่า จากนั้นโปรแกรมที่ไว้วางใจที่เปิด `/A/B/foo` จะเปิดไฟล์ ที่ไม่ถูกต้องและจะถูกลวงให้ใช้ข้อมูลไม่ถูกต้องของไฟล์

โปรแกรมที่ไว้วางใจขึ้นอยู่กับชื่อพาธที่ถูกต้องเพื่อเข้าถึงไฟล์ TCB ด้วยเหตุนี้ ไฟล์ลิงก์สัญลักษณ์ที่ใช้ในชื่อพาธสำหรับไฟล์ TCB ควรถูกป้องกัน อย่างเคร่งครัดเหมือนตัวไฟล์เอง

ในบางกรณี MIC สามารถถูกใช้เพื่อป้องกันการเปิดเผยข้อมูลที่ไม่อนุญาต อย่างไรก็ตาม MIC มีหน้าที่หลักสำหรับการป้องกันการแก้ไขเท่านั้น (การเขียนข้อมูล) และไม่เหมาะนักสำหรับการป้องกันการเปิดเผยข้อมูล

การดำเนินการเลเบลระดับความลับ:

มีคำแนะนำของโปรแกรมที่ไว้วางใจสำหรับสถานการณ์ที่เกี่ยวข้องกับซับเจ็คต์หรือ อ็อบเจ็กต์ที่มีเลเบลระดับความลับต่างกัน

คุณควรทำความคุ้นเคยกับฟอร์มของเลเบลระดับความลับและความสัมพันธ์ การควบคุมระหว่างเลเบล ค่าสูงกว่าคือการได้ควบคุม และค่าต่ำกว่า คือการถูกควบคุม ขณะที่การอัพเกรดหมายถึงการเพิ่มการจัดประเภท ของข้อมูลเป็นเลเบลที่สูงกว่า และการดาวน์เกรดหมายถึงการลดการจัดประเภท ของข้อมูลสู่เลเบลต่ำกว่า

ข้อจำกัด MAC ระดับต้น:

ข้อจำกัดค่าควบคุมการเข้าใช้พื้นฐานคือซับเจ็คต์ที่ไม่ไว้วางใจ ไม่สามารถทำให้ข้อมูลที่มีเลเบลที่ระดับความลับเลเบล A กลายเป็นเลเบลที่ B นอกจาก B ควบคุม A

ข้อจำกัด MAC ระดับต้นครอบคลุมคลาสข้อมูลทั้งหมด ซึ่งรวมถึงข้อจำกัด การกำหนดเลเบลข้อมูลใหม่ (นั่นคือการเปลี่ยนเลเบลบนคอนเทนเนอร์ข้อมูล) และใน การย้ายข้อมูลที่เบเบลระหว่างคอนเทนเนอร์ข้อมูล

ที่ระดับของระบบต่างกัน (การเรียกระบบ ยูทิลิตีเซอร์วิสระบบ และอื่นๆ)ข้อจำกัดระดับต้นนี้ถูกแปลงเป็นชุดกฎที่จำเพาะมากกว่า แต่มีปรัชญา พื้นฐานเดิมเสมอ ข้อมูลสามารถถูกอัพเกรด อย่างมากที่สุด ตัวอย่างเช่น ระดับของส่วนขยายแรกคือกระบวนการนั้นสามารถเปิดเพื่ออ่าน คลาสขนาดใหญ่ของอ็อบเจ็กต์ เมื่อเลเบลของกระบวนการควบคุมเลเบลของ อ็อบเจ็กต์ และเปิดสำหรับการเขียน ถ้าเลเบลของอ็อบเจ็กต์ควบคุม กระบวนการ

สำหรับไฟล์ปกติ การเขียนถูกจำกัดเพิ่มเติมกับไฟล์ ที่เลเบลเดียวกันกับกระบวนการ สำหรับไดเร็กทอรีและอุปกรณ์ การดำเนินการเขียน ได้รับอนุญาต ถ้าซับเจ็คต์ SL ควบคุมอ็อบเจ็กต์ minimum SL และอ็อบเจ็กต์ maximum SL ควบคุมซับเจ็คต์ SL สำหรับไฟล์พิเศษ FIFO (named pipes) การดำเนินการอ่านถูกจำกัดกับไฟล์พิเศษ FIFO เช่นกันที่เลเบลเดียวกัน กับกระบวนการสำหรับเหตุผลแชนเนลการแปลง

ขณะที่ข้อมูลสามารถย้ายไปที่เลเบลระดับความลับที่สูงกว่า ความสามารถนี้ไม่จำเป็น สำหรับอ็อบเจ็กต์และสถานการณ์ที่กำหนด ตัวอย่างเช่น ระบบปฏิบัติการ เองไม่อนุญาตให้กระบวนการที่ไม่มี privilege เปิดไฟล์เลเบลที่สูงกว่า สำหรับการเขียน แม้ว่าทำได้ภายใต้ข้อจำกัด MAC ระดับต้น การอนุญาต การอัพเกรดนี้กับซับเจ็คต์ที่ไม่ไว้วางใจเป็นเรื่องของการออกแบบและปรัชญา ในบางกรณีมีประโยชน์ และบางกรณีก็ไม่มีประโยชน์ ตัวอย่างเช่น ความยาก ที่เชื่อมโยงกับการเขียนโดยตรงไปที่ไฟล์ที่มีเลเบลสูงกว่า คือกระบวนการ ไม่สามารถอ่านไฟล์เหล่านี้ได้ ดังนั้นการเขียนไปที่ไฟล์ที่เลเบลสูงกว่า ไม่มีประโยชน์ อย่างไรก็ตาม ยูทิลิตีที่ไว้วางใจธรรมดา ที่เพิ่มเลเบล ของไฟล์ที่การร้องขอของซับเจ็คต์ที่ไม่ไว้วางใจเป็นเรื่องยอมรับได้และ เป็นยูทิลิตีที่มีประโยชน์

ที่ระดับการเรียกใช้ระบบ ข้อจำกัดมีเฉพาะบนกระบวนการที่ไม่มี privilege ซึ่งหมายความว่ากระบวนการที่มี privilege ไม่ถูกโยงกับข้อจำกัดนี้ อย่างไรก็ตาม เซอร์วิสทั้งหมดที่ระบบที่ไว้วางใจดำเนินการจะถูกออกแบบสำหรับผู้ใช้ที่ไม่ไว้วางใจ และดังนั้นที่ระดับ user-service ข้อจำกัดมีการควบคุมล่วงหน้า

ข้อจำกัด MAC ระดับต้นใช้กับวิธีทั้งหมดที่โปรแกรมที่ไม่ไว้วางใจ ใช้ในการจัดการเพื่อถ่ายโอนข้อมูล อย่างไรก็ตาม ข้อจำกัด MAC ระดับต้น บ่อยครั้งแบ่งออกเป็นสองคอมโพเนนต์ คอมโพเนนต์แรกจัดการเฉพาะกับคุณลักษณะ ระบบปฏิบัติการที่กำหนดสำหรับการถ่ายโอนข้อมูล (หรือการเลเบล) คุณลักษณะ เหล่านี้รวมถึงการอ่านและการเขียนไฟล์ และการสื่อสารข้อมูลระหว่างกระบวนการ เป็นตัวอย่าง คอมโพเนนต์ที่สองจัดการกับวิธีของการสื่อสารที่ไม่ได้กำหนดไว้ดังกล่าว เรียกว่าแชนเนลการแปลง มันเกือบจะเป็นไปไม่ได้ที่จะบังคับใช้ ข้อจำกัด MAC ระดับต้นอย่างสมบูรณ์โดยมีการเกี่ยวข้องกับแชนเนลการแปลง ด้วยเหตุนี้ อัตราข้อมูลต่ำ (ตัวอย่างเช่น 0.1 บิตต่อวินาที) แชนเนลการแปลง ได้รับอนุญาตให้มีอยู่ แม้ว่าเฉพาะเมื่อมีการแลกที่สมเหตุผลกับ ปัจจัยอื่น

ข้อจำกัด MAC ระดับต้นตรงไปตรงมาและธรรมดา และมีแนวทาง ละเอียดไม่มากนักสำหรับการจัดการข้อมูลหลายระดับ

การดำเนินการหลายระดับ:

การเรียกระบบ **sec_setplab** อนุญาตให้กระบวนการที่มี privilege ในการเปลี่ยนแปลงเลเบลของกระบวนการแบบไม่มีเงื่อนไข

เนื่องจากข้อบังคับ MAC และ MIC ทั้งหมดในกระบวนการที่ไม่มี privilege ถูกบังคับใช้ ด้วยเช่นกันสำหารับกระบวนการที่มี privilege ในการเรียกระบบที่มีอยู่ก่อนแล้ว (นั่นคือ ที่ถูกกำหนดในระบบปฏิบัติการฐาน) กระบวนการที่มี privilege ที่จำเป็นในการดำเนินการหลายระดับ ต้องขึ้นอยู่กับการเรียกระบบ **sec_setplab** อย่างมาก อย่างไรก็ตาม โปรแกรมที่ไว้วางใจควรใช้เฉพาะ sec_setplab() ใน รูปแบบดังต่อไปนี้:

- การใช้การเรียกระบบ **sec_setplab** ทั้งหมดเพื่อดำเนินการหลายระดับ (ตัวอย่างเช่น การเปิดไฟล์เลเบลที่สูงกว่าเพื่ออ่าน) ควรกระทำเฉพาะ ผ่านไลบรารีรูทีนที่สะท้อนซีแมนทิกส์ของค่าจริง การดำเนินการระดับสูง ที่ดำเนินการที่ซ่อนรายละเอียดการใช้การเรียกระบบ **sec_setplab**
- ข้อยกเว้นเดียวคือการเปลี่ยนเลเบลกระบวนการที่ธรรมดามากซึ่งไม่ได้เป็น ส่วนหนึ่งของการดำเนินการหลายระดับที่ใหญ่กว่า การดำเนินการง่ายๆ เหล่านี้สามารถใช้การเรียกระบบ **sec_setplab** ได้โดยตรง

มีสองเหตุผลสำหรับคำแนะนำนี้สำหรับการเรียกระบบ **sec_setplab** ข้อแรกคุณลักษณะที่มีความสำคัญและอาจมีอันตรายเช่น การเรียกระบบ **sec_setplab** ควรถูกใช้เฉพาะเมื่อมีการออกแบบมาดี มอดูลาร์ ข้อสอง ตามมาตรฐาน สำหรับการพัฒนาระบบที่ไว้วางใจ การเรียกระบบระดับต่ำอาจสนับสนับกลไกต่างๆ สำหรับการดำเนินการหลายระดับ

การครอบคลุมการดำเนินการระดับสูงให้อยู่ในไลบรารีรูทีนจัดให้มี ความเข้ากันได้แบบรุดหน้าที่ดีและปรับเปลี่ยนได้ เพื่อพัฒนาเวอร์ชันของระบบปฏิบัติการ และช่วยประกันความสามารถในการพอร์ตระหว่างเวอร์ชันของระบบ UNIX

ระบบที่ไว้วางใจจัดเตรียมชุดระดับต้นของรูทีนดังกล่าว รูทีนเหล่านี้ ควรใช้เมื่อใดก็ตามที่เป็นไปได้ ชุดของรูทีนนี้ควรถูกขยายด้วยเวอร์ชันระบบปฏิบัติการต่อเนื่อง โปรแกรมเมอร์ระบบที่ไว้วางใจสามารถ สร้างไลบรารีรูทีนดังกล่าวได้เมื่อต้องการ

อีกข้อยกเว้นกับข้อบังคับ MAC และ MIC คือการใช้ MAC ที่มีหรือ MIC privilege เพื่อเลี่ยงการยับยั้ง MAC หรือ MIC ควรระมัดระวังเมื่ออนุญาตให้ใช้ privileges เหล่านี้

System V Interprocess Communication (IPC):

กลไก Interprocess Communication (IPC) (คิวข้อความ semaphores และหน่วยความจำที่แบ่งใช้) เป็นเรื่องเกี่ยวกับข้อจำกัด DAC, MIC และ MAC โดยปกติ ไม่มีคำสั่งสำหรับการสร้างและการใช้อ็อบเจ็กต์ System V IPC

การเรียกของระบบ AIX IPC-related ได้ถูกแก้ไขให้เป็น multilevel-aware สำหรับ Trusted AIX การเรียกระบบที่แก้ไขนี้มี:

- **msgget**
- **msgsnd**
- **msgrcv**
- **msgctl**
- **semget**
- **semop**
- **semctl**
- **shmget**
- **shmctl**
- **shmat**
- **shmdt**

นอกจากนี้ การเรียกระบบดังต่อไปนี้ออกแบบเป็นพิเศษเพื่อจัดการ แอ็ตทริบิวต์ MAC ของอ็อบเจ็กต์ IPC ได้ถูกเพิ่มให้กับTrusted AIX:

sec_getmsgsec
รับแอ็ตทริบิวต์การรักษาความปลอดภัยของคิวข้อความ

sec_getsemsec
รับแอ็ตทริบิวต์การรักษาความปลอดภัยของ semaphores

sec_getshmsec
รับแอ็ตทริบิวต์การรักษาความปลอดภัยของเซ็กเมนต์หน่วยความจำที่แบ่งใช้

sec_setmsglab
เซ็ตแอ็ตทริบิวต์การรักษาความปลอดภัยของคิวข้อความ

sec_setsemlab
เซ็ตแอ็ตทริบิวต์การรักษาความปลอดภัยของ semaphores

sec_setshmlab
เซ็ตแอ็ตทริบิวต์การรักษาความปลอดภัยของเซ็กเมนต์หน่วยความจำที่แบ่งใช้

ดูที่ การเข้าถึง อ็อบเจ็กต์ IPC สำหรับข้อกำหนด privilege สำหรับกระบวนการ ในการจัดการอ็อบเจ็กต์ IPC คำสั่ง **settxattr** สามารถถูกใช้ เพื่อจัดการแอ็ตทริบิวต์ IPC

การสร้างเลเบล high และ system high ของ MIC และ MAC:

บ่อยครั้งที่มีความจำเป็นสำหรับกระบวนการที่ไว้วางใจในการระบุเลเบล MAC ที่ควบคุมเลเบลอื่นทั้งหมดในระบบ มีเลเบล MAC ต่างกันสองเลเบลที่สามารถใช้ได้ การใช้เลเบล high MAC หรือเลเบล system high MAC

การใช้เลเบล high MAC คือเลเบล MAC ที่สูงสุดที่สนับสนุนโดย Trusted AIX เป็นไปได้ที่เลเบลนี้ มีการจัดประเภทลำดับชั้นและมีหมวดหมู่ที่ไม่ถูก ใช้สำหรับไซต์ เลเบลนี้ถูกสร้างได้ง่าย แต่เลเบลต้องใช้ ด้วยความระมัดระวัง ไม่มีกระบวนการใดควรสร้างอ็อบเจ็กต์ที่เลเบลนี้

เลเบล system high MAC เป็นเลเบล MAC สูงที่สุดที่ถูกใช้สำหรับ ไซต์ ถูกกำหนดโดยผู้ดูแลระบบในไฟล์ **LabelEncodings**

การใช้เลเบล system high MAC มีประสิทธิภาพน้อยกว่า แต่ขอแนะนำให้ใช้ เนื่องจากผู้ดูแลระบบสามารถจำกัดการดำเนินการได้อย่างมีประสิทธิภาพแม้แต่กระบวนการที่มี privilege โดยการตั้งค่าพารามิเตอร์ที่เหมาะสมอย่างถูกต้องในไฟล์ **LabelEncodings**

MIC มีการสร้างเลเบล high และ system high เหมือนกัน

ขอบเขตการล็อกอินของ ผู้ใช้และระบบ:

โปรแกรมที่ไว้วางใจที่ดำเนินเซอร์วิสสำหรับผู้ใช้อาจจำเป็นต้องจำกัดเลเบล MIC และ MAC ที่เกี่ยวข้องในการดำเนินการเหล่านี้เป็นค่าซึ่งผู้ใช้ ได้รับอนุญาตให้ล็อกอิน และ/หรือ กับทั้งระบบที่อนุญาตล็อกอินเลเบล

clearances ที่ถูกกำหนดให้ดับผู้ใช้บนระบบอยู่ในไฟล์ฐานข้อมูล **user** `/etc/security/user` และเข้าถึงโดยใช้ไลบรารีรูทีน **getuserattr** and **getuserattrs**

Trusted AIX อนุญาตให้ผู้ใช้ ทำงานบนระบบในทุกระดับที่แสดงในขอบเขตการแต่งตั้งของระบบ และที่ถูกควบคุมโดย clearance สูงสุดของผู้ใช้และที่ควบคุม clearance ต่ำสุดของผู้ใช้ โปรแกรมทั้งหมดที่อนุญาตให้ผู้ใช้ทำงานที่เลเบลต่างกัน ควรตรวจสอบเสมอว่าเลเบลใหม่ถูกต้องสำหรับ ผู้ใช้

ตัวอย่างเช่น สมมุติว่ายูทิลิตีชื่อ **upgrade** ถูกกำหนดให้เพิ่ม เลเบล MAC ในไฟล์ตามการร้องขอของผู้ใช้ ข้อจำกัด MAC พื้นฐาน ต้องการให้ **upgrade** ยอมรับเฉพาะไฟล์ซึ่งเลเบล MAC ถูกควบคุม โดยผู้ใช้นั้น นอกจากนี้ เป็นการรอบคอบ (แม้ว่าไม่จำเป็นต้องเคร่งครัด ตามข้อจำกัด MAC พื้นฐาน) ที่เลเบลใหม่เป็นเลเบลที่ ผู้ใช้ได้รับอนุญาตให้ล็อกอิน ซึ่งรวมถึงข้อจำกัดขอบเขตเลเบลทั้งแบบ ต่อผู้ใช้และ ทั้งระบบ ยูทิลิตี **upgrade** จะใช้ทั้งอินเตอร์เฟส **sl_cmp** และ **accredrange** สำหรับจุดประสงค์นี้

โครงสร้างแผนผังไดเร็กทอรี:

ระบบเรียกฟังก์ชันเพื่อที่แผนผังไดเร็กทอรีที่สร้างโดยกระบวนการ unprivileged เป็นไปตามโครงสร้างเลเบลที่ไม่มีการเพิ่มโดยที่เลเบลของไฟล์ เท่ากับไดเร็กทอรีพาเรนท์ หรืออยู่ภายในช่วงของไดเร็กทอรีที่พาร์ติชัน และเลเบลของการควบคุมไดเร็กทอรีของไดเร็กทอรรีพาเรนท์ (หมายเหตุ การควบคุมรวมความเท่ากัน) นี่เป็นโครงสร้างปกติสำหรับ โปรแกรมที่ไม่ไว้วางใจ

อย่างไรก็ตามกระบวนการ privileged ไม่ได้ถูกเชื่อมโยงโดยข้อจำกัดและสามารถสร้างแผนผัง ไดเร็กทอรี ซึ่งความสัมพันธ์เลเบล MAC ไดเร็กทอรีพาเรนท์ ไม่แน่นอน คอนฟิกูเรชันนี้มีประโยชน์เนื่องจากการค้นหาการเข้าถึง MAC ถูกจำกัดใกล้ขึ้นกับ root ของแผนผัง ตัวอย่างเช่น การปกป้อง aggregation ซึ่งเลเบล MAC ของคอลเล็กชันอ็อบเจ็กต์ข้อมูลสูงกว่าเลเบลของอ็อบเจ็กต์ สามารถถูกสร้างโดยการตั้งค่าเลเบล MAC ของไดเร็กทอรี สูงกว่าองค์ประกอบ จากนั้นกระบวนการที่ไม่ไว้วางใจต้องควบคุมเลเบล ของไดเร็กทอรีเพื่อรับการเข้าถึง aggregation ของข้อมูล

การสร้างแผนผังไดเร็กทอรีที่มีการลดเลเบลต้องทำ อย่างระมัดระวัง เป็นไปไม่ได้สำหรับกระบวนการ unprivileged ที่จะเปิดไฟล์เพื่อเขียน เมื่อไฟล์ไม่ได้ควบคุมหรือเท่าเทียมกับเลเบลของพาเรนท์

การจัดการไดเร็กทอรีที่พาร์ติชัน:

มีการเรียกระบบที่มีการทำงานต่างกัน จาก ผลของการนำไดเร็กทอรีที่ไดเร็กทอรีมาใช้

การเรียกระบบดังต่อไปนี้ทำงานต่างกันจากผลของการนำไดเร็กทอรีที่ ไดเร็กทอรีมาใช้:

- getdirents
- link
- mkdir
- mount
- rename
- rmdir
- stat
- lstat
- fstat

โหมดกระบวนการ:

คำสั่ง **pdmode** สามารถปฏิบัติคำสั่งตามโหมดที่ระบุ กระบวนการสามารถใช้ **setppdmode** (การเรียกระบบ) เพื่อตั้งค่าโหมดของตัวเองเป็นโหมดจริงหรือโหมดเสมือน การเรียกระบบ **setppdmode** ต้องการ **PV_PROC_PDMODE** privilege ไม่มีกลไกสำหรับกระบวนการในการเปลี่ยนโหมดของกระบวนการอื่น

ชนิดไดเร็กทอรี:

