

WebSphere Application Server



Load Balancer 관리 안내서

버전 6.1

WebSphere Application Server



Load Balancer 관리 안내서

버전 6.1

주!

이 정보와 이 정보가 지원하는 제품을 사용하기 전에, 527 페이지의 부록 E 『주의사항』을 읽으십시오.

초판(2006년 5월)

이 책은 다음에 적용됩니다.

WebSphere Application Server, 버전 6.1

새 개정판에 달리 명시되지 않는 한 후속 릴리스 및 수정판에 적용됩니다.

IBM 담당자 또는 해당 지역의 IBM 지사를 통해 책을 주문할 수 있습니다.

© Copyright International Business Machines Corporation 2005. All rights reserved.

목차

표	xiii
그림	xv
이 책에 대한 정보	xvii
이 책의 사용자.	xvii
참조서 정보.	xvii
내게 필요한 옵션	xvii
독자 의견 전송 방법.	xviii
관련 문서 및 웹 페이지	xix

제 1 부 Load Balancer 소개 1

제 1 장 Load Balancer 개요	3
Load Balancer 개념.	3
사용할 수 있는 Load Balancer의 컴포넌트는?	3
Load Balancer 사용의 장점은?	4
Load Balancer가 고가용성을 제공하는 방법	6
Dispatcher	6
CBR	6
Cisco CSS Controller 또는 Nortel Alteon Controller	6
새로운 기능.	7

제 2 장 Load Balancer 컴포넌트 개요	9
Load Balancer 컴포넌트 종류	9
Dispatcher 컴포넌트 개요	9
Dispatcher로 로컬 서버 관리	11
Dispatcher 및 Metric Server를 사용하여 서버 관리.	12
Dispatcher로 로컬 및 원격 서버 관리	13
CBR(Content Based Routing) 컴포넌트 개요.	13
CBR로 로컬 서버 관리	14
Site Selector 컴포넌트 개요.	15
Site Selector 및 Metric Server를 사용하여 로컬 및 원격 서버 관리	16
Cisco CSS Controller 컴포넌트 개요	17
Nortel Alteon Controller 컴포넌트 개요	18

제 3 장 네트워크 관리: 사용할 Load Balancer 기 능 결정.	21
관리자, 어드바이저 및 Metric Server 기능 (Dispatcher, CBR 및 Site Selector 컴포넌트)	21

Dispatcher 컴포넌트 기능	21
원격 관리	21
결합 배치	22
고가용성	22
클라이언트 대 서버 연관 관계	22
규칙 기반 로드 밸런스.	22
Dispatcher의 cbr 전달 메소드를 사용한 Content Based Routing	23
광역 로드 밸런스	24
포트 맵핑	25
개인용 네트워크에 Dispatcher 설정	25
와일드 카드 클러스터 및 와일드 카드 포트.	25
"서비스 거부" 중지 감지	25
2진 로그	25
경보.	25
CBR(Content Based Routing) 컴포넌트 기능.	26
CBR 컴포넌트와 Dispatcher 컴포넌트의 cbr 전 달 메소드 간의 비교	26
원격 관리	26
결합 배치	27
Caching Proxy의 복수 인스턴스에 대한 CBR	27
SSL 접속을 위한 Content Based Routing 제공	27
서버 파티션	27
규칙 기반 로드 밸런스.	27
클라이언트 대 서버 연관 관계	28
Dispatcher 및 CBR을 사용한 고가용성	28
2진 로그	28
경보.	28
Site Selector 컴포넌트 기능.	28
원격 관리	29
결합 배치	29
고가용성	29
클라이언트 대 서버 연관 관계	29
규칙 기반 로드 밸런스.	29
광역 로드 밸런스	30
경보.	30
Cisco CSS Controller 컴포넌트 기능	30
원격 관리	30
결합 배치	31
고가용성	31
2진 로그	31
경보.	31

Nortel Alteon Controller 컴포넌트 기능	31
원격 관리	31
결합 배치	32
고가용성	32
2진 로그	32
정보	32
제 4 장 Load Balancer 설치	33
AIX 시스템 요구사항 및 설치	34
AIX 시스템의 요구사항	34
AIX 시스템용 설치	34
설치하기 전에	35
설치 단계	35
HP-UX 시스템 요구사항 및 설치	38
HP-UX 시스템의 요구사항	38
HP-UX 시스템용 설치	38
설치하기 전에	38
설치 단계	38
Linux 시스템 요구사항 및 설치	40
Linux 시스템의 요구사항	40
Linux 시스템용 설치	40
설치하기 전에	40
설치 단계	40
Solaris 시스템 요구사항 및 설치	42
Solaris 요구사항	42
Solaris에 설치	42
설치하기 전에	42
설치 단계	43
Windows 시스템 요구사항 및 설치	44
Windows 시스템의 요구사항	44
Windows 시스템용 설치	44
설치하기 전에	44
설치 단계	45

제 2 부 Dispatcher 컴포넌트. 47

제 5 장 빠른 시작 구성	49
필요한 내용	50
준비 방법	50
Dispatcher 컴포넌트 구성	51
명령행을 사용한 구성	51
구성 검사	52
GUI(Graphical User Interface)를 사용한 구성	52
구성 마법사	52
클러스터, 포트, 서버 구성의 유형	53
제 6 장 Dispatcher 계획	57

계획 고려사항	57
전달 메소드	59
Dispatcher의 MAC 레벨 경로 지정(MAC 전달 메소드).	59
Dispatcher의 NAT/NAPT(nat 전달 메소드)	59
Dispatcher content-based routing(cbr 전달 메소드)	61
Dispatcher의 nat 또는 cbr 전달 메소드 구성에 대한 샘플 단계	63
서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)	64
HTTP 또는 HTTPS 어드바이저를 사용한 서버 파티셔닝	65
물리적 서버를 논리적 서버로 구성하는 예	65
고가용성	66
단순 고가용성	66
상호 고가용성	67

제 7 장 Dispatcher 구성	69
구성 TASK 개요	69
구성 방법	69
명령행	70
스크립트	70
GUI	71
구성 마법사를 사용한 구성	72
Dispatcher 시스템 설정	72
1단계. 서버 기능 시작.	75
2단계. 실행 프로그램 기능 시작	75
3단계. 비전달 주소 정의(호스트 이름과 다른 경우)	75
4단계. 클러스터 정의 및 클러스터 옵션 설정	75
5단계. 네트워크 인터페이스 카드의 별명 지정	76
6단계. 포트 정의 및 포트 옵션 설정	77
7단계. 로드 밸런스 서버 시스템 정의.	78
8단계. 관리자 기능 시작(선택)	78
9단계. 어드바이저 기능 시작(선택)	78
10단계. 필요한 클러스터 비율 설정	79
서버 시스템의 시스템 설정	79
1단계. 루프백 장치에 별명 지정	79
2단계. 여분의 라우트 확인	83
3단계. 여분의 라우트 삭제	84
4단계. 서버의 올바른 구성 여부 확인.	84
Load Balancer의 mac 전달 사용 시 Linux 루프백 별명 지정 대안	85

제 8 장 IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치	89
--	-----------

IPv4 및 IPv6용 Load Balancer에 지원되는 플랫폼	90
사용자 영역의 로드 밸런스에 지원되는 플랫폼	90
특수 Linux 플랫폼 고려사항	90
백엔드 서버 제한사항	91
IPv4 및 IPv6용 Load Balancer 설치	91
IPv4 및 IPv6용 Load Balancer에 대한 특수 고려 사항 및 제한사항	92
IPv6 링크 로컬 주소 구성	92
동일한 클러스터/서버 쌍	92
지원되지 않는 Dispatcher 기능	92
어드바이저 구성	93
고가용성 구성	93
서버 결합 배치	94
사용자 영역에서 실행되는 시스템의 동질 관계 기 능(Linux)	94
Metric Server 구성	95
IPv4 및 IPv6용 Load Balancer의 IPv6 패킷 처리 사용 가능	96
IPv4 및 IPv6용 Load Balancer의 인터페이스 장치 별명 지정	97
zSeries Linux에 필요한 클러스터 구성 단계	100
IPv4 및 IPv6용 Load Balancer용 Dispatcher 명 령(dscontrol)	101
명령 구문 차이점	101
지원되는 dscontrol 명령	101
지원되지 않는 dscontrol 명령	105

제 3 부 CBR(Content Based Routing) 컴포넌트 107

제 9 장 빠른 시작 구성	109
필요한 내용	109
준비 방법	110
CBR 컴포넌트 구성	110
명령행을 사용한 구성	110
구성 검사	112
GUI(Graphical User Interface)를 사용한 구성	112
구성 마법사를 사용한 구성	112
클러스터, 포트, 서버 구성의 유형	113

제 10 장 Content Based Routing 계획	117
계획 고려사항	117
다른 유형의 콘텐츠에 대한 요청 로드 밸런스	118
더 나은 응답 시간을 위해 사이트의 콘텐츠 나누 기	118
웹 서버 콘텐츠의 백업 제공	119

CPU 이용 향상을 위해 복수 Caching Proxy 프 로세스 사용	119
CBR과 함께 규칙 기반 로드 밸런스 사용	119
완전 보안(SSL) 연결의 로드 밸런스	120
SSL의 클라이언트-투-프록시 및 HTTP의 프록 시-투-서버의 로드 밸런스	120

제 11 장 Content Based Routing 구성	123
구성 TASK 개요	123
구성 방법	123
명령행	124
스크립트	125
GUI	126
구성 마법사	127
CBR 시스템 설정	128
1단계. CBR을 사용하기 위해 Caching Proxy 구성	128
2단계. 서버 기능 시작	129
3단계. 실행 프로그램 기능 시작	130
4단계. 클러스터 정의 및 클러스터 옵션 설정	130
5단계. 네트워크 인터페이스 카드 별명 지정(선택 적)	130
6단계. 포트 정의 및 포트 옵션 설정	131
7단계. 로드 밸런스 서버 시스템 정의	132
8단계. 구성에 규칙 추가	132
9단계. 규칙에 서버 추가	132
10단계. 관리자 기능 시작(선택적)	132
11단계. 어드바이저 기능 시작(선택적)	132
12단계. 필요한 클러스터 비율 설정	133
13단계. Caching Proxy 시작	133
CBR 구성 예제	133

제 4 부 Site Selector 컴포넌트 135

제 12 장 빠른 시작 구성	137
필요한 내용	137
준비 방법	138
Site Selector 컴포넌트 구성	138
명령행을 사용한 구성	138
구성 검사	139
GUI(Graphical User Interface)를 사용한 구성	140
구성 마법사를 사용한 구성	140

제 13 장 Site Selector 계획	141
계획 고려사항	141
TTL 고려사항	143
네트워크 근접 기능 사용	144

제 14 장 Site Selector 구성	145
구성 TASK 개요	145
구성 방법	145
명령행	146
스크립트	146
GUI	147
구성 마법사	148
Site Selector 시스템 설정	148
1단계. 서버 기능 시작	148
2단계. 이름 서버 시작	149
3단계. 사이트 이름 정의 및 사이트 이름 옵션 설정	149
4단계. 로드 밸런스 서버 시스템 정의	149
5단계. 관리자 기능 시작(선택적)	149
6단계. 어드바이저 기능 시작(선택적)	150
7단계. 시스템 메트릭 정의(선택적)	150
8단계. 필요한 사이트 이름 비율 설정	150
서버 시스템의 시스템 설정	150

제 5 부 Cisco CSS Controller 컴포넌트 151

제 15 장 빠른 시작 구성	153
필요한 내용	153
준비 방법	154
Cisco CSS Controller 컴포넌트 구성	154
명령행을 사용한 구성	154
구성 검사	155
GUI(Graphical User Interface)를 사용한 구성	155

제 16 장 Cisco CSS Controller 계획	157
시스템 요구사항	157
계획 고려사항	157
네트워크의 컨설턴트 배치	158
고가용성	160
가중치 계산	161
문제점 판별	161

제 17 장 Cisco CSS Controller 구성	163
구성 TASK 개요	163
구성 방법	163
명령행	164
XML	165
GUI	165
Controller for Cisco CSS Switches 시스템 설정	166
1단계. 서버 기능 시작	167
2단계. 명령 인터페이스 시작	167

3단계. 컨설턴트 구성	167
3단계. ownercontent 구성	167
4단계. 서비스의 올바른 정의 확인	167
5단계. 메트릭 구성	167
6단계. 컨설턴트 시작	168
7단계. Metric Server 시작(선택)	168
8단계. 고가용성 구성(선택)	168
구성 검사	168

제 6 부 Nortel Alteon Controller 컴포넌트 169

제 18 장 빠른 시작 구성	171
필요한 내용	171
준비 방법	172
Nortel Alteon Controller 컴포넌트 구성	173
명령행을 사용한 구성	173
구성 검사	174
GUI(Graphical User Interface)를 사용한 구성	174

제 19 장 Nortel Alteon Controller 계획	175
시스템 요구사항	175
계획 고려사항	176
네트워크의 컨설턴트 배치	176
스위치에서의 서버 속성(제어기에 의해 설정됨)	179
백업 서버 구성	179
그룹 구성	180
고가용성	181
조정	182
문제점 판별	183

제 20 장 Nortel Alteon Controller 구성	185
구성 TASK 개요	185
구성 방법	185
명령행	185
XML	186
GUI	187
Nortel Alteon Controller 설정	188
1단계. 서버 기능 시작	189
2단계. 명령 인터페이스 시작	189
3단계. Nortel Alteon Web Switch consultant 정의	189
4 단계. 스위치 컨설턴트에 서비스 추가	189
5단계. 메트릭 구성	189
6단계. 컨설턴트 시작	189
7단계. 고가용성 구성(선택)	190
8단계. Metric Server 시작(선택)	190

9단계. Nortel Alteon Controller 구성 새로 고침	190
구성 검사	190

제 7 부 Load Balancer의 기능 및 고급 기능. 191

제 21 장 Dispatcher, CBR 및 Site Selector에 대한 관리자, 어드바이저 및 Metric Server 기능 . 193	
Load Balancer에서 제공하는 로드 밸런스 최적화 . 194	
상태 정보에 제공되는 중요성 비율	194
가중치.	195
관리자 간격.	197
감도 임계치.	197
스무스 색인.	198
스크립트를 사용하여 정보나 레코드 서버 장에 생성	198
어드바이저	199
어드바이저 작동 방법	200
어드바이저 시작 및 정지	200
어드바이저 간격	201
어드바이저 보고서 제한시간	202
어드바이저 연결 제한시간 및 서버의 수신 제한 시간	202
어드바이저 재시도.	203
어드바이저 목록	203
응답 및 요청(URL) 옵션을 사용하여 HTTP 또는 HTTPS 어드바이저 구성	205
상하단부 WAN 구성에서 자가 어드바이저 사용 . 206	
사용자 정의(사용자 정의 가능) 어드바이저 작성 . 207	
WAS 어드바이저	208
이름 지정 규칙.	209
컴파일.	209
실행	209
필수 루틴	210
탐색 순서	210
이름 지정 및 경로.	211
예제 어드바이저	211
Metric Server	211
WLM 제한사항	211
전제조건	212
Metric Server 사용 방법	212
작업로드 관리자 어드바이저	214
Metric Server 제한사항.	214

제 22 장 Dispatcher, CBR 및 Site Selector에 대한 고급 기능. 215

결합 배치된 서버 사용	216
Dispatcher 컴포넌트의 경우	216
CBR 컴포넌트.	218
Site Selector 컴포넌트의 경우	218
고가용성	218
고가용성 구성	219
하트 비트 및 도달 목표를 사용하는 고장 검색 기능	222
복구 전략	222
스크립트 사용	223
결합 배치 및 고가용성 구성(Windows 시스템) . 225	
규칙 기반 로드 밸런스 구성	226
규칙 평가 방법.	227
클라이언트 IP 주소에 따라 규칙 사용	228
클라이언트 포트에 따라 규칙 사용	228
시간에 따라 규칙 사용	229
서비스 유형(TOS)에 기반하여 규칙 사용	229
초당 연결에 따라 규칙 사용	229
충 활성 연결에 따라 규칙 사용	230
예약된 대역폭 및 공유 대역폭에 따라 규칙 사용 . 230	
Metric all 규칙	232
Metric average 규칙.	233
항상 참인 규칙 사용.	233
요청 콘텐츠에 따라 규칙 사용.	234
포트 연관 관계 무시.	234
사용자 구성에 규칙 추가	234
규칙에 대한 서버 평가 옵션	235
Load Balancer에 대한 연관 관계 기능 사용법 . . 236	
연관 관계가 사용 불가능할 때의 작동	237
연관 관계가 사용 가능할 때의 작동.	237
포트간 연관 관계	237
연관 관계 주소 마스크(stickymask).	238
서버 연결 처리 작업중지	239
클라이언트 요청의 콘텐츠에 기본적인 규칙에 대한 연 관 관계 옵션	239
활성 쿠키 연관 관계	240
수동 쿠키 연관 관계	242
URI 연관 관계.	243
광역 Dispatcher 지원 구성.	244
명령 구문	245
Dispatcher의 광역 지원으로 원격 어드바이저 사 용	245
구성 예제	248
GRE(일반 경로 지정 캡슐화) 지원	250
명시적 링크 사용	252
개인용 네트워크 구성 사용.	252

와일드 카드 클러스터를 사용하여 서버 구성 조합	253
와일드 카드 클러스터를 사용하여 방화벽 로드 밸런싱 수행	254
투명 프록시의 경우 Caching Proxy가 있는 와일드 카드 클러스터 사용	254
와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정	255
FTP 통신량을 처리하기 위한 와일드 카드 포트	255
서비스 거부 중지 감지	255
서버 통제를 분석하기 위해 2진 로그 사용	257
결합 배치된 클라이언트 사용	259

제 23 장 Cisco CSS Controller 및 Nortel

Alteon Controller의 고급 기능	261
결합 배치	261
고가용성	261
구성	262
장애 발견	263
복구 전략	263
예제	264
Load Balancer에서 제공하는 로드 밸런싱 최적화	264
메트릭 정보에 제공된 중요도	264
가중치	265
가중치 계산 휴면 시간	266
감도 임계치	266
어드바이저	266
어드바이저 작동 방법	266
어드바이저 휴면 시간	267
어드바이저 연결 제한시간 및 서버의 수신 제한 시간	267
어드바이저 재시도	268
사용자 정의(사용자 정의 기능) 어드바이저 작성	268
이름 지정 규칙	269
컴파일	270
실행	270
필수 루틴	271
탐색 순서	271
이름 지정 및 경로	271
예제 어드바이저	272
Metric Server	272
전제조건	272
Metric Server 사용 방법	272
작업로드 관리자 어드바이저	274
서버 통제를 분석하기 위해 2진 로그 사용	275
스크립트를 사용하여 정보나 레코드 서버 장애 생성	276

제 8 부 Load Balancer 관리 및 문제점

해결 279

제 24 장 Load Balancer 작동 및 관리	281
Load Balancer의 원격 관리	281
원격 메소드 호출(RMI)	282
웹 기반 관리	283
Load Balancer 로그 사용	285
Dispatcher, CBR 및 Site Selector	285
Cisco CSS Controller 및 Nortel Alteon Controller	287
Dispatcher 컴포넌트 사용	288
Dispatcher 시작 및 정지	288
활동해제 제한시간 값 사용	288
finetimeout 및 staletimeout을 사용한 연결 레코 드 정리 제어	289
GUI 보고 — 모니터 메뉴 옵션	289
Dispatcher 컴포넌트에 Simple Network Management Protocol 사용	290
ipchain 또는 iptable을 사용하여 Load Balancer 시스템(Linux 시스템)을 굳히는 모든 통신량을 거절함	297
Content Based Routing 컴포넌트 사용	298
CBR 시작 및 정지	298
CBR 제어	298
CBR 로그 사용	299
Site Selector 컴포넌트 사용	299
Site Selector 시작 및 정지	299
Site Selector 제어	299
Site Selector 로그 사용	299
Cisco CSS Controller 컴포넌트 사용	299
Cisco CSS Controller 시작 및 정지	299
Cisco CSS Controller 제어	300
Cisco CSS Controller 로그 사용	300
Nortel Alteon Controller 컴포넌트 사용	300
Nortel Alteon Controller 시작 및 정지	300
Nortel Alteon Controller 제어	300
Nortel Alteon Controller 로그 사용	300
Metric Server 컴포넌트 사용	301
Metric Server 시작 및 정지	301
Metric Server 로그 사용	301
제 25 장 문제점 해결	303
문제점 해결 정보 집적	303
일반 정보(항상 필수)	303
고가용성(HA) 문제점	305

어드바이저 문제점	305
Content Based Routing 문제점	306
클러스터를 히트할 수 없음	307
기타 모두 장애	307
업그레이드	308
Java 코드	308
유용한 링크	308
문제점 해결 테이블	308
Dispatcher 포트 번호 확인	319
CBR 포트 번호 확인	319
Site Selector 포트 번호 확인	320
Cisco CSS Controller 포트 번호 확인	321
Nortel Alteon Controller 포트 번호 확인	322
공통 문제점 해결—Dispatcher	323
문제점: Dispatcher가 실행되지 않음	323
문제점: Dispatcher 및 서버가 응답하지 않음	323
문제점: Dispatcher 요청이 밸런스를 이루지 않음	323
문제점: Dispatcher 고가용성 기능이 작동하지 않음	324
문제점: 하트비트를 추가할 수 없음(Windows 플랫폼)	324
문제점: 추가 라우트(Windows 2000)	324
문제점: 어드바이저가 제대로 작동하지 않음	324
문제점: Dispatcher, Microsoft IIS 및 SSL이 작동하지 않음(Windows 플랫폼)	325
문제점: 원격 시스템에 대한 Dispatcher 연결	325
문제점: dscontrol 또는 lbadm 명령 실패	325
온라인 도움말을 보려고 할 때 문제점: “파일을 찾을 수 없습니다...”라는 오류 메시지 발생 (Windows 플랫폼)	326
문제점: GUI(Graphical User Interface)가 올바르게 시작되지 않음	326
문제점: Caching Proxy가 설치된 Dispatcher 실행 중 오류	327
문제점: GUI(Graphical User Interface)가 올바르게 표시되지 않음	327
문제점: Windows 플랫폼에서 때로 도움말 창이 다른 열린 창 뒤로 사라짐	327
문제점: Load Balancer가 프레임을 처리하고 전달할 수 없음	327
문제점: Load Balancer 실행 프로그램을 시작할 때 파란색 화면이 표시됨	328
문제점: Discovery 경로로 인해 Load Balancer와의 리턴 통신이 발생하지 못함	328

문제점: Load Balancer의 광역 모드에서 고가용성이 작동되지 않음	329
문제점: GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작업을 합니다	329
문제점: 구성 갱신 후 lbadm이 서버로부터 연결이 끊어짐	330
문제점: 원격 연결을 통해 IP 주소가 제대로 분 석되지 않음	330
문제점: 한글 Load Balancer 인터페이스가 AIX 및 Linux 시스템에서 겹치거나 원치 않는 글꼴 을 표시함	330
문제점: Windows 시스템에서 hostname과 같은 명령을 발행하면 로컬 주소 대신 별명 주소가 리턴됨	331
문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동	331
문제점: “rmmod ibmlb”를 실행할 때 예상치 않은 작동(Linux 시스템)	332
문제점: Dispatcher 시스템에서 명령 실행 중 응답 시간이 느림	332
문제점: SSL 또는 HTTPS 어드바이저가 서버 로드를 등록하지 않음(mac 전달을 사용할 경우)	332
문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐	333
문제점: 소켓 풀링이 사용 가능하며 웹 서버가 0.0.0.0으로 바인드됨	333
문제점: Windows 시스템에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨	334
문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생	334
문제점: Windows 시스템에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함	335
문제점: Windows 플랫폼에서 어댑터에 둘 이상의 주소를 구성할 경우 호스트 이름에 대한 IP 주소를 확인하는 중에 문제가 발생함	335
문제점: Windows 시스템에서 네트워크 정지 후 고가용성 설정에서 어드바이저가 작동하지 않음	336
문제점: 루프백 장치에서 여러 클러스터 별명을 지정할 경우 Linux 시스템에서 “IP address add” 명령을 사용하지 않음	337
문제점: “포트 메소드에 대해 지정되지 않은 또는 올바른지 않은 라우터 주소” 오류 메시지	337
문제점: Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 창을 종료할 때 프로세스가 종료됨	338

문제점: Load Balancer 구성 로드 시 지연 . . .	338
문제점: Windows 시스템에서 IP 주소 충돌 오류 메시지가 나타남	339
문제점: 고가용성 구성에서 기본 시스템 및 백업 시스템 모두가 활성화됨	339
문제점: 대형 페이지 응답 리턴 시 클라이언트 요청 실패	339
문제점: Windows 시스템에서 dscontrol 또는 lbadm 명령을 발행하면 "서버가 응답하지 않음" 오류가 발생	340
문제점: 고가용성 Dispatcher 시스템의 qeth 드라이버에 있는 S/390 시스템용 Linux에 동기화가 불가능할 수도 있음	340
문제점: 고가용성 구성에 대한 팁	340
문제점: Linux에서는 개방형 시스템 어댑터 (OSA) 카드가 있는 zSeries 또는 S/390 서버 사용 시 Dispatcher 구성 제한사항이 있음 . . .	342
문제점: 일부 Linux 버전에서는 관리자 및 어드바이저로 구성된 Dispatcher 실행 시 메모리 누수가 발생함	344
문제점: SUSE Linux Enterprise Server 9에서는 Dispatcher가 패킷을 전달하지만 패킷은 백엔드 서버에 도달하지 않음	345
문제점: Windows 시스템에서는 고가용성 인계 중에 IP 주소 충돌 메시지가 나타남	346
문제점: Linux iptables가 패킷 라우팅에 간섭할 수 있음	346
문제점: IPv6 서버를 Solaris 시스템의 Load Balancer 구성에 추가할 수 없음	347
서비스 수정사항 설치 시 Java 경고 메시지가 나타남	347
Load Balancer 설치가 공급된 Java 파일 세트 업그레이드	347
공통 문제점 해결—CBR	348
문제점: CBR이 실행되지 않음	348
문제점: cbrcontrol 또는 lbadm 명령 실패	348
문제점: 요청이 로드 밸런스로 되지 않음	348
문제점: Solaris 시스템에서 cbrcontrol executor start 명령 실패	349
문제점: 구문 또는 구성 오류	349
문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동 . . .	349
문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐	349

문제점: Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨 . . .	350
문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생	350
문제점: Windows 시스템에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함	350
문제점: Windows 시스템에서 어댑터에 둘 이상의 주소를 구성할 경우 호스트 이름에 대한 IP 주소를 확인하는 중에 문제가 발생함	350
공통 문제점 해결—Site Selector	351
문제점: Site Selector가 실행되지 않음	351
문제점: Site Selector가 Solaris 클라이언트로부터 통신을 라운드 로빙하지 않음	351
문제점: sscontrol 또는 lbadm 명령 실패	351
문제점: ssserver가 Windows 플랫폼에서 시작에 실패함	352
문제점: Site Selector가 중복 라우트를 통해 올바르게 로드 밸런스하지 않음	352
문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동 . . .	352
문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐	353
문제점: Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨 . . .	353
문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생	353
문제점: Windows 시스템에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함	353
공통 문제점 해결—Cisco CSS Controller	354
문제점: ccserver가 시작되지 않음	354
문제점: ccocontrol 또는 lbadm 명령 실패	354
문제점: 포트 13099에서 레지스트리를 작성할 수 없음	355
문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동 . . .	355
문제점: 컨설턴트를 추가할 때 연결 오류를 받음	355
문제점: 스위치의 가중치가 갱신되지 않음	355
문제점: Refresh 명령이 컨설턴트 구성을 갱신하지 않음	356
문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐	356
문제점: Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨 . . .	356

문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생	356
공통 문제점 해결—Nortel Alteon Controller	356
문제점: nalserver가 시작되지 않음	356
문제점: nalcontrol 또는 lbadm 명령 실패	357
문제점: 포트 14099에서 레지스트리를 작성할 수 없음	357
문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동	358
문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐	358
문제점: 컨설턴트를 추가할 때 연결 오류를 받음	358
문제점: 스위치의 가중치가 갱신되지 않음	358
문제점: Refresh 명령이 컨설턴트 구성을 갱신하지 않음	359
문제점: Windows 시스템에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨	359
문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생	359
공통 문제점 해결—Metric Server	359
문제점: .bat 또는 .cmd 사용자 메트릭 파일을 실행하는 Windows 플랫폼의 Metric Server IOException	359
문제점: Metric Server가 Load Balancer 시스템에 로드를 보고하지 않음	360
문제점: Metric Server 로그에서 "에이전트에 액세스하려면 서명이 필요합니다."라고 보고합니다.. . . .	360
문제점: AIX 시스템에서 과부하된 상태로 Metric Server를 실행할 경우, ps -vg 명령 출력이 손상될 수도 있음	360
문제점:고가용성 Dispatcher에서 Site Selector 로드 밸런싱을 사용하여 2계층 구성으로 Metric Server를 구성함	361
문제점: 여러 개의 CPU가 있는 Solaris 시스템에서 실행 중인 스크립트가 원치 않는 콘솔 메시지를 생산	362
문제점: IPv6용 Load Balancer에서 Linux 시스템의 Metric Server로부터 값을 검색할 수 없음.	362
문제점: Metric Server 시작 후 메트릭 값이 -1을 리턴함	363

제 9 부 명령어 참조서 365

제 26 장 구문 다이어그램 읽는 방법	367
기호 및 구두점	367
매개변수	367

구문 예제	367
-----------------	-----

제 27 장 Dispatcher 및 CBR 명령어 참조서	369
CBR 및 Dispatcher 간의 구성 차이점	370
dscontrol advisor — 어드바이저 제어	372
dscontrol binlog — 2진 로그 파일 제어	378
dscontrol cluster — 클러스터 구성	379
dscontrol executor — 실행 프로그램 제어	383
dscontrol file — 구성 파일 관리	388
dscontrol help — 이 명령의 도움말 표시 또는 인쇄	390
dscontrol highavailability — 고가용성 제어	391
dscontrol host — 원격 시스템 구성	395
dscontrol logstatus — 서버 로그 설정 표시	396
dscontrol manager — 관리자 제어	397
dscontrol metric — 시스템 메트릭 구성	403
dscontrol port — 포트 구성	404
dscontrol rule — 규칙 구성	411
dscontrol server — 서버 구성	418
dscontrol set — 서버 로그 구성	425
dscontrol status — 관리자 및 어드바이저가 실행 여부 표시	426
dscontrol subagent — SNMP 서브에이전트 구성	427

제 28 장 Site Selector 명령어 참조서	429
sscontrol advisor — 어드바이저 제어	430
sscontrol file — 구성 파일 관리	435
sscontrol help — 이 명령의 도움말 표시 또는 인쇄	437
sscontrol logstatus — 서버 로그 설정 표시	438
sscontrol manager — 관리자 제어	439
sscontrol metric — 시스템 메트릭 구성	444
sscontrol nameserver — NameServer 제어	445
sscontrol rule — 규칙 구성	446
sscontrol server — 서버 구성	449
sscontrol set — 서버 로그 구성	451
sscontrol sitename — 사이트 이름 구성	452
sscontrol status — 관리자 및 어드바이저가 실행 여부 표시	455

제 29 장 Cisco CSS Controller 명령어 참조서	457
ccocontrol consultant — 컨설턴트 구성 및 제어	458
ccocontrol controller — 제어기 관리	461
ccocontrol file — 구성 파일 관리	463
ccocontrol help — 이 명령의 도움말 표시 또는 인쇄	465
ccocontrol highavailability — 고가용성 제어	466

cococontrol metriccollector — 메트릭 콜렉터 구성	470
cococontrol ownercontent — 소유자 이름 및 콘텐츠 규칙 제어	472
cococontrol service — 서비스 구성	475

제 30 장 Nortel Alton Controller 명령어 참조

서	477
nalcontrol consultant — 컨설턴트 구성 및 제어	478
nalcontrol controller — 제어기 관리	482
nalcontrol file — 구성 파일 관리	484
nalcontrol help — 이 명령의 도움말 표시 또는 인쇄	486
nalcontrol highavailability — 고가용성 제어	487
nalcontrol metriccollector — 메트릭 콜렉터 구성	491
nalcontrol server — 서버 구성	493
nalcontrol service — 서비스 구성	495

부록 A. GUI: 일반 명령	499
------------------	-----

부록 B. 콘텐츠 규칙(패턴) 구문	507
---------------------	-----

콘텐츠 규칙(패턴) 구문:	507
예약된 키워드	507

부록 C. 예제 구성 파일	511
예제 Load Balancer 구성 파일	511
Dispatcher 구성 파일 — AIX, Linux 및 Solaris 시스템	511
Dispatcher 구성 파일 — Windows 시스템	514
어드바이저 예제	518

부록 D. Dispatcher, CBR 및 Caching Proxy를 사용하는 2단 고가용성 구성 예제	523
서버 시스템 설정	523

부록 E. 주의사항	527
상표	529

용어	531
----	-----

색인	541
----	-----

표

1. AIX installp 이미지	34	11. Nortel Alteon Controller 컴포넌트 구성 타스 크	185
2. AIX 설치 명령	36	12. Load Balancer의 고급 구성 타스크	193
3. Load Balancer용 HP-UX 패키지 설치 세부사 항.	38	13. Load Balancer의 고급 구성 타스크	215
4. Dispatcher 기능의 구성 타스크	69	14. Dispatcher 문제점 해결 테이블	308
5. Dispatcher의 루프백 장치(lo0)에 별명 지정 명 령.	80	15. CBR 문제점 해결 테이블	313
6. Dispatcher에 대한 여분의 라우트 삭제 명령	84	16. Site Selector 문제점 해결 테이블.	314
7. CBR 컴포넌트의 타스크 구성	123	17. Controller for Cisco CSS Switches 문제점 해결 테이블	315
8. NIC 별명 명령	131	18. Nortel Alteon Controller 문제점 해결 테이블	316
9. Site Selector 컴포넌트 구성 타스크	145	19. Metric Server 문제점 해결 테이블	317
10. Cisco CSS Controller 컴포넌트 구성 타스크	163		

그림

1. Dispatcher를 사용하여 로컬 서버를 관리하는 사이트의 물리적 표현 예제	11	25. 간단한 Cisco CSS Controller 구성	153
2. Dispatcher 및 Metric Server를 사용하여 서버를 관리하는 사이트 예제	12	26. 스위치 뒤에 접속된 컨설턴트의 예	159
3. Dispatcher를 사용하여 로컬 및 원격 서버를 관리하는 사이트 예제	13	27. 스위치 앞의 사용자 인터페이스를 사용하여 스위치 뒤에 구성된 컨설턴트 예제(선택적 고가용성 상대가 있음)	160
4. CBR을 사용하여 로컬 서버를 관리하는 예제	14	28. 간단한 Nortel Alteon Controller 구성	171
5. Site Selector 및 Metric Server를 사용하여 로컬 및 원격 서버를 관리하는 사이트 예제.	16	29. 스위치 뒤에 접속된 컨설턴트의 예	177
6. Cisco CSS Controller 및 Metric Server를 사용하여 로컬 서비스를 관리하는 사이트 예제.	18	30. 스위치 앞에 인트라넷을 통해 접속된 컨설턴트의 예	178
7. Nortel Alteon Controller를 사용하여 로컬 서버를 관리하는 사이트 예제	19	31. 스위치 뒤의 컨설턴트 및 스위치 앞의 사용자 인터페이스 예	178
8. 간단한 로컬 Dispatcher 구성	49	32. 백업 서버에 구성된 컨설턴트 예제	180
9. 단일 클러스터와 두 개의 포트에 구성된 Dispatcher에 대한 예제	53	33. Nortel Alteon Controller 및 Nortel Alteon Web Switch 고가용성 예제	182
10. 각각 단일 포트인 두 개의 클러스터로 구성된 Dispatcher에 대한 예제	54	34. 자가 어드바이저를 사용하는 상하단부 WAN 구성의 예제	206
11. 각각 포트가 두 개인 두 개의 클러스터로 구성된 Dispatcher에 대한 예제	55	35. 단일 LAN 세그먼트를 구성하는 구성 예제	244
12. Dispatcher의 nat 또는 cbr 전달 메소드 사용 예제	63	36. 로컬 및 원격 서버를 사용하는 구성 예제	245
13. 고가용성을 사용하는 Dispatcher 예제.	66	37. 원격 Load Balancer의 광역 구성 예제	248
14. 상호 고가용성을 사용하는 Dispatcher 예제	67	38. GRE를 지원하는 서버 플랫폼의 광역 구성 예제	251
15. Dispatcher 시스템에 대해 필요한 IP 주소 예제.	75	39. Dispatcher를 사용하는 사설 네트워크 예제	253
16. 간단한 로컬 CBR 구성	109	40. Linux 및 UNIX 시스템의 SNMP 명령	291
17. 단일 클러스터와 두 개의 포트에 구성된 CBR에 대한 예제	113	41. GUI(Graphical User Interface)는 Dispatcher 컴포넌트의 GUI 트리 구조 확장을 표시합니다.	500
18. 각각 단일 포트인 두 개의 클러스터로 구성된 CBR에 대한 예제	114	42. GUI(Graphical User Interface)는 CBR 컴포넌트의 GUI 트리 구조 확장을 표시합니다.. . . .	501
19. 각각 포트가 두 개인 두 개의 클러스터로 구성된 CBR에 대한 예제.	115	43. GUI(Graphical User Interface)는 Site Selector 컴포넌트의 GUI 트리 구조 확장을 표시합니다.	502
20. AIX, Linux 및 Solaris 시스템용 CBR 구성 파일	129	44. GUI(Graphical User Interface)는 Cisco CSS Controller 컴포넌트의 GUI 트리 구조 확장을 표시합니다.. . . .	503
21. HP-UX 시스템용 CBR 구성 파일	129	45. GUI(Graphical User Interface)는 Nortel Alteon Controller 컴포넌트의 GUI 트리 구조 확장을 표시합니다.	504
22. Windows 시스템용 CBR 구성 파일	129	46. Dispatcher, CBR 및 Caching Proxy를 사용하는 2단 고가용성 구성 예제	523
23. 간단한 Site Selector 구성	137		
24. DNS 환경의 예제	141		

이 책에 대한 정보

이 책에서는 AIX®, HP-UX, Linux™, Solaris 및 Windows® 운영 체제용 IBM® WebSphere® Application Server Load Balancer의 계획, 설치, 구성, 사용 및 문제점 해결 방법을 설명합니다. 이전에는 이 제품을 Edge Server Network Dispatcher, SecureWay® Network Dispatcher, eNetwork Dispatcher 및 Interactive Network Dispatcher라고 했습니다.

이 책의 사용자

*Load Balancer 관리 안내서*는 운영 체제 및 인터넷 서비스에 대해 잘 알고 있는 숙련된 네트워크 및 시스템 관리자용으로 작성되었습니다. Load Balancer를 사용하기 전에는 필요하지 않습니다.

이 책은 이전 Load Balancer 릴리스는 지원하지 않습니다.

참조서 정보

Edge Components Information Center 웹 사이트는 HTML 및 PDF 형식으로 이 책의 최신 버전에 연결합니다.

Load Balancer에 관한 최신 갱신사항을 보려면 웹 사이트 지원 페이지 방문 및 테크 노트 사이트 링크를 실행하십시오.

이들 및 관련 웹 페이지를 액세스하려면, xix 페이지의 『관련 문서 및 웹 페이지』에 나열된 URL로 이동하십시오.

내게 필요한 옵션

내게 필요한 옵션은 신체적 장애가 있는 사용자(예: 지체 부자유자 및 시각 장애인)가 성공적으로 소프트웨어 제품에 액세스할 수 있게 합니다. 다음은 Load Balancer에서의 기본 옵션 기능입니다.

- 화면 판독기 소프트웨어 및 디지털 음성 합성기를 사용하여 화면에 표시된 내용을 들을 수 있습니다. IBM ViaVoice®와 같은 음성 인식 소프트웨어를 사용하여 데이터를 입력하고 사용자 인터페이스를 탐색할 수도 있습니다.
- 마우스 대신 키보드를 사용하여 기능을 조작할 수 있습니다.
- 제공된 그래픽 인터페이스 대신 표준 텍스트 편집기 또는 명령 인터페이스를 사용하여 Load Balancer 기능을 구성하고 관리할 수 있습니다. 특정 기능의 액세스 기능성에 대한 자세한 정보는 이 기능에 대한 문서를 참조하십시오.

독자 의견 전송 방법

여러분의 피드백은 정확하고 고품질의 정보를 제공하는 데 중요한 역할을 합니다. 이 책이나 그림의 Edge Components 문서에 관한 의견이 있는 경우,

- 여러분의 의견을 전자 우편으로 ibmkspoe@kr.ibm.com에 보내 주십시오. 의견에는 책 이름, 책 번호, 버전 및 가능한 경우 설명하고 있는 텍스트의 특정 위치(예: 페이지 번호 또는 테이블 번호)를 명시하십시오.

관련 문서 및 웹 페이지

- *Edge Components* 개념, 계획 및 설치 GA30-2919-00
- *Edge Components* 프로그래밍 안내서 GA30-2915-00
- *Caching Proxy* 관리 안내서 GA30-2916-00
- IBM 웹 사이트 홈: www.ibm.com/
- IBM WebSphere Application Server 제품:
www.ibm.com/software/webservers/appserv/
- IBM WebSphere Application Server 라이브러리 웹 사이트:
www.ibm.com/software/webservers/appserv/was/library/
- IBM WebSphere Application Server 지원 웹 사이트:
www.ibm.com/software/webservers/appserv/was/support/
- IBM WebSphere Application Server Information Center:
www.ibm.com/software/webservers/appserv/infocenter.html
- IBM WebSphere Application Server Edge Components Information Center:
www.ibm.com/software/webservers/appserv/ecinfocenter.html

제 1 부 Load Balancer 소개

이 파트에서는 Load Balancer의 개요 및 해당 컴포넌트, 사용 가능한 구성 기능에 대한 상위 레벨의 설명, 하드웨어 및 소프트웨어 요구사항 목록, 설치 명령을 제공합니다. 다음 장을 포함합니다.

- 3 페이지의 제 1 장 『Load Balancer 개요』
- 9 페이지의 제 2 장 『Load Balancer 컴포넌트 개요』
- 21 페이지의 제 3 장 『네트워크 관리: 사용할 Load Balancer 기능 결정』
- 33 페이지의 제 4 장 『Load Balancer 설치』

제 1 장 Load Balancer 개요

이 장에는 Load Balancer 개요와 다음 섹션으로 구성되어 있습니다.

- 『Load Balancer 개념』
- 『사용할 수 있는 Load Balancer의 컴포넌트는?』
- 4 페이지의 『Load Balancer 사용의 장점은?』
- 6 페이지의 『Load Balancer가 고가용성을 제공하는 방법』
- 7 페이지의 『새로운 기능』

어떤 기능을 사용하여 네트워크를 관리할 것인지를 계획할 수 있도록 각 Load Balancer 컴포넌트에서 제공하는 상위 레벨의 구성 기능 목록을 보려면 21 페이지의 제 3 장 『네트워크 관리: 사용할 Load Balancer 기능 결정』의 내용을 참조하십시오.

Load Balancer 개념

[[는 서버를 통한 수신 클라이언트 요청을 분배하는 소프트웨어 솔루션입니다. 이 제품은 서버 그룹 내에서 다른 서버로 TCP 세션 요청을 지정하여 서버의 성능을 향상시킵니다. 이런 방법으로, 모든 서버 간 요청의 균형을 유지합니다. 이 로드 밸런스는 사용자와 다른 응용프로그램에 명료하게 이루어집니다. [[는 전자 우편 서버, WWW(World Wide Web) 서버, 분산 병렬 데이터베이스 조회 및 다른 TCP/IP 응용프로그램과 같은 응용프로그램에 유용합니다.

웹 서버에 사용하면, [[는 가장 많이 요구되는 문제점에 대한 강력하고 융통성 있는 확장 가능한 솔루션을 제공하여 사용자 사이트의 잠재력을 최대화하는 데 도움을 줄 수 있습니다. 방문객이 꼭 원하는 시기에 사이트에 접속할 수 없는 경우, Load Balancer는 자동으로 최적의 서버를 찾아 수신 요청을 처리하므로 고객 만족도와 사용자의 수익성이 향상됩니다.

사용할 수 있는 Load Balancer의 컴포넌트는?

중요: IPv4 및 IPv6용 Load Balancer를 사용하는 경우, Dispatcher 컴포넌트만이 사용 가능합니다. 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』에서 자세한 정보를 참조하십시오.

Load Balancer는 탁월한 로드 밸런스 결과를 제공하기 위해 독립적으로 사용하거나 함께 사용할 수 있는 5개의 컴포넌트로 이루어져 있습니다.

- **Dispatcher** 컴포넌트를 사용하여 Dispatcher에서 동적으로 설정된 다수의 가중치와 측정치를 사용하는 근거리 통신망 또는 광역 통신망 내에서 서버의 로드 밸런스를 수

행할 수 있습니다. 이 컴포넌트는 HTTP, FTP, SSL, NNTP, IMAP, POP3, SMTP, SIP 및 Telnet과 같은 특정 서비스 레벨의 로드 밸런스를 제공합니다. 이 컴포넌트는 도메인 이름을 IP 주소에 매핑하기 위해 도메인 이름 서버를 사용하지 않습니다.

HTTP 프로토콜의 경우, Dispatcher content-based routing 기능을 사용하여 클라이언트 요청 콘텐츠를 기반으로 로드 밸런스를 수행할 수 있습니다. 선택된 서버는 지정된 규칙과 일치하는 URL 결과입니다. Dispatcher의 CBR(cbr 전달 메소드)에는 Caching Proxy가 필요하지 않습니다.

- HTTP 및 HTTPS(SSL) 프로토콜의 경우, CBR(Content Based Routing) 컴포넌트를 사용하여 클라이언트 요청 콘텐츠를 기반으로 로드 밸런스를 수행할 수 있습니다. 클라이언트는 Caching Proxy로 요청을 전송하며 Caching Proxy는 이 요청을 해당 서버로 전송합니다. 선택된 서버는 지정된 규칙과 일치하는 URL 결과입니다.

주: CBR(Content Based Routing) 컴포넌트는 64비트 JVM 실행 플랫폼을 제외한 모든 지원 플랫폼에서 사용 가능합니다. 혹은 Load Balancer에 있는 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 Caching Proxy를 사용하지 않고도 CBR을 제공할 수 있습니다. 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 페이지를 참조하십시오.

- **Site Selector** 컴포넌트를 사용하여 DNS 라운드 로빙 접근 방식이나 보다 고급의 사용자 지정 접근 방식을 사용하는 로컬 또는 광역 네트워크 내의 서버에 로드 밸런스를 수행할 수 있습니다. Site Selector는 이름 서버와 함께 DNS 이름을 IP 주소에 매핑하는 작업을 합니다.
- **Cisco CSS Controller** 또는 **Nortel Alteon Controller** 컴포넌트를 사용하여 최적 서버 선택, 로드 최적화 및 결합 허용을 위해 각각 Cisco CSS Switch 또는 Nortel Alteon Web Switch로 전송된 서버 가중치를 생성할 수 있습니다.

Dispatcher, CBR, Site Selector, Cisco CSS Controller 및 Nortel Alteon Controller 컴포넌트에 대한 자세한 정보는 9 페이지의 『Load Balancer 컴포넌트 종류』를 참조하십시오.

Load Balancer 사용의 장점은?

글로벌 인터넷에 연결되는 사용자와 네트워크 수는 기하급수적으로 증가하고 있습니다. 이러한 증가로 인해 자주 사용되는 사이트에서 사용자 액세스가 제한되는 확장성 문제가 발생합니다.

현재, 네트워크 관리자는 다양한 방법을 사용하여 액세스를 최대화하려고 노력하고 있습니다. 이러한 방법 중 일부에서는 이전 선택이 느리거나 응답이 없을 때 임의로 다른 서버를 선택할 수 있습니다. 이러한 접근 방식은 다루기 번거로우며 비효율적입니다. 다른 방법으로는 표준 라운드 로빙 방식이 있는데, 이 방법은 도메인 이름 서버가 요청을 처리하기 위해 차례대로 서버를 선택합니다. 이 방식이 더 낫기는 하지만 서버의 작

업 부하를 고려하지 않고 너무 많은 통신량을 전송하므로 여전히 비효율적입니다. 또한 서버가 실패할 경우에도 요청은 계속 전송됩니다.

더 강력한 솔루션의 필요성에 대한 결과물이 Load Balancer입니다. 이것은 이전의 경쟁적인 솔루션에 비해 많은 이점이 있습니다.

확장성

클라이언트 요청 수가 증가함에 따라 서버를 동적으로 추가할 수 있으며, 10개 또는 수백 개의 서버에서도 하루에 천만 개의 요청을 지원할 수도 있습니다.

장비 사용의 효율성

로드 밸런스는 각 서버 그룹이 표준 라운드 로빙 방법을 사용할 때 자주 발생하는 핫 스팟(hot spot)을 최소화하여 하드웨어 사용을 최적화합니다.

용이한 통합

Load Balancer는 표준 TCP/IP 또는 UDP/IP 프로토콜을 사용합니다. 네트워크를 물리적으로 변경하지 않고도 기존의 네트워크에 이를 추가할 수 있습니다. 설치하고 구성하는 방법도 아주 간단합니다.

낮은 오버헤드

간단한 mac 레벨 전달 메소드를 사용하면, Dispatcher 컴포넌트는 클라이언트에서 서버로 들어오는 인바운드 플로우만 살펴봅니다. 서버에서 클라이언트로의 아웃바운드 플로우는 보지 않아도 됩니다. 이로 인해, 다른 접근 방식에 비해 응용프로그램으로의 영향력이 현저하게 감소되며 네트워크의 성능은 향상될 수 있습니다.

고가용성

Dispatcher, Cisco CSS Controller 및 Nortel Alteon Controller 컴포넌트는 고가용성을 제공하며, 기본 서버 시스템에 오류가 발생할 경우, 로드 밸런스를 언제든지 인계 받도록 계속 준비 상태로 남아 있는 대기 시스템을 이용합니다. 서버 중 하나가 실패하면, 요청은 다른 서버에 의해 계속 제공됩니다. 이 프로세스는 단일 장애 지점인 서버를 제거하여 사이트 사용 가능성이 높아집니다. 자세한 정보는 6 페이지의 『Load Balancer가 고가용성을 제공하는 방법』을 참조하십시오.

Content Based Routing(CBR 컴포넌트 또는 Dispatcher 컴포넌트 사용)

Caching Proxy와 함께 CBR 컴포넌트는 요청된 콘텐츠를 기반으로 특정 서버에 HTTP 및 HTTPS(SSL) 요청을 위임할 수 있습니다. 예를 들어, URL의 디렉토리 부분에 있는 "/cgi-bin/" 문자열이 요청에 포함되어 있고 서버 이름이 로컬 서버이면 CBR은 cgi 요청 처리를 위해 특별히 할당된 서버 세트 중 최상의 서버로 요청을 지정할 수 있습니다.

주: CBR(Content Based Routing) 컴포넌트는 64비트 JVM 실행 플랫폼을 제외한 모든 지원 플랫폼에서 사용 가능합니다. 혹은 Load Balancer에 있는 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 Caching Proxy를 사용하지 않고도 CBR을 제공할 수 있습니다. 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 페이지를 참조하십시오.

Dispatcher 컴포넌트는 Content Based Routing을 제공하기도 하지만 Caching Proxy는 설치하지 않아도 됩니다. 패킷이 수신될 때 Dispatcher 컴포넌트의 Content Based Routing이 커널에서 수행되기 때문에 CBR 컴포넌트에 비해 신속한 Content Based Routing을 제공할 수 있습니다. Dispatcher 컴포넌트는 HTTP("content" 유형 규칙 사용) 및 HTTPS(SSL 세션 ID 연관 관계 사용)에 대해 content-based routing을 수행합니다.

주: 메시지 암호 해독 및 재암호화를 요구하는 HTTP 요청의 콘텐츠를 기반으로 통신량을 로드 밸런싱할 때 CBR 컴포넌트만이 HTTPS(SSL)의 콘텐츠 규칙을 사용할 수 있습니다.

Load Balancer가 고가용성을 제공하는 방법

Dispatcher

Dispatcher 컴포넌트는 내장 고가용성 기능을 제공하여 네트워크에서 단일 장애 지점인 Dispatcher를 제거합니다. 이 기능은 기본 시스템을 모니터링하는 보조 Dispatcher 시스템의 사용을 포함하며, 기본 시스템이 실패할 경우 언제든지 로드 밸런싱 작업을 넘겨 받도록 대기합니다. Dispatcher 컴포넌트는 두 시스템이 모두 상호 기본 또는 보조(백업) 시스템이 되도록 허용하는 상호 고가용성 기능도 제공합니다. 219 페이지의 『고가용성 구성』을 참조하십시오.

CBR

또한 상하단부 구성을 CBR이 있는 다중 서버로 통신량을 로드 밸런싱하는 Dispatcher 시스템과 함께 사용할 때 CBR 컴포넌트를 사용하여 고가용성 레벨을 달성할 수도 있습니다.

Cisco CSS Controller 또는 Nortel Alteon Controller

제어기는 단일 장애 지점인 제어기를 제거하는 고가용성 기능을 갖고 있습니다. 한 시스템의 제어기는 기본 제어기로 구성되고 다른 시스템의 제어기는 백업 제어기로 구성될 수 있습니다. 백업 제어기는 기본 제어기를 모니터링하고 기본 제어기가 실패할 경우 스위치에 서버 가중치를 제공하는 작업을 인수하려고 대기합니다. 261 페이지의 『고가용성』에서 자세한 정보를 참조하십시오.

새로운 기능

IBM WebSphere Application Server용 Load Balancer 버전 6.1에는 많은 새로운 기능이 포함되어 있습니다. 여기서는 가장 중요한 새로운 기능만 나열합니다.

- **Linux** 시스템 사용자 영역의 로드 밸런스 프로세스 실행 지원

IPv4 및 IPv6용 Load Balancer 설치에 지원이 추가되어 커널 영역이 아닌 사용자 영역에서 로드 밸런스 프로세스를 실행합니다. Linux 시스템의 경우 더이상 커널 모듈에 대한 종속성이 없습니다.

사용자 영역(kernel free)의 처리를 지원하는 시스템에 대한 최신 정보는 다음의 웹 사이트를 참조하십시오.

<http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921>

자세한 내용은 90 페이지의 『IPv4 및 IPv6용 Load Balancer에 지원되는 플랫폼』을 참조하십시오.

- **PA-RISC의 HP 11iv2** 지원(HP 11iv1에서는 제거된 지원)

지원되는 하드웨어 및 소프트웨어 요구사항에 대한 정보는 다음 웹 사이트

<http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921>을 참조하십시오.

- **zSeries 64비트** 시스템의 **Linux** 지원

zSeries 64비트 시스템의 Linux 지원은 IPv4 및 IPv6용 Load Balancer 설치에 대해서만 지원됩니다.

zSeries 64비트 시스템의 Linux 실행에 필요한 특수 고려사항 및 IPv4 및 IPv6용 Load Balancer에 대한 정보는 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』 페이지를 참조하십시오.

지원되는 하드웨어 및 소프트웨어 요구사항에 대한 정보는 다음 웹 사이트 <http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921>을 참조하십시오.

- **SIP** 어드바이저 지원

SIP(Session Initiation Protocol) 어드바이저 지원이 추가되었습니다. 지원되는 SIP 어드바이저는 TCP 프로토콜에서만 실행됩니다.

자세한 정보는 203 페이지를 참조하십시오.

- **Linux** 시스템의 경우, 결합 배치된 클라이언트 구성 지원

이 기능은 모든 Load Balancer 컴포넌트에 적용됩니다.

Load Balancer와 동일한 시스템에 있는 클라이언트는 Linux 시스템에서만 지원됩니다.

자세한 내용은 259 페이지의 『결합 배치된 클라이언트 사용』을 참조하십시오.

- **Firefox** 브라우저 지원

지원되는 Firefox 버전 및 지원되는 모든 브라우저에 대한 정보는 다음 웹 사이트 <http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921>을 참조하십시오.

제 2 장 Load Balancer 컴포넌트 개요

이 장에는 Load Balancer 컴포넌트의 개요와 다음 섹션으로 구성되어 있습니다.

- 『Load Balancer 컴포넌트 종류』
- 『Dispatcher 컴포넌트 개요』
- 13 페이지의 『CBR(Content Based Routing) 컴포넌트 개요』
- 15 페이지의 『Site Selector 컴포넌트 개요』
- 17 페이지의 『Cisco CSS Controller 컴포넌트 개요』
- 18 페이지의 『Nortel Alteon Controller 컴포넌트 개요』

어떤 기능을 사용하여 네트워크를 관리할 것인지를 계획할 수 있도록 각 Load Balancer 컴포넌트에서 제공하는 상위 레벨의 구성 기능 목록을 보려면 21 페이지의 제 3 장 『네트워크 관리: 사용할 Load Balancer 기능 결정』의 내용을 참조하십시오.

Load Balancer 컴포넌트 종류

Load Balancer의 다섯 개 컴포넌트는 Dispatcher, CBR(Content Based Routing), Site Selector, Cisco CSS Controller 및 Nortel Alteon Controller 입니다. Load Balancer 는 사용자의 사이트 구성에 따라 컴포넌트를 개별적으로 사용하거나 함께 사용할 수 있는 융통성을 제공합니다. 여기서는 이들 컴포넌트에 대한 개요를 제공합니다.

중요: IPv4 및 IPv6용 Load Balancer를 사용하는 경우, Dispatcher 컴포넌트만이 사용 가능합니다. 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』에서 자세한 정보를 참조하십시오.

Dispatcher 컴포넌트 개요

Dispatcher 컴포넌트는 로드 밸런스 및 관리 소프트웨어를 고유하게 조합하여 서버 간 통신량 밸런스를 조절합니다. Dispatcher는 실패한 서버를 감지하여 그 주위의 통신량을 전달할 수도 있습니다. Dispatcher는 HTTP, FTP, SSL, SMTP, NNTP, IMAP, POP3, Telnet, SIP와 기타 TCP 또는 독립적인 UDP 기반 응용프로그램을 지원합니다.

Dispatcher 시스템으로 전송된 모든 클라이언트 요청은 동적으로 설정된 특정 가중치에 따라 적절하게 최적의 서버로 전송됩니다. 그 가중치의 기본값을 사용하거나 구성 프로세스 동안 값을 변경할 수 있습니다.

Dispatcher는 다음 세 가지 전달 메소드(포트에 지정됨)를 제공합니다.

- **MAC 전달 메소드(mac)**. 이 전달 메소드를 통해 Dispatcher는 수신 요구를 서버로 로드 밸런스합니다. 서버는 Dispatcher 개입없이 응답을 클라이언트로 직접 리턴합니다.
- **NAT/NAPT 전달 메소드(nat)**. Dispatcher의 NAT(네트워크 주소 변환) / NAPT(네트워크 주소 포트 변환) 기능을 사용하면 백엔드 서버가 로컬로 연결된 네트워크에 있어야 한다는 제한사항이 제거됩니다. 서버가 원격 위치에 있을 때는 GRE(Generic Routing Encapsulation)/WAN(Wide Area Network) 기술을 사용하지 않고 nat 기술을 사용할 수 있습니다. nat 전달 메소드를 통해 Dispatcher는 수신 요청을 서버로 로드 밸런스합니다. 서버는 Dispatcher로 응답을 리턴합니다. 그런 다음, Dispatcher 시스템은 클라이언트로 응답을 리턴합니다.
- **Content Based Routing 전달 메소드(cbr)**. Caching Proxy 없이 Dispatcher 컴포넌트를 사용하여 HTTP("content" 유형 규칙 사용) 및 HTTPS(SSL 세션 ID 연관 관계 사용)에 대한 content-based routing을 수행할 수 있습니다. HTTP 및 HTTPS 통신량의 경우, Dispatcher 컴포넌트는 CBR 컴포넌트에 비해 신속한 content-based routing을 제공할 수 있습니다. cbr 전달 메소드를 통해 Dispatcher는 수신 요구를 서버로 로드 밸런스합니다. 서버는 Dispatcher로 응답을 리턴합니다. 그런 다음, Dispatcher 시스템은 클라이언트로 응답을 리턴합니다.

Dispatcher 컴포넌트는 대규모의 확장 가능한 서버 네트워크의 안전하고도 효율적인 관리를 위한 핵심요소입니다. Dispatcher를 사용하여, 하나의 가상 서버에 여러 대의 개별 서버를 연결할 수 있습니다. 따라서 사용자 사이트는 전세계에서 하나의 IP 주소로 표시됩니다. Dispatcher는 도메인 이름 서버와 독립적으로 기능하므로, 모든 요청은 Dispatcher 시스템의 IP 주소로 전송됩니다.

Dispatcher를 사용하면 클러스터된 서버로 통신량 로드를 밸런스하는 데 있어 탁월한 이점이 있으므로, 사용자의 사이트를 안정적이면서 효율적으로 관리할 수 있습니다.

Dispatcher로 로컬 서버 관리

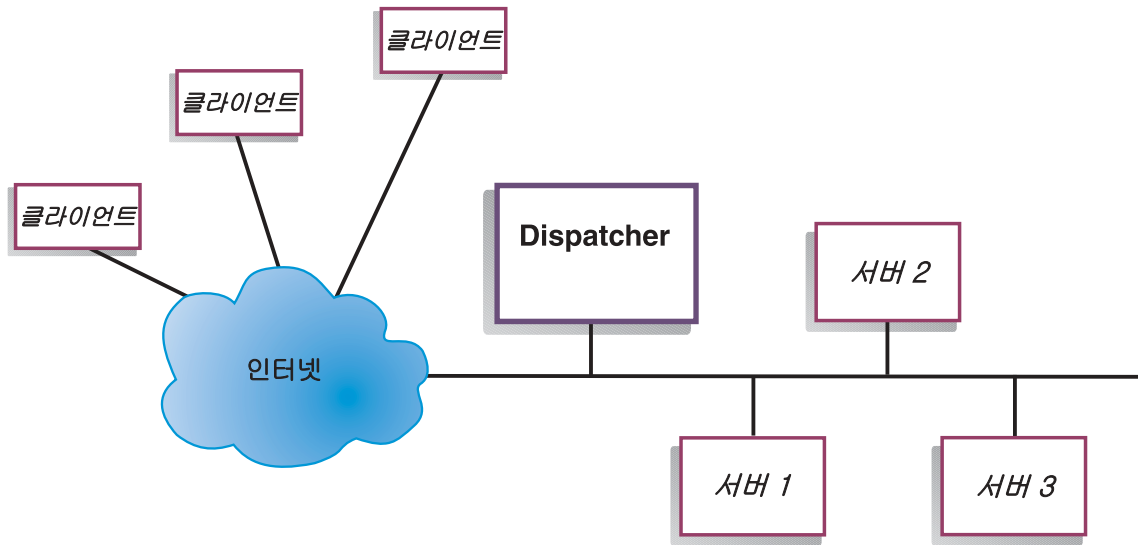


그림 1. Dispatcher를 사용하여 로컬 서버를 관리하는 사이트의 물리적 표현 예제

그림 1은 이더넷 네트워크 구성을 사용하는 사이트의 물리적 표현을 보여줍니다. Dispatcher 시스템은 네트워크를 물리적으로 변경하지 않고 설치할 수 있습니다. Dispatcher가 클라이언트 요청을 최적의 서버에 지정한 후 응답은 MAC 전달 메소드를 사용할 때 Dispatcher가 개입하지 않고 서버에서 클라이언트로 직접 전송됩니다.

Dispatcher 및 Metric Server를 사용하여 서버 관리

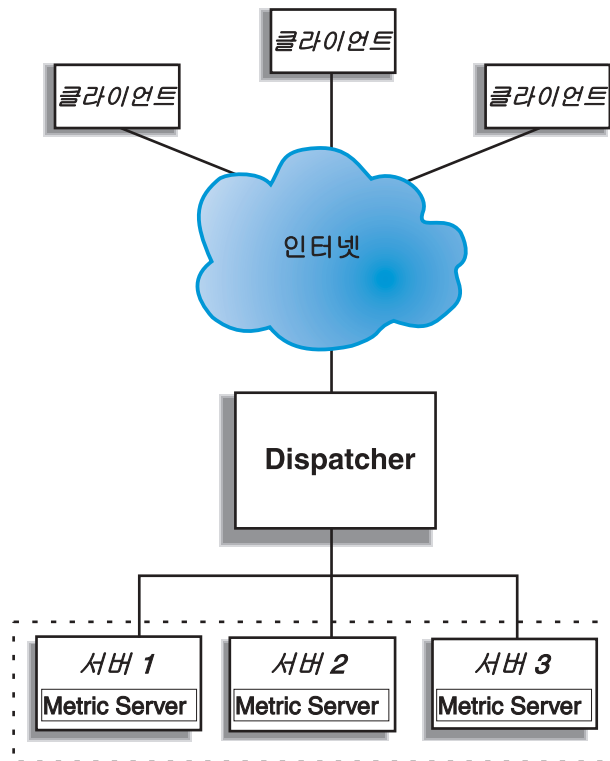


그림 2. Dispatcher 및 Metric Server를 사용하여 서버를 관리하는 사이트 예제

그림 2는 모든 서버가 로컬 네트워크에 있는 사이트를 보여줍니다. Dispatcher 컴포넌트는 요청을 전달하는 데 사용되며, Metric Server는 Dispatcher 시스템에 시스템 로드 정보를 제공하는 데 사용됩니다.

이 예제에서 Metric Server 디먼은 각 백엔드 서버에 설치됩니다. Metric Server를 Load Balancer 컴포넌트 또는 다른 Network Dispatcher 컴포넌트와 함께 사용할 수 있습니다.

Dispatcher로 로컬 및 원격 서버 관리

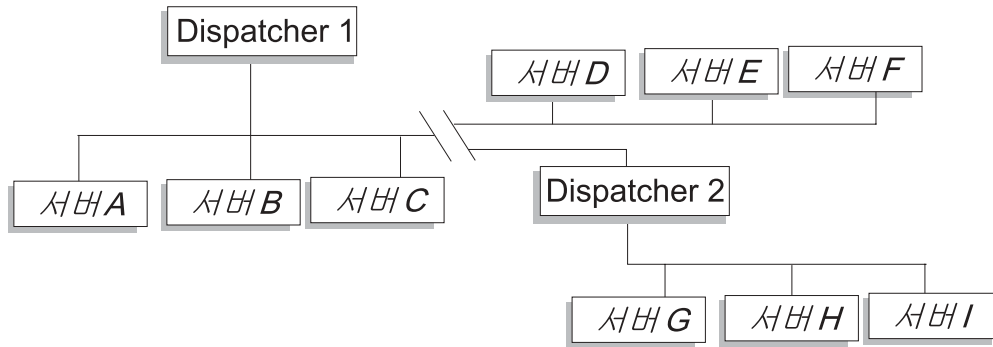


그림 3. Dispatcher를 사용하여 로컬 및 원격 서버를 관리하는 사이트 예제

Dispatcher에서 광역 지원을 사용하여 로컬 및 원격 서버(다른 서브넷에 있는 서버)를 모두 사용할 수 있습니다. 그림 3에는 하나의 "로컬" Dispatcher(Dispatcher 1)가 모든 요청의 시작점 역할을 하는 구성이 표시되어 있습니다. 이러한 요청을 고유 로컬 서버(서버 A, 서버 B, 서버 C) 및 원격 Dispatcher(Dispatcher 2)로 분배하여 로컬 서버(서버 G, 서버 H, 서버 I)로 로드 밸런싱합니다.

Dispatcher의 NAT 전달 메소드를 사용하거나 GRE 지원을 사용할 경우 원격 사이트(서버 D, 서버 E 및 서버 F가 있는 곳)에서 Dispatcher를 사용하지 않고 Dispatcher를 사용하는 광역 지원을 수행할 수도 있습니다. 자세한 정보는 59 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 메소드)』 및 250 페이지의 『GRE(일반 경로 지정 캡슐화) 지원』을 참조하십시오.

CBR(Content Based Routing) 컴포넌트 개요

CBR은 Caching Proxy에 대한 작업을 하여 클라이언트 요청을 지정된 HTTP 또는 HTTPS(SSL) 서버에 위임합니다. 이를 사용하여 낮은 네트워크 대역폭 요구사항으로 더 빠른 웹 문서 검색을 위해 캐시 세부사항을 조작할 수 있습니다. CBR 및 Caching Proxy는 지정된 규칙 유형을 사용하여 HTTP 요청을 검사합니다.

주: CBR(Content Based Routing) 컴포넌트는 64비트 JVM 실행 플랫폼을 제외한 모든 지원 플랫폼에서 사용 가능합니다. 혹은 Load Balancer에 있는 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 Caching Proxy를 사용하지 않고도 CBR을 제공할 수 있습니다. 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 페이지를 참조하십시오.

CBR을 사용하면 요청 콘텐츠와 일치하는 일반 표현에 기초하여 요청을 처리하는 서버 세트를 지정할 수 있습니다. CBR을 통해 각 유형의 요청에 대해 여러 서버를 지정할 수 있기 때문에, 최적의 클라이언트 응답을 위해 로드 밸런스를 조정할 수 있습니다. 또한 CBR은 한 세트의 한 서버가 고장난 시기를 감지하여 해당 서버로의 라우팅 요청을

정지합니다. CBR 컴포넌트가 사용하는 로드 밸런스 알고리즘은 Dispatcher 컴포넌트가 사용하는 입증된 알고리즘과 동일합니다.

Caching Proxy가 요청을 수신하면 CBR 컴포넌트에 정의된 규칙에 맞는지 확인합니다. 일치할 경우, 해당 규칙과 연관된 서버 중 하나가 선택되어 요청을 처리하게 됩니다. 그런 다음, Caching Proxy는 선택된 서버로 요청을 위임하는 표준 처리를 수행합니다.

CBR에는 고가용성, SNMP 서브에이전트, 광역 및 몇 가지 다른 구성 명령을 제외하고는 Dispatcher와 동일한 기능이 있습니다.

Caching Proxy는 CBR이 클라이언트 요청 로드 밸런스를 시작하기 전에 실행되고 있어야 합니다.

CBR로 로컬 서버 관리

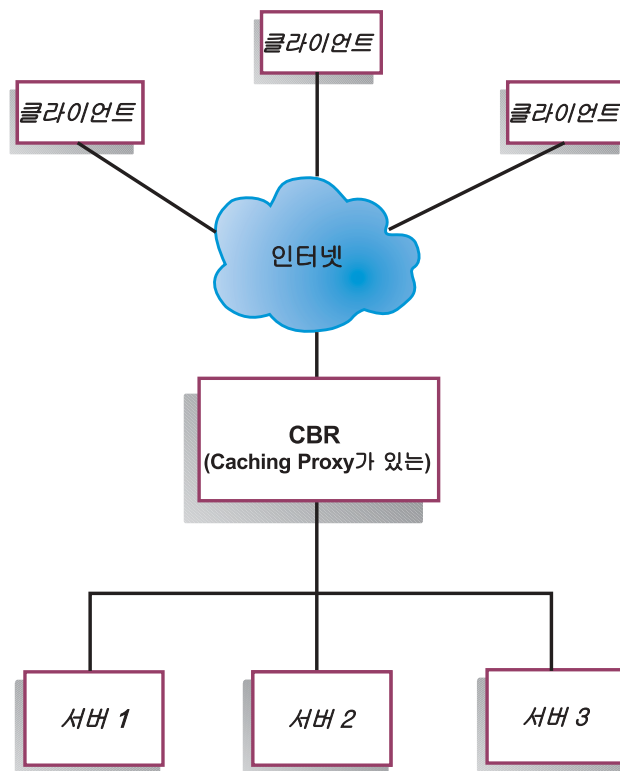


그림 4. CBR을 사용하여 로컬 서버를 관리하는 예제

그림 4에서는 CBR을 사용하여 로컬 서버에서 일부 콘텐츠를 위임하는 사이트의 논리적 표현을 보여 줍니다. CBR 컴포넌트는 Caching Proxy를 사용하여 URL 콘텐츠를 기반으로 서버로 클라이언트 요청(HTTP 또는 HTTPS)을 전달합니다.

Site Selector 컴포넌트 개요

Site Selector는 수집된 가중치와 측정값을 사용하여 서버 그룹 사이에서 로드 밸런스를 수행하기 위해 도메인 이름 시스템에 다른 이름 서버와 함께 작업하는 이름 서버 역할을 합니다. 클라이언트의 요청에 사용할 도메인 이름에 따라 서버 그룹 간 통신을 로드 밸런싱할 수 있도록 사이트 구성을 작성할 수 있습니다.

클라이언트는 도메인 이름에 대한 분석 요청을 네트워크 내의 이름 서버로 제출합니다. 이름 서버는 Site Selector 시스템으로 요청을 전달합니다. 그러면, Site Selector가 사이트 이름에 따라 구성된 서버 중 한 서버의 IP 주소로 도메인 이름을 분석합니다. Site Selector는 선택한 서버의 IP 주소를 이름 서버로 리턴합니다. 이름 서버는 IP 주소를 클라이언트로 리턴합니다.

Metric Server는 구성 내의 각 로드 밸런스 서버에 설치해야 하는 Load Balancer의 시스템 모니터링 컴포넌트입니다. Metric Server를 사용하여 Site Selector는 서버에서 활동 레벨을 모니터링하고 서버 부하가 가장 적은 시기를 감지하고 실패한 서버를 감지할 수 있습니다. 로드는 서버가 얼마나 어렵게 작동하는지에 대한 측정입니다. 시스템 메트릭 스크립트 파일을 사용자 정의하면 로드 측정에 사용되는 측정 유형을 제어할 수 있습니다. Site Selector는 액세스 빈도, 총 사용자 수, 액세스 유형(예: 간단한 조회, 장기 조회 또는 CPU 집중 로드)과 같은 요소를 고려하여 환경에 맞게 구성될 수 있습니다.

Site Selector 및 Metric Server를 사용하여 로컬 및 원격 서버 관리

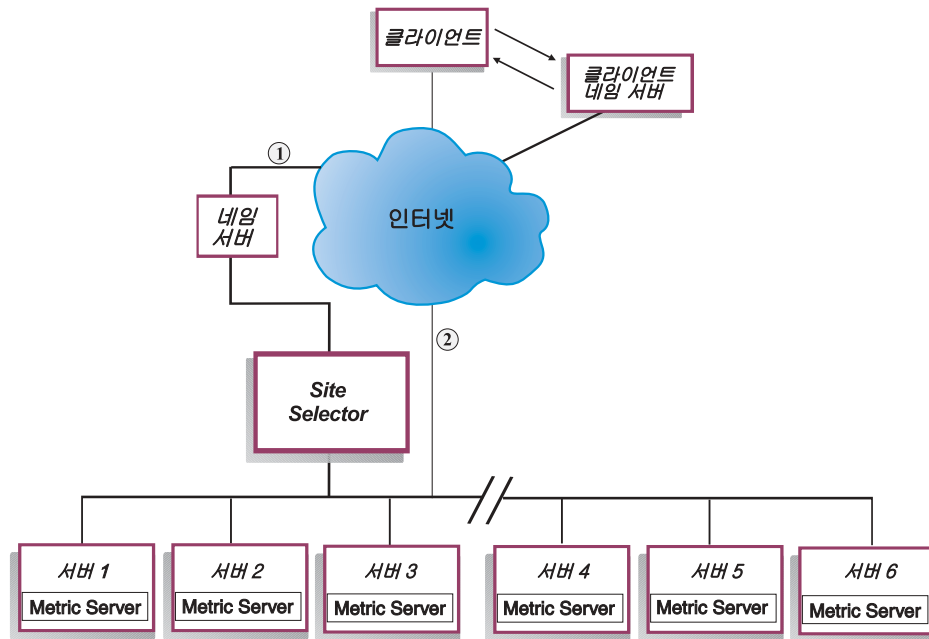


그림 5. Site Selector 및 Metric Server를 사용하여 로컬 및 원격 서버를 관리하는 사이트 예제

그림 5에는 Site Selector 컴포넌트를 사용하여 요청에 응답하는 사이트가 표시되어 있습니다. 서버 1, 서버 2 및 서버 3은 로컬 서버입니다. 서버 4, 서버 5 및 서버 6은 원격 서버입니다.

클라이언트는 도메인 이름 분석 요청을 클라이언트 이름 서버에 제출합니다. 클라이언트 이름 서버는 DNS를 통해 요청을 Site Selector 시스템으로 전달합니다(경로 1). Site Selector는 도메인 이름을 서버 중 하나의 IP 주소로 분석합니다. Site Selector는 선택된 서버의 IP 주소를 클라이언트 이름 서버에 리턴합니다. 이름 서버는 IP 주소를 클라이언트에 리턴합니다.

클라이언트가 서버의 IP 주소를 수신하면, 클라이언트는 응용프로그램 요청을 선택된 서버로 직접 라우트합니다(경로 2).

주: 이 예제에서 Metric Server는 Site Selector 시스템에 시스템 로드 정보를 제공합니다. Metric Server 에이전트는 각 백엔드 서버에 설치됩니다. Site Selector는 Metric Server와 연계하여 사용해야 합니다. 그렇지 않을 경우 Site Selector는 로드 밸런스를 위해 라운드 로빙 선택 방법만 사용할 수 있습니다.

Cisco CSS Controller 컴포넌트 개요

Cisco CSS Controller는 Cisco의 CSS 11000 Switch 시리즈와 함께 추가 보완 솔루션을 구성합니다. 결합된 솔루션은 CSS 11000 시리즈의 견고한 패킷 전달 메소드와 콘텐츠 경로 지정 기능을 Load Balancer의 정교한 인식 알고리즘과 혼합하여 서비스(백엔드 서버 응용프로그램 또는 데이터베이스)의 로드 정보와 가용성을 판별합니다. Cisco CSS Controller 기능은 Load Balancer의 가중치 계산 알고리즘, 표준 및 사용자 정의 어드바이저, Metric Server를 활용하여 메트릭, 상태 및 서비스의 로드를 판별합니다. 이 정보를 사용하여 Cisco CSS Controller는 서비스 가중치를 생성하여 최적 서비스 선택, 로드 최적화 및 결합 허용을 위해 Cisco CSS Switch로 전송합니다.

Cisco CSS Controller는 다음을 포함하여 다수의 기준을 추적합니다.

- 활성 연결 및 연결 비율(가중치 계산 주기 내의 새 연결 수)
- 표준 및 사용자 정의 어드바이저와 특정 응용프로그램에 맞게 조정된 서비스 상주 에이전트를 사용하여 편리해진 응용프로그램 및 데이터베이스 가용성
- CPU 활용
- 메모리 활용
- 사용자 조정 가능 시스템 메트릭

Cisco CSS Controller를 사용하지 않을 경우, Cisco CSS Switch는 콘텐츠 제공 서비스 상태를 판별할 때 콘텐츠 요청 또는 다른 네트워크 측정에 응답 시간을 사용합니다. Cisco CSS Controller를 사용하면 이런 활동이 Cisco CSS Switch에서 Cisco CSS Controller로 오프로드됩니다. Cisco CSS Controller는 콘텐츠 제공 기능 또는 서비스 가중치에 영향을 주며, 서비스가 가용성을 다시 확보하거나 유실할 때 적절히 서비스를 활성화시키거나 일시중단합니다.

Cisco CSS Controller:

- 공개된 SNMP 인터페이스를 사용하여 Cisco CSS Switch에서 연결 정보 확보
- 어드바이저 입력을 사용하여 서비스 사용가능성 및 응답 분석
- Metric Server 정보를 사용하여 시스템 로드 분석
- 구성에 있어 각 서비스에 가중치 생성

가중치는 포트의 모든 서비스에 적용됩니다. 특정 포트의 경우, 요청은 각각 상대적인 가중치에 따라 서비스 간에 분배됩니다. 예를 들어, 한 서비스의 가중치를 10으로 설정하고 다른 서비스의 가중치를 5로 설정하면 10으로 설정된 서비스의 요청 수는 5로 설정된 서비스 요청 수의 두 배가 됩니다. 이 가중치는 SNMP를 사용하여 Cisco CSS Switch에 제공됩니다. 서비스 가중치를 높게 설정할수록 Cisco CSS Switch가 해당 서비스로 더 많은 요청을 전달합니다.

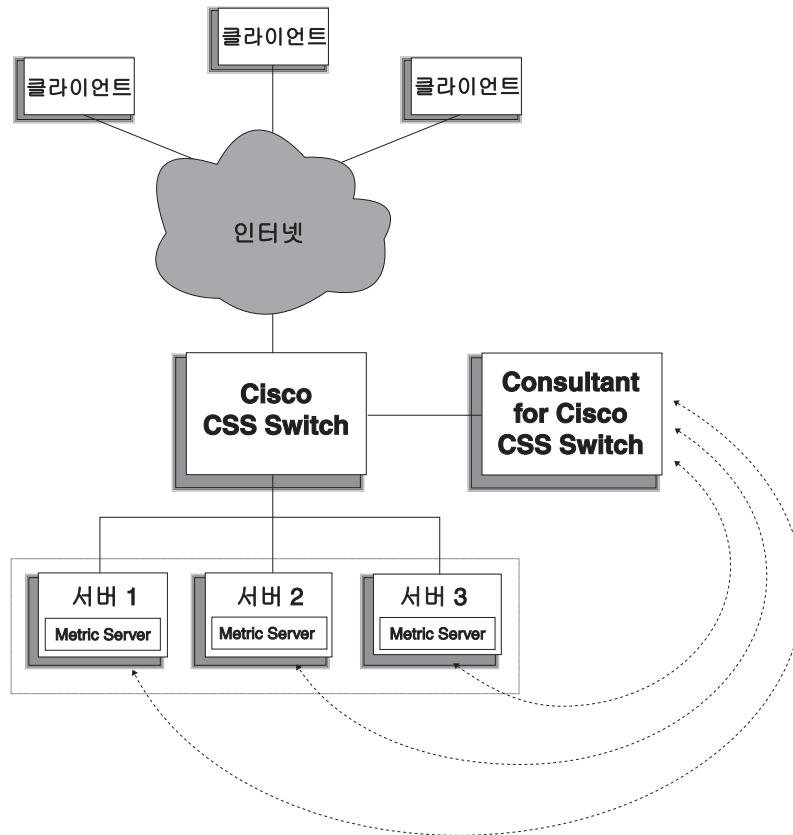


그림 6. Cisco CSS Controller 및 Metric Server를 사용하여 로컬 서비스를 관리하는 사이트 예제

Cisco CSS Switch와 함께 Cisco CSS Controller는 정교한 응용프로그램 인식으로 우선 속도의 콘텐츠 전환, 결합 허용 및 서비스 로드 최적화를 결합하는 "최적의" 솔루션을 제공합니다. Cisco CSS Controller는 Cisco CSS Switch와 IBM WebSphere Application Server Load Balancer 간의 전반적인 보완 솔루션의 일부입니다.

Nortel Alteon Controller 컴포넌트 개요

Web switch의 Nortel Alteon 그룹과 함께 Nortel Alteon Controller는 서버의 가중치를 판별하기 위해 스위치의 패킷 전달 속도 및 성능을 Load Balancer의 복잡한 인식 알고리즘 결합하는 보충 솔루션을 제공합니다.

Nortel Alteon Controller를 사용하여, 서비스 전개에 사용되는 응용프로그램의 가용성 및 로드제에 대한 보다 지능적인 응용프로그램 인지 평가를 수행할 수 있는 사용자 정의 어드바이저를 개발할 수 있습니다.

Metric Server는 CPU 및 메모리 사용 정보와 같은 시스템 로드 정보와 조정 시스템 로드 조치를 개발하는 프레임워크를 제공합니다.

Nortel Alteon Controller는 Nortel Alteon Web Switch가 로드 밸런스 중인 서버에 대한 가중치를 결정하기 위해 다음과 같은 여러 유형을 수집합니다.

- 활성 및 새 연결
- 표준 및 사용자 정의 어드바이저와 특정 응용프로그램에 맞게 조정된 서버 상주 에이전트를 사용하여 편리해진 응용프로그램 및 데이터베이스 가용성
- CPU 활용
- 메모리 활용
- 사용자 정의 가능 서버 메트릭
- 도달 가능성

Nortel Alteon Controller는 SNMP를 사용하여 스위치와 통신합니다. 스위치에서 구성, 상태 및 연결 정보를 검색합니다. 제어기가 서버 가중치를 계산하면 가중치는 스위치에 설정됩니다. 스위치는 제어기가 설정한 가중치를 사용하여 서비스에 대한 클라이언트 요청을 처리하기에 가장 적합한 서버를 선택합니다.

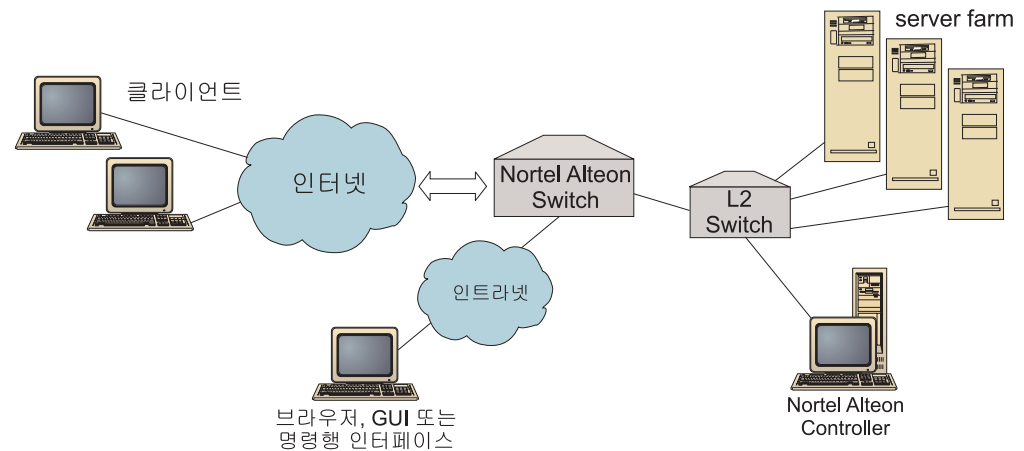


그림 7. Nortel Alteon Controller를 사용하여 로컬 서버를 관리하는 사이트 예제

브라우저, 원격 GUI 또는 원격 명령 인터페이스를 사용하여 제어기를 관리할 수 있습니다.

Nortel Alteon Controller는 Web Switch의 Nortel Alteon 그룹과 함께 유선 속도 패킷 전환을 복잡한 응용프로그램 인식, 결합 허용 및 서버 로드 최적화에 결합하는 "최적의" 솔루션을 제공합니다. Nortel Alteon Controller는 Web Switch의 Nortel Alteon 그룹 및 IBM WebSphere 간의 보충 솔루션의 일부입니다.

제 3 장 네트워크 관리: 사용할 Load Balancer 기능 결정

이 장에서는 네트워크 관리에 사용할 기능을 결정할 수 있도록 Load Balancer 컴포넌트의 구성 기능을 나열합니다.

- 『관리자, 어드바이저 및 Metric Server 기능(Dispatcher, CBR 및 Site Selector 컴포넌트)』
- 『Dispatcher 컴포넌트 기능』
- 26 페이지의 『CBR(Content Based Routing) 컴포넌트 기능』
- 28 페이지의 『Site Selector 컴포넌트 기능』
- 30 페이지의 『Cisco CSS Controller 컴포넌트 기능』
- 31 페이지의 『Nortel Alteon Controller 컴포넌트 기능』

중요: IPv4 및 IPv6용 Load Balancer를 사용하는 경우, Dispatcher 컴포넌트만이 사용 가능합니다. 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』에서 자세한 정보를 참조하십시오.

관리자, 어드바이저 및 Metric Server 기능(Dispatcher, CBR 및 Site Selector 컴포넌트)

여러 서버에 걸친 로드 밸런스를 최적화하고 "올바른" 서버가 선택되도록 보장하려면 다음을 참조하십시오.

- 194 페이지의 『Load Balancer에서 제공하는 로드 밸런스 최적화』
- 199 페이지의 『어드바이저』
- 211 페이지의 『Metric Server』

Dispatcher 컴포넌트 기능

Dispatcher는 HTTP, FTP, SSL, SMTP, NNTP, IMAP, POP3, Telnet, SIP 및 기타 다른 TCP 또는 독립적인 UDP 기반 응용프로그램에 대해 서버를 통한 로드 밸런스를 지원합니다.

원격 관리

- Load Balancer가 상주하는 시스템이 아닌 다른 시스템에서 Load Balancer 구성을 실행하려면 281 페이지의 『Load Balancer의 원격 관리』의 내용을 참조하십시오.

(IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, 이 기능을 사용할 수 없습니다.)

결합 배치

- 로드 밸런싱하는 웹 서버와 동일한 시스템에서 Dispatcher를 실행하려면 216 페이지의 『결합 배치된 서버 사용』의 내용을 참조하십시오.

고가용성

- Dispatcher를 사용하여 네트워크에서 단일 장애 지점 제한사항을 제거하려면 66 페이지의 『단순 고가용성』 및 67 페이지의 『상호 고가용성』의 내용을 참조하십시오.

(IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, 단순 고가용성 기능이 사용 가능하지만 상호 고가용성은 사용할 수 없습니다.)

클라이언트 대 서버 연관 관계

SSL(HTTPS) 통신량을 로드 밸런싱하는 경우:

- 클라이언트가 여러 접속에 대해 동일한 SSL 서버를 사용하도록 하려면 236 페이지의 『Load Balancer에 대한 연관 관계 기능 사용법』의 내용을 참조하십시오.
- 클라이언트가 HTTP 및 SSL 통신량에 대해 동일한 서버를 사용하도록 하려면 237 페이지의 『포트간 연관 관계』의 내용을 참조하십시오.

(IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, 포트간 유사성 기능을 사용할 수 없습니다.)

- 클라이언트가 여러 접속에 대해 동일한 서버를 사용하도록 하려면 236 페이지의 『Load Balancer에 대한 연관 관계 기능 사용법』의 내용을 참조하십시오.
- 클라이언트 그룹이 여러 접속에 대해 동일한 서버를 사용하도록 하려면 238 페이지의 『연관 관계 주소 마스크(stickymask)』의 내용을 참조하십시오.

(IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, 결합 마스크 기능을 사용할 수 없습니다.)

- 클라이언트 통신량을 방해하지 않고 구성에서 서버를 제거하려면(예: 유지보수 목적으로) 239 페이지의 『서버 연결 처리 작업중지』의 내용을 참조하십시오.

규칙 기반 로드 밸런스

동일한 웹 주소에 대해 서로 다른 서버 세트에 클라이언트를 지정하려면 Dispatcher 구성에 "규칙"을 추가해야 합니다. 자세한 내용은 226 페이지의 『규칙 기반 로드 밸런스 구성』을 참조하십시오.

- 클라이언트 소스 IP 주소에 기초하여 여러 서버 세트에 클라이언트를 지정하려면 228 페이지의 『클라이언트 IP 주소에 따라 규칙 사용』의 내용을 참조하십시오.
- 클라이언트 포트에 기초하여 서로 다른 서버 세트에 클라이언트를 지정하려면 228 페이지의 『클라이언트 포트에 따라 규칙 사용』의 내용을 참조하십시오.
- 시간에 기초하여 여러 서버 세트에 클라이언트를 지정하려면 229 페이지의 『시간에 따라 규칙 사용』의 내용을 참조하십시오.
- 네트워크 패킷의 TOS(Type of Service) 비트에 기초하여 서버에 클라이언트를 지정하려면 229 페이지의 『서비스 유형(TOS)에 기반하여 규칙 사용』의 내용을 참조하십시오.
- 사이트 통신량에 기초하여 여러 서버 세트에 클라이언트를 지정하려면 다음을 참조하십시오.
 - 초당 접속을 사용할 경우, 229 페이지의 『초당 연결에 따라 규칙 사용』의 내용을 참조하십시오.
 - 활성 연결 총계를 사용할 경우, 230 페이지의 『총 활성 연결에 따라 규칙 사용』의 내용을 참조하십시오.
 - 서로 다른 웹 주소에 대해 예약 및 공유 대역폭을 사용할 경우, 230 페이지의 『예약된 대역폭 및 공유 대역폭에 따라 규칙 사용』의 내용을 참조하십시오.
 - 각 서버 세트에 대해 통신량이 제대로 측정되도록 보장하려면 235 페이지의 『규칙에 대한 서버 평가 옵션』의 내용을 참조하십시오.
- 오버플로우 통신량을 기본 서버 세트(예: "사이트 사용 중"으로 응답할 서버)에 지정하려면 233 페이지의 『항상 참인 규칙 사용』의 내용을 참조하십시오.
- 클라이언트가 오버플로우 서버에 "결합"하지 않도록 하기 위해 클라이언트 연관 관계를 덮어쓰려면 234 페이지의 『포트 연관 관계 무시』의 내용을 참조하십시오.

IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, 규칙 기반 로드 밸런스를 사용할 수 없습니다.

Dispatcher의 cbr 전달 메소드를 사용한 Content Based Routing

클라이언트 요청의 SSL ID를 기준으로 SSL 클라이언트가 동일한 SSL 서버로 리턴되도록 하려면 다음을 수행하십시오.

- 62 페이지를 참조하십시오.

클라이언트 요청의 URL 콘텐츠 일치에 기초한 규칙을 사용하여 서로 다른 서버 세트에 HTTP 클라이언트를 지정하려면 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 및 234 페이지의 『요청 콘텐츠에 따라 규칙 사용』을 참조하십시오.

- 특정 URL과 그 서비스 응용프로그램을 구분하려면 64 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』의 내용을 참조하십시오.
- 웹 서버가 작성한 쿠키를 사용하여 여러 접속에서 비슷한 콘텐츠를 요청할 경우 동일한 서버로 클라이언트가 리턴하도록 하려면 242 페이지의 『수동 쿠키 연관 관계』의 내용을 참조하십시오.
- 각 서버에 고유한 콘텐츠가 캐시되도록 하는 Caching Proxy 서버에 웹 통신량을 로드 밸런싱하려면(따라서 여러시스템에서 중복되는 콘텐츠 캐시를 제거하여 사이트의 캐시 크기가 증가) 243 페이지의 『URI 연관 관계』의 내용을 참조하십시오.

(IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, Dispatcher의 cbr 전달 메소드를 사용할 수 없습니다.)

Dispatcher 컴포넌트의 cbr 전달 메소드 및 CBR 컴포넌트 간 비교

Dispatcher의 cbr 전달 메소드 사용의 장점은 클라이언트 요청에 CBR 컴포넌트보다 더 빠른 응답을 제공하는 것입니다. 또한 Dispatcher의 cbr 전달은 Caching Proxy 설치 및 사용을 요구하지 않습니다.

네트워크에 완전히 보안된 SSL(서버를 통한 클라이언트) 통신량이 필요할 경우, CBR 컴포넌트(Caching Proxy와 결합하여) 사용의 장점은 Content Based Routing을 수행하기 위해 필요한 암호화 및 암호 해독을 처리할 수 있다는 것입니다. 완전히 보안된 접속을 위해 SSL ID 연관 관계로 Dispatcher의 cbr 전달을 구성할 수 있습니다. 클라이언트 요청의 URL에 대해 진정한 Content Based Routing을 수행하기 위한 암호화 및 암호 해독을 처리할 수 없기 때문입니다.

광역 로드 밸런스

여러 가지 방법을 사용하여 광역 로드 밸런스를 달성할 수 있습니다.

- Dispatcher의 광역 기능을 사용하여 원격 서버로 로드 밸런싱하려면 244 페이지의 『광역 Dispatcher 지원 구성』 및 250 페이지의 『GRE(일반 경로 지정 캡슐화) 지원』의 내용을 참조하십시오.

주: 원격 사이트에서 GRE가 지원되지 않을 경우 원격 사이트에서 추가 Dispatcher가 필요합니다.

- Dispatcher의 nat 전달 메소드를 사용하여 원격 서버로 로드 밸런싱하려면 59 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 메소드)』의 내용을 참조하십시오.

주: nat 전달 메소드를 사용할 경우에는 원격 사이트에 추가 Dispatcher가 필요하지 않습니다.

(IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, 광역 로드 밸런스 기능을 사용할 수 없습니다.)

포트 매핑

- 동일한 시스템의 여러 서버 디면에 한 웹 주소를 로드 밸런스하도록 하려면(각 디면이 고유 포트를 인식) 59 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 메소드)』의 내용을 참조하십시오.

(IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, 이 기능을 사용할 수 없습니다.)

개인용 네트워크에 Dispatcher 설정

- 클라이언트 통신량과 다른 네트워크에 Dispatcher 통신량을 배치하려면(외부 네트워크의 회선 경합을 줄여 성능을 향상시키기 위해) 252 페이지의 『개인용 네트워크 구성 사용』의 내용을 참조하십시오.

와일드 카드 클러스터 및 와일드 카드 포트

- 여러 웹 주소를 단일 구성으로 결합하려면 253 페이지의 『와일드 카드 클러스터를 사용하여 서버 구성 조합』의 내용을 참조하십시오.
- 밸런스 방화벽을 로드하려면 254 페이지의 『와일드 카드 클러스터를 사용하여 방화벽 로드 밸런스 수행』을 참조하십시오.
- 모든 대상 포트에 대해 통신량을 지정하려면 255 페이지의 『와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정』의 내용을 참조하십시오.

"서비스 거부" 중지 감지

- 가능한 "서비스 거부" 공격을 발견하려면 255 페이지의 『서비스 거부 중지 감지』의 내용을 참조하십시오.

2진 로그

- 서버 통신량을 분석하려면 257 페이지의 『서버 통제를 분석하기 위해 2진 로그 사용』을 참조하십시오.

경보

- 서버가 연결 표시 또는 작동 중단 표시될 때 경보를 생성하려면 198 페이지의 『스크립트를 사용하여 경보나 레코드 서버 장애 생성』의 내용을 참조하십시오.

CBR(Content Based Routing) 컴포넌트 기능

CBR에서는 로드 밸런싱을 WebSphere Application Server의 Caching Proxy와 통합하여 클라이언트 요청을 지정한 HTTP 또는 HTTPS(SSL) 서버에 위임합니다. CBR을 사용하려면 Caching Proxy를 동일한 서버에 설치하고 구성해야 합니다. CBR을 사용하기 위해 Caching Proxy를 구성하는 방법에 관한 정보는 128 페이지의 『1단계. CBR을 사용하기 위해 Caching Proxy 구성』의 내용을 참조하십시오.

주: CBR(Content Based Routing) 컴포넌트는 64비트 JVM 실행 플랫폼을 제외한 모든 지원 플랫폼에서 사용 가능합니다. 혹은 Load Balancer에 있는 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 Caching Proxy를 사용하지 않고도 CBR을 제공할 수 있습니다. 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 페이지를 참조하십시오.

CBR 컴포넌트(또는 Dispatcher 컴포넌트의 cbr 전달 메소드)로, 클라이언트에 다음과 같은 장점을 제공할 수 있습니다.

- 여러 유형의 콘텐츠에 대한 클라이언트 요청을 서버 세트에 로드 밸런싱합니다. (118 페이지의 『다른 유형의 콘텐츠에 대한 요청 로드 밸런싱』 참조).
- 웹 서버의 사이트 콘텐츠를 최적으로 나누어 응답 시간을 개선합니다. (118 페이지의 『더 나은 응답 시간을 위해 사이트의 콘텐츠 나누기』 참조).
- 각 유형의 콘텐츠에 여러 서버를 지정할 수 있어서 서버 장애 중에 클라이언트 통신량이 방해받지 않습니다. (119 페이지의 『웹 서버 콘텐츠의 백업 제공』 참조).

CBR 컴포넌트와 Dispatcher 컴포넌트의 cbr 전달 메소드 간의 비교

네트워크에서 완전히 보안된 SSL 통신량(서버를 통한 클라이언트)을 요구할 경우, CBR 컴포넌트(Caching Proxy와 결합하여) 사용의 장점은 Content Based Routing을 수행하기 위해 SSL 암호화/암호 해독을 처리할 수 있다는 것입니다.

완전히 보안된 SSL 접속을 위해 SSL ID 연관 관계로 Dispatcher의 cbr 전달을 구성할 수 있습니다. 클라이언트 요청의 URL에 대해 진정한 Content Based Routing을 수행하기 위한 암호화/암호 해독을 처리할 수 없기 때문입니다.

HTTP 통신량의 경우, Dispatcher의 cbr 전달 메소드 사용의 장점은 클라이언트 요청에 CBR 컴포넌트보다 더 빠른 응답을 제공하는 것입니다. 또한 Dispatcher의 cbr 전달은 Caching Proxy 설치 및 사용을 요구하지 않습니다.

원격 관리

- Load Balancer가 상주하는 시스템이 아닌 다른 시스템에서 Load Balancer 구성을 실행하려면 281 페이지의 『Load Balancer의 원격 관리』의 내용을 참조하십시오.

결합 배치

- CBR은 로드 밸런스 중인 서버와 동일한 시스템에서 실행할 수 있습니다. 216 페이지의 『결합 배치된 서버 사용』에서 자세한 정보를 참조하십시오.

Caching Proxy의 복수 인스턴스에 대한 CBR

- 복수 Caching Proxy 프로세스를 사용하여 CPU 이용을 향상시키려면 119 페이지의 『CPU 이용 향상을 위해 복수 Caching Proxy 프로세스 사용』의 내용을 참조하십시오.

SSL 접속을 위한 Content Based Routing 제공

SSL 통신량의 Content Based Routing을 허용하려면 다음과 같이 수행하십시오.

- 양쪽에서 보안 연결을 사용하려면(클라이언트 대 프록시 및 프록시 대 서버) 120 페이지의 『완전 보안(SSL) 연결의 로드 밸런스』의 내용을 참조하십시오.
- 클라이언트 대 프록시 쪽에서만 보안 연결을 사용하려면 120 페이지의 『SSL의 클라이언트-투-프록시 및 HTTP의 프록시-투-서버의 로드 밸런스』의 내용을 참조하십시오.

서버 파티션

- 특정 URL과 그 서비스 응용프로그램을 구분하려면 64 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』의 내용을 참조하십시오.

규칙 기반 로드 밸런스

동일한 웹 주소에 대해 여러 서버 세트에 클라이언트를 지정하기 위해 CBR 구성에 "규칙"을 추가할 수 있습니다. 자세한 내용은 226 페이지의 『규칙 기반 로드 밸런스 구성』을 참조하십시오.

- 요청된 RUL의 콘텐츠에 기초하여 여러 서버 세트에 클라이언트를 지정하려면 234 페이지의 『요청 콘텐츠에 따라 규칙 사용』의 내용을 참조하십시오.
- 클라이언트 소스 IP 주소에 기초하여 여러 서버 세트에 클라이언트를 지정하려면 228 페이지의 『클라이언트 IP 주소에 따라 규칙 사용』의 내용을 참조하십시오.
- 시간에 기초하여 여러 서버 세트에 클라이언트를 지정하려면 229 페이지의 『시간에 따라 규칙 사용』의 내용을 참조하십시오.
- 사이트 통신량에 기초하여 여러 서버 세트에 클라이언트를 지정하려면 다음을 참조하십시오.

초당 접속을 사용할 경우, 229 페이지의 『초당 연결에 따라 규칙 사용』의 내용을 참조하십시오.

활성 연결 총계를 사용할 경우, 230 페이지의 『총 활성 연결에 따라 규칙 사용』의 내용을 참조하십시오.

- 오버플로우 통신량을 기본 서버 세트(예: "사이트 사용 중"으로 응답할 서버)에 지정하려면 233 페이지의 『항상 참인 규칙 사용』의 내용을 참조하십시오.
- 클라이언트가 오버플로우 서버에 "결합"하지 않도록 하기 위해 클라이언트 연관 관계를 덮어쓰려면 234 페이지의 『포트 연관 관계 무시』의 내용을 참조하십시오.

클라이언트 대 서버 연관 관계

- 클라이언트가 여러 연결에 대해 동일한 서버로 리턴하도록 하려면 236 페이지의 『Load Balancer에 대한 연관 관계 기능 사용법』의 내용을 참조하십시오.
- 클라이언트 통신량을 방해하지 않고 구성에서 서버를 제거하려면(예: 유지보수 목적으로) 239 페이지의 『서버 연결 처리 작업중지』의 내용을 참조하십시오.
- 웹 서버가 작성한 쿠키에 의하지 않고 여러 접속에서 비슷한 콘텐츠를 요청할 경우, 동일한 서버로 클라이언트가 리턴하도록 하려면 240 페이지의 『활성 쿠키 연관 관계』의 내용을 참조하십시오.
- 웹 서버가 작성한 쿠키를 사용하여 여러 접속에서 비슷한 콘텐츠를 요청할 경우, 동일한 서버로 클라이언트가 리턴하도록 하려면 242 페이지의 『수동 쿠키 연관 관계』의 내용을 참조하십시오.
- 각 서버에 고유한 콘텐츠가 캐시되도록 하는 Caching Proxy 서버에 웹 통신량을 로드 밸런싱하려면(따라서 여러시스템에서 중복되는 콘텐츠 캐시를 제거하여 사이트의 캐시 크기가 증가) 243 페이지의 『URI 연관 관계』의 내용을 참조하십시오.

Dispatcher 및 CBR을 사용한 고가용성

- CBR이 있는 2 티어 구성에서 Dispatcher를 사용하여 네트워크의 단일 지점의 장애 제한사항을 제거하려면 6 페이지의 『Load Balancer가 고가용성을 제공하는 방법』의 내용을 참조하십시오.

2진 로그

- 서버 통신량을 분석하려면 257 페이지의 『서버 통제를 분석하기 위해 2진 로그 사용』을 참조하십시오.

경보

- 서버가 연결 표시 또는 작동 중단 표시될 때 경보를 생성하려면 198 페이지의 『스크립트를 사용하여 경보나 레코드 서버 장애 생성』의 내용을 참조하십시오.

Site Selector 컴포넌트 기능

Site Selector는 서버 그룹에 걸쳐 이름 서비스 요청을 로드 밸런싱합니다.

원격 관리

- Load Balancer가 상주하는 시스템이 아닌 다른 시스템에서 Load Balancer 구성을 실행하려면 281 페이지의 『Load Balancer의 원격 관리』의 내용을 참조하십시오.

결합 배치

- Site Selector는 추가 구성 단계없이 로드 밸런싱하는 서버와 동일한 시스템에서 실행될 수 있습니다.

고가용성

- 고가용성은 상위 이름 서버가 제대로 구성되어 있고 정상 DNS 복구 방법이 적절하다는 가정 하에, 여러 중복 Site Selector를 사용하여 DNS(Domain Name System) 방법을 통해 본래부터 사용 가능합니다. 정상 DNS 복구 방법의 예는 조회의 재전송 및 영역 전송의 재시도입니다.
- Site Selector가 있는 2 티어 구성에서 Dispatcher를 사용하여 네트워크의 단일 지점의 장애 제한사항을 제거하려면 6 페이지의 『Load Balancer가 고가용성을 제공하는 방법』의 내용을 참조하십시오.

클라이언트 대 서버 연관 관계

- 클라이언트가 여러 이름 서버에 대해 동일한 서버를 사용하도록 하려면 236 페이지의 『Load Balancer에 대한 연관 관계 기능 사용법』의 내용을 참조하십시오.
- TTL(Time To Live) 설정의 표준 DNS 방법을 사용하여 클라이언트 대 서버 연관 관계를 보장하려면 143 페이지의 『TTL 고려사항』의 내용을 참조하십시오.

규칙 기반 로드 밸런스

도메인 이름 분석을 위해 여러 서버 세트에 클라이언트 요청을 지정하려면 Site Selector 구성에 "규칙"을 추가해야 합니다. 자세한 내용은 226 페이지의 『규칙 기반 로드 밸런스 구성』을 참조하십시오.

- 클라이언트 소스 IP 주소에 기초하여 여러 서버 세트에 클라이언트를 지정하려면 228 페이지의 『클라이언트 IP 주소에 따라 규칙 사용』의 내용을 참조하십시오.
- 시간에 기초하여 여러 서버 세트에 클라이언트를 지정하려면 229 페이지의 『시간에 따라 규칙 사용』의 내용을 참조하십시오.
- 서버 세트의 매트릭 로드 값에 기초하여 여러 서버 세트에 클라이언트를 지정하려면 다음을 참조하십시오.

232 페이지의 『Metric all 규칙』

233 페이지의 『Metric average 규칙』

- 오버플로우 통신량을 기본 서버 세트(예: "사이트 사용 중"으로 응답할 서버)에 지정하려면 233 페이지의 『항상 참인 규칙 사용』의 내용을 참조하십시오.

광역 로드 밸런스

Site Selector는 LAN(Local Area Network) 또는 WAN(Wide Area Network)에서 실행될 수 있습니다.

WAN 환경에서:

- 가중치가 있는 무순위 선택 방법을 사용하여 클라이언트 이름 서버 요청을 로드 밸런스하려면 추가 구성 단계가 필요하지 않습니다.
- 요청된 응용프로그램(대상 서버)을 제공하는 서버에 대한 클라이언트 이름 서버의 네트워크 근접성을 고려하려면 144 페이지의 『네트워크 근접 기능 사용』의 내용을 참조하십시오.

경보

- 서버가 연결 표시 또는 작동 중단 표시될 때 경보를 생성하려면 198 페이지의 『스크립트를 사용하여 경보나 레코드 서버 장애 생성』의 내용을 참조하십시오.

Cisco CSS Controller 컴포넌트 기능

Cisco CSS Controller의 큰 응용프로그램 및 시스템 인식에 대한 Cisco switch의 서버 로드 밸런스 성능이 향상되었습니다. 제어기는 더 많은 응용프로그램 감도 및 시스템 감도 메트릭을 사용하여 서버 가중치를 동적으로 계산합니다. SNMP를 사용하여 스위치에 가중치가 제공됩니다. 스위치는 클라이언트 요청을 처리할 때 가중치를 사용하여 서버 로드를 최적화하고 결합 허용을 개선합니다.

여러 서버에 걸친 로드 밸런스를 최적화하고 "올바른" 서버가 선택되도록 보장하려면 다음을 참조하십시오.

- 264 페이지의 『Load Balancer에서 제공하는 로드 밸런스 최적화』
- 266 페이지의 『어드바이저』 및 268 페이지의 『사용자 정의(사용자 정의 가능) 어드바이저 작성』
- 272 페이지의 『Metric Server』

원격 관리

- Load Balancer가 상주하는 시스템이 아닌 다른 시스템에서 Load Balancer 구성을 실행하려면 281 페이지의 『Load Balancer의 원격 관리』의 내용을 참조하십시오.

결합 배치

- Cisco CSS Controller는 추가 구성 단계없이 로드 밸런싱하는 서버와 동일한 시스템에서 실행될 수 있습니다.

고가용성

- 네트워크에서 단일 지점의 장애 제한사항을 제거하려면 Cisco CSS Switch 및 Cisco CSS Controller에 고가용성 성능이 있어야 합니다. 스위치의 경우, CSS 중복 프로토콜을 사용하여 고가용성 성능이 가능합니다. Cisco CSS Controller의 경우, 두 제어기의 항상 대기 구성이 가능한 독점 프로토콜이 사용됩니다.

고가용성 구성에 대한 자세한 정보는 160 페이지의 『고가용성』을 참조하십시오.

2진 로그

- 서버 통신량을 분석하려면 275 페이지의 『서버 통계를 분석하기 위해 2진 로그 사용』을 참조하십시오.

경보

- 서버가 연결 표시 또는 작동 중지 표시될 때 경보를 생성하려면 276 페이지의 『스크립트를 사용하여 경보나 레코드 서버 장애 생성』의 내용을 참조하십시오.

Nortel Alteon Controller 컴포넌트 기능

Nortel Alteon Controller의 큰 응용프로그램 및 시스템 인식에 대한 Nortel switch의 서버 로드 밸런싱 성능이 향상되었습니다. 제어기는 더 많은 응용프로그램 감도 및 시스템 감도 메트릭을 사용하여 서버 가중치를 동적으로 계산합니다. SNMP를 사용하여 스위치에 가중치가 제공됩니다. 스위치는 클라이언트 요청을 처리할 때 가중치를 사용하여 서버 로드를 최적화하고 결합 허용을 개선합니다.

여러 서버에 걸친 로드 밸런싱을 최적화하고 "올바른" 서버가 선택되도록 보장하려면 다음을 참조하십시오.

- 264 페이지의 『Load Balancer에서 제공하는 로드 밸런싱 최적화』
- 266 페이지의 『어드바이저』 및 268 페이지의 『사용자 정의(사용자 정의 기능) 어드바이저 작성』
- 272 페이지의 『Metric Server』

원격 관리

- Load Balancer가 상주하는 시스템이 아닌 다른 시스템에서 Load Balancer 구성을 실행하려면 281 페이지의 『Load Balancer의 원격 관리』의 내용을 참조하십시오.

결합 배치

- Nortel Alteon Controller는 추가 구성 단계없이 로드 밸런싱하는 서버와 동일한 시스템에서 실행될 수 있습니다.

고가용성

- 네트워크에서 단일 지점 장애 제한사항을 제거하려면 Nortel Alteon Web Switch 및 Nortel Alteon Controller에 고가용성 성능이 있어야 합니다. 스위치의 경우, 서버에 대한 연결 또는 서비스에 대해 중복 프로토콜을 사용하여 고가용성 성능이 가능합니다. Nortel Alteon 제어기는 두 제어기의 긴급 대기 구성이 가능한 비율 프로토콜을 사용하여 고가용성을 제공합니다.

고가용성 구성에 대한 자세한 정보는 181 페이지의 『고가용성』을 참조하십시오.

2진 로그

- 서버 통신량을 분석하려면 275 페이지의 『서버 통제를 분석하기 위해 2진 로그 사용』을 참조하십시오.

경보

- 서버가 연결 표시 또는 작동 중지 표시될 때 경보를 생성하려면 276 페이지의 『스크립트를 사용하여 경보나 레코드 서버 장애 생성』의 내용을 참조하십시오.

제 4 장 Load Balancer 설치

이 장에서는 모든 지원 운영 체제의 시스템 패키징 도구 및 요구사항을 사용한 Load Balancer 설치에 대해 설명합니다.

- 34 페이지의 『AIX 시스템 요구사항 및 설치』
- 38 페이지의 『HP-UX 시스템 요구사항 및 설치』
- 40 페이지의 『Linux 시스템 요구사항 및 설치』
- 42 페이지의 『Solaris 시스템 요구사항 및 설치』
- 44 페이지의 『Windows 시스템 요구사항 및 설치』

제품 설치 프로그램을 사용하는 설치 지시사항에 대해서는 *Edge Components*용 개념, 계획 및 설치 문서를 참조하십시오.

Java 2 SDK는 Load Balancer와 함께 모든 플랫폼에 자동으로 설치됩니다.

Load Balancer의 이전 버전에서 이주하는 경우 또는 운영 체제 재설치의 경우, 설치 전에 Load Balancer의 스크립트 파일 또는 이전 구성 파일을 저장할 수 있습니다.

- 설치 후에 `.../ibm/edge/lb/servers/configurations/component` 디렉토리에 구성 파일을 배치하십시오.(여기서 *component*는 dispatcher, cbr, ss, cco 또는 nali입니다.)
- 설치 후에 실행하기 위해 `.../ibm/edge/lb/servers/bin` 디렉토리에 스크립트 파일(예: goIdle 및 goStandby)을 배치하십시오.

이 섹션에 나열된 모든 Load Balancer 컴포넌트 패키지가 제공되는 것은 아니며, 설치 유형에 따라 다릅니다.

- Load Balancer 및 Caching Proxy 모두를 제공할 수 있는 Edge Component 설치인 경우, 모든 Load Balancer 설치 컴포넌트의 사용이 가능합니다.
- Load Balancer는 제공할 수 있지만 Caching Proxy는 제공할 수 없는 Edge Component 설치인 경우, CBR 컴포넌트 패키지는 Load Balancer와 함께 포함되지 않습니다.
- IPv6 설치를 위한 Edge Component(IPv4 및 IPv6용 Load Balancer)의 경우, Dispatcher 컴포넌트 패키지는 Load Balancer와 함께 포함됩니다. CBR, Site Selector 및 제어기 컴포넌트는 포함되지 않습니다. IPv4 및 IPv6용 Load Balancer 패키지의 권장 설치 순서에 대해서는 91 페이지의 『IPv4 및 IPv6용 Load Balancer 설치』 페이지를 참조하십시오.

AIX 시스템 요구사항 및 설치

AIX 시스템의 요구사항

지원되는 브라우저를 포함하는 하드웨어 및 소프트웨어 요구사항에 대한 정보는 다음 웹 페이지 <http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921> 을 참조하십시오.

AIX 시스템용 설치

표 1 Load Balancer에 대한 installp 이미지 및 시스템 패키지 설치 도구를 사용한 권장 설치 순서를 나열합니다.

표 1. AIX installp 이미지

기본	ibmlb.base.rte
Administration(메시지)	<ul style="list-style-type: none">• ibmlb.admin.rte• ibmlb.msg.language.admin
장치 드라이버	ibmlb.lb.driver
License	ibmlb.lb.license
Load Balancer 컴포넌트(메시지)	<ul style="list-style-type: none">• ibmlb.component.rte• ibmlb.msg.language.lb
문서(메시지)	<ul style="list-style-type: none">• ibmlb.doc.rte• ibmlb.msg.en_US.doc
Metric Server	ibmlb.ms.rte

여기서 컴포넌트는 disp(Dispatcher), cbr(CBR), ss(Site Selector), cco(Cisco CSS Controller) 또는 nal(Nortel Alteon Controller)입니다. 설치하고자 하는 컴포넌트를 선택적으로 선택하십시오.

여기서 언어는 다음과 같습니다.

- en_US
- de_CH
- de_DE
- es_ES
- fr_CA
- fr_CH
- fr_FR
- it_CH
- it_IT
- ja_JP

- Ja_JP
- ko_KR
- pt_BR
- zh_CN
- ZH_CN
- zh_TW
- Zh_TW

문서 패키지는 영어로만 이루어져 있습니다. Load Balancer 문서 세트의 번역은 다음 웹 사이트에 있습니다. www.ibm.com/software/webserver/appserv/ecinfocenter.html 입니다.

설치하기 전에

이미 이전 버전이 설치되어 있으면, 현재 버전을 설치하기 전에 이전 버전의 설치를 제거하십시오. 먼저 모든 실행 프로그램과 모든 서버가 정지되었는지 확인하십시오. 그런 다음, 전체 제품을 설치 제거하려면 **installp -u ibmlb**(또는 이전 이름(예: **intnd**))를 입력하십시오. 특정 파일 세트를 설치 제거하려면, 패키지 이름을 지정하지 말고 명확하게 해당 파일 세트를 나열하십시오.

제품을 설치할 때, 다음 항목 중에 하나 또는 모두를 설치하기 위한 옵션이 제공됩니다.

- 기본
- Administration(메시지)
- 장치 드라이버(필수)
- 사용권(필수)
- Dispatcher 컴포넌트(메시지)
- CBR 컴포넌트(메시지)
- Site Selector 컴포넌트(메시지)
- Cisco CSS Controller 컴포넌트(메시지)
- Nortel Alteon Controller 컴포넌트(메시지)
- 문서(메시지)
- Metric Server

설치 단계

AIX 시스템용 Load Balancer를 설치하려면 다음 단계를 수행하십시오.

1. 루트로 로그인하십시오.
2. 제품 매체를 넣거나 웹에서 설치할 경우 설치 이미지를 디렉토리에 복사하십시오.

3. 설치 이미지를 설치하십시오. SMIT는 모든 메시지가 자동으로 설치되도록 하므로, SMIT를 사용하여 AIX용 Load Balancer를 설치하십시오.

SMIT 사용:

선택 소프트웨어 설치 및 관리

선택 소프트웨어 설치 및 갱신

선택 사용 가능한 최신 소프트웨어에서 설치 및 갱신

입력 installp 이미지를 포함하는 장치 및 디렉토리

입력 설치 행의 *SOFTWARE에 옵션을 지정할(또는 목록을 선택할) 적절한 정보

누름 확인

명령이 완료되면, 완료를 누른 후 종료 메뉴에서 **Smit** 종료를 선택하거나 **F12**를 누르십시오. SMITTY를 사용 중이면, **F10**을 눌러 프로그램을 종료하십시오.

명령행 사용:

CD에서 설치할 경우, CD를 마운트하려면 다음 명령을 입력해야 합니다.

```
mkdir /cdrom
mount -v cdrfs -p -r /dev/cd0 /cdrom
```

원하는 AIX 시스템용 Load Balancer 패키지를 설치하기 위해 입력할 명령을 판별하려면 다음 테이블을 참조하십시오.

표 2. AIX 설치 명령

기본	installp -acXgd device ibmlb.base.rte
Administration(메시지)	installp -acXgd device ibmlb.admin.rte ibmlb.msg.language.admin
장치 드라이버	installp -acXgd device ibmlb.lb.driver
License	installp -acXgd device ibmlb.lb.license
Load Balancer 컴포넌트(메시지). Dispatcher, CBR, Site Selector, Cisco CSS Controller 및 Nortel Alteon Controller가 포함됩니다.	installp -acXgd device ibmlb.component.rte ibmlb.msg.language.lb
문서(메시지)	installp -acXgd device ibmlb.doc.rte ibmlb.msg.en_US.lb
Metric Server	installp -acXgd device ibmlb.ms.rte

여기서 device는 다음과 같습니다.

- CD에서 설치할 경우 /cdrom.
- 파일 시스템에서 설치할 경우 /dir(installp 이미지를 포함하는 디렉토리)

요약의 결과 열에 설치할 Load Balancer의 각 파트마다 SUCCESS가 표시되어 있는지 확인하십시오(APPLYing). 설치할 모든 파트가 적용될 때까지 진행하지 마십시오.

주: installp 이미지에서 사용 가능한 모든 메시지 카탈로그를 포함하여 파일 세트 목록을 작성하려면, 다음을 입력하십시오.

```
installp -ld device
```

여기서 *device*는 다음과 같습니다.

- CD에서 설치할 경우 /cdrom.
- 파일 시스템에서 설치할 경우 /dir(installp 이미지를 포함하는 디렉토리)

CD를 마운트 해제하려면, 다음을 입력하십시오.

```
umount /cdrom
```

4. 제품이 설치되어 있는지 확인하십시오. 다음 명령을 입력하십시오.

```
lsipp -h | grep ibmlb
```

전체 제품이 설치되고 나면 이 명령은 다음을 표시합니다.

```
ibmlb.base.rte
ibmlb.admin.rte
ibmlb.lb.driver
ibmlb.lb.license
ibmlb.<component>.rte
ibmlb.doc.rte
ibmlb.ms.rte
ibmlb.msg.language.admin
ibmlb.msg.en_US.doc
ibmlb.msg.language.lb
```

Load Balancer 설치 경로는 다음과 같습니다.

- Administration - **/opt/ibm/edge/lb/admin**
- Load Balancer 컴포넌트 - **/opt/ibm/edge/lb/servers**
- Metric Server - **/opt/ibm/edge/lb/ms**
- 문서(관리 안내서) -**/opt/ibm/edge/lb/documentation**

RMI(Remote Method Invocation)를 사용하여 Load Balancer를 원격으로 관리하려면 클라이언트에 관리, 기본, 컴포넌트 및 사용권 패키지를 설치해야 합니다. RMI에 대한 자세한 정보는 282 페이지의 『원격 메소드 호출(RMI)』을 참조하십시오.

HP-UX 시스템 요구사항 및 설치

HP-UX 시스템의 요구사항

지원되는 브라우저를 포함하는 하드웨어 및 소프트웨어 요구사항에 대한 정보는 다음 웹 페이지 <http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921> 을 참조하십시오.

HP-UX 시스템용 설치

이 절에서는 제품 CD를 사용하여 HP-UX 시스템에 Load Balancer를 설치하는 방법에 대해 설명합니다.

설치하기 전에

설치 프로시저를 시작하기 전에, 소프트웨어를 설치하는 데 필요한 루트 권한이 있는지 확인하십시오.

이미 이전 버전이 설치되어 있으면, 현재 버전을 설치하기 전에 이전 버전의 설치를 제거해야 합니다. 먼저 실행 프로그램과 서버를 모두 중단하십시오. 그런 다음, []를 설치 제거하려면 39 페이지의 『패키지 설치 제거 지시사항』을 참조하십시오.

설치 단계

표 3에는 Load Balancer에 대한 설치 패키지 이름과 시스템의 패키지 설치 도구를 사용하여 패키지를 설치하는 권장 순서가 나열됩니다.

표 3. Load Balancer용 HP-UX 패키지 설치 세부사항

패키지 설명	HP-UX 패키지 이름
기본	ibmlb.base
관리 및 메시지	ibmlb.admin ibmlb.nlv-lang
Load Balancer 라이선스	ibmlb.lic
Load Balancer 컴포넌트	ibmlb.component
Documentation	ibmlb.doc
Metric Server	ibmlb.ms
주:	
1. lang 변수는 다음 언어 고유 코드(de_DE, en_US, es_ES, fr_FR, it_IT, ja_JP, ko_KR, zh_CN, zh_TW) 중 하나의 대체를 가리킵니다.	
2. component 변수는 disp(dispatcher), cbr(CBR), ss(Site Selector), cco(Cisco CSS Controller) 또는 nal(Nortel Alteon Controller) 중 하나의 대체를 가리킵니다.	
3. 문서 패키지(ibmlb.doc)는 영어로만 이루어져 있습니다. Load Balancer 문서 세트의 번역은 다음 웹 사이트에 있습니다. www.ibm.com/software/webservers/appserv/ecinfocenter.html 입니다.	

주: HP-UX 시스템에서는 포르투갈 브라질어(pt_BR) 로케일을 지원하지 않습니다.
HP-UX 시스템에서 지원되는 로케일은 다음과 같습니다.

- de_DE.iso88591
- en_US.iso88591
- es_ES.iso88591
- fr_FR.iso88591
- it_IT.iso88591
- ja_JP.SJIS
- ko_KR.eucKR
- zh_CN.hp15CN
- zh_TW.big5

패키지 설치 지시사항

다음 프로시저에서는 이 작업을 완료하는 데 필요한 단계를 자세히 설명합니다.

1. 로컬 슈퍼유저 루트가 되십시오.

```
su - root  
암호: password
```

2. 설치 명령을 발행하여 패키지를 설치하십시오.

```
swinstall -s /source package_name
```

여기서 *source*는 패키지 위치의 디렉토리 경로이고, *package_name*은 패키지 이름입니다.

CD의 루트에서 설치할 경우, 다음 명령은 Load Balancer의 기본 패키지(ibmlb.base)를 설치합니다.

```
swinstall -s /source ibmlb.base
```

CD의 루트에서 설치할 경우, Load Balancer의 모든 패키지를 설치하려면 다음 명령을 발행하십시오.

```
swinstall -s /source ibmlb
```

3. Load Balancer 패키지의 설치를 확인하십시오.

swlist 명령을 발행하여 설치한 모든 패키지를 나열하십시오. 예를 들어,

```
swlist -l filesset ibmlb
```

패키지 설치 제거 지시사항

swremove 명령을 사용하여 패키지를 설치 제거하십시오. 설치된 순서의 반대로 패키지를 제거하십시오. 예를 들어, 다음 명령을 발행하십시오.

- 모든 Load Balancer 패키지를 설치 제거하려면 다음을 수행하십시오.

```
swremove ibmlb
```

개별 패키지(예: Dispatcher 컴포넌트)를 설치 제거하려면 다음을 수행하십시오.

```
swremove ibmlb.disp
```

Linux 시스템 요구사항 및 설치

Linux 시스템의 요구사항

지원되는 브라우저를 포함하는 하드웨어 및 소프트웨어 요구사항에 대해서는 다음 웹 페이지 <http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921>을 참조하십시오.

Linux 시스템용 설치

이 절에서는 제품 CD를 사용하여 Linux에 Load Balancer를 설치하는 방법에 대해 설명합니다.

설치하기 전에

설치 프로시저를 시작하기 전에, 소프트웨어를 설치하는 데 필요한 루트 권한이 있는지 확인하십시오.

이미 이전 버전이 설치되어 있으면, 현재 버전을 설치하기 전에 이전 버전의 설치를 제거해야 합니다. 먼저 모든 실행 프로그램과 모든 서버가 정지되었는지 확인하십시오. 그런 다음, 전체 제품을 설치 제거하려면 **rpm -e pkgname**을 입력하십시오. 설치 제거할 때는 패키지 설치에 사용되는 순서를 반대로 따라 작업하여 관리 패키지가 맨 마지막에 설치 제거하십시오.

설치 단계

Load Balancer를 설치하려면 다음을 수행하십시오.

1. 설치를 준비하십시오.

- 루트로 로그인하십시오.
- 제품 매체를 삽입하거나 웹 사이트에서 제품을 다운로드하고 RPM(Red Hat Packaging Manager)을 사용하여 설치 이미지를 설치하십시오.

설치 이미지는 **eLBLX-version:tar.z** 형식의 파일입니다.

- **tar -xf eLBLX-version:tar.z**를 입력하여 임시 디렉토리에 있는 tar 파일을 해제하십시오 그러면 .rp 확장자를 가진 파일 세트가 생성됩니다.

다음은 RPM 설치 가능 패키지의 목록입니다.

- **ibmlb-base-release-version.hardw.rpm** (Base)
- **ibmlb-admin-release-version.hardw.rpm** (Administration)

- *ibmlb-lic-release-version.hardw.rpm* (License)
- *ibmlb-component-release-version.hardw.rpm* (LB component)
- *ibmlb-doc-release-version.hardw.rpm*(Documentation)
- *ibmlb-ms-release-version.hardw.rpm* (Metric Server)

여기서 —

- *release-version*은 현재 릴리스(예: 6.1-0)입니다.
- *hardw*는 i386, ppc64, ppc, s390, s390x, x86_64 값 중 하나입니다.
- *component*는 disp(Dispatcher component), cbr(CBR component), ss(Site Selector component), cco(Cisco CSS Controller), nal(Nortel Alteon Controller) 값 중 하나입니다.

문서 패키지는 영어로만 이루어져 있습니다. Load Balancer 문서 세트의 번역은 다음 웹 사이트에 있습니다. www.ibm.com/software/webervers/appserv/ecinfocenter.html입니다.

- 패키지가 설치되는 순서는 중요합니다. 다음은 필요한 패키지 목록과 준수해야 하는 설치 순서입니다.
 - Base(기본)
 - Administration (admin)
 - 사용권 (lic)
 - Load Balancer 컴포넌트(disp, cbr, ss, cco, nal)
 - Metric Server(ms)
 - 문서 (doc)

패키지를 설치하기 위한 명령은 RPM 파일이 있는 동일한 디렉토리에서 발행되어야 합니다. **rpm -i package.rpm** 명령을 발행하여 각 패키지를 설치하십시오.

Red Hat Linux 시스템: 알려진 Red Hat Linux 문제점 해결을 위해 *_db** RPM 파일을 삭제해야 하며, 삭제하지 않을 경우 오류가 발생합니다.

- Load Balancer 설치 경로는 다음과 같습니다.
 - Administration - **/opt/ibm/edge/lb/admin**
 - Load Balancer 컴포넌트 - **/opt/ibm/edge/lb/servers**
 - Metric Server- **/opt/ibm/edge/lb/ms**
 - 문서 - **/opt/ibm/edge/lb/documentation**
- 패키지를 설치 제거할 때는 패키지 설치에 사용되는 순서를 반대로 따라 작업하여 관리 패키지를 맨 마지막에 설치 제거하십시오.

2. 제품이 설치되어 있는지 확인하십시오. 다음 명령을 입력하십시오.

```
rpm -qa | grep ibmlb
```

전체 제품을 설치하면 다음 예와 같은 목록이 생성됩니다.

- *ibmlb-base-release-version*
- *ibmlb-admin-release-version*
- *ibmlb-lic-release-version*
- *ibmlb-dsp-release-version*
- *ibmlb-cbr-release-version*
- *ibmlb-ss-release-version*
- *ibmlb-cco-release-version*
- *ibmlb-nal-release-version*
- *ibmlb-doc-release-version*
- *ibmlb-ms-release-version*

RMI(Remote Method Invocation)를 사용하여 Load Balancer를 원격으로 관리하려면 클라이언트에 관리, 기본, 컴포넌트 및 사용권 패키지를 설치해야 합니다. RMI에 대한 자세한 정보는 282 페이지의 『원격 메소드 호출(RMI)』을 참조하십시오.

Solaris 시스템 요구사항 및 설치

Solaris 요구사항

지원되는 브라우저를 포함하는 하드웨어 및 소프트웨어 요구사항에 대한 정보는 다음 웹 페이지 <http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921> 을 참조하십시오.

Solaris에 설치

이 절에서는 제품 CD를 사용하여 Solaris 시스템에 Load Balancer를 설치하는 방법에 대해 설명합니다.

설치하기 전에

설치 프로시저를 시작하기 전에, 소프트웨어를 설치하는 데 필요한 루트 권한이 있는지 확인하십시오.

이미 이전 버전이 설치되어 있으면, 현재 버전을 설치하기 전에 이전 버전의 설치를 제거하십시오. 먼저 실행 프로그램과 서버를 모두 중지하십시오. 그런 다음, Load Balancer를 설치 제거하려면 **pkgrm pkgname**을 입력하십시오.

설치 단계

Load Balancer를 설치하려면 다음을 수행하십시오.

1. 설치를 준비하십시오.

- 루트 사용자로 로그인하십시오.
- Load Balancer 소프트웨어가 들어 있는 CD-ROM을 해당 드라이브에 넣으십시오.

명령 프롬프트에 **pkgadd -d pathname**을 입력하십시오. 여기서 *pathname*은 패키지가 있는 하드 디스크의 디렉토리 또는 CD-ROM 드라이브의 장치 이름입니다 (예: **pkgadd -d /cdrom/cdrom0/**).

다음은 표시된 패키지 목록과 설치해야 할 권장 순서입니다.

- ibmlbbase(기본)
- ibmlbadm(관리)
- ibmlblic(사용권)
- ibmlbdisp(Dispatcher 컴포넌트)
- ibmlbcbr(CBR 컴포넌트)
- ibmlbss(Site Selector 컴포넌트)
- ibmlbcc(Cisco CSS Controller 컴포넌트)
- ibmlbnal(Nortel Alteon Controller 컴포넌트)
- ibmlbdoc(문서)
- ibmlbms(Metric Server)

문서 패키지(ibmlbdoc)는 영어로만 이루어져 있습니다. Load Balancer 문서 세트의 번역은 다음 웹 사이트에 있습니다. www.ibm.com/software/webservers/appserv/ecinfocenter.html입니다.

모든 패키지를 설치하려면 “all”을 입력한 후 리턴 키만 누르면 됩니다. 일부 컴포넌트만 설치하려면 설치할 패키지에 해당되는 이름을 공백이나 쉼표로 구분하여 입력한 후 리턴 키를 누르십시오. 기존 디렉토리나 파일의 권한을 변경하라는 프롬프트가 표시될 수 있습니다. 리턴 키를 누르거나 “yes”로 답하면 됩니다. 전제조건 순서가 아닌 알파벳 순서로 설치되기 때문에 전제된 패키지를 설치해야 합니다. “all”을 입력한 후 모든 프롬프트에 “yes”로 답하면 설치가 완료됩니다.

문서 및 Metric Server와 함께 Dispatcher 컴포넌트만 설치할 경우, ibmlbbase, ibmlbadm, ibmlblic, ibmlbdisp, ibmlbdoc 및 ibmlbms 패키지를 설치해야 합니다.

RMI(Remote Method Invocation)를 사용하여 Load Balancer를 원격으로 관리하려면 클라이언트에 관리, 기본, 컴포넌트 및 사용권 패키지를 설치해야 합니다. RMI에 대한 자세한 정보는 282 페이지의 『원격 메소드 호출(RMI)』을 참조하십시오.

Load Balancer 설치 경로는 다음과 같습니다.

- Load Balancer 컴포넌트는 `/opt/ibm/edge/lb/servers` 설치 디렉토리에 있습니다.
- 설치된 Administration은 `/opt/ibm/edge/lb/admin` 디렉토리에 있습니다.
- 설치된 Metric Server는 `/opt/ibm/edge/lb/ms` 디렉토리에 있습니다.
- 설치된 문서는 `/opt/ibm/edge/lb/documentation` 디렉토리에 있습니다.

2. 제품이 설치되어 있는지 확인하십시오. `pkginfo | grep ibm` 명령을 실행하십시오.

Windows 시스템 요구사항 및 설치

Windows 시스템의 요구사항

지원되는 브라우저를 포함하는 하드웨어 및 소프트웨어 요구사항에 대한 정보는 다음 웹 페이지 <http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921> 을 참조하십시오.

Windows 시스템용 설치

이 절에서는 제품 CD를 사용하여 Windows 시스템에 Load Balancer를 설치하는 방법에 대해 설명합니다.

설치 패키지

설치할 패키지 선택사항이 제공됩니다.

- Administration
- License
- Dispatcher
- Content Based Routing
- Site Selector
- Cisco CSS Controller
- Nortel Alteon Controller
- Documentation
- Metric Server

RMI(Remote Method Invocation)를 사용하여 Load Balancer를 원격으로 관리하려면 클라이언트에 관리, 사용권 및 컴포넌트 패키지를 설치해야 합니다. RMI에 대한 자세한 정보는 282 페이지의 『원격 메소드 호출(RMI)』을 참조하십시오.

설치하기 전에

제한사항: Windows 버전의 []는 IBM Firewall과 동일한 시스템에 설치할 수 없습니다.

설치 프로시저를 시작하기 전에 관리자 또는 관리 권한이 있는 사용자로 로그인했는지 확인하십시오.

이미 이전 버전이 설치되어 있으면, 현재 버전을 설치하기 전에 이전 버전의 설치를 제거해야 합니다. 프로그램 추가/제거를 사용하여 설치 제거하려면 다음을 수행하십시오.

1. 시작 > 설정(Windows 2000의 경우) > 제어판을 클릭하십시오.
2. 프로그램 추가/제거를 두 번 클릭하십시오.
3. *IBM WebSphere Edge Components*(또는 이전 이름(예: *IBM Edge Server*))를 선택하십시오.
4. 변경/제거 단추를 클릭하십시오.

설치 단계

Load Balancer를 설치하려면 다음을 수행하십시오.

1. CD-ROM 드라이브에 Load Balancer CD-ROM을 넣으면 설치 창이 자동으로 표시됩니다.
2. 다음 단계는 사용자 컴퓨터에서 CD의 자동 실행이 작동되지 않는 경우에만 필요합니다. 마우스 왼쪽 단추를 클릭하여 다음 작업을 수행하십시오.
 - 시작을 클릭하십시오.
 - 실행을 선택하십시오.
 - CD-ROM 디스크 드라이브를 지정한 후, 다음 예와 같이 setup.exe를 지정하십시오.

E:\setup

3. 설치할 언어를 선택하십시오.
4. 확인을 클릭하십시오.
5. 설치 프로그램의 지침을 따르십시오.
6. 드라이브 또는 디렉토리 대상을 변경하려면, 찾아보기를 클릭하십시오.
7. “모든 Load Balancer 제품” 또는 “컴포넌트 선택” 중 하나를 선택할 수 있습니다.
8. 설치가 끝나면 Load Balancer를 사용하기 전에 시스템을 재부트하라는 메시지가 표시됩니다. 이 작업을 수행해야만 모든 파일이 제대로 설치되고 IBMMLBPATH 환경 변수가 레지스트리에 제대로 추가되었는지 확인할 수 있습니다.

Load Balancer 설치 경로는 다음과 같습니다.

- Administration – **C:\Program Files\IBM\edge\lb\admin**
- Load Balancer 컴포넌트 – **C:\Program Files\IBM\edge\lb\servers**
- Metric Server – **C:\Program Files\IBM\edge\lb\ms**
- Documentation(관리 안내서) – **C:\Program Files\IBM\edge\lb\documentation**

주: 설치 디렉토리의 문서는 영어로만 이루어져 있습니다. Load Balancer 문서 세트의 번역은 다음 웹 사이트에 있습니다.

www.ibm.com/software/webservers/appserv/ecinfocenter.html

제 2 부 Dispatcher 컴포넌트

이 파트에서는 빠른 시작 구성, 계획 고려사항에 대한 정보를 제공하며 Load Balancer의 Dispatcher 컴포넌트 구성 메소드에 대해 설명합니다. 다음 장을 포함합니다.

- 49 페이지의 제 5 장 『빠른 시작 구성』
- 57 페이지의 제 6 장 『Dispatcher 계획』
- 69 페이지의 제 7 장 『Dispatcher 구성』
- 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』

제 5 장 빠른 시작 구성

빠른 시작 예제에서는 두 개의 웹 서버 간에 웹 통신량을 로드 밸런스하기 위해 Dispatcher 컴포넌트의 mac 전달 메소드를 사용하여 로컬로 연결된 세 대의 워크스테이션을 구성하는 방법을 보여줍니다. 이 구성은 본질적으로 임의의 다른 TCP나 UDP 응용프로그램 통신량 밸런스와 동일하게 됩니다.

중요: IPv4 및 IPv6용 Load Balancer를 사용하는 경우, 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』를 참조하십시오.

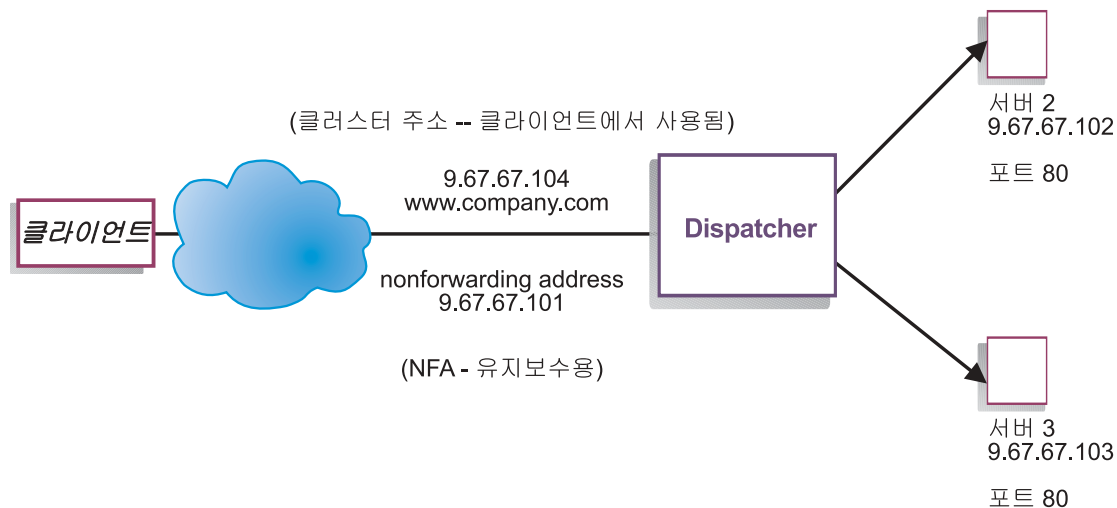


그림 8. 간단한 로컬 Dispatcher 구성

mac 전달 메소드는 기본 전달 메소드며 그에 따라 Dispatcher는 수신 요청을 서버에 로드 밸런스하고 서버는 직접 클라이언트에 응답을 리턴합니다. Dispatcher의 MAC 전달 메소드에 대한 자세한 정보는 59 페이지의 『Dispatcher의 MAC 레벨 경로 지정 (MAC 전달 메소드)』을 참조하십시오.

주: 웹 서버 워크스테이션 중 하나에 위치한 Dispatcher가 있는 두 대의 워크스테이션만을 사용하여 구성을 완료할 수 있습니다. 이 설정은 결합 배치된 구성을 나타냅니다. 보다 복잡한 구성 설정의 절차는 72 페이지의 『Dispatcher 시스템 설정』에 나와 있습니다.

필요한 내용

빠른 시작 예제의 경우 세 대의 워크스테이션과 네 개의 IP 주소가 필요합니다. 하나의 워크스테이션은 Dispatcher 시스템이고 나머지 워크스테이션은 웹 서버입니다. 웹 서버마다 하나의 IP 주소가 필요합니다. Dispatcher 워크스테이션에는 웹 사이트에 액세스하기 위해 클라이언트에 제공하는 두 개의 주소(비전달 주소(NFA) 및 클러스터 주소(로드 밸런싱되는 주소)가 필요합니다.

주: NFA는 **hostname** 명령이 리턴하는 주소입니다. 이 주소는 원격 구성과 같은 관리 목적에 사용됩니다.