คำสั่ง **pdset** สามารถถูกใช้เพื่อเปลี่ยนไดเร็กทอรีปกติ ลงในไดเร็กทอรีที่พาร์ติชัน แต่ไม่มีคำสั่งในการเปลี่ยน ไดเร็กทอรีที่พาร์ติชัน (หรือไดเร็กทอรีย่อยหรือ sub-subdirectory ที่พาร์ติชัน) ไปเป็นไดเร็กทอรีปกติ

การเรียกระบบ **pdmkdir** สามารถถูกใช้เพื่อสร้างไดเร็กทอรีที่พาร์ติชันได้เช่นกัน การเรียกระบบ **pdmkdir** ต้องการ **PV_FS_PDMODE** privilege

ข้อควรพิจารณาเลเบล MIC และ MAC:

โปรแกรมทั้งหมดควรใช้เฉพาะฟังก์ชัน **sl_cmp** and **tl_cmp** ในการกำหนดความสัมพันธ์ระหว่างเลเบล MIC และ MAC

เรื่องนี้สำคัญมากเนื่องจากรูปแบบเลเบลภายในสามารถเปลี่ยนแปลง ตามเวอร์ชันระบบภายหลัง และไลบรารีรูทีนเหล่านี้ติดตามการเปลี่ยนแปลงรูปแบบ เช่นเดียวกัน มีไลบรารีรูทีนอื่นที่จัดการเลเบล MIC และ MAC ที่ควรถูกใช้เมื่อเป็นไปได้

การเรียกระบบ **setea, lsetea** และ **fsetea** เปลี่ยน เลเบล MIC หรือ MAC ของไฟล์ การเรียกระบบ **fsetea** ยอมรับไฟล์ descriptor

ไดรเวอร์อุปกรณ์:

มีหลักการและคำแนะนำที่ควรปฏิบัติตาม เมื่อสร้างไดรเวอร์อุปกรณ์สำหรับระบบ Trusted AIX คุณควรคุ้นเคยกับกลไกสำหรับการสร้างไดรเวอร์อุปกรณ์สำหรับ ระบบฐานและควรระมัดระวังในการใช้กลไลเหล่านี้

ระบบย่อยการจัดการอุปกรณ์:

อุปกรณ์ในระบบ AIX เป็น abstraction และถูกใช้เพื่อครอบคลุมอ็อบเจ็กต์ข้อมูลทั้งหมดที่เข้าถึงโดยการอ้างอิง ไฟล์พิเศษ อุปกรณ์ ในบางกรณี อ็อบเจ็กต์ข้อมูลเหล่านี้แสดงอุปกรณ์ฟิสิคัลจริง และในบางกรณีก็ต่างออกไป (รวมถึงกรณีเช่น `/dev/null` ซึ่ง ไม่มีอ็อบเจ็กต์หน่วยเก็บข้อมูลเลย) ตัวอย่างหลังบ่อยครั้งเรียกว่า pseudo-devices

ระบบ Trusted AIX จัดเตรียม อุปกรณ์สองชนิด: อุปกรณ์เลเบลเดี่ยวและอุปกรณ์หลายระดับ อุปกรณ์หลายระดับ เชื่อถือข้อมูลกระบวนการมากกว่าหนึ่งระดับความลับต่อครั้ง อุปกรณ์ เลเบลเดี่ยวโดยปกติไม่ได้รับการไว้วางใจ เลเบลบนข้อมูลโดยปกติ ถูกเชื่อมโยงกับข้อมูลที่อุปกรณ์หลายระดับจัดการใน แบบที่ประกันว่าข้อมูลจะถูกเลเบลถูกต้องเสมอ อุปกรณ์เลเบลเดี่ยว โดยปกติขึ้นกับการเลเบลภายนอก

ฮาร์ดดิสก์เป็นตัวอย่างของอุปกรณ์หลายระดับ ข้อมูลทั้งหมดที่ถูกเก็บ ไว้ที่ฮาร์ดดิสก์ถูกเชื่อมโยงเลเบลระดับความลับ พรินเตอร์มีที่ตั้ง ทางกายภาพในสภาวะแวดล้อมซึ่งต้องการ clearance ความปลอดภัยในการเข้าถึง เป็นตัวอย่างของอุปกรณ์เลเบลเดี่ยว เฉพาะข้อมูลที่ clearance นั้นสามารถถูกส่ง ไปที่พรินเตอร์

ข้อควรระวังในการพัฒนาไดรเวอร์อุปกรณ์:

ไดรเวอร์อุปกรณ์เป็นส่วนหนึ่งของเคอร์เนลระบบปฏิบัติการดังนั้น ไม่ถูกจำกัดการดำเนินการ การสร้างหรือการแก้ไขไดรเวอร์อุปกรณ์ มีความสำคัญเหมือนการแก้ไขตัวเคอร์เนลเอง โชคไม่ดี ผู้ใช้มักจำเป็นต้องสร้างหรือแก้ไขไดรเวอร์อุปกรณ์ซึ่งควรกระทำ ด้วยความระมัดระวังอย่างสูง

เป็นไปไม่ได้ที่จะแสดงรายการของข้อควรระวังจำเพาะทั้งหมดที่จะถูกใช้เมื่อเขียน ไดรเวอร์อุปกรณ์ เนื่องมีหลายวิธีที่ไดรเวอร์ (บางครั้งไม่ได้ เจตนา) ทำให้การรักษาความปลอดภัยของระบบเสียหาย ดังนั้นการสร้าง ไดรเวอร์อุปกรณ์ที่ปลอดภัยขึ้นอยู่กับการตัดสินและประสบการณ์ของ ผู้ออกแบบ

ไดรเวอร์อุปกรณ์ไม่ควรดำเนินการอื่นใดนอกจากการจัดการอุปกรณ์ปกติ ไดรเวอร์อุปกรณ์ที่สร้างเฉพาะจุดประสงค์ในการเพิ่มการเรียกระบบใหม่ ให้กับระบบ รวมถึงไดรเวอร์ pseudo-device เช่นที่มีไว้สำหรับ `/dev/kmem` ควรถูกพิจารณาการเรียกระบบใหม่และถูกออกแบบตามลำดับ แนวทาง ในส่วนนี้อ้างอิงโดยหลักการถึงไดรเวอร์ที่เป็นไปตามตัวจัดการ อุปกรณ์

คุณควรเรียนรู้ไดรเวอร์อุปกรณ์มาตรฐานก่อนพยายามสร้างไดรเวอร์ ใหม่ การดำเนินการความปลอดภัยโดยหลักการของไดรเวอร์อุปกรณ์เกี่ยวข้องกับ การเรียกใช้การเรียกระบบ **open** และ **ioctl**

การเปิดอุปกรณ์:

กับอ็อบเจ็กต์ระบบส่วนใหญ่ การตรวจสอบความปลอดภัยส่วนใหญ่เชื่อมโยง กับการเข้าถึงอุปกรณ์ถูกดำเนินการ เมื่ออุปกรณ์ถูกเปิดด้วยการเรียกระบบ **open**

ขั้นแรกเคอร์เนลดำเนินปฏิบัติการพื้นฐานจากนั้นส่งการประมวลผล ของการร้องขอการเปิด ไปที่ไดรเวอร์อุปกรณ์ เคอร์เนลทำการตรวจสอบความปลอดภัย ดังต่อไปนี้ ก่อนการส่งผ่านการควบคุมไปที่ไดรเวอร์อุปกรณ์:

- ถ้ากระบวนการไม่มีการเข้าถึง MAC ให้กับไฟล์พิเศษอุปกรณ์ การเปิด ล้มเหลว
- ถ้ากระบวนการไม่มีการเข้าถึง MIC ให้กับไฟล์พิเศษอุปกรณ์ การเปิด ล้มเหลว
- ถ้ากระบวนการไม่มีการเข้าถึง DAC ให้กับไฟล์พิเศษอุปกรณ์ การเปิด ล้มเหลว

เมื่อมีอุปกรณ์จำนวนมาก การอ่านจากอุปกรณ์ (ด้วยการเรียกระบบ **read**) เปลี่ยนภาวะของอุปกรณ์ในแบบที่สามารถถูกตรวจพบโดยกระบวนการอื่น ซึ่งเลเบล MAC ไม่ได้ควบคุมกระบวนการอ่าน นี่ทำให้เกิด แชนเนลการแปลงได้ อุปกรณ์ที่เป็น first-in-first-out (FIFO) โดย ธรรมชาติทำให้เกิดปัญหานี้ ในกรณีเหล่านี้ เป็นแนวทางการปฏิบัติปกติ ที่จะจำกัดการเข้าถึงเพื่ออ่านกระบวนการ ที่เลเบล MAC เหมือนกับ อุปกรณ์ ทำได้โดยการตรวจสอบภายในไดรเวอร์อุปกรณ์

มีกฎหรือคำแนะนำสองสามประการสำหรับการออกแบบอุปกรณ์ พิเศษ คุณต้องเข้าใจและใช้หลักการพื้นฐานของข้อบังคับและ ค่าควบคุมการเข้าใช้โดยระมัดระวัง โชคดีที่ไดรเวอร์อุปกรณ์ส่วนใหญ่สามารถถูกตั้งค่า เป็นอุปกรณ์ธรรมดาและความผิดปกติของไดรเวอร์อุปกรณ์พิเศษไม่จำเป็น ต้องถูกจัดการบ่อยครั้ง

ตัวอย่างไดรเวอร์อุปกรณ์เปิด:

ต่อไปนี้เป็นตัวอย่างของการจัดการอุปกรณ์ที่ไม่ปกตินำมาจาก ไดรเวอร์อุปกรณ์ระบบมาตรฐาน มีจุดประสงค์เพื่อแสดงความหลากหลาย ที่เป็นไปได้ของไดรเวอร์อุปกรณ์ดังกล่าว

/dev/null

/dev/null เป็น pseudo-device ซึ่งไม่มีคอนเทนเนอร์ข้อมูล ข้อมูลที่เขียนไปที่ /dev/null ถูกละเว้น และส่งคืน end-of-file (EOF) เสมอให้กับการร้องขอการอ่านข้อมูล ดังนั้น ไม่จำเป็นต้องมีข้อจำกัดอุปกรณ์ MAC ในการเปิด เพื่อความเข้ากันได้ จำเป็นต้องใช้การเข้าถึง DAC บนไฟล์อุปกรณ์ /dev/null แม้ว่าไม่จำเป็น

/dev/tty

เมื่อกระบวนการส่งการเปิดบน /dev/tty ไดรเวอร์อุปกรณ์พยายามเปิดเทอร์มินัล ซึ่งคือการควบคุม เทอร์มินัลของกระบวนการที่ร้องขอ ดังนั้นต้องมีการตรวจสอบการเข้าถึง MIC, MAC, และ DAC สำหรับกระบวนการของเทอร์มินัลการควบคุมของกระบวนการ แทน /dev/tty เพื่อความเข้ากันได้ จำเป็นต้องมีการเข้าถึง DAC กับ /dev/tty แม้ว่าไม่จำเป็น

ข้อจำกัด ioctl:

ถึงแม้ฟังก์ชันอินเตอร์เฟสไดรเวอร์อุปกรณ์ทั้งหมดต้องได้รับการไว้วางใจ อินเตอร์เฟส **ioctl** โดยปกติต้องการการสนใจพิเศษ

กฎทั่วไป มีเพียงกระบวนการที่มีการเข้าถึงเพื่อเขียนเท่านั้นที่สามารถเปลี่ยน คุณสมบัติของไฟล์ที่สามารถถูกตรวจพบได้โดยกระบวนการอื่น ซึ่งไม่มีการเข้าถึงเพื่อเขียน การมีการเข้าถึงเพื่อเขียนหมายถึงกระบวนการมีไฟล์เปิดอยู่เพื่อเขียน หรือเลเบล MAC ของกระบวนการเท่ากับเลเบลของอุปกรณ์ ข้อจำกัดนี้เริ่มจากข้อจำกัด MAC พื้นฐานที่ไม่มีกระบวนการที่สามารถดำเนินปฏิบัติการที่สามารถถูกตรวจพบได้โดยกระบวนการที่มีเลเบล MAC ต่ำกว่า

ถ้าจุดประสงค์ของการดำเนินการคือการดำเนินการ อ่าน/เขียน ของผู้ใช้ ข้อจำกัดต้องถูกบังคับใช้ตามที่กำหนดไว้ มิฉะนั้นกรณีที่ข้อจำกัดไม่ได้ถูก บังคับใช้จะถือว่าเป็นแชนเนลการแปลง และควรถูกจำกัดแบนด์วิธ และ/หรือ ตรวจสอบได้

บางการดำเนินการควบคุมอุปกรณ์อาจจำเป็นต้องถูกจำกัดกับกระบวนการที่มี privilege แม้เมื่ออุปกรณ์ไม่ได้ถูกตั้งค่าเหมือนอุปกรณ์ที่ไว้วางใจ

ข้อบังคับอื่น:

มีกรณีอื่นน้อยมากที่ไดรเวอร์อุปกรณ์อาจจำเป็น ต้องมีการบังคับการตรวจสอบความปลอดภัยพิเศษ

ตัวอย่างหนึ่งคือเมื่อการอ่านข้อมูลบนอุปกรณ์เปลี่ยนสถานะของอุปกรณ์ใน แบบที่ถูกตรวจจับได้โดยกระบวนการซึ่งเลเบล MAC ไม่ได้ถูกควบคุม โดยกระบวนการอ่านนั้น นี่แสดงถึงแชนเนลการแปลงที่เป็นไปได้ ซึ่งอาจ จำเป็นต้องถูกจำกัดหรือตรวจสอบโดยตัวไดรเวอร์อุปกรณ์เอง

ข้อสรุปโปรแกรมมิงไดรเวอร์อุปกรณ์:

แนวทางดังต่อไปนี้ควรถูกนำมาพิจารณาเมื่อสร้าง ไดรเวอร์อุปกรณ์

หมายเหตุ: การเรียกระบบใหม่ได้ถูกเพิ่มเพื่อสนับสนุนความปลอดภัยที่ขยายสำหรับแต่ละ การอ่าน/เขียน บนอุปกรณ์ Streams และ FIFO สองไลบรารี API ใหม่ eread() และ ewrite() สนับสนุนแอ็ตทริบิวต์ความปลอดภัยที่ขยายนี้ ถ้าเป็นเคอร์เนล MLS แฟล็กความปลอดภัย DEV_SEC_ERDWR ถูกเซ็ตบนอุปกรณ์ เช่นเดียวกับ FIFO GNF_SEC_ERDWR ถูกเซ็ตบนอุปกรณ์ แฟล็กเหล่านี้เปิดใช้งานการตรวจสอบความปลอดภัย เพิ่มเติมในแต่ละการ อ่าน/เขียน

เทคนิคการออกแบบทั่วไป

การตรวจสอบความปลอดภัยทั้งหมดภายใน ไดรเวอร์อุปกรณ์ควรถูกเขียนในแบบมอดูลาร์และควรสามารถจำแนก ได้ง่าย

การตรวจสอบภายในไดรเวอร์อุปกรณ์

เป็นการดีกว่าเสมอ ที่จะตรวจสอบ MIC, MAC และ DAC ของไดรเวอร์อุปกรณ์ไดรเวอร์อุปกรณ์ที่ไม่มี การตรวจสอบดังกล่าว สามารถถูกพอร์ตได้ง่าย ไปที่หรือมาจาก ระบบที่ไม่ไว้วางใจหรือชนิด ของระบบที่ไว้วางใจอื่น

ในการสร้างไดรเวอร์อุปกรณ์ปกติ เคอร์เนล ทำการตรวจสอบ MIC, MAC และ DAC และไดรเวอร์ทำการตรวจสอบ privilege ที่จำเป็นเพิ่มเติม ในการสร้างไดรเวอร์อุปกรณ์แบบพิเศษ การตรวจสอบทั้งหมด (MIC, MAC, DAC และการตรวจสอบ privilege) ถูกดำเนินการในไดรเวอร์อุปกรณ์ ตัวเลือกในการสร้างไดรเวอร์อุปกรณ์ธรรมดาหรือพิเศษคือ เรื่องของการตัดสินใจในการออกแบบ

DAC

DAC ถูกบังคับใช้สำหรับแต่ละไฟล์พิเศษของอุปกรณ์ จาก entry point ของระบบไฟล์ที่ใช้ในการเข้าถึงอุปกรณ์

การตรวจสอบเพื่อแก้ไขการติดตั้ง

ไดรเวอร์อุปกรณ์ ที่ทำการตรวจสอบ MAC ควรจัดการ (ภายในขอบเขตของเหตุผล) ความเป็นไปได้ที่อุปกรณ์ถูกกำหนดอย่างไม่ถูกต้องได้อย่างปลอดภัย

Privilege การเข้าถึง

อาจเป็นการเหมาะสมสำหรับไดรเวอร์อุปกรณ์ ที่จะจำกัดการดำเนินการอุปกรณ์กับกระบวนการที่มี privilege อย่างไรก็ตาม มีข้อแนะนำบางข้อสำหรับสถานการณ์เหล่านี้

คุณสามารถ ใช้ฟังก์ชันเคอร์เนล **refmon** เพื่อกำหนดว่าคุณมี privilege ที่จำเป็นหรือไม่

privilege ขั้นต่ำ:

Trusted AIX แนะนำ แนวคิด privilege ขั้นต่ำ privilige ขั้นต่ำแยกผู้ใช้ root ที่เคยมีอำนาจมากมาเป็นกลไก privilge ที่มีการปรับแต่ง การแบ่ง privileges ประกันว่าถ้ามีข้อผิดพลาดด้านโปรแกรมมิงหรือข้อบกพร่องอื่นในซอฟต์แวร์ที่ไว้วางใจ จะมีความเสียหายกับการรักษาความปลอดภัยระบบน้อยมาก

การดำเนินการ Privilege:

มีสี่ privilege เวกเตอร์ที่เชื่อมโยงกับแต่ละกระบวนการ: effective, maximum, inheritable และ limiting

ค่า privilege เวกเตอร์สูงสุดกำหนดขีดจำกัดบนสำหรับ privileges ที่สามารถ แอ็คทีฟสำหรับแต่ละกระบวนการ effective privilege เวกเตอร์กำหนด privileges ที่ถูกตรวจสองเพื่อทำการการตัดสิน privilege หมายเหตุไว้ว่าชุด effective privilege เป็นเซ็ตย่อยของชุด maximum privilege เสมอ ซึ่งตามลำดับเป็นเซ็ตย่อย ของเซ็ต limiting privilege เสมอ ชุด limiting privilege กำหนด privileges ที่กระบวนการจะมีได้ในชุด maximum, inheritable และ effective privilege ชุด privilege ที่สืบทอดได้แสดงชุดของ privileges ที่ถูกสืบทอดโดยกระบวนการไชลด์ผ่าน forks และ execs

เมื่ออิมเมจข้อความใหม่ถูกเรียกใช้งาน การเพิ่ม privilege ถูกดำเนินการ จากอัลกอริทึมดังต่อไปนี้ privileges พิเศษที่กล่าวถึงคือ **PV_ROOT, PV_SU_, PV_SU_EMUL, PV_SU_ROOT, PV_AZ_ROOT** และ **PV_SU_UID**

อัลกอริทึมดังต่อไปนี้แสดงสองแนวคิดสำคัญเกี่ยวกับระบบย่อย privilege ที่สำคัญน้อยที่สุด แนวคิดแรกคือ privileges พิเศษ (**PV_ROOT, PV_SU_, PV_SU_EMUL, PV_SU_ROOT, PV_AZ_ROOT** และ **PV_SU_UID**) คือเฉพาะ privileges ที่ได้รับอนุญาตกับให้ถ่ายทอดแบบไม่เป็นไปตาม เงื่อนไขข้ามการเรียกใช้อิมเมจกระบวนการใหม่ แนวคิดที่สอง คือเวกเตอร์ effective privilege ของกระบวนการถูกเคลียร์ออกจาก privileges ทั้งหมด ยกเว้นไฟล์ที่เซ็ต **FSF_EPS** ซึ่งเป็นการประกันความเข้ากันได้ย้อนหลัง กับแอ็พพลิเคชันที่อาจจำเป็นต้องรันภายในระบบที่ไว้วางใจโดยไม่ต้องถูก จัดประเภทสำหรับระบบ privilege ที่สำคัญน้อยที่สุด

```
new_max_privs = old_inheritable_privs
new_max_privs = new_max_privs | file_innate_privs
IF (user was assigned some of authorizations in file PAS)
new_max_privs = new_max_privs | file_authorized_privs
new_max_privs = new_max_privs & old_limiting_privs
IF (old_max_privs contain one or more special privileges)
new_max_privs += same set of special privileges
IF (FSF_EPS is set for the executable)
new_eff_privs = new_max_privs
ELSE
new_eff_privs = old_inheritable_privs
IF (old_eff_privs contain one or more special privileges)
new_eff_privs += same set of special privileges
new_limiting_privs = old_limiting_privs
```

การกำหนดและการลบ privileges:

รูทีนไลบรารีระบบมาตรฐานดังต่อไปนี้แสดงวิธีที่ privileges ถูกจัดการบนระบบ รูทีนเหล่านี้มีประโยชน์เฉพาะกับโปรแกรมที่ privilege บนระบบ

priv_raise

เปลี่ยนเวกเตอร์ effective privilege ของกระบวนการโดยการเพิ่ม (หรือยกขึ้น) รายการที่ระบุของ privileges รายการของ privileges ต้องอยู่ในเวกเตอร์ privilege สูงสุดของกระบวนการหรือการแจ้งว่ามีข้อผิดพลาดถูกส่งกลับ

priv_remove

เปลี่ยนเวกเตอร์ effective และ maximum privilege ของกระบวนการโดย ลบรายการของ privileges ที่ระบุ ถ้ากระบวนการไม่สามารถลบ effective หรือ maximum privileges ออกการแจ้งว่ามีข้อผิดพลาดถูกส่งกลับ

priv_lower

เปลี่ยนเวกเตอร์ effective privilege ของกระบวนการโดยการลบ (หรือลดลง) รายการที่ระบุของ privileges ถ้ากระบวนการไม่สามารถลด effective privileges ออกการแจ้งว่ามีข้อผิดพลาดถูกส่งกลับ

แต่ละรูทีนเหล่านี้ยอมรับรายการที่คั่นด้วยคอมมาของ privileges ที่ปิดท้าย โดย **-1** (ลบหนึ่ง หมายเลข privilege ที่ไม่ถูกต้อง) เทคนิคสำหรับการเพิ่มหรือลด privileges กับส่วนของโค้ด ที่เล็กที่สุดที่อาจต้องการ privileges เหล่านี้เรียกว่า privilege bracketing แอ็พพลิเคชันที่ไว้วางใจทั้งหมดควรใช้ privilege bracketing เพื่อลดการละเมิด ความปลอดภัย โดยการออกแบบหรือการสร้างซอฟต์แวร์ที่ไม่ดี

setppriv

เปลี่ยนเวกเตอร์ effective, maximum, inheritable และ limiting privilege ของกระบวนการโดยตั้งค่าชุด privilege ถ้าชุด privilege ที่ส่งผ่านไม่ถูกต้อง หรือไม่ได้รับอนุญาต การแจ้งว่ามีข้อผิดพลาดถูกส่งกลับ

การอนุญาต:

การอนุญาตจัดเตรียมชุดต่างๆ ของ privileges ให้กับผู้ใช้ พร้อมกับการอนุญาต

โดยปกติ คำสั่งหรือยูทิลิตีตรวจสอบการอนุญาตที่เกี่ยวข้อง เมื่อเริ่มดำเนินการแล้วเซ็ต privileges ของตัวเองต่อมา ดังนั้น ผู้ใช้ที่มีการอนุญาตได้รับชุดของ privileges ต่างกันสำหรับแต่ละคำสั่งที่ดำเนินการ ขึ้นกับว่าคำสั่งถูกโปรแกรมอย่างไร

เมื่อต้องการค่าติดตั้ง privilege ที่เป็นปัญหาออกจากตัวโค้ดเอง AIX มี ชุดการอนุญาตและชุด privilege ภายนอกไปเป็นไบนารี ด้วย Privileged Authorization Set (PAS) และ Authorized Privilege Set (APS) ระบบไม่ใช่ตัว คำสั่งเอง ดำเนินการตั้งค่า privilege จากการอนุญาต

checkauths
เปรียบเทียบข้อมูลที่ส่งในรายการการอนุญาตกับการอนุญาตที่เกี่ยวข้อง กับกระบวนการปัจจุบัน

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการตรวจสอบการอนุญาต ดูที่ “การอนุญาต RBAC” ในหน้า 96

การตรวจสอบ:

Trusted AIX ประกอบด้วย ชุดของคำสั่งสำหรับการจัดการการสร้างและข้อมูลหลักฐานการตรวจสอบ เป็นสิ่งที่ไม่น่าจะเกิดขึ้น คือโปรแกรมเมอร์ระบบที่ไว้วางใจจะจำเป็นต้องแก้ไขหรือเพิ่มเติม โปรแกรมเหล่านี้

audit ควบคุม daemon การตรวจสอบ

auditbin
ควบคุมไฟล์หลักฐานการตรวจสอบ

auditselect
ผสมรวมและเลือกเร็กคอร์ดการตรวจสอบจากไฟล์หลักฐานการตรวจสอบ

auditpr แสดงเหตุการณ์การตรวจสอบที่เลือกในแบบที่อ่านได้

พื้นที่หลักที่การตรวจสอบที่เกี่ยวข้องกับโปรแกรมเมอร์ระบบที่ไว้วางใจ คือในเหตุการณ์การตรวจสอบที่ถูกสร้างโดยโปรแกรมที่ไว้วางใจ โปรแกรมที่ไว้วางใจ ส่วนใหญ่จำเป็นต้องส่งข้อความไปที่หลักฐานการตรวจสอบระบบ

สถานการณ์ในการการตรวจสอบ:

มีแนวทางที่ถูกต้องสองสามแนวทางสำหรับการระบุสถานการณ์ ที่ควรถูกตรวจพบและตรวจสอบโดยโปรแกรมที่ไว้วางใจ โดยหลักแล้วเป็นเรื่องของ การตัดสินและกลยุทธ์ในการตรวจสอบ ระบบฐานแบ่งสถานการณ์ออกเป็น successes, failures, object accesses และ possible covert channels

Successes:

เป็นสิ่งสำคัญที่จะตรวจสอบการดำเนินการสำเร็จเพื่อสร้างประวัติ การใช้พื้นฐาน

ตัวอย่างเช่น เป็นสิ่งสำคัญที่โปรแกรมการจัดสรรอุปกรณ์บันทึกเวลาที่ ผู้ใช้จัดสรรและคืนการจัดสรรอุปกรณ์ซึ่งอนุญาตให้โปรแกรม ติดตามโฟล์วของข้อมูลผ่านระบบและระบุหน้าที่ ถ้าอุปกรณ์ถูกระบุในภายหลังว่าถูกใช้ไม่ถูกต้อง ในอีกด้านหนึ่ง บางปรัชญาการตรวจสอบคำนึงถึงน้อยมากเกี่ยวกับความสำเร็จของการดำเนินการ เนื่องจากการดำเนินการดังกล่าวถูกพิจารณาเป็นถูกต้องและเหมาะสมโดยซอฟต์แวร์ ที่ไว้วางใจ

ความล้มเหลว:

การตรวจสอบการดำเนินการล้มเหลวมีประโยชน์ในการตรวจจับผู้ใช้ซึ่งพยายาม รับการเข้าถึงเซอร์วิสหรือข้อมูลที่ไม่ได้รับอนุญาต การเกิดขึ้นบ่อยครั้งของความล้มเหลวดังกล่าว อาจชี้ถึงผู้ที่ประสงค์ร้าย (ถ้าไม่ฉลาดเป็นพิเศษ)

ระบบฐานแบ่งความล้มเหลวออกเป็นห้าหมวดหมู่:

- ความล้มเหลว Privilege failures (an attempt by an unprivileged process to perform an action that is restricted to privileged processes)
- ความล้มเหลว MAC (ความล้มเหลวของการดำเนินการ เนื่องจากการดำเนินการจะละเมิดข้อบังคับ MAC)
- ความล้มเหลว MIC (ความล้มเหลวของการดำเนินการ เนื่องจากการดำเนินการจะละเมิดข้อบังคับ MIC)
- ความล้มเหลว DAC (ความล้มเหลวของการดำเนินการ เนื่องจากการดำเนินการจะละเมิดข้อบังคับ DAC)
- ความล้มเหลวอื่น (ตัวอย่างเช่น ความพยายามล็อกอินด้วยรหัสผ่านที่ไม่ถูกต้อง)

การเข้าถึงอ็อบเจ็กต์:

เป็นสิ่งจำเป็นในการตรวจสอบการเข้าอ็อบเจ็กต์เพื่อมอนิเตอร์ผู้ใช้ซึ่งเข้าถึง อ็อบเจ็กต์ที่กำหนด (ตัวอย่างเช่น ไฟล์รหัสผ่าน shadow)

แชนเนลการแปลงที่เกิดขึ้นได้:

การตรวจสอบแชนเนลการแปลงที่เกิดขึ้นได้มีความสำคัญ เนื่องจากแชนเนลการแปลง สามารถถูกใช้เพื่อส่งข้อมูลระหว่างกระบวนการที่เลเบล MAC ต่างกัน การใช้แชนเนลการแปลงที่เกิดขึ้นได้ ไม่ได้หมายความว่าแชนเนลเหล่านี้ ถูกใช้สำหรับจุดประสงค์นี้ เพียงว่าการใช้ดังกล่าวเป็นไปได้

แต่ละรายการที่เขียนโดยระบบการตรวจสอบ รวมถึงสาเหตุสำหรับรายการตรวจสอบ (success, MAC failure, MIC failure, DAC failure, privilege failure, other failure, object access หรือ potential covert channel) นี่รวมถึง ทั้งเร็กคอร์ดการตรวจสอบที่เขียนโดยตัวระบบเองและเร็กคอร์ดการตรวจสอบที่เขียน โดยโปรแกรมผู้ใช้

เป็นประโยชน์ในการพิจารณาว่าผู้ใช้ได้รับการไว้วางใจหรือไม่ (คือเป็น ผู้ดูแลระบบ) แต่ไม่มีวิธีการตายตัวในการระบุว่าผู้ใช้ที่ไว้วางใจหรือ ไม่ไว้วางใจต้องการการตรวจสอบมากกว่า ตัวอย่างเช่น แม้ว่าผู้ดูแลระบบ ถือว่าไว้วางใจ และด้วยเหตุนี้อาจต้องการการตรวจสอบน้อยกว่า การดำเนินการ สามารถตรวจสอบได้ยาก และมีจะประโยชน์ในการบันทึกการดำเนินการของผู้ดูแลระบบ ที่ไม่ได้รับอนุญาต ผู้ใช้ธรรมดาสามารถทำความเสียหายได้น้อยกว่า และด้วยเหตุนี้ ต้องการการตรวจสอบน้อยกว่า แต่ก็มีความน่าเชื่อถือน้อยกว่า และดังนั้นอาจจำเป็นต้อง การการตรวจสอบมากขึ้น ผู้ดูแลระบบบ่อยครั้งที่ใช้การตรวจสอบเพิ่มขึ้นกับการดำเนินการ เพื่อแสดงความบริสุทธิ์ใจ ในกรณีที่มีการละเมิดการรักษาความปลอดภัย

เหตุการณ์ดังต่อไปนี้ควรตรวจสอบได้:

- การดำเนินการที่สำเร็จ โดยเฉพาะที่เกี่ยวกับการถ่ายโอนข้อมูล หรือการเปลี่ยนพารามิเตอร์ค่าควบคุมการเข้าใช้
- การดำเนินการที่ล้มเหลวจากเหตุผลด้านความปลอดภัย

- การดำเนินการโดยผู้ดูแลระบบไม่ว่าสำเร็จหรือไม่
- การใช้แชนเนลการแปลงที่เกิดขึ้นได้
- การดำเนินการที่เข้าถึงอ็อบเจ็กต์จำเพาะ
- การดำเนินการที่กระทบเนื้อหาต่อมาของหลักฐานการตรวจสอบจริง

ระดับข้อมูลการตรวจสอบ:

ข้อมูลการตรวจสอบ High-level มีประโยชน์มากกว่าข้อมูลการตรวจสอบ low-level โปรแกรมที่ไว้วางใจดูแลมุมมองระดับสูงของการดำเนินการและ สามารถสร้างข้อความการตรวจสอบที่ดีเลิศได้

บันทึกเฉพาะผู้ดูแลระบบที่เปิดไฟล์ความปลอดภัยเพื่อการเขียน มีประโยชน์น้อยการการบันทึกการดำเนินการ higher-level จริง ที่ถูกดำเนินการบนไฟล์ (ตัวอย่างเช่น การบันทึกผู้ดูแลระบบได้สร้าง รายการใหม่ในไฟล์ รวมถึงข้อมูลหลักสำหรับรายการใหม่) ขอแนะนำเป็นอย่างสูงว่าข้อมูลการตรวจสอบอยู่ในระดับสูงเท่าที่ เป็นไปได้

เป็นการดีกว่าที่จะรวมข้อมูลเกี่ยวกับเหตุการณ์เดี่ยวแทนการรวม ข้อมูลเกี่ยวกับหลายๆ เหตุการณ์ เหตุผลหลักในการแยกการเกิดขึ้นของการตรวจสอบมากกว่าหนึ่งเหตุการณ์เพื่อที่การเกิดขึ้นแยกกัน สามารถถูกเลือกเพื่อเปิดใช้งานได้

คลาสและเหตุการณ์การตรวจสอบ:

แต่ละโปรแกรมที่ไว้วางใจต้องพิจารณาคลาสการตรวจสอบ ประเภทเหตุการณ์ การตรวจสอบ และเหตุผลที่ใช้เมื่อออกข้อความการตรวจสอบโดยใช้การเรียกใช้ระบบ **auditlog**

แต่ละเหตุการณ์การตรวจสอบเป็นสมาชิกอยู่ในคลาสการตรวจสอบ โดยการกำหนดเหตุการณ์ให้แก่คลาส คุณสามารถจัดการกับเหตุการณ์จำนวนมากได้อย่างมีประสิทธิภาพมากยิ่งขึ้น นิยามคลาสการตรวจสอบ ถูกกำหนดในไฟล์ `/etc/security/audit/config`

คลาสการตรวจสอบถูกใช้เพื่อเปิดใช้งานและปิดใช้งานการบันทึกเหตุการณ์ ถ้าเป็นเรื่องสำคัญสำหรับสองเหตุการณ์ที่ควรต้องถูกเปิดใช้งานแยกกัน เหตุการณ์เหล่านี้ ไม่ควรอยู่ในคลาสการตรวจสอบเดียวกัน อย่างไรก็ตาม โดยทั่วไปถือเป็นแนวทางปฏิบัติที่ดี ที่จะจัดกลุ่มเหตุการณ์เข้าเป็นคลาส โดยปกติ แต่ละโปรแกรมที่ไว้วางใจหรือชุดของ โปรแกรมที่ไว้วางใจที่เกี่ยวข้องจะสงวนชื่อคลาสการตรวจสอบไว้หนึ่งชื่อ (หรือในกรณีที่เป็นไปได้ยาก จะสงวนชื่อ คลาสการตรวจสอบสองสามชื่อ) สำหรับใช้งานของตน

การดำเนินการระบบที่สามารถตรวจสอบได้จะถูกกำหนดเป็นเหตุการณ์การตรวจสอบในไฟล์ `/etc/security/audit/events`

แปลงแชนเนล:

ซอฟต์แวร์ที่ไว้วางใจทั้งหมดถือว่าไม่อยู่ในแบบแผนของแชนเนล การแปลง นอกจากนี้ ซอฟต์แวร์ต้องถูกออกแบบเพื่อที่ไม่ให้ถูกใช้โดยซอฟต์แวร์ที่ไม่ไว้วางใจในการสร้างช่องโหว่แชนเนลการแปลง ส่วนนี้กำหนดแชนเนลการแปลง และให้แนวทางสำหรับการตรวจหาและการจำกัด

นิยามของแชนเนลการแปลง:

ไม่มีกระบวนการที่ระดับ A ควรจะสามารดำเนินการที่ตรวจพบได้โดยกระบวนการอื่นที่เลเบล B นอกจากเมื่อเลเบล B ควบคุม เลเบล A

นิยามนี้สามารถถูกแยกได้เป็นสองสถานการณ์: การดำเนินการข้อมูลโดยตรง และการดำเนินการจิปาถะ การดำเนินการข้อมูลโดยตรงมีไว้สำหรับผู้ใช้ เป็นวิธีตรงของการเก็บหรือการสื่อสารข้อมูลของผู้ใช้ เช่นการอ่านและ การเขียนไฟล์ การดำเนินการเหล่านี้ต้องยึดตามข้อกำหนด MAC ระดับต้นอย่างเคร่งครัด การดำเนินการอื่นทั้งหมดเป็นการดำเนินการจิปาถะ การใช้การดำเนินการจิปาถะ ในการส่งข้อมูลที่ขัดแย้งกับข้อจำกัด MAC ระดับต้นเรียกว่าแชนเนลการแปลง

การสร้างช่องโหว่ของแชนเนลการแปลงต้องการสองกระบวนการที่ไม่ไว้วางใจ ซึ่งจะถูกอ้างอิงกับผู้ส่ง (ที่เลเบล X) และผู้รับ (ที่เลเบล Y) จะถือว่าเลเบล MAC ของผู้รับไม่ได้ควบคุมผู้ส่ง (ถ้าควบคุม โฟล์วข้อมูลจากผู้ส่งไปที่ผู้รับจะเป็น การอัพเกรดอย่างถูกต้อง) เมื่อต้องการสร้างช่องโหว่แชนเนลนี้ ทั้งผู้ส่งและผู้รับ ใช้แบบแผนเกี่ยวกับการใช้ agreed-upon รีซอร์สเพื่อ รับส่งข้อมูลที่ขัดกับ MAC

เกณฑ์เดียวสำหรับการสร้างช่องโหว่ ในการแปลงคือเลเบลของผู้รับไม่ได้ควบคุมเลเบลของผู้ส่ง และทำให้ทั้งผู้ส่งและผู้รับไม่มีการไว้วางใจ ทั้งผู้ส่งและผู้รับจะทำงานในชื่อผู้ใช้ เดียวกัน ซึ่งจะถือว่าตัว TCB เองสนับสนุนข้อจำกัด MAC ระดับต้น และไมมีโค้ดที่ละเมิดข้อจำกัดนี้โดยการใช้แชนเนล การแปลงที่ประสงค์ร้าย (โดยข้อเท็จจริง กระบวนการที่มี privilege มีวิธีที่ได้ผลมากกว่า ในการละเมิด MAC โดยไม่ต้องพึ่งแชนเนลการแปลง) ซึ่งเป็นความสามารถ ของกระบวนการที่ไม่ไว้วางใจในการสร้างช่องโหว่แชนเนลการแปลงโดยการใช้โปรแกรม ที่ไว้วางใจ

โดยทั่วไป แชนเนลการแปลงควรขจัดออกจากระบบ อย่างไรก็ตาม มีบางกรณีที่เป็นความจำเป็นของระบบอื่น (ตัวอย่างเช่น ผลการทำงาน ความเชื่อถือได้ หรือความเข้ากันได้) ถูกสร้างข้อจำกัดที่ยอมรับไม่ได้ถ้าปราศจากแชนเนล การแปลง

คำแนะนำแบนด์วิธ:

ระบบฐานใช้คำแนะนำดังต่อไปนี้สำหรับการจำกัด แชนเนลการแปลงจากแบนด์วิธ:

มากกว่า 100 บิต/วินาที
แชนเนลเหล่านี้ไม่ได้รับอนุญาตให้มีอยู่

0.1 ถึง 100 บิต/วินาที
แชนเนลในขอบเขตนี้มีได้ เมื่อจำเป็นจริงๆ แต่การใช้งานู ถูกตรวจพบและตรวจสอบ ทุกครั้งที่เป็นไปได้

น้อยกว่า 0.1 บิต/วินาที
แชนเนลอยู่ในขอบเขตนี้มีได้เมื่อจำเป็น แต่ไม่มีความจำเป็นเป็นพิเศษ ที่จะต้องตรวจจับการใช้งาน

ขอแนะนำว่าโปรแกรม TCB เพิ่มเติมทั้งหมดให้ปฏิบัติตาม แนวทางเดียวกันนี้ นอกจากนี้ พิจารณาว่าแม้แชนเนลที่ช้า 10 บิตต่อวินาทีสามารถส่งข้อมูล 4,500 ไบต์ต่อชั่วโมง ซึ่งเป็นจำนวนข้อมูล ไม่น้อยที่จะถูกดาวน์เกรดอย่างไม่ถูกต้อง ดังนั้น ความพยายามทั้งหมดควร กระทำเพื่อจำกัดแชนเนลการแปลงให้มีแบนด์วิธต่ำเท่าที่จะเป็นไปได้

แบนด์วิธของแชนเนลการแปลงส่วนใหญ่โดยปกติถูกลดโดยกิจกรรม ของกระบวนการไม่ใช่กระบวนการที่อาจสร้างช่องโหว่กับแชนเนล อย่างไรก็ตาม ขอแนะนำไม่ควรใช้ผลดังกล่าวมาเป็นตัวกำหนดการจำกัดแบนด์วิธ แชนเนลการแปลง เนื่องจากมีระยะเวลาที่มีกิจกรรมต่ำบนระบบทั้งหมด

การตรวจหาแชนเนลการแปลง:

การตรวจหาแชนเนลการแปลงเป็นเรื่องของ การวิเคราะห์และออกแบบอย่างระมัดระวัง มีแนวทางจำเพาะสำหรับการตรวจหาแชนเนลการแปลง

คำว่าโมดูลหมายถึงหน่วยของโค้ด TCB ที่ตรวจพบหรือ จำกัดการใช้แชนเนลการแปลง ไม่ว่าในเคอร์เนลหรือในกระบวนการ การตรวจหาแชนเนลการแปลงโดยหลักแล้วเป็นเรื่องของการระบุว่า กระบวนการที่ไม่ไว้วางใจ (ผู้ส่ง) ที่ระดับ A สามารถใช้โมดูลในการ ดำเนินการที่ถูกตรวจพบได้โดยกระบวนการอื่น (ผู้รับ) ที่ระดับ B เมื่อระดับ B ไม่ได้ควบคุมระดับ A

ตัวอย่างเช่น แชนเนลการแปลงทั่วไปคือข้อมูลที่ถูกเขียนลงไฟล์โดยกระบวนการที่ไว้วางใจในนามของผู้ใช้ที่ไม่ไว้วางใจ เมื่อเลเบล MAC ของไฟล์ไม่ควบคุมเลเบล MAC ของผู้ใช้

มีหลักการไม่มากสำหรับตรวจหาแชนเนลการแปลงที่มีการ นำเสนอ ที่มีชื่อเสียงที่สุดคือ Shared Resource Matrix (SRM) อ้างถึงข้อมูลดังต่อไปนี้สำหรับคำอธิบายของเทคนิคนี้:

- Kemmerer, R.A. ″Shared Resource Matrix Methodology: An Approach to Identifying Storage and Timing Channels,″ ACM Transactions on Computing Systems 1(3) 1983, 256-277.
- • Tsai, CR. ″A Formal Method for the Identification of Covert Storage Channels in Source Code,″ Proceedings of the 1987 IEEE Symposium on Security and Privacy, 74-87.