준비 방법

1. 로컬로 접속된 이 구성 예제의 경우, 동일한 LAN 세그먼트에 워크스테이션을 설정하십시오. 세 시스템 간의 네트워크 통신량이 라우터 또는 브리지를 통과해서는 안 됩니다. (원격 서버에 대한 구성 설정에 대해서는 244 페이지의 『광역 Dispatcher 지원 구성』의 내용을 참조하십시오.)
2. 세 워크스테이션의 네트워크 어댑터를 구성하십시오. 이 예제에서는 다음과 같이 네트워크가 구성되어 있다고 가정합니다.

워크스테이션	이름	IP 주소
1	server1.Intersplashx.com	9.47.47.101
2	server2.Intersplashx.com	9.47.47.102
3	server3.Intersplashx.com	9.47.47.103
Netmask = 255.255.255.0		

각 워크스테이션에는 표준 이더넷 네트워크 인터페이스 카드가 하나만 있습니다.

3. server1.Intersplashx.com이 server2.Intersplashx.com과 server3.Intersplashx.com을 둘다 ping하는지 확인하십시오.
4. server2.Intersplashx.com과 server3.Intersplashx.com이 server1.Intersplashx.com을 ping하는지 확인하십시오.
5. 두 개의 웹 서버(서버 2와 서버 3)에서 콘텐츠가 동일한지 확인하십시오. 이는 두 워크스테이션 모두의 데이터를 복제하고 NFS, AFS[®] 또는 DFS[™]와 같은 공유 파일 시스템을 사용하거나 사용자 사이트에 적절한 다른 방법으로 수행될 수 있습니다.
6. server2.Intersplashx.com과 server3.Intersplashx.com의 웹 서버가 작동하는지 확인하십시오. 웹 브라우저를 사용하여 **http://server2.Intersplashx.com**과 **http://server3.Intersplashx.com**에서 페이지를 직접 요청하십시오.
7. 이 LAN 세그먼트에 유효한 또 다른 IP 주소를 확보하십시오. 이 주소는 사용자 사이트에 액세스할 클라이언트에 제공할 주소입니다. 이 예제에서는 다음을 사용합니다.

Name= www.Intersplashx.com
IP=9.47.47.104

8. www.Intersplashx.com의 통신을 승인하는 두 개의 웹 서버 워크스테이션을 구성하십시오.

www.Intersplashx.com 별명을 server2.Intersplashx.com 및
server3.Intersplashx.com의 루프백 인터페이스에 추가하십시오.

- AIX 시스템의 경우:

ifconfig lo0 alias www.Intersplashx.com netmask 255.255.255.0

- Solaris 9 시스템의 경우:

ifconfig lo0:1 plumb www.Intersplashx.com netmask 255.255.255.0 up

- 기타 운영체제의 경우 80 페이지의 표 5를 참조하십시오.

9. 루프백 인터페이스의 별명 지정으로 인해 작성되었을 수 있는 추가 라우트를 삭제하십시오. 83 페이지의 『2단계. 여분의 라우트 확인』을 참조하십시오.

두 대의 웹 서버 워크스테이션에 필요한 모든 구성 단계를 완료했습니다.

Dispatcher 컴포넌트 구성

Dispatcher를 통해 명령행, 구성 마법사 또는 GUI(Graphical User Interface)를 사용하여 구성을 작성할 수 있습니다.

주: 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름과 파일 이름의 매개변수 값만 예외입니다.

명령행을 사용한 구성

명령행을 사용 중이면, 다음 단계를 수행하십시오.

1. Dispatcher에서 dsserver를 시작하십시오.

- AIX, HP-UX, Linux 또는 Solaris 시스템의 경우, **dsserver** 명령을 루트 사용자로 실행하십시오.
- Windows 시스템의 경우, dsserver는 자동으로 시작되는 서비스로서 실행됩니다.

2. Dispatcher의 실행 프로그램 기능을 시작하십시오.

dscontrol executor start

3. 클러스터 주소를 Dispatcher 구성에 추가하십시오.

dscontrol cluster add www.Intersplashx.com

4. HTTP 프로토콜 포트를 Dispatcher 구성에 추가하십시오.

dscontrol port add www.Intersplashx.com:80

5. 각 웹 서버를 Dispatcher 구성에 추가하십시오.

```
dscontrol server add www.Intersplashx.com:80:server2.Intersplashx.com
```

```
dscontrol server add www.Intersplashx.com:80:server3.Intersplashx.com
```

6. 클러스터 주소의 통신량을 승인하려면, 다음과 같이 워크스테이션을 구성하십시오.

```
dscontrol executor configure www.Intersplashx.com
```

7. Dispatcher의 관리자 기능을 시작하십시오.

```
dscontrol manager start
```

Dispatcher는 서버 성능에 따라 로드 밸런스를 수행합니다.

8. Dispatcher의 어드바이저 기능을 시작하십시오.

```
dscontrol advisor start http 80
```

Dispatcher에서는 실패한 웹 서버로 클라이언트 요청이 전송되지 않았음을 확인합니다.

로컬로 연결된 서버의 기본 구성을 완료했습니다.

구성 검사

구성이 작동되고 있는지 확인하십시오.

1. 웹 브라우저에서 **http://www.Intersplashx.com** 위치로 이동하십시오. 페이지가 표시되면 구성이 작동하는 것입니다.
2. 웹 브라우저에서 페이지를 재로드하십시오.
3. **dscontrol server report www.Intersplashx.com:80:** 명령의 결과를 확인하십시오. 두 서버의 총 연결 컬럼은 “2”까지 추가해야 합니다.

GUI(Graphical User Interface)를 사용한 구성

Dispatcher GUI 사용에 대한 자세한 정보는 71 페이지의 『GUI』 및 499 페이지의 부록 A 『GUI: 일반 명령』을 참조하십시오.

구성 마법사

구성 마법사 사용에 대한 정보는 72 페이지의 『구성 마법사를 사용한 구성』을 참조하십시오.

클러스터, 포트, 서버 구성의 유형

사용자 사이트를 지원하기 위해 []를 구성할 수 있는 여러 가지 방법이 있습니다. 사용자 사이트에 모든 고객을 연결할 하나의 호스트 이름만 있는 경우, 서버의 단일 클러스터를 정의할 수 있습니다. 이들 각 서버에서 Load Balancer가 통신할 포트를 구성하십시오. 그림 9를 참조하십시오.

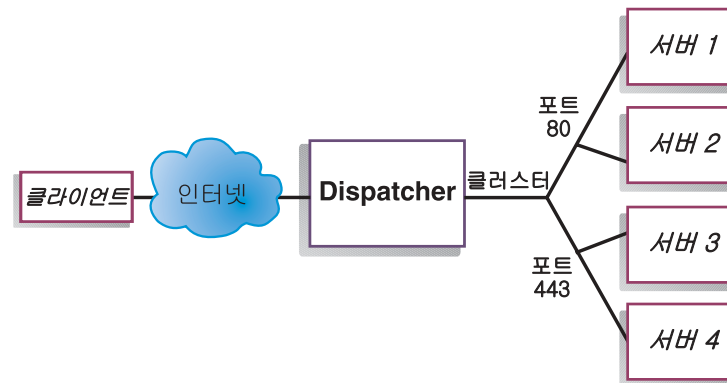


그림 9. 단일 클러스터와 두 개의 포트에 구성된 Dispatcher에 대한 예제

Dispatcher 컴포넌트 예제에서는 하나의 클러스터가 `www.productworks.com`에 정의됩니다. 이 클러스터에는 두 개의 포트가 있습니다(HTTP용 포트 80과 SSL용 포트 443). `http://www.productworks.com`(포트 80)에 요청을 작성하는 클라이언트는 `https://www.productworks.com`(포트 443)을 요청하는 클라이언트와 다른 서버로 이동합니다.

지원되는 각 프로토콜에 대한 전용 서버가 많은 매우 큰 사이트에서는 다른 Load Balancer 구성 방법이 적합합니다. 이 경우, 54 페이지의 그림 10에 표시된 대로 단일 포트이지만 많은 서버가 있는 각 프로토콜마다 클러스터를 정의하고자 할 수 있습니다.

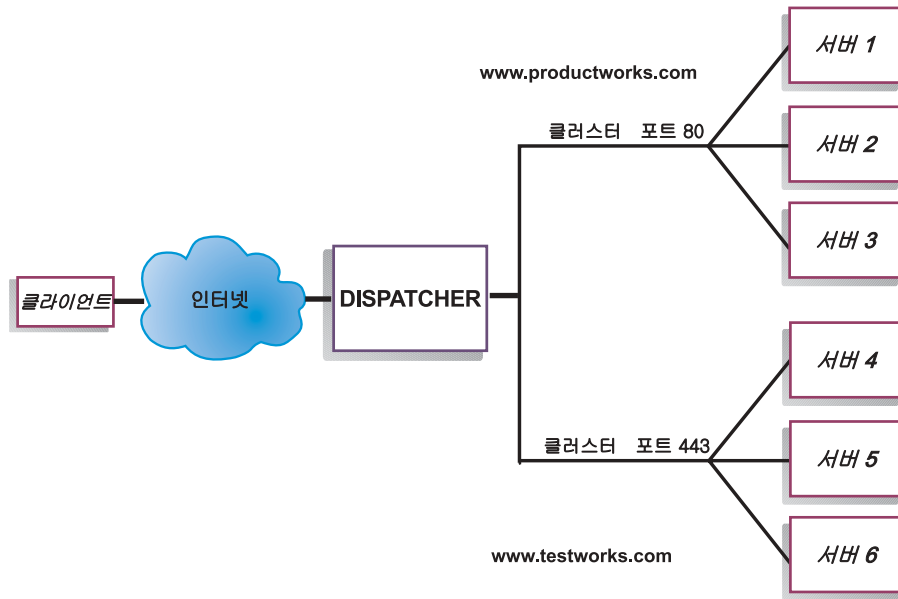


그림 10. 각각 단일 포트인 두 개의 클러스터로 구성된 Dispatcher에 대한 예제

Dispatcher 컴포넌트 예제에서, 두 개의 클러스터는 포트 80(HTTP)의 경우, `www.productworks.com`, 포트 443(SSL)의 경우, `www.testworks.com` 사이트에 정의됩니다.

각각 다른 URL로 사이트에 들어가는 여러 회사나 부서에 대해 사이트가 콘텐츠를 호스트할 경우에 Load Balancer를 구성하기 위한 세 번째 방법이 필요합니다. 이 경우에는 55 페이지의 그림 11에 표시된 것처럼 회사나 부서의 클러스터를 각각 정의한 다음, 해당 URL에서 연결을 받을 포트를 정의할 수 있습니다.

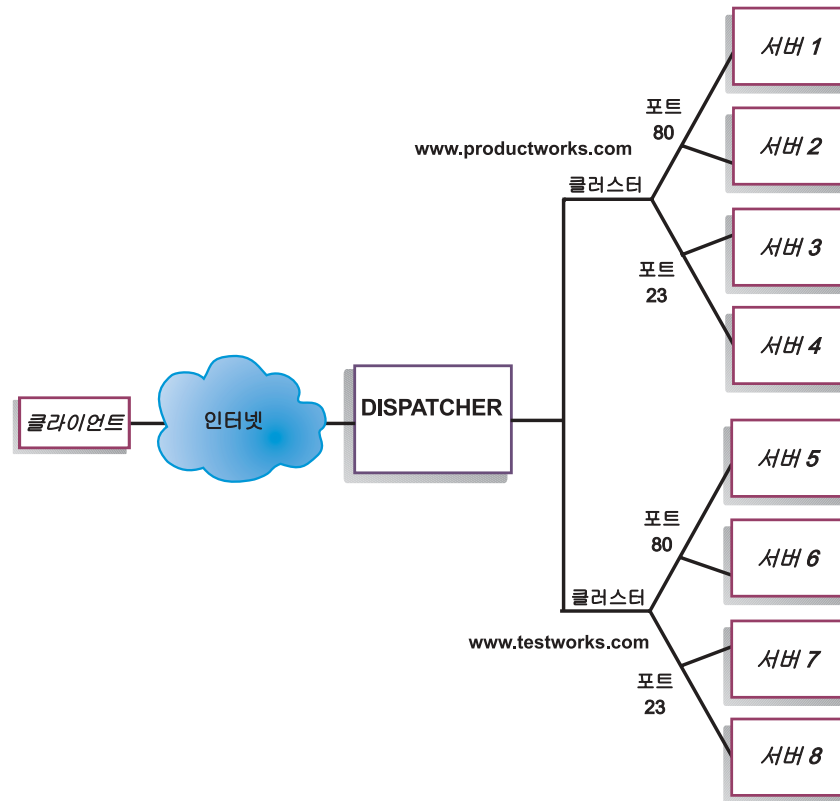


그림 11. 각각 포트가 두 개인 두 개의 클러스터로 구성된 Dispatcher에 대한 예제

Dispatcher 컴포넌트 예제에서, 두 개의 클러스터는 HTTP용 포트 80과 Telnet용 포트 23으로 정의되며, 각각 www.productworks.com 및 www.testworks.com 사이트에 정의됩니다.

제 6 장 Dispatcher 계획

이 장에서는 Dispatcher 컴포넌트를 설치하고 구성하기 전에 네트워크 계획자가 고려해야 할 사항에 대해 설명합니다.

- 네트워크 관리에 필요한 기능의 개요에 대해서는 21 페이지의 제 3 장 『네트워크 관리: 사용할 Load Balancer 기능 결정』 페이지를 참조하십시오.
- Dispatcher의 로드 밸런스 매개변수 구성에 대한 정보는 69 페이지의 제 7 장 『Dispatcher 구성』을 참조하십시오.
- IPv4 및 IPv6용 Load Balancer를 사용하는 경우 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』를 참조하십시오.
- 더 많은 고급 기능을 위한 Load Balancer 설정 방법에 대한 정보는 215 페이지의 제 22 장 『Dispatcher, CBR 및 Site Selector에 대한 고급 기능』을 참조하십시오.
- 원격 인증 관리, Load Balancer 로그 및 Load Balancer 컴포넌트 사용법에 대한 정보는 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』를 참조하십시오.

이 장에는 다음과 같은 섹션이 수록되어 있습니다.

- 『계획 고려사항』
- 59 페이지의 『Dispatcher의 MAC 레벨 경로 지정(MAC 전달 메소드)』
- 59 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 메소드)』
- 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』
- 64 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』
- 66 페이지의 『고가용성』

주: 제품이 Network Dispatcher로 알려진 이전 버전의 경우, Dispatcher 제어 명령은 `ndcontrol`이었습니다. Dispatcher 제어 명령 이름은 **`dscontrol`**입니다.

계획 고려사항

중요: IPv4 및 IPv6용 Load Balancer를 사용하는 경우, 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』를 참조하십시오.

Dispatcher는 다음과 같은 기능으로 구성됩니다.

- **`dsserver`**는 명령행에서 실행 프로그램, 관리자 및 어드바이저에 대한 요청을 처리합니다.
- 실행 프로그램은 TCP 및 UDP 연결의 포트 기반 로드 밸런스를 지원합니다. 받은 요청 유형(예: HTTP, FTP, SSL 등)에 따라 서버에 연결을 전달할 수 있습니다. 실행 프로그램은 항상 Dispatcher 컴포넌트가 로드 밸런스에 사용될 때 실행됩니다.

- 관리자에서는 다음에 따라 실행 프로그램에서 사용하는 가중치를 설정합니다.
 - 실행 프로그램의 내부 카운터
 - 어드바이저가 제공하는 서버에서의 피드백
 - Metric Server 또는 WLM과 같은 시스템 모니터링 프로그램에서의 피드백

관리자를 사용하는 것은 선택입니다. 그러나 관리자를 사용하지 않으면, 현재 서버 가중치에 따라 가중된 라운드 로빙 스케줄링을 사용하여 로드 밸런스가 수행되며, 어드바이저는 사용가능하지 않습니다.

- 어드바이저는 관리자를 호출하여 가중치를 적절하게 설정하기 전에 서버를 조회하고 프로토콜별로 결과를 분석합니다. 현재 어드바이저는 HTTP, FTP, SSL, SMTP, NNTP, IMAP, POP3, SIP 및 Telnet 프로토콜에 대해 사용할 수 있습니다.

또한 Dispatcher는 DB2[®] 서버의 상태에 대해 보고하는 DB2 어드바이저 및 서버가 ping 명령에 응답하는지 여부를 보고하는 Ping 어드바이저와 같은 프로토콜 고유 정보를 교환하지 않는 어드바이저를 제공합니다. 어드바이저에 대한 전체 목록은 203 페이지의 『어드바이저 목록』을 참조하십시오.

또한 사용자 고유의 어드바이저를 작성하는 옵션도 있습니다(207 페이지의 『사용자 정의(사용자 정의 가능) 어드바이저 작성』 참조).

어드바이저를 사용하는 것은 선택입니다.

- 실행 프로그램, 어드바이저 및 관리자를 구성하고 관리하려면, 명령행(**dscontrol**) 또는 그래픽 사용자 인터페이스(**lbadmin**)를 사용하십시오.
- Dispatcher 시스템의 구성과 관리에 사용할 예제 구성 파일이 제공됩니다. 511 페이지의 부록 C 『예제 구성 파일』을 참조하십시오. 제품 설치 후, 이 파일은 Load Balancer가 있는 **...ibm/edge/lb/servers/samples** 하위 디렉토리에서 찾을 수 있습니다.
- SNMP 기본 관리 응용프로그램은 **SNMP** 서브 에이전트로 Dispatcher의 상태를 모니터링할 수 있습니다.

Dispatcher의 세 가지 핵심 기능(실행 프로그램, 관리자 및 어드바이저)은 서버 간의 수신 요청의 밸런스를 조정하고 디스패치하기 위해 상호작용합니다. 요청을 로드 밸런스 하면, 실행 프로그램은 새 연결 수, 활성 연결 수 및 완료 상태에 있는 연결 수를 모니터링합니다. 또한 실행 프로그램은 완료된 연결이나 재설정 연결의 가비지 콜렉션을 수행하고 이 정보를 관리자에 제공합니다.

관리자는 실행 프로그램, 어드바이저 및 Metric Server와 같은 시스템 모니터링 시스템으로부터 정보를 수집합니다. 관리자에서 수신하는 정보에 따라, 어떻게 서버 시스템이 각 서버 포트에 가중되는지를 조정하고 실행 프로그램에 새로운 연결의 밸런스 조정 에 사용할 새로운 가중 방법을 제공합니다.

어드바이저는 지정된 포트에서 각 서버를 모니터링하여 서버의 응답 시간과 사용 가능성을 판별한 다음, 이 정보를 관리자에 제공합니다. 또한 어드바이저는 서버의 연결 또는 단절도 모니터링합니다. 관리자와 어드바이저 없이, 실행 프로그램은 현재 서버의 가중치에 따라 라운드 로빙 스케줄링을 수행합니다.

전달 메소드

Dispatcher를 사용하여, 포트 레벨에서 지정된 세 가지 전달 메소드 즉, MAC 전달, NAT/NAPT 전달 또는 CBR(content-based routing) 전달 중 하나를 선택할 수 있습니다.

Dispatcher의 MAC 레벨 경로 지정(MAC 전달 메소드)

Dispatcher는 Dispatcher의 MAC 전달 메소드(기본 전달 메소드)를 통해 선택한 서버로 들어오는 요청을 로드 밸런싱하고, 서버는 Dispatcher 없이 클라이언트로 직접 응답을 리턴합니다. Dispatcher는 이 전달 메소드를 통해서 인바운드 클라이언트-서버 플로우만 확인합니다. 서버에서 클라이언트로의 아웃바운드 플로우는 보지 않아도 됩니다. 이로 인해 응용프로그램에 미치는 영향이 크게 줄어들고 네트워크 성능이 향상될 수 있습니다.

전달 메소드는 `dscontrol port add cluster:port method value` 명령을 사용하여 포트를 추가할 때 선택될 수 있습니다. 기본 전달 메소드 값은 `mac`입니다. 방법 매개변수는 포트가 추가될 때만 지정할 수 있습니다. 포트가 추가되면 전달 메소드의 설정을 변경할 수 없습니다. 404 페이지의 『dscontrol port — 포트 구성』에서 자세한 정보를 참조하십시오.

Linux 제한사항: Linux 시스템은 광고 하드웨어 주소의 호스트기반 모델을 ARP 사용 IP 주소에 씁니다. 이 모델은 Load Balancer의 mac 전달 메소드에 대한 고가용성 결합 배치 서버 및 백엔드 서버 요구사항과 호환이 불가능합니다. Linux 시스템 동작을 변경하여 Load Balancer의 mac 전달과 호환이 가능하도록 하는 여러 가지 해결책이 설명되어 있는 85 페이지의 『Load Balancer의 mac 전달 사용 시 Linux 루프백 별명 지정 대안』 페이지를 참조하십시오.

zSeries 또는 S/390 서버 사용 시 Linux 제한사항: 개방형 시스템 어댑터(OSA) 카드가 있는 zSeries 또는 S/390 서버 사용 시 제한사항이 있습니다. 가능한 해결책에 대해서는 342 페이지의 『문제점: Linux에서는 개방형 시스템 어댑터(OSA) 카드가 있는 zSeries 또는 S/390 서버 사용 시 Dispatcher 구성 제한사항이 있음』 페이지를 참조하십시오.

Dispatcher의 NAT/NAPT(nat 전달 메소드)

Dispatcher의 NAT(네트워크 주소 변환) 또는 NAPT(네트워크 주소 포트 변환) 기능을 사용하면 로드 밸런싱된 서버가 로컬로 연결된 네트워크에 있어야 한다는 제한사항

이 제거됩니다. 서버가 원격 위치에 있을 때는 GRE/WAN 캡슐화 기술을 사용하지 않고 NAT 전달 메소드 기술을 사용할 수 있습니다. 또한 각 디먼이 고유한 포트를 인식할 경우, NAT 기능을 사용하여 로드 밸런싱된 각 서버 시스템에 있는 여러 서버 디먼에 액세스할 수도 있습니다.

여러 디먼의 서버는 두 가지 방법으로 구성할 수 있습니다.

- NAT를 사용하면, 여러 서버 디먼을 구성하여 다른 IP 주소에 대한 요청에 응답할 수 있습니다. 이를 IP 주소에 대한 서버 디먼 바인딩이라고 합니다.
- NAT를 사용하면, 여러 서버 디먼(동일한 물리적 서버에서 실행)을 구성하여 다른 포트 번호를 인식할 수 있습니다.

이 응용프로그램은 HTTP, SSL, IMAP, POP3, NNTP, SMTP, Telnet 프로토콜에서 잘 작동합니다.

제한사항

- Dispatcher의 NAT/NAPT 구현은 이러한 기능의 간단한 구현입니다. TCP/IP 패킷 헤더의 콘텐츠에 대해서만 분석하고 작동합니다. 패킷 데이터 부분의 콘텐츠는 분석하지 않습니다. Dispatcher의 경우, NAT/NAPT는 메시지의 데이터 부분에 주소나 포트 번호가 있는 FTP와 같은 응용프로그램 프로토콜에서 작동하지 않습니다. 이는 헤더 기반 NAT/NAPT의 제한사항으로 잘 알려져 있습니다.
- Dispatcher의 NAT/NAPT는 와일드 카드 클러스터나 와일드 카드 포트 기능과 함께 작동할 수 없습니다.

Dispatcher 시스템에 대한 세 개의 IP 주소, 즉 nfa, 클러스터 및 리턴 주소가 필요합니다. NAT/NAPT를 구현하려면 다음을 수행하십시오. 63 페이지의 『Dispatcher의 nat 또는 cbr 전달 메소드 구성에 대한 샘플 단계』:

- **dscontrol executor set** 명령에 **clientgateway** 매개변수를 설정하십시오. Clientgateway는 리턴되는 방향의 통신량을 Load Balancer에서 클라이언트로 전달하는 라우터 주소로 사용되는 IP 주소입니다. 이 값은 NAT/NAPT를 사용하기 전에 0이 아닌 IP 주소로 설정되어야 합니다. 383 페이지의 『dscontrol executor — 실행 프로그램 제어』에서 자세한 정보를 참조하십시오.
- 전달 메소드는 **dscontrol port add cluster:port method value** 명령을 사용하여 포트를 추가할 때 선택될 수 있습니다. 전달 메소드 값은 **nat**로 설정해야 합니다. 방법 매개변수는 포트가 추가될 때만 지정할 수 있습니다. 포트를 추가한 후에는 전달 메소드의 설정을 변경할 수 없습니다. 404 페이지의 『dscontrol port — 포트 구성』에서 자세한 정보를 참조하십시오.

주: 클라이언트 게이트웨이 주소를 0이 아닌 값으로 설정하지 않은 경우, 전달 메소드는 **mac**(MAC 기반 전달 메소드)만 될 수 있습니다.

- **dscontrol** 명령 및 **mapport**, **returnaddress**, **router** 매개변수를 사용하여 서버를 추가하십시오. 예를 들어,

```
dscontrol server add cluster:port:server mapport value returnaddress  
rtrnaddress router rtraddress
```

– **mapport**(선택적)

Dispatcher가 클라이언트의 요청을 로드 밸런싱하는 데 사용하는 서버의 포트 번호로 클라이언트 요청의 대상 포트 번호(Dispatcher용)를 맵핑합니다. **mapport**는 Load Balancer가 한 포트에서 클라이언트의 요청을 받아서 서버 시스템의 다른 포트로 전송할 수 있도록 허용합니다. **mapport**를 사용하면 여러 서버 디먼을 실행하는 서버 시스템에 대한 클라이언트의 요청을 로드 밸런싱할 수 있습니다. **mapport**에 대한 기본값은 클라이언트 요청의 대상 포트 번호입니다.

– **returnaddress**

리턴 주소는 Dispatcher 시스템에서 구성하는 고유한 주소 또는 호스트 이름입니다. Dispatcher는 서버에 대한 클라이언트의 요청을 로드 밸런싱할 때 출발지 주소로서 리턴 주소를 사용합니다. 그러면 패킷을 클라이언트로 직접 전송하는 대신 서버가 Dispatcher 시스템으로 패킷을 리턴할 수 있도록 합니다. (Dispatcher는 IP 패킷을 클라이언트에 전달하지 않습니다). 서버를 추가할 때 리턴 주소 값을 지정해야 합니다. 서버를 제거하고 다시 추가하지 않으면, 리턴 주소를 수정할 수 없습니다. 리턴 주소는 클러스터, 서버 또는 NFA 주소와 같을 수 없습니다.

– **router**

원격 서버로의 라우터 주소. 로컬에 접속된 서버의 경우, 서버가 Load Balancer와 동일한 시스템에 위치하지 않았으면 서버 주소를 입력하십시오. 실제 라우터 주소를 계속해서 사용하십시오.

mapport, **returnaddress** 및 **router** 매개변수를 사용하는 **dscontrol server** 명령에 대한 자세한 정보는 418 페이지의 『dscontrol server — 서버 구성』을 참조하십시오.

Dispatcher content-based routing(cbr 전달 메소드)

Dispatcher 컴포넌트를 사용하면 Caching Proxy를 사용하지 않고 HTTP("content" 유형 규칙 사용)와 HTTPS(SSL 세션 ID 연관 관계 사용)의 content-based routing을 수행할 수 있습니다. HTTP와 HTTPS 통신의 경우, Dispatcher 컴포넌트의 cbr 전달 메소드는 Caching Proxy를 요구하는 CBR 컴포넌트보다 content-based routing을 빨리 제공할 수 있습니다.

HTTP: Dispatcher content-based routing에 대해 HTTP 헤더나 URL의 콘텐츠를 기반으로 서버를 선택합니다. "content" 유형 규칙을 사용하여 구성됩니다. 콘텐츠 규칙을

구성할 때 탐색 문자열 "pattern"과 일련의 서버를 규칙에 지정하십시오. 이 규칙은 새 수신 요청을 처리할 때, 지정된 문자열을 클라이언트의 URL 또는 클라이언트 요청으로 지정된 HTTP 헤더와 비교합니다.

Dispatcher가 클라이언트 요청에서 문자열을 찾으면, Dispatcher는 규칙 내의 한 서버에 요청을 전달합니다. 그리고 나서, Dispatcher는 서버에서 클라이언트로 응답 데이터를 릴레이합니다("cbr" 전달 메소드).

Dispatcher가 클라이언트 요청에서 문자열을 찾지 못하면, Dispatcher는 규칙 내의 서버에서 서버를 선택하지 않습니다.

주: 콘텐츠 규칙은 CBR 컴포넌트에 구성된 것과 동일한 방법으로 Dispatcher 컴포넌트에 구성됩니다. Dispatcher는 HTTP 통신에 콘텐츠 규칙을 사용할 수 있습니다. 그러나 CBR 컴포넌트는 HTTP와 HTTPS(SSL) 통신에 모두 콘텐츠 규칙을 사용할 수 있습니다.

HTTPS(SSL): Dispatcher의 Content Based Routing은 클라이언트 요청의 SSL ID 세션 필드에 근거하여 로드 밸런싱합니다. SSL을 사용하면, 클라이언트 요청은 사전 세션의 SSL 세션 ID를 포함하며, 서버는 사전 SSL 연결의 캐시를 유지합니다. Dispatcher의 SSL ID 세션 연관 관계는 클라이언트와 서버가 서버와의 이전 연결에 대한 보안 매개변수를 사용하여 새 연결을 설정할 수 있도록 허용합니다. 공유 키 및 암호화 알고리즘과 같은 SSL 보안 매개변수의 재조정을 제거하면 서버는 CPU 주기를 절감하며 클라이언트는 보다 빠른 응답을 수신합니다. SSL 세션 ID 연관 관계를 사용 가능하게 하기 위해서는 포트에 지정된 **protocol** 유형은 **SSL**이고 포트 **stickytime**은 0이 아닌 값으로 설정해야 합니다. stickytime이 초과되면 클라이언트가 이전과 다른 서버로 전송될 수 있습니다.

Dispatcher 시스템에 대한 세 개의 IP 주소, 즉 nfa, 클러스터 및 리턴 주소가 필요합니다. Dispatcher content-based routing(63 페이지의 『Dispatcher의 nat 또는 cbr 전달 메소드 구성에 대한 샘플 단계』도 참조)을 구현하려면 다음을 수행하십시오.

- **dscontrol executor set** 명령에 **clientgateway** 매개변수를 설정하십시오. Clientgateway는 리턴되는 방향의 통신량을 Network Dispatcher에서 클라이언트로 전달하는 라우터 주소로 사용되는 IP 주소입니다. clientgateway의 기본값은 0입니다. content-based routing 전달 메소드를 추가하기 전에 이 값을 0이 아닌 IP 주소로 설정해야 합니다. 383 페이지의 『dscontrol executor — 실행 프로그램 제어』에서 자세한 정보를 참조하십시오.
- **dscontrol port add** 명령에 **method** 매개변수 및 **protocol** 매개변수를 사용하여 포트를 추가하십시오. 전달 메소드 값은 **cbr**로 설정해야 합니다. 포트 프로토콜 유형은 HTTP 또는 SSL중 하나일 수 있습니다. 404 페이지의 『dscontrol port — 포트 구성』에서 자세한 정보를 참조하십시오.

주: 클라이언트 게이트웨이 주소를 0 이외의 값으로 설정하지 않으면, 전달 메소드는 오직 **mac** 전달 메소드여야 합니다.

- `mapport`, `returnaddress` 및 `router` 매개변수를 사용하여 서버를 추가하십시오.

dscontrol server add cluster:port:server mapport value returnaddress rtnaddress router rtraddress

주: `mapport`(선택적), `returnaddress` 및 `router` 매개변수를 사용한 서버 구성에 대한 정보는 61 페이지를 참조하십시오.

- **HTTP:** 클라이언트 요청 콘텐츠(규칙 유형 **content**)를 기반으로 규칙을 사용하여 구성하십시오. 예를 들어,

dscontrol rule 125.22.22.03:80:contentRule1 type content pattern pattern

여기서, *pattern*은 콘텐츠 유형 규칙에 사용할 패턴을 지정합니다. 콘텐츠 규칙 유형에 대한 자세한 정보는 234 페이지의 『요청 콘텐츠에 따라 규칙 사용』을 참조하십시오. *pattern*의 유효한 표현식에 대한 자세한 정보는 507 페이지의 부록 B 『콘텐츠 규칙(패턴) 구문』을 참조하십시오.

주: 고가용성의 연결 레코드 복제본 기능(백업 Dispatcher 시스템이 기본 시스템을 대신할 때 클라이언트의 연결이 끊어지는지 확인)은 Dispatcher의 Content Based Routing으로 지원되지 않습니다.

Dispatcher의 nat 또는 cbr 전달 메소드 구성에 대한 샘플 단계

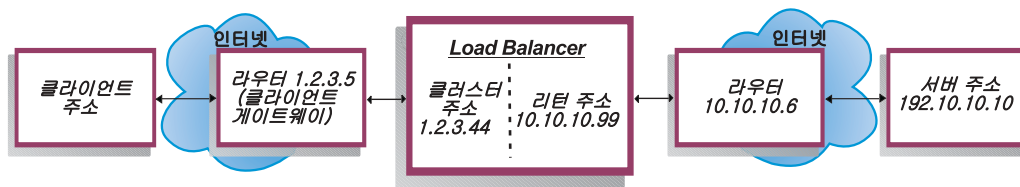


그림 12. Dispatcher의 nat 또는 cbr 전달 메소드 사용 예제

Dispatcher 시스템에 대해 최소한 세 개의 IP 주소가 필요합니다. 그림 12의 경우, 다음은 Dispatcher의 nat 또는 cbr 전달 메소드를 최소로 구성하기 위해 필요한 단계입니다.

1. Start the executor
`dscontrol executor start`
2. Define the client gateway
`dscontrol executor set clientgateway 1.2.3.5`
 NOTE: If your subnet does not have a local router, then you must configure a machine to do IP forwarding and use that as the clientgateway. Consult your operating system documentation to determine how to enable IP forwarding.

```

3. Define the cluster address
dscontrol cluster add 1.2.3.44

4. Configure the cluster address
dscontrol executor configure 1.2.3.44

5. Define the port with a method of nat or cbr
dscontrol port add 1.2.3.44:80 method nat
or
dscontrol port add 1.2.3.44:80 method cbr protocol http

6. Configure an alias return address on Load Balancer (using ethernet card 0)
NOTE: On Linux systems, you do not need to alias the return address if using
nat forwarding on a collocated machine.

dscontrol executor configure 10.10.10.99

or use the ifconfig command (for Linux or UNIX only):
AIX: ifconfig en0 alias 10.10.10.99 netmask 255.255.255.0
HP-UX: ifconfig lan0:1 10.10.10.99 netmask 255.255.255.0 up
Linux: ifconfig eth0:1 10.10.10.99 netmask 255.255.255.0 up
Solaris: ifconfig eri0 addif 10.10.10.99 netmask 255.255.255.0 up

7. Define the backend servers
dscontrol server add 1.2.3.4:80:192.10.10.10
router 10.10.10.6 returnaddress 10.10.10.99

```

클라이언트 게이트웨이(1.2.3.5)는 Load Balancer와 클라이언트 사이의 라우터 1 주소입니다. 라우터(10.10.10.6)는 Load Balancer와 백엔드 서버 사이의 라우터 2 주소입니다. 클라이언트 게이트웨이 또는 라우터 2 주소를 확실히 알지 못할 경우, 클라이언트(또는 서버) 주소와 함께 traceroute 프로그램을 사용하여 라우터 주소를 판단할 수 있습니다. 이 프로그램의 정확한 구문은 사용하는 운영 체제에 따라 달라집니다. 이 프로그램에 관한 자세한 정보는 운영 체제 문서를 참조해야 합니다.

서버가 Load Balancer와 동일한 서브넷에 있을 경우 즉, traceroute를 사용하여 리턴된 라우터가 없을 경우 서버 주소를 라우터 주소로 입력하십시오. 그러나 서버가 Load Balancer와 동일한 시스템에 위치한 경우, 라우터 주소는 서버 주소 대신 라우터 필드에 입력되어야 합니다. 라우터 주소는 7 단계에서 Load Balancer 시스템에서 "server add" 명령에 사용된 주소입니다.

서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)

서버를 분할하면 특정 URL과 고유 응용프로그램을 판별할 수 있습니다. 예를 들어, 하나의 웹 서버에서 JSP 페이지, HTML 페이지, GIF 파일, 데이터베이스 요청 등을 제공할 수 있습니다. 이제 []는 하나의 클러스터와 특정 포트 서버를 몇 개의 논리 서버로 분할하는 기능을 제공합니다. 이 기능으로 시스템의 특정 서비스에 권고하여 servlet 엔진이나 데이터베이스 요청이 더 빠르게 실행 중인지 또는 그렇지 않은지를 발견할 수 있습니다.

서버 파티션을 사용하면 예를 들어, Load Balancer에서 HTML 서비스가 페이지를 신속하게 제공하는 것을 발견할 수 있지만 데이터베이스 연결이 중지된 것도 발견할 수

있습니다. 이 기능으로 서버 규모의 가중치가 아니라 보다 세밀한 특정 서비스 작업로 드를 기반으로 로드를 분산시킬 수 있습니다.

HTTP 또는 HTTPS 어드바이저를 사용한 서버 파티셔닝

서버 파티셔닝은 HTTP 및 HTTPS 어드바이저와 함께 사용될 경우 유용할 수 있습니다. 예를 들어, HTML, GIF 및 JSP 페이지를 처리하는 HTML 서버가 있을 경우 포트 80에 서버를 추가하여 한 번 정의하면, 전체 HTTP 서버에 대해 하나의 로드 값만 수신하게 됩니다. GIF 서비스가 서버에서 기능하지 않을 가능성이 있으므로 이는 잘못 해석될 수 있습니다. Dispatcher는 여전히 GIF 페이지를 서버에 전달하지만, 클라이언트는 제한시간 또는 장애를 인식합니다.

서버를 포트에 세 번(예: ServerHTML, ServerGIF, ServerJSP) 정의하며 서버 **advisorrequest** 매개변수를 각 논리적 서버에 대해 다른 문자열로 정의할 경우, 서버의 특정 서비스의 상태를 조회할 수 있습니다. ServerHTML, ServerGIF 및 ServerJSP는 단일 물리적 서버에서 파티션된 세 개의 논리적 서버를 나타냅니다. ServerJSP의 경우, **advisorrequest** 문자열을 정의하여 JSP 페이지를 처리하는 시스템의 서비스를 조회할 수 있습니다. ServerGIF의 경우, **advisorrequest** 문자열을 정의하여 GIF 서비스를 조회할 수 있습니다. 그리고 ServerHTML의 경우, **advisorrequest**를 정의하여 HTML 서비스를 조회합니다. 따라서 GIF 서비스를 조회하는 **advisorrequest**에서 클라이언트가 아무 응답도 받지 못할 경우, Dispatcher는 논리적 서버(ServerGIF)가 작동 중지 상태인 것으로 표시하는 한편 다른 두 논리적 서버는 양호한 것으로 표시합니다. Dispatcher는 물리적 서버에 더 이상 GIF를 전달하지 않지만, 여전히 JSP 및 HTML 요청을 서버에 전송할 수 있습니다.

advisorrequest 매개변수에 대한 자세한 정보는 205 페이지의 『응답 및 요청(URL) 옵션을 사용하여 HTTP 또는 HTTPS 어드바이저 구성』을 참조하십시오.

물리적 서버를 논리적 서버로 구성하는 예

Dispatcher 구성에서 **cluster:port:server** 계층을 사용하여 물리적 서버 또는 논리 서버를 표시할 수 있습니다. 서버는 기호 이름 또는 IP 주소 형식으로 된 시스템(물리적 서버)의 고유한 IP 주소입니다. 또는 파티션된 서버를 나타내도록 서버를 정의할 경우, **dscontrol server add** 명령의 **address** 매개변수에서 물리적 서버에 대한 해석 가능한 서버 주소를 제공해야 합니다. 418 페이지의 『dscontrol server — 서버 구성』에서 자세한 정보를 참조하십시오.

다음은 다른 요청 유형을 처리하기 위해 물리적 서버를 논리 서버로 분할하는 예제입니다.

```
클러스터: 1.1.1.1
  포트: 80
    서버: A(IP 주소 1.1.1.2)
      HTML 서버
    서버: B(IP 주소 1.1.1.2)
```

GIF 서버
 서버: C(IP 주소 1.1.1.3)
 HTML 서버
 서버: D(IP 주소 1.1.1.3)
 JSP 서버
 서버: E(IP 주소 1.1.1.4)
 GIF 서버
 서버: F(IP 주소 1.1.1.4)
 JSP 서버

규칙1: /*.htm
 서버: A
 서버: C

규칙2: /*.jsp
 서버: D
 서버: F

규칙3: /*.gif
 서버: B
 서버: E

이 예제에서 서버 1.1.1.2는 두 개의 논리 서버 A(HTML 요청 처리) 및 B(GIF 요청 처리)로 분할됩니다. 서버 1.1.1.3은 두 개의 논리 서버 C(HTML 요청 처리) 및 D(JSP 요청 처리)로 분할됩니다. 서버 1.1.1.4는 두 개의 논리 서버 E(HTML 요청 처리) 및 F(JSP 요청 처리)로 분할됩니다.

고가용성

단순 고가용성

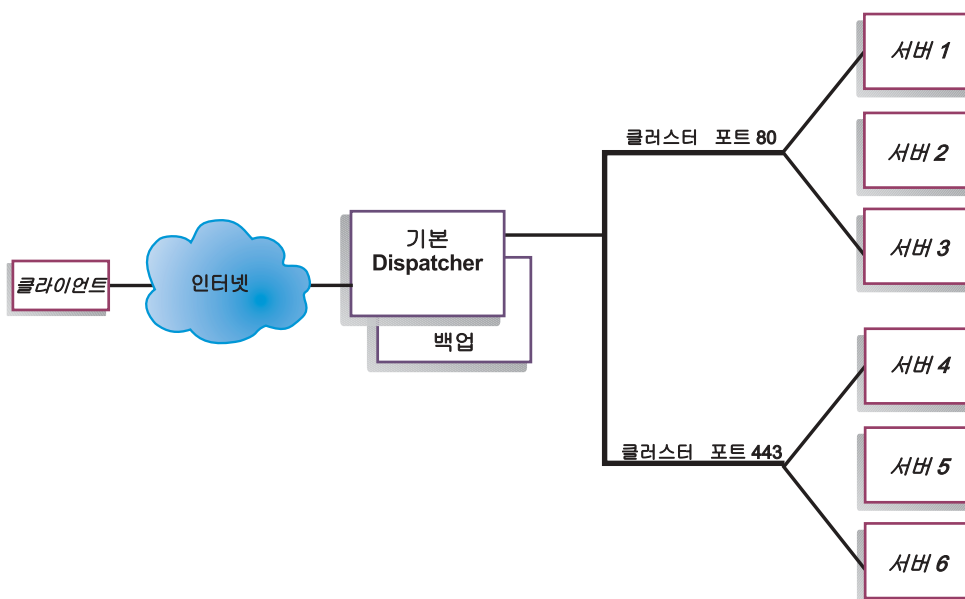


그림 13. 고가용성을 사용하는 Dispatcher 예제

고가용성 기능은 두 번째 Dispatcher 시스템의 사용을 포함합니다. 첫 번째 Dispatcher 시스템은 단일 Dispatcher 구성에 있는 것처럼 모든 클라이언트 통신에 대한 로드 밸

런스를 수행합니다. 두 번째 Dispatcher 시스템은 먼저 “상태”를 모니터하고 첫 번째 Dispatcher 시스템에 장애가 발생하는 것을 감지할 경우 로드 밸런스 작업을 인계 받습니다.

두 시스템 각각에 기본 또는 백업의 특정 역할이 지정됩니다. 기본 시스템은 진행 기준으로 연결 데이터를 백업 시스템으로 전송합니다. 기본 시스템이 활성화(로드 밸런스 중)인 동안 백업 시스템은 대기 상태에 있다가 계속 갱신되며 필요에 따라 인계를 받을 준비가 됩니다.

두 시스템 간의 통신 세션은 하트 비트라고 합니다. 하트 비트로 각 시스템은 다른 시스템의 상태를 모니터할 수 있습니다.

백업 시스템에서 활성화된 시스템의 실패를 검출하면, 백업 시스템은 이를 넘겨 받아 로드 밸런스를 시작합니다. 이 때, 두 시스템의 상태는 확보되어 있습니다. 백업 시스템은 활성화 상태가 되고 기본 시스템은 대기 상태가 됩니다.

고가용성 구성에서 기본 및 백업 시스템 모두 동일한 구성을 가진 같은 서버넷에 있어야 합니다.

고가용성 구성에 대해서는 218 페이지의 『고가용성』을 참조하십시오.

상호 고가용성

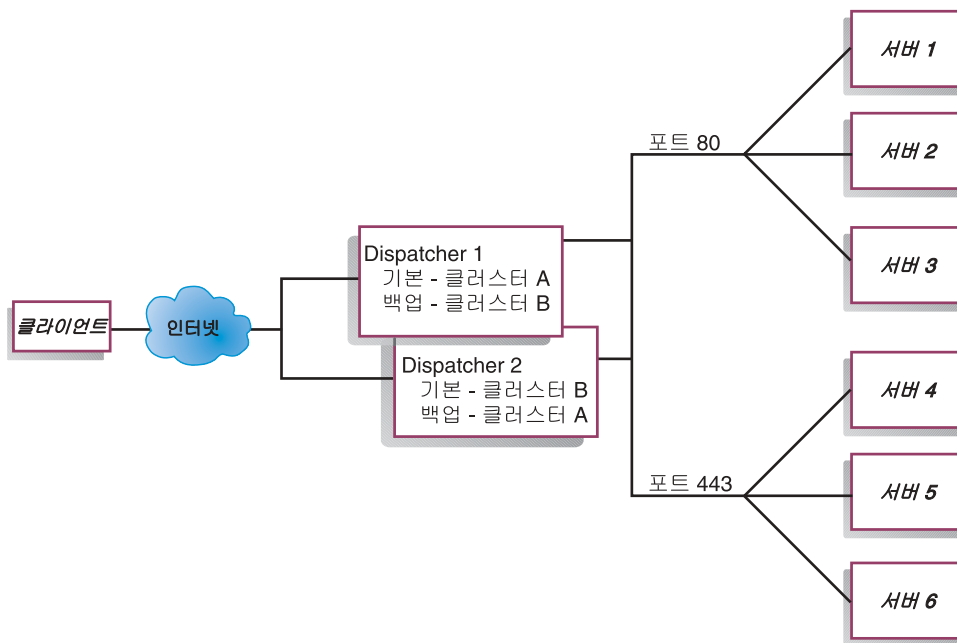


그림 14. 상호 고가용성을 사용하는 Dispatcher 예제

상호 고가용성 기능은 두 Dispatcher 시스템의 사용과 관련됩니다. 두 시스템이 클라이언트 통신에 대한 로드 밸런스를 수행하고 두 시스템이 서로에 대한 백업을 제공합니다.

다. 단순 고가용성 구성에서는 한 시스템만 로드 밸런스를 수행합니다. 상호 고가용성 구성에서는 두 시스템이 모두 클라이언트 통신량의 한 부분에 대한 로드 밸런스를 수행합니다.

상호 고가용성의 경우 클라이언트 통신량이 클러스터 주소 단위로 각 Dispatcher 시스템에 지정됩니다. 각 클러스터는 기본 Dispatcher의 NFA(비전달 주소)로 구성될 수 있습니다. 기본 Dispatcher 시스템은 보통 해당 클러스터에 대한 로드 밸런스를 수행합니다. 장애가 발생하면 다른 시스템이 자체 클러스터와 장애가 발생한 Dispatcher의 클러스터 둘다에 대해 로드 밸런스를 수행합니다.

공유 “클러스터 세트 A”와 공유 “클러스터 세트 B”가 있는 상호 고가용성 구성에 대한 설명은 67 페이지의 그림 14를 참조하십시오. 각 Dispatcher는 기본 클러스터의 패킷을 활성적으로 라우트할 수 있습니다. 임의 Dispatcher가 실패했거나 더이상 기본 클러스터의 패킷을 활성적으로 라우트할 수 없으면 나머지 Dispatcher가 백업 클러스터의 패킷 경로 지정을 대신합니다.

주: 두 시스템은 공유된 클러스터 설정이 동일하게 구성되어야 합니다. 즉, 사용된 포트 및 각 포트 아래의 서버는 두 구성에서 동일해야 합니다.

고가용성 및 상호 고가용성 구성에 대한 자세한 내용은 218 페이지의 『고가용성』을 참조하십시오.

제 7 장 Dispatcher 구성

이 장에 나와 있는 단계를 수행하기 전에 57 페이지의 제 6 장 『Dispatcher 계획』을 참조하십시오. 이 장에서는 Load Balancer의 Dispatcher 컴포넌트에 대한 기본 구성을 작성하는 방법에 대해 설명합니다.

- IPv4 및 IPv6용 Load Balancer를 사용하는 경우 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』를 참조하십시오.
- Load Balancer의 복합 구성에 대해서는 193 페이지의 제 21 장 『Dispatcher, CBR 및 Site Selector에 대한 관리자, 어드바이저 및 Metric Server 기능』 및 215 페이지의 제 22 장 『Dispatcher, CBR 및 Site Selector에 대한 고급 기능』을 참조하십시오.
- 원격 인증 관리, Load Balancer 로그 및 Load Balancer 컴포넌트 사용법에 대한 정보는 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』를 참조하십시오.

주: 제품이 Network Dispatcher로 알려진 이전 버전의 경우, Dispatcher 제어 명령은 `ndcontrol`이었습니다. Dispatcher 제어 명령 이름은 **`dscontrol`**입니다.

구성 task 개요

중요: IPv4 및 IPv6용 Load Balancer를 사용하는 경우, 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』를 참조하십시오.

이 표의 구성 단계를 시작하기 전에, Dispatcher 시스템과 모든 서버 시스템은 네트워크에 연결되어 있고 유효한 IP 주소를 가지며 서로 ping할 수 있어야 합니다.

표 4. Dispatcher 기능의 구성 task

task	설명	관련 정보
Dispatcher 시스템 설정.	로드 밸런스 구성을 설정합니다.	72 페이지의 『Dispatcher 시스템 설정』
로드 밸런스를 수행할 시스템 설정.	루프백 장치에 별명을 지정하고 여분의 라우트를 확인하여 여분의 라우트를 삭제합니다.	79 페이지의 『서버 시스템의 시스템 설정』

구성 방법

Dispatcher 구성에는 세 가지의 기본적인 방법이 있습니다.

- 명령행
- 스크립트
- GUI(Graphical User Interface)

- 구성 마법사

명령행

이 방법은 Dispatcher를 구성하는 가장 직접적인 방법입니다. 명령 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름(클러스터, 서버, 고가용성 명령에 사용)과 파일 이름(파일 명령에 사용)만 예외입니다.

명령행에서 Dispatcher를 시작하려면 다음을 수행하십시오.

1. 명령 프롬프트에서 **dsserver** 명령을 실행하십시오. 서비스를 정지하려면 **dsserver stop**을 입력하십시오.

주: Windows 시스템의 경우, 시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스를 클릭하십시오. **IBM Dispatcher**를 마우스 오른쪽 단추로 클릭한 다음 시작을 선택하십시오. 서비스를 정지하려면 동일한 단계를 수행한 후 정지를 선택하십시오.

2. 그런 다음, 구성 설정을 위해 원하는 Dispatcher 제어 명령을 발행하십시오. 이 책의 절차에서는 명령행을 사용한다고 가정합니다. 명령은 **dscontrol**입니다. 명령에 대한 자세한 내용은 369 페이지의 제 27 장 『Dispatcher 및 CBR 명령어 참조서』를 참조하십시오.

매개변수의 고유한 문자를 입력하여 **dscontrol** 명령 매개변수의 최소화된 버전을 사용할 수 있습니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면 **dscontrol help file** 대신에 **dscontrol he f**를 입력할 수 있습니다.

명령 인터페이스를 시작하려면 **dscontrol** 명령을 실행하여 **dscontrol** 명령 프롬프트를 받으십시오.

명령 인터페이스를 종료하려면 **exit** 또는 **quit** 명령을 실행하십시오.

스크립트

구성 스크립트 파일에 Dispatcher 구성 명령을 입력하여 함께 실행할 수 있습니다. 511 페이지의 『예제 Load Balancer 구성 파일』을 참조하십시오.

주: 스크립트 파일(예: **myscript**)의 콘텐츠를 신속히 실행하려면 다음 명령 중 하나를 사용하십시오.

- 현재 구성을 갱신하려면 스크립트 파일에서 다음 실행 가능 명령을 실행하십시오.

dscontrol file appendload myscript

- 현재 구성을 완전히 바꾸려면 스크립트 파일에서 다음 실행 가능 명령을 실행하십시오.

dscontrol file newload myscript

현재 구성을 스크립트 파일(예: `savescript`)에 저장하려면 다음 명령을 실행하십시오.

```
dscontrol file save savescript
```

이 명령은 `...ibm/edge/lb/servers/configurations/dispatcher` 디렉토리에서 구성 스크립트 파일을 저장합니다.

GUI

GUI(Graphical User Interface)의 일반 명령 및 예제는 500 페이지의 그림 41을 참조하십시오.

GUI를 시작하려면 다음 단계를 따르십시오.

1. `dsserver`가 실행 중인지 확인하십시오.

- AIX, HP-UX, Linux 또는 Solaris 시스템의 경우, 다음 명령을 루트로 실행하십시오.

```
dsserver
```

- Windows 시스템의 경우, `dsserver`는 자동으로 시작되는 서비스로서 실행됩니다.

2. 사용자의 운영 체제에 따라 다음 조치 중 하나를 실행하십시오.

- AIX, HP-UX, Linux 또는 Solaris 시스템의 경우: `ladmin`을 입력하십시오.
- Windows 시스템의 경우: 시작 > 프로그램 > IBM WebSphere > Edge Components > IBM Load Balancer > Load Balancer를 클릭하십시오.

GUI에서 Dispatcher 컴포넌트를 구성하려면 우선 트리 구조에서 **Dispatcher**를 선택해야 합니다. 호스트에 연결한 후에 실행 프로그램 및 관리자를 시작할 수 있습니다. 또한 포트와 서버가 들어 있는 클러스터를 작성하고 관리자에 대한 어드바이저를 시작할 수 있습니다.

GUI를 사용하면 **dscontrol** 명령으로 수행할 수 있는 모든 작업을 수행할 수 있습니다. 예를 들어, 명령행을 사용하여 클러스터를 정의하려면 **dscontrol cluster add cluster** 명령을 입력하십시오. GUI에서 클러스터를 정의하려면 실행 프로그램을 마우스 오른쪽 단추로 클릭하여 표시되는 팝업 메뉴에서 클러스터 추가를 마우스 왼쪽 단추로 클릭하십시오. 팝업 창에 클러스터 주소를 입력한 후 **확인**을 클릭하십시오.

호스트 팝업 메뉴에 있는 새 구성 로드 옵션(현재 구성을 완전히 바꾸기 위한 옵션) 및 현재 구성에 추가 옵션(현재 구성을 갱신하기 위한 옵션)을 사용하여 기존 Dispatcher 구성 파일을 로드할 수 있습니다. 호스트 팝업 메뉴에 있는 다른 옵션인 구성 파일 저장 옵션을 사용하여 Dispatcher 구성을 파일에 정기적으로 저장해야 합니다. GUI의 맨 위에 있는 파일 메뉴를 사용하여 현재 호스트 연결을 파일로 저장하거나 모든 Load Balancer 컴포넌트에 걸쳐 기존 파일의 연결을 복원할 수 있습니다.

구성 명령은 원격으로 실행될 수도 있습니다. 자세한 내용은 282 페이지의 『원격 메소드 호출(RMI)』을 참조하십시오.

GUI에서 명령을 실행하려면 GUI 트리에서 **호스트** 노드를 강조표시하고 **호스트** 팝업 메뉴에서 **명령 전송....**을 선택하십시오. 명령 입력 필드에서는 **executor report**와 같이 실행하려는 명령을 입력하십시오. 현재 세션에서 실행된 명령의 결과 및 히스토리가 제공된 창에 나타납니다.

Load Balancer 창 오른쪽 상단 구석의 물음표를 클릭하면, 도움말에 액세스할 수 있습니다.

- **도움말: 필드 레벨** — 각 필드 및 기본값을 설명합니다.
- **도움말: 수행 방법** — 해당 화면에서 수행할 수 있는 작업이 나열되어 있습니다.
- **InfoCenter** — 제품 정보에 대한 중앙집중화된 액세스를 제공합니다.

GUI 사용에 대한 자세한 내용은 499 페이지의 부록 A 『GUI: 일반 명령』을 참조하십시오.

구성 마법사를 사용한 구성

구성 마법사를 사용 중이면, 다음 단계를 수행하십시오.

1. Dispatcher에서 dserver를 시작하십시오.

- AIX, HP-UX, Linux 또는 Solaris 시스템의 경우, 다음을 루트 사용자로서 실행하십시오.

dserver

- Windows 시스템의 경우, dserver는 자동으로 시작되는 서비스로서 실행됩니다.

2. Dispatcher의 마법사 기능 **dswizard**를 시작하십시오.

마법사는 Dispatcher 컴포넌트의 기본 구성 작성 프로세스를 단계별로 안내합니다. 사용자에게 사용자 네트워크에 관해 질문하며, 서버 그룹간의 통신량 로드 밸런스를 위한 Dispatcher의 클러스터 설치를 안내합니다.

Dispatcher 시스템 설정

Dispatcher 시스템을 설정하려면 루트 사용자(AIX, HP-UX, Linux 또는 Solaris 시스템의 경우) 또는 Windows 시스템에서는 관리여야 합니다.

지원되는 모든 플랫폼에서 Load Balancer에 결합 배치된 서버가 있을 수 있습니다. 결합 배치는 Load Balancer가 로드 밸런스를 유지하고 있는 서버 시스템에 실제로 상주할 수 있음을 의미할 뿐입니다.

Dispatcher 시스템의 경우 mac 전달 메소드를 사용할 경우 최소한 두 개의 유효한 IP 주소가 필요합니다. cbr 또는 nat 전달 메소드의 경우 최소한 세 개의 유효한 IP 주소가 필요합니다.

- Dispatcher 시스템에 고유한 IP 주소

이 IP 주소는 Dispatcher 시스템의 기본 IP 주소로서 비전달 주소(NFA)라고 합니다. 이 주소는 기본적으로 **hostname** 명령으로 리턴되는 주소와 같아야 합니다. Telnet 을 사용하여 원격 구성을 수행하거나 SNMP 서버에이전트에 액세스하는 작업과 같이 관리용으로 시스템을 연결하려면 이 주소를 사용하십시오. Dispatcher 시스템이 이미 네트워크에서 다른 시스템을 ping할 수 있으면, 비전달 주소를 설정하기 위해 취해야 할 조치는 더 이상 없습니다.

- 클러스터당 하나의 IP 주소

클러스터 주소는 호스트 이름(예: www.yourcompany.com)과 연관된 주소입니다. 이 IP 주소는 클러스터에서 서버에 연결하기 위해 클라이언트에서 사용합니다. 이 주소는 Dispatcher에서 로드 밸런스된 주소입니다.

- cbr 또는 nat 전달의 경우 리턴 주소에 대한 IP 주소

Dispatcher는 서버에 대한 클라이언트의 요청을 로드 밸런스할 때 출발지 주소로서 리턴 주소를 사용합니다. 그러면 패킷을 클라이언트로 직접 전송하는 대신 서버가 Dispatcher 시스템으로 패킷을 리턴할 수 있도록 합니다. (Dispatcher는 IP 패킷을 클라이언트에 전달하지 않습니다). 서버를 추가할 때 리턴 주소 값을 지정해야 합니다. 서버를 제거하고 다시 추가하지 않으면, 리턴 주소를 수정할 수 없습니다.

Solaris 시스템 전용:

- 기본적으로, Dispatcher는 100Mbps 이더넷 네트워크 인터페이스 카드의 통신량을 로드 밸런스하도록 구성됩니다. 기본 100Mbps 이더넷 어댑터는 ibmlb.conf 파일에 eri 로 지정되어 있습니다. 그러나 다음과 같은 인터페이스 카드의 다른 유형에 대한 지원도 제공됩니다. le, ce, ge, hme, eri, bge, vge, qfe, dfme, fjgi 및 fjge입니다.

예를 들어, 기본 설정을 변경하려면 **/opt/ibm/edge/lb/servers/ibmlb.conf** 파일을 다음과 같이 편집하십시오.

- 10Mbps 이더넷 어댑터를 사용하려면 eri를 le로 바꾸십시오.
- 1Gbps 이더넷 어댑터를 사용하려면 eri를 ge로 바꾸십시오.
- 멀티포트 어댑터를 사용하려면 eri를 qfe로 바꾸십시오.

여러 유형의 어댑터를 지원하려면, ibmlb.conf 파일의 행을 복제하여 사용자 장치 유형에 맞도록 각 행을 수정하십시오.

예를 들어, 두 개의 100Mbps 이더넷 어댑터를 사용하려면 eri 장치를 지정하는 ibmlb.conf 파일의 단일 행이 있어야 합니다.

10Mbps 이더넷 어댑터와 하나의 100Mbps 이더넷 어댑터를 사용하려면, `ibmlb.conf` 파일에 두 행을 지정해야 하는데, 한 행에서는 `le` 장치를 지정하고 다른 한 행에서는 `eri` 장치를 지정합니다.

주: **ibmlb.conf** 파일은 Solaris **autopush** 명령에 대한 입력을 제공하며 **autopush** 명령과 호환 가능해야 합니다.

- 사용자의 시스템에 쓰이는 이더넷 네트워크 인터페이스의 유형을 판별하려면, Solaris 명령 프롬프트에서 다음의 명령을 발행하십시오.

```
ifconfig -a
```

다음의 결과물이 나오는 경우입니다.

```
lo0: flags=2001000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4,VIRTUAL>
      mtu 8232 index 1 inet 127.0.0.1 netmask ff000000
eri0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4>
      mtu 1500 index 2 inet 9.42.93.208
      netmask fffffc00 broadcast 9.42.95.255 ether 0:3:ba:2d:24:45
```

그런 다음, `ibmlb.conf` 파일을 다음과 같이 편집해야 합니다.

```
eri -1 0 ibmlb
```

- Dispatcher 실행 프로그램을 시작하거나 정지하면 `ibmlb.conf` 파일에 나열된 어댑터의 모든 별명이 구성 해제됩니다. 이들 어댑터(Load Balancer의 Dispatcher 컴포넌트에 의해 사용되는 어댑터는 제외)에 대해 별명을 자동으로 다시 구성하려면 **goAliases** 스크립트 파일을 사용합니다. 예제 스크립트는 **...ibm/edge/lb/servers/samples** 디렉토리에 있으며 실행하기 전에 **...ibm/edge/lb/servers/bin**으로 옮겨야 합니다. **goAliases** 스크립트는 Dispatcher 실행 프로그램이 시작하거나 정지할 때 자동으로 실행됩니다.

예를 들어, 클러스터 X 및 Y가 `ibmlb.conf`에 나열된 어댑터의 CBR 컴포넌트가 사용하도록 구성된 경우 클러스터 X 및 Y는 **dscontrol executor start** 또는 **dscontrol executor stop** 명령이 실행될 때 구성 해제됩니다. 이는 원하지 않는 결과일 수 있습니다. 클러스터 X 및 Y가 **goAliases** 스크립트에서 구성된 경우 Dispatcher 실행 프로그램이 시작하거나 정지된 후 클러스터는 자동으로 다시 구성됩니다.

TCP/IP 프로토콜에 대해 IP 전달이 작동되지 않는지 확인하십시오.

75 페이지의 그림 15는 단일 클러스터, 두 개의 포트 및 세 개의 서버로 설정된 Dispatcher 예를 보여줍니다.

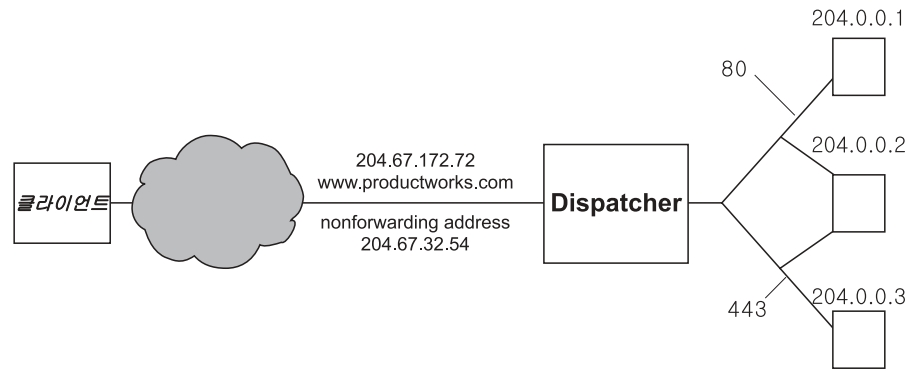


그림 15. Dispatcher 시스템에 대해 필요한 IP 주소 예제

이 절차에서 사용되는 명령에 대한 도움말을 보려면 369 페이지의 제 27 장 『Dispatcher 및 CBR 명령어 참조서』를 참조하십시오.

예제 구성 파일을 보려면, 511 페이지의 『예제 Load Balancer 구성 파일』을 참조하십시오.

1단계. 서버 기능 시작

AIX, HP-UX, Linux 또는 Solaris 시스템: 서버 기능을 시작하려면 **dsserver**를 입력하십시오.

Windows 시스템: 서버 기능이 서비스로서 자동으로 시작됩니다.

주: 기본 구성 파일(default.cfg)은 dsserver를 시작할 때 자동으로 로드됩니다. 사용자가 default.cfg에 Dispatcher 구성을 저장하기로 하면 이 파일에 저장된 모든 내용이 dsserver가 시작되는 다음 번에 자동으로 로드됩니다.

2단계. 실행 프로그램 기능 시작

실행 프로그램 기능을 시작하려면, **dscontrol executor start** 명령을 입력하십시오. 이 때 여러가지 실행 프로그램 설정을 변경할 수도 있습니다. 369 페이지의 제 27 장 『Dispatcher 및 CBR 명령어 참조서』를 참조하십시오.

3단계. 비전달 주소 정의(호스트 이름과 다른 경우)

비전달 주소는 이 시스템에 Telnet 또는 SNMP 사용과 같은 관리용으로 시스템에 연결하는 데 사용됩니다. 기본적으로 이 주소는 호스트 이름입니다.

비전달 주소를 정의하려면, **dscontrol executor set nfa IP_address** 명령을 입력하거나 예제 구성 파일을 편집하십시오. **IP_address**는 기호 이름 또는 IP 주소입니다.

4단계. 클러스터 정의 및 클러스터 옵션 설정

Dispatcher는 클러스터 주소로 전송된 요청을 해당 클러스터의 포트에 구성된 서버로 밸런싱합니다.

클러스터는 기호 이름, 점분리 10진수 주소 또는 와일드 카드 클러스터를 정의하는 특수 주소 0.0.0.0입니다. 클러스터를 정의하려면, **dscontrol cluster add** 명령을 실행하십시오. 클러스터 옵션을 설정하려면, **dscontrol cluster set** 명령을 발행하거나 GUI를 사용하여 명령을 실행하십시오. 와일드 카드 클러스터는 로드 밸런스를 수행할 수신 패킷에 여러 IP 주소를 대응시키는 데 사용할 수 있습니다. 자세한 정보는 253 페이지의 『와일드 카드 클러스터를 사용하여 서버 구성 조합』, 254 페이지의 『와일드 카드 클러스터를 사용하여 방화벽 로드 밸런스 수행』 및 254 페이지의 『투명 프록시의 경우 Caching Proxy가 있는 와일드 카드 클러스터 사용』을 참조하십시오.

5단계. 네트워크 인터페이스 카드의 별명 지정

클러스터가 정의되었으면, 일반적으로 Dispatcher 시스템의 네트워크 인터페이스 카드 중 하나에 클러스터 주소를 구성해야 합니다. 이를 수행하려면, **dscontrol executor configure cluster_address** 명령을 실행하십시오. 이로써, 클러스터 주소와 동일한 서브넷에 속하는 기존의 주소가 있는 어댑터를 찾게 됩니다. 그 다음, 찾은 어댑터와 해당 어댑터에 있는 기존 주소의 넷마스크를 사용하여 클러스터 주소에 운영 체제의 어댑터 구성 명령을 실행합니다. 예를 들어,

```
dscontrol executor configure 204.67.172.72
```

클러스터 주소를 구성하지 않는 환경은고가용성 모드의 대기 서버에 추가된 클러스터나 원격 서버의 역할을 하는 광역 Dispatcher로 추가된 클러스터를 포함합니다. 또한 독립 실행 모드에서 예제 **goIdle** 스크립트를 사용할 경우, **executor configure** 명령을 실행하지 않아도 됩니다. **goIdle** 스크립트에 대해서는 223 페이지의 『스크립트 사용』에서 자세한 정보를 참조하십시오.

드문 경우지만 기존 주소의 어떤 서브넷과도 일치하지 않는 클러스터 주소가 있을 수 있습니다. 이 경우에는 **executor** 구성 명령의 두 번째 양식을 사용하여 인터페이스 이름과 넷마스크를 명시적으로 제공하십시오. **dscontrol executor configure cluster_address interface_name netmask**를 사용하십시오.

몇 가지 예제는 다음과 같습니다.

```
dscontrol executor configure 204.67.172.72 en0 255.255.0.0
(AIX systems)
dscontrol executor configure 204.67.172.72 eth0:1 255.255.0.0
(Linux systems)
dscontrol executor configure 204.67.172.72 eri0 255.255.0.0
(Solaris systems)
dscontrol executor configure 204.67.172.72 en1 255.255.0.0
(Windows systems)
```

Windows 시스템

Windows 시스템에서 **executor configure** 명령의 두 번째 양식을 사용하려면 사용할 인터페이스 이름을 판별해야 합니다. 사용자 시스템에 이더넷 카드가 하나만 있으며

로, 인터페이스 이름은 en0입니다. 토큰링 카드가 하나만 있으면 인터페이스 이름은 tr0입니다. 어느 유형이든 카드가 여러 개 있으면, 카드 매핑을 판별해야 합니다. 다음 단계를 수행하십시오.

1. 다음 명령행에서 실행 프로그램을 시작하십시오. `dscontrol executor start`
2. 명령을 실행하십시오. `dscontrol executor xm 1`

스크린 출력이 표시됩니다. 사용자의 Load Balancer 구성에 쓸 인터페이스 이름을 판별하려면, Number of NIC records를 따르는 행에 있는 Load Balancer 시스템의 IP 주소를 확인하십시오.

Load Balancer 시스템의 IP 주소는 다음과 같이 나열됩니다. `ia->ia_addr`입니다. 연결된 인터페이스 이름은 다음과 같이 나열됩니다. `ifp->if_name`입니다.

`executor configure` 명령이 할당한 인터페이스 이름은 해당 명령에 나열된 인터페이스 이름에 매핑됩니다.

이 매핑 정보를 확보한 후에는 클러스터 주소로 네트워크 인터페이스상의 별명을 작성할 수 있습니다.

ifconfig 명령을 사용하여 클러스터 별명 구성

Linux 또는 UNIX® 시스템의 경우, `executor configure` 명령이 `ifconfig` 명령을 실행합니다.

Solaris 및 HP-UX 시스템: 서버의 IP를 포함하지 않는 IP 주소 목록에 바인드되는 바인드 고유 서버 응용프로그램을 사용할 때는 `ifconfig` 대신 **arp publish** 명령을 사용하여 Load Balancer 시스템에 IP 주소를 동적으로 설정하십시오. 예를 들어,

```
arp -s <cluster> <Load Balancer MAC address> pub
```

6단계. 포트 정의 및 포트 옵션 설정

포트를 정의하려면, `dscontrol port add cluster:port` 명령을 입력하고, 예제 구성 파일을 편집하거나 GUI를 사용하십시오. *Cluster*는 기호 이름 또는 IP 주소입니다. *Port*는 해당 프로토콜에 사용 중인 포트 번호입니다. 이때, 여러 가지 포트 설정을 변경할 수도 있습니다. 포트에 대한 모든 서버를 정의하고 구성해야 합니다. 369 페이지의 제 27 장 『Dispatcher 및 CBR 명령어 참조서』를 참조하십시오.

포트 번호 0은 와일드 카드 포트를 지정하는 데 사용됩니다. 이 포트는 클러스터에 정의된 포트에 정해지지 않은 포트의 통신량을 승인합니다. 와일드 카드 포트는 임의의 포트에 규칙 및 서버를 구성하는 데 사용됩니다. 또한 이 기능은 여러 개의 포트에 대해 동일한 서버 및 규칙 구성이 있는 경우에 사용될 수도 있습니다. 한 포트의 포트 통신량은 다른 포트 통신량의 로드 밸런스 결정에 영향을 줍니다. 와일드 카드 포트를 사용할 시기에 대한 자세한 내용은 255 페이지의 『와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정』을 참조하십시오.

7단계. 로드 밸런스 서버 시스템 정의

로드 밸런스 서버 시스템을 정의하려면 **dscontrol server add cluster:port:server** 명령을 입력하고 예제 구성 파일을 편집하거나 GUI를 사용하십시오. *Cluster* 및 *server* 는 기호 이름 또는 IP 주소입니다. *Port*는 해당 프로토콜에 사용 중인 포트 번호입니다. 로드 밸런스를 수행하기 위해서는 클러스터의 포트에 둘 이상의 서버를 정의해야 합니다.

바인드 고유 서버: Dispatcher 컴포넌트가 바인드 고유 서버로 로드 밸런스 중이면 반드시 클러스터 주소로 바인드되도록 서버를 구성해야 합니다. Dispatcher가 대상 IP 주소를 변경하지 않고 패킷을 전달하기 때문에 패킷이 서버에 도달할 때 패킷에는 대상으로서 클러스터 주소를 여전히 포함합니다. 서버가 클러스터 주소가 아닌 다른 IP 주소로 바인드되도록 구성된 경우, 서버는 클러스터에 지정된 요청을 승인할 수 없습니다.

서버가 바인드 특정인지 판별하려면 **netstat -an** 명령을 발행하여 서버:포트를 확인하십시오. 서버가 바인드 특정이 아니면 해당 명령의 결과는 0.0.0.0:80입니다. 바인드 특정이면 192.168.15.103:80 같은 주소가 표시됩니다.

주: Solaris 및 Linux 시스템의 경우: 어드바이저를 사용할 때 바인드 고유 서버가 결합 배치되어서는 안 됩니다.

다중 주소 결합 배치: 결합 배치 구성에서 결합 배치된 서버 시스템의 주소는 NFA(비 전달 주소)와 동일하지 않아도 됩니다. 시스템이 여러 개의 IP 주소를 갖도록 정의된 경우 다른 주소를 사용할 수 있습니다. Dispatcher 컴포넌트의 경우, 결합 배치 서버 시스템이 **dscontrol server** 명령을 사용하여 결합 배치로 정의되어야 합니다. 결합 배치된 서버에 대한 자세한 내용은 216 페이지의 『결합 배치된 서버 사용』을 참조하십시오.

dscontrol 서버 명령 구문에 대한 자세한 내용은 418 페이지의 『dscontrol server — 서버 구성』을 참조하십시오.

8단계. 관리자 기능 시작(선택)

관리자 기능은 로드 밸런스를 향상시킵니다. 관리자를 시작하려면, **dscontrol manager start** 명령을 입력하고, 예제 구성 파일을 편집하거나 GUI를 사용하십시오.

9단계. 어드바이저 기능 시작(선택)

어드바이저는 요청에 응답할 수 있도록 로드 밸런스 서버 시스템의 기능에 대한 자세한 정보를 관리자에 제공합니다. 어드바이저는 프로토콜마다 고유합니다. 예를 들어 HTTP 어드바이저를 시작하려면 다음 명령을 실행하십시오.

```
dscontrol advisor start http port
```

어드바이저 목록 및 기본 포트에 대해서는 369 페이지의 제 27 장 『Dispatcher 및 CBR 명령어 참조서』를 참조하십시오. 각 어드바이저에 대한 설명은 203 페이지의 『어드바이저 목록』을 참조하십시오.

10단계. 필요한 클러스터 비율 설정

어드바이저를 시작할 경우, 로드 밸런스 결정 시 포함되는 어드바이저 정보에 부여된 중요성의 비율을 수정할 수 있습니다. 클러스터 비율을 설정하려면 **dscontrol cluster set cluster proportions** 명령을 실행하십시오. 자세한 내용은 194 페이지의 『상태 정보에 제공되는 중요성 비율』을 참조하십시오.

서버 시스템의 시스템 설정

해당 조건 중 하나가 참이면 다음 단계를 수행하십시오.

- mac 전달 메소드를 사용 중이고 백엔드 서버 시스템인 경우입니다.
- mac 전달 메소드를 사용 중이고 고가용성 standby 시스템으로 구성된 결합 배치된 서버 시스템인 경우입니다.

주:

1. 시스템이 활성 상태로 변경되는 경우 루프백의 별명 지정을 삭제하는 프로시저가 go* 스크립트에 위치해야 합니다.
2. 고가용성 활성 시스템으로 구성된 경우, 시스템이 대기 상태로 변경되면 루프백 장치의 별명 지정 프로시저가 go* 스크립트에 위치해야 합니다.

mac 전달 메소드를 사용할 때 Dispatcher는 IP 주소를 추가하여 루프백 어댑터를 구성할 수 있도록 하는 백엔드 서버를 통해서만 로드 밸런스를 수행하기 때문에 백엔드 서버는 ARP(Address Resolution Protocol) 요청에 응답하지 않습니다. 이 절의 단계에 따라 로드 밸런스 서버 시스템을 설정하십시오.

1단계. 루프백 장치에 별명 지정

로드 밸런스 서버 시스템이 작동하려면, 클러스터 주소에 루프백 장치(lo0이라고 함)를 설정(또는 별명 지정)해야 합니다. mac 전달 메소드를 사용할 때, Dispatcher 컴포넌트는 TCP 서버 시스템으로 패킷을 전달하기 전에 TCP/IP 패킷의 대상 IP 주소를 변경하지 않습니다. 루프백 장치를 클러스터 주소로 설정하거나 별명을 지정함으로써, 로드 밸런스 서버 시스템은 클러스터 주소로 지정된 패킷을 승인합니다.

네트워크 인터페이스 별명 지정을 지원하는 운영 체제(예: AIX, HP-UX, Linux, Solaris 또는 Windows 시스템)가 있으면, 클러스터 주소에 루프백 장치의 별명을 지정해야 합니다. 별명을 지원하는 운영 체제를 사용하면, 로드 밸런스 서버 시스템을 구성하여 여러 개의 클러스터 주소를 제공할 수 있다는 장점이 있습니다.

중요: Linux 시스템에 대해서는 85 페이지의 『Load Balancer의 mac 전달 사용 시 Linux 루프백 별명 지정 대안』을 참조하십시오.

별명을 지원하지 않는 운영 체제가 있는 서버의 경우, 클러스터 주소에 대해 루프백 장치를 설정해야 합니다.

표 5에 나와 있는 대로 운영 체제의 명령을 사용하여 루프백 장치를 설정하거나 그 별명을 지정하십시오.

표 5. Dispatcher의 루프백 장치(lo0)에 별명 지정 명령

AIX 4.3 또는 이전 버전	ifconfig lo0 alias cluster_address netmask netmask 주: 기본 어댑터의 넷마스크 사용
AIX 5.x	ifconfig lo0 alias cluster_address netmask 255.255.255.255
HP-UX	ifconfig lo0:1 cluster_address up
Linux	다음 명령 중 하나를 선택하십시오. <ul style="list-style-type: none"> • ip -4 addr add cluster_address/32 dev lo • ifconfig lo:1 cluster_address netmask 255.255.255.255 up <p>중요: 시스템에 하나의 구성 명령을 발행했으면 계속해서 동일한 구성 명령(ip 또는 ifconfig)을 사용해야 하며, 그렇지 않은 경우 예기치 못한 결과가 발생할 수 있습니다.</p>
OS/2®	ifconfig lo cluster_address
OS/390®	OS/390 시스템에서 루프백 별명 구성 <ul style="list-style-type: none"> • IP 매개변수 구성원(파일)에서 관리자는 홈 주소 목록에 항목을 작성해야 합니다. 예를 들면, 다음과 같습니다. <pre>HOME ;Address Link 192.168.252.11 tr0 192.168.100.100 ltr1 192.168.252.12 loopback</pre> <ul style="list-style-type: none"> • 루프백에 대해 몇 가지 주소가 정의될 수 있습니다. • 기본적으로 127.0.0.1의 루프백 주소가 구성됩니다.
Solaris 7	ifconfig lo0:1 cluster_address 127.0.0.1 up
Solaris 8, Solaris 9 및 Solaris 10	ifconfig lo0:1 plumb cluster_address netmask netmask up

표 5. Dispatcher의 루프백 장치(lo0)에 별명 지정 명령 (계속)

Windows Server 2003	<ol style="list-style-type: none"> 1. 시작을 클릭한 다음 제어판을 클릭하십시오. 2. 아직 수행되지 않았으면, MS 루프백 어댑터 드라이버를 추가하십시오. <ol style="list-style-type: none"> a. 하드웨어 추가를 클릭하십시오. 그러면 하드웨어 추가 마법사가 실행됩니다. b. 다음을 클릭하십시오. c. 예, 이미 하드웨어에 연결했습니다를 선택한 후, 다음을 클릭하십시오. d. MS 루프백 어댑터가 목록에 있으면 이미 설치되었습니다. 취소를 클릭하여 종료하십시오. e. MS 루프백 어댑터가 목록에 없으면 새 장치 추가를 선택하고 다음을 클릭하십시오. f. 목록에서 하드웨어를 선택하려면 새 하드웨어 찾기 창에서 아니오를 클릭하고 다음을 클릭하십시오. g. 네트워크 어댑터를 선택하고 다음을 클릭하십시오. h. 네트워크 어댑터 선택 패널의 제조업체 목록에서 Microsoft®를 선택한 다음, Microsoft 루프백 어댑터를 선택하십시오. i. 다음을 클릭하고 다시 다음을 클릭하여 기본 설정을 설치하십시오.(또는 디스크 있음을 선택하고 CD를 넣은 다음 CD에서 설치하십시오.) j. 마침을 클릭하여 설치를 끝내십시오. 3. 제어판에서 네트워크를 및 전화 접속 연결을 두 번 클릭하십시오. 4. 장치 이름 “Microsoft 루프백 어댑터”와의 연결을 선택하십시오. 5. 드롭 다운 목록에서 등록 정보를 선택하십시오. 6. 인터넷 프로토콜(TCP/IP)을 선택한 다음 등록 정보를 클릭하십시오. 7. IP 주소를 클릭하십시오. 클러스터 주소로 IP 주소를 입력하고 백엔드 서버의 서브넷 마스크로 서브넷 마스크를 입력하십시오. <p>주: 라우터 주소는 입력하지 마십시오. 로컬 호스트를 기본 DNS 서버로 사용하십시오.</p>
------------------------	---

표 5. Dispatcher의 루프백 장치(lo0)에 별명 지정 명령 (계속)

Windows 2000	<ol style="list-style-type: none"> 1. 시작을 클릭하고 설정을 클릭한 다음 제어판을 클릭하십시오. 2. 아직 수행되지 않았으면, MS 루프백 어댑터 드라이버를 추가하십시오. <ol style="list-style-type: none"> a. 하드웨어 추가/제거를 두 번 클릭하십시오. 이렇게 하면 하드웨어 추가/제거 마법사가 실행됩니다. b. 다음을 클릭하고 장치 추가/문제 해결을 선택한 후 다음을 클릭하십시오. c. 화면이 깜박 거린 다음 하드웨어 장치 선택 창이 나타납니다. d. MS 루프백 어댑터가 목록에 있으면 이미 설치되었습니다. 취소를 클릭하여 종료하십시오. e. MS 루프백 어댑터가 목록에 없으면 새 장치 추가를 선택하고 다음을 클릭하십시오. f. 목록에서 하드웨어를 선택하려면 새 하드웨어 찾기 창에서 아니오를 클릭하고 다음을 클릭하십시오. g. 네트워크 어댑터를 선택하고 다음을 클릭하십시오. h. 네트워크 어댑터 선택 패널의 제조업체 목록에서 Microsoft를 선택한 다음, Microsoft 루프백 어댑터를 선택하십시오. i. 다음을 클릭하고 다시 다음을 클릭하여 기본 설정을 설치하십시오.(또는 디스크 있음을 선택하고 CD를 넣은 다음 CD에서 설치하십시오.) j. 마침을 클릭하여 설치를 끝내십시오. 3. 제어판에서 네트워크를 및 전화 접속 연결을 두 번 클릭하십시오. 4. 장치 이름 “Microsoft 루프백 어댑터”와의 연결을 선택하고 마우스 오른쪽 단추를 클릭하십시오. 5. 드롭 다운 목록에서 등록 정보를 선택하십시오. 6. 인터넷 프로토콜(TCP/IP)을 선택한 다음 등록 정보를 클릭하십시오. 7. IP 주소를 클릭하십시오. 클러스터 주소로 IP 주소를 입력하고 기본 서브넷 마스크(255.0.0.0)로 서브넷 마스크를 입력하십시오. 주: 라우터 주소는 입력하지 마십시오. 로컬 호스트를 기본 DNS 서버로 사용하십시오.
--------------	--

표 5. Dispatcher의 루프백 장치(lo0)에 별명 지정 명령 (계속)

Windows NT®	<ol style="list-style-type: none"> 1. 시작을 클릭한 다음 설정을 클릭하십시오. 2. 제어판을 클릭한 다음 네트워크를 두 번 클릭하십시오. 3. 아직 수행되지 않았으면, MS 루프백 어댑터 드라이버를 추가하십시오. <ol style="list-style-type: none"> a. 네트워크 창에서 어댑터를 클릭하십시오. b. MS 루프백 어댑터를 선택한 다음 확인을 클릭하십시오. c. 프롬프트되면 설치 CD나 디스크를 넣으십시오. d. 네트워크 창에서 프로토콜을 클릭하십시오. e. TCP/IP 프로토콜을 선택한 다음 등록 정보를 클릭하십시오. f. MS 루프백 어댑터를 선택한 다음 확인을 클릭하십시오. 4. 클러스터 주소에 루프백 주소를 설정하십시오. 기본 서브넷 마스크(255.0.0.0)를 승인하고 게이트웨이 주소를 입력하지 마십시오. <p>주: TCP/IP 구성 아래에 MS 루프백 드라이버를 표시하려면 먼저 네트워크 설정을 종료한 후 다시 입력해야 합니다.</p>
-------------	--

2단계. 여분의 라우트 확인

일부 운영 체제에서는 기본 라우트가 작성되었을 수 있으나 이 라우트는 제거되어야 합니다.

- 다음 명령으로 Windows 운영 체제에 여분의 라우트가 있는지 확인하십시오.

```
route print
```

중요: Windows 2003의 여분 라우트는 무시해야 합니다. 별명 지정 이후 라우팅에 문제가 생기면, 다른 넷마스크를 사용하여 별명을 삭제하고 다시 추가하십시오.

- 다음 명령으로 모든 Linux 및 UNIX 시스템에 여분의 라우트가 있는지 확인하십시오.

```
netstat -nr
```

Windows 예제:

1. **route print**를 입력하면 다음 예제와 유사한 테이블이 표시됩니다. (이 예제에서는 기본 넷마스크가 255.0.0.0인 9.67.133.158 클러스터에 대한 여분의 라우트를 찾아 제거하는 것을 보여줍니다).

활성화된 라우트:

Network Address	Netmask	Gateway Address	Interface	Metric
0.0.0.0	0.0.0.0	9.67.128.1	9.67.133.67	1
9.0.0.0	255.0.0.0	9.67.133.158	9.67.133.158	1
9.67.128.0	255.255.248.0	9.67.133.67	9.67.133.67	1
9.67.133.67	255.255.255.255	127.0.0.1	127.0.0.1	1
9.67.133.158	255.255.255.255	127.0.0.1	127.0.0.1	1
9.255.255.255	255.255.255.255	9.67.133.67	9.67.133.67	1
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1

224.0.0.0	224.0.0.0	9.67.133.158	9.67.133.158	1
224.0.0.0	224.0.0.0	9.67.133.67	9.67.133.67	1
255.255.255.255	255.255.255.255	9.67.133.67	9.67.133.67	1

2. “게이트웨이 주소” 컬럼에서 클러스터 주소를 찾으십시오. 여분의 라우트가 있으면, 클러스터 주소가 두 번 나타납니다. 위의 예제에서, 클러스터 주소(9.67.133.158)는 2행과 8행에 나타납니다.
3. 클러스터 주소가 나타나는 각 행에서 네트워크 주소를 찾으십시오. 이 라우트 중 하나가 필요하며, 나머지는 여분의 라우트이므로 삭제해야 합니다. 삭제할 여분의 라우트는 네트워크 주소가 클러스터 주소의 첫 번째 자리에서 시작하며 그 다음에 3개의 0이 옵니다. 위의 예제에서, 여분의 라우트는 2행에 있으며, 네트워크 주소는 **9.0.0.0**입니다.

9.0.0.0	255.0.0.0	9.67.133.158	9.67.133.158	1
---------	-----------	--------------	--------------	---

3단계. 여분의 라우트 삭제

여분의 라우트는 삭제해야 합니다. 표 6에 나와 있는 운영 체제에 따른 명령을 사용하여 여분의 라우트를 삭제하십시오.

예제: 2단계의 “활성 라우트” 예제 테이블에 표시된 여분의 라우트를 삭제하려면 다음을 입력하십시오.

```
route delete 9.0.0.0 9.67.133.158
```

표 6. Dispatcher에 대한 여분의 라우트 삭제 명령

HP-UX	route delete cluster_address cluster_address
Windows	route delete network_address cluster_address (MS-DOS 프롬프트에서) 주: 서버를 재부트할 때마다 여분의 라우트를 삭제해야 합니다. Windows 2003에서는 라우트를 삭제할 수 없습니다. Windows 2003의 여분 라우트는 무시해야 합니다. 별명 지정 이후 라우팅에 문제가 생기면, 다른 넷마스크를 사용하여 별명을 삭제하고 다시 추가하십시오.

75 페이지의 그림 15에 나와 있는 예제를 사용하여 AIX 시스템을 실행 중인 서버 시스템을 설정하면 명령은 다음과 같습니다.

```
route delete -net 204.0.0.0 204.67.172.72
```

4단계. 서버의 올바른 구성 여부 확인

백엔드 서버가 올바르게 구성되었는지 확인하려면, Load Balancer가 실행되지 않고 *cluster*를 구성하지 않았을 때 동일한 서브넷의 서로 다른 시스템에서 다음 단계를 수행하십시오.

1. 다음 명령을 실행하십시오.

```
arp -d cluster
```

2. 다음 명령을 실행하십시오.

```
ping cluster
```

응답이 없어야 합니다. ping 명령에 대한 응답이 있으면, 인터페이스에 대해 클러스터 주소를 ifconfig하지 않았는지 확인하십시오. 클러스터 주소에 발표된 arp 항목이 있는 시스템이 없는지 확인하십시오.

3. 백엔드 서버를 ping하고 즉시 다음 명령을 실행하십시오.

```
arp -a
```

명령 출력에 서버의 MAC 주소가 나타나야 합니다. 다음 명령을 실행하십시오.

```
arp -s cluster server_mac_address
```

4. 클러스터를 ping하십시오. 응답이 있어야 합니다. http, telnet 또는 백엔드 서버가 처리할 클러스터로 주소 지정된 기타 요청을 발행하십시오. 올바르게 작동하는지 확인하십시오.

5. 다음 명령을 실행하십시오.

```
arp -d cluster
```

6. 클러스터를 ping하십시오. 응답이 없어야 합니다.

주: 응답이 있으면 **arp cluster** 명령을 실행하여 잘못 구성된 시스템의 MAC 주소를 받으십시오. 그 다음, 1단계부터 6단계를 반복하십시오.

Load Balancer의 mac 전달 사용 시 Linux 루프백 별명 지정 대안

Linux 시스템의 일부 버전은 인터페이스 출현 시스템에 구성된 IP 주소에 대해 ARP 응답을 발행합니다. 또한 모든 시스템 출현 IP 주소에 기반한 ARP who-has 구성쿼리에 대해 구성된 주소의 인터페이스와 상관 없이 ARP 소스 IP 주소를 선택할 수 있습니다. 이로 인해 모든 클러스터 통신량이 단일 서버에 중간 정도로 경로 지정됩니다.

Dispatcher의 mac 전달 메소드 사용 시, 메커니즘을 사용해 고가용성 및 결합 배치 모두를 사용 중인 경우 결합 배치된 고가용성 대기 시스템을 포함한 백엔드 서버의 스택이 클러스터 주소 지정된 통신량을 승인하도록 해야 합니다.

대부분의 경우 루프백에 클러스터 주소를 별명 지정해야 하므로 백엔드 서버에는 루프백에 별명 지정된 클러스터가 있어야 하며, 고가용성 및 결합 배치를 사용하는 경우 대기 로드 밸런스 서버에는 루프백에 별명 지정된 클러스터가 있어야 합니다.

Linux 시스템이 루프백의 주소를 광고하지 않도록 하려면, 다음 네 가지 해결방안 중 하나를 사용하여 Linux 시스템을 Dispatcher의 mac 전달과 호환 가능하도록 할 수 있습니다.

1. 주소를 광고하지 않는 커널을 사용하십시오. 패킷당 오버헤드를 발생하지 않고 커널당 재구성이 필요 없으므로 권장되는 옵션입니다.
 - SP2(x86) 또는 SP3(다른 아키텍처 모두)이 있는 통합된 Linux 1 / SLES8 및 상위 버전에는 Julian ARP 숨김 패치가 있습니다. 다음 명령으로 클러스터 주소를 별명 지정하기 전에 패치가 항상 유효하도록 하십시오.

```
# sysctl -w net.ipv4.conf.all.hidden=1 net.ipv4.conf.lo.hidden=1
```

이제 다음과 같은 일반적인 방법으로 클러스터를 별명 지정할 수 있습니다.

```
# ifconfig lo:1 $CLUSTER_ADDRESS netmask 255.255.255.255 up
```

- 배분이 종종 기능을 백포트한다는 점에 유의하여 2.4.25 및 2.6.5와 상위 버전에서 쓸 수 있는 `arp_ignore` `sysctl`을 사용하십시오. 다음 명령으로 클러스터 주소를 별명 지정하기 전에 사용 가능하도록 하십시오.

```
# sysctl -w net.ipv4.conf.all.arp_ignore=3  
net.ipv4.conf.all.arp_announce=2
```

그런 다음, 클러스터는 다음 명령으로 별명이 지정됩니다.

```
# ip addr add $CLUSTER_ADDRESS/32 scope host dev lo
```

유사한 명령이 고가용성 결합 배치 구성의 `go*` 스크립트에 있어야 합니다.

- 참고: `sysctl` 사용 시, 설정에 `/etc/sysctl.conf`를 추가하여 재부트해도 해당 설정이 계속 적용되도록 하십시오.

2. IP 테이블을 사용하여 수신 클러스터 통신량 모두를 로컬 호스트에 경로 재지정하십시오. 이 방법을 사용하는 경우 별명과 함께 루프백 어댑터를 구성하지 마십시오. 대신, 다음 명령을 사용하십시오.

```
# iptables -t nat -A PREROUTING -d $CLUSTER_ADDRESS -j REDIRECT
```

이 명령은 Linux 시스템이 각 패킷에 대상 NAT하도록 하여 클러스터 주소를 인터페이스 주소로 변환하게 합니다. 이 방법에는 6.4% 초당 연결 수 처리량 페널티가 있습니다. 이 방법은 지원되는 재고 분배 모두에서 작동하고, 커널 모듈 또는 커널 `patch+build+install`이 필요 없습니다.

3. `noarp` 모듈 버전 1.2.0 또는 상위 버전을 적용하십시오. 커널 소스는 사용 가능해야 하고 바르게 구성되어야 하며, 개발 도구(`gcc`, `gnu make` 등) 사용이 가능해야 합니다. 커널이 업그레이드될 때마다 모듈을 빌드하고 설치해야 합니다. <http://www.masarlabs.com/noarp/>에서 할 수 있습니다. 커널 코드가 수정되지 않으므로 아래에 기술된 해결방안 4번 보다 덜 간섭적이고 오류가 덜 발생합니다. 마찬가지로 루프백에 클러스터 주소가 별명 지정되기 전에 구성되어야 합니다. 예를 들어,

```
# modprobe noarp  
# noarpctl add $CLUSTER_ADDRESS nic-primary-addr
```

`nic-primary-addr`가 클러스터 주소와 동일한 서브넷에 위치한 주소인 경우입니다. 이제 다음과 같은 일반적인 방법으로 클러스터를 별명 지정할 수 있습니다.

```
# ifconfig lo:1 cluster address netmask 255.255.255.255 up
```

주: 고가용성 결합 배치 구성을 위해 `go*` 스크립트에 `noarpctl adds` 및 `dels`가 있어야 합니다. 해당 명령은 활성 Load Balancer가 클러스터 주소에 대해 ARP

를 사용할 수 있도록 하고, 서버로 작동하는 대기 Load Balancer가 클러스터 통신량 모두를 잘못하여(즉 불확정하게) 수신하기 시작하지 않도록 합니다.

4. 다음 웹 사이트에서 Julian 패치를 확보하십시오. <http://www.ssi.bg/~ja/#hidden>입니다. 해당 분배를 사용하는데 적합한 커널의 패치 및 컴파일에 필요한 분배 지시사항을 따르십시오. 결합 배치된 고가용성 Load Balancer의 경우, `uname -r`이 분배 제공된 커널과 일치하도록 하고 분배 커널 `.config` 파일로 시작하도록 하십시오. Julian 숨김 패치로 커널을 빌드, 설치 및 실행한 후 패치 사용을 위해 나열된 첫 번째 해결방안의 지시사항을 따릅니다.

주: 분배 지원과의 밀접한 관련성은 사용자 정의 커널 실행을 위해 존재할 수 있습니다.

제 8 장 IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치

IPv4 및 IPv6용 Load Balancer에서 IPv6의 확장 IP 주소 설치가 가능합니다. IPv4 및 IPv6용 Load Balancer는 Dispatcher 컴포넌트로만 이루어진 독립 설치 이미지입니다. 이 설치 유형은 Dispatcher의 MAC 기반 패킷 전달을 사용하는 사용자의 네트워크에서 구성된 서버에 IPv4 및 IPv6 통신량 로드 밸런스를 제공합니다.

이 장에서는 제품의 IPv4 및 IPv6용 Load Balancer 설치에 있는 Dispatcher에 대한 구성 차이점 및 제한사항에 대해 설명하며 다음의 섹션이 있습니다.

- 90 페이지의 『IPv4 및 IPv6용 Load Balancer에 지원되는 플랫폼』
- 91 페이지의 『IPv4 및 IPv6용 Load Balancer 설치』
- 92 페이지의 『IPv4 및 IPv6용 Load Balancer에 대한 특수 고려사항 및 제한사항』
- 96 페이지의 『IPv4 및 IPv6용 Load Balancer의 IPv6 패킷 처리 사용 가능』
- 97 페이지의 『IPv4 및 IPv6용 Load Balancer의 인터페이스 장치 별명 지정』
- 100 페이지의 『zSeries Linux에 필요한 클러스터 구성 단계』
- 101 페이지의 『IPv4 및 IPv6용 Load Balancer용 Dispatcher 명령(dscontrol)』

Dispatcher 컴포넌트 일반 정보에 대해서는 다음 장을 참조하십시오.

- 네트워크 관리에 필요한 Dispatcher 기능의 개요에 대해서는 21 페이지의 『Dispatcher 컴포넌트 기능』 페이지를 참조하십시오.
- Dispatcher의 로드 밸런스 매개변수 계획에 대한 정보는 57 페이지의 제 6 장 『Dispatcher 계획』을 참조하십시오.
- Dispatcher의 로드 밸런스 매개변수 구성에 대한 정보는 69 페이지의 제 7 장 『Dispatcher 구성』을 참조하십시오.
- 더 많은 고급 기능을 위한 Load Balancer 설정 방법에 대한 정보는 215 페이지의 제 22 장 『Dispatcher, CBR 및 Site Selector에 대한 고급 기능』을 참조하십시오.
- Load Balancer 로그 및 Load Balancer 컴포넌트 사용법에 대한 정보는 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』를 참조하십시오.

IPv4 및 IPv6용 Load Balancer 설치에 있어 Dispatcher 명령(dscontrol)의 구문이란 가지 예외와 일치한다는 점에 유의하십시오. IPv4 및 IPv6용 Load Balancer 사용 시 dscontrol 명령의 분리문자는 콜론(:)이 아니라 앳 마크(@) 기호입니다. 이 문서의 다른 장 전체에서 명령을 참조할 경우, dscontrol 명령의 분리문자로 는 (:) 대신 (@) 기호를 사용하십시오.

IPv4 및 IPv6용 Load Balancer에 지원되는 플랫폼

IPv4 및 IPv6용 Load Balancer 설치에 Windows 2000을 제외한 모든 지원 플랫폼에서 사용 가능합니다.

하드웨어 및 소프트웨어 시스템 요구사항에 대한 정보는 다음 웹 페이지

<http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921>을 참조하십시오.

사용자 영역의 로드 밸런스에 지원되는 플랫폼

Linux 아키텍처 같은 일부 지원 플랫폼에서는 IPv4 및 IPv6용 Load Balancer 설치가 커널 영역이 아닌 사용자 영역에서 로드 밸런스 프로세스를 실행합니다. 이러한 시스템의 경우 더이상 커널 모듈에 대한 종속성이 없습니다.

사용자 영역(kernel free)의 로드 밸런스를 지원하는 플랫폼에 대한 최신 정보는 다음 웹 사이트를 참조하십시오.

<http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921>

사용자 영역에서 로드 밸런스 프로세스를 실행하는 지원 시스템에는 커널 영역에서 로드 밸런스 프로세스를 실행하는 시스템과 다른 구성 프로시저가 일부 존재합니다. 차이점에 대해서는 IPv4 및 IPv6용 Load Balancer에 관한 이 섹션 전체에서 논의됩니다.

특수 Linux 플랫폼 고려사항

zSeries 시스템의 Linux

- **zSeries 시스템의 Linux에는 libstdc++.so.5가 필요합니다.** zSeries 시스템의 Linux에는 올바른 설치를 위해 rpm 패키지 libstdc++.so.5가 있어야 한다는 요구사항이 있으며, 패키지가 없으면 설치할 수 없습니다.
- **qeth/OSA 인터페이스 사용 시 제한사항:** zSeries 시스템의 Linux에는 qeth/OSA 인터페이스 사용 시 제한사항이 있습니다. qeth/OSA 인터페이스를 외부로 고유하게 전달하는 것은 지원되지 않습니다. 그러나 Linux 시스템이 사용자 영역에서 작동하고 Linux 터널링을 지원하므로 해결책이 있습니다.

Linux 터널링 지원

Linux 시스템에서는 IPv4 및 IPv6용 Load Balancer 설치가 IPIP 및 IPGRE 같은 터널 전반에 전달될 수 있습니다. qeth/OSA 인터페이스가 있는 zSeries 시스템의 Linux를 사용하는 경우 Linux 터널은 qeth/OSA 인터페이스를 횡단하도록 정의될 수 있습니다. Linux 시스템은 동일한 또는 다른 qeth/OSA 장치에 있는 시스템 또는 네트워크의 다른 곳에 있는 시스템 사이에서 전달될 수 있습니다.

백엔드 서버 제한사항

Solaris 시스템: 백엔드 Solaris 5.8 서버의 IPv6 통신량 로드 밸런스가 지원되지 않습니다. Solaris 5.8에는 MAC 전달 IPv6 패킷 및 Solaris IPv6 스택과의 호환성이 없습니다. 클러스터가 `ifconfig lo0(loopback)` 명령을 사용하여 Solaris 5.8 백엔드 서버에 구성된 경우, 패킷이 Solaris 5.8 노드에 도달하기는 하지만 승인되지는 않습니다. 그러나 IPv4 및 IPv6용 Load Balancer 설치를 사용하여 IPv4 통신량을 백엔드 Solaris 5.8 서버에 로드 밸런스할 수 있습니다.

z/OS 시스템: 백엔드 z/OS 서버의 IPv6 통신량 로드 밸런스가 지원되지 않습니다. 그러나 IPv4 및 IPv6용 Load Balancer 설치를 사용하여 IPv4 통신량을 백엔드 z/OS 서버에 로드 밸런스할 수 있습니다.

IPv4 및 IPv6용 Load Balancer 설치

IPv4 및 IPv6용 Load Balancer의 설치 단계 및 패키지 이름은 IPv4 서버 주소만 지원하는 Load Balancer에 사용되는 설치 단계 및 패키지 이름과 동일합니다. 그러나 Dispatcher 컴포넌트만 사용 가능하므로 제공되는 Load Balancer 컴포넌트 패키지의 수가 더 적습니다.

시스템 패키징 도구를 사용할 경우, 패키지 설치 권장 순서는 IPv4 및 IPv6용 Load Balancer 설치와 약간 다릅니다. 관리 컴포넌트 패키지는 Dispatcher 컴포넌트 패키지 다음에 설치되어야 합니다. 시스템 패키징 도구를 사용한 IPv4 및 IPv6용 Load Balancer 패키지 설치 권장 순서는 기본, 라이선스, Dispatcher 컴포넌트, 관리, Metric Server 순입니다.

예를 들어 AIX 시스템에서는 다음이 IPv4 및 IPv6용 Load Balancer 패키지의 설치 권장 순서 항목입니다.

- `ibmlb.base.rte`(기본 패키지)
- `ibmlb.lb.license`(CD 설치의 경우, 라이선스 패키지)
- `ibmlb.lb.driver`(장치 드라이버 패키지, AIX에만 해당하는 고유 패키지)
- `ibmlb.disp.rte` and `ibmlb.msg.lang.lb`(메시지 패키지가 있는 Dispatcher 컴포넌트 패키지)
- `ibmlb.admin.rte` and `ibmlb.msg.lang.admin`(메시지 패키지가 있는 관리 패키지)
- `ibmlb.doc.rte` and `ibmlb.msg.en_US.doc`(메시지 패키지가 있는 문서 패키지)
- `ibmlb.ms.rte`(Metric Server 패키지)

IPv4 및 IPv6용 Load Balancer를 설치하기 전에 이전 Load Balancer를 설치 제거해야 합니다. 동일한 시스템에 두 개의 Load Balancer를 설치할 수 없습니다.

제품 설치 지시사항에 대해서는 33 페이지의 제 4 장 『Load Balancer 설치』를 참조하십시오.

IPv4 및 IPv6용 Load Balancer에 대한 특수 고려사항 및 제한사항

Dispatcher 컴포넌트는 IPv4만 지원하는 Load Balancer 설치에 있는 Dispatcher 컴포넌트와 사용 가능한 기능을 많이(전부는 아님) 제공합니다. 다음 주제는 IPv4 및 IPv6용 Load Balancer가 제공되는 Dispatcher에 대한 특수 구성 차이점 및 기능적 제한사항에 대한 것입니다.

IPv6 링크 로컬 주소 구성

Load Balancer 구성의 각 시스템에는 IPv6 주소 지정을 사용한 IPv6 링크 로컬 주소가 있어야 합니다.

링크 로컬 주소는 IPv6에 대한 이웃 발견 통신량에 쓰이는 주소입니다. Load Balancer 시스템 및 백엔드 서버에 이 주소가 없으면 이웃 발견이 일어나지 않으며 시스템은 서로에게 알려지지 않습니다. IPv6용 Load Balancer는 Load Balancer 구성에 있는 각 시스템 인터페이스에 구성된 링크 로컬 IPv6 주소가 없으면 통신량을 전달할 수 없습니다.

동일한 클러스터/서버 쌍

IPv4 및 IPv6용 Load Balancer 구성 시, 모든 서버는 클러스터와 동일해야 합니다. 예를 들어 Cluster1이 IPv4 주소로 정의된 경우, Cluster1의 모든 서버는 IPv4여야 합니다. Cluster2가 IPv6 주소로 정의된 경우, Cluster2에서 정의된 모든 서버는 IPv6여야 합니다. 그리고 클라이언트가 IP 패킷 전송에 사용하는 프로토콜은 클러스터 IP 포맷과 일치해야 합니다.

IPv4 및 IPv6이 혼합된 클라이언트 환경을 지원하기 위해서는 각 논리 클러스터 정의에 대해 두 개의 실제 클러스터(IPv4 클러스터 및 IPv6 클러스터)가 정의되어야 합니다. Load Balancer는 IPv4 패킷을 전송하는 클라이언트를 클러스터에 대해 구성된 IPv4 주소를 사용하는 논리 클러스터에 라우트합니다. Load Balancer는 IPv6 패킷을 전송하는 클라이언트를 클러스터에 대해 구성된 IPv6 주소를 사용하는 논리 클러스터에 라우트합니다.

지원되지 않는 Dispatcher 기능

57 페이지의 제 6 장 『Dispatcher 계획』에 기술된 Dispatcher 기능 중 다수 및 215 페이지의 제 22 장 『Dispatcher, CBR 및 Site Selector에 대한 고급 기능』에 기술된 Dispatcher 기능이 IPv4 및 IPv6용 Load Balancer에서 사용 가능합니다.

다음은 IPv4 및 IPv6용 Load Balancer에서 지원하지 않는 Dispatcher 기능 요약 목록입니다.

- cbr 전달 메소드
- nat 전달 메소드
- 원격 관리

- 규칙 기반 로드 밸런스
- SNMP 서브에이전트
- 광역 로드 밸런스
- UDP 프로토콜 지원

네트워크 관리에 필요한 Dispatcher 기능의 상위 레벨 설명은 21 페이지의 『Dispatcher 컴포넌트 기능』 페이지를 참조하십시오.

어드바이저 구성

시스템에 IPv6 프로토콜을 사용 중이고 어드바이저 사용을 원하는 경우, 프로토콜 파일에 다음 행이 들어 있어야 합니다.

```
ipv6-icmp 58 IPv6-ICMP
```

Linux 및 UNIX 시스템의 경우, 프로토콜 파일은 /etc/protocols 디렉토리에 있습니다. Windows 시스템의 경우, 프로토콜 파일은 C:\windows\system32\drivers\etc\디렉토리에 있습니다.

어드바이저 사용 시 제한사항: 여러 개의 네트워크 어댑터 카드가 설치된 컴퓨터에서 []를 실행 중이고 사용자가 어드바이저 통신량이 특정 어댑터를 통해 플로우되길 원할 경우, 패킷의 소스 IP 주소를 특정 주소로 강제 실행할 수 없습니다. (등록 정보 -DLB_ADV_SRC_ADDR을 IPv4 및 IPv6용 Load Balancer 설치와 함께 사용할 수 없습니다.)

어드바이저에 대한 자세한 정보는 266 페이지의 『어드바이저』를 참조하십시오.

고가용성 구성

시스템에 IPv6 프로토콜을 사용 중이고 고가용성 사용을 원하는 경우, protocol 58 이 프로토콜 파일에 ICMPv6가 되도록 정의되어 있는지 확인해야 합니다. 프로토콜 편집에 대한 정보는 『어드바이저 구성』을 참조하십시오.

IPv4 및 IPv6용 Load Balancer 설치에서는 고가용성 Dispatcher 시스템 구성이 다음 제한사항 또는 특수 고려사항과 함께 지원됩니다.

- 상호 고가용성은 지원되지 않습니다.
- 하트비트 짝(Dispatcher 장애 감지를 위해 기본 Dispatcher 및 대기 Dispatcher 사이에 있는 메커니즘)은 모두 IPv4 포맷이거나 모두 IPv6 포맷이어야 합니다.
- Linux 시스템처럼 사용자 영역에서 실행되는 시스템에 해당합니다. 고가용성 또는 자립형 환경에서는 네트워크 어댑터에 대한 클러스터 주소를 별명 지정하면 안 됩니다.
- Linux 시스템처럼 사용자 영역에서 실행되는 시스템에 해당합니다. go* 및 highavailChange 스크립트는 ...ibm/edge/lb/servers/samples 디렉토리에서

.../ibm/edge/lb/servers/bin 디렉토리로 옮겨져 Dispatcher 시스템에 대한 고가용성 상태 변화를 기록할 수 있지만, 해당 스크립트는 변경할 필요가 없습니다.

- qeth/OSA 인터페이스를 사용하는 zSeries 시스템의 Linux에 해당합니다. 클러스터 주소의 인터페이스 별명 사용에 대한 일반적인 금지가 이 네트워크 인터페이스 유형에는 적용되지 않습니다. 대신 다음의 프로시저를 사용하여 클러스터 통신량이 OSA 전반에 걸쳐 Linux 게스트에 전달되었는지 확인하십시오.
 - go* 스크립트가 필요하고 100 페이지의 『zSeries Linux에 필요한 클러스터 구성 단계』에 지정된 명령을 사용하여 다음과 같이 수정해야 합니다.
 - goActive: ip 및 iptables/ip6tables 명령을 추가하여 클러스터 주소를 구성하고 iptables 규칙을 추가하십시오.
 - goStandby: ip 및 iptables/ip6tables 명령을 추가하여 클러스터 주소를 해제하고 iptables 규칙을 삭제하십시오.
 - goInOp: ip 및 iptables/ip6tables 명령을 추가하여 클러스터 주소를 해제하고 iptables 규칙을 삭제하십시오.
 - goIdle: 이 스크립트는 작성하지 마십시오.

고가용성 기능에 대한 자세한 정보는 218 페이지의 『고가용성』을 참조하십시오.

서버 결합 배치

결합 배치는 요청의 로드 밸런스가 유지되는 서버와 동일한 시스템에 상주할 수 있습니다.

IPv4 및 IPv6용 Load Balancer 설치 사용 시, Windows 시스템 및 사용자 영역에서 실행되는 시스템(예: Linux 시스템)을 제외한 모든 지원 운영 체제에서 결합 배치 기능을 사용할 수 있습니다.

서버 결합 배치에 대한 자세한 정보는 216 페이지의 『결합 배치된 서버 사용』을 참조하십시오.

사용자 영역에서 실행되는 시스템의 동질 관계 기능(Linux)

Linux처럼 사용자 영역에서 실행되는 시스템의 Load Balancer 동질 관계 기능은 커널 영역에서 실행되는 운영 체제의 동질 관계 기능과 다르게 작동합니다.

사용자 영역에서 실행되는 시스템의 경우 Load Balancer는 클라이언트 IP 주소를 백엔드 서버에 맵핑합니다. 패킷의 대상 IP 주소가 클러스터와 일치하고, 대상 포트가 Load Balancer 포트와 일치하며, 소스 IP 주소가 일치한 후에 동질 관계가 성립됩니다.

동질 관계 성립 시 다음에 오는 패킷은 동일한 백엔드 서버로 전송됩니다. 서버 단절 또는 서버 삭제의 이유로 동질 관계가 깨지면, 모든 동질 관계 및 서버 연결이 중단됩니다.

명령행 또는 GUI 클라이언트에 보고된 "연결" 정보도 없습니다. 활성 동질 관계 레코드 수만 사용됩니다.

이 접근법에는 견고한 동질 관계 제공 및 Load Balancer에 보다 효율적이라는 장점이 있습니다.

커널에서 로드 밸런스를 프로세스하는 시스템의 단점은 IP 동질 관계 사용이 CPU 및 메모리 오버헤드를 메커니즘 전달 연결에 추가한다는 점입니다. 사용자 영역에서 로드 밸런스를 프로세스하는 시스템의 경우, 사용되는 동질 관계 메소드가 연결 전달에 비해 메모리 및 CPU 활용을 감소시킵니다.

그리고 사용자 영역에서 실행되는 시스템의 해당 단일 레코드 모델 때문에, 동질 관계와 연결된 stickytime 및 staletimeout 값이 단일 값 — staletimeout으로 통합됩니다. 동질 관계 레코드의 삭제도 연결을 중단시키므로 커널 영역에서 프로세스하는 시스템으로부터 사용자 영역에서 프로세스되는 시스템으로 이주 시, 최대 staletimeout 및 stickytime 값은 사용자 영역 시스템에서 실행되는 Load Balancer의 새 staletimeout으로 사용되어야 합니다.

사용자 영역과 반대되는 커널 영역 시스템 처리의 동질 관계 기능에 대한 일반적인 정보는 236 페이지의 『Load Balancer에 대한 연관 관계 기능 사용법』 페이지를 참조하십시오.

Metric Server 구성

시스템에 IPv6 프로토콜을 사용 중이고 Metric Server 사용을 원하는 경우, protocol 58이 프로토콜 파일에 ICMPv6가 되도록 정의되어 있는지 확인해야 합니다. 프로토콜 편집에 대한 정보는 93 페이지의 『어드바이저 구성』을 참조하십시오.

IPv4 및 IPv6 클러스터 모두를 지원하는 Load Balancer 구성에서, Metric Server 기능을 실행하는 서버는 IPv4 서버로만 또는 IPv6 서버로만 구성될 수 있지만 두 가지 모두로 구성될 수는 없습니다. Metric Server가 특정 프로토콜(IPv4 or IPv6)을 사용하도록 강제하려면, metricserver 스크립트에 Java 등록 정보 java.rmi.server.hostname을 지정하십시오.

중요: Java 등록 정보에 지정된 hostname은 Metric Server의 실제 IP 주소여야 합니다.

UNIX 또는 Linux 시스템: IPV6 주소 2002:92a:8f7a:162:9:42:92:67과 통신하는 Metric Server의 경우, 다음과 같이 metricserver startup 스크립트(/usr/bin 디렉토리에 위치)의 \$LB_CLASSPATH 다음에 Java 등록 정보를 지정하십시오.

```
/opt/ibm/edge/lb/java/jre/bin/java ..... $LB_CLASSPATH
-Djava.rmi.server.hostname=2002:92a:8f7a:162:9:42:92:67
com.ibm.internet.nd.sma.SMA_Agent
$LB_RMI_PORT $LOG_LEVEL $LOG_SIZE $LOG_DIRECTORY $KEYS_DIRECTORY
$SCRIPT_DIRECTORY &
```

Windows 시스템: IPv6 주소 2002:92a:8f7a:162:9:42:92:67과 통신하는 Metric Server의 경우, 다음과 같이 metricserver.cmd 파일(C:\winnt\system32 디렉토리에 위치)을 편집해야 합니다.

```
start/min /wait %IBMLBPATH%\java\jre\bin\java
-Djava.rmi.server.hostname=2002:92a:8f7a:162:9:42:92:67
-Djava.net.preferIPv4Stack=false
-Djava.net.preferIPv6Stack=true
-Xrs -cp
%LB_CLASSPATH% com.ibm.internet.nd.sma.SMA_Agent
%RMI_PORT% %LOG_LEVEL% %LOG_SIZE% %LOG_DIRECTORY% %KEYS_DIRECTORY%
%SCRIPT_DIRECTORY%
goto done

:stop
%IBMLBPATH%\java\jre\bin\java
-Djava.rmi.server.hostname=2002:92a:8f7a:162:9:42:92:67
-Djava.net.preferIPv4Stack=false
-Djava.net.preferIPv6Stack=true
-cp %LB_CLASSPATH% com.ibm.internet.nd.sma.SMA_AgentStop %RMI_PORT%
:done
```

자세한 내용은 211 페이지의 『Metric Server』를 참조하십시오.

IPv4 및 IPv6용 Load Balancer의 IPv6 패킷 처리 사용 가능

AIX, Linux 및 Windows 시스템: 실행 프로그램(dscontrol executor start) 시작 전에 명령행에서 루트로 다음이 발행되어야 합니다.

- AIX 시스템의 경우: autoconf6

시스템 재부트 후에도 IPv6 패킷을 인터럽트 받지 않고 처리하려면 /etc/rc.tcpip 파일을 편집하고 다음 행의 주석을 제거한 후 -A 플래그를 추가합니다.

```
start /usr/sbin/autoconf6 " " -A
```

- Linux 시스템의 경우: modprobe ipv6
- Windows 시스템의 경우: netsh interface ipv6 install

해당 명령은 각각의 운영 체제에서 IPv6 패킷 처리를 가능하게 합니다. 이 명령은 한번만 발행하십시오. 이후에는 원하는 만큼 실행 프로그램을 시작 및 중지할 수 있습니다.

해당 시스템에 IPv6 패킷 처리를 사용 가능하게 하는 명령을 발행하지 않으면 실행 프로그램이 시작되지 않습니다.

HP-UX 및 Solaris 시스템: ifconfig 명령을 사용하여 Dispatcher가 IPv6 패킷을 검사하도록 IPv6 주소를 설정하고 인터페이스를 구성해야 합니다. 실행 프로그램(dscontrol executor start) 시작 전에 명령행에서 루트로 다음을 발행하십시오.

- HP-UX 시스템의 경우:


```
ifconfig device inet6 up
```

- Solaris 시스템의 경우:

```
ifconfig device inet6 plumb
ifconfig device inet6 address/prefix up
```

해당 명령을 발행하지 않으면 실행 프로그램이 시작되기는 하나 IPv6 패킷을 볼 수 없습니다.

IPv4 및 IPv6용 Load Balancer의 인터페이스 장치 별명 지정

네트워크 인터페이스 카드(NIC)에 클러스터 주소를 구성하기 위해 `dscontrol executor configure cluster_address` 명령을 발행할 수 있습니다. `dscontrol executor configure` 명령은 운영 체제의 어댑터 구성 명령(예: `ifconfig`, `dsconfig`(IPv6만 해당) 또는 `ip` 명령)을 실행합니다. 그밖에 Dispatcher 시스템의 NIC를 별명 지정하려면, `executor configure` 명령 대신 운영 체제의 어댑터 구성 명령을 직접 발행할 수 있습니다.

주: Linux 시스템처럼 사용자 영역에서 실행되는 시스템의 경우 — `dscontrol executor configure`, `ip` 또는 `ifconfig` 명령을 사용하여 클러스터 주소를 구성하면 안 됩니다. Load Balancer는 네트워크에 클러스터 주소를 고유하게 광고합니다. 그리고 클러스터 주소는 인터페이스에 별명 지정되어 나타나지 않습니다. 이것이 표준입니다.

그러나 이 표준은 qeth/OSA 인터페이스를 사용하는 zSeries Linux에는 적용되지 않습니다. 이 플랫폼의 경우, 클러스터 주소를 구성합니다. 100 페이지의 『zSeries Linux에 필요한 클러스터 구성 단계』에서 자세한 내용을 참조하십시오.

로드 밸런스 중인 서버에 루프백(lo0) 장치를 별명 지정하려면 운영 체제의 어댑터 구성 명령을 사용해야 합니다.

IPv4 및 IPv6용 Load Balancer 설치의 경우 다음 명령을 사용하여 네트워크 인터페이스 및 루프백 장치를 별명 지정할 수 있습니다(`interface_name`).

AIX(5.x) 시스템에서,

- IPv6 주소의 경우:

```
ifconfig interface_name inet6 cluster_address/prefix_length alias
```

로드 밸런스 중인 서버에 루프백 장치를 별명 지정하는 예제입니다.

```
ifconfig lo0 inet6 2002:4a::541:56/128 alias
```

- IPv4 주소의 경우: 변경되지 않습니다. 로드 밸런스 중인 서버에 루프백 장치를 별명 지정하려면 80 페이지의 표 5 페이지를 참조하십시오.

HP-UX 시스템에서,

- IPv6 주소의 경우:

```
ifconfig interface_name:alias inet6 cluster_address up prefix prefix_length
```

로드 밸런스 중인 서버에 루프백 장치를 별명 지정하는 예제입니다.

```
ifconfig lo0:1 inet6 3ffe:34::24:45 up prefix 128
```

- IPv4 주소의 경우: 변경되지 않습니다. 로드 밸런스 중인 서버에 루프백 장치를 별명 지정하려면 80 페이지의 표 5 페이지를 참조하십시오.

Linux 시스템에서,

- IPv6 또는 IPv4 주소의 경우:

```
ip -version addr add cluster_address/prefix_length dev lo
```

로드 밸런스 중인 서버에 루프백 장치를 별명 지정하는 예제입니다.

```
ip -6 addr add 3ffe:34::24:45/128 dev lo
```

```
ip -4 addr add 12.42.38.125/32 dev lo
```

주: 또한 ifconfig 명령을 사용할 수도 있습니다. ifconfig 명령을 사용하여 루프백 장치를 별명 지정하려면 80 페이지의 표 5 페이지를 참조하십시오.

시스템에 하나의 구성 명령을 발행했으면 계속해서 동일한 구성 명령(**ip** 또는 **ifconfig**)을 사용해야 하며, 그렇지 않은 경우 예기치 못한 결과가 발생할 수 있습니다.

Solaris 8, 9 및 10 시스템에서,

- IPv6 주소의 경우:

```
ifconfig interface_name inet6 addif cluster_address/prefix_length up
```

로드 밸런스 중인 서버에 루프백 장치를 별명 지정하는 예제입니다.

```
ifconfig lo0 inet6 addif 3ffe:34::24:45/128 up
```

- IPv4 주소의 경우: 변경되지 않습니다. 로드 밸런스 중인 서버에 루프백 장치를 별명 지정하려면 80 페이지의 표 5 페이지를 참조하십시오.

Windows 2003 시스템(Windows 2000 및 Windows NT는 IPv6를 지원하지 않음)의 경우입니다.

- IPv4 주소의 경우: 변경되지 않습니다. 로드 밸런스 중인 서버에 루프백 장치를 별명 지정하려면 80 페이지의 표 5 페이지를 참조하십시오.
- IPv6 주소의 경우:

1. ipconfig /all 명령을 사용하여 루프백 장치의 인터페이스 이름을 판별하십시오. 해당 명령은 Microsoft Loopback Adapter의 설명으로 연결을 찾습니다. 다

음의 예제는 Microsoft Loopback Adapter가 이더넷 어댑터 Local Area Connection 2인 경우 ipconfig /all 명령의 결과물이며, 해당 연결은 Local Area Connection 2가 됩니다.

Windows IP Configuration

```
Host Name . . . . . : ndserv10
Primary Dns Suffix . . . . . : rtp.raleigh.ibm.com
Node Type . . . . . : Unknown
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No
DNS Suffix Search List. . . . . : rtp.raleigh.ibm.com
```

Ethernet adapter Local Area Connection 2:

```
Connection-specific DNS Suffix . :
Description . . . . . : Microsoft Loopback Adapter
Physical Address. . . . . : 02-00-4C-4F-4F-50
DHCP Enabled. . . . . : No
IP Address. . . . . : 9.42.92.158
Subnet Mask . . . . . : 255.255.252.0
IP Address. . . . . : 9.42.92.159
Subnet Mask . . . . . : 255.255.252.0
IP Address. . . . . : 2002:92a:8f7a:162:9:42:92:160
IP Address. . . . . : 2002:92a:8f7a:162:9:42:92:159
IP Address. . . . . : fe80::4cff:fe4f:4f50%4
Default Gateway . . . . . :
DNS Servers . . . . . : 127.0.0.1
                        fec0:0:0:ffff::1%1
                        fec0:0:0:ffff::2%1
                        fec0:0:0:ffff::3%1
```

2. netsh 명령을 사용하여 루프백에 클러스터 주소를 추가하십시오. 예를 들어,

```
netsh interface ipv6 add address "Local Area Connection 2"
2002:92a:8f7a:162:9:42:92:161
```

3. ipconfig /all 명령을 다시 발행하여 루프백 어댑터에 추가된 주소를 확인하십시오. 예를 들어,

Ethernet adapter Local Area Connection 2:

```
Connection-specific DNS Suffix . :
Description . . . . . : Microsoft Loopback Adapter
Physical Address. . . . . : 02-00-4C-4F-4F-50
DHCP Enabled. . . . . : No
IP Address. . . . . : 9.42.92.158
Subnet Mask . . . . . : 255.255.252.0
IP Address. . . . . : 9.42.92.159
Subnet Mask . . . . . : 255.255.252.0
IP Address. . . . . : 2002:92a:8f7a:162:9:42:92:161
IP Address. . . . . : 2002:92a:8f7a:162:9:42:92:160
IP Address. . . . . : 2002:92a:8f7a:162:9:42:92:159
IP Address. . . . . : fe80::4cff:fe4f:4f50%4
Default Gateway . . . . . :
DNS Servers . . . . . : 127.0.0.1
                        fec0:0:0:ffff::1%1
                        fec0:0:0:ffff::2%1
                        fec0:0:0:ffff::3%1
```

4. netsh interface ipv6 show interface 명령을 사용하여 시스템 인터페이스 모두에 대한 전달이 가능하도록 하십시오. Local Area Connection 이름으로 나열된 인터페이스에는 전달 사용 가능 IP가 있어야 합니다. 예를 들어,

```
netsh interface ipv6>show interface
Querying active state...
```

Idx	Met	MTU	State	Name
6	2	1280	Disconnected	Teredo Tunneling Pseudo-Interface
5	0	1500	Connected	Local Area Connection
4	0	1500	Connected	Local Area Connection 2
3	1	1280	Connected	6to4 Pseudo-Interface
2	1	1280	Connected	Automatic Tunneling Pseudo-Interface
1	0	1500	Connected	Loopback Pseudo-Interface

```
netsh interface ipv6>set interface "Local Area Connection"
forwarding=enabled
Ok.
```

```
netsh interface ipv6>set interface "Local Area Connection 2"
forwarding=enabled
Ok.
```

OS/2 시스템에서,

- IPv6 또는 IPv4 주소의 경우: 변경되지 않습니다. 로드 밸런스 중인 서버에 루프백 장치를 별명 지정하려면 80 페이지의 표 5 페이지를 참조하십시오.

zSeries Linux에 필요한 클러스터 구성 단계

zSeries Linux의 경우 Load Balancer 설치를 위해 다음의 추가 구성 단계가 필요합니다.

1. ip 또는 ifconfig 명령을 사용하여 클러스터 주소를 구성하십시오.

IPv6 또는 IPv4 주소의 경우:

```
ip -version addr add cluster_address/prefix_length dev device
```

예를 들어,

```
ip -4 addr add 12.42.38.125/24 dev eth0
ip -6 addr add 3ffe:34::24:45/64 dev eth0
```

2. iptables 규칙을 추가하여 클러스터 주소에 정해진 수신 패킷을 삭제하십시오.

IPv4 주소의 경우:

```
iptables -t filter -A INPUT -d cluster_address -j DROP
```

IPv6 주소의 경우:

```
ip6tables -t filter -A INPUT -d cluster_address -j DROP
```

예를 들어,

```
iptables -t filter -A INPUT -d 12.42.38.125 -j DROP
iptables -t filter -A INPUT -d 3ffe:34::24:45 -j DROP
```

위의 구성을 실행 취소하려면 다음 명령을 사용하십시오.

```
ip -version addr del cluster_address/prefix_length dev device
iptables -t filter -D INPUT -d cluster_address -j DROP
ip6tables -t filter -D INPUT -d cluster_address -j DROP
```

IPv4 및 IPv6용 Load Balancer용 Dispatcher 명령(dscontrol)

IPv4 및 IPv6용 Load Balancer는 모든 컴포넌트 기능을 지원하지 않으므로, 이 설치에 유효한 dscontrol 명령은 IPv4만 지원하는 Load Balancer 설치의 dscontrol 명령의 서브세트입니다. 이 장에서는 명령 구문 차이점에 대해 논의하고 IPv4 및 IPv6용 Load Balancer의 Dispatcher 컴포넌트에 대한 모든 지원 dscontrol 명령을 나열합니다.

명령 구문 차이점

IPv4 및 IPv6용 Load Balancer 설치에 있어 Dispatcher 명령(dscontrol)의 구문이 한 가지 중요한 예외와 일치합니다. IPv4 및 IPv6용 Load Balancer 사용 시 dscontrol 명령의 분리문자는 콜론(:)이 아니라 앳 마크(@) 기호입니다.

IPv6 포맷이 주소 지정 설계에서 콜론을 사용하기 때문에 콜론(:)이 아닌 분리문자를 정의해야 합니다.

다음은 앳 마크(@) 분리문자를 사용하는 dscontrol 명령의 예입니다.

- IPv6 클러스터(30::100)에 있는 포트 80에 IPv6 서버(30::200)를 추가
dscontrol server add 30::100@80@30::200
- IPv4 클러스터(192.4.40.30)에 있는 포트 80에 IPv4 서버(192.4.40.35) 추가
dscontrol server add 192.4.40.30@80@192.4.20.35

중요: 이 문서 전체에서 명령을 참조할 경우 dscontrol 명령의 분리문자로는 (:) 대신 (@) 기호를 사용하십시오.

지원되는 dscontrol 명령

모든 dscontrol 명령에 관한 자세한 정보 및 예제는 369 페이지의 제 27 장 『Dispatcher 및 CBR 명령어 참조서』 페이지를 참조하십시오.

다음은 IPv4 및 IPv6용 Load Balancer 설치의 Dispatcher에 대해 지원되는 명령 요약입니다.

- dscontrol advisor
 - 모든 인수 및 키 값이 유효합니다.

- 자세한 명령 구문 설명은 372 페이지의 『dscontrol advisor — 어드바이저 제어』 페이지를 참조하십시오.
- dscontrol binlog
 - 모든 인수 및 키 값이 유효합니다.
 - 자세한 명령 구문 설명은 378 페이지의 『dscontrol binlog — 2진 로그 파일 제어』 페이지를 참조하십시오.
- dscontrol cluster
 - 모든 인수가 유효합니다. 유효한 키 값은 address 및 proportions입니다.
 - 자세한 명령 구문 설명은 379 페이지의 『dscontrol cluster — 클러스터 구성』 페이지를 참조하십시오.
- dscontrol executor
 - 모든 인수가 유효합니다. 인수 set의 경우 유효한 키 값은 nfa, hatimeout 및 hasynctimeout입니다.

Linux 시스템처럼 사용자 영역에서 실행되는 시스템에 해당합니다.

- configure 및 unconfigure를 제외한 모든 인수가 유효합니다. 클러스터 주소를 시스템 스택에 별명 지정하면 안 됩니다.
- 인수 set의 경우 유효한 키 값은 nfa 및 hatimeout입니다.
- configure 인수의 경우, *netmask* 대신 *prefix_length*를 사용해야 합니다.

IPv6의 경우 접두부 길이는 IPv6 주소의 네트워크 부분에 있는 비트수를 나타냅니다. 접두부 길이는 호스트 주소에서 네트워크 주소를 묘사합니다.

IPv4의 경우, 접두부를 다음과 같이 판별합니다. 서브넷 마스크가 255.255.252.0이면 16진 등가는 FF.FF.FC.0입니다. 2진에서는 값이 11111111 11111111 11111100 00000000입니다. 서브넷 마스크의 1의 계수는 접두부 길이를 판별합니다. 서브넷 마스크에 22개가 있으면 접두부는 22가 됩니다.

executor configure의 구문은 다음과 같습니다.

```
dscontrol executor configure interface_address interface_name prefix_length
```

IPv6 주소 지정의 예제입니다.

```
dscontrol executor configure 2002:092a:8f7a:4226:9:37:240:99 en0 112
```

서브넷 마스크가 255.255.252.0인 경우의 IPv4 주소 지정의 예입니다.

```
dscontrol e config 191.60.20.20 en1 22
```

executor configure 명령은 IPv4 및 IPv6용 Load Balancer 설치의 사용자 영역에서 실행되는 시스템(예: Linux 시스템)에 사용되지 않습니다.

- 자세한 명령 구문 설명은 383 페이지의 『dscontrol executor — 실행 프로그램 제어』 페이지를 참조하십시오.
- dscontrol file
 - 모든 인수 및 키 값이 유효합니다.
 - 자세한 명령 구문 설명은 388 페이지의 『dscontrol file — 구성 파일 관리』 페이지를 참조하십시오.
- dscontrol help
 - host(원격 시스템 구성), rule(규칙 구성) 및 subagent(SNMP 서브에이전트 구성)를 제외한 모든 인수가 유효합니다. host, rule 및 subagent 명령은 지원되지 않습니다.
 - 자세한 명령 구문 설명은 390 페이지의 『dscontrol help — 이 명령의 도움말 표시 또는 인쇄』 페이지를 참조하십시오.
- dscontrol highavailability
 - 모든 인수가 유효합니다. 상호 고가용성이 지원되지 않기 때문에 both를 제외한 모든 키 값이 유효합니다.
 - 자세한 명령 구문 설명은 391 페이지의 『dscontrol highavailability — 고가용성 제어』 페이지를 참조하십시오.
- dscontrol logstatus
 - 모든 인수 및 키 값이 유효합니다.
 - 자세한 명령 구문 설명은 396 페이지의 『dscontrol logstatus — 서버 로그 설정 표시』 페이지를 참조하십시오.
- dscontrol manager
 - version을 제외한 모든 인수가 유효합니다. 모든 키 값이 유효함
 - 자세한 명령 구문 설명은 397 페이지의 『dscontrol manager — 관리자 제어』 페이지를 참조하십시오.
- dscontrol metric
 - 모든 인수 및 키 값이 유효합니다.
 - 자세한 명령 구문 설명은 403 페이지의 『dscontrol metric — 시스템 메트릭 구성』 페이지를 참조하십시오.
- dscontrol port
 - 지원되지 않는 halfopenaddressreport를 제외한 모든 인수가 유효합니다.

다음의 키 값은 dscontrol port 명령의 add 및 set 인수에 유효합니다.

 - staletimeout
 - weightbound
 - stickymask

Linux 시스템처럼 사용자 영역에서 실행되는 시스템의 경우입니다. 다음의 키 값은 `dscontrol port` 명령의 `add` 및 `set` 인수에 유효합니다.

- `staletimeout`
- `weightbound`
- `selectionalgorithm`

`selectionalgorithm`(서버 선택 알고리즘)의 옵션은 다음과 같습니다.

- `connection` – 서버 선택사항은 단순 라운드 로빈 선택사항에 기반(기본값)
- `affinity` – 서버 선택사항은 클라이언트 동질 관계에 기반합니다.

예를 들어,

```
dscontrol port add cluster@port selectionalgorithm affinity
```

- 자세한 명령 구문 설명은 404 페이지의 『`dscontrol port` — 포트 구성
- 』 페이지를 참조하십시오.

- `dscontrol server`
 - 모든 인수가 유효합니다.

다음의 키 값은 `dscontrol server` 명령의 `add` 인수에 유효합니다.

- `address`
- `advisorrequest`
- `advisorresponse`
- `collocated`

`collocated` 키워드는 Windows 시스템 및 사용자 영역에서 실행되는 시스템 (예: Linux 시스템)을 제외한 모든 지원 운영 체제에서 사용 가능합니다.

- `fixedweight`
- `weight`

다음의 키 값은 `dscontrol server` 명령의 `set` 인수에 유효합니다.

- `advisorrequest`
- `advisorresponse`
- `collocated`

`collocated` 키워드는 Windows 시스템 및 사용자 영역에서 실행되는 시스템 (예: Linux 시스템)을 제외한 모든 지원 운영 체제에서 사용 가능합니다.

- `fixedweight`
- `weight`

- 자세한 명령 구문 설명은 418 페이지의 『`dscontrol server` — 서버 구성
- 』 페이지를 참조하십시오.

- `dscontrol set`
 - 모든 인수 및 키 값이 유효합니다.
 - 자세한 명령 구문 설명은 425 페이지의 『`dscontrol set` — 서버 로그 구성
』 페이지를 참조하십시오.
- `dscontrol status`
 - 모든 인수 및 키 값이 유효합니다.
 - 자세한 명령 구문 설명은 426 페이지의 『`dscontrol status` — 관리자 및 어드바이저가 실행 여부 표시
』 페이지를 참조하십시오.

지원되지 않는 **dscontrol** 명령

다음 명령은 IPv4 및 IPv6용 Load Balancer 설치의 Dispatcher에 대해 사용할 수 없습니다.

- `dscontrol host`(원격 시스템 구성)
- `dscontrol rule`(규칙 구성)
- `dscontrol subagent`(SNMP 서브에이전트 구성)

제 3 부 CBR(Content Based Routing) 컴포넌트

이 파트에서는 빠른 시작 구성, 계획 고려사항에 대한 정보를 제공하며 Load Balancer의 CBR 컴포넌트 구성 메소드에 대해 설명합니다. 다음 장을 포함합니다.

- 109 페이지의 제 9 장 『빠른 시작 구성』
- 117 페이지의 제 10 장 『Content Based Routing 계획』
- 123 페이지의 제 11 장 『Content Based Routing 구성』

제 9 장 빠른 시작 구성

빠른 시작 예제에서는 두 개의 웹 서버 간에 웹 통신량을 로드 밸런스하는 Caching Proxy와 함께 CBR을 사용하여 로컬로 연결된 세 대의 워크스테이션을 구성하는 방법을 보여줍니다. (단순성을 위해, 이 예제는 동일한 LAN 세그먼트상의 서버를 예시하지만, CBR에서 동일한 LAN의 서버 사용에 대해서 아무 제한도 없습니다.)

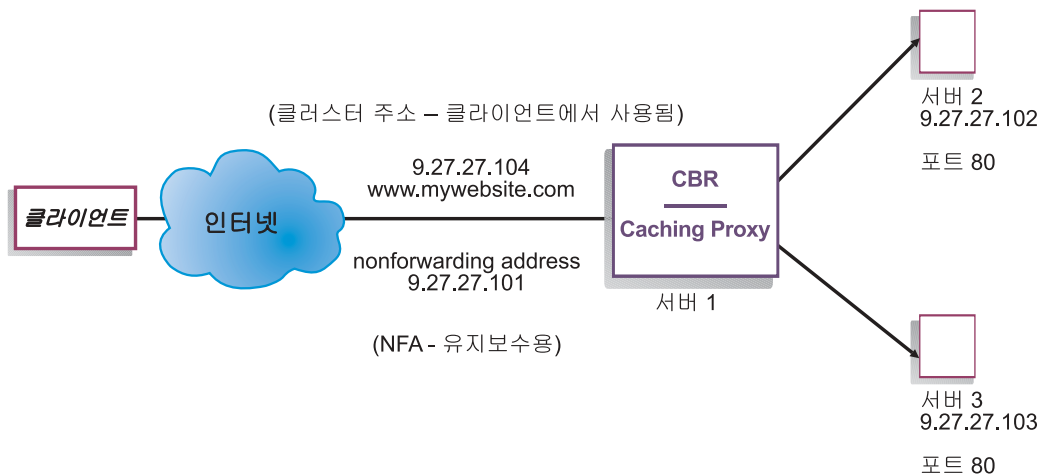


그림 16. 간단한 로컬 CBR 구성

필요한 내용

빠른 시작 예제의 경우 세 대의 워크스테이션과 네 개의 IP 주소가 필요합니다. 하나의 워크스테이션은 CBR 시스템으로 사용되고 나머지 워크스테이션은 웹 서버로 사용됩니다. 웹 서버마다 하나의 IP 주소가 필요합니다. CBR 워크스테이션에는 하나의 실제 주소와 로드 밸런스될 하나의 주소가 필요합니다.

주: CBR(Content Based Routing) 컴포넌트는 64비트 JVM 실행 플랫폼을 제외한 모든 지원 플랫폼에서 사용 가능합니다. 혹은 Load Balancer에 있는 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 Caching Proxy를 사용하지 않고도 CBR을 제공할 수 있습니다. 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 페이지를 참조하십시오.

CBR을 사용하려면 Caching Proxy를 동일한 서버에 설치해야 합니다. CBR에 대한 Caching Proxy를 구성하려면, 128 페이지의 『1단계. CBR을 사용하기 위해 Caching Proxy 구성』을 참조하십시오.

준비 방법

1. 이 예제의 경우, 동일한 LAN 세그먼트에 워크스테이션을 설정하십시오. 세 시스템 간의 네트워크 통신량이 라우터 또는 브리지를 통과해서는 안됩니다.
2. 세 워크스테이션의 네트워크 어댑터를 구성하십시오. 이 예제에서는 다음과 같이 네트워크가 구성되어 있다고 가정합니다.

워크스테이션	이름	IP 주소
1	server1.mywebsite.com	9.27.27.101
2	server2.mywebsite.com	9.27.27.102
3	server3.mywebsite.com	9.27.27.103
Netmask = 255.255.255.0		

각 워크스테이션에는 표준 이더넷 네트워크 인터페이스 카드가 하나만 있습니다.

3. server1.mywebsite.com이 server2.mywebsite.com과 server3.mywebsite.com을 둘 다 ping할 수 있어야 합니다.
4. server2.mywebsite.com과 server3.mywebsite.com이 server1.mywebsite.com을 ping할 수 있어야 합니다.
5. server2.mywebsite.com과 server3.mywebsite.com의 웹 서버가 작동하는지 확인하십시오. 웹 브라우저를 사용하여 **http://server2.mywebsite.com** (예: .../member/index.html)과 **http://server3.mywebsite.com**(예: .../guest/index.html)에서 페이지를 직접 요청하십시오.
6. 이 LAN 세그먼트에 유효한 또 다른 IP 주소를 확보하십시오. 이 주소는 사용자 사이트에 액세스할 클라이언트에 제공할 클러스터 주소입니다. 이 예제에서는 다음을 사용합니다.