การตรวจหาแชนเนลการแปลงผ่านการตรวจสอบ:

ความสามารถในการตรวจสอบการใช้แชนเนลการแปลงที่เกิดขึ้นได้สามารถ โต้ตอบการคุกคามนี้อย่างได้ผล อย่างไรก็ตาม เพื่อให้การตรวจสอบ มีประโยชน์ เหตุการณ์การตรวจสอบต้องมีน้อย การตรวจสอบถูกใช้น้อย ถ้าอัตราของการสร้างช่องโหว่จริงกับการใช้เหตุการณ์ที่ไม่เจตนา ที่ทำให้เกิดการตรวจสอบมีค่าน้อย

การจำกัดแชนเนลการแปลง:

วิธีที่ดีที่สุดในการจำกัดแชนเนลการแปลงคือให้ลบออก

มิฉะนั้น ควรถูกจำกัดตามแนวทางที่ได้กล่าวไว้ใน Bandwidth Guidelines นอกจากนี้ เมื่อใดที่เป็นไปได้และมีผลดี การใช้แชนเนลควรถูกตรวจสอบ

โดยทั่วไป เป็นเรื่องยากสำหรับเคอร์เนลหรือโค้ดไดรเวอร์อุปกรณ์ที่จะจำกัด แชนเนลการแผลง เนื่องจากเคอร์เนลและโค้ดไดรเวอร์อุปกรณ์ถูกออกแบบสำหรับ ความมีประสิทธิภาพและแชนเนลมีแบนด์วิธสูงกว่า กระบวนการที่ไว้วางใจสามารถจำกัดแชนเนลการแปลงได้ง่ายกว่า

หมายเหตุ: ไม่มีเหตุผลในการจำกัดแชนเนลการแปลงที่ใช้โดยกระบวนการที่เลเบลเดียวกัน หรือเมื่อตัวรับควบคุมตัวส่งข้อมูล ดังนั้น โมดูล TCB ส่วนใหญ่ สามารถเพิ่มผลการทำงานระบบโดยการกำหนดให้ไม่มีการจำกัดในกรณีเหล่านี้

โควต้า Per-label:

แชนเนลการแปลงจำนวนมากเกี่ยวข้องกับการใช้รีซอร์สพูลที่ถูกแบ่งใช้ ระหว่างกระบวนการที่เลเบล MAC ต่างกัน ซึ่งสามารถถูกจำกัดได้อย่างมีประสิทธิภาพ โดยการสร้างรีซอร์สพูลขนาดคงที่แยกสำหรับแต่ละเลเบล MAC เพื่อที่ กระบวนการจะสามารถ modulate การใช้รีซอร์สเท่านั้นจากพูลสำหรับเลเบล MAC

รีซอร์สที่ไม่ได้ใช้สามารถถูกเอาออกจากพูลหนึ่งไปที่อีกพูลหนึ่งเพื่อบรรลุความต้องการที่ไดนามิก การย้ายรีซอร์สนี้โดยตัวเองคือแชนเนลการแปลงแต่ ใช้แบนด์วิธต่ำกว่าซึ่งจำกัดได้ง่าย

การหน่วงเวลา:

หนึ่งเทคนิคสำหรับการจำกัดแชนเนลการแปลงคือสำหรับ TCB เพื่อประกัน ว่าเวลาที่ส่งเมื่อเซอร์วิสถูกดำเนินการ โดยที่มีแชนเนลอยู่ นี่สามารถเป็นเรื่องง่ายเหมือนการมีโมดูล sleep ตามเวลาที่ระบุ ซึ่งสามารถถูกคำนวณจากจำนวนข้อมูลที่ถูก ส่งผ่าน

อย่างไรก็ตาม นอกจากดำเนินการอย่างถูกต้อง การหน่วงเวลาบ่อยครั้งสามารถขัดขวางได้โดย โปรแกรมที่ทำให้เกิดแชนเนลการแปลง ตัวอย่างเช่น กระบวนการสร้างช่องโหว่ สามารถสร้างชุดของกระบวนการ ผู้รับ/ผู้ส่ง จำนวนมาก ขณะที่ TCB สามารถ จำกัดแต่ละชุดกับแบนด์วิธได้อย่างง่ายๆ โดยใช้เทคนิคการหน่วง การรวม กันของชุดทั้งหมดคือแบนด์วิธของแชนเนลเดี่ยวนี้

เป็นการดีกว่าสำหรับเซอร์วิส TCB เพื่อประกันว่าการหน่วงเวลาถูกใช้ในบางรูปแบบกับกระบวนการทั้งหมดที่อาจใช้เซอร์วิสอยู่

การหน่วงเวลามีประโยชน์สำหรับการจำกัด แต่มีแนวโน้มที่จะถูกโต้ตอบได้ง่าย โดยโปรแกรมประสงค์ร้ายและต้องถูกออกแบบอย่างระมัดระวัง

การจำกัดข้อมูล:

แบนด์วิธแชนเนลการแปลงถูกทำให้ลดลงได้ไม่เฉพาะโดยการเพิ่ม เวลา แต่โดยการลดจำนวนของข้อมูลที่ถูกส่งกลับ โปรแกรมที่ส่งกลับข้อมูลเป็นชุดของการดำเนินการสามารถส่งกลับ แพ็กเก็ตของข้อมูลที่น้อยกว่าหรือเล็กกว่า ภายในกรอบเวลาเดียวกัน

เวลาโดยประมาณ:

เทคนิคจำนวนมากสำหรับการสร้างช่องโหว่แชนเนลการแปลงต้อการ กระบวนการที่มีช่องโหว่เพื่อให้มีวิธีการที่แม่นยำในการวัดเวลาสัมพันธ์หรือ เวลาสัมบูรณ์ แชนเนลเหล่านี้บางครั้งสามารถถูกจำกัดโดยไม่อนุญาตให้กระบวนการ ระบุเวลาได้แม่นยำ

ขณะที่เป็นการง่ายในการประกันว่าเซอร์วิส TCB ที่ส่งกลับข้อมูลเวลา เป็นเวลาโดยประมาณ บางครั้งกระบวนการมีวิธีอื่นในการวันเวลาการส่ง ผ่าน เช่นการนับเวลาทำคำสั่งเครื่องของตัวเอง เทคนิคสำหรับการจำกัดแชนเนลดังกล่าวควรถูกใช้ด้วยความระมัดระวัง

Noisemakers:

แบนด์วิธของแชนเนลการแปลงส่วนใหญ่ถูกลดลง บางครั้งลดลงมา โดยกิจกรรมของกระบวนการไม่ใช่จากการทำให้แชนเนลมีช่องโหว่ เป็นไปได้ แม้ว่าจะไม่แนะนำ ในการสร้างโปรแกรมที่ไว้วางใจ ซึ่งมีจุดประสงค์เพื่อประกันว่าระดับที่แน่นอนของกิจกรรมมีอยู่เสมอ ซึ่งบางครั้งเรียกว่า noisemakers

ขณะที่การใช้ noisemakers อาจน่าสนใจในแนวคิด โดยปกติแล้ว เป็นเรื่องยากสำหรับ noisemakers ในการระบุเวลาที่ควรสร้างสัญญาณและ เมื่อใดที่ไม่ควร ดังนั้น นี่เป็นเทคนิคที่ไม่แนะนำสำหรับการจำกัด แชนเนลการแปลง

ลูกโซ่ U-T-U:

อาจมีสถานการณ์ซึ่งกระบวนการที่ไม่ไว้วางใจ **U1** ร้องขอ privilege กระบวนการที่ไว้วางใจ **T** ซึ่งจากนั้นร้องขอกระบวนการที่ไม่ไว้วางใจอื่น **U2** ซึ่งมีเลเบลต่างจาก **U1** **U1** และ **U2** แสดง กระบวนการที่ไม่ไว้วางใจที่เลเบล MAC ที่ต่างกันด้วยแชนเน

ลการแปลงพิเศษโดยความถูกต้องถูกต่อผ่านไปยังกระบวนการอื่น (จริงๆ แล้ว T และ U สามารถเป็น ลำดับของกระบวนการที่ไว้วางใจและ/หรือไม่ไว้วางใจ) เราเรียกสถานการณ์นี้ว่า ลูกโซ่ U-T-U

กระบวนการที่ไว้วางใจต้องประกันว่าข้อมูลไม่ถูกส่งผ่านระหว่างสองกระบวนการ ที่ไม่ไว้วางใจตามหลัก MAC พื้นฐาน ซึ่งรวมถึงทั้งการแยก การดำเนินการข้อมูลโดยตรงที่ไม่อนุญาต และแชนเนลการแปลง คุณควร พิจารณาข้อมูลดังต่อไปนี้:

- ไฟล์ descriptors ไม่สามารถถูกเปิดทิ้งไว้เมื่อ **U2** ไม่อาจเปิดไฟล์ ในโหมด read/write ซึ่งไฟล์ถูกเปิด
- ตัวแปรสภาวะแวดล้อมต้องถูกลบ ถ้าเลเบลของ **U2** ไม่ได้ ควบคุม U1
- ไดเร็กทอรีทำงานที่ผ่านจาก **U1** ไปที่ **U2** สามารถสร้าง แชนเนลการแปลง (อาจไม่มาก) ถ้าเลเบลของ **U2** ไม่ได้ควบคุม **U1** เช่นเดียวกัน พารามิเตอร์กระบวนการจำนวนมากที่สืบทอดโดยอัตโนมัติ โดยกระบวนการไชลด์สามารถสร้างแชนเนลการแปลง

เป็นไปได้สำหรับลูกโซ่ U-T-U ที่จะถูกจัดการอย่างถูกต้อง (นั่นคือ แชนเนลการแปลง สามารถถูกจำกัดอย่างมีประสิทธภาพ) อย่างไรก็ตาม เป็นเรื่องยากที่จะแน่ใจได้ และลูกโซ่ U-T-U ควรหลีกเลี่ยง หมายเหตุ อย่างไรก็ตามข้อกังวลที่ว่า U2 ไม่ได้การไว้วางใจ--อาจปลอดภัยที่จะกำหนดการไว้วางใจ แต่ไม่ได้รับ unprivilege

ตัวอย่างของการแปลงแชนเนล:

ต่อไปนี้เป็นตัวอย่างของการแปลงแชนเนลที่อาจมีอยู่ ในโมดูลที่สร้างโดยโปรแกรมเมอร์ระบบ

ตัวอย่างแชนเนลการแปลงเซอร์วิสการพิมพ์:

นี่เป็นตัวอย่างแชนเนลการแปลงเซอร์วิสการพิมพ์

เซอร์วิสพรินเตอร์รายบรรทัดที่ไว้วางใจ แท็กแต่ละงานที่ส่งอย่างถูกต้องด้วยเลเบล MAC ของกระบวนการที่ร้องขอและดูแลเลเบลนั้นด้วยงานที่คิว สำหรับใช้ในการพิพม์ตอนท้าย งานมีชื่อยาวได้

โปรแกรมสถานะอนุญาตให้ผู้ใช้ดูงานทั้งหมดที่ถูกคิวสำหรับผู้ใช้ รวมถึงชื่องานที่ผู้ใช้กำหนด ไม่ขึ้นกับเลเบล ของงาน ซึ่งสามารถถูกใช้เป็นแชนเนลการแปลง เนื้องจากกระบวนการผู้ส่ง สามารถสร้างงานซึ่งชื่อมีข้อมูลที่จะถูกส่งโดยมีการแปลงไปที่ผู้รับ ซึ่งทำงานกับผู้ใช้คนเดียวกัน

หมายเหตุ: เกณฑ์เดียวสำหรับการสร้างช่องโหว่ ในการแปลงคือเลเบลของผู้รับไม่ได้ควบคุมเลเบลของผู้ส่ง และทำให้ทั้งผู้ส่งและผู้รับไม่มีการไว้วางใจ ทั้งผู้ส่งและผู้รับ จะทำงานในชื่อของผู้ใช้เดียวกัน

แชนเนลนี้ถูกปิดโดยอนุญาตให้ผู้ใช้ดูงานที่ถูกควบคุมเท่านั้น โดยเลเบล MAC ปัจจุบันของผู้ใช้ ซึ่งบังคับเลเบล MAC ของตัวรับให้ควบคุมผู้ส่ง และแชนเนลสามารถถูกใช้ได้เฉพาะ การอัพเกรดถูกต้อง เป็นเรื่องของความสุภาพ โปรแกรมสถานะจะให้ข้อความ แก่ผู้ใช้ "other jobs exist" ถ้ามีงานที่ไม่ถูกควบคุมอยู่ นี่แสดงถึง แชนเนลที่เล็กกว่ามาก กับเหตุผลการดำเนินการที่ดีสำหรับการมีอยู่

หมายเหตุ: การตรวจสอบการตรวจหางานระดับสูงกว่าจะมีประโยชน์ เนื่องจากการตรวจหา นี้บางครั้งเกิดขึ้นได้ยากในการดำเนินการปกติ

นี่เป็นตัวอย่างปกติของแชนเนลการแปลง ซึ่งอ็อบเจ็กต์ข้อมูลที่มีชื่อหลายระดับ (งานพิมพ์ที่เข้าคิวอยู่ในกรณีนี้) เข้าถึงได้โดยกระบวนการ ที่เลเบล MAC ต่างกัน แชนเนลถูกเอาออกได้โดยการใช้เลเบล MAC ของอ็อบเจ็กต์กับชื่อ แอ็ตทริบิวต์ที่ไม่ใช้ชื่อ, เช่น ขนาด สามารถถือข้อมูลการแปลงได้เช่นกัน

ตัวอย่างรีซอร์สพูล:

เมื่อโปรแกรมที่ไว้วางใจดำเนินเซอร์วิสสำหรับไคลเอ็นต์ที่ไม่ไว้วางใจ โปรแกรมที่ไว้วางใจจัดสรรชนิดของรีซอร์สจำเพาะ (ตัวอย่างเช่นบัฟเฟอร์) จากพูลของรีซอร์สที่ถูกแบ่งใช้ระหว่างกระบวนการที่เลเบล MAC ต่างกัน

วิธีหนึ่งที่ใช้นี่เป็นแชนเนลการแปลงคือสำหรับผู้ส่งและผู้รับ ในการจัดเพื่อรับทั้งหมดนอกจากหนึ่งรีซอร์สที่จัดสรร อาจโดยโปรแกรมอื่น ที่รันอยู่เลเบล MAC ต่างกันหรือหลากหลาย หรือID ผู้ใช้ ที่ต่างกัน จากนั้นผู้ส่งทำให้รีซอร์สที่เหลืออยู่หนึ่งเดียวถูกจัดสรรหรือไม่ถูกจัดสรร และผู้รับตรวจพบโดยการพยายามจัดสรรรีซอร์ส เช่นกัน

นี่เป็นตัวอย่างคลาสสิคของรีซอร์สแชนเนลที่แบ่งใช้ ซึ่งสามารถถูกจำกัดได้ โดยการจัดสรรรีซอร์สพูลต่อเลเบลตามที่อธิบายด้านบน และยังสามารถ ตรวจพบได้โดยการตรวจสอบ

ตัวอย่างฐานข้อมูล:

ระบบฐานข้อมูลที่ไว้วางใจอนุญาตให้โปรแกรมผู้ใช้แทนที่ข้อมูลลงใน ฐานข้อมูลหลายระดับ การแอ็คเซสโดยตรงถูกควบคุมอย่างถูกต้องผ่านข้อจำกัด MAC ระดับต้น

อย่างไรก็ตาม เวลาที่จำเป็นในการนำรายการไปไว้ในฐานข้อมูลขึ้นกับ ขนาดรวมปัจจุบันของฐานข้อมูลเป็นอย่างมาก ดังนั้นผู้ส่ง สามารถเพิ่มหรือลบรายการเพื่อให้มีผลกับขนาดของฐานข้อมูล และผู้รับ เพียงวันเวลาที่ใช้เพื่อเพิ่มรายการเพื่อตรวจหาขนาดนี้ได้ แชนเนลนี้มีแบนด์วิธต่ำนอกจากการเข้าถึงฐานข้อมูลมีประสิทธิภาพ ดี

ช่วงเวลาการแอ็กเซสที่รับประกันสามารถถูกกำหนดในความพยายามจำกัด แชนเนล การหน่วงเวลาสามารถเป็น pseudorandom เพื่อที่เวลาที่เสียไปโดยเฉลี่ย ถูกลดลง อย่างไรก็ตาม นี่ยังคงเป็นรูปแบบการหน่วงเวลาและควรนำมาใช้อย่างระมัดระวัง

การตรวจสอบธรรมดาของการเข้าถึงทั้งหมดไม่มีประสิทธิภาพมากนัก เนื่องจาก เป็นการยากที่จะตรวจพบการสร้างช่องโหว่ของแชนเนลท่ามกลางการใช้ฐานข้อมูล จำนวนมากที่ไม่ได้เป็นการประสงค์ร้าย

ตัวอย่างโปรแกรมมิง:

ส่วนนี้จัดเตรียมตัวอย่างโปรแกรมมิงที่ไว้วางใจ

ตัวอย่างการตรวจสอบ privilege โปรแกรมที่ไว้วางใจ:

นี่เป็นมอดูลาร์รูทีนสำหรับโปรแกรมที่ไว้วางใจเพื่อตรวจสอบว่า กระบวนการที่เรียกมี privilege ที่เจาะจงหรือไม่

```
#include <sys/priv.h>
#include <sys/secattr.h>

int
priv_check (int priv)
{
  /* the process's security attributes */
  secattr_t secattr;

/* get the calling process's security attributes */
  if ( sec_getpsec(-1, &secattr;) != 0 )
  {
    return (-1);
```

```
 /* error retrieving the process's cred structure */
   }

   /*
    * return whether or not specified priv is in the
    * calling process's maximum privilege set
    */
   return privbit_test(secattr.sc_maxpriv, priv);
 }
```

ตัวอย่างการเปลี่ยนเลเบลระดับความลับ effective:

โปรแกรมนี้เปลี่ยนเลเบลระดับความลับ effective ของ กระบวนการปัจจุบันไปเป็น system high

privileges ดังต่อไปนี้จำเป็นในชุด privilege การสืบทอดของ โปรแกรม:

- **PV_LAB_LEF**
- **PV_LAB_SLUG**
- **PV_LAB_SL_SELF**

```
#include <stdio.h>
#include <mls/mls.h>
#include <unistd.h>
#include <sys/secattr.h>
#include <userpriv.h>
#include <sys/mac.h>
#include <sys/secconf.h>

#define SUCCESS 0
#define ERROR 1

int
main()
{
 sl_t sl_syshi; /* System high SL */
 secattr_t attr;
 char *clBuffer = NULL;

 /*
  * Get the system high and low SLs.
  */
 if ((sec_getsyslab(NULL, &sl_syshi, NULL, NULL)) != 0 ) {
  fprintf (stderr, "Call to sec_getsyslab failed.\n");
  exit(ERROR);
 }

 /*
  * Initialize this process with initlabeldb() to access the
  * system default Label database.
  */
 priv_raise(PV_LAB_LEF , -1);
 if (initlabeldb(NULL) != 0) {
  fprintf(stderr, "Could not read the Label Encodings Database.\n");
  exit(ERROR);
```

```
}
priv_remove(PV_LAB_LEF, -1);

/*
 * Get the process clearance range and effective SL.
 */
priv_raise(PV_LAB_SLUG, PV_LAB_SL_SELF, -1);
if (sec_getpsec(-1, &attr) != 0) {
 fprintf (stderr, "Problem getting Trusted AIX security attributes of program.\n");
 exit(ERROR);
}

/* malloc for the maximum SL label length that can be formed for process */
if((clBuffer = (char *) malloc(maxlen_cl())) == NULL) {
 perror("malloc");
 exit(ERROR);
}
/* Convert the binary effective SL to human readable */
if (clbtohr(clBuffer, &attr.sc_sl, HR_LONG) != 0) {
 fprintf(stderr, "Unable to convert SL to human readable form.\n");
 exit(ERROR);
}
printf("Program's intitial effective SL = %s.\n",clBuffer);

/*
 * Set the process effective SL to system high.
 * The process may not have its maximum SL at system high,
 * so set it also to system high.
 */
attr.sc_sl = sl_syshi;
attr.sc_sl_cl_max = sl_syshi;

if (sec_setplab(-1, &attr.sc_sl, NULL, &attr.sc_sl_cl_max,
  NULL, NULL, NULL) != 0) {
 fprintf (stderr, "Problem setting the effective SL of program.\n");
 exit(ERROR);
}

priv_lower(PV_LAB_SLUG, PV_LAB_SL_SELF, -1);

if (sec_getpsec(-1, &attr) != 0) {
 fprintf (stderr, "Problem getting Trusted AIX security attributes of program.\n");
 exit(ERROR);
}

/* Convert the binary effective SL to human readable */
if (clbtohr(clBuffer, &attr.sc_sl, HR_LONG) != 0) {
 fprintf(stderr, "Unable to convert to SL to human readable form.\n");
 exit(ERROR);
}
printf("Program's modified effective SL = %s.\n",clBuffer);
return(SUCCESS);
}
```

ตัวอย่างการตั้งค่าการจัดประเภทเลเบลระดับความลับและการเปรียบเทียบเลเบล ระดับความลับ:

นี่เป็นตัวอย่างของการตั้งค่าการจัดประเภทของเลเบลระดับความลับ และการใช้ไลบรารีรูทีนสำหรับการเปรียบเทียบระหว่างเลเบล ระดับความลับ

PV_LAB_LEF privilege จำเป็นในชุด privilege พร็อกซีของโปรแกรม และในการเรียกชุด privilege สูงสุดของกระบวนการ

```
#include <stdio.h>
#include <mls/mls.h>
#include <userpriv.h>
#include <errno.h>

#define SUCCESS 0
#define ERROR 1
int
main (int argc, char **argv)
{
/* Sensitivity labels */
sl_t sl1, sl2;

/* strings to hold labels' names */
char *slBuffer1 = NULL;
char *slBuffer2 = NULL;

if (argc != 3) {
fprintf(stderr, "Usage: compare slabel1 slabel2\n");
exit(ERROR);
}
/*
* Initialize this process with initlabeldb() to access the
* system default Label database.
*/
priv_raise(PV_LAB_LEF , -1);
if (initlabeldb(NULL) != 0) {
fprintf(stderr, "Could not read the Label Encodings Database.\n");
exit(ERROR);
}
priv_remove(PV_LAB_LEF, -1);

/* Convert the passed SL to binary format */
if (slhrtob(&sl1, argv[1]) != 0) {
fprintf(stderr, "Unable to convert %s to binary form.\n", argv[1]);
exit(ERROR);
}
if (slhrtob(&sl2, argv[2]) != 0) {
fprintf(stderr, "Unable to convert %s to binary form.\n", argv[2]);
exit(ERROR);
}

/* malloc for the maximum SL label length that can be formed */
slBuffer1 = (char *) malloc(maxlen_sl());
slBuffer2 = (char *) malloc(maxlen_sl());

if ((slBuffer1 == NULL) || (slBuffer2 == NULL)) {
perror("malloc");
```

```
exit(ERROR);
}

/*
* Translate the label back to human readable (long) form.
* This is not a necessary step. It is shown as an example
* usage of slbtohr() API.
*/
if (slbtohr(slBuffer1, &sl1, HR_LONG) != 0) {
fprintf(stderr, "Unable to convert to binary human readable form.\n");
exit(ERROR);
}

if (slbtohr(slBuffer2, &sl2, HR_LONG) != 0) {
fprintf(stderr,"Unable to convert to binary human readable form.\n");
exit(ERROR);
}

/*
* Use sl_cmp() to compare the dominance of the two labels.
*/
if (sl_cmp(&sl1, &sl2) == LAB_SAME) {
printf("label (%s) equals label (%s).\n",
slBuffer1, slBuffer2);
}
else if (sl_cmp(&sl1, &sl2) == LAB_DOM) {
printf("label (%s) dominates label (%s).\n",
slBuffer1, slBuffer2);
}
else if (sl_cmp(&sl2, &sl1) == LAB_DOM) {
printf("label (%s) dominates label (%s).\n",
slBuffer2, slBuffer1);
}
else {
printf("The two labels are disjoint.\n");
}

return (SUCCESS);
}
```

ตัวอย่างค่าติดตั้งข้อมูลการตรวจสอบ:

โปรแกรมนี้เรียกและเซ็ตข้อมูลการตรวจสอบ

privileges ดังต่อไปนี้จำเป็นในชุด privilege การสืบทอดของโปรแกรม:

- **PV_AU_ADMIN**
- **PV_DAC_GID**

```
#include <sys/types.h>
#include <sys/priv.h>
#include <sys/audit.h>

char buf[1024];
int main(int argc, char *argv[])
```

```
{
 int rc, len, p;
 /* *Get process audit preseclection mask */
 priv_raise(PV_AU_ADMIN, -1);
 rc = auditproc(0, AUDIT_QEVENTS, buf, sizeof (buf));
 priv_lower(PV_AU_ADMIN, -1);
 if (rc)
  fprintf(stderr, "Failed to get audit info\n");
        /* *Add the `kernel audit class to the preselection mask */
 p = 0;
 while ((len = strlen(&buf;[p])) > 0)
  p += len +1;
        strncat(&buf;[p], "kernel", (sizeof(buf)-p-1));
 p += strlen("kernel") + 2;
 buf[p] = 0;
 priv_raise(PV_AU_ADMIN, -1);
 rc = auditproc(0, AUDIT_EVENTS, buf, p);

 priv_lower(PV_AU_ADMIN, -1);
 if (rc)
  fprintf(stderr, "Failed to set audit info\n");
 /* *Set the GID of the process to generate an audit record */
 priv_raise(PV_DAC_GID, -1);
 rc = setgid(129);
 priv_lower(PV_DAC_GID, -1);
 if (rc)
  fprintf(stderr, "Failed to setgid\n");
 exit(0);
}
```

ตัวอย่างไคลเอ็นต์:

โปรแกรมนี้ส่งสองข้อความไปที่เซิร์ฟเวอร์โดยใช้ รูทีน write มาตรฐานและอีกวิธีใช้รูทีน ewrite

ข้อความที่ปลอดภัยถูกส่งที่ SECRET หมายเหตุว่าข้อความที่ไม่ปลอดภัย ส่งโดยใช้การเรียก write ถูกกำหนดชุดดีฟอลต์ของแอ็ตทริบิวต์ความปลอดภัย ซึ่งกำหนดค่าได้ผ่าน netrule

privileges **ดังต่อไปนี้จำเป็นในชุด** privilege **การสืบทอดของ โปรแกรม:**

- **PV_LAB_LEF**
- **PV_MAC_CL**
- **PV_LAB_SLUG_STR**

```
#include  <sys/mac.h>
#include  <sys/socket.h>
#include  <netinet/in.h>
#include  <sys/priv.h>
#include  <sys/secattr.h>
#include  <errno.h>
#include  <stdio.h>
#define SECURE 1
int
main(int  argc,  char  *argv[])
```

```
{
 int  sockfd;
 int  uid,  gid;
 char  buf[BUFSIZ];

 struct  sockaddr_in  serv_addr;

#ifdef  SECURE
 int  l_init_result  =  0;

 int  ewrite_result  =  0;

 sec_labels_t  seclab;

#endif  /*SECURE*/


 uid  =  getuid();

 gid  =  getgid();


 if  (  argc  !=  3  )

 {
  fprintf(stderr,  "Usage:%s:  ADDR  PORT\n",  argv[0]);

  exit(1);
 }
#ifdef  SECURE
 /*
  * *  Gain  access  to  the  Label  Encodings  Database
  *
  *     */


 priv_raise(PV_LAB_LEF,-1);
 l_init_result  =  initlabeldb(NULL);
 if  (  priv_remove(PV_LAB_LEF,  -1)  !=  0  )
 {
  fprintf(stderr,  "Privilege  Failure\n");
  exit(1);
 }
 if  (  l_init_result  !=  0  )
 {
  fprintf(stderr,  "Could  not  read  the  Label  Encodings  Database\n");
  exit(0);
 }
#endif  /*SECURE*/
  /*
   * *  Fill  in  the  structure  "serv_addr"  with  the  address
   * of
   * *  the  server  that  we  want  to  connect  with.
   * */
 memset  ((char  *)  &serv_addr;,  '\0',  sizeof(serv_addr));
```

```
 serv_addr.sin_family  =  AF_INET;
 serv_addr.sin_addr.s_addr  =  inet_addr(argv[1]);
 serv_addr.sin_port  =  htons(atoi(argv[2]));
 /*  Open  a  TCP  socket  (an  Internet  stream  socket).  */
 if  (  (sockfd  =  socket(AF_INET,  SOCK_STREAM,  0))  <  0)
 {
  perror("tcpclient:  ");
  fprintf(stderr,  "client:  Cant  open  stream  socket\n");
  exit(0);
 }
 if  (  connect(sockfd,  (struct  sockaddr  *)  &serv_addr;,
    sizeof(serv_addr))  <  0  )
 {
  perror("tcpclient:  ");
  fprintf(stderr,  "client:  Cant  connect  to  server\n");
  exit(0);
 }
 /*
  * *  Send  a  normal  write  to  the  server,  which  will  be
  * *  assigned  default  security  attributes
  * */
 strcpy(buf,  "This  has  the  default  security  attributes.\n");
 if  (  write(sockfd,  buf,  strlen(buf)+1)  ==  -1  )
 {
  perror("tcpclient:  ");
  fprintf(stderr,  "write  error\n");
 }
#ifdef  SECURE
  strcpy(buf,  "This  message  is  at  SECRET\n");
 /*  Set up  the  SL  and  CLs  */
 slhrtob(&seclab.sl;, "SECRET");
 slhrtob(&seclab.sl_cl_min;,  "SECRET");
 slhrtob(&seclab.sl_cl_max;,  "SECRET  A  B");
 seclab.sl.sl_format  =  STDSL_FORMAT;
 seclab.sl_cl_min.sl_format  =  STDSL_FORMAT;
 seclab.sl_cl_max.sl_format  =  STDSL_FORMAT;
 /*  This  ewrite  call  needs  PV_MAC_CL  and  PV_LAB_SLUG_STR  */
 priv_raise(PV_MAC_CL,PV_LAB_SLUG_STR,-1);
 ewrite_result  =  ewrite(sockfd,  buf,strlen(buf)+1,  &seclab;);
 priv_lower(PV_MAC_CL,PV_LAB_SLUG_STR,-1);

 if  (ewrite_result  ==  -1)
 {
  perror("tcpclient  call");
  fprintf(stderr,  "ewrite  error\n");
 }
 fflush(stderr);
#endif  /*SECURE*/
 fprintf(stderr,  "exiting  .....  \n");
 sleep(3);
 close(sockfd);
 exit(0);
}
```

ตัวอย่างเซิร์ฟเวอร์:

โปรแกรมนี้ทำงานเหมือนเซิร์ฟเวอร์และใช้รูทีน **eread** เพื่อรับ ข้อความที่ถูกส่งไปที่พอร์ต หลังจากการรับข้อความสมบูรณ์ โปรแกรมจะเอาต์พุตแอ็ตทริบิวต์ความปลอดภัยของข้อความ

privileges ดังต่อไปนี้จำเป็นในชุด privilege การสืบทอดของโปรแกรม (โดยไม่มีการกำหนด FSF_EPS secflags):

- **PV_LAB_LEF**
- **PV_MAC_CL**
- **PV_MAC_R_STR**

```
#include  <sys/mac.h>
#include  <sys/socket.h>
#include  <sys/priv.h>
#include  <sys/secattr.h>
#include  <sys/stropts.h>
#include  <netinet/in.h>
#include  <errno.h>
#include  <stropts.h>
#include  <unistd.h>
#include  <stdio.h>
#include  <mls/mls.h>
#define  MAX_HR_LABEL_LEN 2048
#define  SECURE 1
int
main(int  argc,  char  *argv[])
{
 pid_t  childpid;
 uint  clilen;
 int  sockfd,  newsockfd;
 struct  sockaddr_in  cli_addr,  serv_addr;


#ifdef  SECURE
 int  l_init_result;
 char  label_str[MAX_HR_LABEL_LEN];
 sec_labels_t  seclab;
#endif  /*  SECURE  */
 if  (  argc  !=  2  )
 {
  fprintf(stderr,  "Usage:%s  PORT\n",  argv[0]);
  exit(1);
 }
#ifdef  SECURE
 priv_raise(PV_LAB_LEF,  -1);
 l_init_result  =  initlabeldb(NULL);
 if  (priv_remove(PV_LAB_LEF,  -1)  !=  0)
 {
  fprintf(stderr,  "Privilege  Failure\n");
  exit(1);
 }

 if  (l_init_result  !=  0)
 {
```

```
 fprintf(stderr, "Could not read the Label Encodings Database\n");
 exit(1);
}
#endif /* SECURE */
/* Open a TCP socket (an Internet stream socket). */
if ( (sockfd = socket(AF_INET, SOCK_STREAM, 0)) < 0 )
{
 perror("tcpserver: ");
 fprintf(stderr, "server: Cant open stream socket\n");
 exit(1);
}
/*Bind our local address so that the client can send to us*/
memset((char *) &serv_addr;, '\0', sizeof(serv_addr));
serv_addr.sin_family = AF_INET;
serv_addr.sin_addr.s_addr = htonl(INADDR_ANY);
serv_addr.sin_port = htons(atoi(argv[1]));
if ( bind(sockfd, (struct sockaddr *) & serv_addr,
   sizeof(serv_addr)) < 0 )
{
 perror("tcpserver: ");
 fprintf(stderr, "server: Cant bind local address\n");
 exit(0);
}
listen(sockfd, 5);
for (;;)
{
 /*
  * * Wait for a connection from a client process.
  * */
 fprintf(stdout, "Waiting for a connection from a client\n");
 clilen = sizeof(cli_addr);
 newsockfd = eaccept(sockfd, (struct sockaddr *) &     cli_addr,
   &clilen;, &seclab;);
 if ( newsockfd < 0 )
 {
  perror("tcpserver: ");
  fprintf(stderr, "server: accept error\n");
 }
 /* Print SL */
 if ( slbtohr(label_str, &seclab.sl;, HR_SHORT) != 0 )
 {
  fprintf(stderr,"problem converting sl to string\n");
 }
 else
 {
  fprintf(stdout, "sl = %s.\n",label_str);
 }
 /* Print MIN CLEARANCE */
 if ( slbtohr(label_str, &seclab.sl_cl_min;, HR_SHORT) != 0 )
 {
  fprintf(stderr,"problem converting min clearance to string\n");
 }
 else
 {
  fprintf(stdout, "sl_cl_min = %s.\n",label_str);
```

```
  }

  /*  Print  MAX  CLEARANCE  */
  if  (  slbtohr(label_str,  &seclab.sl_cl_max;,  HR_SHORT)  !=  0  )
  {
   fprintf(stderr,"problem  converting  max  clearance  to  string\n");
  }
  else
  {
   fprintf(stdout,  "sl_cl_max  =  %s.\n",label_str);
  }
  if  (  (childpid  =  fork())  <  0  )
  {
   perror("tcpserver:  ");
   fprintf(stderr,  "server:  fork  error\n");
   exit(0);
  }
  else  if  (  childpid  ==  0  )  /*  child  process  */
  {
   int  i,  j;
   char  buf[BUFSIZ];
#ifdef  SECURE
  sec_labels_t  e_seclab;
#endif  /*  SECURE  */
  close(sockfd);
  for  (;;)
  {
      int  ret,  flag;
      struct  strbuf  ctstr,  dtstr;
      char  ctbuf[2048],  dtbuf[2048];
      ctstr.maxlen=2048;
      ctstr.buf  =  ctbuf;
      dtstr.maxlen=2048;
      dtstr.buf  =  dtbuf;
#ifdef  SECURE
      fprintf(stdout,  "Calling  eread\n");
      priv_raise(PV_MAC_CL,PV_MAC_R_STR,-1);
      ret  =  eread(newsockfd,  buf,  sizeof(buf),&e_seclab;);
      priv_lower(PV_MAC_CL,PV_MAC_R_STR,-1);
      if  (  ret  <  1  )
     {
         if  (  ret  ==  -1  )
   fprintf(stderr,  "eread  error\n");
        else
   fprintf(stderr,  "eread  no  data\n");
   close(newsockfd);
   exit(ret);
      }
      fprintf(stdout,  "\n%s",  buf);
      fprintf(stdout,  "\n");
      /*  Print  SL  */
      if  (  slbtohr(label_str,  &e_seclab.sl;,  HR_SHORT)  !=  0  )
      {
   fprintf(stderr,  "problem  converting  sl  to  string\n");
       }
```

```
      else
      {
   fprintf(stdout,  "sl  =  %s.\n",label_str);
       }
       /*  Print  MIN  CLEARANCE  */
       if  (  slbtohr(label_str,&e_seclab.sl_cl_min;,HR_SHORT)!=  0)
      {
   fprintf(stderr,"problem  converting  min  CL  to string\n");
       }
      else
      {
   fprintf(stdout,  "sl_cl_min  =  %s.\n",label_str);
       }
       /*  Print  MAX  CLEARANCE  */
      if  (  slbtohr(label_str,&e_seclab.sl_cl_max;,HR_SHORT)  !=0)
      {
   fprintf(stderr,"problem  converting  max  CL  to  string\n");
       }
       else
              {
   fprintf(stdout,  "sl_cl_max  =  %s.\n",label_str);
        }
        fflush(stdout);
#else  /*  NOT  SECURE  */
        fprintf(stdout,  "Calling  read\n");
        if  (read(newsockfd,  buf,  sizeof(buf))  <  1)
       {
   if  (ret  ==  -1)
      fprintf(stderr,  "read  error\n");
   else
      fprintf(stderr,  "read  no  data\n");
   close(newsockfd);
   exit(ret);
        }
        fprintf(stdout,  "%s\n",  buf);
        fflush(stdout);
#endif  /*  NOT  SECURE  */
              }
 }
 /*  parent  process  */
 close(newsockfd);
        }
}
```

แอ็ตทริบิวต์ผู้ใช้และการรักษาความปลอดภัยพอร์ต Trusted AIX:

แอ็ตทริบิวต์ผู้ใช้และการรักษาความปลอดภัยพอร์ตถูกใช้เพื่อเรียกข้อมูล แอ็ตทริบิวต์ clearance ของผู้ใช้และพอร์ต แล้วเปรียบเทียบ แอ็ตทริบิวต์ clearance ของผู้ใช้กับพอร์ต