Name= www.mywebsite.com
IP=9.27.27.104

CBR 컴포넌트 구성

CBR을 통해 명령행, 구성 마법사 또는 GUI(Graphical User Interface)를 사용하여 구성을 작성할 수 있습니다. 이 빠른 시작 예의 경우, 구성 단계는 명령행을 사용하여 예시됩니다.

주: 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름과 파일 이름의 매개변수 값만 예외입니다.

명령행을 사용한 구성

명령 프롬프트에서 다음 단계를 따르십시오

1. cbrserver를 시작하십시오. 루트 사용자 또는 관리자로 **cbrserver** 명령을 실행하십시오.

주: Windows 플랫폼의 경우: 서비스 패널(시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스)에서 cbrserver(Content Based Routing)을 시작하십시오.

2. CBR의 실행 프로그램 기능을 시작하십시오.

cbrcontrol executor start

3. Caching Proxy를 시작하십시오. (Caching Proxy는 실행 프로그램 기능을 시작한 후 언제라도 시작될 수 있습니다.)

ibmproxy

주: Windows 플랫폼의 경우: 서비스 패널에서 Caching Proxy를 시작할 수도 있습니다(시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스).

4. 클러스터(클라이언트가 연결할 호스트 이름, 웹 사이트)를 CBR 구성에 추가하십시오.

cbrcontrol cluster add www.mywebsite.com

5. CBR 시스템의 네트워크 인터페이스 카드에 웹 사이트에 대한 클러스터 주소 (9.27.27.104)를 추가하십시오. 130 페이지의 『5단계. 네트워크 인터페이스 카드 별명 지정(선택적)』에서 자세한 정보를 참조하십시오.
6. Http 프로토콜 포트를 CBR 구성에 추가하십시오.

cbrcontrol port add www.mywebsite.com:80

7. 각 웹 서버를 CBR 구성에 추가하십시오.

cbrcontrol server add www.mywebsite.com:80:server2.mywebsite.com

cbrcontrol server add www.mywebsite.com:80:server3.mywebsite.com

8. 콘텐츠 규칙을 사용자의 CBR 구성에 추가하십시오. (콘텐츠 규칙은 URL 요청을 구별하고 서버 또는 서버 세트 중 하나로 전송하는 방법을 정의합니다.)

cbrcontrol rule add www.mywebsite.com:80:memberRule type content pattern uri=*/member/*

cbrcontrol rule add www.mywebsite.com:80:guestRule type content pattern uri=*/guest/*

이 예에서, 콘텐츠 규칙을 사용하여 www.mywebsite.com 웹 사이트에 대한 클라이언트 요청이 URI 요청 경로의 디렉토리에 따라 여러 서버로 전송됩니다. 507 페이지의 부록 B 『콘텐츠 규칙(패턴) 구문』에서 자세한 정보를 참조하십시오.

9. 서버를 사용자의 규칙에 추가하려면 다음 명령을 실행하십시오.

```
cbrcontrol rule useserver www.mywebsite:80:memberRule
server2.mywebsite.com
```

```
cbrcontrol rule useserver www.mywebsite:80:guestRule
server3.mywebsite.com
```

CBR은 콘텐츠 기반 규칙에 따라 로드 밸런스를 수행합니다. **/member/**를 포함하는 URL 요청을 가진 클라이언트는 **server2.mywebsite.com**에 지정됩니다. **/guest/**를 포함하는 URL 요청을 가진 클라이언트는 **server3.mywebsite.com**에 지정됩니다.

10. CBR의 관리자 기능을 시작하십시오.

```
cbrcontrol manager start
```

11. CBR의 어드바이저 기능을 시작하십시오.

```
cbrcontrol advisor start http 80
```

CBR에서는 실패한 웹 서버로 클라이언트 요청이 전송되지 않았음을 확인합니다.

로컬로 연결된 서버의 기본 구성을 완료했습니다.

구성 검사

구성이 작동되고 있는지 확인하십시오.

1. 웹 브라우저에서 **http://www.mywebsite.com/member/index.htm** 위치로 이동하십시오. 페이지가 표시되면 구성이 작동하는 것입니다.
2. 웹 브라우저에서 페이지를 재로드하십시오.
3. 다음 명령의 결과를 보십시오.

```
cbrcontrol server report www.mywebsite.com:80:
```

두 서버의 총 연결 컬럼은 “2”까지 추가해야 합니다.

GUI(Graphical User Interface)를 사용한 구성

CBR GUI 사용에 대한 정보는 126 페이지의 『GUI』 및 499 페이지의 부록 A 『GUI: 일반 명령』을 참조하십시오.

구성 마법사를 사용한 구성

CBR 마법사 사용에 대한 정보는 127 페이지의 『구성 마법사』를 참조하십시오.

클러스터, 포트, 서버 구성의 유형

사용자 사이트를 지원하기 위해 CBR을 구성할 수 있는 여러 가지 방법이 있습니다. 사용자 사이트에 모든 고객을 연결할 하나의 호스트 이름만 있는 경우, 서버의 단일 클러스터를 정의할 수 있습니다. 이들 각 서버에서 CBR이 통신할 포트를 구성합니다. 53 페이지의 그림 9를 참조하십시오.

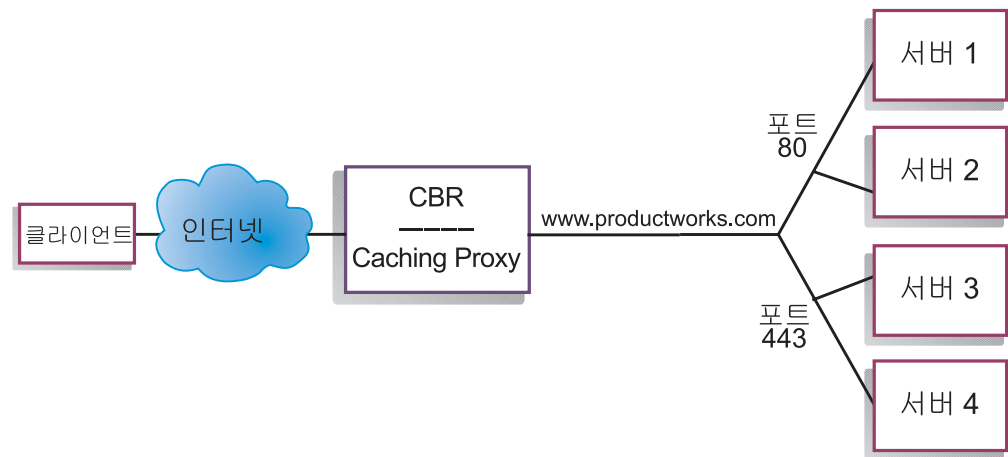


그림 17. 단일 클러스터와 두 개의 포트 구성된 CBR에 대한 예제

CBR 컴포넌트 예제에서는 하나의 클러스터가 `www.productworks.com`에 정의됩니다. 이 클러스터에는 두 개의 포트가 있습니다(HTTP용 포트 80과 SSL용 포트 443). `http://www.productworks.com`(포트 80)에 요청을 작성하는 클라이언트는 `https://www.productworks.com`(포트 443)을 요청하는 클라이언트와 다른 서버로 이동합니다.

지원되는 각 프로토콜에 공용으로 제공된 많은 서버가 있는 매우 큰 사이트에서는 다른 CBR 구성 방법이 적합합니다. 이 경우, 54 페이지의 그림 10에 표시된 대로 단일 포트이지만 많은 서버가 있는 각 프로토콜마다 클러스터를 정의하고자 할 수 있습니다.

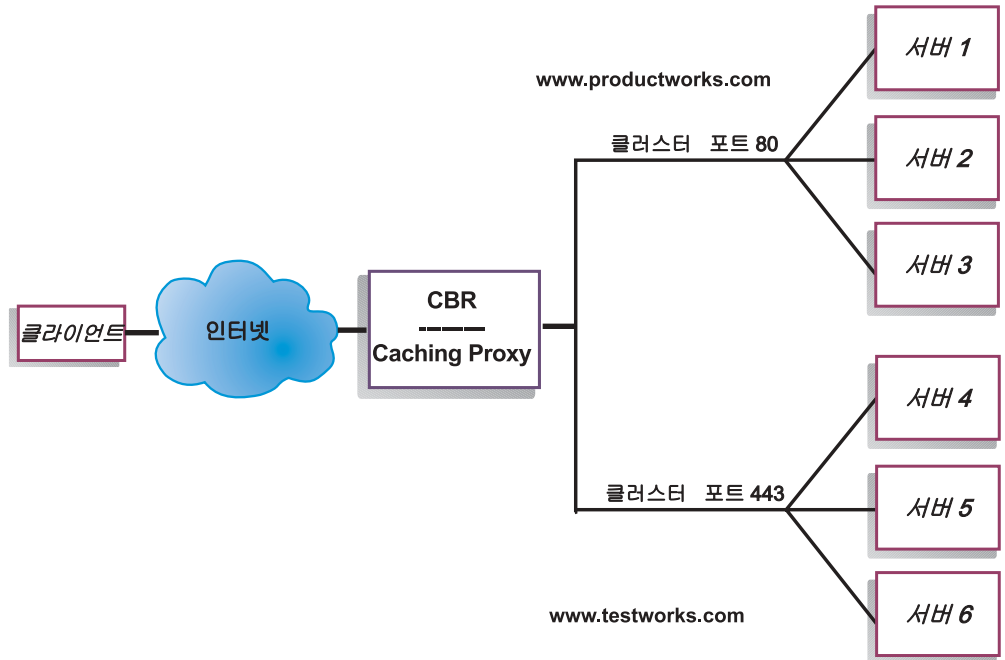


그림 18. 각각 단일 포트인 두 개의 클러스터로 구성된 CBR에 대한 예제

CBR 컴포넌트 예제에서, 두 개의 클러스터는 포트 80(HTTP)의 경우, www.productworks.com, 포트 443(SSL)의 경우, www.testworks.com 사이트에 정의됩니다.

각각 다른 URL로 사이트에 들어가는 여러 회사나 부서에 대해 사이트가 콘텐츠를 호스트할 경우에 CBR을 구성하기 위한 세 번째 방법이 필요합니다. 이 경우에는 55 페이지의 그림 11에 표시된 것처럼 회사나 부서의 클러스터를 각각 정의한 다음, 해당 URL에서 연결을 받을 포트를 정의할 수 있습니다.

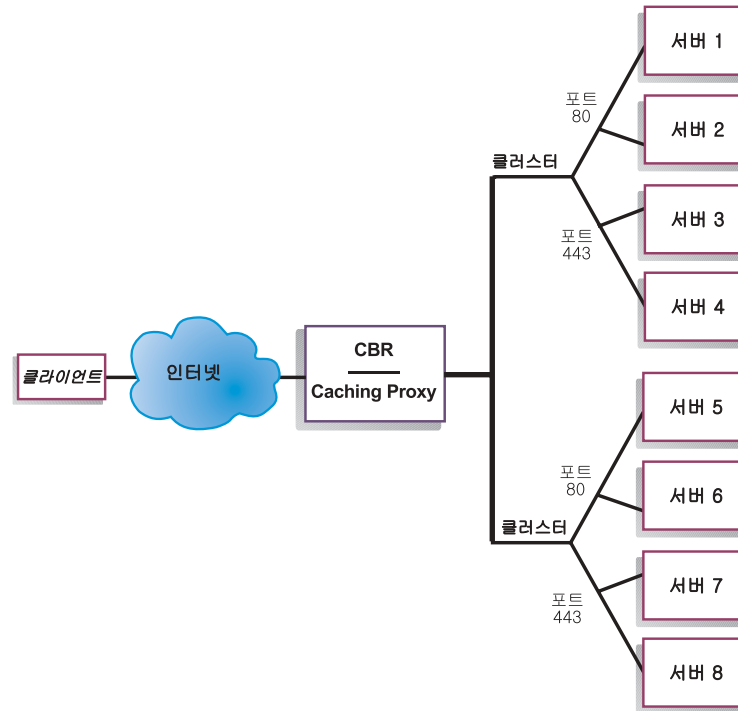


그림 19. 각각 포트가 두 개인 두 개의 클러스터로 구성된 CBR에 대한 예제

CBR 컴포넌트 예제에서, 두 개의 클러스터는 포트 80(HTTP)과 포트 443(SSL)으로 정의되며, 각각 www.productworks.com 및 www.testworks.com 사이트에 정의됩니다.

제 10 장 Content Based Routing 계획

이 장에서는 Caching Proxy를 사용하여 CBR 컴포넌트를 설치 및 구성하기 전에 네트워크 계획표에서 고려해야 할 사항을 설명합니다.

- 네트워크 관리에 필요한 기능의 개요에 대해서는 21 페이지의 제 3 장 『네트워크 관리: 사용할 Load Balancer 기능 결정』 페이지를 참조하십시오.
- CBR 로드 밸런스 매개변수 구성에 대한 정보는 123 페이지의 제 11 장 『Content Based Routing 구성』을 참조하십시오.
- 더 많은 고급 기능을 위한 Load Balancer 설정 방법에 대한 정보는 215 페이지의 제 22 장 『Dispatcher, CBR 및 Site Selector에 대한 고급 기능』을 참조하십시오.
- 원격 인증 관리, Load Balancer 로그 및 Load Balancer 컴포넌트 사용법에 대한 정보는 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』를 참조하십시오.

이 장에는 다음과 같은 섹션이 수록되어 있습니다.

- 『계획 고려사항』
- 119 페이지의 『CBR과 함께 규칙 기반 로드 밸런스 사용』
- 120 페이지의 『완전 보안(SSL) 연결의 로드 밸런스』
- 120 페이지의 『SSL의 클라이언트-투-프록시 및 HTTP의 프록시-투-서버의 로드 밸런스』

계획 고려사항

CBR 컴포넌트를 사용하면 Caching Proxy를 통해 HTTP와 SSL 통신량을 로드 밸런싱하여 요청을 프록시할 수 있습니다. CBR을 사용하면, cbrcontrol 명령을 사용하여 CBR 구성 파일에서 구성하는 서버를 로드 밸런싱할 수 있습니다.

주: CBR(Content Based Routing) 컴포넌트는 64비트 JVM 실행 플랫폼을 제외한 모든 지원 플랫폼에서 사용 가능합니다. 혹은 Load Balancer에 있는 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 Caching Proxy를 사용하지 않고도 CBR을 제공할 수 있습니다. 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 페이지를 참조하십시오.

CBR의 컴포넌트 구조는 Dispatcher와 매우 유사합니다. CBR은 다음과 같은 기능으로 구성됩니다.

- **cbrserver**는 실행 프로그램, 관리자 및 어드바이저에 대한 명령행의 요청을 처리합니다.

- 실행 프로그램은 클라이언트 요청의 로드 밸런스를 지원합니다. CBR 컴포넌트를 사용하려면 실행 프로그램을 시작해야 합니다.
- 관리자에서는 다음에 따라 실행 프로그램에서 사용하는 가중치를 설정합니다.
 - 실행 프로그램의 내부 카운터
 - 어드바이저가 제공하는 서버에서의 피드백
 - Metric Server와 같은 시스템 모니터링 프로그램 피드백

관리자를 사용하는 것은 선택입니다. 그러나 관리자를 사용하지 않으면, 현재 서버 가중치에 따라 가중된 라운드 로빙 스케줄링을 사용하여 로드 밸런스가 수행되며, 어드바이저는 사용 가능하지 않습니다.
- 어드바이저는 관리자를 호출하여 가중치를 적절하게 설정하기 전에 서버를 조회하고 프로토콜별로 결과를 분석합니다. 일반 구성에서 이들 어드바이저를 사용하는 것은 바람직하지 않습니다. 또한 사용자 고유의 어드바이저를 작성하는 옵션도 있습니다. 어드바이저를 사용하는 것은 선택입니다. Load Balancer에서는 Caching Proxy(cachingproxy) 어드바이저를 제공합니다. 199 페이지의 『어드바이저』에서 자세한 정보를 참조하십시오.
- 실행 프로그램, 어드바이저 및 관리자를 구성하고 관리하려면, 명령행(cbrcontrol) 또는 그래픽 사용자 인터페이스(lbadmin)를 사용하십시오.

CBR의 핵심적인 세 기능(실행 프로그램, 관리자 및 어드바이저)은 서버 간의 수신 요청의 밸런스를 조정하고 디스패치하기 위해 상호작용합니다. 로드 밸런스 요청과 함께 실행 프로그램은 새로운 연결 및 활성 연결을 모니터링하고, 이러한 정보를 관리자에 제공합니다.

다른 유형의 콘텐츠에 대한 요청 로드 밸런스

CBR 컴포넌트는 클라이언트 요청 콘텐츠와 일치하는 일반 표현식에 기초하여 요청을 처리하는 서버 세트를 지정할 수 있게 합니다. CBR 컴포넌트를 사용하면, 서로 다른 콘텐츠가 서로 다른 서버 세트에 의해 처리되도록 사용자 사이트의 파티션을 나눌 수 있습니다. 사이트에 액세스하는 클라이언트는 이러한 파티션 구분 과정을 볼 수 없습니다.

더 나은 응답 시간을 위해 사이트의 콘텐츠 나누기

사이트를 나누는 한 가지 방법은 일부 서버는 cgi 요청만을 처리하고 다른 서버 세트는 다른 모든 요청을 처리하도록 지정하는 것입니다. 이렇게 하면 집약적 cgi 스크립트 계산으로 인해 표준 HTML 통신에서 서버 속도가 느려지지 않으므로, 전반적인 응답 시간이 줄어듭니다. 또한 이 설계를 사용하면, 표준 요청에 대해 보다 강력한 워크스테이션을 지정할 수 있게 됩니다. 이 경우 모든 서버를 업그레이드하는 수고를 들이지 않고도 클라이언트 응답 시간을 개선할 수 있습니다. 또한 cgi 요청에 대해 보다 강력한 워크스테이션을 지정할 수 있습니다.

사이트의 파티션을 나누면, 등록을 요하는 페이지를 액세스하는 클라이언트를 하나의 서버 세트에 지정하고 다른 모든 요청을 두 번째 서버 세트에 지정할 수 있습니다. 그러면 등록을 예약한 클라이언트가 사용할 수 있는 자원을 사용자 사이트의 일반 브라우저에서 입력할 필요가 없어집니다. 또한 보다 강력한 워크스테이션을 사용하여 등록된 클라이언트를 처리할 수 있습니다.

물론 위의 두 방법을 결합하여 보다 융통성 있고 수준 높은 서비스를 구현할 수 있습니다.

웹 서버 콘텐츠의 백업 제공

CBR을 통해 각 유형의 요청에 대해 여러 서버를 지정할 수 있기 때문에, 최적의 클라이언트 응답을 위해 로드 밸런스를 조정할 수 있습니다. 여러 서버가 각 유형의 콘텐츠에 지정되도록 하여 워크스테이션이나 서버가 고장날 경우에 대비할 수 있습니다. CBR은 장애를 인식하고, 서버 세트의 다른 서버로 계속 클라이언트 요청을 전달합니다.

CPU 이용 향상을 위해 복수 Caching Proxy 프로세스 사용

Caching Proxy는 해당 플러그인 인터페이스를 통해 CBR 프로세스와 통신합니다. Caching Proxy가 작동하려면 CBR이 로컬 시스템에서 실행 중이어야 합니다. 로컬 시스템에는 두 개의 별도의 프로세스가 있기 때문에 여러 Caching Proxy 인스턴스를 실행하여 단일 CBR 인스턴스에 대해 작업할 수 있습니다. Caching Proxy 간에 주소 또는 기능을 분리하거나 여러 Caching Proxy가 클라이언트 통신량을 처리하게 함으로써 시스템의 자원 활용도를 향상시키기 위해 이렇게 설정을 구성할 수 있습니다. 프록시 인스턴스는 통신량 요구사항에 어느 것이 가장 적합한지에 따라 다른 포트에서 인식하거나 고유한 IP 주소에 바인드될 수 있습니다.

CBR과 함께 규칙 기반 로드 밸런스 사용

Caching Proxy와 함께 CBR은 지정된 규칙 유형을 사용하여 HTTP 요청을 검사합니다. Caching Proxy는 실행 중일 때 클라이언트 요청을 승인하고 최상의 서버에 대한 CBR 컴포넌트를 조회합니다. 이 조회에서 CBR은 우선순위가 설정된 규칙 세트에 요청을 대응시킵니다. 규칙이 일치하는 경우, 사전 구성된 서버 세트에서 적절한 서버가 선택됩니다. 마지막으로, CBR은 서버가 선택되고 요청이 프록시되었다는 것을 Caching Proxy에게 알립니다.

로드 밸런스를 조정할 클러스터를 정의한 후, 해당 클러스터로의 모든 요청이 하나의 서버를 선택하는 하나의 규칙을 가지고 있는지 확인해야 합니다. 특정 요청과 일치하는 규칙을 찾을 수 없으면, 클라이언트는 Caching Proxy로부터 오류 페이지를 받게 됩니다. 모든 요청이 일부 규칙과 일치하도록 하는 가장 간단한 방법은 우선순위가 높은 번호에서 "항상 참"인 규칙을 작성하는 것입니다. 이 규칙에 사용되는 서버가 우선순위가 낮은 규칙에서 명시적으로 처리하지 않는 모든 요청을 처리할 수 있는지 확인하십시오 (주: 우선순위가 낮은 규칙이 처음에 평가됩니다).

자세한 정보는 226 페이지의 『규칙 기반 로드 밸런스 구성』을 참조하십시오.

완전 보안(SSL) 연결의 로드 밸런스

Caching Proxy가 있는 CBR은 프록시에서 SSL 서버(프록시에서 서버쪽으로)로의 전송을 지원할 뿐만 아니라 클라이언트에서 프록시(클라이언트에서 프록시쪽으로)로의 SSL 전송을 받을 수도 있습니다. CBR 구성에서 클라이언트로부터 SSL 요청을 받도록 서버의 SSL 포트를 정의하면, CBR을 사용하여 보안(SSL) 서버를 로드 밸런싱하는 완전 보안 사이트를 유지할 수 있습니다.

CBR에 대한 다른 `ibmproxy.conf` 파일 변경사항 외에 프록시 대 서버 측면에서 SSL 암호화를 사용 가능하게 하려면 구성 명령문을 Caching Proxy의 `ibmproxy.conf` 파일에 추가해야 합니다. 형식은 다음과 같아야 합니다.

```
proxy uri_pattern url_pattern address
```

여기에서 `uri_pattern`은 일치할 패턴이고(예: `/secure/*`), `url_pattern`은 대체 URL이며(예: `https://clusterA/secure/*`), `address`는 클러스터 주소입니다(예: `clusterA`).

SSL의 클라이언트-투-프록시 및 HTTP의 프록시-투-서버의 로드 밸런스

또한 CBR은 요청을 HTTP 서버로 프록시하기 전에 Caching Proxy를 사용하여 클라이언트로부터 SSL 전송을 받은 다음, SSL 요청을 암호 해독할 수 있습니다. CBR이 SSL의 클라이언트-프록시와 HTTP의 프록시-서버를 지원하기 위해 `cbrcontrol` 서버 명령에 선택적 키워드 **mapport**가 있습니다. 이 키워드를 사용하면, 서버의 포트가 클라이언트에서 수신하는 포트와 다르다는 것을 표시할 수 있습니다. 다음은 클라이언트의 포트가 443(SSL)이고 서버의 포트가 80(HTTP)인 `mapport` 키워드를 사용하여 포트를 추가하는 예제입니다.

```
cbrcontrol server add cluster:443 mapport 80
```

`mapport`의 포트 번호는 양의 정수 값이 될 수 있습니다. 기본값은 클라이언트에서 수신하는 포트의 포트 번호 값입니다.

CBR이 포트 443(SSL)에 구성된 서버에 대한 HTTP 요청을 권고할 수 있어야 하므로, 특수 어드바이저 `ssl2http`가 제공됩니다. 이 어드바이저는 포트 443(클라이언트에서 수신하는 포트)에서 시작되고 해당 포트에 구성된 서버에 권고합니다. 두 개의 클러스터가 구성되고 클러스터마다 포트 443이 있고, 서버가 서로 다른 `mapport`로 구성되면 어드바이저의 단일 인스턴스가 해당 포트를 열 수 있습니다. 다음은 이 구성에 대한 예제입니다.

```
Executor
  Cluster1
    Port:443
      Server1 mapport 80
      Server2 mapport 8080
  Cluster2
    Port:443
```

```
Server3 mapport 80
Server4 mapport 8080
Manager
Advisor ssl2http 443
```

제 11 장 Content Based Routing 구성

이 장의 단계를 수행하기 전에, 117 페이지의 제 10 장 『Content Based Routing 계획』을 참조하십시오. 이 장에서는 Load Balancer의 CBR 컴포넌트에 대한 기본 구성을 작성하는 방법에 대해 설명합니다.

- Load Balancer의 복합 구성에 대해서는 193 페이지의 제 21 장 『Dispatcher, CBR 및 Site Selector에 대한 관리자, 어드바이저 및 Metric Server 기능』 및 215 페이지의 제 22 장 『Dispatcher, CBR 및 Site Selector에 대한 고급 기능』을 참조하십시오.
- 원격 인증 관리, Load Balancer 로그 및 Load Balancer 컴포넌트 사용법에 대한 정보는 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』를 참조하십시오.

구성 task 개요

이 테이블의 구성 단계를 시작하기 전에, CBR 시스템과 모든 서버 시스템이 네트워크에 연결되어 있고 유효한 IP 주소를 가지며 서로 ping할 수 있어야 합니다.

주: CBR(Content Based Routing) 컴포넌트는 64비트 JVM 실행 플랫폼을 제외한 모든 지원 플랫폼에서 사용 가능합니다. 혹은 Load Balancer에 있는 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 Caching Proxy를 사용하지 않고도 CBR을 제공할 수 있습니다. 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 페이지를 참조하십시오.

표 7. CBR 컴포넌트의 task 구성

task	설명	관련 정보
CBR 시스템 설정	요구사항에 대해 찾기	128 페이지의 『CBR 시스템 설정』
로드 밸런스를 수행할 시스템 설정	로드 밸런스 구성을 설정합니다.	132 페이지의 『7단계. 로드 밸런스 서버 시스템 정의』

구성 방법

다음은 Load Balancer의 CBR 컴포넌트에 대해 기본 구성을 작성하기 위한 네 가지 기본 방법입니다.

- 명령행
- 스크립트
- GUI(Graphical User Interface)
- 구성 마법사

CBR을 사용하려면, Caching Proxy가 설치되어 있어야 합니다.

주: Caching Proxy는 기본적으로 설치 후 자동으로 시작하는 서비스입니다. CBR 서버 기능(cbrserver)을 시작하기 전에 Caching Proxy를 정지시킨 후 Caching Proxy 서비스가 자동이 아닌 수동으로 시작되도록 수정하십시오.

- Linux 또는 UNIX 시스템의 경우: `ps -ef | grep ibmproxy` 명령을 사용하여 프로세스 ID를 찾은 후 `kill process_id` 명령으로 프로세스를 종료하여 Caching Proxy를 정지하십시오.
- Windows 시스템의 경우: 서비스 패널에서 Caching Proxy를 정지하십시오.

명령행

이 방법은 CBR을 구성하는 가장 직접적인 방법입니다. 명령 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름(예: 클러스터 및 서버 명령에 사용됨) 및 파일 이름은 예외입니다.

명령행에서 CBR을 시작하려면 다음을 수행하십시오.

- Linux 또는 UNIX 시스템의 경우: 루트 사용자로, 명령 프롬프트에서 **cbrserver** 명령을 실행하십시오. 서비스를 정지하려면, **cbrserver stop**을 실행하십시오.

Windows 시스템의 경우: 시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스를 클릭하십시오. **IBM Content Based Routing**을 마우스 오른쪽 단추로 클릭한 다음 시작을 선택하십시오. 서비스를 정지하려면 동일한 단계를 수행한 후 정지를 선택하십시오.

- 그 다음, 구성을 설정하기 위해 원하는 CBR 제어 명령을 실행하십시오. 이 책의 절차에서는 명령행을 사용한다고 가정합니다. 명령은 **cbrcontrol**입니다. 명령에 대한 자세한 내용은 369 페이지의 제 27 장 『Dispatcher 및 CBR 명령어 참조서』를 참조하십시오.
- Caching Proxy를 시작하십시오. 명령 프롬프트에서 **ibmproxy** 명령을 실행하십시오(Caching Proxy를 시작하기 전에 실행 프로그램을 시작해야 함).

주: Windows 플랫폼의 경우: 서비스 패널(시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스)에서 Caching Proxy를 시작하십시오.

cbrcontrol 명령 매개변수의 축약된 버전을 입력할 수 있습니다. 매개변수의 고유한 문자만 입력해야 합니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면, **cbrcontrol help file** 대신에 **cbrcontrol he f**를 입력할 수 있습니다.

명령 인터페이스를 시작하려면 **cbrcontrol**을 실행하여 cbrcontrol 명령 프롬프트를 받으십시오.

명령 인터페이스를 종료하려면 **exit** 또는 **quit** 명령을 실행하십시오.

주:

1. Windows 플랫폼에서 Dispatcher 컴포넌트의 dsserver가 자동으로 시작됩니다. CBR만을 사용하고 Dispatcher 컴포넌트를 사용하지 않는 경우 다음과 같이 dsserver가 자동으로 시작되지 않게 할 수 있습니다.
 - a. 서비스 창에서 IBM Dispatcher를 마우스 오른쪽 단추로 클릭하십시오.
 - b. 등록 정보를 선택하십시오.
 - c. 시동 유형 필드에서 수동을 선택하십시오.
 - d. 확인을 클릭하고, 서비스 창을 닫으십시오.
2. cbrcontrol>> 프롬프트가 아니라 운영 체제의 명령 프롬프트에서 CBR(Content Based Routing)을 구성할 경우 다음 문자 사용 시 주의하십시오.

() 오른쪽 괄호 및 왼쪽 괄호

& 앰퍼샌드

| 수직 막대

! 느낌표

* 별표

운영 체제 셸에서 이들 문자를 특수 문자로 해석하여 cbrcontrol이 평가하기도 전에 이들 문자를 대체 텍스트로 변환할 수 있습니다.

위에 나열한 특수 문자는 **cbrcontrol rule add** 명령에서 선택적인 문자이며 콘텐츠 규칙의 패턴을 지정할 때 사용됩니다. 예를 들어, 다음 명령은 cbrcontrol>> 프롬프트를 사용할 경우에만 유효합니다.

```
rule add 10.1.203.4:80:cbr_prod_rule_ek type content
pattern uri=/nipoe/*
```

동일한 명령이 운영 체제의 프롬프트에서도 작동하려면 다음과 같이 패턴에 큰따옴표(" ")가 있어야 합니다.

```
cbrcontrol rule add 10.1.203.4:80:cbr_prod_rule_ek type content
pattern "uri=/nipoe/*"
```

큰따옴표를 사용하지 않으면 규칙이 CBR에서 저장될 때 일부 패턴이 잘릴 수 있습니다. cbrcontrol>> 명령 프롬프트를 사용할 경우 따옴표가 지원되지 않습니다.

스크립트

구성 스크립트 파일에 CBR 구성 명령을 입력하여 함께 실행할 수 있습니다.

주: 스크립트 파일(예: myscript)의 콘텐츠를 신속히 실행하려면 다음 명령 중 하나를 사용하십시오.

- 현재 구성을 갱신하려면 스크립트 파일에서 다음 실행 가능 명령을 실행하십시오.

```
cbrcontrol file appendload myscript
```

- 현재 구성을 완전히 바꾸려면 스크립트 파일에서 다음 실행 가능 명령을 실행하십시오.

```
cbrcontrol file newload myscript
```

현재 구성을 스크립트 파일(예: `savescript`)에 저장하려면 다음 명령을 실행하십시오.

```
cbrcontrol file save savescript
```

이 명령은 `...ibm/edge/lb/servers/configurations/cbr` 디렉토리에서 구성 스크립트 파일을 저장합니다.

GUI

GUI(Graphical User Interface)의 일반 명령 및 예제는 500 페이지의 그림 41을 참조하십시오.

GUI를 시작하려면 다음 단계를 따르십시오.

1. `cbrserver`가 실행 중인지 확인하십시오. 루트 사용자 또는 관리자로서, 명령 프롬프트에서 **cbrserver**를 실행하십시오.
2. 사용자의 운영 체제에 따라 다음 조치 중 하나를 실행하십시오.
 - AIX, HP-UX, Linux 또는 Solaris 시스템의 경우: **lbadm**을 입력하십시오.
 - Windows 시스템의 경우: 시작 > 프로그램 > **IBM WebSphere > Edge Components > IBM Load Balancer > Load Balancer**를 클릭하십시오.
3. Caching Proxy를 시작하십시오. (GUI에서 Caching Proxy를 시작하기 전에 먼저 Host에 연결하여 CBR 컴포넌트에 대해 실행 프로그램을 실행해야 함). 다음 중 하나를 수행하십시오.
 - AIX, HP-UX, Linux 또는 Solaris 시스템의 경우: Caching Proxy를 시작하려면 **ibmproxy**를 입력하십시오.
 - Windows 시스템의 경우: Caching Proxy를 시작하려면 서비스 패널(시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스)로 이동하십시오.

GUI에서 CBR 컴포넌트를 구성하려면 우선 트리 구조에서 **Content Based Routing**을 선택해야 합니다. 호스트에 연결한 후에 관리자를 시작할 수 있습니다. 또한 포트와 서버가 들어 있는 클러스터를 작성하고 관리자에 대한 어드바이저를 시작할 수 있습니다.

GUI를 사용하면 **cbrcontrol** 명령으로 수행할 수 있는 모든 작업을 수행할 수 있습니다. 예를 들어, 명령행을 사용하여 클러스터를 정의하려면, **cbrcontrol cluster add cluster** 명령을 입력하십시오. GUI에서 클러스터를 정의하려면, 실행 프로그램을 마우스 오른쪽 단추로 클릭하여 표시되는 팝업 메뉴에서 클러스터 추가를 마우스 왼쪽 단추로 클릭하십시오. 팝업 창에 클러스터 주소를 입력한 후 확인을 클릭하십시오.

호스트 팝업 메뉴에 있는 새 구성 로드 옵션(현재 구성을 완전히 바꾸기 위한 옵션) 및 현재 구성에 추가 옵션(현재 구성을 갱신하기 위한 옵션)을 사용하여 기존 CBR 구성 파일을 로드할 수 있습니다. 호스트 팝업 메뉴에 있는 다른 옵션인 구성 파일 저장 옵션을 사용하여 CBR 구성을 파일에 정기적으로 저장해야 합니다. GUI의 맨 위에 있는 파일 메뉴를 사용하여 현재 호스트 연결을 파일로 저장하거나 모든 Load Balancer 컴포넌트에 걸쳐 기존 파일의 연결을 복원할 수 있습니다.

Load Balancer 창 오른쪽 상단 구석의 물음표를 클릭하면, 도움말에 액세스할 수 있습니다.

- 도움말: 필드 레벨 — 각 필드 및 기본값을 설명합니다.
- 도움말: 수행 방법 — 해당 화면에서 수행할 수 있는 작업이 나열되어 있습니다.
- InfoCenter — 제품 정보에 대한 중앙집중화된 액세스를 제공합니다.

GUI에서 명령을 실행하려면 GUI 트리에서 호스트 노드를 강조표시하고 호스트 팝업 메뉴에서 명령 전송...을 선택하십시오. 명령 입력 필드에서는 **executor report**와 같이 실행하려는 명령을 입력하십시오. 현재 세션에서 실행되는 명령의 결과 및 히스토리가 제공된 창에 나타납니다.

GUI 사용에 대한 자세한 내용은 499 페이지의 부록 A 『GUI: 일반 명령』을 참조하십시오.

구성 마법사

구성 마법사를 사용 중이면, 다음 단계를 수행하십시오.

1. cbrserver를 시작하십시오. 루트 사용자나 관리자로 명령 프롬프트에서 **cbrserver**를 발행하십시오.
2. CBR의 마법사 기능을 시작하십시오.

cbrwizard를 실행하여 명령 프롬프트에서 마법사를 실행하십시오. 또는 GUI에 표시된 것처럼 CBR 컴포넌트 메뉴에서 구성 마법사를 선택하십시오.

3. Caching Proxy를 시작하여 HTTP 또는 HTTPS(SSL) 통신의 로드 밸런스를 수행하십시오.

AIX, HP-UX, Linux 또는 Solaris 시스템의 경우: Caching Proxy를 시작하려면 **ibmproxy**를 입력하십시오.

Windows 시스템의 경우: Caching Proxy를 시작하려면 서비스 패널(시작 > 설정 (Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스)로 이동하십시오.

CBR 마법사는 CBR 컴포넌트의 기본 구성 작성 프로세스를 단계별로 안내합니다. 또한 네트워크에 대한 질문을 요청하고 CBR에서 서버 그룹 간의 통신량 로드 밸런스를 유지할 수 있도록 클러스터를 설정하는 단계를 안내합니다.

CBR 시스템 설정

CBR 시스템을 설정하려면 루트 사용자(AIX, HP-UX, Linux 또는 Solaris 시스템의 경우) 또는 관리자(Windows 시스템의 경우)여야 합니다.

설정할 서버의 각 클러스터당 하나의 IP 주소가 필요합니다. 클러스터 주소는 호스트 이름(예: `www.yourcompany.com`)과 연관된 주소입니다. 이 IP 주소는 클러스터에서 서버에 연결하기 위해 클라이언트에서 사용합니다. 특히, 이 주소는 클라이언트로부터의 URL 요청에서 찾을 수 있습니다. 동일한 클러스터 주소로의 모든 요청은 CBR에 의해 로드 밸런스가 수행됩니다.

Solaris 시스템 전용의 경우: CBR 컴포넌트를 사용하기 전에 프로세스 간 통신(IPC)의 시스템 기본값을 수정해야 합니다. 공유 메모리 세그먼트의 최대 크기 및 세마포어 ID 수를 늘려야 합니다. 시스템이 CBR을 지원하도록 조정하려면 시스템의 `/etc/system` 파일을 편집하여 다음 명령문을 추가하고 재부트하십시오.

```
set shmsys:shminfo_shmmax=0x02000000
set semsys:seminfo_semmap=750
set semsys:seminfo_semmni=30
set semsys:seminfo_semmns=750
set semsys:seminfo_semmnu=30
set semsys:seminfo_sesume=30
```

공유 메모리 세그먼트를 위에 표시된 값으로 늘리지 않을 경우, `cbrcontrol executor start` 명령은 실패합니다.

1단계. CBR을 사용하기 위해 Caching Proxy 구성

CBR을 사용하려면, Caching Proxy가 설치되어 있어야 합니다.

주: Caching Proxy는 기본적으로 설치 후 자동으로 시작하는 서비스입니다. CBR 서버 기능을 시작하기 전에 Caching Proxy를 정지시킨 후 Caching Proxy 서비스가 자동으로 아닌 수동으로 시작되도록 수정하십시오.

- AIX, HP-UX, Linux 및 Solaris 시스템의 경우: `ps -ef | grep ibmproxy` 명령을 사용하여 프로세스 ID를 찾은 후 `kill process_id` 명령으로 프로세스를 종료하여 Caching Proxy를 정지하십시오.
- Windows 시스템의 경우: 서비스 패널에서 Caching Proxy를 정지하십시오.

Caching Proxy 구성 파일(`ibmproxy.conf`)에서 다음과 같이 수정해야 합니다.

수신 URL 지시문 **CacheByIncomingUrl**이 "off"(기본값)로 설정되어 있는지 확인하십시오.

구성 파일의 맵핑 규칙 섹션에서 모든 클러스터에 대해 다음과 같은 맵핑 규칙을 추가하십시오.

```
Proxy    /* http://cluster.domain.com/*    cluster.domain.com
```

주: CBR은 나중에 프로토콜, 서버 및 대상 포트를 설정합니다.

CBR 플러그인에 대해 편집할 4가지 항목은 다음과 같습니다.

- ServerInit
- PostAuth
- PostExit
- ServerTerm

각 항목은 단일 행에 있어야 합니다. ibmproxy.conf 파일에는 "ServerInit" 인스턴스가 여러 개 있는데, 각 플러그인에 하나씩 대응됩니다. "CBR Plug-in"에 대응하는 항목은 편집하되 주석은 붙이지 마십시오.

각 운영 체제의 구성 파일에 대한 특정 추가 사항은 다음과 같습니다.

그림 20. AIX, Linux 및 Solaris 시스템용 CBR 구성 파일

```
ServerInit /opt/ibm/edge/lb/servers/lib/liblbcbr.so:ndServerInit
PostAuth /opt/ibm/edge/lb/servers/lib/liblbcbr.so:ndPostAuth
PostExit /opt/ibm/edge/lb/servers/lib/liblbcbr.so:ndPostExit
ServerTerm /opt/ibm/edge/lb/servers/lib/liblbcbr.so:ndServerTerm
```

그림 21. HP-UX 시스템용 CBR 구성 파일

```
ServerInit /opt/ibm/edge/lb/servers/lib/liblbcbr.sl:ndServerInit
PostAuth /opt/ibm/edge/lb/servers/lib/liblbcbr.sl:ndPostAuth
PostExit /opt/ibm/edge/lb/servers/lib/liblbcbr.sl:ndPostExit
ServerTerm /opt/ibm/edge/lb/servers/lib/liblbcbr.sl:ndServerTerm
```

그림 22. Windows 시스템용 CBR 구성 파일

```
ServerInit C:\Program Files\IBM\edge\lb\servers\lib\liblbcbr.dll:ndServerInit
PostAuth C:\Program Files\IBM\edge\lb\servers\lib\liblbcbr.dll:ndPostAuth
PostExit C:\Program Files\IBM\edge\lb\servers\lib\liblbcbr.dll:ndPostExit
ServerTerm C:\Program Files\IBM\edge\lb\servers\lib\liblbcbr.dll:ndServerTerm
```

2단계. 서버 기능 시작

CBR 서버 기능을 시작하려면 명령행에서 **cbrserver**를 입력하십시오.

기본 구성 파일(default.cfg)은 cbrserver를 시작할 때 자동으로 로드됩니다. CBR 구성을 default.cfg에 저장하면 이 파일에 저장된 모든 내용이 다음 cbrserver를 시작할 때 자동으로 로드됩니다.

3단계. 실행 프로그램 기능 시작

실행 프로그램 기능을 시작하려면, **cbrcontrol executor start** 명령을 입력하십시오. 이때 여러가지 실행 프로그램 설정을 변경할 수도 있습니다. 383 페이지의 『dscontrol executor — 실행 프로그램 제어』를 참조하십시오.

4단계. 클러스터 정의 및 클러스터 옵션 설정

CBR은 해당 클러스터의 포트에 구성된 해당 서버에 대한 클러스터로 전송된 요청의 밸런스를 조정합니다.

클러스터는 URL의 호스트 부분에 위치한 기호 이름으로 ibmproxy.conf 파일의 Proxy 명령문에 사용된 이름과 일치해야 합니다.

CBR에 정의된 클러스터는 수신 요청과 일치하도록 정의되어야 합니다. 클러스터는 수신 요청에 있는 것과 동일한 IP 주소 및 호스트 이름을 사용하여 정의되어야 합니다. 예를 들어 요청이 IP 주소로 입력되면 클러스터는 IP 주소로 정의되어야 합니다. 단일 IP 주소로 해석되는 둘 이상의 호스트 이름이 있는 경우(및 요청이 해당 호스트 이름 중 하나와 함께 도착하는 경우), 모든 호스트 이름은 클러스터로 정의되어야 합니다.

클러스터를 정의하려면 다음 명령을 실행하십시오.

```
cbrcontrol cluster add cluster
```

클러스터 옵션을 설정하려면 다음 명령을 실행하십시오.

```
cbrcontrol cluster set cluster option value
```

자세한 정보는 369 페이지의 제 27 장 『Dispatcher 및 CBR 명령어 참조서』를 참조하십시오.

5단계. 네트워크 인터페이스 카드 별명 지정(선택적)

역방향 프록시로 구성된 Caching Proxy를 실행하는 경우 여러 웹 사이트에 대해 로드 밸런스를 수행하려면 각 웹 사이트의 클러스터 주소를 Load Balancer 시스템의 최소한 하나의 네트워크 인터페이스 카드에 추가해야 합니다. 그렇지 않은 경우에는 이 단계를 생략할 수 있습니다.

AIX, HP-UX, Linux 또는 **Solaris** 시스템의 경우: 클러스터 주소를 네트워크 인터페이스에 추가하려면 ifconfig 명령을 사용하십시오. 131 페이지의 표 8에 나타난 운영 체제에 해당하는 명령을 사용하십시오.

표 8. NIC 별명 명령

AIX	ifconfig <i>interface_name</i> alias <i>cluster_address</i> netmask <i>netmask</i>
HP-UX	ifconfig <i>interface_name</i> <i>cluster_address</i> netmask <i>netmask</i> up
Linux	ifconfig <i>interface_name</i> <i>cluster_address</i> netmask <i>netmask</i> up
Solaris 8, Solaris 9 및 Solaris 10	ifconfig <i>interface_name</i> addif <i>cluster_address</i> netmask <i>netmask</i> up

주: Linux 및 HP-UX 시스템의 경우, *interface_name*에는 추가된 각각의 클러스터 주소에 해당하는 고유 번호가 있습니다(예: eth0:1, eth0:2 등).

Windows 2000의 경우: 클러스터 주소를 네트워크 인터페이스에 추가하려면 다음을 수행하십시오.

1. 시작 > 설정 > 제어판을 클릭하십시오.
2. 네트워크 및 전화 접속 연결을 두 번 클릭하십시오.
3. LAN 접속을 마우스 오른쪽 단추로 클릭하십시오.
4. 등록 정보를 선택하십시오.
5. 인터넷 프로토콜(TCP/IP)을 선택하고 등록 정보를 클릭하십시오.
6. 다음 IP 주소 사용을 선택하고 고급을 클릭하십시오.
7. 추가를 클릭한 후 클러스터의 IP 주소 및 서브넷 마스크를 입력하십시오.

Windows 2003의 경우: 클러스터 주소를 네트워크 인터페이스에 추가하려면 다음을 수행하십시오.

1. 시작 > 제어판 > 네트워크 연결 > 로컬 영역 연결 클릭
2. 등록 정보를 클릭하십시오.
3. 인터넷 프로토콜(TCP/IP)을 선택하고 등록 정보를 클릭하십시오.
4. 다음 IP 주소 사용을 선택하고 고급을 클릭하십시오.
5. 추가를 클릭한 다음 클러스터의 IP 주소 및 서브넷 마스크를 입력하십시오.

6단계. 포트 정의 및 포트 옵션 설정

포트 번호는 서버 응용프로그램이 인식하는 포트입니다. HTTP 통신을 실행 중인 Caching Proxy가 있는 CBR의 경우, 일반적으로 포트 80입니다.

이전 단계에서 정의한 클러스터에 포트를 정의하려면 다음 명령을 실행하십시오.

```
cbrcontrol port add cluster:port
```

포트 옵션을 설정하려면 다음 명령을 실행하십시오.

```
cbrcontrol port set cluster:port option value
```

자세한 정보는 369 페이지의 제 27 장 『Dispatcher 및 CBR 명령어 참조서』를 참조하십시오.

7단계. 로드 밸런스 서버 시스템 정의

서버 시스템은 로드 밸런스를 수행하는 응용프로그램이 실행되는 시스템입니다. *server*는 서버 시스템의 기호 이름이거나 점분리 10진수 주소입니다. 포트 및 클러스터의 서버를 정의하려면 다음 명령을 발행하십시오.

```
cbrcontrol server add cluster:port:server
```

로드 밸런스를 수행하기 위해서는 클러스터의 포트별로 두 개 이상의 서버를 정의해야 합니다.

8단계. 구성에 규칙 추가

이 단계는 Caching Proxy가 있는 CBR을 구성하는 핵심 단계입니다. 규칙은 URL 요청을 구별하고 적절한 서버 세트에 요청을 전송하는 방법을 정의합니다. CBR이 사용하는 특수한 규칙 유형을 콘텐츠 규칙이라고 합니다. 콘텐츠 규칙을 정의하려면 다음 명령을 실행하십시오.

```
cbrcontrol rule add cluster:port:rule type content pattern pattern
```

pattern 값은 각 클라이언트 요청에서 URL과 비교할 일반 표현식입니다. 패턴 구성법에 대한 자세한 내용은 507 페이지의 부록 B 『콘텐츠 규칙(패턴) 구문』을 참조하십시오.

Dispatcher에 정의된 일부 기타 규칙 유형은 CBR에서도 사용될 수 있습니다. 자세한 내용은 226 페이지의 『규칙 기반 로드 밸런스 구성』을 참조하십시오.

9단계. 규칙에 서버 추가

규칙이 클라이언트 요청에 의해 일치되면, 최적의 서버를 판별하기 위해 규칙의 서버 세트가 조회됩니다. 규칙의 서버 세트는 포트에 정의된 서버의 서브세트입니다. 규칙의 서버 세트에 서버를 추가하려면 다음 명령을 발행하십시오.

```
cbrcontrol rule useserver cluster:port:rule server
```

10단계. 관리자 기능 시작(선택적)

관리자 기능은 로드 밸런스를 향상시킵니다. 관리자를 시작하려면 다음 명령을 발행하십시오.

```
cbrcontrol manager start
```

11단계. 어드바이저 기능 시작(선택적)

어드바이저는 요청에 응답하는 로드 밸런스 서버 시스템의 능력에 대한 자세한 정보를 관리자에 제공합니다. 어드바이저는 프로토콜마다 고유합니다. 예를 들어 HTTP 어드바이저를 시작하려면 다음 명령을 실행하십시오.

```
cbrcontrol advisor start http port
```

12단계. 필요한 클러스터 비율 설정

어드바이저를 시작할 경우, 로드 밸런스 결정 시 포함되는 어드바이저 정보에 부여된 중요성의 비율을 수정할 수 있습니다. 클러스터 비율을 설정하려면 **cbrcontrol cluster set cluster proportions** 명령을 발행하십시오. 자세한 내용은 194 페이지의 『상태 정보에 제공되는 중요성 비율』을 참조하십시오.

13단계. Caching Proxy 시작

- AIX 시스템: LIBPATH 환경 변수에 추가:

`/opt/ibm/edge/lb/servers/lib`

- Linux, HP-UX 또는 Solaris 시스템: LD_LIBRARY_PATH 환경 변수에 추가:

`/opt/ibm/edge/lb/servers/lib`

- Windows 시스템: PATH 환경 변수에 추가:

`C:\Program Files\IBM\edge\lb\servers\lib`

새로운 환경에서, 명령 프롬프트에서 Caching Proxy를 시작하고 **ibmproxy**를 발행하십시오.

주: Windows 시스템의 경우: 서비스 패널(시작 -> 설정(Windows 2000의 경우) -> 제어판 -> 관리 도구 -> 서비스)에서 Caching Proxy를 시작하십시오.

CBR 구성 예제

CBR을 구성하려면 다음 단계를 따르십시오.

1. CBR을 시작하려면, **cbrserver** 명령을 발행하십시오.
2. 명령 인터페이스를 시작하려면, **cbrcontrol** 명령을 발행하십시오.
3. **cbrcontrol** 프롬프트가 표시됩니다. 다음 명령을 발행하십시오. (*cluster(c),port(p),rule(r),server(s)*)
 - `executor start`
 - `cluster add c`
 - `port add c:p`
 - `server add c:p:s`
 - `rule add c:p:r type content pattern uri=*`
 - `rule use server c:p:r s`
4. Caching Proxy를 시작하십시오. **ibmproxy** 명령을 발행하십시오. (Windows 플랫폼의 경우, 서비스 패널에서 Caching Proxy를 시작하십시오.)
5. 브라우저에서 프록시 구성을 모두 제거하십시오.
6. "'c'가 이전에 구성한 클러스터인 브라우저로 `http://c/`를 로드하십시오.

- 서버 's'가 호출됩니다.
- http://s/라는 웹 페이지가 표시됩니다.

제 4 부 Site Selector 컴포넌트

이 파트에서는 빠른 시작 구성, 계획 고려사항에 대한 정보를 제공하며 Load Balancer의 Site Selector 컴포넌트 구성 메소드에 대해 설명합니다. 다음 장을 포함합니다.

- 137 페이지의 제 12 장 『빠른 시작 구성』
- 141 페이지의 제 13 장 『Site Selector 계획』
- 145 페이지의 제 14 장 『Site Selector 구성』

제 12 장 빠른 시작 구성

이 빠른 시작 예는 클라이언트 요청에 사용되는 도메인 이름에 기초한 서버 세트 사이에서 통신량을 로드 밸런스하기 위해 Site Selector를 사용하여 사이트 이름 구성을 작성하는 방법을 보여 줍니다.

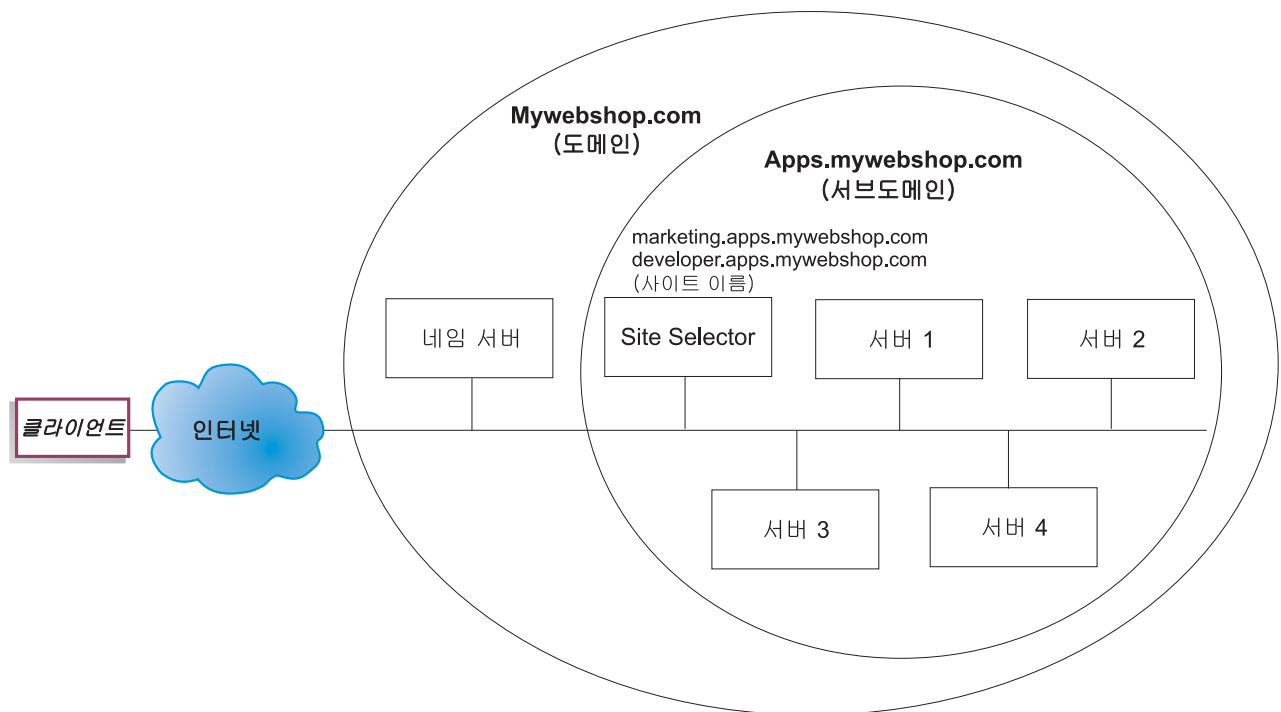


그림 23. 간단한 Site Selector 구성

필요한 내용

이 빠른 시작 구성 예의 경우, 다음이 필요합니다.

- 사이트의 이름 서버에 대한 관리 액세스 권한
- 네트워크에 구성된 네 서버(server1, server2, server3, server4) 및 Site Selector 컴포넌트가 설치된 추가 서버

주: 로드 밸런스된 서버 중 하나에 Site Selector를 결합 배치할 경우, 다섯 개가 아닌 네 개의 서버가 필요합니다. 그러나 결합 배치는 로드 밸런스된 서버의 성능에 영향을 미칠 수 있습니다.

준비 방법

이 빠른 시작 예의 경우, 회사의 사이트 도메인은 mywebshop.com입니다. Site Selectors는 mywebshop.com 서브도메인에 대한 권한이 있습니다. mywebshop.com에 서브도메인을 정의해야 합니다. (예: apps.mywebshop.com). Site Selector는 BIND와 같이 완전히 구현된 DNS가 아니며 DNS 계층 구조의 리프노드처럼 작동합니다. Site Selector는 apps.mywebshop.com 서브도메인에 대한 권한을 가집니다. 서브도메인 apps.mywebshop.com은 사이트 이름 marketing.apps.mywebshop.com 및 developer.apps.mywebshop.com을 포함합니다.

1. 회사의 도메인 이름 서버를 갱신하십시오(137 페이지의 그림 23 참조). Site Selector가 권한이 있는 이름 서버인 서브도메인(apps.mywebshop.com)에 대한 이름 서버 레코드를 named.data 파일에 작성하십시오.

apps.mywebshop.com. IN NS siteselector.mywebshop.com

2. 완전한 호스트 이름 또는 사이트가 현재 도메인 이름 시스템에서 분석되지 않는지 확인하십시오.
3. Site Selector 로드 밸런스를 포함하려는 서버(server1, server2, server3, server4)에 Metric Server를 설치하십시오. 211 페이지의 『Metric Server』에서 자세한 정보를 참조하십시오.

Site Selector 컴포넌트 구성

Site Selector를 통해 명령행, 구성 마법사 또는 GUI(Graphical User Interface)를 사용하여 구성을 작성할 수 있습니다. 이 빠른 시작 예의 경우, 구성 단계는 명령행을 사용하여 예시됩니다.

주: 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름과 파일 이름의 매개변수 값만 예외입니다.

명령행을 사용한 구성

명령 프롬프트에서 다음 단계를 따르십시오

1. Site Selector를 호스트하는 시스템에서 ssserver를 시작하십시오. 루트 사용자 또는 관리자로 명령 프롬프트에서 **ssserver** 명령을 실행하십시오.

주: Windows 플랫폼의 경우: 서비스 패널에서 ssserver(IBM Site Selector)를 시작하십시오(시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스).

2. Site Selector 구성에서 이름 서버를 시작하십시오.

sscontrol nameserver start

3. Site Selector에서 사이트 이름(marketing.apps.mywebshop.com and developer.apps.mywebshop.com)을 구성하십시오.

sscontrol sitename add marketing.apps.mywebshop.com

sscontrol sitename add developer.apps.mywebshop.com

4. 서버를 Site Selector 구성에 추가하십시오. (server1 및 server2를 사이트 이름 marketing.apps.mywebshop.com에 구성하십시오. server3 및 server4를 사이트 이름 developer.apps.mywebshop.com에 구성하십시오.)

sscontrol server add marketing.apps.mywebshop.com:server1+server2

sscontrol server add developer.apps.mywebshop.com:server3+server4

5. Site Selector의 관리자 기능을 시작하십시오.

sscontrol manager start

6. Site Selector의 어드바이저 기능을 시작하십시오(marketing.apps.mywebshop.com의 HTTP 어드바이저 및 developer.apps.mywebshop의 FTP 어드바이저):

sscontrol advisor start http marketing.apps.mywebshop.com:80

sscontrol advisor start ftp developer.apps.mywebshop.com:21

Site Selector에서는 실패한 서버로 클라이언트 요청이 전송되지 않았음을 확인합니다.

7. Metric Server가 각 로드 밸런스 서버에서 시작되었는지 확인하십시오.

사용자의 기본 Site Selector 구성을 지금 완료했습니다.

구성 검사

구성이 작동되고 있는지 확인하십시오.

1. mywebshop.com에 대한 권한이 있는 네임 서버로 기본 DNS 구성이 된 클라이언트에서 구성된 사이트 이름 중 하나에 ping하십시오.
2. 응용프로그램에 연결하십시오. 예를 들어,
 - 브라우저를 열고 marketing.apps.mywebshop.com을 요청하면 유효한 페이지가 제공됩니다.
 - FTP 클라이언트를 developer.apps.mywebshop.com에서 열어 유효한 사용자와 암호를 입력하십시오.
3. 다음 명령의 결과를 보십시오.

sscontrol server status marketing.apps.mywebshop.com:

sscontrol server status developer.apps.mywebshop.com:

각 서버의 히트 항목 총계가 ping 및 응용프로그램 요청까지 추가되어야 합니다.

GUI(Graphical User Interface)를 사용한 구성

Site Selector GUI 사용에 대한 정보는 147 페이지의 『GUI』 및 499 페이지의 부록 A 『GUI: 일반 명령』을 참조하십시오.

구성 마법사를 사용한 구성

Site Selector 마법사 사용에 대한 정보는 148 페이지의 『구성 마법사』를 참조하십시오.

제 13 장 Site Selector 계획

이 장에서는 Site Selector 컴포넌트를 설치하고 구성하기 전에 네트워크 계획자가 고려해야 할 사항에 대해 설명합니다.

- 네트워크 관리에 필요한 기능의 개요에 대해서는 21 페이지의 제 3 장 『네트워크 관리: 사용할 Load Balancer 기능 결정』 페이지를 참조하십시오.
- Site Selector의 로드 밸런스 매개변수 구성에 대한 정보는 145 페이지의 제 14 장 『Site Selector 구성』을 참조하십시오.
- 더 많은 고급 기능을 위한 Load Balancer 설정 방법에 대한 정보는 215 페이지의 제 22 장 『Dispatcher, CBR 및 Site Selector에 대한 고급 기능』을 참조하십시오.
- 원격 인증 관리, Load Balancer 로그 및 Load Balancer 컴포넌트 사용법에 대한 정보는 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』를 참조하십시오.

이 장에는 다음과 같은 섹션이 수록되어 있습니다.

- 『계획 고려사항』
- 143 페이지의 『TTL 고려사항』
- 144 페이지의 『네트워크 근접 기능 사용』

계획 고려사항

Site Selector는 도메인 이름 서버와 함께 작동하여 수집한 단위와 가중치를 통해 서버 그룹의 로드 밸런스를 조정합니다. 클라이언트의 요청에 사용할 도메인 이름에 따라 서버 그룹 간 통신을 로드 밸런스할 수 있도록 사이트 구성을 작성할 수 있습니다.

제한사항: Site Selector는 A 유형 쿼리만 DNS 쿼리로 지원합니다. 다른 유형의 쿼리는 NOTIMPL(Not Implemented: 구현되지 않음) 리턴 코드로 귀착됩니다. 전체 도메인을 Site Selector에 위임하는 경우, A 유형 쿼리만을 수신하도록 하십시오.

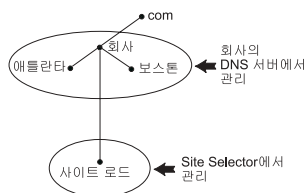


그림 24. DNS 환경의 예제

DNS 환경 내에서 Site Selector에 대한 서브도메인을 설치할 때, Site Selector는 자신의 서브도메인에 대한 권한을 가져야 합니다. 예를 들어(그림 24 참조), **company.com** 도메인에 대한 권한이 사용자 회사에 지정되었습니다. 회사 내에는 여러 개의 서브도메

인이 있습니다. Site Selector는 **siteload.company.com**에 대한 권한을 가지며 DNS 서버도 계속 **atlanta.company.com** 및 **boston.company.com**에 대한 권한을 유지합니다.

회사 이름 서버가 Site Selector를 사이트 로드 서브도메인의 권한을 가진 것으로 인식하기 위해서는, 이름이 지정된 해당 데이터 파일에 이름 서버 항목을 추가해야 합니다. 예를 들어, AIX 시스템에서 이름 서버 항목은 다음과 같습니다.

```
siteload.company.com. IN NS siteselector.company.com.
```

여기에서 **siteselector.company.com**은 Site Selector 시스템의 호스트 이름입니다. DNS 서버에서 사용하려면 이름이 지정된 다른 데이터베이스 파일에 이에 상응하는 항목이 작성되어야 합니다.

클라이언트는 도메인 이름에 대한 분석 요청을 네트워크 내의 이름 서버로 제출합니다. 이름 서버는 Site Selector 시스템으로 요청을 전달합니다. 그러면, Site Selector가 사이트 이름에 따라 구성된 서버 중 한 서버의 IP 주소로 도메인 이름을 분석합니다. Site Selector는 선택한 서버의 IP 주소를 이름 서버로 리턴합니다. 이름 서버는 IP 주소를 클라이언트로 리턴합니다(Site Selector는 순환하지 않는(리프 노트) 이름 서버로 사용되며, 도메인 이름 요청을 분석하지 않으면 오류를 리턴합니다).

로컬 서버와 원격 서버를 로드 밸런싱하기 위해 Site Selector가 DNS 시스템과 함께 사용되는 사이트를 나타내는 16 페이지의 그림 5를 참조하십시오.

Site Selector는 다음과 같은 기능으로 구성됩니다.

- **ssserver**는 이름 서버, 관리자 및 어드바이저에 대한 명령행의 요청을 처리합니다.
- 이름 서버 기능은 수신 이름 서버 요청에 대한 로드 밸런스를 지원합니다. DNS 분석을 시작하려면 Site Selector의 이름 서버 기능을 시작해야 합니다. Site Selector는 포트 53에서 수신 DNS 요청을 인식합니다. 요청하는 사이트 이름이 구성되면 Site Selector는 사이트 이름과 연관된 단일 서버 주소(일련의 서버 주소에서)를 리턴합니다.
- 관리자는 다음에 따라 이름 서버에 사용할 가중치를 설정합니다.
 - 어드바이저가 제공하는 서버에서의 피드백
 - Metric Server와 같은 시스템 모니터링 프로그램 피드백

관리자를 사용하는 것은 선택입니다. 그러나 관리자를 사용하지 않으면, 현재 서버 가중치에 따라 가중된 라운드 로빙 스케줄링을 사용하여 로드 밸런스가 수행되며, 어드바이저는 사용 가능하지 않습니다.

- **Metric Server**는 백엔드 서버 시스템에 설치한 Load Balancer의 시스템 모니터링 컴포넌트입니다. (로드 밸런싱 서버 시스템에 []를 배치할 경우, Load Balancer 시스템에 Metric Server를 설치합니다.)

Metric Server를 사용하면, Site Selector는 서버의 활동 레벨을 모니터하고, 서버가 가장 심하지 않게 로드되는 시기 및 실패한 서버를 발견할 수 있습니다. 로드는 서버가 얼마나 어렵게 작동하는지에 대한 측정입니다. Site Selector 시스템 관리자는 로드를 측정하는 데 사용하는 측정 유형을 제어합니다. Site Selector는 액세스 빈도, 총 사용자 수, 액세스 유형(예: 간단한 조회, 장기 조회 또는 CPU 집중 로드)과 같은 요소를 고려하여 환경에 맞게 구성될 수 있습니다.

로드 밸런스는 서버 가중치를 기반으로 합니다. Site Selector의 경우, 관리자에서 가중치를 판별하는 데 사용하는 4가지 비율이 있습니다.

- CPU
- 메모리
- port
- 시스템

CPU 및 메모리 값은 모두 Metric Server에 의해 제공됩니다. 따라서 Site Selector 컴포넌트에 Metric Server를 사용하는 것이 권장됩니다.

211 페이지의 『Metric Server』에서 자세한 정보를 참조하십시오.

- 어드바이저는 관리자를 호출하여 가중치를 적절하게 설정하기 전에 서버를 조회하고 프로토콜별로 결과를 분석합니다. 일반 구성에서 이들 어드바이저를 사용하는 것은 바람직하지 않습니다. 또한 사용자 고유의 어드바이저를 작성하는 옵션도 있습니다. 어드바이저를 사용하는 것은 선택입니다. 199 페이지의 『어드바이저』에서 자세한 정보를 참조하십시오.
- 이름 서버, 어드바이저, Metric Server, 관리자를 구성하고 관리하려면, 명령행(sscontrol) 또는 그래픽 사용자 인터페이스(lbadmin)를 사용하십시오.

Site Selector의 핵심적인 네 기능(이름 서버, 관리자, Metric Server, 어드바이저)은 서버 간의 수신 요청을 밸런스하고 분석하기 위해 상호작용합니다.

TTL 고려사항

DNS 기반 로드 밸런스를 사용하면 이름 분석 캐시를 사용할 수 없습니다. TTL(Time To Live) 값이 DNS 기반 로드 밸런스의 효율성을 결정합니다. TTL은 이름 서버가 해석된 응답을 캐시하는 시간을 결정합니다. 작은 TTL 값은 서버 또는 네트워크 로드에서 일어나는 미묘한 변화를 더 빨리 인식할 수 있습니다. 그러나 캐시를 사용하지 않으면 이름 해석 요청이 있을 때마다 클라이언트가 권한 이름 서버에 접속해야 하기 때문에 클라이언트의 대기 시간을 증가시킬 수 있습니다. TTL 값을 선택할 경우, 캐시를 사용하지 않을 때 그 설정이 환경에 어떤 영향을 미칠 것인지를 주의깊게 고려해야 합니다. 또한 이름 분석의 클라이언트측 캐시가 DNS 기반 로드 밸런스를 제한할 수 있다는 것을 알아야 합니다.

sscontrol sitename [add | set] 명령을 사용하여 TTL을 구성할 수 있습니다. 452 페이지의 『sscontrol sitename — 사이트 이름 구성』에서 자세한 정보를 참조하십시오.

네트워크 근접 기능 사용

네트워크 근접은 요청하는 클라이언트와 각 서버와의 근접에 대한 계산입니다. 네트워크 근접을 판별하려면, Metric Server 에이전트(로드 밸런스된 각 서버에 있음)는 클라이언트 IP 주소로 ping을 전송하고 응답 시간을 Site Selector에 리턴합니다. Site Selector는 로드 밸런스 결정 시 근접 응답을 사용합니다. Site Selector는 네트워크 근접 응답 값을 매니저의 가중치와 결합시켜서 서버에 대해 결합된 최종 가중치 값을 작성합니다.

Site Selector에 네트워크 근접 기능을 사용하는 것은 선택사항입니다.

Site Selector는 각 사이트 이름에 설정할 수 있는 네트워크 근접 옵션을 제공합니다.

- 캐시 수명: 근접 응답이 유효하고 캐시에 저장되는 시간
- 근접 퍼센트: 근접 응답 대 서버의 상태(관리자 가중치의 입력)의 중요성.
- 모두 대기: 클라이언트 요청에 응답하기 전에 서버로부터 모든 근접(ping 명령) 응답의 대기 여부를 판별합니다.

yes로 설정하면, Metric Server는 근접 응답 시간을 받기 위해 클라이언트에 ping 명령을 실행합니다. 이름 서버는 모든 Metric Server가 응답할 때까지 또는 제한시간이 초과될 때까지 대기합니다. 그리고 나서, 서버마다 이름 서버는 관리자가 계산한 가중치를 근접 응답 시간과 결합시켜 각 서버에 대해 "결합된 가중치" 값을 작성합니다. Site Selector는 서버 IP 주소가 있는 클라이언트에게 최상의 결합 가중치를 제공합니다. (대부분의 클라이언트 이름 서버의 종료 시간은 5초입니다. Site Selector는 제한시간이 초과되기 전에 응답을 시도합니다.)

no로 설정하면, 이름 분석이 현재 관리자 가중치를 기반으로 하는 클라이언트에게 제공됩니다. 그리고 나서, Metric Server가 근접 응답 시간을 받기 위해 클라이언트에 ping 명령을 실행합니다. 이름 서버는 Metric Server에서 받은 응답 시간을 캐시합니다. 두 번째 요청에서 클라이언트가 리턴되면, 이름 서버는 현재 관리자 가중치를 각 서버에 대해 캐시된 ping 명령 응답 값과 결합시켜 최상의 "결합된 가중치"가 있는 서버를 가져옵니다. Site Selector는 이 서버의 IP 주소를 두 번째 요청의 클라이언트로 리턴합니다.

네트워크 근접성 옵션은 **sscontrol sitename [add | set]** 명령에서 설정될 수 있습니다. 429 페이지의 제 28 장 『Site Selector 명령어 참조서』에서 자세한 정보를 참조하십시오.

제 14 장 Site Selector 구성

이 장의 단계를 수행하기 전에, 141 페이지의 제 13 장 『Site Selector 계획』을 참조하십시오. 이 장에서는 Load Balancer의 Site Selector 컴포넌트에 대한 기본 구성을 작성하는 방법을 설명합니다.

- Load Balancer의 복합 구성에 대해서는 193 페이지의 제 21 장 『Dispatcher, CBR 및 Site Selector에 대한 관리자, 어드바이저 및 Metric Server 기능』 및 215 페이지의 제 22 장 『Dispatcher, CBR 및 Site Selector에 대한 고급 기능』을 참조하십시오.
- 원격 인증 관리, Load Balancer 로그 및 Load Balancer 컴포넌트 사용법에 대한 정보는 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』를 참조하십시오.

구성 task 개요

주: 이 테이블의 구성 단계를 시작하기 전에, Site Selector 시스템과 모든 서버 시스템은 네트워크에 연결되어 있고 유효한 IP 주소를 가지며 서로 ping할 수 있어야 합니다.

표 9. Site Selector 컴포넌트 구성 task

task	설명	관련 정보
Site Selector 시스템 설정.	요구사항에 대해 찾기.	148 페이지의 『Site Selector 시스템 설정』
로드 밸런스를 수행할 시스템 설정.	로드 밸런스 구성을 설정합니다.	149 페이지의 『4단계. 로드 밸런스 서버 시스템 정의』

구성 방법

Load Balancer의 Site Selector 컴포넌트의 기본 구성을 작성할 경우 네 가지 기본 방법으로 Site Selector 컴포넌트를 구성할 수 있습니다.

- 명령행
- 스크립트
- GUI(Graphical User Interface)
- 구성 마법사

명령행

이것은 Site Selector를 구성하는 가장 직접적인 방법입니다. 명령 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름(사이트 이름 및 서버 명령에 사용됨)과 파일 이름만 예외입니다.

명령행에서 Site Selector를 시작하려면 다음을 수행하십시오.

1. 명령 프롬프트에서 **ssserver** 명령을 실행하십시오. 서비스를 정지하려면 **ssserver stop**을 입력하십시오.

주: Windows 시스템의 경우, 시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스를 클릭하십시오. **IBM Site Selector**를 마우스 오른쪽 단추로 클릭한 다음 시작을 선택하십시오. 서비스를 정지하려면 동일한 단계를 수행한 후 정지를 선택하십시오.

2. 그런 다음, 구성 설정을 위해 원하는 Site Selector 제어 명령을 실행하십시오. 이 책의 절차에서는 명령행을 사용한다고 가정합니다. 명령은 **sscontrol**입니다. 명령에 대한 자세한 내용은 429 페이지의 제 28 장 『Site Selector 명령어 참조서』를 참조하십시오.

sscontrol 명령 매개변수의 최소 버전을 입력할 수 있습니다. 매개변수의 고유한 문자만 입력해야 합니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면 **sscontrol help file** 대신에 **sscontrol he f**를 입력할 수 있습니다.

명령 인터페이스를 시작하려면 **sscontrol** 명령을 실행하여 **sscontrol** 명령 프롬프트를 받으십시오.

명령 인터페이스를 종료하려면 **exit** 또는 **quit** 명령을 실행하십시오.

주: Windows 플랫폼에서 Dispatcher 컴포넌트의 **dsserver**가 자동으로 시작됩니다. Dispatcher 컴포넌트가 아닌 Site Selector만을 사용하는 경우 다음과 같이 **dsserver**의 자동 시작을 정지시킬 수 있습니다.

1. Windows 서비스에서 IBM Dispatcher를 마우스 오른쪽 단추로 클릭하십시오.
2. 등록 정보를 선택하십시오.
3. 시동 유형 필드에서 수동을 선택하십시오.
4. 확인을 클릭하고, 서비스 창을 닫으십시오.

스크립트

Site Selector 구성 명령은 구성 스크립트 파일에 입력되어 함께 실행될 수 있습니다.

주: 스크립트 파일(예: **myscript**)의 콘텐츠를 신속히 실행하려면 다음 명령 중 하나를 사용하십시오.

- 현재 구성을 갱신하려면 다음을 사용하여 스크립트 파일에서 실행 가능 명령을 실행하십시오.

```
sscontrol file appendload myscript
```

- 현재 구성을 완전히 바꾸려면 다음을 사용하여 스크립트 파일에서 실행 가능 명령을 실행하십시오.

```
sscontrol file newload myscript
```

현재 구성을 스크립트 파일(예: `savescript`)에 저장하려면 다음 명령을 실행하십시오.

```
sscontrol file save savescript
```

이 명령은 `...ibm/edge/lb/servers/configurations/ss` 디렉토리에서 구성 스크립트 파일을 저장합니다.

GUI

GUI의 일반 명령 및 예제는 500 페이지의 그림 41을 참조하십시오.

GUI를 시작하려면 다음 단계를 따르십시오.

1. `ssserver`가 실행되고 있는지 확인하십시오. 루트 사용자 또는 관리자로 명령 프롬프트에서 `ssserver` 명령을 실행하십시오.
2. 그런 후 다음 중 하나를 수행하십시오.
 - AIX, HP-UX, Linux 또는 Solaris 시스템의 경우: `lbadm`을 입력하십시오.
 - Windows 시스템의 경우: 시작 > 프로그램 IBM WebSphere > Edge Components > IBM Load Balancer > Load Balancer를 클릭하십시오.

GUI에서 Site Selector 컴포넌트를 구성하려면 먼저 트리 구조에서 **Site Selector**를 선택해야 합니다. 호스트 실행 `ssserver`에 연결한 후, 서버를 포함하는 사이트 이름을 작성하고 관리자를 시작하며 어드바이저를 시작할 수 있습니다.

GUI를 사용하여 `sscontrol` 명령으로 수행할 수 있는 모든 작업이 가능합니다. 예를 들어, 명령행을 사용하여 사이트 이름을 정의하려면 `sscontrol sitename add sitename` 명령을 입력하십시오. GUI에서 사이트 이름을 정의하려면 이름 서버를 마우스 오른쪽 단추로 클릭한 후 표시되는 팝업 메뉴에서 **사이트 이름 추가**를 클릭하십시오. 팝업 창에 사이트 이름을 입력한 후 **확인**을 클릭하십시오.

호스트 팝업 메뉴에 있는 **새 구성 로드** 옵션(현재 구성을 완전히 바꾸기 위한 옵션) 및 **현재 구성에 추가** 옵션(현재 구성을 갱신하기 위한 옵션)을 사용하여 기존 Site Selector 구성 파일을 로드할 수 있습니다. 호스트 팝업 메뉴에 표시된 **다른 이름으로 구성 파일 저장** 옵션을 사용하여 Site Selector 구성을 파일로 정기적으로 저장해야 합니다. GUI의 맨 위에 있는 **파일** 메뉴는 파일에 대한 현재 호스트 연결을 저장하거나 모든 Load Balancer 컴포넌트 전반에 있는 기존 파일의 연결을 복원하게 합니다.

GUI에서 명령을 실행하려면 GUI 트리에서 호스트 노드를 강조표시하고 호스트 팝업 메뉴에서 명령 전송....을 선택하십시오. 명령 입력 필드에서는 **nameserver status**와 같이 실행하려는 명령을 입력하십시오. 현재 세션에서 실행되는 명령의 결과 및 히스토리가 제공된 창에 나타납니다.

Load Balancer 창 오른쪽 상단 구석의 물음표를 클릭하면, 도움말에 액세스할 수 있습니다.

- 도움말: 필드 레벨 — 각 필드 및 기본값을 설명합니다.
- 도움말: 수행 방법 — 해당 화면에서 수행할 수 있는 작업이 나열되어 있습니다.
- InfoCenter — 제품 정보에 대한 중앙집중화된 액세스를 제공합니다.

GUI 사용에 대한 자세한 내용은 499 페이지의 부록 A 『GUI: 일반 명령』을 참조하십시오.

구성 마법사

구성 마법사를 사용 중이면, 다음 단계를 수행하십시오.

1. Site Selector에서 ssserver를 시작합니다.
 - 루트 사용자 또는 관리자로 다음 명령을 실행하십시오.

ssserver

2. Site Selector의 마법사 기능 **sswizard**를 시작하십시오.

sswizard를 실행하면 명령 프롬프트에서 이 마법사를 실행할 수 있습니다. 또는 GUI에 표시된 대로 Site Selector 컴포넌트 메뉴에서 구성 마법사를 선택하십시오.

Site Selector 마법사는 Site Selector 컴포넌트의 기본 구성을 작성하는 프로세스를 단계별로 안내합니다. 네트워크에 대한 질문을 하고 Site Selector가 서버 그룹 간에 통신량을 로드 밸런싱할 수 있도록 사이트 이름을 설정하는 단계를 안내합니다.

Site Selector 시스템 설정

Site Selector 시스템을 설정하려면 루트 사용자(AIX, HP-UX, Linux 또는 Solaris 시스템의 경우) 또는 관리자(Windows 시스템의 경우)여야 합니다.

설정하는 서버 그룹에 사이트 이름으로 사용할 분석할 수 없는 완전한 호스트 이름이 필요합니다. 사이트 이름은 클라이언트가 사용자의 사이트(예: www.yourcompany.com)에 액세스하는 데 사용하는 이름입니다. Site Selector는 DNS를 사용하여 서버 그룹 간에 이 사이트 이름에 대한 통신량을 로드 밸런싱합니다.

1단계. 서버 기능 시작

Site Selector 서버 기능을 시작하려면 명령행에서 **ssserver**를 입력하십시오.

주: 기본 구성 파일(default.cfg)은 ssserver를 시작할 때 자동으로 로드됩니다. default.cfg에 구성을 저장할 경우 이 파일에 저장된 모든 내용이 ssserver가 다음 번에 시작될 때 자동으로 로드됩니다.

2단계. 이름 서버 시작

이름 서버를 시작하려면 **sscontrol nameserver start** 명령을 입력하십시오.

선택적으로, bindaddress 키워드를 사용하는 이름 서버를 시작하여 지정된 주소로만 바인드하십시오.

3단계. 사이트 이름 정의 및 사이트 이름 옵션 설정

Site Selector는 구성된 해당 서버의 사이트 이름으로 전송된 요청 밸런스를 조정합니다.

사이트 이름은 클라이언트가 요청할 분석할 수 없는 호스트 이름입니다. 사이트 이름은 완전한 도메인 이름(예: www.dnsdownload.com)이어야 합니다. 클라이언트가 이 사이트 이름을 요청할 때 사이트 이름과 연관된 서버 IP 주소 중 하나가 리턴됩니다.

사이트 이름을 정의하려면 다음 명령을 발행하십시오.

```
sscontrol sitename add sitename
```

사이트 이름 옵션을 설정하려면 다음 명령을 발행하십시오.

```
sscontrol sitename set sitename option value
```

자세한 내용은 429 페이지의 제 28 장 『Site Selector 명령어 참조서』를 참조하십시오.

4단계. 로드 밸런스 서버 시스템 정의

서버 시스템은 로드 밸런스를 수행하는 응용프로그램이 실행되는 시스템입니다. *server*는 서버 시스템의 기호 이름이거나 점분리 10진수 주소입니다. 3단계에서 사이트 이름에 서버를 정의하려면 다음 명령을 실행하십시오.

```
sscontrol server add sitename:server
```

로드 밸런스를 수행하려면 하나의 사이트 이름 밑에 둘 이상의 서버를 정의해야 합니다.

5단계. 관리자 기능 시작(선택적)

관리자 기능은 로드 밸런스를 향상시킵니다. 관리자 기능을 시작하기 전에 로드 밸런스가 된 모든 시스템에 Metric Server가 설치되었는지 확인하십시오.

관리자를 시작하려면 다음 명령을 발행하십시오.

```
sscontrol manager start
```

6단계. 어드바이저 기능 시작(선택적)

어드바이저는 요청에 응답하는 로드 밸런스 서버 시스템의 능력에 대한 자세한 정보를 관리자에 제공합니다. 어드바이저는 프로토콜마다 고유합니다. []는 다수의 어드바이저를 제공합니다. 예를 들어, 지정된 사이트 이름에 대해 HTTP 어드바이저를 시작하려면 다음 명령을 발행하십시오.

```
sscontrol advisor start http sitename:port
```

7단계. 시스템 메트릭 정의(선택적)

시스템 메트릭 및 Metric Server 사용에 대한 정보는 211 페이지의 『Metric Server』를 참조하십시오.

8단계. 필요한 사이트 이름 비율 설정

어드바이저를 시작하면 로드 밸런스 결정에 포함되는 어드바이저(포트) 정보에 부여된 중요도를 수정할 수 있습니다. 사이트 이름 비율을 설정하려면 **sscontrol sitename set sitename proportions** 명령을 실행하십시오. 자세한 내용은 194 페이지의 『상태 정보에 제공되는 중요성 비율』을 참조하십시오.

서버 시스템의 시스템 설정

Metric Server를 Site Selector 컴포넌트와 함께 사용하십시오. Site Selector가 로드 밸런스하는 모든 서버 시스템에 Metric Server를 설정하는 것에 대한 정보는 211 페이지의 『Metric Server』를 참조하십시오.

제 5 부 Cisco CSS Controller 컴포넌트

이 파트에서는 빠른 시작 구성, 계획 고려사항에 대한 정보를 제공하며 Load Balancer의 Cisco CSS Controller 컴포넌트 구성 메소드에 대해 설명합니다. 다음 장을 포함합니다.

- 153 페이지의 제 15 장 『빠른 시작 구성』
- 157 페이지의 제 16 장 『Cisco CSS Controller 계획』
- 163 페이지의 제 17 장 『Cisco CSS Controller 구성』

제 15 장 빠른 시작 구성

이 빠른 시작 예제에서는 Cisco CSS Controller 컴포넌트를 사용하여 구성을 작성하는 방법을 표시합니다. Cisco CSS Controller는 로드 밸런스 결정을 위한 최적의 서버 선택을 판별하는 데 Cisco CSS Switch를 돕는 서버 가중치 정보를 제공합니다.

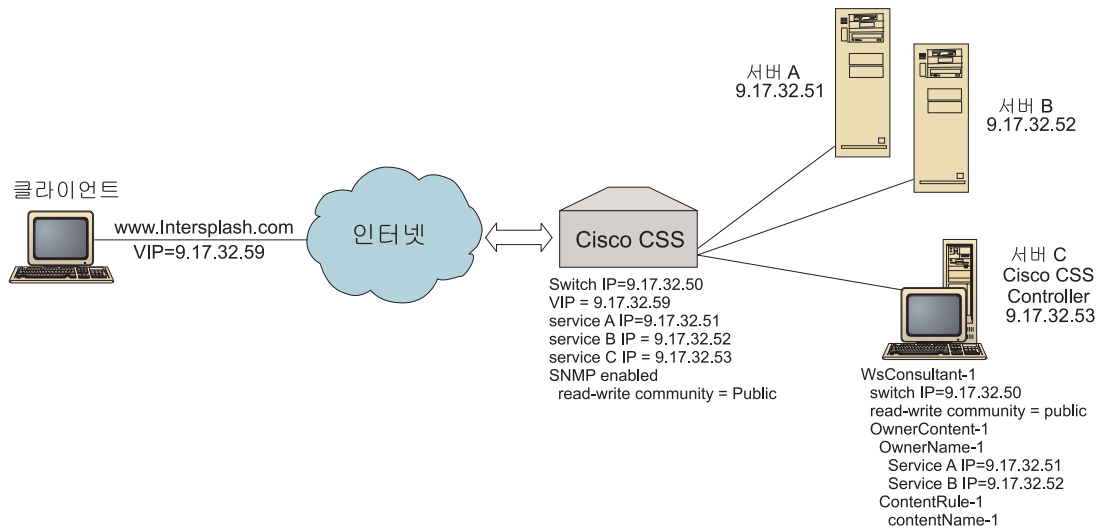


그림 25. 간단한 Cisco CSS Controller 구성

필요한 내용

이 빠른 시작 구성 예의 경우, 다음이 필요합니다.

- Cisco CSS Switch
- Cisco CSS Controller 컴포넌트가 설치된 서버 시스템
- 두 개의 웹 서버 시스템
- 이 구성 예제는 다섯 개의 IP 주소를 요구합니다.
 - www.Intersplashx.com 웹 사이트에 액세스하기 위해 클라이언트에 제공하는 IP 주소(9.17.32.59)
 - Cisco CSS Switch의 인터페이스(게이트웨이)의 IP 주소(9.17.32.50)
 - 서버 A의 IP 주소(9.17.32.51)
 - 서버 B의 IP 주소(9.17.32.52)
 - Cisco CSS Controller 서버 C의 IP 주소(9.17.32.53)

준비 방법

이 예에 대한 구성을 시작하려면 먼저 다음 단계가 완료되어야 합니다.

- Cisco CSS Switch가 올바르게 구성되어 있는지 확인하십시오. 구성 정보는, *Cisco Content Services Switch Getting Started Guide*를 참조하십시오.
- Cisco CSS Controller 시스템이 Cisco CSS Switch(9.17.32.50), 서버 A(9.17.32.51) 및 서버 B(9.17.32.52)를 ping할 수 있는지 확인하십시오.
- 클라이언트 시스템이 VIP(9.17.32.59)를 ping할 수 있는지 확인하십시오.

Cisco CSS Controller 컴포넌트 구성

Cisco CSS Controller를 통해 명령행 또는 GUI(Graphical User Interface)를 사용하여 구성을 작성할 수 있습니다. 이 빠른 시작 예의 경우, 구성 단계는 명령행을 사용하여 예시됩니다.

주: 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름과 파일 이름의 매개변수 값만 예외입니다.

명령행을 사용한 구성

명령 프롬프트에서 다음 단계를 따르십시오

1. Load Balancer에서 ccoserver를 시작하십시오. 루트 사용자 또는 관리자로 명령 프롬프트에서 **ccoserver** 명령을 실행하십시오.
2. Cisco CSS Switch IP 인터페이스 주소 및 읽기-쓰기 공동체 이름을 지정하여, Cisco CSS Controller 구성에 스위치 컨설턴트를 추가하십시오. 이 값은 Cisco CSS Switch의 해당 속성과 일치해야 합니다.

```
cococontrol consultant add SwConsultant-1 address 9.17.32.50 community public
```

이 명령행은 Cisco CSS Switch에 대한 연결을 확인하고 SNMP 읽기-쓰기 공동체 이름이 제대로 작동하는지 확인합니다.

3. **ownername(OwnerName-1)**과 **contentrule(ContentRule-1)**을 지정하여, **ownercontent(OwnerContent-1)**를 스위치 컨설턴트에 추가하십시오.

```
cococontrol ownercontent add SwConsultant-1:OwnerContent-1 ownername OwnerName-1 contentrule ContentRule-1
```

이 값은 Cisco CSS Switch의 해당 속성과 일치해야 합니다.

이제 Cisco CSS Controller는 SNMP를 통해 스위치와 통신할 수 있으며 스위치에서 필수 정보를 확보할 수 있습니다. 이 단계 다음에, 지정된 ownercontent에 대해 Cisco CSS Switch에 구성된 서비스에 관하여 Cisco CSS Controller의 정보를 참조해야 합니다.

4. 수집할 메트릭의 유형(활성 연결, 연결 비율, HTTP) 및 ownercontent의 각 메트릭에 대한 비례를 구성하십시오.

```
cococontrol ownercontent metrics SwConsultant-1:OwnerContent-1 activeconn  
45 connrate 45 http 10
```

이 명령은 가중치 계산에 사용될 서비스에서 수집하려는 메트릭 정보 및 비례를 구성합니다. 모든 메트릭의 비례 총계는 100과 같아야 합니다.

5. Cisco CSS Controller의 스위치 컨설턴트 기능을 시작하십시오.

```
cococontrol consultant start SwConsultant-1
```

이 명령으로 모든 메트릭 콜렉터가 시작되고, 서비스 가중치 계산이 시작됩니다. Cisco CSS Controller는 서비스 가중치 계산의 결과를 SNMP를 사용하여 Cisco CSS Switch와 통신합니다.

사용자의 기본 Cisco CSS Controller 구성을 지금 완료했습니다.

구성 검사

구성이 작동되고 있는지 확인하십시오.

1. 클라이언트 웹 브라우저에서 **http://www.Intersplashx.com** 위치로 이동하십시오. 페이지가 표시되면 구성이 작동하는 것입니다.
2. 웹 브라우저에서 페이지를 재로드하십시오.
3. **cococontrol service report SwConsultant-1:OwnerContent-1:Service-1** 명령의 결과를 확인하십시오. 두 웹 서버의 총 연결 컬럼은 “2”까지 추가해야 합니다.

GUI(Graphical User Interface)를 사용한 구성

Cisco CSS Controller GUI 사용에 대한 정보는 165 페이지의 『GUI』 및 499 페이지의 부록 A 『GUI: 일반 명령』을 참조하십시오.

제 16 장 Cisco CSS Controller 계획

이 장에서는 Cisco CSS Controller 컴포넌트를 설치하고 구성하기 전에 네트워크 계획자가 고려해야 할 사항에 대해 설명합니다.

- Cisco CSS Controller 컴포넌트의 로드 밸런스 매개변수 구성에 대한 정보는 163 페이지의 제 17 장 『Cisco CSS Controller 구성』을 참조하십시오.
- 추가 고급 기능에 대한 Load Balancer 설정 방법에 대한 정보는 261 페이지의 제 23 장 『Cisco CSS Controller 및 Nortel Alteon Controller의 고급 기능』을 참조하십시오.
- 원격 인증 관리, Load Balancer 로그 및 Load Balancer 컴포넌트 사용법에 대한 정보는 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』를 참조하십시오.

이 장에는 다음의 내용이 있습니다.

- 『시스템 요구사항』
- 『계획 고려사항』
 - 158 페이지의 『네트워크의 컨설턴트 배치』
 - 160 페이지의 『고가용성』
 - 161 페이지의 『가중치 계산』
 - 161 페이지의 『문제점 판별』

시스템 요구사항

하드웨어 및 소프트웨어 요구사항에 대해서는 다음 웹 페이지 <http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921>을 참조하십시오.

기타 요구사항

- Cisco CSS Controller를 실행할 시스템.
- Cisco CSS 11000 series content services 스위치가 설치 및 구성되어야 합니다.

계획 고려사항

Cisco CSS Controller는 스위치 컨설턴트의 세트를 관리합니다. 각 컨설턴트는 한 스위치로 로드 밸런싱된 서비스에 대한 가중치를 결정합니다. 컨설턴트가 가중치를 제공하는 스위치가 콘텐츠 로드 밸런싱에 대해 구성됩니다. 컨설턴트는 계산된 가중치를 스위치에 전송하는 데 SNMP 프로토콜을 사용합니다. 스위치는 로드 밸런스 알고리즘이

가중치가 있는 무순위일 경우 로드 밸런싱하는 콘텐츠 규칙에 대한 서비스를 선택하는데 가중치를 사용합니다. 가중치를 판별하기 위해, 컨설턴트는 다음 정보 중 하나 이상을 사용합니다.

- 서비스에서 실행되는 응용프로그램과 통신하는 응용프로그램 어드바이저의 사용을 통해 결정되는 가용성 및 응답 시간
- 서비스에서 실행되는 **Metric Server** 에이전트에서 메트릭 값을 검색하여 결정되는 시스템 로드 정보
- 스위치에서 확보한 서비스에 대한 접속 정보
- 서비스를 ping하여 구하는 도달 가능성 정보.

콘텐츠 로드 밸런싱의 설명 및 스위치 구성에 관한 자세한 정보는 *Cisco Content Services Switch Getting Started Guide*를 참조하십시오.

컨설턴트가 서비스 가중치 결정에 필요한 정보를 확보하려면 다음이 필요합니다.

- 가중치가 계산되는 서비스와 컨설턴트 사이의 IP 연결성
- 가중치가 계산되는 서버를 로드 밸런싱하는 스위치와 컨설턴트 사이의 IP 연결성
- 스위치에서 사용 가능한 SNMP. 읽기 및 쓰기 성능이 모두 사용 가능해야 합니다.

네트워크의 컨설턴트 배치

159 페이지의 그림 26에 지시된 대로, 컨설턴트는 가중치를 제공하는 스위치 뒤에서 네트워크에 접속될 수 있습니다. 제어기, 스위치 및 서비스 사이의 연결성을 사용 가능화하기 위해 일부 매개변수가 스위치에 대해 구성되어야 하며 일부는 제어기에 대해 구성되어야 합니다.

159 페이지의 그림 26에서,

- 컨설턴트는 가중치를 제공하는 스위치 뒤에 네트워크에 접속됩니다.
- 네트워크는 두 VLAN으로 구성됩니다.
- 두 VLAN 모두에서 서비스와 통신하는 컨설턴트의 경우, 서비스가 접속된 인터페이스 및 컨설턴트가 접속된 인터페이스에서 IP 전달이 사용 가능해야 합니다.
- 스위치의 IP 주소는 컨설턴트 및 서비스 시스템에 대한 기본 게이트웨이로 구성되어야 합니다.

VLAN 구성 및 스위치에서 IP 경로 지정에 관한 자세한 정보는 *Cisco Content Services Switch Getting Started Guide*를 참조하십시오.

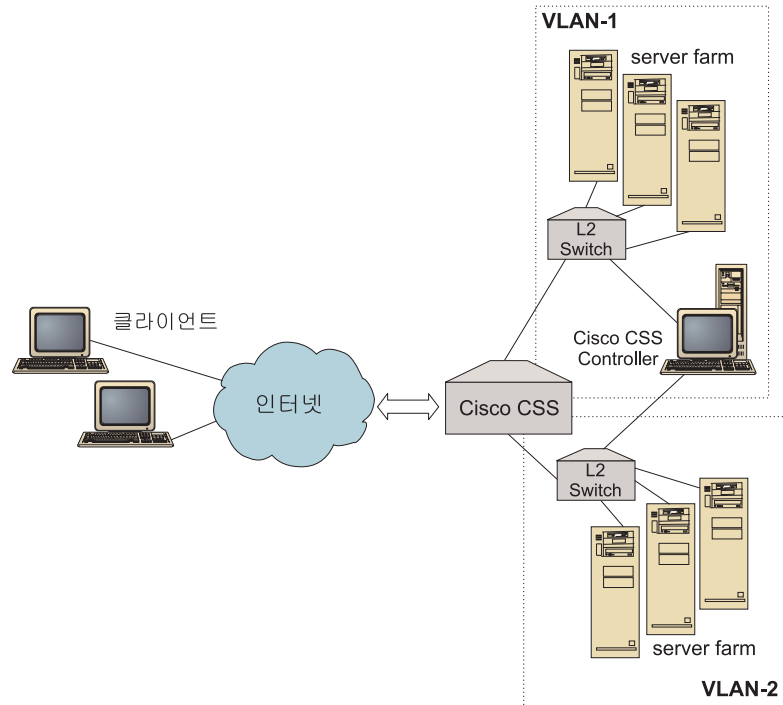


그림 26. 스위치 뒤에 접속된 컨설턴트의 예

다음 인터페이스 중 하나를 사용하여 Cisco CSS Controller를 관리할 수 있습니다.

- 브라우저
- GUI(원격 또는 로컬)
- 명령행(원격 또는 로컬)

원격 관리의 경우, 160 페이지의 그림 27에서:

- 컨설턴트는 가중치를 제공하는 스위치 뒤에 접속됩니다.
- 사용자 인터페이스는 원격 시스템에서 스위치 앞에서 실행됩니다.
- 원격 시스템이 스위치를 통해 제어기 시스템과 통신하도록 스위치가 구성되어야 합니다.

자세한 정보는 *Cisco Content Services Switch Getting Started Guide*를 참조하십시오.

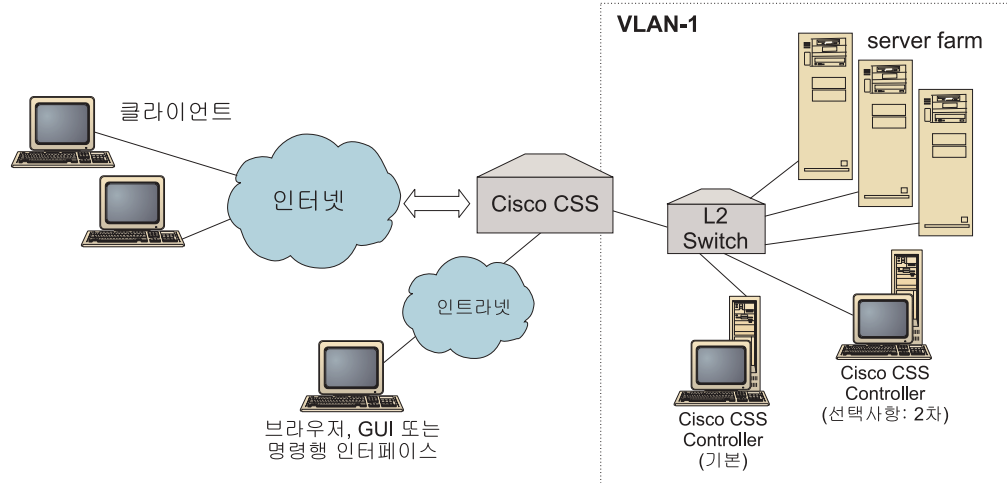


그림 27. 스위치 앞의 사용자 인터페이스를 사용하여 스위치 뒤에 구성된 컨설턴트 예제(선택적 고가용성 상대가 있음)

고가용성

제어기 고가용성은 Load Balancer의 기본 결합 허용 성능을 향상시킵니다. 패킷 전달 고가용성을 염두에 두고 설계되어 제어기 고가용성에는 동시에 실행되는 두 개의 제어기(하나는 기본 역할, 다른 하나는 보조 역할을 함)가 포함됩니다.

각 제어기는 동일한 스위치 정보를 사용하여 구성되며 한 번에 하나의 제어기만이 활성화됩니다. 이는 고가용성 논리에 의해 결정된 대로 활성화 제어기만이 새 가중치를 계산하여 스위치를 갱신함을 의미합니다.

제어기 고가용성은 구성된 주소 및 포트를 통해 단순한 UDP(User Datagram Protocol) 패킷을 사용하여 상대와 통신합니다. 이 패킷은 제어기 간에 고가용성(도달 정보)에 속하는 정보를 교환하고 상대 제어기의 사용가능성(하트 비트)을 결정하는 데 사용됩니다. 대기 제어기가 임의의 이유로 활성화 제어기가 실패했다고 판단하면 대기 제어기는 실패한 활성화 제어기로부터 기능을 인수합니다. 그런 다음 대기 제어기는 활성화 제어기가 되고 새 가중치를 계산하여 스위치를 갱신합니다.

상대 사용가능성뿐 아니라 고가용성에 대해서도 도달 목표를 구성할 수 있습니다. 제어기 고가용성은 도달 정보를 사용하여 활성화 제어기와 대기 제어기를 판별합니다. 활성화 제어기는 대상에 ping을 수행할 수 있고 상대에서 도달할 수 있는 제어기입니다.

261 페이지의 『고가용성』에서 자세한 정보를 참조하십시오.

가중치 계산

컨설턴트가 서비스가 사용 불가능하다고 판단한 경우, 요청을 로드 밸런스할 때 스위치가 서버를 고려하지 못하도록 스위치에서 해당 서버를 일시중단합니다. 서비스가 다시 사용 가능해지면 컨설턴트는 요청을 로드 밸런스할 때 고려할 수 있게 스위치에서 서버를 활성화합니다.

문제점 판별

Cisco CSS Controller는 다음 로그로 항목을 보냅니다

- server.log
- consultant.log
- highavailability.log
- metriccollector.log
- binary.log

이 로그는 다음 디렉토리에 있습니다.

- AIX, HP-UX, Linux 및 Solaris 시스템의 경우: ...ibm/edge/lb/servers/logs/ccol/*consultantName*
- Windows 시스템의 경우: ...ibm\edge\lb\servers\logs\cco*consultantName*

각 로그에서 로그 크기 및 로그 레벨을 설정할 수 있습니다. 285 페이지의 『Load Balancer 로그 사용』에서 자세한 정보를 참조하십시오.

제 17 장 Cisco CSS Controller 구성

이 장에 나와 있는 단계를 수행하기 전에 157 페이지의 제 16 장 『Cisco CSS Controller 계획』을 참조하십시오. 이 장에서는 Load Balancer의 Cisco CSS Controller 컴포넌트에 대한 기본 구성을 작성하는 방법을 설명합니다.

- 복합 구성에 대한 정보는 261 페이지의 제 23 장 『Cisco CSS Controller 및 Nortel Alteon Controller의 고급 기능』을 참조하십시오.
- 원격 인증 관리, 로그 및 Cisco CSS Controller 컴포넌트 사용법에 대한 정보는 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』를 참조하십시오.

구성 task 개요

여기서 구성 방법을 시작하기 전에 다음을 수행하십시오.

1. Cisco CSS Switch 및 모든 서버 시스템이 올바르게 구성되어 있는지 확인하십시오.
2. Cisco CSS Controller를 구성하여 Cisco CSS Switch의 주소 및 SNMP 공동체 이름이 Cisco CSS Switch의 해당 속성과 일치하는지 확인하십시오. 컨설턴트 구성에 대한 정보는 458 페이지의 『ccocontrol consultant — 컨설턴트 구성 및 제어』를 참조하십시오.

표 10. Cisco CSS Controller 컴포넌트 구성 task

task	설명	관련 정보
Cisco CSS Controller 시스템 설정	요구사항 찾기	166 페이지의 『Controller for Cisco CSS Switches 시스템 설정』
구성 검사	구성의 작동 여부 확인	168 페이지의 『구성 검사』

구성 방법

다음은 Load Balancer의 Cisco CSS Controller 컴포넌트에 대해 기본 구성을 작성하기 위한 3가지 방법입니다.

- 명령행
- XML 파일
- GUI(Graphical User Interface)

명령행

이 방법은 Cisco CSS Controller를 구성하는 가장 직접적인 방법입니다. 이 책의 절차에서는 명령행을 사용한다고 가정합니다. 명령 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름(예를 들면, **consultant add** 명령에 사용된)과 파일 이름만은 예외입니다.

명령행에서 Cisco CSS Controller를 시작하려면 다음을 수행하십시오.

1. 명령 프롬프트에서 **ccoserver** 명령을 실행하십시오. 서버를 정지하려면 **ccoserver stop**을 입력하십시오.

주:

- a. Windows 시스템의 경우, 시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스를 클릭하십시오. **IBM Cisco CSS Controller**를 마우스 오른쪽 단추로 클릭한 다음 시작을 선택하십시오. 서비스를 정지하려면 동일한 단계를 수행한 후 정지를 선택하십시오.
 - b. Windows 시스템의 경우, 부트 중에 **ccoserver**를 자동으로 시작할 수 있습니다.
 - 1) 시작 > 설정 > 제어판 > 관리 도구 > 서비스를 클릭하십시오.
 - 2) **IBM Cisco CSS Controller**를 마우스 오른쪽 단추로 클릭한 다음 등록 정보를 선택하십시오.
 - 3) 시동 유형 필드에 대한 화살표를 클릭하고 **자동**을 선택하십시오.
 - 4) 확인을 클릭하십시오.
2. 그 다음, 원하는 Cisco CSS Controller 제어 명령을 발행하여 구성을 설정하십시오. 이 책의 절차에서는 명령행을 사용한다고 가정합니다. 명령은 **cococontrol**입니다. 명령에 대한 자세한 내용은 457 페이지의 제 29 장 『Cisco CSS Controller 명령어 참조서』를 참조하십시오.

cococontrol 명령 매개변수의 축약된 버전을 입력할 수 있습니다. 매개변수의 고유한 문자만 입력해야 합니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면 **cococontrol help file** 대신에 **cococontrol he f**를 입력할 수 있습니다.

명령 인터페이스를 시작하려면 **cococontrol** 명령을 실행하여 **cococontrol** 명령 프롬프트를 받으십시오.

명령 인터페이스를 종료하려면 **exit** 또는 **quit** 명령을 실행하십시오.

주: Windows 플랫폼에서 Dispatcher 컴포넌트의 **dsserver**가 자동으로 시작됩니다. Cisco CSS Controller만을 사용하고 Dispatcher 컴포넌트를 사용하지 않는 경우 다음과 같이 **dsserver**가 자동으로 시작되지 않게 할 수 있습니다.

1. Windows 서비스에서 **IBM Dispatcher**를 마우스 오른쪽 단추로 클릭하십시오.

2. 등록 정보를 선택하십시오.
3. 시동 유형 필드에서 수동을 선택하십시오.
4. 확인을 클릭하고, 서비스 창을 닫으십시오.

XML

현재 정의되어 있는 구성을 XML 파일에 저장할 수 있습니다. 그러면 나중에 구성을 신속하게 다시 작성하고자 할 때 구성을 로드할 수 있습니다.

ML 파일(예: **myscript.xml**)의 콘텐츠를 실행하려면 다음 두 명령 중 하나를 사용하십시오.

- 현재 구성을 XML 파일에 저장하려면 다음 명령을 발행하십시오.

```
cococontrol file save XMLFilename
```

- 저장된 구성을 로드하려면 다음 명령을 발행하십시오.

```
cococontrol file load XMLFileName
```

이전에 파일 저장을 수행한 경우에만 로드 명령을 사용하십시오.

XML 파일은 **...ibm/edge/lb/servers/configurations/cco/** 디렉토리에 저장됩니다.

GUI

GUI(Graphical User Interface)의 일반 명령 및 예제는 500 페이지의 그림 41을 참조하십시오.

GUI를 시작하려면 다음 단계를 따르십시오.

1. ccoserver가 아직 실행되지 않으면 루트로 다음과 같이 실행하여 바로 시작하십시오.

```
ccoserver
```

2. 그런 후 다음 중 하나를 수행하십시오.
 - AIX, HP-UX, Linux 또는 Solaris 시스템의 경우: **lbadm**을 입력하십시오.
 - Windows 시스템의 경우: 시작 > 프로그램 > **IBM WebSphere > Edge Components > IBM Load Balancer > Load Balancer**를 클릭하십시오.

GUI에서 Cisco CSS Controller 컴포넌트를 구성하려면 다음을 수행하십시오.

1. 트리 구조에서 마우스 오른쪽 단추로 Cisco CSS Controller를 클릭하십시오.
2. 호스트에 연결하십시오.
3. 원하는 ownercontents 및 그와 연관된 메트릭이 포함된 switch consultant를 하나 이상 작성하십시오.
4. 컨설턴트를 시작합니다.

GUI를 사용하면 **cococontrol** 명령으로 할 수 있는 모든 작업을 수행할 수 있습니다. 예를 들어,

- 명령행을 사용하여 컨설턴트를 정의하려면 **cococontrol consultant add consultantID address IPAddress community name**을 입력하십시오.
- GUI로부터 컨설턴트를 정의하려면 마우스 오른쪽 단추로 호스트 노드를 클릭하고 스위치 컨설턴트 추가를 클릭하십시오. 팝업 창에 전환 주소 및 공동체 이름을 입력한 후 확인을 클릭하십시오.
- 기존 Cisco CSS Controller 구성 파일을 로드하여 현재 구성을 추가하려면 호스트 팝업 메뉴에 있는 구성 로드를 사용하십시오.
- 다른 구성 파일 이름으로 저장을 선택하여 Cisco CSS Controller 구성을 파일에 주기적으로 저장하십시오.
- 메뉴 표시줄에서 파일을 선택하여 파일에 대한 현재 호스트 연결을 저장하거나 모든 Load Balancer 컴포넌트에 있는 기존 파일의 연결을 복원하십시오.

GUI에서 명령을 실행하려면 다음을 수행하십시오.

1. 호스트 노드를 마우스 오른쪽 단추로 클릭하고 명령 전송...을 선택하십시오.
2. 명령 입력 필드에서 **consultant report**를 실행할 명령을 입력하십시오.
3. 전송을 클릭하십시오.

현재 세션에서 실행하는 명령의 결과 및 히스토리가 결과 상자에 나타납니다.

도움말에 액세스하려면 Load Balancer 창 오른쪽 상단 구석의 물음표를 클릭하십시오.

- **도움말:** 필드 레벨 — 각 필드 및 기본값을 설명합니다.
- **도움말:** 수행 방법 — 해당 화면에서 수행할 수 있는 작업이 나열되어 있습니다.
- **InfoCenter** — 제품 정보에 대한 중앙집중화된 액세스를 제공합니다.

GUI 사용에 대한 자세한 내용은 499 페이지의 부록 A 『GUI: 일반 명령』을 참조하십시오.

Controller for Cisco CSS Switches 시스템 설정

Cisco CSS Controller 시스템을 설정하려면 루트 사용자(AIX, HP-UX, Linux 또는 Solaris 시스템에서) 또는 관리자 (Windows 시스템에서)여야 합니다.

컨설턴트는 Cisco CSS Switch에 Cisco CSS Switch 관리자로서 연결될 수 있습니다.

컨설턴트를 구성할 때, 주소를 구성하고 SNMP 공동체 이름이 Cisco CSS Switch의 해당 속성과 일치해야 합니다.

이 절차에서 사용되는 명령에 대한 도움말을 보려면 457 페이지의 제 29 장 『Cisco CSS Controller 명령어 참조서』를 참조하십시오.

1단계. 서버 기능 시작

ccoserver가 아직 실행되고 있지 않을 경우, 지금 시작하도록 **ccoserver**를 루트로 입력하십시오.

주: Windows 시스템의 경우, 시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스를 클릭하십시오. IBM Cisco Controller를 마우스 오른쪽 단추로 클릭한 다음 시작을 선택하십시오.

2단계. 명령 인터페이스 시작

명령 인터페이스를 시작하려면 **cococontrol**을 입력하십시오.

3단계. 컨설턴트 구성

전환 주소 및 SNMP 공동체 이름을 구성해야 합니다. 이 값은 Cisco CSS Switch의 해당 속성과 일치해야 합니다.

컨설턴트를 추가하려면 다음을 입력하십시오.

```
consultant add switchConsultantID address switchIPAddress  
community communityName
```

3단계. ownercontent 구성

ownercontent는 소유자에 대한 콘텐츠 규칙의 표현이며, Cisco CSS Switch에 정의됩니다. 소유자 이름 및 콘텐츠 규칙 이름은 스위치에 정의되어 있는 방식과 일치해야 합니다.

ownercontent를 정의하려면 다음을 입력하십시오.

```
ownercontent add switchConsultantID:ownercontentID ownername ownerName  
contentrule contentRuleName
```

4단계. 서비스의 올바른 정의 확인

ownercontent가 정의되어 있을 경우, 컨설턴트는 스위치에 구성된 서비스를 검색하여 구성을 완료합니다. 서비스가 일치하는지 확인하려면 스위치에 있는 구성과 컨설턴트에 대한 구성을 비교하십시오.

5단계. 메트릭 구성

메트릭은 서비스 가중치 및 연관된 비례(한 메트릭의 다른 메트릭에 대한 중요도)를 판별하는 데 사용되는 측정치이며, 연결 데이터 메트릭, 응용프로그램 어드바이저 메트릭 및 Metric Server 메트릭의 조합이 될 수 있습니다. 비례의 총계는 항상 100입니다.

ownercontent가 구성되어 있을 경우, 기본 메트릭은 **activeconn** 및 **connrate**로 정의됩니다. 추가 메트릭을 원하거나 모두 기본값과 다른 메트릭을 원할 경우, 다음을 입력하십시오.

```
ownercontent metrics switchConsultantID:ownercontentID metric1 proportion1  
metric2 proportion2...metricN proportionN
```

6단계. 컨설턴트 시작

컨설턴트를 시작하려면 다음을 입력하십시오.

```
consultant start switchConsultantID
```

그러면 메트릭 콜렉터가 시작되고 가중치 계산이 시작됩니다.

7단계. Metric Server 시작(선택)

5단계에서 시스템 메트릭을 정의할 경우, Metric Server가 서버 시스템에서 시작되어야 합니다. Metric Server 사용에 대한 정보는 211 페이지의 『Metric Server』를 참조하십시오.

8단계. 고가용성 구성(선택)

고가용성을 구성하려면 다음을 입력하십시오.

```
highavailability add address IPaddress partneraddress IPaddress port 80  
role primary
```

고가용성 환경에서는 복수 스위치를 구성할 수 있습니다. 한 스위치가 다른 스위치를 이어 받을 때 가중치 정보가 항상 사용 가능하도록 하기 위해, Cisco CSS Controller는 모든 스위치와 그 백업에 대한 가중치를 제공하도록 구성되어야 합니다.

제어기 고가용성 구성 및 사용 방법에 대한 자세한 정보는 261 페이지의 제 23 장 『Cisco CSS Controller 및 Nortel Alteon Controller의 고급 기능』을 참조하십시오.

구성 검사

구성이 작동되고 있는지 확인하십시오.

1. consultant loglevel을 4로 설정하십시오.
2. 1분 동안 Cisco CSS Switch에서 서버의 연결을 끊으십시오. 또는 1분 동안 Application Server를 종료하십시오.
3. 서버를 다시 연결하거나 Application Server를 재시작하십시오.
4. consultant loglevel을 원하는 레벨(1)로 다시 설정하십시오.
5. 다음 디렉토리에 있는 consultant.log 파일을 보고, **setServerWeights setting service**를 찾으십시오.
 - AIX, HP-UX, Linux 및 Solaris 시스템의 경우: ...ibm/edge/lb/servers/logs/ccol/consultantName
 - Windows 시스템의 경우: ...ibm\edge\lb\servers\logs\cco\consultantName

제 6 부 Nortel Alteon Controller 컴포넌트

이 파트에서는 빠른 시작 구성, 계획 고려사항에 대한 정보를 제공하며 Load Balancer의 Nortel Alteon Controller 컴포넌트 구성 메소드에 대해 설명합니다. 다음 장을 포함합니다.

- 171 페이지의 제 18 장 『빠른 시작 구성』
- 175 페이지의 제 19 장 『Nortel Alteon Controller 계획』
- 185 페이지의 제 20 장 『Nortel Alteon Controller 구성』

제 18 장 빠른 시작 구성

이 빠른 시작 예제에서는 Nortel Alteon Controller 컴포넌트를 사용하여 구성을 작성하는 방법을 표시합니다. Nortel Alteon Controller는 Nortel Alteon Web Switch에 서버 가중치를 제공합니다. 이 가중치는 스위치가 로드 밸런스 중인 서비스에 대한 서버를 선택하는 데 사용됩니다.

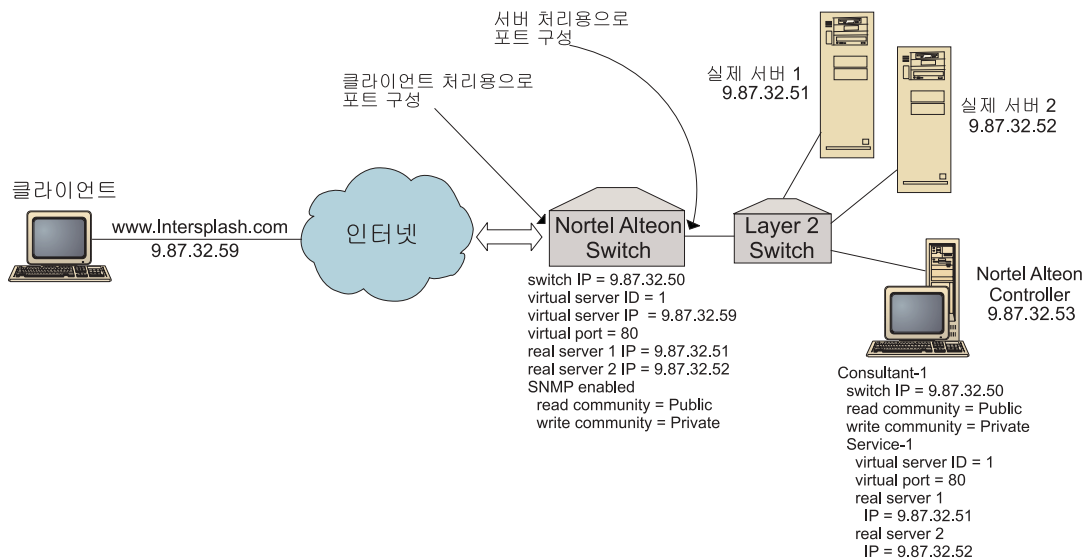


그림 28. 간단한 Nortel Alteon Controller 구성

필요한 내용

이 빠른 시작 구성 예의 경우, 다음이 필요합니다.

- Web OS 버전 9.0 또는 버전 10.0을 실행하는 Nortel Alteon Web Switch
- Nortel Alteon Controller 컴포넌트가 설치된 서버 시스템
- 두 개의 웹 서버 시스템
- Nortel Alteon Web Switch의 포트에 접속된 Layer 2 Switch

주: Layer 2 Switch가 사용되지 않을 경우, Nortel Alteon Controller 시스템 및 웹 서버 시스템이 Nortel Alteon Web Switch의 포트에 직접 접속될 수 있습니다.

- 이 구성 예제는 다섯 개의 IP 주소를 요구합니다.

- www.Intersplashx.com 웹 사이트에 액세스하기 위해 클라이언트에 제공하는 IP 주소(9.87.32.59)
- Nortel Alteon Web Switch에 구성된 인터페이스의 IP 주소(9.87.32.50)
- 실제 서버 1의 IP 주소(9.87.32.51)
- 실제 서버 2의 IP 주소(9.87.32.52)
- Nortel Alteon Controller의 IP 주소(9.87.32.53)

준비 방법

이 예에 대한 구성을 시작하려면 먼저 다음 단계가 완료되어야 합니다.

- Nortel Alteon Web Switch가 올바르게 구성되어 있는지 확인하십시오. (보다 완전한 구성 정보는, Nortel Alteon Web OS Application Guide를 참조하십시오.)
 - 스위치에서 계층 4 서버 로드 밸런스를 사용 가능하게 하십시오.
 - Nortel Alteon Web Switch에서 IP 인터페이스(9.87.32.50)를 구성하십시오.
 - Nortel Alteon Web Switch에서 SNMP를 사용 가능화하십시오.
 - 클라이언트 요청을 받은 Nortel Alteon Web Switch 포트에서 서버 로드 밸런스 클라이언트 처리를 사용 가능화하십시오.
 - 서버가 연결된 Nortel Alteon Web Switch 포트에서 로드 밸런스 서버 처리를 사용 가능하게 하십시오.
 - 실제 서버 1, 실제 서버 2 및 Nortel Alteon Controller에서 switch IP 인터페이스(9.87.32.50)가 될 기본 게이트웨이를 구성하십시오.
 - 실제 서버 1과 실제 서버 2에 Nortel Alteon Web Switch를 구성하십시오.
 - 실제 서버 1과 실제 서버 2로 구성된 서버 그룹에 Nortel Alteon Web Switch를 구성하십시오. 그룹 ID에 1을 지정하십시오.
 - 가상 서버에 Nortel Alteon Web Switch를 구성하십시오. 가상 서버 IP 주소는 9.87.32.59입니다. 가상 서버의 ID로 1을 지정하십시오.
 - 가상 포트 80을 사용하고 그룹 1에서 제공하는 서비스를 사용하여 Nortel Alteon Web Switch를 구성하십시오.
- 클라이언트 시스템이 가상 서버 IP 주소 9.87.32.59를 ping할 수 있는지 확인하십시오.
- Nortel Alteon Controller 시스템이 Nortel Alteon Web Switch IP 인터페이스 (9.87.32.50), 실제 서버 1(9.87.32.51) 및 실제 서버 2(9.87.32.52)를 ping할 수 있는지 확인하십시오.

Nortel Alteon Controller 컴포넌트 구성

Nortel Alteon Controller를 통해 명령행 또는 GUI(Graphical User Interface)를 사용하여 구성을 작성할 수 있습니다. 이 빠른 시작 예의 경우, 구성 단계는 명령행을 사용하여 예시됩니다.

주: 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름과 파일 이름의 매개변수 값만 예외입니다.

명령행을 사용한 구성

명령 프롬프트에서 다음 단계를 따르십시오

1. Nortel Alteon Controller에서 `nalserver`를 시작하십시오. 루트 사용자 또는 관리자 명령 프롬프트에서 **`nalserver`** 명령을 실행하십시오.
2. Nortel Alteon Web Switch IP 인터페이스 주소를 지정하여, Nortel Alteon Controller 구성에 컨설턴트를 추가하십시오. (기본값(public, private)과 다를 경우 읽기 공동체 및 쓰기 공동체만 지정하십시오.)

`nalcontrol consultant add Consultant-1 address 9.87.32.50`

이 명령행은 Nortel Alteon Web Switch에 대한 연결을 확인하고 SNMP 공동체 이름이 제대로 작동하는지 확인합니다.

3. 서비스에 대한 가상 서버 ID(1) 및 가상 포트 번호(80)를 지정하여, 컨설턴트(Consultant-1)에 서비스(Service-1)를 추가하십시오.

`nalcontrol service add Consultant-1:Service-1 vsid 1 vport 80`

이제 Nortel Alteon Controller는 SNMP를 통해 스위치와 통신하여 스위치에서 필수 정보를 확보합니다. 이 단계 다음에, 서비스를 위해 Nortel Alteon Web Switch에 구성된 서버에 관하여 Nortel Alteon Controller의 정보를 참조해야 합니다.

4. 서비스와 연관된 서버 세트에 대해 수집되는 메트릭을 구성하십시오.

`nalcontrol service metrics Consultant-1:Service-1 http 40 activeconn 30 connrate 30`

이 명령은 서버에서 수집할 메트릭 정보와 가중치를 계산하는 중에 이 메트릭의 상대적 중요도를 구성합니다.

5. Nortel Alteon Controller의 컨설턴트 기능을 시작하십시오.

`nalcontrol consultant start Consultant-1`

이 명령으로, 모든 메트릭 콜렉터가 시작되고 서버 가중치 계산이 시작됩니다. Nortel Alteon Controller는 서버 가중치 계산의 결과를 SNMP를 사용하여 Nortel Alteon Web Switch와 통신합니다.

사용자의 기본 Nortel Alteon Controller 구성을 지금 완료했습니다.

구성 검사

구성이 작동되고 있는지 확인하십시오.

1. 클라이언트 웹 브라우저에서 **http://www.Intersplashx.com** 위치로 이동하십시오.
페이지가 표시되면 구성이 작동하는 것입니다.
2. 웹 브라우저에서 페이지를 재로드하십시오.
3. **nalcontrol service report Consultant-1:Service-1** 명령의 결과를 확인하십시오.
두 웹 서버의 총 연결 컬럼은 “2”까지 추가해야 합니다.

GUI(Graphical User Interface)를 사용한 구성

Nortel Alteon Controller GUI 사용에 대한 정보는 187 페이지의 『GUI』 및 499 페이지의 부록 A 『GUI: 일반 명령』을 참조하십시오.

제 19 장 Nortel Alteon Controller 계획

이 장에서는 Nortel Alteon Controller 컴포넌트를 설치하고 구성하기 전에 네트워크 계획자가 고려해야 할 사항에 대해 설명합니다.

- Nortel Alteon Controller 컴포넌트의 로드 밸런스 매개변수 구성에 대한 정보는 185 페이지의 제 20 장 『Nortel Alteon Controller 구성』을 참조하십시오.
- 어드바이저 및 Metric Server의 구성 방법에 대한 정보는 261 페이지의 제 23 장 『Cisco CSS Controller 및 Nortel Alteon Controller의 고급 기능』을 참조하십시오.
- 원격 인증 관리, Load Balancer 로그 및 Load Balancer 컴포넌트 사용법에 대한 정보는 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』를 참조하십시오.

이 장에는 다음의 내용이 있습니다.

- 『시스템 요구사항』
- 176 페이지의 『계획 고려사항』
 - 176 페이지의 『네트워크의 컨설턴트 배치』
 - 179 페이지의 『스위치에서의 서버 속성(제어기에 의해 설정됨)』
 - 179 페이지의 『백업 서버 구성』
 - 180 페이지의 『그룹 구성』
 - 181 페이지의 『고가용성』
 - 182 페이지의 『조정』
 - 183 페이지의 『문제점 판별』

시스템 요구사항

하드웨어 및 소프트웨어 요구사항에 대해서는 다음 웹 페이지 <http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921>을 참조하십시오.

기타 요구사항

- Nortel Alteon Controller를 실행할 시스템.
- 설치 및 구성된 Nortel Alteon Web Switch. 웹 스위치 하드웨어 플랫폼은 AD3, AD4, 180e 184 및 layer 4/7 blade for the Passport 8600입니다.

계획 고려사항

Nortel Alteon Controller는 스위치 컨설턴트의 세트를 관리합니다. 각 컨설턴트는 한 스위치에 의해 로드 밸런싱된 서버에 대한 가중치를 결정합니다. 컨설턴트가 가중치를 제공하는 스위치가 서버 로드 밸런싱에 대해 구성됩니다. 컨설턴트는 계산된 가중치를 스위치에 전송하는 데 SNMP 프로토콜을 사용합니다. 스위치는 로드 밸런싱하는 서비스에 대한 서버를 선택하는 데 가중치를 사용합니다. 가중치를 판별하기 위해, 컨설턴트는 다음 정보 중 하나 이상을 사용합니다.

- 서버에서 실행되는 응용프로그램과 통신하는 어드바이저의 사용을 통해 결정되는 가용성 및 응답 시간
- 서버에서 실행되는 **Metric Server** 에이전트에서 메트릭 값을 검색하여 결정되는 시스템 로드 정보
- 스위치에서 확보한 서버에 대한 접속 정보
- 서버를 ping하여 구하는 도달 가능성 정보.

서버 로드 밸런싱의 설명 및 스위치 구성에 대한 자세한 정보는 Nortel Alteon Web OS Application Guide를 참조하십시오.

컨설턴트가 서버 가중치 결정에 필요한 정보를 확보하려면 다음이 필요합니다.

- 가중치가 계산될 서버와 컨설턴트 사이의 IP 연결성
- 가중치가 계산되는 서버를 로드 밸런싱하는 스위치와 컨설턴트 사이의 IP 연결성
- 스위치에서 사용 가능한 SNMP. 읽기 및 쓰기 성능이 모두 사용 가능해야 합니다.

네트워크의 컨설턴트 배치

컨설턴트는 가중치를 제공하는 스위치 앞이나 뒤에서 네트워크에 접속될 수 있습니다. 제어기, 스위치 및 서버 사이의 연결성을 사용 가능화하기 위해 일부 매개변수는 스위치에 대해 구성되어야 하며 일부는 제어기에 대해 구성되어야 합니다.

177 페이지의 그림 29에서,

- 컨설턴트는 가중치를 제공하는 스위치 뒤에 네트워크에 접속됩니다.
- 네트워크는 두 VLAN으로 구성됩니다.
- 두 VLAN에서 서버와 통신하는 컨설턴트의 경우, 서버가 접속된 인터페이스 및 컨설턴트가 접속된 인터페이스에서 IP 전달이 사용 가능해야 합니다.
- 스위치의 IP 주소는 컨설턴트 및 서버 시스템에 대한 기본 게이트웨이로 구성되어야 합니다.

스위치에서의 IP 경로 지정 및 VLAN 구성에 관한 자세한 정보는 Nortel Alteon Web OS Application Guide 또는 명령어 참조서를 참조하십시오.

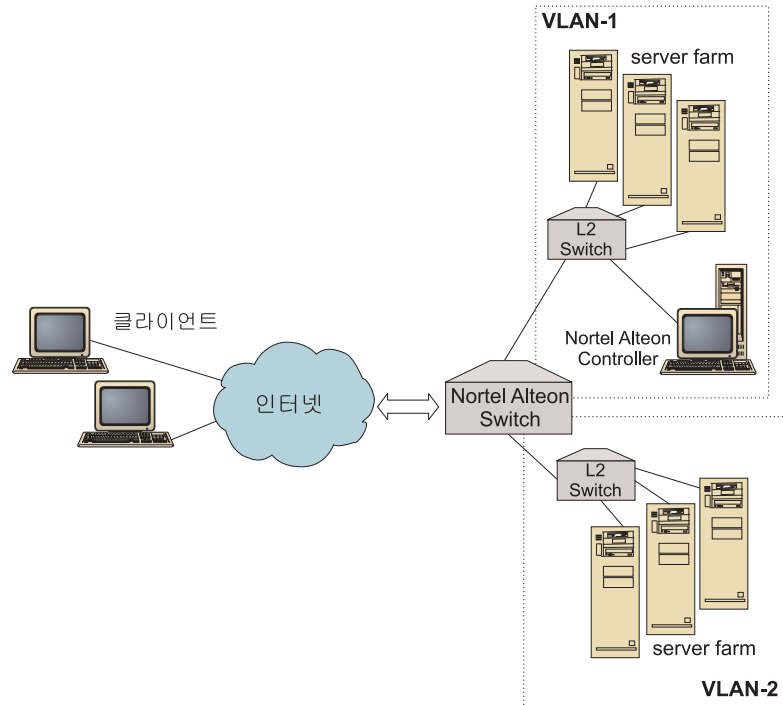


그림 29. 스위치 뒤에 접속된 컨설턴트의 예

178 페이지의 그림 30에서,

- 컨설턴트가 스위치 앞의 인트라넷을 통해 스위치에 접속됩니다.
- 컨설턴트가 스위치 및 서버와 통신하려면 서버 로드 밸런스 직접 액세스 모드가 스위치에서 사용 가능해야 합니다.
- 서버 로드 밸런스 직접 액세스 모드가 사용 가능하면 임의의 클라이언트가 서버에 직접 통신량을 전송할 수 있습니다. 직접 서버 액세스를 컨설턴트로만 제한하기 위해 로드 밸런스 *mnet* 및 *mmask*를 스위치에 지정할 수 있습니다. 서버 로드 밸런스 구성 및 직접 서버 상호작용에 관한 자세한 정보는 Nortel Alteon Web OS Application Guide 또는 명령어 참조서를 참조하십시오.

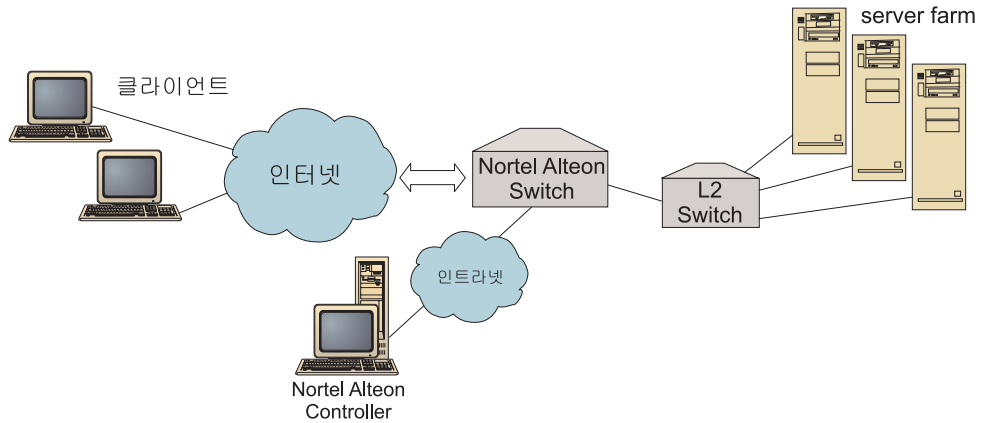


그림 30. 스위치 앞에 인트라넷을 통해 접속된 컨설턴트의 예

다음 인터페이스 중 하나를 사용하여 Nortel Alteon Controller를 관리할 수 있습니다.

- 브라우저
- GUI
- 원격 명령행

그림 31에서,

- 컨설턴트는 가중치를 제공하는 스위치 뒤에 접속됩니다.
- 사용자 인터페이스는 원격 시스템에서 스위치 앞에 실행됩니다.
- 네트워크가 구성되어 있어야 사용자 인터페이스가 제어기와 통신할 수 있습니다.

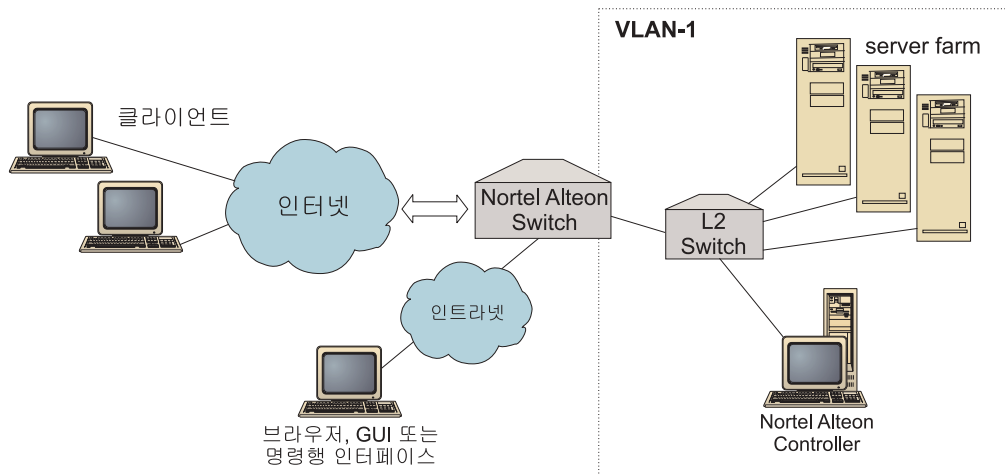


그림 31. 스위치 뒤의 컨설턴트 및 스위치 앞의 사용자 인터페이스 예

스위치에서의 서버 속성(제어기에 의해 설정됨)

컨설턴트가 스위치에 의해 로드 밸런싱된 서비스를 제공하는 서버에 대해 가중치를 계산할 경우, 컨설턴트는 서버로의 불필요한 통신량을 줄이기 위해 스위치에서 정상 서버 헬스 점검을 사용 불가능화합니다. 컨설턴트는 서비스에 대한 가중치 제공을 정지할 때 헬스 점검을 다시 사용 가능화합니다. 서버 헬스 점검 간격은 MIB 변수 `slbNewCgRealServerPingInterval`에 해당합니다.

컨설턴트가 서버가 사용 불가능하다고 판별할 경우, 컨설턴트는 서버의 최대 접속 수를 0으로 설정하여 요청을 로드 밸런싱할 때 스위치가 서버를 고려하지 못하도록 합니다. 서버가 다시 사용 가능하면 최대 접속 수는 원래 값으로 복원됩니다. 서버 최대 접속 값은 MIB 변수 `slbNewCfgRealServerMaxCons`에 해당합니다.

실제 서버에 대한 가중치를 계산할 때 서버에 대한 가중치가 설정됩니다. 서버 가중치 값은 MIB 변수 `slbNewCfgRealServerWeight`에 해당합니다.

백업 서버 구성

스위치를 사용하면 일부 서버가 다른 서버의 백업이 되는 구성이 가능합니다. 스위치가 백업이 있는 서버가 사용 불가능하다고 판단하면 스위치는 백업으로 요청을 전송합니다. 컨설턴트가 백업이 있는 서비스에 대해 가중치를 계산할 때 백업 및 기본 서버 모두에 대한 가중치를 계산하므로 백업이 필요할 때 서버 선택에 사용할 가중치를 갖게 됩니다.

백업 서버의 가중치는 기본 서버의 가중치보다 높을 수 있습니다. 이는 어떤 요청도 전달되지 않으므로 스위치가 사용하기로 결정하기 전에는 로드가 낮기 때문입니다.

대기 서버 자원이 없도록 하기 위해 한 서비스에 지정된 서버를 다른 서비스에 지정된 서버의 백업으로 사용하는 것이 일반적인 방법입니다. 이와 같이 구성을 구현할 경우 동일한 실제 서버를 여러 동시 활성 서비스에 지정하지 마십시오. 이런 현상이 발생할 경우, 서버의 가중치는 서버가 파트인 각 서비스의 컨설턴트에 의해 겹쳐쓰여집니다.

각 실제 서버는 정수로 식별되며 가중치 및 IP 주소 속성을 갖고 있습니다. 두 개의 실제 서버가 동일한 IP 주소를 가질 수 있습니다. 이 경우, 두 개의 실제 서버는 동일한 물리적 서버 시스템에 연관될 수 있습니다. 백업으로 식별된 실제 서버는 단일 서비스의 백업으로만 구성될 수 있습니다. 동일한 물리적 서버 시스템이 여러 서비스에 연관된 서버를 백업할 경우 이 시스템은 한 서비스에 대해 한 번만 구성되어 있어야 하며 각 서비스에 대해 고유한 서버 ID를 제공해야 합니다. 이를 통해 백업은 백업 중인 각 서비스에 고유한 가중치를 지정합니다.

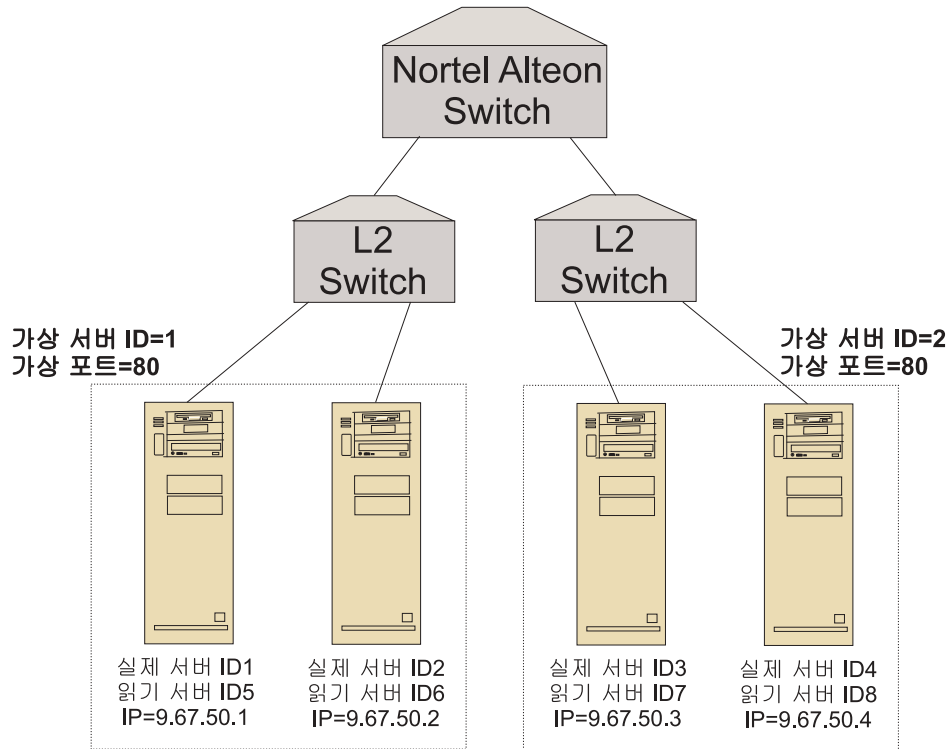


그림 32. 백업 서버에 구성된 컨설턴트 예제

그룹 구성

스위치의 서버는 여러 그룹의 일부로서 구성될 수 있고, 스위치의 그룹은 여러 서비스를 서비스하도록 구성될 수 있습니다.

여러 서비스에 대해 동일한 서버를 구성할 수 있기 때문에 서버가 파트인 각 서비스에 대한 가중치를 계산합니다. 따라서 가중치가 의도하는 서비스가 무엇인지 명확하지 않기 때문에 가중치가 틀릴 가능성이 있습니다.

또한 컨설턴트가 한 서비스에 대해서는 가중치를 결정하고 다른 서비스에 대해서는 결정하지 않을 경우, 컨설턴트가 가중치를 계산하지 않은 서비스가 서버 상태 점검을 사용 불가능하게 할 가능성이 있습니다. 이 경우, 스위치가 해당 서비스를 제대로 로드 밸런스하지 못합니다.

이러한 가능성 때문에, 실제 서버가 로드 밸런스 중인 여러 서비스에 지정되지 않는지 확인해야 합니다. 동일한 서버 시스템이 여러 서비스에 대한 요청을 처리할 수 없다는 것을 의미하지는 않습니다. 스위치에 서버 시스템이 요청을 처리할 각 서비스에 대해 고유한 ID를 가진 실제 서버가 구성되어 있어야 합니다.

고가용성

Nortel Alteon Controller 및 Nortel Alteon Web Switch에 고가용성 성능이 있습니다.

항상 대기 구성의 여러 시스템에서 실행할 두 제어기를 구성할 수 있습니다.

둘 이상의 스위치를 VIR(virtual IP interface router)로 구성하거나 VSR(virtual IP server router)로 구성하면 두 스위치가 서로 백업할 수 있습니다.

한 컨설턴트(제어기가 관리하는)가 유일한 스위치에 대한 가중치를 제공합니다. 백업 스위치가 마스터를 인수할 수 있으므로, 마스터가 될 가능성이 있는 각 스위치에 대해 한 컨설턴트가 있도록 제어기를 구성해야 합니다. 이런 방법으로, 스위치가 마스터가 될 경우 가중치가 제공되도록 보장됩니다.