แอ็ตทริบิวต์เพิ่มเติมต่อไปนี้ถูกนิยามในไฟล์ **usersec.h** สำหรับ Trusted AIX

S_MINSL

เลเบล clearance ระดับความลับต่ำสุดของผู้ใช้ Type SEC_CHAR

S_MAXSL

เลเบล clearance ระดับความลับสูงสุดของผู้ใช้ Type SEC_CHAR

S_DEFSL

เลเบลระดับความลับดีฟอลต์ของผู้ใช้ Type SEC_CHAR

S_MINTL

เลเบล integrity clearance ระดับความลับต่ำสุดของผู้ใช้ Type SEC_CHAR

S_MAXTL

เลเบล integrity clearance ระดับความลับสูงสุดของผู้ใช้ Type SEC_CHAR

S_DEFTL

เลเบล integrity ดีฟอลต์ของผู้ใช้ Type SEC_CHAR

แอ็ตทริบิวต์ดังต่อไปนี้ใช้ได้สำหรับพอร์ต

S_MINSL

เลเบลระดับความลับต่ำสุดที่กำหนดให้กับพอร์ต Type SEC_CHAR

S_MAXSL

เลเบลระดับความลับสูงสุดที่กำหนดให้กับพอร์ต Type SEC_CHAR

S_TL เลเบล Integrity ที่กำหนดให้กับพอร์ต Type SEC_CHAR

ตัวอย่างดังต่อไปนี้กำหนดว่าผู้ใช้สามารถล็อกอินบนพอร์ต ที่ระบุหรือไม่

```
#include <mls/mls.h>
#include <usersec.h>
#include <stdio.h>
#include <errno.h>

struct userlabels {
 sl_t minsl;
 sl_t maxsl;
 sl_t defsl;
 tl_t mintl;
 tl_t maxtl;
 tl_t deftl;
};

struct portlabels {
 sl_t minsl;
 sl_t maxsl;
 tl_t tl;
};

void getuserlabels(char * username, struct userlabels *usrlab);
void getportlabels (char * portname, struct portlabels *portlab);
void displayuseraccess (char * username, struct userlabels *usrlab,
     struct portlabels *portlab);

int
main (int argc, char **argv)
```

```
{

 struct userlabels usrlab;
 struct portlabels portlab;
 char *username = NULL;
 char *portname = NULL;


 if (argc != 3 ) {
  fprintf (stderr, "Usage: %s <username> <portname>\n", argv[0]);
  exit(1);
 }
 username = argv[1];
 portname = argv[2];

 initlabeldb(NULL);
 getuserlabels(username, &usrlab;);
 getportlabels(portname, &portlab;);
 displayuseraccess(username , &usrlab;, &portlab;);
 endlabeldb();
}

void getuserlabels(char *username,  struct userlabels *userlab)
{

 dbattr_t attributes[6];
 memset (attributes, 0, sizeof(attributes));

 attributes[0].attr_name = S_MINSL;
 attributes[0].attr_type = SEC_CHAR;

 attributes[1].attr_name = S_MAXSL;
 attributes[1].attr_type = SEC_CHAR;

 attributes[2].attr_name = S_DEFSL;
 attributes[2].attr_type = SEC_CHAR;

 attributes[3].attr_name = S_MINTL;
 attributes[3].attr_type = SEC_CHAR;

 attributes[4].attr_name = S_MAXTL;
 attributes[4].attr_type = SEC_CHAR;

 attributes[5].attr_name = S_DEFTL;
 attributes[5].attr_type = SEC_CHAR;

 if (getuserattrs(username, attributes, 6)) {
  fprintf(stderr,
    "Error retriving attributes for user %s\n", username);
  exit (1);
 }

 if (clhrtob (&(userlab->minsl), attributes[0].attr_char)) {
  fprintf(stderr, "minsl conversion error\n");
  exit (1);
```

```
 }

 if (clhrtob(&(userlab->maxsl), attributes[1].attr_char)) {
  fprintf(stderr, "maxsl conversion error\n");
  exit (1);
 }

 if (clhrtob(&(userlab->defsl), attributes[2].attr_char)) {
  fprintf(stderr, "defsl conversion error\n");
  exit (1);
 }

 if (tlhrtob(&(userlab->mintl), attributes[3].attr_char)) {
  fprintf(stderr, "mintl conversion error\n");
  exit (1);
 }

 if (tlhrtob(&(userlab->maxtl), attributes[4].attr_char)) {
  fprintf(stderr, "maxtl conversion error\n");
  exit (1);
 }

 if (tlhrtob(&(userlab->deftl), attributes[5].attr_char)) {
  fprintf(stderr, "deftl conversion error\n");
  exit (1);
 }

 printf("User %s has the following clearance values\n", username);
 printf("minsl:%s\n", attributes[0].attr_char);
 printf("maxsl:%s\n", attributes[1].attr_char);
 printf("defsl:%s\n", attributes[2].attr_char);
 printf("mintl:%s\n", attributes[3].attr_char);
 printf("maxtl:%s\n", attributes[4].attr_char);
 printf("deftl:%s\n", attributes[5].attr_char);

 return;
}

void getportlabels(char *portname, struct portlabels *portlab)
{
 int rc =0;
 char *val = NULL;
 if ( (rc = getportattr(portname,S_MINSL,(char*)&val;, SEC_CHAR)) != 0 ) {
  perror ("Error retrieving port attributes");
  exit(1);
 }

 if (slhrtob(&(portlab->minsl), val)) {
  fprintf(stderr, "port minsl conversion error\n");
  exit (1);
 }

 if ( (rc = getportattr(portname,S_MAXSL, (char*)&val;, SEC_CHAR)) != 0 ) {
  perror ("Error retrieving port attributes");
  exit(1);
```

```
  }

  if (slhrtob(&(portlab->maxsl), val)) {
   fprintf(stderr, "port maxsl conversion error\n");
   exit (1);
  }

  if ( (rc = getportattr(portname,S_TL, (char*)&val;, SEC_CHAR)) != 0 ) {
   perror ("Error retrieving port attributes");
  }

  if (tlhrtob(&(portlab->tl), val)) {
   fprintf(stderr, "port tl conversion error\n");
   exit (1);
  }

  return;
 }

 void displayuseraccess (char *username, struct userlabels *usrlab, struct portlabels *portlab)
 {
  CMP_RES_T cmpres;
  cmpres = sl_cmp(&(usrlab->defsl), &(portlab->minsl));
  if (cmpres != LAB_DOM && cmpres != LAB_SAME) {
   printf("Default SL of user does not dominate the minimum SL of tty \n");
   exit(1);
  }

  cmpres = sl_cmp(&(portlab->maxsl), &(usrlab->defsl));
  if (cmpres != LAB_DOM && cmpres != LAB_SAME) {
   printf("Default SL of user is not dominated by maximum SL of tty \n");
   exit(1);
  }

  cmpres = tl_cmp(&(portlab->tl), &(usrlab->deftl));
  if (cmpres != LAB_SAME) {
   printf("Default TL of user is not same as TL of tty \n");
   exit(1);
  }

  printf("The user can login on the specified port\n");
  return;

}
```

การเรียกระบบ Trusted AIX:

การเรียกระบบถูกจัดเตรียมเพื่อจัดการการทำงาน Trusted AIX เพิ่มเติม

eaccept
ยอมรับการเชื่อมต่อบนซ็อกเก็ต

ebind เชื่อมส่วนขยายเพื่อจัดการแอ็ตทริบิวต์ความปลอดภัย

econnect
เริ่มการเชื่อมต่อบนซ็อกเก็ตที่ขยายเพื่อจัดการแอ็ตทริบิวต์ ความปลอดภัย

eread อ่านข้อมูลจาก stream และเรียกข้อมูลแอ็ตทริบิวต์ความปลอดภัยของข้อความ

ereadv อ่านข้อมูลจาก stream และเรียกข้อมูลแอ็ตทริบิวต์ความปลอดภัยของข้อความ

erecv recv, recvfrom, recvmsg ที่ขยายเพื่อจัดการแอ็ตทริบิวต์ความปลอดภัย

erecvfrom
recv, recvfrom, recvmsg ที่ขยายเพื่อจัดการแอ็ตทริบิวต์ความปลอดภัย

erecvmsg
recv, recvfrom, recvmsg ที่ขยายเพื่อจัดการแอ็ตทริบิวต์ความปลอดภัย

esend send, sendto, sendmsg ที่ขยายเพื่อจัดการแอ็ตทริบิวต์ความปลอดภัย

esendmsg
send, sendto, sendmsg ที่ขยายเพื่อจัดการแอ็ตทริบิวต์ความปลอดภัย

esendto send, sendto, sendmsg ที่ขยายเพื่อจัดการแอ็ตทริบิวต์ความปลอดภัย

ewrite เขียนข้อมูลไปที่ stream และเซ็ตแอ็ตทริบิวต์ความปลอดภัยของข้อความ

ewritev เขียนข้อมูลไปที่ stream และเซ็ตแอ็ตทริบิวต์ความปลอดภัยของข้อความ

sec_getmsgsec
รับแอ็ตทริบิวต์การรักษาความปลอดภัยของคิวข้อความ

sec_getpsec
รับข้อมูลความปลอดภัยที่เชื่อมโยงกับกระบวนการ

sec_getrunmode
เรียกข้อมูลโหมดการดำเนินการของเคอร์เนล

sec_getsecconf
ส่งกลับแฟล็กคอนฟิกูเรชันความปลอดภัยปัจจุบัน

sec_getsemsec
รับแอ็ตทริบิวต์การรักษาความปลอดภัยของ semaphores

sec_getshmsec
รับแอ็ตทริบิวต์การรักษาความปลอดภัยของเซ็กเมนต์หน่วยความจำที่แบ่งใช้

sec_getsyslab
รับเลเบลระดับความลับระบบดีฟอลต์

sec_gettlibbufsize
เรียกข้อมูลรายการพาธไลบรารีในเคอร์เนล

sec_gettlibpath
เรียกข้อมูลรายการพาธไลบรารีในเคอร์เนล

pdmkdir
สร้าง/เซ็ต/ไม่เซ็น ไดเร็กทอรีหรือไดเร็กทอรีย่อยที่พาร์ติชัน

sec_setauditrange
เซ็ตขอบเขตเลเบลการตรวจสอบโกลบอลของระบบ

sec_setplab
เซ็ต effective sensitivity label, minimum sensitivity clearance, maximum sensitivity clearance, และ integrity label ของกระบวนการที่ระบุ

setppdmode
เซ็ตโหมดไดเร็กทอรีที่พาร์ติชัน (จริงหรือเสมือน) ของกระบวนการ

setppriv
เซ็ตชุด privilege ที่เชื่อมโยงกับกระบวนการ

sec_setptlibmode
เซ็ตโหมด TLIB ของกระบวนการ

sec_setrunmode
เซ็ตโหมดการดำเนินการของเคอร์เนล

sec_setsecconf
เซ็ตแฟล็กคอนฟิกูเรชันของความปลอดภัยเคอร์เนล

sec_setsemlab
เซ็ตแอ็ตทริบิวต์การรักษาความปลอดภัยของ semaphores

sec_setshmlab
เซ็ตแอ็ตทริบิวต์การรักษาความปลอดภัยของเซ็กเมนต์หน่วยความจำที่แบ่งใช้

sec_setsyslab
เซ็ต ระดับความลับระบบดีฟอลต์ ข้อมูล และเลเบล integrity

AIX C ไลบรารีฟังก์ชัน:

รูทีนย่อยและแมโครถูกจัดเตรียมเพื่อจัดการการทำงาน Trusted AIX เพิ่มเติม

accredrange
กำหนดว่าเลเบลระดับความลับอยู่ภายในขอบเขตการแต่งตั้งหรือไม่

clbtohr แปลงเลเบล clearance ไบนารีที่กำหนดเป็นรูปแบบที่อ่านได้

clhrtob แปลงเลเบล clearance ที่อ่านได้ที่กำหนดไปเป็นรูปแบบไบนารี

getfsfbitindex, getfsfbitstring
รูทีนเพื่อรับสตริงแฟล็ก File Security และดัชนี

getmax_sl, getmax_tl
เรียกข้อมูลเลเบล sensitivity และ integrity สูงสุดจากไฟล์ Label Encoding

getmin_sl, getmin_tl
เรียกข้อมูลเลเบล sensitivity และ integrity ต่ำสุดจากไฟล์ Label Encoding

getsecconfig, setsecconfig
รูทีนที่เรียกข้อมูลและเซ็ตแฟล็กคอนฟิกูเรชันความปลอดภัยเคอร์เนล สำหรับ runmodes

initlabeldb, endlabeldb

การเตรียมข้อมูล Label Database และจบการทำงานรูทีน

maxlen_sl, maxlen_cl, maxlen_tl

เรียกข้อมูลความยาวสูงสุดของเลเบลที่อ่านได้ในไฟล์ Label Encoding ที่เตรียมข้อมูลเบื้องต้น

priv_isnull

กำหนดว่ามี privileges ถูกเซ็ตในชุด privilege ที่กำหนดหรือไม่

priv_lower

การดำเนินการชุด Privilege

priv_raise

การดำเนินการชุด Privilege

priv_remove

การดำเนินการชุด Privilege

priv_subset

การดำเนินการชุด Privilege

privbit_clr

ลบ privilege ที่ระบุในชุด privilege ที่ระบุ

priv_clrall

ลบ privileges ทั้งหมดในชุด privilege ที่ระบุ

priv_comb

รวมสองชุด privilege ที่ระบุก่อน และนำผลลัพธ์ไปไว้ในชุด privilege ที่สามที่ระบุ

priv_copy

คัดลอกชุด privilege ที่ระบุชุดแรกไปไว้ที่ชุด privilege ที่สอง ที่ระบุ

priv_isnull

กำหนดว่าไม่มี privileges ถูกเซ็ตในชุด privilege ที่กำหนดหรือไม่

priv_mask

คำนวณการตัดกันของชุด privilege ที่ระบุสองชุดแรก และนำผลลัพธ์ไปไว้ในชุด privilege ที่ระบุชุดที่สาม

priv_rem

ลบ privileges ในชุด privilege ที่ระบุที่สองออกจาก ชุด privilege ที่ระบุชุดแรก และนำผลลัพธ์ไปไว้ในชุด privilege ที่ระบุ ชุดที่สาม

privbit_set

เซ็ต privilege ที่ระบุในชุด privilege ที่ระบุ

priv_setall

เซ็ต privileges ทั้งหมดในชุด privilege ที่ระบุ

priv_subset

กำหนดว่าชุด privilege ที่ระบุชุดแรกเป็นเซ็ตย่อยของ ชุด privilege ที่ระบุชุดที่สองหรือไม่

privbit_test
ทดสอบเพื่อดูว่า privilege ที่ระบุถูกเซ็ตในชุด privilege ที่ระบุ

slbtohr, clbtohr, tlbtohr
รูทีนการแปลงเลเบลไบนารีไปเป็นรูปแบบที่อ่านได้

slhrtob, clhrtob, tlhrtob
รูทีนการแปลงรูปแบบที่อ่านได้ไปเป็นเลเบลไบนารี

sl_clr, tl_clr
รูทีนที่รีเซ็ตเลเบล

sl_cmp, tl_cmp
รูทีนการเปรียบเทียบเลเบล

tl_cmp เปรียบเทียบเลเบล integrity

Trusted AIX privileges

privileges ดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

Audit privileges:

audit privileges ดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_AU_
เท่ากับ **PV_AU_** privileges อื่นทั้งหมดรวมกัน

PV_AU_ADD
อนุญาตให้กระบวนการทำการ บันทึก/เพิ่ม เร็กคอร์ดการตรวจสอบ

PV_AU_ADMIN
อนุญาตให้กระบวนการกำหนดค่าและเคียวรีระบบการตรวจสอบ

PV_AU_PROC
อนุญาตให้กระบวนการรับและเซ็ตสถานะของกระบวนการ

PV_AU_READ

อนุญาตให้กระบวนการอ่านไฟล์ที่ทำเครื่องหมายเป็นไฟล์การตรวจสอบ

PV_AU_WRITE

อนุญาตให้กระบวนการเขียนหรือลบไฟล์ที่ทำเครื่องหมายเป็นไฟล์การตรวจสอบ หรือ ทำเครื่องหมายไฟล์เป็นไฟล์การตรวจสอบ

privilege การอนุญาต:

privileges การอนุญาตดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_AZ_ADMIN

อนุญาตให้กระบวนการแก้ไขตารางความปลอดภัยเคอร์เนล

PV_AZ_READ

อนุญาตให้กระบวนการเรียกข้อมูลตารางความปลอดภัยเคอร์เนล

PV_AZ_ROOT

ทำให้กระบวนการผ่านการตรวจสอบการอนุญาตระหว่างการเรียกระบบ **exec**

PV_AZ_CHECK

อนุญาตให้กระบวนการผ่านการตรวจสอบการอนุญาตทั้งหมด

DAC privileges:

DAC privileges ดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_DAC_

เท่ากับ PV_DAC_ privileges อื่นทั้งหมดรวมกัน

PV_DAC_O

อนุญาตให้กระบวนการแทนที่ข้อจำกัดความเป็นเจ้าของ DAC

PV_DAC_R

อนุญาตให้กระบวนการแทนที่ข้อจำกัดการอ่าน DAC

PV_DAC_W
อนุญาตให้กระบวนการแทนที่ข้อจำกัดการเขียน DAC

PV_DAC_X
อนุญาตให้กระบวนการแทนที่ข้อจำกัดการเรียกใช้งาน DAC

PV_DAC_UID
อนุญาตให้กระบวนการเซ็ตหรือเปลี่ยน user ID (UID)

PV_DAC_GID
อนุญาตให้กระบวนการเซ็ตหรือเปลี่ยน group ID (GID)

PV_DAC_RID
อนุญาตให้กระบวนการเซ็ตหรือเปลี่ยน role ID (RID)

privileges ระบบไฟล์:

privileges ระบบไฟล์ดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_FS_
เท่ากับ **PV_FS_** privileges อื่นทั้งหมดรวมกัน

PV_FS_MKNOD
อนุญาตให้กระบวนการทำการเรียกระบบ **mknod** เพื่อสร้างไฟล์ ทุกชนิด

PV_FS_MOUNT
อนุญาตให้กระบวนการเมาท์และเลิกเมาท์ระบบไฟล์

PV_FS_CHOWN
อนุญาตให้กระบวนการเปลี่ยนความเป็นเจ้าของไฟล์

PV_FS_QUOTA
อนุญาตให้กระบวนการจัดการข้อมูลที่เกี่ยวกับโควต้าดิสก์

PV_FS_LINKDIR
อนุญาตให้กระบวนการสร้างฮาร์ดลิงก์ไปยังไดเร็กทอรี

PV_FS_RESIZE
อนุญาตให้กระบวนการดำเนินการประเภทการขยายหรือย่อบนระบบไฟล์

PV_FS_CNTL
อนุญาตให้กระบวนการดำเนินการควบคุมต่างๆ ยกเว้น การขยาย หรือการย่อระบบไฟล์

PV_FS_CHROOT
อนุญาตให้กระบวนการเปลี่ยนไดเร็กทอรี root

PV_FS_PDMODE

อนุญาตให้กระบวนการจัดทำหรือตั้งค่าไดเร็กทอรีประเภทพาร์ติชัน

privileges กระบวนการ:

privileges กระบวนการดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_PROC_

เท่ากับ **PV_PROC_** privileges อื่นทั้งหมดรวมกัน

PV_PROC_PRIO

อนุญาตให้กระบวนการ/เธรดเปลี่ยนระดับความสำคัญ, นโยบาย และ พารามิเตอร์การกำหนดการอื่นๆ

PV_PROC_CORE

อนุญาตให้กระบวนการดัมพ์ข้อมูลคอร์

PV_PROC_RAC

อนุญาตให้กระบวนการสร้างกระบวนการมากกว่าที่จำกัดต่อหนึ่งผู้ใช้

PV_PROC_RSET

อนุญาตให้รวมชุดรีซอร์ส (**rset**) กับกระบวนการหรือ thread

PV_PROC_ENV

อนุญาตให้กระบวนการตั้งค่าข้อมูลผู้ใช้ในโครงสร้างผู้ใช้

PV_PROC_CKPT

อนุญาตให้กระบวนการกำหนดจุดตรวจสอบหรือรีสตาร์ทกระบวนการอื่น

PV_PROC_CRED

อนุญาตให้กระบวนการตั้งค่าแอ็ตทริบิวต์ credential ของกระบวนการ

PV_PROC_SIG

อนุญาตให้กระบวนการส่งสัญญาณไปยังกระบวนการที่ไม่เกี่ยวข้อง

PV_PROC_PRIV

อนุญาตให้กระบวนการแก้ไขหรือดูชุดสิทธิ์พิเศษที่สัมพันธ์ กับกระบวนการ

PV_PROC_TIMER

อนุญาตให้กระบวนการส่งและใช้ตัวจับเวลารายละเอียดย่อย

PV_PROC_RTCLK

อนุญาตให้กระบวนการเข้าถึงนาฬิกาที่เป็นเวลาของ CPU

PV_PROC_VARS

อนุญาตให้กระบวนการเรียกข้อมูลและอัพเดตพารามิเตอร์ที่เปลี่ยนได้ของกระบวนการ

PV_PROC_PDMODE
อนุญาตให้กระบวนการเปลี่ยนโหมด REAL ของไดเร็กทอรีที่ทำพาร์ติชัน

เคอร์เนล privileges:

เคอร์เนล privileges ดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_KER_
เท่ากับ **PV_KER_** privileges อื่นทั้งหมดรวมกัน

PV_KER_ACCT
อนุญาตให้กระบวนการสามารถดำเนินการที่จำกัดที่เกี่ยวกับ ระบบย่อยการจัดการบัญชีผู้ใช้

PV_KER_DR
อนุญาตให้กระบวนการเรียกใช้การดำเนินการตั้งค่าใหม่แบบไดนามิก

PV_KER_TIME
อนุญาตให้กระบวนการแก้ไขนาฬิการะบบและเวลา

PV_KER_RAC
อนุญาตให้กระบวนการใช้หน้าขนาดใหญ่ (ไม่สามารถจัดหน้า) สำหรับ เซ็กเมนต์หน่วยความจำที่แบ่งใช้

PV_KER_WLM
อนุญาตให้กระบวนการเตรียมข้อมูลเบื้องต้นและแก้ไขการตั้งค่า WLM

PV_KER_EWLM
อนุญาตให้กระบวนการเตรียมข้อมูลเบื้องต้นหรือเคียวรีสภาวะแวดล้อม eWLM

PV_KER_VARS
อนุญาตให้กระบวนการตรวจสอบหรือตั้งค่าพารามิเตอร์ที่เปลี่ยนได้ตอนรันไทม์ของ เคอร์เนล

PV_KER_REBOOT
อนุญาตให้กระบวนการปิดทำงานระบบ

PV_KER_RAS
อนุญาตให้กระบวนการตั้งค่าหรือเขียนเร็กคอร์ด RAS การ บันทึกข้อผิดพลาด การติดตาม และฟังก์ชัน dump

PV_KER_LVM
อนุญาตให้กระบวนการตั้งค่าระบบย่อย LVM

PV_KER_NFS
อนุญาตให้กระบวนการตั้งค่าระบบย่อย NFS

PV_KER_VMM
อนุญาตให้กระบวนการแก้ไขกระบวนการการสสลับค่าและพารามิเตอร์ที่เปลี่ยนได้ VMM อื่นๆ ในเคอร์เนล

PV_KER_WPAR
อนุญาตให้กระบวนการตั้งค่าเวิร์กโหลดพาร์ติชัน

PV_KER_CONF
อนุญาตให้กระบวนการดำเนินการตั้งค่าระบบที่แตกต่างกัน

PV_KER_EXTCONF
อนุญาตให้กระบวนการดำเนินงานการตั้งค่าต่างๆ ในส่วนขยายเคอร์เนล

PV_KER_IPC
อนุญาตให้กระบวนการเพิ่มค่าของบัฟเฟอร์คิวข้อความ IPC และอนุญาตให้ **shmget** ที่มีช่วงที่จะรวม