또한 제어기가 VIR에 접속될 때 서버, 스위치 및 백업 제어기(스위치 중 하나와의 연결이 끊길 경우)와의 통신이 보장됩니다.

스위치의 고가용성에 관한 정보는 Nortel Alteon Web OS Application Guide를 참조하십시오.

제어기 고가용성은 Load Balancer의 기본 결합 허용 성능을 향상시킵니다. 대표적인 패킷 전달 고가용성을 염두에 두고 설계되어 제어기 고가용성에는 동시에 실행되는 두 개의 제어기(하나는 기본 역할, 다른 하나는 보조 역할을 함)가 포함됩니다.

각 제어기는 동일한 스위치 정보로 구성됩니다. 대표적인 고가용성과 유사하게 한 번에 하나의 제어기가 활성화됩니다. 이는 고가용성 논리에 의해 결정된 대로 활성 제어기만이 새 가중치를 계산하여 스위치를 갱신함을 의미합니다.

제어기 고가용성은 구성된 주소 및 포트를 통해 단순한 UDP(User Datagram Protocol) 패킷을 사용하여 상대와 통신합니다. 이 패킷은 제어기 간에 고가용성(도달 정보)에 속하는 정보를 교환하고 상대 제어기의 사용가능성(하트 비트)을 결정하는 데 사용됩니다. 대기 제어기가 임의의 이유로 활성 제어기가 실패했다고 판단하면 대기 제어기는 실패한 활성 제어기로부터 기능을 인수합니다. 그런 다음 대기 제어기는 활성 제어기가 되고 새 가중치를 계산하여 스위치를 갱신합니다.

상대 사용가능성뿐 아니라 고가용성에 대해서도 도달 목표를 구성할 수 있습니다. 대표적인 고가용성에서와 같이, 제어기 고가용성은 도달 정보를 사용하여 활성 제어기와 대기 제어기를 판별합니다. 활성 제어기는 대상에 ping을 수행할 수 있고 상대에서 도달할 수 있는 제어기입니다.

261 페이지의 『고가용성』에서 자세한 정보를 참조하십시오.

182 페이지의 그림 33에서,

- 두 Nortel Alteon Controller가 스위치 뒤에 접속됩니다.

- 한 제어기는 기본이며 활동적으로 스위치에 서버 가중치를 제공합니다. 다른 제어기는 백업입니다.
- 제어기에는 백업이 기본 책임을 인수할 때 인식해야 할 TCP/IP 통신이 있어야 합니다.
- VIR 및 VSR 두 개의 Nortel Alteon Web Switch가 구성됩니다.
- VIR은 서버와의 접속에 대한 고가용성을 제공합니다.
- VSR은 스위치에 구성된 가상 서버로의 액세스를 위한 고가용성을 제공합니다.
- 스위치 중 하나는 마스터이고 다른 하나는 백업입니다.
- 기본 제어기가 두 스위치에 대한 가중치를 제공합니다.
- 백업 제어기가 인수 시기를 결정하기 위해 기본 제어기로 하트 비트를 보냅니다.

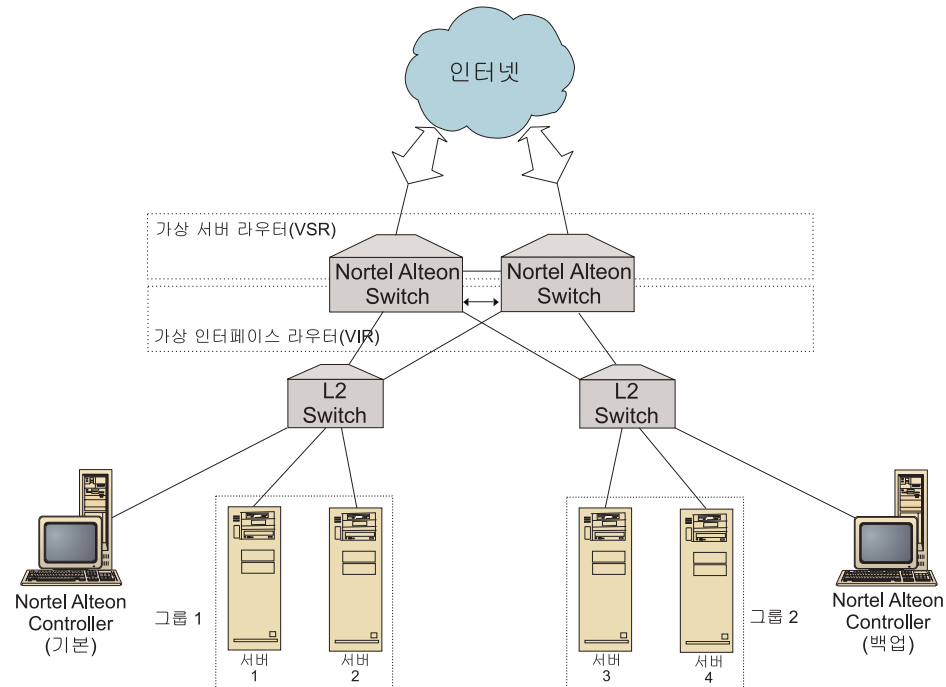


그림 33. Nortel Alteon Controller 및 Nortel Alteon Web Switch 고가용성 예제

조정

가중치를 너무 자주 변경하는 것을 방지하기 위해, 감도 임계치로 컨설턴트를 구성할 수 있습니다. 감도 임계치는 가중치를 변경하기 위해서 발생해야 하는 이전 가중치와 새 가중치 사이의 변경의 양을 지정합니다. 266 페이지의 『감도 임계치』에서 자세한 정보를 참조하십시오.

스위치가 가중치를 갱신하는 데 너무 많이 사용될 경우, 컨설턴트 휴면 시간을 늘려 제 어기와 서버 및 스위치 간의 통신량을 줄일 수 있습니다. 휴면 시간은 가중치 설정 주기 사이에 휴면해야 하는 시간(초)을 설정합니다.

서버가 컨설턴트로부터 너무 많은 모니터링 요청을 처리할 경우 메트릭 콜렉터의 휴면 시간을 수정할 수 있습니다. 세부 설명은 266 페이지의 『가중치 계산 휴면 시간』을 참조하십시오.

문제점 판별

Cisco CSS Controller는 다음 로그로 항목을 보냅니다

- server.log
- consultant.log
- highavailability.log
- metriccollector.log
- binary.log

이 로그는 다음 디렉토리에 있습니다.

- AIX, HP-UX, Linux 및 Solaris 시스템의 경우: ...ibm/edge/lb/servers/logs/nal/*consultantName*
- Windows 시스템의 경우: ...ibm\edge\lb\servers\logs\nal\consultantName

각 로그에서 로그 크기 및 로그 레벨을 설정할 수 있습니다. 285 페이지의 『Load Balancer 로그 사용』에서 자세한 정보를 참조하십시오.

제 20 장 Nortel Alteon Controller 구성

이 장의 단계를 수행하기 전에, 175 페이지의 제 19 장 『Nortel Alteon Controller 계획』을 참조하십시오. 이 장에서는 Load Balancer의 Nortel Alteon Controller 컴포넌트에 대한 기본 구성을 작성하는 방법을 설명합니다.

- 복합 구성에 대한 정보는 261 페이지의 제 23 장 『Cisco CSS Controller 및 Nortel Alteon Controller의 고급 기능』을 참조하십시오.
- 원격 인증 관리, 로그 및 Nortel Alteon Controller 컴포넌트 사용법에 대한 정보는 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』를 참조하십시오.

구성 task 개요

여기서 구성 방법을 시작하기 전에 Nortel Alteon Web Switch 및 모든 서버 시스템이 적절하게 구성되었는지 확인하십시오.

표 11. Nortel Alteon Controller 컴포넌트 구성 task

task	설명	관련 정보
Nortel Alteon Web Switch 및 서버 구성	스위치 구성	페이지 188의 스위치 구성
Nortel Alteon Controller 시스템 설정	제어기 구성	189 페이지의 『1단계. 서버 기능 시작』
구성 검사	구성의 작동 여부 확인	190 페이지의 『구성 검사』

구성 방법

다음은 Load Balancer의 Nortel Alteon Controller 컴포넌트에 대해 기본 구성을 작성하기 위한 3가지 방법입니다.

- 명령행
- XML 파일
- GUI(Graphical User Interface)

명령행

이 방법은 Nortel Alteon Controller를 구성하는 가장 직접적인 방법입니다. 이 책의 절차에서는 명령행을 사용한다고 가정합니다.

명령행에서 Nortel Alteon Controller를 시작하려면 다음을 수행하십시오.

1. 명령 프롬프트에서 **nalserver** 명령을 실행하십시오. 서비스를 정지하려면 **nalserver stop**을 입력하십시오.

주:

- a. Windows 시스템의 경우, 시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스를 클릭하십시오. IBM Nortel Alteon Controller를 마우스 오른쪽 단추로 클릭한 다음 시작을 선택하십시오. 서비스를 정지하려면 동일한 단계를 수행한 후 정지를 선택하십시오.
 - b. Windows 시스템의 경우, 부트 중에 nalserver를 자동으로 시작할 수 있습니다.
 - 1) 시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스를 클릭하십시오.
 - 2) IBM Nortel Alteon Controller를 마우스 오른쪽 단추로 클릭한 다음 등록 정보를 선택하십시오.
 - 3) 시동 유형 필드에 대한 화살표를 클릭한 다음 자동을 선택하십시오.
 - 4) 확인을 클릭하십시오.
2. 그 다음, 원하는 Nortel Alteon Controller 제어 명령을 발행하여 구성을 설정하십시오. 이 책의 절차에서는 명령행을 사용한다고 가정합니다. 명령은 **nalcontrol**입니다. 명령에 대한 자세한 내용은 477 페이지의 제 30 장 『Nortel Alteon Controller 명령어 참조서』를 참조하십시오.

매개변수의 고유한 문자를 입력하여 **nalcontrol** 명령의 축약된 버전을 사용할 수 있습니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면 **nalcontrol help file** 대신에 **nalcontrol he f**를 입력할 수 있습니다.

명령 인터페이스를 종료하려면 **exit** 또는 **quit** 명령을 입력하십시오.

주:

1. 모든 명령 매개변수 값에는 영어 문자를 사용해야 합니다. 호스트 이름(서버 명령에서 사용)과 파일 이름(파일 명령에 사용)만 예외입니다.
2. Windows 시스템에서 Dispatcher 컴포넌트의 dsserver가 자동으로 시작됩니다. Nortel Alteon Controller만 사용하고 Dispatcher 컴포넌트는 사용하지 않는 경우, 다음과 같이 ndserver가 자동으로 시작하는 것을 정지시킬 수 있습니다.
 - a. Windows 서비스에서 IBM Dispatcher를 마우스 오른쪽 단추로 클릭하십시오.
 - b. 등록 정보를 선택하십시오.
 - c. 시동 유형 필드에서 수동을 선택하십시오.
 - d. 확인을 클릭하고, 서비스 창을 닫으십시오.

XML

현재 정의되어 있는 구성을 XML 파일에 저장할 수 있습니다. 그러면 나중에 구성을 신속하게 다시 작성하고자 할 때 구성을 로드할 수 있습니다.

XML 파일(예: **myscript.xml**)의 콘텐츠를 실행하려면 다음 명령을 사용하십시오.

- 현재 구성을 XML 파일에 저장하려면 다음 명령을 발행하십시오.

```
nalcontrol file save XMLFilename
```

이전에 파일 저장을 수행한 경우에만 로드 명령을 사용하십시오.

- 저장된 구성을 로드하려면 다음 명령을 발행하십시오.

```
nalcontrol file load XMLFileName
```

이전에 파일 저장을 수행한 경우에만 로드 명령을 사용하십시오.

XML 파일은 **...ibm/edge/lb/servers/configurations/nal/** 디렉토리에 저장됩니다.

GUI

GUI(Graphical User Interface) 예제는 500 페이지의 그림 41을 참조하십시오.

GUI를 시작하려면 다음을 실행하십시오.

1. nalserver가 아직 실행되지 않으면 루트로 **nalserver**를 입력하여 바로 시작하십시오.
2. 그런 후 다음 중 하나를 수행하십시오.
 - AIX, HP-UX, Linux 또는 Solaris 시스템의 경우: **lbadm**을 입력하십시오.
 - Windows 시스템의 경우: 시작 > 프로그램 > **IBM WebSphere > Edge Components > IBM Load Balancer > Load Balancer**를 클릭하십시오.

GUI에서 Nortel Alteon Controller 컴포넌트를 구성하려면 다음을 수행하십시오.

1. 트리 구조에서 마우스 오른쪽 단추로 Nortel Alteon Controller를 클릭하십시오.
2. 호스트에 연결하십시오.
3. 원하는 서비스 및 이와 연관된 메트릭이 포함된 switch consultant를 하나 이상 작성하십시오.
4. 컨설턴트를 시작합니다.

GUI를 사용하면 **nalcontrol** 명령으로 할 수 있는 모든 작업을 수행할 수 있습니다. 예를 들어,

- 명령행을 사용하여 도달 목표를 정의하려면 **nalcontrol highavailability usereach address**를 입력하십시오. GUI에서 도달 목표를 정의하려면 고가용성 > 도달 목표 추가....를 마우스 오른쪽 단추로 클릭하십시오. 팝업 창에 도달 주소를 입력한 다음 확인을 클릭하십시오.
- 호스트 팝업 메뉴에 표시된 구성 로드를 사용하여 파일에 저장된 구성을 실행 중인 구성에 추가하십시오. 새 구성을 로드하려면, 파일을 로드하기 전에 서버를 정지시킨 후 재시작해야 합니다.
- 호스트 노드를 마우스 오른쪽 단추로 클릭한 다음, 다른 구성 파일 이름으로 저장을 선택하여 Nortel Alteon Controller 구성을 파일에 주기적으로 저장하십시오.

- 메뉴 표시줄에서 파일을 선택하여 파일에 대한 현재 호스트 연결을 저장하거나 모든 Load Balancer 컴포넌트에 있는 기존 파일의 연결을 복원하십시오.

GUI에서 명령을 실행하려면 다음을 수행하십시오.

1. 호스트 노드를 마우스 오른쪽 단추로 클릭하고 **명령 전송....**을 선택하십시오.
2. 명령 입력 필드에서 **consultant report**를 실행할 명령을 입력하십시오.
3. 전송을 클릭하십시오.

현재 세션에서 실행하는 명령의 결과 및 히스토리가 결과 상자에 나타납니다.

도움말에 액세스하려면 Load Balancer 창 오른쪽 상단 구석의 물음표를 클릭하십시오.

- **도움말: 필드 레벨** — 각 필드 및 기본값을 설명합니다.
- **도움말: 수행 방법** — 해당 화면에서 수행할 수 있는 작업이 나열되어 있습니다.
- **InfoCenter** — 제품 정보에 대한 중앙집중화된 액세스를 제공합니다.

GUI 사용에 대한 자세한 내용은 499 페이지의 부록 A 『GUI: 일반 명령』을 참조하십시오.

Nortel Alteon Controller 설정

이 절차에서 사용되는 명령에 대한 도움말을 보려면 477 페이지의 제 30 장 『Nortel Alteon Controller 명령어 참조서』를 참조하십시오.

Nortel Alteon Controller 시스템 설정 전에:

- AIX, HP-UX, Linux 및 Solaris 시스템에서는 루트 사용자에게 하고, Windows 시스템에서는 관리자여야 합니다.
- Nortel Alteon Controller는 가중치를 계산하는 모든 서버 및 Nortel Alteon Web Switch에 IP 연결성을 갖고 있어야 합니다.
- Nortel Alteon Web Switch는 다음과 같이 구성되어야 합니다.
 1. 스위치에서 계층 4 서버 로드 밸런스를 사용 가능하게 하십시오.
 2. IP 인터페이스를 구성하십시오.
 3. SNMP를 사용 가능화하십시오.
 4. 클라이언트 요청을 받은 포트에서 서버 로드 밸런스 클라이언트 처리를 사용 가능화하십시오.
 5. 실제 서버가 연결되어 있는 포트에 대한 서버 로드 밸런스 서버 처리를 사용 가능화하십시오.
 6. 웹 서버 시스템에 대한 실제 서버를 구성하십시오.
 7. Application Server에서 실행 중인 실제 서버로 구성된 실제 서버 그룹을 구성 하십시오.

8. 가상 서버를 구성합니다.
9. 가상 포트에 대한 서비스를 구성하고 서비스를 제공할 실제 서버 그룹에 지정하십시오.

1단계. 서버 기능 시작

`nalserver`가 아직 실행되지 않으면 루트로 `nalserver`를 입력하여 바로 시작하십시오.

주: Windows 시스템의 경우, 시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스를 클릭하십시오. IBM Nortel Alteon Controller를 마우스 오른쪽 단추로 클릭한 다음 시작을 선택하십시오.

2단계. 명령 인터페이스 시작

명령 인터페이스를 시작하려면 `nalcontrol`을 입력하십시오.

3단계. Nortel Alteon Web Switch consultant 정의

스위치 컨설턴트를 추가하려면 다음을 입력하십시오.

```
consultant add switchconsultantID address switchIPAddress
```

4 단계. 스위치 컨설턴트에 서비스 추가

서비스를 추가하려면 다음을 입력하십시오.

```
service add switchConsultantID:serviceID vsid virtualServerID vport  
virtualPortNumber
```

서비스는 VSID(Virtual Server Identifier)와 VPORT(Virtual Port) 번호로 식별되며 둘 다 스위치에 이전에 구성된 가상 서버와 연관되어 있습니다.

5단계. 메트릭 구성

메트릭은 서버 가중치에 사용되는 정보입니다. 각 메트릭에는 다른 메트릭에 상대적인 중요도를 나타내는 비율이 지정됩니다. 연결 데이터 메트릭, 응용프로그램 어드바이저 메트릭 및 Metric Server 메트릭의 조합을 구성할 수 있습니다. 비례의 총계는 항상 100입니다.

서비스를 구성할 때 기본 메트릭은 `activeconn` 및 `connrate`로 정의됩니다. 추가 메트릭을 원하거나 기본값과 모두 다른 메트릭을 원할 경우, 다음을 입력하십시오.

```
service metrics switchConsultantID:serviceID metricName 50  
metricName2 50
```

6단계. 컨설턴트 시작

컨설턴트를 시작하려면 다음을 입력하십시오.

```
consultant start switchConsultantID
```

그러면 메트릭 콜렉터가 시작되고 가중치 계산이 시작됩니다.

7단계. 고가용성 구성(선택)

고가용성을 구성하려면 다음을 입력하십시오.

```
highavailability add address IPaddress partneraddress IPaddress port 80  
role primary
```

제어기 고가용성 구성 및 사용 방법에 대한 자세한 정보는 261 페이지의 제 23 장 『Cisco CSS Controller 및 Nortel Alteon Controller의 고급 기능』을 참조하십시오.

8단계. Metric Server 시작(선택)

5단계에서 시스템 메트릭을 정의할 경우, Metric Server가 서버 시스템에서 시작되어야 합니다. Metric Server 사용에 대한 정보는 272 페이지의 『Metric Server』를 참조하십시오.

9단계. Nortel Alteon Controller 구성 새로 고침

Nortel Alteon Web Switch에서 구성을 수정할 경우, 제어기 구성을 새로 고칠 수 있습니다. 다음을 입력하십시오.

```
service refresh
```

구성을 새로 고치기 전에 컨설턴트를 정지시키십시오. Refresh 명령으로 구성을 갱신한 후 컨설턴트를 재시작하십시오.

구성 검사

구성이 작동되고 있는지 확인하십시오.

1. consultant loglevel을 4로 설정하십시오.
2. 1분 동안 Nortel Alteon Web Switch에서 서버의 연결을 끊으십시오. 또는 1분 동안 Application Server를 종료하십시오.
3. 서버를 다시 연결하거나 Application Server를 재시작하십시오.
4. consultant loglevel을 원하는 레벨(1)로 다시 설정하십시오.
5. 다음 디렉토리에 있는 consultant.log 파일을 보고, **setServerWeights setting service** 를 찾으십시오. 이는 스위치로 가중치를 전송하려는 시도가 이루어졌음을 의미합니다.
 - AIX, HP-UX, Linux 및 Solaris 시스템의 경우: ...ibm/edge/lb/servers/logs/ccol/consultantName
 - Windows 시스템의 경우: ...ibm\edge\lb\servers\logs\cco\consultantName
6. 스위치에 서버 가중치를 표시하고 이 가중치가 제어기 보고서에 표시된 가중치와 일치하는지 확인하십시오.

제 7 부 Load Balancer의 기능 및 고급 기능

이 파트에서는 Load Balancer에 사용 가능한 기능 및 고급 구성 기능에 대한 정보를 제공합니다. 다음 장을 포함합니다.

- 193 페이지의 제 21 장 『Dispatcher, CBR 및 Site Selector에 대한 관리자, 어드바이저 및 Metric Server 기능』
- 215 페이지의 제 22 장 『Dispatcher, CBR 및 Site Selector에 대한 고급 기능』
- 261 페이지의 제 23 장 『Cisco CSS Controller 및 Nortel Alteon Controller의 고급 기능』

제 21 장 Dispatcher, CBR 및 Site Selector에 대한 관리자, 어드바이저 및 Metric Server 기능

이 장에서는 Load Balancer의 로드 밸런스 매개변수를 구성하는 방법과 Load Balancer의 관리자, 어드바이저 및 Metric Server 기능을 설정하는 방법을 설명합니다.

주: 이 장을 읽을 때 Dispatcher 컴포넌트를 사용하고 있지 않는 경우, "dscontrol"을 다음 사항으로 대체하십시오.

- CBR의 경우, **cbrcontrol** 사용
- Site Selector의 경우, **sscontrol** 사용(429 페이지의 제 28 장 『Site Selector 명령어 참조서』 참조)

중요: 해당 제품의 IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, 이 섹션의 내용을 보기 전에 제한사항 및 구성 차이점에 대해 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』 페이지를 참조하십시오.

표 12. Load Balancer의 고급 구성 태스크

태스크	설명	관련 정보
선택적으로 로드 밸런스 설정을 변경합니다.	<p>다음의 로드 밸런스 설정을 변경할 수 있습니다.</p> <ul style="list-style-type: none"> 상태 정보에 제공되는 중요성의 비율 <p>기본 비율은 50-50-0-0입니다. 기본값을 사용할 경우, 어드바이저, Metric Server 및 WLM은 사용되지 않습니다.</p> <ul style="list-style-type: none"> 가중치 관리자 고정 가중치 관리자 간격 감도 임계치 스무스 색인(smoothing index) 	194 페이지의 『Load Balancer에서 제공하는 로드 밸런스 최적화』
관리자에서 서버의 작동 중지 또는 가동을 표시할 때 스크립트를 사용하여 경고 또는 레코드 서버 장애를 생성하십시오.	Load Balancer는 관리자에서 서버의 작동 중지 또는 가동을 표시할 때 사용자 정의할 수 있는 스크립트를 트리거하는 사용자 엑시트를 제공합니다.	198 페이지의 『스크립트를 사용하여 경고나 레코드 서버 장애 생성』
어드바이저 사용	서버의 특정 상태에 대해 보고하는 어드바이저를 설명하고 나열합니다.	199 페이지의 『어드바이저』
HTTP 또는 HTTPS 어드바이저 요청 및 응답(URL) 옵션 사용	시스템에 조회할 서비스 고유의 클라이언트 HTTP URL 문자열 정의	205 페이지의 『응답 및 요청(URL) 옵션을 사용하여 HTTP 또는 HTTPS 어드바이저 구성』
자가 어드바이저 사용	Load Balancer2 티어 WAN 구성에서 백엔드 서버 로드 상태를 제공합니다.	206 페이지의 『상하단부 WAN 구성에서 자가 어드바이저 사용』

표 12. Load Balancer의 고급 구성 TASK (계속)

TASK	설명	관련 정보
사용자 정의 어드바이저 작성	사용자 정의 어드바이저 작성 방법을 설명합니다.	207 페이지의 『사용자 정의(사용자 정의 가능) 어드바이저 작성』
Metric Server 에이전트 사용	Metric Server는 Load Balancer에 시스템 로드 정보를 제공합니다.	211 페이지의 『Metric Server』
작업로드 관리자 어드바이저(WLM) 사용	WLM 어드바이저는 Load Balancer에 시스템 로드 정보를 제공합니다.	214 페이지의 『작업로드 관리자 어드바이저』

Load Balancer에서 제공하는 로드 밸런스 최적화

Load Balancer의 관리자 기능은 다음 설정에 따라 로드 밸런스를 수행합니다.

- 『상태 정보에 제공되는 중요성 비율』
- 195 페이지의 『가중치』
- 197 페이지의 『관리자 간격』
- 201 페이지의 『어드바이저 간격』
- 202 페이지의 『어드바이저 보고서 제한시간』
- 197 페이지의 『감도 임계치』
- 198 페이지의 『스무스 색인』

이 설정을 변경하여 네트워크에 대한 로드 밸런스를 최적화할 수 있습니다.

상태 정보에 제공되는 중요성 비율

관리자에서는 가중치 결정 시 다음 외부 요소를 일부 또는 모두 사용할 수 있습니다.

- **활성 연결:** 각 로드 밸런스 서버 시스템의 활성 연결 수(실행 프로그램에서 추적함). 이 비율은 Site Selector에 적용되지 않습니다.

아니면 —

CPU: 각 로드 밸런스 서버 시스템에서 사용 중인 CPU 백분율(Metric Server 에이전트에서 입력). Site Selector의 경우에만 이 비율이 활성 연결 비율 컬럼 위치에 표시됩니다.

- **새 연결:** 각 로드 밸런스 서버 시스템의 새 연결 수(실행 프로그램에서 추적함). 이 비율은 Site Selector에 적용되지 않습니다.

아니면 —

메모리: 각 로드 밸런스 서버에서 사용 중인 메모리 백분율(Metric Server 에이전트가 입력). Site Selector의 경우에만 이 비율이 새 연결 비율 컬럼 위치에 표시됩니다.

- **특정 포트:** 포트에서 인식 중인 어드바이저가 입력.

- **시스템 메트릭:** Metric Server 또는 WLM과 같은 시스템 모니터링 도구로 입력.

관리자에서는 각 서버의 현재 가중치 및 계산에 필요한 다른 일부의 정보와 함께 실행 프로그램에서 처음 두 값(활성 연결 및 새 연결)을 확보합니다. 이들 값은 실행 프로그램에서 내부적으로 생성되어 저장되는 정보에 따라 달라집니다.

주: Site Selector의 경우, 관리자가 Metric Server에서 처음 두 개의 값(CPU 및 메모리)을 확보합니다.

네 값의 상대적인 중요성 비율을 클러스터(또는 사이트 이름)를 기반으로 변경할 수 있습니다. 비율은 백분율로 나타내므로 상대 비율의 합은 100%여야 합니다. 기본 비율은 50/50/0/0이며, 이 값은 어드바이저와 시스템 정보를 무시합니다. 사용자 환경에서 최상의 성능을 제공하는 조합을 찾기 위해 다른 비율을 시도할 필요가 있습니다.

주: 어드바이저(WLM 이외)를 추가할 때, 포트 비율이 0이면 관리자에서 이 값을 1로 늘립니다. 상대 비율의 합이 100이어야 하므로 최고값이 1만큼 줄어듭니다.

WLM 어드바이저를 추가할 때, 시스템 메트릭 비율이 0이면 관리자에서 이 값을 1로 늘립니다. 상대 비율의 합이 100이어야 하므로 최고값이 1만큼 줄어듭니다.

활성 연결 수는 로드 밸런스 서버 시스템에서 제공되고 있는 서비스를 사용하는 데 필요한 시간과 클라이언트 수에 따라 달라집니다. 클라이언트 연결이 빠르면(예: HTTP GET을 사용하여 제공되는 작은 웹 페이지), 활성 연결 수는 상당히 적어집니다. 클라이언트 연결이 느려지면(예: 데이터베이스 조회), 활성 연결 수는 많아집니다.

활성 및 새 연결 비율을 너무 작게 설정하지 마십시오. 처음 두 값을 각각 최소 20으로 설정하지 않으면 로드 밸런스 및 스무스 색인은 사용 불가능합니다.

중요성 비율의 값을 설정하려면 **dscontrol cluster set cluster proportions** 명령을 사용하십시오. 379 페이지의 『dscontrol cluster — 클러스터 구성』에서 자세한 정보를 참조하십시오.

가중치

가중치는 실행 프로그램의 내부 카운터, 어드바이저의 피드백 및 Metric Server와 같은 시스템 모니터링 프로그램의 피드백에 따라 관리자 기능에서 설정됩니다. 관리자를 실행 중에 수동으로 가중치를 설정하려면 dscontrol 서버 명령에서 **fixedweight** 옵션을 지정하십시오. **fixedweight** 옵션에 대한 설명은 196 페이지의 『관리자 고정 가중치』를 참조하십시오.

가중치는 포트의 모든 서버에 적용됩니다. 특정 포트의 경우, 요청은 각각 상대적인 가중치에 따라 서버 간에 분산됩니다. 예를 들어, 하나의 서버가 가중치 10으로 설정되고 다른 서버가 가중치 5로 설정되면 10으로 설정된 서버의 요청수는 5로 설정된 서버 요청수의 두 배가 됩니다.

서버가 가질 수 있는 최대 가중치 경계를 지정하려면, **dscontrol port set port weightbound weight** 명령을 사용하십시오. 이 명령은 각 서버가 확보할 요청수 사이에 있을 수 있는 차이에 영향을 줍니다. 최대 weightbound를 1로 설정할 경우, 모든 서버의 가중치는 1, 작업 정지된 경우에는 0, 단절 표시인 경우에는 -1입니다. 이 숫자가 증가할수록 서버의 가중치 차이도 증가합니다. 최대 weightbound가 2이면 한 서버가 다른 서버 요청 수의 두 배를 확보할 수 있습니다. 최대 weightbound가 10이면, 한 서버가 다른 서버 요청 수의 10배를 확보할 수 있습니다. 기본 최대 weightbound는 20입니다.

어드바이저가 서버 종료로 발견하면, 이를 관리자에 알려 서버의 가중치를 0으로 설정합니다. 결과적으로, 실행 프로그램에서 해당 가중치가 0으로 남아 있는 한, 해당 서버에 대한 어떠한 추가 연결도 전송하지 않습니다. 가중치가 변경되기 전에 해당 서버에 대한 활성 연결이 있었으면, 이 연결은 일반적으로 완료된 채로 남아 있습니다.

서버가 모두 단절되면 관리자가 가중치 바운드의 반으로 가중치를 설정합니다.

관리자 고정 가중치

관리자가 없으면 어드바이저를 실행할 수 없으며 서버가 작동 중단되었는지 검색할 수도 없습니다. 어드바이저를 실행하기로 했으나 관리자가 특정 서버에 대해 사용자가 설정한 가중치를 갱신하지 않기 바라면 **dscontrol server** 명령에서 **fixedweight** 옵션을 사용하십시오. 예를 들어,

```
dscontrol server set cluster:port:server fixedweight yes
```

고정 가중치가 yes로 설정되면 **dscontrol server set weight** 명령을 사용하여 원하는 값으로 가중치를 설정하십시오. 고정 가중치가 no로 설정된 다른 dscontrol 서버 명령을 발행할 때까지는 관리자가 실행 중인 동안 서버 가중치 값이 고정됩니다. 자세한 내용은 418 페이지의 『dscontrol server — 서버 구성』을 참조하십시오.

다운된 서버로 TCP 재설정 전송(Dispatcher 컴포넌트 전용)

TCP 재설정이 활성화되면, 클라이언트가 가중치가 0인 서버에 연결할 경우 Dispatcher가 TCP 재설정을 클라이언트로 전송합니다. 0으로 구성되거나 어드바이저가 다운된 서버로 표시한 경우 서버의 가중치는 0일 수도 있습니다. TCP 재설정은 연결을 즉시 닫습니다. 이 기능은 클라이언트가 실패한 연결을 재협상하도록 촉진하므로 장기간 유지된 연결에 유용합니다. TCP 재설정을 활성화하려면, **dscontrol port addlset port reset yes** 명령을 사용하십시오. 재설정의 기본값은 no입니다.

주: TCP 재설정은 모든 Dispatcher의 전달 메소드에 적용됩니다. 그러나 TCP 재설정 기능을 사용하려면, **dscontrol executor** 명령에서 **clientgateway**를 라우터 주소로 설정해야 합니다.

TCP 재설정과 결합하여 구성하기에 유용한 기능은 어드바이저 재시도입니다. 이 기능을 사용할 경우, 어드바이저는 서버를 다운된 서버로 표시하기 전에 연결을 재시도할 수

있습니다. 이는 어드바이저가 서버를 영구적으로 다운된 서버로 표시하여 연결 재설정 문제점이 발생하는 것을 방지하는 데 도움이 됩니다. 즉, 어드바이저가 첫 번째 시도에서 실패했다고 해서 반드시 기존 연결이 또한 실패한다는 것을 의미하지는 않습니다. 203 페이지의 『어드바이저 재시도』에서 자세한 정보를 참조하십시오.

관리자 간격

전체 성능을 최적화하기 위해 관리자가 실행 프로그램과 상호작용할 수 있는 횟수가 제한됩니다. 이 간격은 **dscontrol manager interval** 및 **dscontrol manager refresh** 명령을 입력하여 변경할 수 있습니다.

관리자 간격은 실행 프로그램이 연결 경로 지정에서 사용하는 서버 가중치를 관리자가 갱신하는 횟수를 지정합니다. 관리자 간격이 너무 작으면, 관리자에서 일정하게 실행 프로그램을 인터럽트한 결과로서 성능이 저하될 수 있다는 것을 나타냅니다. 관리자 간격이 너무 크면, 실행 프로그램의 요청 경로 지정이 정확한 최신 정보를 기초로 하지 않는다는 것을 나타냅니다.

예를 들어, 관리자 간격을 1초로 설정하려면 다음 명령을 입력하십시오.

```
dscontrol manager interval 1
```

관리자 갱신 주기는 관리자에서 실행 프로그램에 상태 정보를 묻는 횟수를 지정합니다. 갱신 주기는 간격(시간)을 기초로 합니다.

예를 들어, 관리자 갱신 주기를 3으로 설정하려면 다음 명령을 입력하십시오.

```
dscontrol manager refresh 3
```

이렇게 하면 관리자에서는 실행 프로그램에 상태를 묻기 전에 세 개의 간격 동안 기다립니다.

감도 임계치

사용자 서버의 로드 밸런스를 최적화하는 다른 방법을 제공합니다. 최고 속도로 작동하도록, 서버의 가중치가 현저하게 변경된 경우에만 가중치를 갱신합니다. 서버 상태가 약간 변경되었거나 변경되지 않은 경우 가중치를 일정하게 갱신하면 불필요한 오버헤드가 작성됩니다. 한 포트상의 모든 서버에 대한 총 가중치의 백분을 가중치 변경값이 감도 임계치보다 큰 경우, 관리자에서는 연결을 분산시키기 위해 실행 프로그램에서 사용하는 가중치를 갱신합니다. 예를 들어, 총 가중치가 100에서 105로 변경된다고 가정합니다. 변경값은 5%입니다. 기본 감도 임계치 5를 사용하는 경우, 백분을 변경값이 임계치를 넘지 않으므로 관리자는 실행 프로그램에서 사용하는 가중치를 갱신하지 않습니다. 그러나 총 가중치가 100에서 106으로 변경되면, 관리자에서는 이 가중치를 갱신합니다. 관리자의 감도 임계치를 기본값(예: 6)이 아닌 다른 값으로 설정하려면, 다음 명령을 입력하십시오.

```
dscontrol manager sensitivity 6
```

대부분의 경우, 이 값을 변경할 필요가 없습니다.

스무스 색인

관리자에서는 서버 가중치를 동적으로 계산합니다. 결과적으로, 갱신된 가중치는 이전의 가중치와 많이 다를 수 있습니다. 대부분의 환경에서, 이러한 점은 문제가 되지 않습니다. 그러나, 경우에 따라 이로 인해 요청이 로드 밸런스되는 방법에 변동(oscillating) 효과가 발생할 수 있습니다. 예를 들면, 한 서버가 높은 가중치로 인해 대부분의 요청 수신을 종료할 수 있습니다. 관리자에서는 서버에 활성 연결 수가 큰지 서버가 느리게 응답하는지 살펴봅니다. 그러면 사용 가능한 서버로 가중치를 이동하게 되며 같은 효과가 너무 자주 발생하여 자원이 비효율적으로 사용됩니다.

이러한 문제점을 줄이기 위해, 관리자에서는 스무스 색인을 사용합니다. 스무스 색인은 서버의 가중치가 변경될 수 있는 정도를 제한하여, 요청 분산 시 변경을 효율적으로 평탄화합니다. 스무스 색인이 높으면 서버 가중치가 상대적으로 낮게 변경됩니다. 지수가 낮으면 서버 가중치가 높게 변경됩니다. 스무스 색인의 기본값은 1.5입니다. 1.5에서 서버 가중치는 보다 동적일 수 있습니다. 지수가 4나 5가 되면, 가중치는 더 안정됩니다. 예를 들어, 스무스 색인을 4로 설정하려면, 다음 명령을 입력하십시오.

```
dscontrol manager smoothing 4
```

대부분의 경우, 이 값을 변경할 필요가 없습니다.

스크립트를 사용하여 경보나 레코드 서버 장애 생성

[[는 사용자 정의할 수 있는 스크립트를 트리거하는 사용자 엑시트를 제공합니다. 스크립트를 작성하여 관리자에서 서버의 단절을 표시할 때나 서버가 장애 이벤트를 기록할 때, 관리자에게 경보를 보내는 것과 같은 자동 조치를 수행할 수 있습니다. 사용자 정의할 수 있는 예제 스크립트는 `...ibm/edge/lb/servers/samples` 설치 디렉토리에 있습니다. 파일을 실행하려면 예제 스크립트를 `...ibm/edge/lb/servers/bin` 디렉토리로 이동하고 "sample" 파일 확장자를 제거하십시오. 다음의 예제 스크립트가 제공됩니다.

- **serverDown** — 관리자에서 서버를 단절 표시합니다.
- **serverUp** — 관리자에서 서버가 가동됨을 표시합니다.
- **managerAlert** — 특정 포트에 대해 모든 서버를 단절 표시합니다.
- **managerClear** — 특정 포트에 대해 모든 서버가 단절 표시된 후, 이제 최소한 하나의 서버가 가동됩니다.

클러스터의 서버가 모두 단절된 것으로 (사용자 또는 어드바이저에 의해) 표시되는 경우, `managerAlert`(구성되어 있는 경우)가 시작되고 Load Balancer가 라운드 로빈 방식을 사용하여 통신량을 서버로 라우트합니다. `serverDown` 스크립트는 클러스터의 마지막 서버가 오프라인으로 감지되면 시작하지 않습니다.

서버가 다시 온라인 상태로 돌아오고 요청에 응답하는 경우 Load Balancer는 계획적으로 통신량을 계속 라우트하려고 합니다. Load Balancer가 대신 통신량을 모두 삭제하면 클라이언트는 응답을 수신하지 못합니다.

Load Balancer가 클러스터의 첫 번째 서버가 다시 온라인 상태임을 감지하면, managerClear 스크립트(구성되어 있는 경우)가 시작되지만 serverUp 스크립트(구성되어 있는 경우)는 추가 서버가 온라인 상태로 돌아올 때까지 실행되지 않습니다.

serverUp 및 serverDown 스크립트를 사용할 경우 고려사항:

- 관리자 주기를 어드바이저 시간의 25% 미만으로 정의한 경우 서버 시작 또는 서버 중지의 오류 보고서가 생길 수 있습니다. 기본적으로 관리자는 2초마다 실행되지만, 어드바이저는 7초마다 실행됩니다. 따라서 관리자는 4주기 내에 새 어드바이저 정보가 필요합니다. 그러나 이 제한사항(즉, 관리자 주기를 어드바이저 시간의 25%보다 큰 값으로 정의)을 제거하면 여러 어드바이저가 단일 서버에서 권고할 수 있기 때문에 성능이 크게 떨어집니다.
- 서버가 중지되면 serverDown 스크립트가 시작됩니다. 그러나 serverUp 명령을 실행할 경우 관리자가 어드바이저 주기에서 새 정보를 가져올 때까지 서버가 시작 중인 것으로 가정합니다. 서버가 계속 중지 상태이면 serverDown 스크립트가 다시 실행됩니다.

어드바이저

어드바이저는 Load Balancer 내의 에이전트입니다. 서버 시스템의 상태와 로딩을 평가하는 기능을 합니다. 어드바이저는 서버와의 활성 클라이언트와 같은 교환으로 이를 수행합니다. 어드바이저는 Application Server의 경량 클라이언트로서 간주될 수 있습니다.

Dispatcher는 가장 널리 쓰이는 프로토콜용으로 몇 가지 프로토콜 고유의 어드바이저를 제공합니다. 그러나 Load Balancer의 모든 컴포넌트와 함께 제공된 모든 어드바이저를 사용할 수 없습니다. (예를 들어, Telnet 어드바이저를 CBR 컴포넌트와 함께 사용하면 안 됩니다.) 또한 []는 사용자가 고유의 어드바이저를 작성할 수 있는 『사용자 정의 어드바이저』의 개념을 지원합니다.

바인드 특정 서버 응용프로그램 사용의 제한사항: 바인드 특정 서버의 어드바이저를 사용하려면 서버에 있는 두 개의 인스턴스를 시작하십시오. 인스턴스 하나는 클러스터:포트를 바인드하는데, 나머지 인스턴스는 서버:포트를 바인드하는데 사용합니다. 서버가 바인드 특정인지 판별하려면 netstat -an 명령을 실행하여 서버:포트를 확인하십시오. 서버가 바인드 특정이 아니면 해당 명령의 결과는 0.0.0.0:80입니다. 바인드 특정이면 192.168.15.103:80 같은 주소가 표시됩니다.

HP-UX 및 Solaris 시스템의 경우, 바인드 고유 서버 응용프로그램 사용에 대한 제한 사항: ifconfig alias 명령 대신에 arp publish를 사용할 경우, 바인드 고유 서버 응용

프로그램(CBR 또는 Site Selector 같은 Load Balancer 컴포넌트 포함)이 클러스터 IP 주소에 바인딩되면 서버를 바인드 고유 서버 응용프로그램과 로드 밸런싱할 때 Load Balancer에서 어드바이저 사용을 지원하지 않습니다. 그러나 바인드에 고유한 서버 응용프로그램에 대해 어드바이저 사용 시, 서버 응용프로그램과 동일한 시스템에 []를 결합 배치하지 마십시오.

주: 여러 개의 네트워크 어댑터 카드가 설치된 컴퓨터에서 []를 실행 중이며 어드바이저 통신량이 특정 어댑터를 통해 플로우되길 원할 경우, 패킷의 소스 IP 주소를 특정 주소로 강제 실행할 수 있습니다. 어드바이저 패킷 소스 주소를 특정 주소로 강제 실행하려면 다음을 해당 Load Balancer 시작 스크립트 파일(dsserver, cbrserver 또는 sssserver)의 `java...SRV_XXXConfigServer...` 행에 추가하십시오.

```
-DLB_ADV_SRC_ADDR=IP_address
```

어드바이저 작동 방법

어드바이저는 정기적으로 각 서버와의 TCP 연결을 열어 서버에 요청 메시지를 전송합니다. 메시지 콘텐츠는 서버에서 실행 중인 프로토콜에 따라 고유합니다. 예를 들어, HTTP 어드바이저는 서버에 HTTP 『HEAD』 요청을 전송합니다.

그런 후 어드바이저는 서버의 응답을 인식합니다. 응답을 받은 후, 어드바이저는 서버를 평가합니다. 이 『로드』값을 계산하기 위해, 대부분의 어드바이저는 응답할 서버의 시간을 측정한 후 이 값(밀리초 단위)을 로드로 사용합니다.

그러면 어드바이저는 『포트』 컬럼에서 관리자 보고서에 표시되는 로드값을 Dispatcher의 관리자 기능에 보고합니다. 관리자는 비율당 모든 소스의 집계 가중치를 계산하여 이 가중치를 실행 프로그램 기능으로 설정합니다. 그러면 실행 프로그램에서는 새로운 수신 클라이언트 연결 로드 밸런스에 이들 가중치를 사용합니다.

어드바이저는 서버가 적당하게 작동된다고 판별한 경우, 관리자에 0이 아닌 양수의 로드 숫자를 보고합니다. 어드바이저가 서버 작동이 되지 않는다고 판별한 경우, 특수 로드값인 음수 1(-1)을 리턴합니다. 관리자 및 실행 프로그램은 서버가 다시 작동할 때까지 해당 서버로 연결을 전달하지 않습니다.

주: 초기 요청 메시지를 전송하기 전에 어드바이저는 서버를 핑합니다. 빠른 상태를 제공하여 시스템이 온라인인지 판별하도록 합니다. 서버가 해당 핑에 응답한 후 더 이상의 핑은 전송되지 않습니다. 핑 사용을 불가능하게 하려면

```
-DLB_ADV_NB_PING을 Load Balancer 시작 스크립트 파일에 추가하십시오.
```

어드바이저 시작 및 정지

모든 클러스터(그룹 어드바이저)에서 특정 포트에 대해 어드바이저를 시작할 수 있습니다. 또는 동일한 포트이지만 다른 클러스터에서 다른 어드바이저를 실행하도록 선택할

수 있습니다(클러스터/사이트 고유의 어드바이저). 예를 들어, 세 개의 클러스터(*clusterA*, *clusterB*, *clusterC*)로 정의된 Load Balancer가 있는 경우, 포트 80이 있는 각 클러스터에서 다음 사항을 수행할 수 있습니다.

- 클러스터/사이트 고유의 어드바이저: *clusterA*의 포트 80에서 어드바이저를 시작하려면 클러스터 및 포트를 모두 지정하십시오.

```
dscontrol advisor start http clusterA:80
```

이 명령은 *clusterA*의 포트 80에서 HTTP 어드바이저를 시작합니다. HTTP 어드바이저는 *clusterA*의 포트 80에 첨부된 모든 서버에서 권고합니다.

- 그룹 어드바이저: 기타 모든 클러스터의 포트 80에서 사용자 정의 어드바이저를 시작하려면 포트를 지정하십시오.

```
dscontrol advisor start ADV_custom 80
```

이 명령은 *clusterB* 및 *clusterC*의 포트 80에서 *ADV_custom* 어드바이저를 시작합니다. 사용자 정의 어드바이저는 *clusterB* 및 *clusterC*의 포트 80에 첨부된 모든 서버에서 권고합니다(사용자 정의 어드바이저에 대한 자세한 정보는 207 페이지의 『사용자 정의(사용자 정의 가능) 어드바이저 작성』을 참조하십시오).

주: 그룹 어드바이저는 현재 클러스터/사이트 고유의 어드바이저가 없는 모든 클러스터/사이트에서 권고합니다.

그룹 어드바이저에 대한 이전의 구성 예제를 사용하여 두 클러스터(*clusterB* 및 *clusterC*) 모두 또는 임의 클러스터의 포트 80에서 *ADV_custom* 사용자 정의 어드바이저를 정지하도록 선택할 수 있습니다.

- *clusterB*의 포트 80에서 사용자 정의 어드바이저를 정지하려면 클러스터 및 포트를 지정하십시오.

```
dscontrol advisor stop ADV_custom clusterB:80
```

- *clusterB* 및 *clusterC*의 포트 80에서 사용자 정의 어드바이저를 정지하려면 포트만 지정하십시오.

```
dscontrol advisor stop ADV_custom 80
```

어드바이저 간격

주: 어드바이저 기본값은 가능한 많은 시나리오에 효율적으로 적용됩니다. 기본값이 아닌 다른 값을 입력할 경우에는 주의하십시오.

어드바이저 간격은 어드바이저가 모니터링하는 포트의 서버로부터 상태를 묻는 횟수를 설정하고, 관리자에 결과를 보고합니다. 어드바이저 간격이 너무 작으면, 어드바이저가 일정하게 서버를 인터럽트한 결과로 성능이 저하될 수 있다는 것을 나타냅니다. 어드바이저 간격이 너무 크면, 관리자의 가중치에 대한 결정이 정확한 최신 정보를 기초로 하지 않는다는 것을 나타냅니다.

예를 들어, 포트 80에 대한 HTTP 어드바이저의 간격을 3초로 설정하려면, 다음 명령을 입력하십시오.

```
dscontrol advisor interval http 80 3
```

관리자 간격보다 적은 어드바이저 간격을 지정할 수 없습니다. 기본 어드바이저 간격은 7초입니다.

어드바이저 보고서 제한시간

로드 밸런스 결정 시 관리자에서 이전 정보를 사용하지 않도록 하기 위해, 시간 소인이 어드바이저 보고서 제한시간에 설정된 시간보다 이전인 어드바이저 정보를 사용하지 않습니다. 어드바이저 보고서 제한시간은 어드바이저 폴링 간격보다 커야 합니다. 제한시간이 더 적으면, 관리자에서는 논리적으로 사용해야 하는 보고서를 무시합니다. 기본적으로 어드바이저 보고서는 종료 시간이 없습니다. 기본값은 제한 없음입니다.

예를 들어, 포트 80에 대한 HTTP 어드바이저의 어드바이저 보고서 제한시간을 30초로 설정하려면, 다음 명령을 입력하십시오.

```
dscontrol advisor timeout http 80 30
```

어드바이저 보고서 제한시간 설정에 대한 자세한 정보는 372 페이지의 『dscontrol advisor — 어드바이저 제어』를 참조하십시오.

어드바이저 연결 제한시간 및 서버의 수신 제한시간

Load Balancer의 경우, 어드바이저가 서버(서비스)의 특정 포트 실패를 발견하는 어드바이저 제한시간 값을 설정할 수 있습니다. 실패한 서버 제한시간 값(connecttimeout 및 receivetimeout)은 어드바이저가 연결이나 수신에 실패했음을 보고하기 전에 대기하는 시간을 결정합니다.

가장 빨리 실패한 서버를 발견하려면 어드바이저 연결 및 수신 제한시간을 최소값(1초)으로 설정하고 어드바이저 및 관리자 간격을 최소값(1초)으로 설정하십시오.

주: 사용자의 환경에서 통신량이 조금 많아서 서버 응답 시간이 늦어지는 경우, connecttimeout 및 receivetimeout 값을 너무 작게 설정하지 마십시오. 그렇지 않으면, 어드바이저가 사용 중인 서버를 실패로 표시할 수 있습니다.

예를 들어, 포트 80에 있는 HTTP 어드바이저에 대해 connecttimeout 및 receivetimeout을 9초로 설정하려면, 다음 명령을 입력하십시오.

```
dscontrol advisor connecttimeout http 80 9
dscontrol advisor receivetimeout http 80 9
```

연결 및 수신 제한시간의 기본값은 어드바이저 간격에 지정된 값의 세 배입니다.

어드바이저 재시도

어드바이저는 서버를 다운된 서버로 표시하기 전에 연결을 재시도할 수 있습니다. 어드바이저는 서버 조회가 재시도 횟수 더하기 1회만큼 실패할 때까지 서버를 다운된 서버로 표시하지 않습니다. **retry** 값은 3을 초과하면 안 됩니다. 다음 명령은 포트 389에서 LDAP 어드바이저에 대한 재시도 값을 2로 설정합니다.

```
dscontrol advisor retry ldap 389 2
```

어드바이저 목록

- **HTTP** 어드바이저는 연결을 열어 기본적으로 HEAD 요청을 전송하고 응답 연결을 기다린 후 로드로서 경과 시간을 리턴합니다. HTTP 어드바이저가 전송한 요청의 유형을 변경하는 방법에 대한 자세한 정보는 205 페이지의 『응답 및 요청(URL) 옵션을 사용하여 HTTP 또는 HTTPS 어드바이저 구성』을 참조하십시오.
- **HTTPS** 어드바이저는 SSL 접속에 대한 "heavyweight" 어드바이저입니다. 서버와의 완전한 SSL 소켓 접속을 수행합니다. HTTPS 어드바이저는 SSL 연결을 열어 HTTPS 요청을 전송하고 응답을 기다리며 연결을 닫은 후 로드로서 경과 시간을 리턴합니다. (SSL 접속에 대한 "lightweight" 어드바이저인 SSL 어드바이저도 참조하십시오.)

주: HTTPS 어드바이저는 서버 키 또는 인증서 콘텐츠에 종속되지 않지만, 만기되지 않아야 합니다.

- **SIP** 어드바이저는 연결을 열어 OPTIONS 요청을 전송하고 응답을 기다리며 연결을 닫은 후 로드로서 경과 시간을 리턴합니다. 지원되는 SIP 어드바이저는 TCP에서만 실행되며 OPTIONS 요청에 응답하는 서버에 설치될 응용프로그램이 필요합니다.
- **FTP** 어드바이저는 연결을 열어 SYST 요청을 전송하고 응답을 기다리며 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **LDAP** 어드바이저는 연결을 열어, 익명 BIND 요청을 전송하고, 응답을 기다리며, 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **Telnet** 어드바이저는 연결을 열고 서버의 초기 메시지를 기다리며 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **NNTP** 어드바이저는 연결을 열어 서버의 초기 메시지를 기다리며 중지 명령을 전송하고 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **IMAP** 어드바이저는 연결을 열어 서버의 초기 메시지를 기다리며 중지 명령을 전송하고 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **POP3** 어드바이저는 연결을 열어 서버의 초기 메시지를 기다리며 중지 명령을 전송하고 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **SMTP** 어드바이저는 연결을 열어 서버의 초기 메시지를 기다리며 중지 명령을 전송하고 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.

- **SSL** 어드바이저는 SSL 접속에 대한 "lightweight" 어드바이저입니다. 서버와의 전체 SSL 소켓 접속을 구축하지 않습니다. SSL 어드바이저는 연결을 열어 SSL CLIENT_HELLO 요청을 전송하고 응답을 기다리며 연결을 닫은 후 로드로서 경과 시간을 리턴합니다. (SSL 접속에 대한 "heavyweight" 어드바이저인 HTTPS 어드바이저도 참조하십시오.)

주: SSL 어드바이저는 키 관리 또는 인증에 종속되지 않습니다.

- **ssl2http** 어드바이저가 시작되어 포트 443에 나열된 서버에서 권고하지만, HTTP 요청에 대해 "mapport"에 소켓을 엽니다. 클라이언트 대 프록시 프로토콜이 SSL이고 프로시 대 서버 프로토콜이 HTTP인 경우, CBR용 ssl2http 어드바이저만 사용 가능합니다. 자세한 정보는 120 페이지의 『SSL의 클라이언트-투-프록시 및 HTTP의 프록시-투-서버의 로드 밸런스』를 참조하십시오.
- **Caching Proxy(cachingproxy)** 어드바이저는 연결을 열어 Caching Proxy 고유의 HTTP GET 요청을 전송한 후 Caching Proxy 로드로서 응답을 해석합니다.

주: Caching Proxy 어드바이저를 사용할 때 Caching Proxy는 로드 밸런스 중인 모든 서버에서 실행되어야 합니다. Load Balancer가 로드 밸런스 중인 동일한 시스템에 배열되어 있지 않으면 상주하는 시스템에 Caching Proxy를 설치해야 할 필요는 없습니다.

- **DNS** 어드바이저는 연결을 열어 DNS에 대한 포인터 조회를 전송하고 응답을 기다리며 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **연결** 어드바이저는 프로토콜 고유의 데이터를 서버와 교환하지 않습니다. 단순히 서버와의 TCP 연결을 열고 닫는 데 걸리는 시간만 측정합니다. 이 어드바이저는 TCP를 사용하지만 IBM 제공 또는 사용자 정의 어드바이저를 사용할 수 없는 고급 프로토콜이 있는 서버 응용프로그램에 유용합니다.
- **ping** 어드바이저는 서버와의 TCP 연결을 열지 않지만 대신, 서버가 ping에 응답하는지 여부를 보고합니다. ping 어드바이저는 모든 포트에서 사용될 수 있지만, 여러 프로토콜 통신량이 이동할 수 있는 와일드 카드 포트를 사용하는 구성을 위해서도 설계되었습니다. 또한 UDP와 같이 그 서버에 비 TCP 프로토콜을 사용하는 구성에서도 유용합니다.
- **도달** 어드바이저는 대상 시스템을 ping합니다. 또한 이 어드바이저는 그 도달 목표의 도달 가능성을 판별하기 위해 Dispatcher의 고가용성 컴포넌트에 맞게 설계되었습니다. 해당 결과는 고가용성 컴포넌트로 이동하며 관리자 보고서에는 표시되지 않습니다. 다른 어드바이저와 달리 도달 어드바이저는 Dispatcher 컴포넌트의 관리자 기능에 의해 자동으로 시작됩니다.
- **DB2** 어드바이저는 DB2 서버와 함께 작동됩니다. Dispatcher에는 고객이 고유의 사용자 정의 어드바이저를 기록하지 않고도 DB2 서버의 상태를 확인할 수 있는 내장 기능이 있습니다. DB2 어드바이저는 DB2 연결 포트와만 통신하고, Java™ 연결 포트와는 통신하지 않습니다.

- 자가 어드바이저는 백엔드 서버에서 로드 상태 정보를 수집합니다. Dispatcher가 자가 어드바이저에서 상단부 Load Balancer로 정보를 공급하는 상하단부 구성의 Dispatcher를 사용할 때 자가 어드바이저를 사용할 수 있습니다. 자가 어드바이저는 특히 실행 프로그램 레벨에서 Dispatcher의 백엔드 서버에서 초당 연결 수를 측정합니다. 206 페이지의 『상하단부 WAN 구성에서 자가 어드바이저 사용』에서 자세한 정보를 참조하십시오.
- **WLM**(작업로드 관리자) 어드바이저는 MVS™ Workload Manager(WLM) 컴포넌트를 실행하는 OS/390 메인프레임의 서버와 함께 작동하도록 설계되었습니다. 자세한 내용은 214 페이지의 『작업로드 관리자 어드바이저』를 참조하십시오.
- Dispatcher는 고객이 조정(사용자 정의 가능) 어드바이저를 작성할 수 있는 기능을 제공합니다. 그러면 IBM이 특정 어드바이저를 개발하지 않은 고유 프로토콜(TCP의 맨 위에 있는)의 지원이 가능합니다. 자세한 정보는 207 페이지의 『사용자 정의(사용자 정의 가능) 어드바이저 작성』을 참조하십시오.
- **WAS**(WebSphere Application Server) 어드바이저는 WebSphere Application Server와 함께 작동됩니다. 이 어드바이저에 대한 사용자 정의 가능 예제 파일은 설치 디렉토리에 제공됩니다. 자세한 정보는 208 페이지의 『WAS 어드바이저』를 참조하십시오.

응답 및 요청(URL) 옵션을 사용하여 HTTP 또는 HTTPS 어드바이저 구성

HTTP 또는 HTTPS 어드바이저의 URL 옵션은 Dispatcher 및 CBR 컴포넌트에 대해 사용 가능합니다.

HTTP 또는 HTTPS 어드바이저를 시작한 후, 서버에 조회할 서비스 고유의 클라이언트 HTTP URL 문자열을 정의할 수 있습니다. 이 기능으로 어드바이저는 서버 내의 개별 서비스의 상태를 평가할 수 있습니다. 동일한 물리적 IP 주소를 가진 고유의 서버 이름으로 논리 서버를 정의하여 이 기능을 수행할 수 있습니다. 64 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』에서 자세한 정보를 참조하십시오.

HTTP 포트 아래 정의된 각 논리 서버에 대해, 서버에 조회할 서비스 고유의 클라이언트 HTTP URL 문자열을 지정할 수 있습니다. HTTP 또는 HTTPS 어드바이저는 **advisorrequest** 문자열을 사용하여 서버의 상태를 조회합니다. 기본값은 HEAD / HTTP/1.0입니다. **advisorresponse** 문자열은 어드바이저가 HTTP 응답에서 스캔하는 응답입니다. 어드바이저는 **advisorresponse** 문자열을 사용하여 서버에서 수신한 실제 응답과 비교합니다. 기본값은 null입니다.

중요: HTTP URL 문자열에 공백이 포함된 경우에는 다음에 주의하십시오.

- **dscontrol>>** 셸 프롬프트에서 명령을 발행할 경우, 공백이 문자열에 포함되어 있으면 문자열을 따옴표(')로 묶어야 합니다. 예를 들어,

```
server set cluster:port:server advisorrequest "head / http/1.0"
server set cluster:port:server advisorresponse "HTTP 200 OK"
```

- 운영 체제 프롬프트에서 **dscontrol** 명령을 발행할 경우, 텍스트 앞에 "\"를, 텍스트 뒤에 \"를 표시해야 합니다. 예를 들어,

```
dscontrol server set cluster:port:server
advisorrequest "\"head / http/1.0\""
```

```
dscontrol server set cluster:port:server advisorresponse "\"HTTP 200 OK\""
```

백엔드 서버가 기능하는지 알아 보기 위해 HTTP 또는 HTTPS 어드바이저가 백엔드 서버에 보내는 요청을 작성할 경우, 사용자가 HTTP 요청의 시작을 입력하면 Load Balancer가 그 뒤로 요청의 끝을 완료합니다.

```
\r\nAccept:
*/*\r\nUser-Agent:IBM_Network_Dispatcher_HTTP_Advisor\r\n\r\n
```

Load Balancer가 요청의 끝에 이 문자열을 추가하기 전에 다른 HTTP 헤더 필드를 추가하려는 경우, 사용자의 \r\n 문자열을 요청에 포함하여 그렇게 할 수 있습니다. 다음은 요청에 HTTP 호스트 헤더 필드를 추가하기 위해 입력할 수 있는 예입니다.

```
GET /pub/WWW/TheProject.html HTTP/1.0 \r\nHost: www.w3.org
```

주: 지정된 HTTP 포트 번호에 대해 HTTP 또는 HTTPS 어드바이저를 시작한 후, 어드바이저 요청 및 응답 값은 해당 HTTP 포트 번호에 대해 사용 가능해집니다.

418 페이지의 『dscontrol server — 서버 구성』에서 자세한 정보를 참조하십시오.

상하단부 WAN 구성에서 자가 어드바이저 사용

자가 어드바이저는 Dispatcher 컴포넌트에서 사용 가능합니다.

상하단부 WAN(Wide Area Network) 구성의 Load Balancer의 경우, Dispatcher가 백엔드 서버에서 로드 상태 정보를 수집하는 자가 어드바이저를 제공합니다.

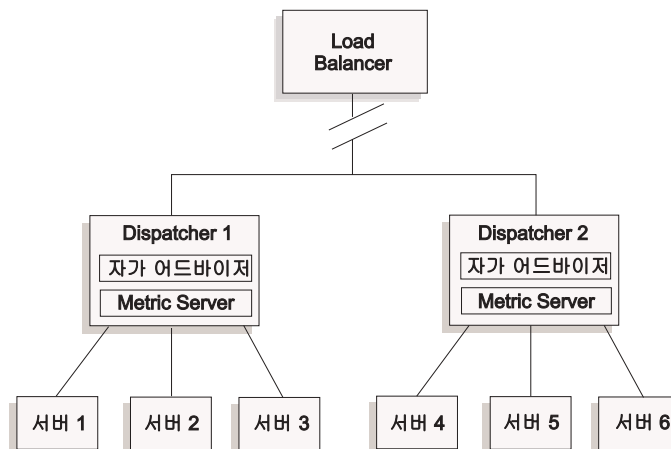


그림 34. 자가 어드바이저를 사용하는 상하단부 WAN 구성의 예제

이 예제에서 자가 어드바이저는 Metric Server와 함께 상단부 Load Balancer가 로드 밸러스 중인 두 대의 Dispatcher 시스템에 있습니다. 자가 어드바이저는 특히 실행 프로그램 레벨에서 Dispatcher의 백엔드 서버에서 초당 연결 수를 측정합니다.

자가 어드바이저는 결과를 dsloadstat 파일에 기록합니다. 또한 Load Balancer는 dsload라는 외부 메트릭을 제공합니다. 각 Dispatcher 시스템의 Metric Server 에이전트는 dsload라는 외부 메트릭을 호출하는 구성을 실행합니다. dsload 스크립트는 dsloadstat 파일에서 문자열을 추출하여 Metric Server 에이전트로 리턴합니다. 그 다음에 각 Metric Server 에이전트(각 Dispatcher의)가 클라이언트 요청을 전달할 Dispatcher를 결정하는데 사용하기 위해 로드 상태 값을 상단부 Load Balancer로 리턴합니다.

dsload 실행 프로그램은 Load Balancer용 `...ibm/edge/lb/ms/script` 디렉토리에 있으며, 경과 시간을 로드로 리턴합니다.

WAN 구성의 Dispatcher 사용에 대한 자세한 정보는 244 페이지의 『광역 Dispatcher 지원 구성』을 참조하십시오. Metric Server에 대한 자세한 정보는 211 페이지의 『Metric Server』를 참조하십시오.

사용자 정의(사용자 정의 기능) 어드바이저 작성

사용자 정의(사용자 정의 기능) 어드바이저는 클래스 파일로 제공되는 Java 코드의 작은 부분으로, 기본 코드에서 호출합니다. 기본 코드는 사용자 정의 어드바이저의 인스턴스 시작 및 정지, 상태 및 보고서 제공, 로그 파일에 히스토리 정보 기록과 같은 모든 관리 서비스를 제공합니다. 또한 관리자 컴포넌트로 결과도 보고합니다. 정기적으로 기본 코드는 개별적으로 그 구성의 모든 서버를 평가하는 어드바이저 주기를 수행합니다. 서버 시스템과의 연결을 열어 시작합니다. 소켓이 열리면, 기본 코드는 사용자 정의 어드바이저의 『getLoad』 메소드(함수)를 호출합니다. 그러면 사용자 정의 어드바이저는 서버의 상태를 평가하는 데 필요한 모든 단계를 수행합니다. 일반적으로 서버에 사용자 정의된 메시지를 전송한 후 응답을 기다립니다. (사용자 정의 어드바이저에게 열린 소켓에 대한 액세스가 제공됩니다.) 그러면 기본 코드는 서버와 함께 소켓을 닫고 로드 정보를 관리자에 보고합니다.

기본 코드 및 사용자 정의 어드바이저는 표준 또는 대체 모드에서 작동될 수 있습니다. 조작 모드의 선택사항은 constructor 메소드의 매개변수로서 사용자 정의 어드바이저 파일에 지정됩니다.

표준 모드에서 사용자 정의 어드바이저는 서버와 데이터를 교환하며, 기본 어드바이저 코드는 교환 시간을 정한 후 로그값을 계산합니다. 그런 후 기본 코드는 이 로그값을 관리자에 보고합니다. 사용자 정의 어드바이저는 0(완료 시) 또는 마이너스 1(오류 시)만 리턴해야 합니다. 표준 모드를 지정하기 위해 구성자의 대체 플래그는 거짓으로 설정됩니다.

대체 모드에서 기본 코드는 타이밍 측정을 수행하지 않습니다. 사용자 정의 어드바이저 코드는 그 고유 요구사항에 필요한 모든 조작을 수행한 후 실제 로드 번호를 리턴합니다. 기본 코드는 번호를 승인하고 이를 관리자에 보고합니다. 최상의 결과를 위해서는 10과 1000 사이의 로드 번호를 빠른 서버를 나타내는 10과 느린 서버를 나타내는 1000으로 표준화하십시오. 대체 모드를 지정하기 위해 구성자의 대체 모드는 참으로 설정됩니다.

이 기능으로 사용자가 요구하는 서버에 관한 정확한 정보를 제공하는 사용자 고유의 어드바이저를 작성할 수 있습니다. 사용자 정의 어드바이저의 한 예로 **ADV_sample.java**는 Load Balancer 제품과 함께 제공됩니다. Load Balancer를 설치하면 `...<install directory>/servers/samples/CustomAdvisors` 설치 디렉토리에서 샘플 코드를 찾을 수 있습니다.

기본 설치 디렉토리는 다음과 같습니다.

- AIX, HP-UX, Linux, Solaris 시스템의 경우: /opt/ibm/edge/lb
- Windows 시스템의 경우: C:\Program Files\IBM\edge\lb

주: 사용자 정의 어드바이저를 Dispatcher 또는 다른 적용 가능한 Load Balancer 컴포넌트에 추가할 경우, Java 프로세스가 새 사용자 정의 어드바이저 클래스 파일을 읽을 수 있으려면 **dsserver**(또는 Windows 시스템용 서비스를)를 정지한 후 재시작해야 합니다. 사용자 정의 어드바이저 클래스 파일은 시동 시에만 로드됩니다. 실행 프로그램을 정지할 필요는 없습니다. 실행 프로그램은 dsserver 또는 서비스가 정지되었더라도 계속 실행됩니다.

사용자 정의 어드바이저가 추가의 Java 클래스를 참조하는 경우, Load Balancer 시작 스크립트 파일(dsserver, cbrserver, ssserver)의 클래스 경로는 갱신되어 위치를 포함해야 합니다.

WAS 어드바이저

WAS(WebSphere Application Server) 어드바이저에 대한 특수한 예제 사용자 정의 어드바이저 파일이 Load Balancer 설치 디렉토리에서 제공됩니다.

- ADV_was.java는 Load Balancer 시스템에서 컴파일되어 실행될 파일입니다.
- LBAdvisor.java.servlet(LBAdvisor.java로 이름이 변경됨)은 WebSphere Application Server 시스템에서 컴파일되어 실행될 파일입니다.

WebSphere Application Server 어드바이저 예제 파일은 ADV_sample.java 파일과 동일한 예제 디렉토리에 있습니다.

이름 지정 규칙

사용자 정의 어드바이저 파일 이름은 『ADV_myadvisor.java』 양식이어야 합니다. 이 이름은 대문자 『ADV_』 접두부로 시작해야 합니다. 모든 후속 문자는 소문자여야 합니다.

Java 규칙에 따라 파일 내에 정의된 클래스 이름은 파일 이름과 일치해야 합니다. 예제 코드를 복사할 경우, 파일 내의 모든 『ADV_sample』 인스턴스를 새로운 클래스 이름으로 변경해야 합니다.

컴파일

사용자 정의 어드바이저는 Java 언어로 작성됩니다. Load Balancer와 함께 설치된 Java 컴파일러를 사용하십시오. 이들 파일은 컴파일 시 참조됩니다.

- 사용자 정의 어드바이저 파일
- ...ibm/edge/lb/servers/lib 설치 디렉토리에 있는 기본 클래스 파일 ibmlb.jar

classpath는 컴파일 시 사용자 정의 어드바이저 파일 및 기본 클래스 파일을 모두 연결해야 합니다.

Windows 시스템의 경우, 샘플 컴파일 명령은 다음과 같습니다.

```
install_dir/java/bin/javac -classpath  
install_dir\lb\servers\lib\ibmlb.jar ADV_fred.java
```

여기서:

- 사용자 어드바이저 파일의 이름은 ADV_fred.java입니다.
- 사용자 어드바이저 파일은 현재 디렉토리에 저장됩니다.

컴파일 출력은 클래스 파일입니다. 예를 들어, 다음과 같습니다.

ADV_fred.class

어드바이저를 시작하기 전에 ...ibm/edge/lb/servers/lib/CustomAdvisors 설치 디렉토리에 클래스 파일을 복사하십시오.

주: 필요에 따라 사용자 정의 어드바이저는 한 운영 체제에서 컴파일되어 다른 운영 체제에서 실행될 수 있습니다. 예를 들어, Windows 시스템에서 어드바이저를 컴파일하고 AIX 시스템에 클래스 파일(2진)을 복사하여 여기에서 사용자 정의 어드바이저를 실행할 수 있습니다.

AIX, HP-UX, Linux 및 Solaris 시스템의 경우, 구문이 유사합니다.

실행

사용자 정의 어드바이저를 실행하려면, 먼저 클래스 파일을 적당한 설치 디렉토리에 복사해야 합니다.

```
...ibm/edge/lb/servers/lib/CustomAdvisors/ADV_fred.class
```

Dispatcher를 구성하고, 그 관리자 기능을 시작한 후 사용자 정의 어드바이저를 시작할 명령을 실행하십시오.

```
dscontrol advisor start fred 123
```

여기서:

- fred는 ADV_fred.java의 경우와 같은 어드바이저 이름입니다.
- 123은 어드바이저가 작동될 포트입니다.

사용자 정의 어드바이저가 추가의 Java 클래스를 참조하는 경우, Load Balancer 시작 스크립트 파일(dsserver, cbrserver, ssserver)의 클래스 경로는 갱신되어 위치를 포함해야 합니다.