PV_KER_IPC_R
อนุญาตให้กระบวนการอ่านคิวข้อความ IPC ชุดเซมาฟอร์ หรือเซ็กเมนต์หน่วยความจำที่แบ่งใช้

PV_KER_IPC_W
อนุญาตให้กระบวนการเขียนลงคิวข้อความ IPC ชุดเซมาฟอร์ หรือเซ็กเมนต์หน่วยความจำที่แบ่งใช้

PV_KER_IPC_O
อนุญาตให้กระบวนการอ่าน การแทนที่ความเป็นเจ้าของ DAC บนอ็อบเจ็กต์ IPC ทั้งหมด

PV_KER_SECCONFIG
อนุญาตให้กระบวนการตั้งค่าแฟล็กการรักษาความปลอดภัยตั้งค่า

PV_KER_PATCH
อนุญาตให้กระบวนการแพ็ตช์ส่วนขยายเคอร์เนล

privileges เลเบล:

privileges เลเบลดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_LAB_
เทียบเท่ากับเลเบล privileges อื่นทั้งหมด (**PV_LAB_***) รวมกัน

PV_LAB_CL
อนุญาตให้กระบวนการแก้ไขซับเจ็คต์ SCL ตาม clearance ของ กระบวนการ

PV_LAB_CLTL
อนุญาตให้กระบวนการแก้ไขซับเจ็คต์ TCL ตาม clearance ของ กระบวนการ

PV_LAB_LEF
อนุญาตให้กระบวนการอ่านฐานข้อมูลการเลเบล

PV_LAB_SLDG
อนุญาตให้กระบวนการดาวน์เกรด SL ตาม clearance ของกระบวนการ

PV_LAB_SLDG_STR
อนุญาตให้กระบวนการดาวน์เกรด SL ของแพ็กเก็ต ตาม clearance ของกระบวนการ

PV_LAB_SL_FILE
อนุญาตให้กระบวนการเปลี่ยนอ็อบเจ็กต์ SL ตาม clearance ของกระบวนการ

PV_LAB_SL_PROC
อนุญาตให้กระบวนการเปลี่ยนซับเจ็คต์ SL ตาม clearance ของ กระบวนการ

PV_LAB_SL_SELF
อนุญาตให้กระบวนการเปลี่ยน SL ของตัวเอง ตาม clearance ของ กระบวนการ

PV_LAB_SLUG
อนุญาตให้กระบวนการอัพเกรด SL ตาม clearance ของกระบวนการ

PV_LAB_SLUG_STR
อนุญาตให้กระบวนการอัพเกรด SL ของแพ็กเก็ต ตาม clearance ของกระบวนการ

PV_LAB_TL
อนุญาตให้กระบวนการแก้ไข subject และ object TLs

MAC privileges:

MAC privileges ดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_MAC_
เทียบเท่ากับ MAC privileges อื่นทั้งหมด (**PV_MAC_***) รวมกัน

PV_MAC_CL
อนุญาตให้กระบวนการข้ามข้อจำกัดการล้างค่าระดับความลับ

PV_MAC_R_PROC
อนุญาตให้กระบวนการข้ามข้อจำกัดการอ่าน MAC เมื่อรับ ข้อมูลเกี่ยวกับกระบวนการ โดยที่เลเบลของกระบวนการเป้าหมาย อยู่ภายใน clearance ของกระบวนการที่กระทำการ

PV_MAC_W_PROC
อนุญาตให้กระบวนการข้ามข้อจำกัดการเขียน MAC เมื่อส่ง สัญญาณไปยังกระบวนการ โดยที่เลเบลของกระบวนการเป้าหมาย อยู่ภายใน clearance ของกระบวนการที่กระทำการ

PV_MAC_R
อนุญาตให้กระบวนการข้ามข้อจำกัดการอ่าน MAC

PV_MAC_R_CL
อนุญาตให้กระบวนการข้ามข้อจำกัดการอ่าน MAC เมื่อ เลเบลของอ็อบเจ็กต์อยู่ภายใน clearance ของกระบวนการ

PV_MAC_R_STR

อนุญาตให้กระบวนการข้ามข้อจำกัดการอ่าน MAC เมื่ออ่าน ข้อความจาก STREAM โดยที่เลเบลของข้อความอยู่ภายใน clearance ของกระบวนการ

PV_MAC_W

อนุญาตให้กระบวนการข้ามข้อจำกัดการเขียน MAC

PV_MAC_W_CL

อนุญาตให้กระบวนการข้ามข้อจำกัดการเขียน MAC เมื่อ เลเบลของอ็อบเจ็กต์อยู่ภายใน clearance ของกระบวนการ

PV_MAC_W_DN

อนุญาตให้กระบวนการข้ามข้อจำกัดการเขียน MAC เมื่อ เลเบลกระบวนการควบคุมเลเบลของอ็อบเจ็กต์ และเลเบลของอ็อบเจ็กต์ อยู่ภายใน clearance ของกระบวนการ

PV_MAC_W_UP

อนุญาตให้กระบวนการข้ามข้อจำกัดการเขียน MAC เมื่อ เลเบลกระบวนการถูกควบคุมโดยเลเบลของอ็อบเจ็กต์ และเลเบลของอ็อบเจ็กต์ อยู่ภายใน clearance ของกระบวนการ

PV_MAC_OVRRD

ข้ามข้อจำกัด MAC สำหรับไฟล์ที่แฟล็กเป็นได้รับยกเว้นจาก MAC

MIC privileges:

MIC privileges ดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges, อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น, กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_MIC

อนุญาตให้กระบวนการข้ามข้อจำกัด integrity

PV_MIC_CL

อนุญาตให้กระบวนการข้ามข้อจำกัดการล้างค่า integrity

เน็ตเวิร์ก privileges:

เน็ตเวิร์ก privileges ดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_NET_

เทียบเท่ากับเน็ตเวิร์ก privileges อื่นทั้งหมด (**PV_NET_***) รวมกัน

PV_NET_CNTL
อนุญาตให้กระบวนการแก้ไขตารางเน็ตเวิร์ก

PV_NET_PORT
อนุญาตให้กระบวนการเชื่อมกับพอร์ตที่จำกัดไว้

PV_NET_RAWSOCK
อนุญาตให้กระบวนการมีการเข้าถึงโดยตรงไปยังเน็ตเวิร์กเลเยอร์

PV_NET_CONFIG
อนุญาตให้กระบวนการตั้งค่าพารามิเตอร์เกี่ยวกับเน็ตเวิร์ก

Superuser privileges:

superuser privileges ดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_SU_
เทียบเท่ากับ super user privileges อื่นทั้งหมด (**PV_SU_***) รวมกัน

PV_SU_ROOT
ให้สิทธิกระบวนการเทียบเท่ากับ privileges ทั้งหมดที่เชื่อมโยงกับ superuser มาตรฐาน

PV_SU_EMUL
ให้สิทธิกระบวนการเทียบเท่ากับ privileges ทั้งหมดที่เชื่อมโยงกับ superuser มาตรฐานเมื่อ process UID เป็น 0

PV_SU_UID
ทำให้การเรียกใช้ระบบ **getuid** ส่งกลับ 0

privileges เบ็ดเตล็ด:

privileges เบ็ดเตล็ดดังต่อไปนี้มีอยู่ใน Trusted AIX สาระสำคัญและรายละเอียดของแต่ละ privilege และการใช้งานถูกจัดเตรียมไว้ บาง privileges ฟอร์มลำดับชั้น ขณะที่หนึ่ง privilege สามารถให้สิทธิทั้งหมด ที่สัมพันธ์กับ privilege อื่น

เมื่อตรวจสอบ privileges, อย่างแรกระบบจะตรวจสอบเพื่อดูว่ากระบวนการ มี privilege ต่ำสุดที่จำเป็นหรือไม่ จากนั้นเลื่อนขึ้นไปในลำดับชั้น เพื่อตรวจสอบ privileges ที่ระดับสูงกว่า ตัวอย่างเช่น, กระบวนการ ที่มี **PV_AU_** privilege จะมี **PV_AU_ADMIN, PV_AU_ADD, PV_AU_PROC, PV_AU_READ,** และ **PV_AU_WRITE** privilege โดยอัตโนมัติและกระบวนการที่มี **PV_ROOT** privilege จะมี privileges ที่หมดที่แสดงอยู่ด้านล่างยกเว้น **PV_SU_** privileges

PV_ROOT
ให้สิทธิกระบวนการเท่ากับ privileges อื่นทั้งหมดยกเว้น **PV_SU_** (และ privileges ที่ **PV_SU_** ควบคุม)

PV_TCB
อนุญาตให้กระบวนการแก้ไขพาธไลบรารีที่ไว้วางใจของเคอร์เนล

PV_TP บ่งชี้ว่ากระบวนการเป็นกระบวนการพาธที่ไว้วางใจ และอนุญาต ให้มีการดำเนินการที่จำกัดกับกระบวนการพาธที่ไว้วางใจ

PV_TP_SET
อนุญาตให้กระบวนการเซ็ตหรือลบแฟล็กพาธที่ไว้วางใจของเคอร์เนล

PV_WPAR_CKPT
อนุญาตให้กระบวนการทำการดำเนินการ checkpoint และ restart ในพาร์ติชัน เวิร์กโหลด

PV_DEV_CONFIG
อนุญานให้กระบวนการตั้งค่าส่วนขยายเคอร์เนลระบบและอุปกรณ์

PV_DEV_LOAD
อนุญาตให้กระบวนการโหลดและยกเลิกการโหลดส่วนขยายเคอร์เนลระบบและ อุปกรณ์ในระบบ

PV_DEV_QUERY
อนุญาตให้กระบวนการเคียวรีเคอร์เนลโมดูล

การแก้ปัญหา Trusted AIX

คำตอบในคำถามอาจช่วยคุณแก้ปัญหา Trusted AIX

ฉันล็อกอินเข้าสู่ Trusted AIX ได้อย่างไร?
Trusted AIX สร้างผู้ใช้ที่ดูแลระบบสามรายในระหว่างการติดตั้งด้วยบทบาทที่เหมาะสม ตามที่กำหนดไว้ด้านล่าง

รหัสผ่านของแอคเคาต์เหล่านี้ต้องตั้งค่าไว้ เมื่อระบบบู๊ตในระหว่างแรกหลังจากการติดตั้ง Trusted AIX ถ้าคุณติดตั้งระบบ ในโหมดที่ไม่มีการแสดงพร้อมต์จากเน็ตเวิร์ก รหัสผ่านของบัญชีผู้ใช้ดีฟอลต์ เหล่านี้จะเป็นดังแสดงด้านล่าง

ผู้ใช้	รหัสผ่าน
isso	isso
sa	sa
so	so

ฉัน su ไปยัง root ได้อย่างไร?
ณ เวลาการติดตั้ง Trusted AIX, แอ็ตทริบิวต์ **su** ของ root ถูกตั้งค่าเป็น **false** ดังนั้นจึงไม่มีผู้ใช้สามารถเข้าถึงแอคเคาต์นี้ ในการเข้าถึงบัญชีผู้ใช้นี้ ผู้ใช้ การดูแลจัดการดีฟอลต์ isso และ sa จะต้องเปลี่ยนแอ็ตทริบิวต์นี้ ของบัญชีผู้ใช้ root เป็น **true** โดยใช้คำสั่ง **chuser**

ถ้า **su** ถูก เปิดใช้งานเป็น root และรหัสผ่านสำหรับบัญชีผู้ใช้ root ไม่ถูกตั้งค่า ผู้ใช้ใดก็ตาม บนระบบจะสามารถเข้าถึงบัญชีผู้ใช้ root ได้ เพื่อหลีกเลี่ยงปัญหานี้ ขอแนะนำให้ตั้งค่ารหัสผ่านของบัญชีผู้ใช้ root ก่อน การรีเซ็ตแอ็ตทริบิวต์ **su**

ฉันควรสร้างผู้ใช้การดูแลจัดการของตนเอง หรือใช้ผู้ใช้ การดูแลจัดการค่าดีฟอลต์?
ผู้ใช้การดูแลจัดการค่าดีฟอลต์ใช้สำหรับการตั้งค่าระบบ เพื่อการกำหนดค่าเองเท่านั้น ขอแนะนำเป็นอย่างยิ่ง แต่ไม่ใช่สิ่งจำเป็น ให้บัญชีผู้ใช้เหล่านี้ถูกใช้เฉพาะสำหรับการกำหนดค่าระบบเองเท่านั้น

สร้าง ผู้ใช้การดูแลจัดการของคุณเองด้วยบทบาทที่เหมาะสมเป็น isso, sa และ so และลบหรือปิดใช้งานผู้ใช้ดีฟอลต์เหล่านี้

เหตุใดฉันจึงไม่สามารถล็อกอินเข้าสู่ระบบ?

ถ้าคุณพยายามล็อกอินเป็น root (บัญชีผู้ใช้ที่มี uid 0) หรือบัญชีผู้ใช้ใดๆ ที่มี uid น้อยกว่า 128 การเข้าถึงจะถูกปฏิเสธ บัญชีผู้ใช้เหล่านี้ ถูกอ้างถึงเป็นบัญชีผู้ใช้ระบบ ในการเข้าถึงบัญชีผู้ใช้ระบบ คุณต้อง ล็อกอินเป็นผู้ใช้ที่มิใช่บัญชีผู้ใช้ระบบและ **su** ไปยังบัญชีผู้ใช้

ข้อผิดพลาดใดๆ เกี่ยวกับไฟล์การเข้ารหัสเลเบลถูกแสดงขณะล็อกอิน หรือไม่?

ถ้าไฟล์การเข้ารหัสเลเบลเสียหาย คุณจะต้องเข้าสู่ โหมดผู้ใช้เดียวเป็นผู้ใช้ root บัญชีผู้ใช้ root สามารถเข้าถึงได้ในโหมดผู้ใช้เดียวเท่านั้น

ตรวจสอบว่าไฟล์การเข้ารหัสเลเบล (`/etc/security/enc/LabelEncodings`) เหมาะสมสำหรับคำสั่ง **labck** ถ้าไฟล์ไม่เหมาะสม แก้ไขไฟล์และตรวจสอบซ้ำกับคำสั่ง **labck** ก่อนออกจาก โหมดผู้ใช้เดียว

รัน **trustchk** ในโหมดการโต้ตอบ (**trustchk -t ALL**) เพื่อตรวจสอบความถูกต้องสถานะของระบบ

ทำไมฉันจึงไม่สามารถคอมไพล์โปรแกรมใดๆ บนไลบรารี Trusted AIX which uses Trusted AIX APIs?

ชุดเครื่องมือการพัฒนาไม่ถูกติดตั้งเป็นค่าดีฟอลต์ คุณจะต้อง ติดตั้งชุดไฟล์ `bos.mls.adt` จาก สื่อบันทึกการติดตั้ง

ฉันแก้ไขการเปลี่ยนแปลงที่ทำกับสิทธิ์พิเศษของคำสั่ง ที่เป็นเหตุให้คำสั่งเหล่านั้นหยุดทำงานอย่างถูกต้องได้อย่างไร?

รัน **trustchk** ในโหมดการโต้ตอบ (**trustchk -t**) สำหรับคำสั่งเหล่านั้นเพื่อแกไขสิทธิ์พิเศษ

เหตุใดฉันไม่สามารถเข้าถึงไดเร็กทอรี `/etc/security/enc` ?

ในการเข้าถึงไดเร็กทอรี `/etc/security/enc` เชลล์จำเป็นต้องมีสิสิทธิ์พิเศษ PV_LAB_LEF และ PV_MAC_R กำหนด สิทธิ์พิเศษเหล่านี้ให้แก่เชลล์ของคุณ

ฉันจะปิดใช้งาน trustchk ตอนบูตได้อย่างไร

ลบหรือให้เครื่องหมายความคิดเห็นที่บรรทัด `trustchk` ในสคริปต์ `/etc/rc.mls`

ฉันป้องกันระบบมิให้พร้อมต์เพื่อทำการพิสูจน์ตัวตนการบูตระบบ ในทุกครั้งที่บูตได้อย่างไร?

คุณอาจเปิดใช้งานการพิสูจน์ตัวตนการบูตระบบสำหรับระบบของคุณ คุณ สามารถปิดใช้งานโดยใช้เมนู SMIT จากเมนูย่อย Trusted AIX

เหตุใดการเปลี่ยนแปลงของฉันไม่ทำงานเมื่อฉันพยายามเปลี่ยน SL ของ อ็อบเจ็กต์ระบบไฟล์?

มีความเป็นไปได้หลายทาง:

/usr/sbin/settxattr ส่งข้อความแสดงความผิดพลาดกลับมาหรือไม่?

ถ้าส่งกลับ ให้ตรวจสอบข้อความเพื่อดูข้อมูลเพิ่มเติม ตัวอย่าง:

คุณมีสิทธิในการทำงาน /usr/sbin/settxattr หรือไม่?

ถ้าไม่มี ให้ตรวจสอบสิทธิ์พิเศษ และการอนุญาตของคุณ

ไวยากรณ์ถูกต้องหรือไม่?

อ้างถึงหน้าหลัก **settxattr** เพื่อดูไวยากรณ์

SL ที่ร้องขอหรือตัวย่อมีอยู่หรือไม่?

การร้องขอ `"con a b"` จะทำงานได้บนระบบที่มี ไฟล์ Label Encodings ดีฟอลต์ (`/etc/security/enc/LabelEncodings`) แต่การร้องขอ `"conf a b"` จะไม่ทำงาน แม้ว่าทั้งสอง มีตัวย่อโลจิคัลเหมือนกันสำหรับ "confidential compartment A compartment B"

คุณจำเป็นต้องใช้เครื่องหมายคำพูดสำหรับเลเบลหลายคำหรือไม่?

`settxattr -f sl=con <filename>` จะทำงาน ได้ `settxattr -f -a sl="con a b" <filename>` จะ ทำงานได้ แต่ `settxattr -a sl=con a b <filename>` จะ ไม่ทำงาน

settxattr ส่งข้อความแสดงความผิดพลาดกลับมาหรือไม่?

ถ้าไม่มีข้อความแสดงความผิดพลาดส่งกลับมา อ็อบเจ็กต์ระบบไฟล์อาจ เป็นลิงก์สัญลักษณ์ ถ้าอ็อบเจ็กต์ที่คุณพยายามเปลี่ยนแปลงเป็นลิงก์สัญลักษณ์ อันดับแรกให้พิจารณาว่าคุณต้องการเปลี่ยน SL ของลิงก์หรืออ็อบเจ็กต์ที่ลิงก์นั้นชี้ไป **settxattr** ไม่ ไปตามค่าลิงก์แต่ตั้งค่าเลเบลสำหรับลิงก์แทน

ฉันจะติดตั้งแอ็พพลิเคชันของบุคคลที่สามเพื่อให้ทำงานบนระบบ ได้อย่างถูกต้องได้อย่างไร?

ถ้าคุณติดตั้งแอ็พพลิเคชันของบุคคลที่สามและทำงานได้ไม่ ถูกต้อง อาจเนื่องจากการเข้าถึงบางไฟล์หรือไดเร็กทอรีที่ถูกจำกัด ซึ่งต้องจำเป็นต้องใช้สิทธิ์พิเศษเพิ่ม หลังการวิเคราะห์ความจำเป็นของ แอ็พพลิเคชันที่จะเข้าถึงอ็อบเจ็กต์ที่ถูกจำกัดเหล่านี้ ให้พิจารณา สิทธิ์พิเศษที่จำเป็นดังแสดงด้านล่าง

- Assign PV_ROOT to your shell
- รัน `tracepriv -f -e <third party command>`

คำสั่งนี้จะแสดงรายการสิทธิ์พิเศษที่แอ็พพลิเคชันต้องใช้ เพื่อสิทธิ์พิเศษเหล่านี้ในฐานข้อมูลคำสั่งสิทธิ์พิเศษโดยใช้คำสั่ง **setsecattr**

เหตุใดฉันไม่สามารถทำงานบางคำสั่งได้?

เนื่องจากคำสั่งส่วนใหญ่ได้รับการป้องกันโดยการอนุญาต การทำงาน ของคำสั่งสิทธิ์พิเศษบางคำสั่งจะได้รับอนุญาตต่อเมื่อผู้ใช้ที่ ร้องขอมีการอนุญาตที่สอดคล้องกับคำสั่งนั้น ซึ่งสามารถตรวจสอบได้โดยการระบุ ว่าการอนุญาตที่จำเป็นสำหรับการทำงานของคำสั่งมีอยู่ ในหนึ่งในบทบาทที่ถูกเรียกทำงานสำหรับเซสชันปัจจุบันหรือไม่

ตรวจสอบ การอนุญาตที่แอ็คทีฟของคุณด้วย `rolelist -ae` และการอนุญาต ที่จำเป็นสำหรับคำสั่งโดยใช้ `lssecattr -c <command>`

เหตุใดบางคำสั่งจึงไม่แสดงเลเบลอย่างถูกต้อง

ส่วนใหญ่ของคำสั่งเหล่านี้ขึ้นอยู่กับไฟล์ `/etc/security/enc/LabelEncodings` สำหรับการแปลงของเลเบลให้เป็นรูปแบบที่สามารถอ่านได้ และในทางตรงกันข้าม ถ้าไฟล์นี้ เสียหาย หรือถูกแก้ไข คำสั่งอาจไม่ทำงาน ตามที่ต้องการ

แฟล็กการรักษาความปลอดภัยของไฟล์

แฟล็กความปลอดภัยของไฟล์มีผลในการเข้าถึงไฟล์ แฟล็กเหล่านี้ถูกเก็บเป็นส่วนหนึ่งของ extended attributes (EA) ของตัวไฟล์เอง แฟล็กความปลอดภัยของไฟล์ถูกกำหนดในไฟล์ส่วนหัว

FSF_APPEND

ไฟล์สามารถถูกผนวกได้เท่านั้น ไม่สามารถถูกแก้ไขได้ในโหมด operational

FSF_AUDIT

ไฟล์ถูกทำเครื่องหมายเป็นส่วนหนึ่งของระบบย่อยการตรวจสอบ เมื่อต้องการอ่านหรือเขียนไฟล์ เหล่านี้ กระบวนการต้องมี **PV_AU_READ** หรือ **PV_AU_WRITE** privileges ตามลำดับ

FSF_MAC_EXMPT

EPS ที่มี **PV_MAC_OVRRD** privilege ละเว้นข้อจำกัด MAC เมื่อมี ความพยายามเข้าถึงอ็อบเจ็กต์

FSF_PDIR

ไดเร็กทอรีเป็นไดเร็กทอรีที่พาร์ติชัน

FSF_PSDIR

ไดเร็กทอรีเป็นไดเร็กทอรีย่อยที่พาร์ติชัน

FSF_PSSDIR
ไดเร็กทอรีเป็น sub-subdirectory ที่พาร์ติชัน

FSF_TLIB
อ็อบเจ็กต์ถูกทำเครื่องหมายเป็นส่วนหนึ่งของ Trusted Library เครื่องต้องถูกรัน ในโหมด configuration หรือแฟล็กความปลอดภัยเคอร์เนล **trustedlib_enabled** ต้องเป็น OFF

FSF_TLIB_PROC
กระบวนการที่ทำเครื่องหมายเป็นกระบวนการ TLIB ทำได้เพียงลิงก์ไปที่ไลบรารี ***.so** ที่มีแฟล็ก **TLIB** เซ็ตไว้ ระบบต้องถูกรันในโหมด configuration หรือแฟล็ก ความปลอดภัยเคอร์เนล **trustedlib_enabled** ต้องเป็น OFF

คำสั่ง Trusted AIX

คำสั่ง Security-related ถูกจัดเตรียมเพื่อจัดการระบบ Trusted AIX:

labck ตรวจสอบไฟล์ LabelEncodings

getsecconf
แสดงแฟล็กการรักษาความปลอดภัยเคอร์เนล

setsecconf
เปลี่ยนแฟล็กการรักษาความปลอดภัยเคอร์เนล Trusted AIX

getsyslab
แสดงเลเบลสูงสุดและต่ำสุดของเคอร์เนล

setsyslab
เซ็ตเลเบลสูงสุดและต่ำสุดของเคอร์เนล

getrunmode
แสดงโหมดที่รันอยู่ปัจจุบันของระบบ

setrunmode
สลับโหมดที่รันอยู่ของระบบ

pdlink ลิงก์ไฟล์ขามไดเร็กทอรีย่อยที่พาร์ติชัน

pdmkdir
สร้างไดเร็กทอรีและไดเร็กทอรีย่อยที่พาร์ติชัน

pdmode
ส่งกลับโหมดการเข้าถึงไดเร็กทอรีที่พาร์ติชันปัจจุบันหรือรัน คำสั่งด้วยโหมดการเข้าถึงไดเร็กทอรีที่พาร์ติชันที่ระบุ

pdrmdir
เอาไดเร็กทอรีและไดเร็กทอรีย่อยที่เกี่ยวข้องที่พาร์ติชันออก

pdset เซ็ต/ไม่เซ็ต ไดเร็กทอรี(ย่อย) ที่พาร์ติชัน

bootauth
ตรวจสอบว่าผู้ใช้ที่ได้รับอนุญาตกำลังบูตระบบ

chuser เปลี่ยนแอ็ตทริบิวต์ clearance ของผู้ใช้

lsuser แสดงแอ็ตทริบิวต์ clearance ของผู้ใช้

chsec เปลี่ยนแอ็ตทริบิวต์ clearance และพอร์ตเลเบลของผู้ใช้

lssec แสดงแอ็ตทริบิวต์ clearance และพอร์ตเลเบลของผู้ใช้

trustchk
ตรวจสอบแอ็ตทริบิวต์ของไฟล์

lstxattr แสดงแอ็ตทริบิวต์เลเบลและแฟล็กความปลอดภัยของไฟล์ , กระบวนการ และอ็อบเจ็กต์ IPC

settxattr
เปลี่ยนแอ็ตทริบิวต์เลเบลและแฟล็กความปลอดภัยของไฟล์ , กระบวนการ และอ็อบเจ็กต์ IPC

คำประกาศ

ข้อมูลนี้จัดทำขึ้นสำหรับผลิตภัณฑ์และเซอร์วิสที่นำเสนอในสหรัฐฯ

IBM อาจไม่นำเสนอผลิตภัณฑ์ เซอร์วิส หรือคุณลักษณะที่อธิบายในเอกสารนี้ในประเทศอื่น โปรดปรึกษาตัวแทน IBM ในท้องถิ่นของคุณสำหรับข้อมูลเกี่ยวกับผลิตภัณฑ์ และเซอร์วิส ที่มีอยู่ในพื้นที่ของคุณในปัจจุบัน การอ้างอิงใดๆ ถึงผลิตภัณฑ์ โปรแกรม หรือเซอร์วิสของ IBM ไม่ได้มีวัตถุประสงค์ที่จะระบุหรือตีความว่า สามารถใช้ได้เฉพาะผลิตภัณฑ์ โปรแกรม หรือเซอร์วิสของ IBM เพียงอย่างเดียว เท่านั้น ผลิตภัณฑ์ โปรแกรม หรือเซอร์วิสใดๆ ที่สามารถทำงานได้เท่าเทียมกัน และไม่ละเมิดสิทธิทรัพย์สินทางปัญญาของ IBM อาจนำมาใช้ แทนได้ อย่างไรก็ตาม ถือเป็นความรับผิดชอบของผู้ใช้ที่จะประเมินและตรวจสอบการดำเนินการของ ผลิตภัณฑ์ โปรแกรม หรือเซอร์วิสใดๆ ที่ไม่ใช่ของ IBM

IBM อาจมีสิทธิบัตร หรืออยู่ระหว่างดำเนินการขอ สิทธิบัตรที่ครอบคลุมถึงหัวข้อซึ่งอธิบายในเอกสารนี้ การนำเสนอเอกสารนี้ไม่ได้เป็นการให้ไลเซนส์ใดๆ ในสิทธิบัตรเหล่านี้แก่คุณ คุณสามารถส่งการสอบถามเกี่ยวกับไลเซนส์ เป็นลายลักษณ์อักษรไปยัง:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America

หากมีคำถามเกี่ยวกับข้อมูลชุดอักขระไบต์คู่ (DBCS) โปรดติดต่อแผนกทรัพย์สินทางปัญญาของ IBM ในประเทศของคุณ หรือส่งคำถาม เป็นลายลักษณ์อักษร ไปยัง:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

ย่อหน้าต่อไปนี้ไม่ได้ใช้กับสหราชอาณาจักร หรือประเทศอื่นใดที่ ข้อกำหนดดังกล่าวไม่สอดคล้องกับกฎหมายท้องถิ่น: INTERNATIONAL BUSINESS MACHINES CORPORATION นำเสนอสิ่งพิมพ์นี้ "ตามสภาพ" โดยไม่มีการรับประกันใดๆ โดยชัดแจ้งหรือ โดยนัย ซึ่งรวมถึงแต่ไม่จำกัดเฉพาะการรับประกันโดยนัยถึงการไม่ละเมิด การขายได้ หรือความเหมาะสมสำหรับวัตถุประสงค์เฉพาะ เนื่องจากบางรัฐไม่อนุญาตให้ปฏิเสธการรับประกันโดยชัดแจ้งหรือ โดยนัยในธุรกรรมบางอย่าง ดังนั้น ข้อความสั่งนี้จึงอาจไม่ใช้กับคุณ

ข้อมูลนี้อาจมีความไม่ถูกต้องด้านเทคนิคหรือข้อผิดพลาดจากการพิมพ์ มีการเปลี่ยนแปลง ข้อมูลในเอกสารนี้เป็นระยะ และการเปลี่ยนแปลงเหล่านี้จะรวมอยู่ในเอดิชันใหม่ของ สิ่งพิมพ์ IBM อาจปรับปรุง และ/หรือเปลี่ยนแปลงในผลิตภัณฑ์ และ/หรือโปรแกรมที่อธิบายในสิ่งพิมพ์นี้ได้ตลอดเวลา โดยไม่ต้องแจ้งให้ทราบ

การอ้างอิงใดๆ ในข้อมูลนี้ถึงเว็บไซต์ไม่ใช่ของ IBM มีการจัดเตรียมเพื่อความสะดวกเท่านั้น และ ไม่ได้เป็นการรับรองเว็บไซต์เหล่านั้นในลักษณะใดๆ เอกสารประกอบที่เว็บไซต์เหล่านั้นไม่ได้เป็น ส่วนหนึ่งของเอกสารประกอบสำหรับผลิตภัณฑ์ IBM นี้ และการใช้เว็บไซต์เหล่านั้นถือเป็นความเสี่ยงของคุณเอง

IBM อาจใช้ หรือแจกจ่าย ข้อมูลใดๆ ที่คุณให้ในวิธีที่ IBM เชื่อว่าเหมาะสมโดยไม่ก่อให้เกิดข้อผูกมัดใดๆ กับ คุณ

ผู้รับไลเซนส์ของโปรแกรมนี้ที่ต้องการข้อมูลเกี่ยวกับโปรแกรมสำหรับวัตถุประสงค์ในการเปิดใช้งาน: (i) การแลกเปลี่ยนข้อมูลระหว่างโปรแกรมที่สร้างขึ้นอย่างอิสระกับโปรแกรมอื่น (รวมถึง โปรแกรมนี้) และ (ii) การใช้ข้อมูลซึ่งแลกเปลี่ยนร่วมกัน ควร ติดต่อ:

IBM Corporation
Dept. LRAS/Bldg. 903
11501 Burnet Road
Austin, TX 78758-3400
USA

ข้อมูลดังกล่าวอาจพร้อมใช้งาน ภายใต้ข้อตกลงและเงื่อนไขที่เหมาะสม รวมถึง การชำระค่าธรรมเนียมในบางกรณี

โปรแกรมที่มีไลเซนส์ซึ่งอธิบายในเอกสารนี้ และเอกสารประกอบที่มีไลเซนส์ทั้งหมดสำหรับโปรแกรม นั้น มีการจัดเตรียมโดย IBM ภายใต้ข้อตกลงของข้อตกลงกับลูกค้าของ IBM, ข้อตกลงไลเซนส์โปรแกรมระหว่างประเทศของ IBM หรือข้อตกลงที่เท่าเทียมกันใดๆ ระหว่างเรา

ข้อมูลประสิทธิภาพใดๆ ที่มีในเอกสารนี้ถูกกำหนดในสภาวะแวดล้อมที่ควบคุม ด้วยเหตุนี้ ผลลัพธ์ที่ได้ในสภาวะแวดล้อมการปฏิบัติการอื่นจึงอาจแตกต่างไปอย่างมาก การวัดบางอย่างอาจ ดำเนินการบนระบบที่อยู่ระหว่างการพัฒนา และไม่มีการรับประกันว่าการวัดเหล่านี้จะ เหมือนกันบนระบบที่พร้อมใช้งานโดยทั่วไป ยิ่งไปกว่านั้น การวัดบางอย่างอาจมีการประเมินโดยวิธีการ ประมาณค่านอกช่วง ผลลัพธ์จริงอาจแตกต่างไป ผู้ใช้เอกสารนี้จึงควรตรวจสอบ ข้อมูลที่สามารถใช้ได้สำหรับสภาวะแวดล้อมของตน

ข้อมูลเกี่ยวกับผลิตภัณฑ์ที่ไม่ใช่ของ IBM ได้รับมาจากซัพพลายเออร์ของผลิตภัณฑ์เหล่านั้น ประกาศที่เผยแพร่ หรือแหล่งข้อมูลที่เปิดเผยต่อสาธารณะ IBM ไม่ได้ทดสอบผลิตภัณฑ์ดังกล่าว และไม่สามารถยืนยันความถูกต้องของ ประสิทธิภาพ ความเข้ากันได้ หรือการเรียกร้องอื่นใดที่เกี่ยวข้องกับผลิตภัณฑ์ที่ไม่ใช่ของ IBM คำถามเกี่ยวกับ ความสามารถของผลิตภัณฑ์ที่ไม่ใช่ของ IBM ควรส่งไปยังซัพพลายเออร์ของผลิตภัณฑ์เหล่านั้น

ข้อความทั้งหมดเกี่ยวกับทิศทางหรือเจตนาในอนาคตของ IBM อาจมีการเปลี่ยนแปลง หรือเพิกถอนได้โดยไม่ต้องแจ้งให้ทราบ และแสดงถึงเป้าหมายและวัตถุประสงค์เท่านั้น

ราคาของ IBM ทั้งหมดที่แสดงเป็นราคาขายปลีกที่แนะนำของ IBM ซึ่งเป็นราคาปัจจุบัน และอาจเปลี่ยนแปลงได้โดยไม่ต้องแจ้งให้ทราบ ราคาของผู้แทนจำหน่ายอาจแตกต่างไป

ข้อมูลนี้ใช้สำหรับวัตถุประสงค์ของการวางแผนเท่านั้น ข้อมูลในเอกสารนี้อาจมีการเปลี่ยนแปลง ก่อนผลิตภัณฑ์ที่อธิบายจะวางจำหน่าย

ข้อมูลนี้มีตัวอย่างของข้อมูลและรายงานที่ใช้ในการดำเนินการทางธุรกิจรายวัน เพื่อ สาธิตข้อมูลให้สมบูรณ์ที่สุดเท่าที่จะเป็นไปได้ ตัวอย่างจึงมีชื่อของแต่ละบุคคล บริษัท ยี่ห้อ และผลิตภัณฑ์ ชื่อทั้งหมดเหล่านี้เป็นชื่อสมมติ และการคล้ายคลึงในชื่อและที่อยู่ซึ่งหน่วยธุรกิจจริงใช้เป็นความบังเอิญโดยสิ้นเชิง

ไลเซนส์ลิขสิทธิ์:

ข้อมูลนี้มีตัวอย่างแอ็พพลิเคชันโปรแกรมในภาษาต้นฉบับ ซึ่งแสดงถึง เทคนิคด้านโปรแกรมในหลากหลายแพล็ตฟอร์ม คุณอาจคัดลอก ปรับเปลี่ยน และแจกจ่าย โปรแกรมตัวอย่างเหล่านี้ในรูปแบบใดๆ โดยไม่ต้องชำระเงินให้แก่ IBM สำหรับวัตถุประสงค์ในการพัฒนา การใช้ การตลาด หรือการแจกจ่ายโปรแกรมแอ็พพลิเคชัน ที่สอดคล้องกับอินเตอร์เฟสการเขียนโปรแกรมแอ็พพลิเคชันสำหรับแพล็ตฟอร์มปฏิบัติการ ซึ่งเขียน โปรแกรมตัวอย่าง ตัวอย่างเหล่านี้ยังไม่ได้ผ่านการทดสอบในทุกสภาพ ดังนั้น IBM จึงไม่สามารถรับประกัน หรือบอกเป็นนัยถึง ความน่าเชื่อถือ ความสามารถบริการได้ หรือฟังก์ชันของโปรแกรมเหล่านี้ โปรแกรมตัวอย่างมีการนำเสนอ "ตาม สภาพ" โดยไม่มีการรับประกันประเภทใดๆ IBM ไม่รับผิดชอบ ต่อความเสียหายใดๆ ที่เกิดขึ้นเนื่องจากการใช้โปรแกรมตัวอย่างของคุณ

แต่ละสำเนา หรือส่วนใดๆ ของโปรแกรมตัวอย่างเหล่านี้ หรืองานที่สืบเนื่องใดๆ ต้องมี คำประกาศลิขสิทธิ์ดังนี้:

ส่วนของโค้ดนี้ได้มาจากโปรแกรมตัวอย่างของ IBM Corp.

© Copyright IBM Corp. (C) ลิขสิทธิ์ IBM Corp. _ป้อน ปี_ สงวนสิทธิ์ทั้งหมด

สิ่งที่ต้องพิจารณาเกี่ยวกับนโยบายความเป็นส่วนตัว

ผลิตภัณฑ์ซอฟต์แวร์ของ IBM รวมถึงโซลูชันบริการระบบซอฟแวร์ ("ข้อเสนอซอฟต์แวร์") อาจใช้คุกกี้หรือเทคโนโลยีอื่นเพื่อรวบรวมข้อมูลการใช้งานผลิตภัณฑ์ เพื่อช่วยในการปรับปรุงประสบการณ์การใช้งานของผู้ใช้ขั้นปลาย เพื่อปรับแต่งการโต้ตอบกับ ผู้ใช้ขั้นปลาย หรือเพื่อวัตถุประสงค์อื่นๆ ในหลายๆ กรณี จะไม่มีการรวบรวม ข้อมูลอัตลักษณ์ส่วนบุคคลโดย ข้อเสนอซอฟต์แวร์ ซึ่งข้อเสนอซอฟต์แวร์บางอย่าง สามารถช่วยให้คุณรวบรวมข้อมูลอัตลักษณ์ ส่วนบุคคลได้ ถ้าข้อเสนอซอฟต์แวร์นี้ใช้คุกกี้ เพื่อรวบรวมข้อมูลอัตลักษณ์, ระบุข้อมูล เกี่ยวกับการใช้คุกกี้ของข้อเสนอนี้ถูกกำหนดไว้ด้านล่าง

ข้อเสนอซอฟต์แวร์นี้ไม่ใช้คุกกี้ หรือเทคโนโลยีอื่นเพื่อรวบรวมข้อมูลอัตลักษณ์ส่วนบุคคล

ถ้าคอนฟิกูเรชันถูกปรับใช้สำหรับ ข้อเสนอกนี้จัดเตรียมให้คุณในฐานะลูกค้าสามารถรวบรวม ข้อมูลอัตลักษณ์ส่วนบุคคลจากผู้ใช้ขั้นปลายผ่านทางคุกกี้ และเทคโนโลยีอื่น คุณควรปรึกษากับที่ปรึกษาด้านกฎหมายเกี่ยวกับ ที่ใช้บังคับในการรวบรวมข้อมูล รวมถึงข้อกำหนดต่างๆ เพื่อการแจ้งเตือนและการยินยอม

สำหรับข้อมูลเพิ่มเติมเกี่ยวกับการใช้ เทคโนโลยีต่างๆ รวมถึงคุกกี้ สำหรับวัตถุประสงค์เหล่านี้ โปรดดู นโยบายความเป็นส่วนตัวของ IBM ที่ http://www.ibm.com/privacy และ คำชี้แจงสิทธิส่วนบุคคลออนไลน์ของ IBM ที่ส่วน http://www.ibm.com/privacy/details "Cookies, Web Beacons and Other Technologies" และ "IBM Software Products and Software-as-a-Service Privacy Statement" ที่ http://www.ibm.com/software/info/product-privacy

---

เครื่องหมายการค้า

IBM, ตราสัญลักษณ์ IBM , และ ibm.com เป็นเครื่องหมายการค้าหรือเครื่องหมายการค้าที่จดทะเบียนของ International Business Machines Corp. ซึ่งจดทะเบียนในหลายเขตอำนาจศาลทั่วโลก ชื่อผลิตภัณฑ์และการบริการอื่นอาจเป็นเครื่องหมายการค้าของ IBM หรือบริษัทอื่น รายการปัจจุบันของเครื่องหมายการค้า IBM มีอยู่บนเว็บไซต์ที่ ข้อมูลลิขสิทธิ์และเครื่องหมายการค้า ที่ www.ibm.com/legal/copytrade.shtml

Linux เป็นเครื่องหมายการค้าจดทะเบียนของ Linus Torvalds ในสหรัฐอเมริกา ประเทศอื่นๆ หรือทั้งสองกรณี

Microsoft และ Windows คือเครื่องหมายการค้าของ Microsoft Corporation ในสหรัฐอเมริกา ประเทศอื่นๆ หรือทั้งสอง

Java และเครื่องหมายการค้าและตราสัญลักษณ์ที่สร้างขึ้นจาก Java ทั้งหมดเป็นเครื่องหมายการค้าที่จดทะเบียนของ Oracle และ/หรือ บริษัทในเครือ

UNIX เป็นเครื่องหมายการค้าที่จดทะเบียนของ The Open Group ในสหรัฐฯ และประเทศอื่นๆ

ดัชนี

อักขระพิเศษ

A

B

C

D

E

F

H

I

K

L

M

N

O

P

R

S

T

V

W

X

ก

ค

จ

ซ

ฐ

ด

ต

น

บ

ป

ผ

พ

ฟ

ม

ร

ล

ว

ส

ห

อ

พิมพ์ในสหรัฐอเมริกา