필수 루틴

모든 어드바이저와 마찬가지로, 사용자 정의 어드바이저는 어드바이저 기본 기능인 ADV_Base를 확장합니다. 이 기능은 Dispatcher 가중치 알고리즘에서 사용하기 위해 로드를 다시 관리자에 보고하는 작업과 같이 대부분의 어드바이저 기능을 실제로 수행하는 어드바이저 기본 기능입니다. 어드바이저 기본 기능은 소켓 연결 및 닫기 조작을 수행하며 어드바이저가 사용할 송수신 메소드도 제공합니다. 어드바이저 자체는 권고 중인 서버의 포트간에 데이터 송수신에만 사용됩니다. 어드바이저 기본 기능 내의 TCP 메소드는 로드를 계산하기 위해 시간 설정됩니다. ADV_base에서 구성자 내의 플래그는 필요에 따라 기존의 로드 위에 어드바이저로부터 리턴된 새로운 로드를 겹쳐씹니다.

주: 구성자의 일련의 값에 따라, 어드바이저 기본 기능은 지정된 간격으로 가중치 알고리즘에 로드를 제공합니다. 실제 어드바이저가 유효한 로드를 리턴할 수 있도록 완료되지 않은 경우, 어드바이저 기본 기능에서는 이전의 로드를 사용합니다.

다음은 기본 클래스 메소드입니다.

- **construct** 루틴. constructor는 기본 클래스 구성자를 호출합니다(예제 어드바이저 파일 참조).
- **ADV_AdvisorInitialize** 메소드. 이 메소드는 기본 클래스가 초기화를 완료한 후 추가 단계가 수행되어야 할 경우에 후크를 제공합니다.
- **getload** 루틴. 기본 어드바이저 클래스는 소켓 열기를 수행하므로, getload는 적절한 송수신 요청을 발행하여 권고 주기를 완료해야 합니다.

탐색 순서

먼저 []는 제공되는 고유한 어드바이저 목록을 검토합니다. 이 목록에 정해진 어드바이저가 없으면 고객의 사용자 정의 어드바이저 목록을 검토합니다.

이름 지정 및 경로

- 사용자 정의 어드바이저 클래스는 Load Balancer 기본 디렉토리의 **...ibm/edge/lb/servers/lib/CustomAdvisors/** 하위 디렉토리 내에 있어야 합니다. 이 디렉토리의 기본값은 운영 체제에 따라 다양합니다.
 - AIX, HP-UX, Linux 및 Solaris 시스템
/opt/ibm/edge/lb/servers/lib/CustomAdvisors/
 - Windows 시스템
C:\Program Files\IBM\edge\lb\servers\lib\CustomAdvisors
- 소문자의 영문자만 허용됩니다. 따라서 조작원은 명령행에 명령을 입력할 때 대소문자를 구별할 필요가 없습니다. 어드바이저 이름에는 **ADV_** 접두부가 있어야 합니다.

예제 어드바이저

예제 어드바이저의 프로그램 목록이 518 페이지의 『어드바이저 예제』에 나와 있습니다. 설치 후에 이 예제 어드바이저는 **...ibm/edge/lb/servers/samples/CustomAdvisors** 디렉토리에서 찾을 수 있습니다.

Metric Server

이 기능은 모든 Load Balancer 컴포넌트에 사용할 수 있습니다.

Metric Server는 Load Balancer에 서버 로드 정보를 시스템별 메트릭 양식으로 제공하여 서버의 상태를 보고합니다. Load Balancer 관리자는 각 서버에 있는 Metric Server를 조회하여 에이전트에서 수집한 메트릭을 사용하여 로드 밸런스 프로세스에 가중치를 지정합니다. 결과도 관리자 보고서에 저장됩니다.

주: 각 서버에 대해 두 개 이상의 메트릭이 수집되어 단일 시스템 로드값으로 표준화될 때 반올림 오류가 발생할 수 있습니다.

Metric Server 운영(시작 및 종료) 및 Metric Server 로그 사용에 대한 정보는 301 페이지의 『Metric Server 컴포넌트 사용』 페이지를 참조하십시오.

16 페이지의 그림 5에서 구성 예제를 참조하십시오.

WLM 제한사항

WLM 어드바이저와 마찬가지로 Metric Server는 개별적인 프로토콜 고유 서버 디먼데서가 아니라 서버 시스템에서 전체로서 보고합니다. WLM 및 Metric Server 모두는 결과를 관리자 보고서의 시스템 컬럼에 저장합니다. 결과적으로 WLM 어드바이저와 Metric Server를 동시에 실행할 수 없습니다.

전제조건

Metric Server 에이전트는 로드 밸런스 중인 모든 서버에서 설치 및 실행되어야 합니다.

Metric Server 사용 방법

다음은 Dispatcher에 대해 Metric Server를 구성하는 단계입니다. Load Balancer의 다른 컴포넌트에 대해 Metric Server를 구성할 때도 이와 비슷한 단계를 사용합니다.

- Load Balancer 관리자(Load Balancer)

1. **dsserver**를 시작하십시오.
2. 명령 발행: **dscontrol manager startmanager.log port**

*port*는 모든 Metric Server 에이전트를 실행하기 위해 선택된 RMI 포트입니다. *metricserver.cmd* 파일에 설정된 기본 RMI 포트는 10004입니다.

3. 명령 발행: **dscontrol metric add cluster::systemMetric**

*systemMetric*은 백엔드 서버에 있는 스크립트의 이름으로 지정된 클러스터(또는 사이트 이름) 아래 구성의 각 서버에서 실행해야 합니다. 고객 **cpuload** 및 **memload**에 대해 두 개의 스크립트가 제공됩니다. 또는 조정 시스템 메트릭 스크립트를 작성할 수 있습니다. 이 스크립트에는 0-100 범위의 숫자 값을 리턴해야 하는 명령 또는 서버가 단절된 경우 -1 값이 들어 있습니다. 이 숫자 값은 가용성 값이 아니라 로드 측정값을 나타내야 합니다.

주: Site Selector의 경우, **cpuload** 및 **memload**가 자동으로 실행됩니다.

제한사항: Windows 플랫폼의 경우 시스템 메트릭 스크립트 이름에 ".exe" 이외의 확장자가 있으면 파일의 전체 이름(예: "mysystemscript.bat")을 지정해야 합니다. 이것은 Java 제한사항 때문입니다.

4. *metricserver.cmd* 파일에 지정된 포트에서 실행하는 Metric Server 에이전트를 포함하는 서버만 구성에 추가하십시오. 포트는 **manager start** 명령에 지정된 포트 값과 같아야 합니다.

주: 보안 확인 —

- Load Balancer 시스템에서 **lbkeys create** 명령을 사용하여 키 파일을 작성하십시오. **lbkeys**에 대한 자세한 정보는 282 페이지의 『원격 메소드 호출(RMI)』을 참조하십시오.
- 백엔드 서버 시스템에서 사용 중인 컴포넌트에 대한 결과 키 파일을 **...ibm/edge/lb/admin/keys** 디렉토리로 복사하십시오. 키 파일의 사용 권한이 루트에서 파일을 읽을 수 있도록 허용하는지 확인하십시오.

- Metric Server 에이전트(서버 시스템)

1. Load Balancer 설치에서 Metric Server 패키지를 설치하십시오.

2. **/usr/bin** 디렉토리에서 **metricsserver** 스크립트를 확인하여 원하는 RMI 포트가 사용되고 있는지 확인하십시오. (Windows 2003의 경우, 디렉토리는 C:\WINDOWS\system32입니다.) 기본 RMI 포트는 10004입니다.

주: 지정된 RMI 포트 값은 Load Balancer 시스템에 있는 Metric Server의 RMI 포트 값과 같아야 합니다.

3. 고객에게 이미 다음의 두 스크립트 **cpuload**(0-100 범위에서 사용 중인 cpu의 비율 리턴) 및 **memload**(0-100 범위에서 사용 중인 메모리 비율 리턴)가 제공되어 있습니다. 이 스크립트는 **...ibm/edge/lb/ms/script** 디렉토리에 있습니다.

선택적으로 사용자는 Metric Server에서 서버 시스템에 대해 발행할 명령을 정의하는 사용자 정의 메트릭 스크립트 파일을 작성할 수 있습니다. 사용자 정의 스크립트는 모두 실행 가능하고 **...ibm/edge/lb/ms/script** 디렉토리에 위치해야 합니다. 사용자 정의 스크립트는 반드시 0-100 범위의 로드값을 리턴해야 합니다.

주: 사용자 정의 메트릭 스크립트는 확장자가 ".bat" 또는 ".cmd"인 유효한 프로그램이나 스크립트여야 합니다. 특히 Linux 및 UNIX 시스템의 경우, 스크립트를 쉘 선언으로 시작해야 합니다. 그렇지 않으면 올바르게 실행되지 않을 수 있습니다.

4. **metricsserver** 명령을 실행하여 에이전트를 시작하십시오.
5. Metric Server 에이전트를 정지하려면 **metricsserver stop** 명령을 실행하십시오.

로컬 호스트 이외의 주소에서 Metric Server를 실행하려면 로드 밸런싱된 서버 시스템의 **metricsserver** 파일을 편집해야 합니다. **metricsserver** 파일에서 "java" 항목 뒤에 다음을 삽입하십시오.

```
-Djava.rmi.server.hostname=OTHER_ADDRESS
```

또한 **metricsserver** 파일에서 "if" 명령문 앞에 다음 행, 즉 **hostname OTHER_ADDRESS**를 추가하십시오.

Windows 플랫폼의 경우: Metric Server 시스템의 Microsoft 스택에서 **OTHER_ADDRESS**의 별명도 지정해야 합니다. 예를 들어,

```
call netsh interface ip add address "Local Area Connection"  
addr=9.37.51.28 mask=255.255.240.0
```

서로 다른 도메인에서 메트릭을 집계할 경우, 서버 스크립트(**dsserver**, **cbrserver** 등)의 **java.rmi.server.hostname**을 메트릭을 요청한 시스템의 완전한 도메인 이름(FQDN)으로 명시적으로 설정해야 합니다. 설정과 운영 체제에 따라 **InetAddress.getLocalHost().getHostName()**에서 FQDN을 리턴하지 않을 수 있기 때문에 이 작업이 필요합니다.

작업로드 관리자 어드바이저

WLM은 MVS 메인프레임에서 실행되는 코드입니다. MVS 시스템에 있는 로드를 문의할 때 조회할 수 있습니다.

MVS Workload Management가 OS/390 시스템에 구성되어 있으면 Dispatcher는 WLM의 용량 정보를 받아들여 로드 밸런싱 프로세스에서 이 정보를 사용할 수 있습니다. WLM 어드바이저를 사용하여 Dispatcher는 정기적으로 Dispatcher에 있는 각 서버의 WLM 포트를 통한 연결을 열어 리턴된 용량 정수를 받아들입니다. 이 정수는 여전히 사용 가능한 용량을 나타내고 Dispatcher에는 각 시스템의 로드를 나타내는 값이 필요하므로 용량 정수는 어드바이저에 의해 변환되어 로드 값으로 표준화됩니다. (즉, 대용량 정수이지만 작은 로드 값은 둘다 보다 안정적인 서버를 나타냅니다.) 결과 로드는 관리자 보고서의 시스템 컬럼에 저장됩니다.

WLM 어드바이저와 기타 Dispatcher 어드바이저 사이의 중요한 몇 가지 차이점은 다음과 같습니다.

1. 기타 어드바이저는 표준 클라이언트 통신량이 이동되는 동일한 포트를 사용하여 서버에 대한 연결을 엽니다. WLM 어드바이저는 표준 통신량과는 다른 포트를 사용하여 서버에 대한 연결을 엽니다. 각 서버 시스템의 WLM 에이전트는 Dispatcher WLM 어드바이저가 시작되는 동일한 포트에서 인식되도록 구성되어야 합니다. 기본 WLM 포트는 10007입니다.
2. 기타 어드바이저는 서버 포트가 어드바이저의 포트와 일치하는 Dispatcher cluster:port:server 구성에서 정의된 서버만 평가합니다. WLM 어드바이저는 Dispatcher 구성(cluster:port와 상관없음)의 모든 서버에 대해 권고합니다. 따라서 WLM 어드바이저 사용시 비 WLM 서버를 정의하지 않아도 됩니다.
3. 기타 어드바이저는 해당 로드 정보를 그 『포트』 컬럼 아래의 관리자 보고서에 저장합니다. WLM 어드바이저는 해당 로드 정보를 그 시스템 아래의 관리자 보고서에 저장합니다.
4. WLM 어드바이저와 함께 프로토콜 고유 어드바이저를 모두 사용할 수 있습니다. 프로토콜 고유 어드바이저는 그 표준 통신 포트에서 서버를 폴링하고 WLM 어드바이저는 WLM 포트를 사용하여 시스템 로드를 폴링합니다.

Metric Server 제한사항

Metric Server 에이전트와 마찬가지로 WLM 에이전트는 개별적인 프로토콜 고유 서버 디먼에서가 아니라 서버 시스템에서 전체로서 보고합니다. Metric Server 및 WLM은 결과를 관리자 보고서의 시스템 컬럼에 저장합니다. 결과적으로 WLM 어드바이저와 Metric Server를 동시에 실행할 수 없습니다.

제 22 장 Dispatcher, CBR 및 Site Selector에 대한 고급 기능

이 장에서는 로드 밸런스 매개변수를 구성하는 방법과 고급 기능의 []를 설정하는 방법에 대해 설명합니다.

주: 이 장을 읽을 때 Dispatcher 컴포넌트를 사용하고 있지 않는 경우, "dscontrol"을 다음 사항으로 대체하십시오.

- CBR의 경우, **cbrcontrol** 사용
- Site Selector의 경우, **sscontrol** 사용(429 페이지의 제 28 장 『Site Selector 명령어 참조서』 참조)

중요: 해당 제품의 IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, 이 장의 내용을 보기 전에 제한사항 및 구성 차이점에 대해 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』 페이지를 참조하십시오.

표 13. Load Balancer의 고급 구성 task

task	설명	관련 정보
[]를 로드 밸런스 중인 시스템에 결합 배치	결합 배치된 Load Balancer 시스템 설정.	216 페이지의 『결합 배치된 서버 사용』
고가용성 또는 상호 고가용성 구성	두 번째 Dispatcher 시스템을 백업 시스템으로 설정합니다.	218 페이지의 『고가용성』
규칙 기반 로드 밸런스 구성	사용자 서버의 서버세트가 사용되는 조건을 정의합니다.	226 페이지의 『규칙 기반 로드 밸런스 구성』
포트 연관 관계 무시를 사용하여 서버가 포트 연관 관계 기능을 무시할 수 있는 메커니즘 제공	서버가 포트에 대한 연관 관계 설정을 무시할 수 있게 허용합니다.	234 페이지의 『포트 연관 관계 무시』
고급(연관 관계)을 사용하여 클러스터의 포트가 엄격하게 유지되도록 구성	클라이언트 요청이 동일한 서버로 전송되도록 허용합니다.	236 페이지의 『Load Balancer에 대한 연관 관계 기능 사용법』
포트간 연관 관계를 사용하여 포트 사이에서 고급 연관 관계 기능 확장	다른 포트에서 수신된 클라이언트 요청이 동일한 서버로 전송되도록 허용합니다.	237 페이지의 『포트간 연관 관계』
연관 관계 주소 마스크를 사용하여 공통 IP 서브넷 주소 지정	동일한 서브넷에서 수신된 클라이언트 요청이 동일한 서버로 전송되도록 허용합니다.	238 페이지의 『연관 관계 주소 마스크 (stickymask)』
활성 쿠키 연관 관계를 사용하여 CBR에 대한 서버의 로드 밸런스	특정 서버에 대해 세션이 연관 관계를 유지할 수 있게 하는 규칙 옵션	240 페이지의 『활성 쿠키 연관 관계』
Dispatcher의 Content Based Routing 및 CBR 컴포넌트의 경우, 수동 쿠키 연관 관계를 사용하여 서버의 로드 밸런스 수행	쿠키 이름/쿠키 값에 기반한 특정 서버의 경우, 세션이 연관 관계를 유지할 수 있도록 허용하는 규칙 옵션	242 페이지의 『수동 쿠키 연관 관계』
URI 연관 관계를 사용하여 개별 서버에 캐시할 고유의 콘텐츠가 있는 Caching Proxy 서버에서 로드 밸런스 수행	URI에 기반한 특정 서버의 경우, 세션이 연관 관계를 유지할 수 있도록 허용하는 규칙 옵션	243 페이지의 『URI 연관 관계』

표 13. Load Balancer의 고급 구성 TASK (계속)

TASK	설명	관련 정보
광역 Dispatcher 지원 구성	원격 Dispatcher를 설정하여 광역 통신망에서 로드 밸런스를 수행합니다. 또는 GRE를 지원하는 서버 플랫폼을 사용하여 원격 Dispatcher 없이 광역 통신망에서 로드 밸런스를 수행합니다.	244 페이지의 『광역 Dispatcher 지원 구성』
명시적 링크 사용	링크에서 Dispatcher를 생략하지 않습니다.	252 페이지의 『명시적 링크 사용』
사설 네트워크 사용	Dispatcher를 구성하여 사설 네트워크 서버의 로드 밸런스를 수행합니다.	252 페이지의 『개인용 네트워크 구성 사용』
와일드 카드 클러스터를 사용하여 공통 서버 구성 조합	명시적으로 구성되지 않은 주소가 통신량 로드 밸런스를 수행하는 방법으로 와일드 카드 클러스터를 사용하게 됩니다.	253 페이지의 『와일드 카드 클러스터를 사용하여 서버 구성 조합』
와일드 카드 클러스터를 사용하여 방화벽 로드 밸런스 수행	방화벽에 대해 모드 통신량의 로드 밸런스를 수행하게 됩니다.	254 페이지의 『와일드 카드 클러스터를 사용하여 방화벽 로드 밸런스 수행』
투명 프록시의 경우, Caching Proxy가 있는 와일드 카드 클러스터 사용	Dispatcher가 투명 프록시를 작동하는 데 사용될 수 있습니다.	254 페이지의 『투명 프록시의 경우 Caching Proxy가 있는 와일드 카드 클러스터 사용』
와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정	특정 포트에 구성되어 있지 않은 통신량을 처리합니다.	255 페이지의 『와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정』
"서비스 거부 중지" 감지를 사용하여 관리자에게 잠재적 시작 통지(경보 사용)	Dispatcher는 서버의 방대한 양의 반개방 TCP 연결에 대한 수신 요청을 분석합니다.	255 페이지의 『서비스 거부 중지 감지』
2진 로그를 사용하여 서버 통계 분석	서버 정보가 2진 파일에 저장되어 이 파일에서 검색될 수 있습니다.	257 페이지의 『서버 통계를 분석하기 위해 2진 로그 사용』
결합 배치된 클라이언트 구성 사용	Load Balancer가 클라이언트와 동일한 시스템에 상주하도록 함	259 페이지의 『결합 배치된 클라이언트 사용』

결합 배치된 서버 사용

[]는 요청의 로드 밸런스가 유지되는 서버와 동일한 시스템에 상주할 수 있습니다. 이것을 보통 서버 결합 배치라고 합니다. 결합 배치는 Dispatcher 및 Site Selector 컴포넌트에 적용됩니다. 결합 배치는 CBR에 대해서도 지원되지만 특정 바인드 웹 서버 및 특정 바인드 Caching Proxy를 사용할 경우에도 지원됩니다.

주: 결합 배치된 서버는 통신량이 많을 때 []와 자원을 차지하기 위해 경쟁합니다. 그러나 과부하된 시스템이 없을 때 결합 배치된 서버를 사용하면 로드 밸런스가 이루어질 때 필요한 총 시스템 수가 줄어듭니다.

Dispatcher 컴포넌트의 경우

Linux 시스템: mac 전달 메소드를 사용하여 Dispatcher 컴포넌트를 실행할 때 결합 배치와고가용성을 동시에 구성하려면 85 페이지의 『Load Balancer의 mac 전달 사용 시 Linux 루프백 별명 지정 대안』을 참조하십시오.

Windows 시스템: mac 전달 메소드를 사용하여 Dispatcher 컴포넌트를 실행할 때 결합 배치와 고가용성을 동시에 구성하려면 225 페이지의 『결합 배치 및 고가용성 구성 (Windows 시스템)』을 참조하십시오.

Solaris 시스템: 시작점 Dispatcher가 결합 배치될 때 WAN 어드바이저를 구성할 수 없는 제한사항이 있습니다. 245 페이지의 『Dispatcher의 광역 지원으로 원격 어드바이저 사용』을 참조하십시오.

이전 버전에서는 결합 배치된 서버의 주소를 구성의 비전달 주소(NFA)와 동일하게 지정해야 했습니다. 이러한 제한은 해결되었습니다.

서버를 결합 배치되도록 구성하기 위해 **dscontrol server** 명령에서 **collocated**라는 옵션을 제공합니다. 이 옵션은 예 또는 아니오로 설정할 수 있습니다. 기본값은 아니오입니다. 서버의 주소는 시스템에 있는 네트워크 인터페이스 카드의 유효한 IP 주소여야 합니다. 결합 배치된 매개변수는 Dispatcher의 nat 또는 cbr 전달 메소드를 사용하여 결합 배치된 서버에 설정될 수 없습니다.

다음 방법 중 하나로 결합 배치된 서버를 구성할 수 있습니다.

- NFA를 결합 배치된 서버 주소로 사용하는 경우, **dscontrol executor set nfa IP_address** 명령을 사용하여 NFA를 설정하십시오. 그리고 **dscontrol server add cluster:port:server** 명령과 함께 NFA 주소를 사용하여 서버를 추가하십시오.
- NFA 이외의 주소를 사용하는 경우, 다음과 같이 yes로 설정된 **collocated** 매개변수를 사용하여 원하는 IP 주소가 있는 서버를 추가하십시오. **dscontrol server add cluster:port:server collocated yes**

Dispatcher의 nat 또는 cbr 전달의 경우 NFA에 사용하지 않은 어댑터 주소를 (별명으로) 구성해야 합니다. 서버가 해당 주소를 청취하도록 구성되어야 합니다. 다음 명령 구문을 사용하여 서버를 구성하십시오.

```
dscontrol server add cluster:port:new_alias address new_alias router router_ip  
returnaddress return_address
```

이 구성을 제대로 하지 않으면 시스템 오류가 발생하거나 서버로부터 응답이 없거나 아니면 두 가지 오류 모두가 발생할 수 있습니다.

Dispatcher의 nat 전달을 사용하여 서버 결합 배치 구성

Dispatcher의 nat 전달 메소드를 사용하여 결합 배치 서버를 구성하는 경우 **dscontrol server add** 명령에 지정된 라우터는 서버 IP 주소가 아니라 실제 라우터 주소여야 합니다.

Dispatcher의 nat 전달 메소드를 구성할 때 Dispatcher 시스템에서 다음 단계를 수행할 경우 모든 운영 체제에서 결합 배치 지원을 수행할 수 있습니다.

- **AIX** 시스템의 경우, 결합 배치된 서버는 임의의 서버와 동일하게 구성됩니다. 구성은 변경할 필요가 없습니다.
- **Linux** 시스템의 경우, 결합 배치된 서버는 임의의 서버와 동일하게 구성됩니다. 구성은 변경할 필요가 없습니다.
- **Solaris** 및 **HP-UX** 시스템의 경우, 클러스터에는 일반적으로 ifconfig를 사용하여 별명이 지정됩니다. 그러나 리턴 주소는 별명을 지정하는 대신 arp publish를 사용해야 합니다. 이렇게 하려면 다음 명령을 실행하십시오.

```
arp -s hostname ether_addr pub
```

ether_addr에는 로컬 MAC 주소를 사용하십시오. 그러면 로컬 응용프로그램이 커널의 리턴 주소에 통신량을 전송할 수 있습니다.

- **Windows** 플랫폼의 경우, 클러스터 및 리턴 주소는 **dscontrol executor configure** 명령을 사용하여 구성되어야 하며 Windows 네트워킹에 배치되어서는 안 됩니다. 로컬 응용프로그램의 경우 새 IP 별명을 Windows 네트워킹의 로컬 어댑터에 추가해야 합니다. TCP/IP 설정 아래에서 고급 옵션을 찾아 어댑터에 추가 IP를 추가하십시오. 이 두 번째 IP는 Dispatcher 구성의 서버 정의로서 사용됩니다.

CBR 컴포넌트

CBR은 추가로 구성할 필요없이 모든 플랫폼에서 결합 배치를 지원합니다. 그러나 사용하는 웹 서버와 Caching Proxy는 특정 바인드이어야 합니다.

Site Selector 컴포넌트의 경우

Site Selector는 추가로 구성할 필요없이 모든 플랫폼에서 결합 배치를 지원합니다.

고가용성

고가용성 기능(**dscontrol highavailability** 명령을 사용하여 구성 가능)은 Dispatcher 컴포넌트에 사용 가능합니다(CBR 또는 Site Selector 컴포넌트에는 사용 가능하지 않음).

Dispatcher 가용도를 향상시키기 위해 Dispatcher 고가용성 기능에서는 다음과 같은 메커니즘을 사용합니다.

- Dispatcher 사이의 연결 가능성과 동일한 클라이언트 및 서버의 동일한 클러스터로의 연결 가능성이 있는 두 개의 Dispatcher. 두 Dispatcher는 동일한 운영 체제 및 플랫폼에서 작동해야 합니다.
- Dispatcher 고장을 발견하기 위한 두 Dispatcher 사이의 『하트 비트(heartbeat)』 메커니즘. 적어도 하나 이상의 하트 비트 쌍에는 출발지 및 대상 주소로서 NFA 쌍이 있어야 합니다.

가능하면 하트 비트 쌍 중 최소한 하나는 일반 클러스터 통신이 아니라 별도의 서브넷을 통해 전송해야 합니다. 하트 비트를 별도로 전송하면 네트워크 로드가 아주 많은 경우 데이터를 가로채지 못하게 막을 수 있으며 장애 발생 후 전체 복구 시간을 줄일 수 있습니다.

- 통신량 로드 밸런스를 정상적으로 유지하기 위해 두 대의 Dispatcher 시스템이 접속할 수 있는 주소인 도달 목표 목록. 자세한 내용은 222 페이지의 『하트 비트 및 도달 목표를 사용하는 고장 검색 기능』을 참조하십시오.
- Dispatcher 정보의 동기화(즉, 연결 테이블, 도달 가능성 테이블 및 기타 정보).
- 제공된 서버 클러스터를 담당하는 활성 Dispatcher와 서버의 해당 클러스터에 연속적으로 동기화되는 대기 Dispatcher를 선택하기 위한 로직.
- 로직 또는 작업원이 활성 및 대기 상태를 전환하기로 결정한 경우 신속한 IP 인계를 수행하기 위한 메커니즘.

주: 두 클러스터 설정을 공유하는 두 개의 Dispatcher 시스템이 서로에 대한 백업을 제공하는 상호 고가용성 구성에 대한 예시 및 설명은 67 페이지의 『상호 고가용성』을 참조하십시오. 상호 고가용성은 고가용성과 유사하지만 Dispatcher 시스템 전체가 아니라 클러스터 주소에 특수하게 기반을 두고 있습니다. 두 시스템은 공유된 클러스터 설정이 동일하게 구성되어야 합니다.

고가용성 구성

dscontrol 고가용성의 완전한 구문은 391 페이지의 『dscontrol highavailability — 고가용성 제어』에 나와 있습니다.

다음의 많은 task에 대해 자세히 알려면 72 페이지의 『Dispatcher 시스템 설정』을 참조하십시오.

1. 두 대의 Dispatcher 시스템에 별명 스크립트 파일을 작성하십시오. 223 페이지의 『스크립트 사용』을 참조하십시오.
2. 두 대의 Dispatcher 서버 시스템 모두에서 서버를 시작하십시오.
3. 두 시스템 모두에서 실행 프로그램을 시작하십시오.
4. 각 Dispatcher 시스템의 비전달 주소(NFA)가 구성되어 있고 Dispatcher 시스템의 서브넷에 유효한 IP 주소인지 확인하십시오.
5. 두 시스템 모두에 하트 비트를 추가하십시오.

```
dscontrol highavailability heartbeat add sourceaddress destinationaddress
```

주: *Sourceaddress* 및 *destinationaddress*는 Dispatcher 시스템의 IP 주소(DNSnames 또는 IP 주소)입니다. 값은 각 시스템에서 역행될 것입니다. 예를 들어,

```
Primary - highavailability heartbeat add 9.67.111.3 9.67.186.8  
Backup - highavailability heartbeat add 9.67.186.8 9.67.111.3
```

적어도 하나 이상의 하트 비트 쌍에는 출발지 및 대상 주소로서 NFA 쌍이 있어야 합니다.

가능하면 하트 비트 쌍 중 최소한 하나는 일반 클러스터 통신이 아니라 별도의 서브넷을 통해 전송해야 합니다. 하트 비트를 별도로 전송하면 네트워크 로드가 아주 많은 경우 데이터를 가로채지 못하게 막을 수 있으며 장애 발생 후 전체 복구 시간을 줄일 수 있습니다.

고가용성 하트비트의 설정을 초과하기 위해 실행 프로그램에서 사용하는 초 수를 설정하십시오. 예를 들어,

```
dscontrol executor set hatimeout 3
```

기본값은 2초입니다.

6. 두 시스템 모두에서 **reach add** 명령을 사용하여 완전한 서비스를 보장하기 위해 Dispatcher가 도달할 수 있어야 하는 IP 주소의 목록을 구성하십시오. 예를 들어,

```
dscontrol highavailability reach add 9.67.125.18
```

도달 목표는 권장되지만 필수는 아닙니다. 222 페이지의 『하트 비트 및 도달 목표』를 사용하는 고장 검색 기능』에서 자세한 내용을 참조하십시오.

7. 각 시스템에 백업 정보를 추가하십시오.

- 기본 시스템의 경우:

```
dscontrol highavailability backup add primary [auto | manual] port
```

- 백업 시스템의 경우:

```
dscontrol highavailability backup add backup [auto | manual] port
```

- 상호 고가용성의 경우, 각 Dispatcher 시스템은 기본 및 백업 역할이 모두 있습니다.

```
dscontrol highavailability backup add both [auto | manual] port
```

주: 시스템에서 사용하지 않는 포트를 *port*로 선택하십시오. 입력 포트 번호는 키로 사용되어 올바른 호스트가 패킷을 수신하도록 합니다.

8. 각 시스템에서 고가용성 상태를 확인하십시오.

```
dscontrol highavailability status
```

시스템에는 각각 올바른 역할(백업 및 기본 또는 둘다), 상태 및 부속 상태가 있어야 합니다. 기본은 활성화되어 동기화되며, 백업은 대기 모드에 있고 단시간 내에 동기화되어야 합니다. 전략어는 같아야 합니다.

9. 두 시스템 모두에 클러스터, 포트 및 서버 정보를 설정하십시오.

주: 상호 고가용성 구성(67 페이지의 그림 14)의 경우, 두 Dispatcher 간에 공유되는 클러스터 설정을 다음과 같이 구성하십시오.

- Dispatcher 1의 경우 다음을 발행하십시오.

```
dscontrol cluster set clusterA primaryhost NFAdispatcher1
dscontrol cluster set clusterB primaryhost NFAdispatcher2
```

- Dispatcher 2의 경우 다음을 발행하십시오.

```
dscontrol cluster set clusterB primaryhost NFAdispatcher2
dscontrol cluster set clusterA primaryhost NFAdispatcher1
```

10. 두 시스템 모두에서 관리자와 어드바이저를 시작하십시오.

주:

1. 하나의 Dispatcher 시스템을 구성하여 백업 없이 패킷을 경로 지정하려면, 시동 시 어떤 고가용성 명령도 발행하지 마십시오.
2. 고가용성으로 구성된 두 대의 Dispatcher 시스템을 단독으로 실행 중인 하나의 시스템으로 변환하려면, 시스템 중 하나에서 실행 프로그램을 정지한 후 나머지 시스템에서 고가용성 기능(하트 비트, 도달 및 백업)을 삭제하십시오.
3. 위의 두 경우 모두에, 필요에 따라 네트워크 인터페이스 카드에 클러스터 주소로 별명을 지정해야 합니다.
4. 두 대의 Dispatcher 시스템이 고가용성 구성에서 실행되어 동기화되면, 먼저 대기 시스템에 모든 dscontrol 명령(구성을 갱신하기 위해)을 입력한 다음에 활성 시스템에 입력하십시오.
5. 고가용성 구성에서 두 대의 Dispatcher 시스템을 실행할 경우, 실행 프로그램, 클러스터, 포트 또는 서버의 매개변수(예: port stickytime)를 두 시스템에서 다른 값으로 설정하면 예기치 못한 결과가 발생할 수도 있습니다.
6. 상호 고가용성의 경우, Dispatcher 중 하나가 백업 클러스터에 대한 패킷의 경로 지정을 인계할 뿐 아니라 기본 클러스터에 대한 패킷을 실제로 라우트해야 하는 상황을 고려해 보십시오. 이것은 이 시스템의 처리량을 초과하지 않아야 합니다.
7. Linux 시스템의 경우, Dispatcher 컴포넌트의 MAC 포트 전달 메소드를 사용할 때 고가용성과 결합 배치를 함께 구성하려면 85 페이지의 『Load Balancer의 mac 전달 사용 시 Linux 루프백 별명 지정 대안』을 참조하십시오.
8. Windows 시스템의 경우, 고가용성과 결합 배치를 함께 구성하려면 225 페이지의 『결합 배치 및 고가용성 구성(Windows 시스템)』을 참조하십시오.
9. 고가용성 구성 문제에서 기인한 다음과 같은 문제점의 완화에 도움이 되는 팁이 있습니다.
 - 인계 후에 연결이 끊김
 - 상대 시스템과 동기화가 불가능함
 - 요청이 백업 상대 시스템으로 잘못 방향 지정됨

340 페이지의 『문제점: 고가용성 구성에 대한 팁』을 참조하십시오

하트 비트 및 도달 목표를 사용하는 고장 검색 기능

고장을 발견하는 기본적인 기준 이외에도(하트 비트를 통해 발견되는 활성 및 대기 Dispatcher간의 연결성 유실), 도달 가능성 기준이라는 또 다른 고장 발견 메커니즘이 있습니다. Dispatcher 구성 시, 올바른 작동을 위해 각 Dispatcher에서 도달할 수 있는 호스트 목록을 제공할 수 있습니다. 두 명의 고가용성 상대가 하트비트로 서로 계속 통신하며 상대 중 하나가 ping할 수 있는 도달 목표의 수에 대해 서로 갱신합니다. 대기 중에서 활성화된 것보다 더 많은 도달 목표를 ping할 경우 장애가 발생합니다.

0.5초마다 활성 Dispatcher에서 하트비트를 전송하고 대기 Dispatcher에서 이를 수신합니다. 2초 안에 대기 Dispatcher에서 하트비트를 받지 못하면 장애가 시작됩니다. 대기 Dispatcher의 인계가 발생하려면 모든 하트비트를 중단해야 합니다. 즉, 두 개의 하트비트 쌍이 구성된 경우 두 하트비트를 모두 중단해야 합니다. 고가용성 환경을 안정시키고 장애를 방지하려면 둘 이상의 하트비트 쌍을 추가하십시오.

도달 목표의 경우 Dispatcher 시스템에서 사용하는 서브넷마다 최소한 하나의 호스트를 선택해야 합니다. 호스트는 라우터, IP 서버 또는 다른 유형의 호스트일 수 있습니다. 호스트를 ping하는 도달 어드바이저에 의해 호스트 도달 가능성이 확보됩니다. 하트비트가 전달될 수 없거나, 도달 가능성 기준에 기본 Dispatcher가 아닌 대기 Dispatcher가 더 적합한 경우에는 장애가 발생합니다. 사용 가능한 모든 정보에 근거하여 결정하려면, 활성 Dispatcher는 정기적으로 대기 Dispatcher에 그 도달 가능성 기능을 보냅니다. 그러면 대기 Dispatcher는 그 고유의 기능과 비교하여 전환 여부를 결정합니다.

주: 도달 목표를 구성할 때 도달 어드바이저도 시작해야 합니다. 관리자 기능을 시작할 경우 도달 어드바이저가 자동으로 시작됩니다. 도달 어드바이저에 대한 자세한 정보는 204 페이지를 참조하십시오.

복구 전략

두 대의 Dispatcher 시스템이 구성되며, 이는 기본 시스템과 2차 시스템인 백업 시스템입니다. 시동 시, 기본 시스템은 해당 시스템이 동기화될 때까지 모든 연결 데이터를 백업 시스템으로 전송합니다. 기본 시스템은 활성화되어 로드 밸런스를 시작합니다. 그 동안 백업 시스템은 기본 시스템의 상태를 모니터링하며, 이를 대기 상태에 있다고 합니다.

어느 지점에서든 백업 시스템이 기본 시스템의 고장을 발견하면, 이 백업 시스템이 기본 시스템의 로드 밸런스 기능의 *takeover*를 수행하여 활성화됩니다. 기본 시스템이 일단 다시 작동하게 되면, 시스템은 사용자가 복구 전략어를 구성한 방법에 따라 대응합니다. 두 종류의 전략어가 있습니다.

자동 기본 시스템은 다시 작동되는 즉시 패킷의 경로 지정을 재개합니다.

수동 백업 시스템은 기본 시스템이 작동된 후에도 패킷의 경로 지정을 계속합니다. 기본 시스템을 활성 상태로 리턴하고 백업 시스템을 대기 상태로 재설정하려면 사용자가 수동으로 개입해야 합니다.

전략 매개변수는 두 시스템에 동일하게 설정되어야 합니다.

수동 복구 전략어로 takeover 명령을 사용하여 특정 시스템에 대한 패킷의 경로 지정을 강행할 수 있습니다. 수동 복구는 다른 시스템에서 유지보수될 때 유용합니다. 자동 복구 전략어는 정상적인 무인 조작을 위해 설계되었습니다.

상호 고가용성 구성의 경우, 클러스터당 장애는 발생하지 않습니다. 한 대의 시스템에 문제가 발생하면 한 클러스터에만 영향을 미치지만 다른 시스템이 두 클러스터에 대한 작업을 인계 받습니다.

주: takeover 상황 동안 일부 연결 갱신사항이 손실될 수 있습니다. 이런 경우 takeover 시 액세스되는 기존의 장기 실행 연결(예: Telnet)이 종료될 수도 있습니다.

스크립트 사용

패킷을 경로 지정할 Dispatcher의 경우, 네트워크 인터페이스 장치에 대해 각 클러스터 주소의 별명이 지정되어야 합니다.

- 독립형 Dispatcher 구성에서 네트워크 인터페이스 카드에 대해 각 클러스터 주소의 별명이 지정되어야 합니다(예: en0, tr0).
- 고가용성 구성의 경우:
 - 활성 시스템에서 네트워크 인터페이스 카드에 대해 각 클러스터 주소의 별명이 지정되어야 합니다(예: en0, tr0).
 - 결합 배치된 서버로 mac 전달 메소드를 사용하는 경우, 대기 시스템에서 루프백 장치에 대해 각 클러스터 주소의 별명이 지정되어야 합니다(예: lo0).
- 실행 프로그램에서 정지한 시스템에서는 모든 별명이 제거되어 시작될 수 있는 다른 시스템과 충돌하지 않아야 합니다.

네트워크 인터페이스 카드의 별명 지정에 대한 정보는 76 페이지의 『5단계. 네트워크 인터페이스 카드의 별명 지정』을 참조하십시오.

고장이 발견되면 Dispatcher 시스템의 상태가 변경되므로, 위의 명령은 자동으로 발행되어야 합니다. 이렇게 하기 위해 Dispatcher는 사용자가 작성한 스크립트를 실행합니다. 예제 스크립트는 **...ibm/edge/lb/servers/samples** 디렉토리에 있으며, 실행시 **...ibm/edge/lb/servers/bin** 디렉토리로 반드시 이동해야 합니다.dsserver가 실행 중인 경우에만 스크립트가 자동 실행됩니다.

주:

1. 상호 고가용성 구성에서 각 "go" 스크립트는 기본 Dispatcher 주소를 식별하는 매개변수를 사용하여 Dispatcher에서 호출합니다. 스크립트에서 이 매개변수를 조회하고 기본 Dispatcher와 연관된 클러스터 주소에 대해 **executor configure** 명령을 수행해야 합니다.
2. Dispatcher의 nat 전달 메소드에 대한 고가용성을 구성하려면 스크립트 파일에 리턴 주소를 추가해야 합니다.

다음의 예제 스크립트를 사용할 수 있습니다.

goActive

goActive 스크립트는 Dispatcher가 활성 상태가 되어 패킷 경로 지정을 시작할 때 실행합니다.

- 고가용성 구성에서 Dispatcher를 실행할 경우, 이 스크립트를 작성해야 합니다. 이 스크립트는 루프백 별명을 삭제하고 장치 별명을 추가합니다.
- 독립형 구성에서 Dispatcher를 실행할 경우, 이 스크립트는 필요없습니다.

goStandby

goStandby 스크립트는 Dispatcher가 대기 상태가 되어 활성 시스템의 상태를 모니터하지만 어떤 패킷의 경로도 지정하지 않을 때 실행합니다.

- 고가용성 구성에서 Dispatcher를 실행할 경우, 이 스크립트를 작성해야 합니다. 이 스크립트는 장치 별명을 삭제하고 루프백 별명을 추가합니다.
- 독립형 구성에서 Dispatcher를 실행할 경우, 이 스크립트는 필요없습니다.

goInOp

Dispatcher 실행 프로그램이 중지되면 goInOp 스크립트가 실행됩니다.

- 고가용성 구성에서 Dispatcher를 실행할 경우, 이 스크립트를 작성해야 합니다. 이 스크립트는 모든 장치 및 루프백 별명을 삭제합니다.
- 독립형 구성에서 Dispatcher를 실행할 경우, 이 스크립트는 선택입니다. 이 스크립트를 작성하여 장치 별명을 삭제하거나 수동으로 별명을 삭제하도록 선택할 수 있습니다.

goIdle goIdle 스크립트는 Dispatcher가 유휴 상태가 되어 패킷의 경로 지정을 시작할 때 실행합니다. 이것은 고가용성 기능이 독립형 구성에서와 마찬가지로 추가되지 않은 경우에 발생합니다. 또한 고가용성 기능이 추가되기 전이나 제거된 후에 고가용성 구성에서도 발생합니다.

- 고가용성 구성에서 Dispatcher를 실행할 경우, 이 스크립트를 작성하지 마십시오.
- 독립형 구성에서 Dispatcher를 실행할 경우, 이 스크립트는 선택입니다. 이 스크립트를 작성하여 여기에 장치 별명을 추가하거나 수동으로 별명을 추가하도록 선택할 수 있습니다. 독립형 구성에 대해 이 스크립트를 작성하지 않

으면 **dscontrol executor configure** 명령을 사용하거나 실행 프로그램이 시작될 때마다 수동으로 별명을 구성해야 합니다.

highavailChange

highavailChange 스크립트는 "go" 스크립트 중 하나가 호출되는 것과 같이 Dispatcher 내에서 고가용성 상태가 변경될 때마다 실행합니다. 이 스크립트에 전달된 단일 매개변수는 Dispatcher에 의해서만 실행되는 "go" 스크립트의 이름입니다. 예를 들어, 관리자에 경보를 보내거나 단순히 이벤트를 기록하기 위해, 이 스크립트를 작성하여 상태 변경 정보를 사용할 수 있습니다.

Windows 시스템의 경우: 구성 설정 시 Site Selector가 고가용성 환경에서 두 개의 Dispatcher 시스템을 로드 밸런스하도록 하는 경우 Metric Server의 Microsoft 스택에 별명을 추가해야 합니다. 이 별명을 goActive 스크립트에 추가해야 합니다. 예를 들어,

```
call netsh interface ip add address "Local Area Connection"  
addr=9.37.51.28 mask=255.255.240.0
```

goStandby 및 goInOp에서 별명을 제거해야 합니다. 예를 들어,

```
call netsh interface ip delete address "Local Area Connection"  
addr=9.37.51.28
```

시스템에 다중 NIC가 있는 경우, 먼저 명령 프롬프트에서 netsh interface ip show address 명령을 발행하여 어떤 인터페이스를 사용해야 할지를 확인하십시오. 이 명령은 최근 구성한 인터페이스 목록을 리턴하고 "Local Area Connection"에 번호를 부여하여(예: "Local Area Connection 2") 사용자가 사용할 인터페이스를 결정할 수 있도록 합니다.

S/390®용 Linux: Dispatcher 간에 IP 주소를 이동하기 위해 Dispatcher에서 불필요한 ARP를 발행합니다. 따라서 이 메커니즘은 기본 네트워크 유형에 연결되어 있습니다. S/390용 Linux를 실행할 경우 Dispatcher는 불필요한 ARP를 발행하고 로컬 인터페이스에 주소를 구성할 수 있는 인터페이스에서만 기본적으로 고가용성 인계(IP 주소 이동이 완료됨)를 수행할 수 있습니다. IUCV 및 CTC 같은 지점간 인터페이스에서는 이 메커니즘이 제대로 작동하지 않고 qeth/QDIO의 특정 구성에서도 제대로 작동하지 않습니다.

Dispatcher의 기본 IP 인계 기능이 제대로 작동하지 않는 인터페이스와 구성의 경우 고객이 적절한 명령을 이동 스크립트에 배치하여 수동으로 주소를 이동할 수 있습니다. 그러면 해당 네트워크 토폴로지는 고가용성의 혜택을 받을 수도 있습니다.

결합 배치 및 고가용성 구성(Windows 시스템)

Windows 서버에서 고가용성 및 결합 배치를 구성할 수 있습니다. Windows 시스템에서 해당 Load Balancer 기능을 구성하기 위해서는 추가 단계가 필요합니다.

Windows 시스템에서 고가용성 및 결합 배치를 사용하는 경우, 시스템의 루프백 어댑터에 추가할 수 있는 더미 IP 주소의 일종인 추가 IP 주소가 필요합니다. 루프백 어댑터는 기본 시스템 및 백업 시스템 모두에 설치되어야 합니다. Windows 시스템에 루프백 장치를 설치하려면 79 페이지의 『서버 시스템의 시스템 설정』에 설명된 단계를 따르십시오.

단계가 클러스터 IP 주소를 루프백에 추가하도록 표시하는 경우, 클러스터 주소가 아닌 더미 IP 주소를 추가해야 합니다. Windows 시스템의 고가용성 go* 스크립트는 Load Balancer 시스템이 활성화 되었는지 대기 상태인지에 따라 클러스터 주소를 삭제하고 루프백 장치에 추가해야 하기 때문입니다.

루프백 장치는 DHCP 모드에서 기능하지 않으므로 Windows 시스템은 마지막 구성된 IP 주소가 루프백 장치에서 삭제되지 않도록 합니다. 더미 주소는 Load Balancer가 언제라도 클러스터 주소를 제거할 수 있도록 합니다. 더미 IP 주소는 통신량 유형에 사용되지 않으며 활성화 및 대기 시스템 모두에 사용될 수 있습니다.

활성 및 대기 시스템의 Load Balancer go* 스크립트를 갱신하고 이동한 후 Dispatcher를 시작하십시오. 클러스터 주소가 추가되어 적절한 시기에 네트워크 인터페이스 및 루프백 장치 모두에서 제거됩니다.

규칙 기반 로드 밸런스 구성

규칙 기반 로드 밸런스를 사용하여 패킷이 언제 무슨 이유로 어떤 서버로 전송되는지를 조정할 수 있습니다. Load Balancer는 사용자가 추가하는 규칙을 첫 번째 우선순위에서 마지막 우선순위까지 검토하며, 처음으로 발견한 올바른 규칙에서 중단한 후 해당 규칙과 연관된 서버 간 콘텐츠의 로드 밸런스를 수행합니다. 대상과 포트에 따라 이미 로드 밸런스를 수행하지만 규칙을 사용하면 사용자의 능력이 확장되어 연결을 분산시킬 수 있습니다.

규칙을 구성하는 대부분의 경우, 우선순위가 높은 다른 규칙에 의해 전달된 요청을 받기 위해 기본적으로 항상 참인 규칙을 구성하십시오. 이 기본값은 다른 모든 서버가 클라이언트 요청에 실패할 때 "죄송합니다. 사이트가 현재 작동 중지되었습니다. 나중에 다시 시도하십시오."라는 응답이 될 수 있습니다.

어떤 이유로 서버의 서브세트를 사용하려는 경우, Dispatcher 컴포넌트 및 Site Selector와 함께 규칙 기반 로드 밸런스를 사용해야 합니다. 항상 CBR 컴포넌트의 규칙을 사용해야 합니다.

다음 유형의 규칙에서 선택할 수 있습니다.

- Dispatcher의 경우:
 - 클라이언트 IP 주소
 - 클라이언트 포트

- 시간
- 서비스 유형(TOS)
- 초당 연결 수
- 총 작동 중인 연결 수
- 예약된 대역폭
- 공유 대역폭
- 항상 참
- 요청 콘텐츠
- CBR의 경우:
 - 클라이언트 IP 주소
 - 시간
 - 초당 연결 수
 - 총 작동 중인 연결 수
 - 항상 참
 - 요청 콘텐츠
- Site Selector의 경우:
 - 클라이언트 IP 주소
 - 시간
 - Metric All
 - Metric Average
 - 항상 참

사용자 구성에 규칙 추가를 시작하기 전에 규칙이 수행해야 할 로직을 계획하십시오.

규칙 평가 방법

모든 규칙에는 이름, 유형, 우선순위가 있으며, 서버 세트와 함께 시작 및 종료 범위가 있을 수도 있습니다. 또한 CBR 컴포넌트의 콘텐츠 유형 규칙에는 규칙과 연관된 일치하는 일반 표현식 패턴이 있습니다. (콘텐츠 규칙 및 콘텐츠 규칙의 유효한 패턴 구문을 사용하는 방법에 대한 예제 및 시나리오는 507 페이지의 부록 B 『콘텐츠 규칙(패턴) 구문』을 참조하십시오.)

규칙은 우선순위대로 평가합니다. 다시 말해서 우선순위가 1(낮은 숫자)인 규칙을 우선순위가 2(높은 숫자)인 규칙보다 먼저 평가합니다. 만족시키는 첫 번째 규칙을 사용합니다. 규칙이 만족되면 더 이상 규칙을 평가하지 않습니다.

만족되는 규칙은 다음 두 조건에 맞아야 합니다.

1. 규칙의 술어가 참이어야 합니다. 즉, 평가 중인 값은 시작 범위와 종료 범위 사이에 있거나 콘텐츠는 콘텐츠 규칙 패턴에 지정된 일반 표현식과 일치해야 합니다. "참" 유형의 규칙인 경우, 술어는 시작 및 종료 범위에 관계없이 항상 만족됩니다.
2. 규칙에 관련된 서버가 있는 경우, 최소한 하나에는 패킷을 전달하는 데 0보다 큰 가중치가 있어야 합니다.

규칙과 관련되는 서버가 없는 경우, 해당 규칙은 조건 1만 만족시키면 됩니다. 이 경우에 Dispatcher는 연결 요청을 끊고 Site Selector는 오류가 있는 이름 서버 요청을 리턴하며 CBR은 Caching Proxy가 오류 페이지를 리턴하도록 합니다.

규칙을 전혀 만족시키지 않는 경우, Dispatcher는 포트에서 사용할 수 있는 전체 서버 세트에서 서버를 선택하고 Site Selector는 사이트 이름에서 사용할 수 있는 전체 서버 세트에서 서버를 선택하며 CBR은 Caching Proxy가 오류 페이지를 리턴하도록 합니다.

클라이언트 IP 주소에 따라 규칙 사용

이 규칙 유형은 Dispatcher, CBR 또는 Site Selector 컴포넌트에서 사용할 수 있습니다.

고객을 화면에 표시하고 그 공급처에 따라 자원을 할당하려면, 클라이언트 IP 주소에 따라 규칙을 사용하려고 할 수 있습니다.

예를 들어, 네트워크에 특정 IP 주소 세트에서 오는 클라이언트로부터 지급되지 않아서 원하지 않는 통신량이 많아진다는 것을 인식합니다. **dscontrol rule** 명령을 사용하여 규칙을 작성할 수 있습니다.

```
dscontrol rule add 9.67.131.153:80:ni type ip
  beginrange 9.0.0.0 endrange 9.255.255.255
```

이 "ni" 규칙은 원하지 않는 클라이언트로부터의 연결을 화면에 표시합니다. 그런 다음 액세스할 서버를 규칙에 추가하거나 서버를 규칙에 추가하지 않을 경우 어떤 서버에서도 9.x.x.x 주소로부터의 요청이 제공되지 않습니다.

클라이언트 포트에 따라 규칙 사용

Dispatcher 컴포넌트의 경우에만 이 규칙 유형을 사용할 수 있습니다.

클라이언트가 요청 작성 시, TCP/IP로부터 특정 포트를 요청하는 몇 가지 종류의 소프트웨어를 사용 중이면 클라이언트 포트에 따라 규칙을 사용하고자 할 수 있습니다.

예를 들어, 사용자는 클라이언트 포트 10002에서의 요청이 엘리트 고객 그룹으로부터 수신 중이라는 것을 알고 있으므로, 해당 포트에서의 요청이 특수한 빠른 서버 세트를 사용하도록 지시하는 규칙을 작성할 수 있습니다.

시간에 따라 규칙 사용

이 규칙 유형은 Dispatcher, CBR 또는 Site Selector 컴포넌트에서 사용할 수 있습니다.

용량을 계획하기 위해 시간에 따라 규칙을 사용하고자 할 수 있습니다. 예를 들어, 웹 사이트를 매일 같은 시간대에 작동하는 경우, 최대 활동 시간 동안 5개의 추가 서버를 전용하려고 할 수 있습니다.

시간에 따라 규칙을 사용할 수 있는 또다른 이유는 매일 밤 자정에 유지보수를 위해 몇몇 서버를 종료시킬 때이므로, 필수 유지보수 기간 동안 이러한 서버들을 제외하는 규칙을 설정할 수 있습니다.

서비스 유형(TOS)에 기반하여 규칙 사용

Dispatcher 컴포넌트의 경우에만 이 규칙 유형을 사용할 수 있습니다.

IP 헤더의 “서비스 유형”(TOS) 필드의 콘텐츠에 기반하여 규칙을 사용할 수 있습니다. 예를 들어, 클라이언트 요청에 표준 서비스를 나타내는 하나의 TOS 값이 포함되면 서버 세트 중 하나로 라우트될 수 있습니다. 다른 클라이언트 요청에 더 높은 우선 순위의 서비스를 나타내는 다른 TOS 값이 포함되면 다른 서버 세트로 라우트될 수 있습니다.

TOS 규칙을 사용하면 **dscontrol rule** 명령을 통해 TOS 바이트의 각 비트를 완전히 구성할 수 있습니다. TOS 바이트에서 일치시키려는 최상위 비트의 경우, 0이나 1을 사용하십시오. 그렇지 않은 경우, x 값이 사용됩니다. 다음은 TOS 규칙 추가에 대한 예제입니다.

```
dscontrol rule add 9.67.131.153:80:tsr type service tos 0xx1010x
```

초당 연결에 따라 규칙 사용

이 규칙 유형은 Dispatcher 및 CBR 컴포넌트에서 사용할 수 있습니다.

주: 다음을 작업하기 위해서는 관리자가 실행되고 있어야 합니다.

다른 응용프로그램과 사용자 서버 중 일부를 공유해야 하는 경우, 초당 연결 수에 따라 규칙을 사용하고자 할 수 있습니다. 예를 들어, 다음과 같은 두 가지 규칙을 설정할 수 있습니다.

1. 포트 80의 초당 연결 수가 0 - 2000인 경우 이들 두 개의 서버를 사용
2. 포트 80의 초당 연결 수가 2000 이상인 경우 이들 10개 서버를 사용

아니면 Telnet을 사용 중일 수도 있으며, 초당 연결 수가 특정 레벨 이상으로 증가되는 경우를 제외하고는 Telnet용으로 5개 서버 중 두 개를 예약하고자 할 수 있습니다. 이 경우, Dispatcher는 피크 시간에 5개의 서버 모두에서 로드 밸런스를 수행합니다.

"연결" 유형 규칙과 함께 "upserversonrule" 규칙 평가 옵션 설정: 연결 유형 규칙을 사용하고 **upserversonrule** 옵션을 설정하면 서버 세트 중 일부의 작동이 중지되어도 나머지 서버가 과부하되지 않음을 보장할 수 있습니다. 235 페이지의 『규칙에 대한 서버 평가 옵션』에서 자세한 정보를 참조하십시오.

총 활성 연결에 따라 규칙 사용

이 규칙 유형은 Dispatcher 또는 CBR 컴포넌트에서 사용할 수 있습니다.

주: 다음을 작업하기 위해서는 관리자가 실행되고 있어야 합니다.

서버에 과부하가 걸리고 패킷 전달을 시작하는 경우, 하나의 포트의 총 작동 중인 연결 수에 따라 규칙을 사용하려고 할 수 있습니다. 특정 웹 서버는 요청에 응답할 만큼 충분한 스레드가 없는 경우에도 연결을 계속해서 승인합니다. 결과적으로, 클라이언트 요청은 제한시간이 초과되고 웹 사이트 고객은 서비스를 받지 못합니다. 서버 풀 내에서 용량의 밸런스를 조정하기 위해 활성 연결 수에 따라 규칙을 사용할 수 있습니다.

예를 들어, 서버가 250개의 연결을 승인한 후 서비스를 중단한 적이 있습니다. 사용자는 **dscontrol rule** 또는 **cbrcontrol rule** 명령을 사용하여 규칙을 작성할 수 있습니다. 예를 들면, 다음과 같습니다.

```
dscontrol rule add 130.40.52.153:80:pool2 type active  
beginrange 250 endrange 500
```

또는

```
cbrcontrol rule add 130.40.52.153:80:pool2 type active  
beginrange 250 endrange 500
```

그런 다음 다른 처리를 위해 달리 사용될 몇몇 추가 서버와 함께 현재 서버를 규칙에 추가합니다.

예약된 대역폭 및 공유 대역폭에 따라 규칙 사용

예약된 대역폭 및 공유 대역폭 규칙은 Dispatcher 컴포넌트의 경우에만 사용할 수 있습니다.

대역폭 규칙의 경우, Dispatcher가 특정 서버 세트에서 데이터를 클라이언트로 전달하는 비율로 대역폭을 계산합니다. Dispatcher는 서버, 규칙, 포트, 클러스터 및 실행 프로그램 레벨에서 용량을 추적합니다. 각 레벨의 경우, 바이트 카운터 필드(초당 전송 킬로바이트)이 있습니다. Dispatcher는 60초 간격으로 이런 비율을 계산합니다. 이러한 비율 값은 GUI 또는 명령행 보고서의 출력에서 볼 수 있습니다.

예약된 대역폭 규칙

예약된 대역폭 규칙을 사용하면 서버 세트에서 전달 중인 초당 킬로바이트 수를 제어할 수 있습니다. 구성 전체의 각 서버 세트에 임계치(지정된 대역폭 범위 할당)를 설정하면, 각 클러스터-포트 조합에서 사용 중인 대역폭의 양을 조정하고 보장할 수 있습니다.

다음은 reservedbandwidth 규칙 추가에 대한 예제입니다.

```
dscontrol rule add 9.67.131.153:80:rbw type reservedbandwidth  
beginrange 0 endrange 300
```

시작 및 종료 범위는 초당 킬로바이트로 지정됩니다.

공유 대역폭 규칙

공유 대역폭 규칙을 구성하기 전에 sharedbandwidth 옵션이 있는 **dscontrol executor** 또는 **dscontrol cluster** 명령을 사용하여 실행 프로그램 또는 클러스터 레벨에서 공유할 수 있는 대역폭의 최대량(초당 킬로바이트)을 지정해야 합니다. sharebandwidth 값은 사용 가능한 총 대역폭(총 네트워크 용량)을 초과해서는 안 됩니다. **dscontrol** 명령을 사용하여 공유 대역폭을 설정하면 규칙에 대한 상한만 제공됩니다.

다음은 명령 구문의 예제입니다.

```
dscontrol executor set sharedbandwidth size  
dscontrol cluster [add | set] 9.12.32.9 sharedbandwidth size
```

sharedbandwidth의 크기는 정수값(초당 킬로바이트)입니다. 기본값은 0입니다. 값이 0이면 대역폭을 공유할 수 없습니다.

클러스터 레벨에서 대역폭을 공유하면 클러스터에 의해 사용될 지정된 최대 대역폭을 공유할 수 있습니다. 클러스터가 사용하는 대역폭이 지정된 양 이하인 중에는 이 규칙은 참으로 평가됩니다. 사용되는 대역폭 총계가 지정된 양보다 클 경우, 이 규칙은 거짓으로 평가됩니다.

실행 프로그램 레벨에서 대역폭을 공유하면 전체 Dispatcher 구성은 최대 대역폭을 공유할 수 있습니다. 실행 프로그램 레벨에서 사용되는 대역폭이 지정된 양 이하인 중에는 이 규칙은 참으로 평가됩니다. 사용되는 대역폭 총계가 정의된 것보다 클 경우, 이 규칙은 거짓으로 평가됩니다.

다음은 sharedbandwidth 규칙 추가 또는 설정의 예제입니다.

```
dscontrol rule add 9.20.30.4:80:shbw type sharedbandwidth sharelevel value  
dscontrol rule set 9.20.34.11:80:shrul sharelevel value
```

sharelevel의 값은 실행 프로그램이나 클러스터입니다. Sharelevel은 sharebandwidth 규칙의 필수 매개변수입니다.

예약 및 공유된 대역폭 규칙 사용

Dispatcher에서는 예약된 대역폭 규칙을 사용하여 사용자의 구성 내의 서버 세트에 지정된 대역폭을 할당할 수 있습니다. 시작 및 종료 범위를 지정하여, 서버 세트가 클라이언트에 전달하는 킬로바이트의 범위를 제어할 수 있습니다. 규칙이 더 이상 참으로 평가되지 않으면(종료 범위가 초과되면), 다음으로 낮은 우선순위 규칙이 평가됩니다. 다음으로 낮은 우선순위 규칙이 "항상 참"인 규칙일 경우, "사이트 사용 중" 응답으로 클라이언트에 응답하도록 서버가 선택될 수 있습니다.

예: 포트 2222의 세 서버 그룹이 있다고 가정합니다. 예약된 대역폭이 300으로 설정될 경우, 60초 동안 초당 최대 킬로바이트는 300이 됩니다. 이 비율이 초과되면, 규칙은 더 이상 참으로 평가되지 않습니다. 이 규칙이 유일한 규칙인 경우, Dispatcher가 요청을 처리하도록 세 서버 중 하나를 선택합니다. 낮은 우선순위의 "항상 참"인 규칙이 있을 경우, 요청은 다른 서버로 경로 재지정되고 "사이트 사용 중"으로 응답될 수 있습니다.

공유 대역폭 규칙은 클라이언트에 대한 추가 서버 액세스를 제공할 수 있습니다. 특히, 예약된 대역폭 규칙 뒤에 더 낮은 우선순위 규칙을 사용할 경우, 클라이언트는 예약된 대역폭이 초과된 경우라도 서버에 계속 액세스할 수 있습니다.

예: 예약된 대역폭 규칙 뒤에 공유 대역폭 규칙을 사용하여, 클라이언트가 제어된 방식으로 세 서버에 대한 액세스 권한을 획득할 수 있도록 할 수 있습니다. 사용될 공유 대역폭이 있는 한, 규칙은 참으로 평가되고 액세스가 허용됩니다. 사용 가능한 대역폭이 없을 경우, 규칙은 참이 아니며 다음 규칙이 평가됩니다. "항상 참"인 규칙이 뒤에 올 경우, 요청은 필요에 따라 경로 재지정될 수 있습니다.

앞의 예에서 설명된 대로 예약된 및 공유 대역폭을 모두 사용하여, 서버에 대한 액세스 권한 부여(또는 거부)에서 보다 큰 유연성 및 제어를 실현할 수 있습니다. 특정 포트의 서버는 대역폭 사용이 제한되는 반면, 기타 서버는 사용 가능한 대로 추가 대역폭을 사용할 수 있습니다.

주: Dispatcher가 서버로 플로우되는 "acks" 데이터와 같은 클라이언트 통신량을 측정하여 대역폭을 추적합니다. 어떤 이유로 이 통신량을 Dispatcher가 "인식할" 수 없을 경우, 대역폭 규칙 사용 시 결과를 예측할 수 없습니다.

Metric all 규칙

이 규칙 유형은 Site Selector 컴포넌트에서만 사용할 수 있습니다.

metric all 규칙의 경우, 시스템 메트릭(cpuload, memload 또는 사용자 고유의 사용자 정의 시스템 메트릭 스크립트)을 선택한 다음, Site Selector가 시스템 메트릭 값(각 로드 밸런스 서버에 있는 Metric Server 에이전트가 리턴한 값)을 규칙에 지정한 시작 및 종료 범위와 비교합니다. 서버 세트에 있는 모든 서버의 현재 시스템 메트릭 값은 규칙이 실행되는 범위 내에 있어야 합니다.

주: 선택한 시스템 메트릭 스크립트는 각 로드 밸런스 서버에 있어야 합니다.

다음은 metric all 규칙을 구성에 추가하는 예제입니다.

```
sscontrol rule add dnsload.com:allrule1 type metricall
metricname cpuload beginrange 0 endrange 100
```

Metric average 규칙

이 규칙 유형은 Site Selector 컴포넌트에서만 사용할 수 있습니다.

metric average 규칙의 경우, 시스템 메트릭(cpuload, memload 또는 사용자 고유의 사용자 정의 시스템 메트릭 스크립트)을 선택한 다음, Site Selector가 시스템 메트릭 값(각 로드 밸런스 서버에 있는 Metric Server 에이전트가 리턴한 값)을 규칙에서 지정한 시작 및 종료 범위와 비교합니다. 서버 세트에 있는 모든 서버의 현재 시스템 메트릭 평균 값은 규칙이 실행되는 범위 내에 있어야 합니다.

주: 선택한 시스템 메트릭 스크립트는 각 로드 밸런스 서버에 있어야 합니다.

다음은 metric average 규칙을 구성에 추가하는 예제입니다.

```
sscontrol rule add dnsload.com:avgrule1 type metricavg
metricname cpuload beginrange 0 endrange 100
```

항상 참인 규칙 사용

이 규칙 유형은 Dispatcher, CBR 또는 Site Selector 컴포넌트에서 사용할 수 있습니다.

“항상 참”인 규칙을 작성할 수 있습니다. 이러한 규칙은 규칙과 관련된 모든 서버가 단절되지 않는 한 항상 선택됩니다. 이러한 이유로, 이는 대부분 다른 규칙보다 낮은 우선순위를 가집니다.

각각이 그와 관련된 서버 세트를 갖는 “항상 참” 규칙을 가질 수도 있습니다. 사용 가능한 서버가 있는 첫 번째 참 규칙이 선택됩니다. 예를 들어, 6개의 서버가 있다고 가정합시다. 이들 서버 중 두 개는 둘 다 단절되지 않는 한, 모든 환경에서 통신량을 처리하는 데 필요할 수 있습니다. 처음 두 개의 서버가 단절되면, 통신량을 처리할 두 번째 서버 세트가 필요합니다. 네 개의 서버 모두가 단절되면, 마지막 두 개의 서버를 사용하여 통신량을 처리합니다. 최대 세 개의 “항상 참” 규칙을 설정할 수 있습니다. 최소한 하나의 서버가 시동되는 한 항상 첫 번째 서버 세트가 선택됩니다. 둘 모두 단절되면, 두 번째 세트 중 하나가 선택되는 방식으로 진행됩니다.

또 다른 예제로서 “항상 참” 규칙은 수신 클라이언트가 설정된 규칙에 맞지 않을 경우, 이들이 제공되지 않는지 확인하는 데 필요할 수 있습니다. 사용자는 **dscontrol rule** 명령을 사용하여 규칙을 작성합니다.

```
dscontrol rule add 130.40.52.153:80:jamaais type true priority 100
```

클라이언트 패킷이 응답 없이 삭제되므로 서버를 규칙에 추가하지 않습니다.

주: 항상 참 규칙을 작성할 경우에는 시작 범위나 종료 범위를 설정할 필요가 없습니다.

둘 이상의 “항상 참” 규칙을 정의한 후 그 우선순위 레벨을 변경함으로써 하나의 규칙이 실행되도록 조정할 수 있습니다.

요청 콘텐츠에 따라 규칙 사용

이 규칙 유형은 CBR 컴포넌트 또는 Dispatcher 컴포넌트에서(Dispatcher의 cbr 전달 메소드 사용 시) 사용 가능합니다.

사용자는 콘텐츠 유형 규칙을 사용하여 사용자 사이트 통신의 일부 서브세트를 처리하도록 특별히 설정된 서버 세트에 요청을 전송하려고 할 수 있습니다. 예를 들어, 한 서버 세트는 모든 *cgi-bin* 요청을 처리하고, 다른 세트는 모든 스트리밍 오디오 요청을 처리하며 또다른 세트는 기타 모든 요청을 처리하도록 설정할 수 있습니다. 이 경우, *cgi-bin* 디렉토리의 경로와 일치하는 패턴을 가진 한 규칙을 추가하고, 스트리밍 오디오 파일의 파일 형식과 일치하는 다른 규칙을 추가한 다음, 나머지 통신량을 처리하는 항상 참인 또 다른 규칙을 추가하면 됩니다. 그런 다음 각 규칙에 적절한 서버를 추가하십시오.

중요: 콘텐츠 규칙 및 콘텐츠 규칙의 유효한 패턴 구문을 사용하는 방법에 대한 예제 및 시나리오는 507 페이지의 부록 B 『콘텐츠 규칙(패턴) 구문』을 참조하십시오.

포트 연관 관계 무시

포트 연관 관계 무시 기능을 사용하면 특정 서버에 대한 포트의 결합을 무시할 수 있습니다. 예를 들어, 각 Application Server에 대한 연결 수를 제한하는 규칙을 사용하고 있으며 해당 응용프로그램에 대해 “나중에 다시 시도하십시오.”라고 하는 규칙이 항상 있는 오버플로우 서버를 가지고 있다고 합시다. 이 포트는 25분의 stickytime 값을 가지므로 클라이언트가 해당 서버에 연결 상태를 유지할 수 있습니다. 포트 연관 관계 무시 기능을 사용하여 오버플로우 서버가 해당 포트와 연관된 연관 관계를 무시하도록 변경할 수 있습니다. 클라이언트가 클러스터를 요청하는 다음 번에 오버플로우 서버가 아니라 사용할 수 있는 Application Server로 로드 밸런스가 진행됩니다.

서버 **sticky** 옵션을 사용한 포트 연관 관계 덮어쓰기의 명령 구문에 대한 자세한 정보는 418 페이지의 『dscontrol server — 서버 구성』을 참조하십시오.

사용자 구성에 규칙 추가

예제 구성 파일을 편집하거나 GUI(Graphical User Interface)로 **dscontrol rule add** 명령을 사용하여 규칙을 추가할 수 있습니다. 하나 이상의 규칙을 정의한 모든 포트에 추가할 수 있습니다.

이것은 2단계 프로세스로서 규칙을 추가한 다음 규칙이 참인 경우, 서비스할 서버를 정의합니다. 예를 들어, 시스템 관리자는 사이트의 각 부서에서 가져온 프록시 서버를 열

마나 많이 사용하는지를 추적하고자 합니다. IP 주소는 각 부서에 제공됩니다. 먼저 클라이언트 IP 주소에 따라 첫 번째 규칙 세트를 작성하여 각 부서의 로드를 구분하십시오.

```
dscontrol rule add 130.40.52.153:80:div1 type ip b 9.1.0.0 e 9.1.255.255
dscontrol rule add 130.40.52.153:80:div2 type ip b 9.2.0.0 e 9.2.255.255
dscontrol rule add 130.40.52.153:80:div3 type ip b 9.3.0.0 e 9.3.255.255
```

그 다음, 다른 서버를 각 규칙에 추가한 후 사용 중인 서비스에 적절하게 청구서를 부서에 보내도록 서버 각각의 로드를 측정하십시오. 예를 들어,

```
dscontrol rule useserver 130.40.52.153:80:div1 207.72.33.45
dscontrol rule useserver 130.40.52.153:80:div2 207.72.33.63
dscontrol rule useserver 130.40.52.153:80:div3 207.72.33.47
```

규칙에 대한 서버 평가 옵션

서버 평가 옵션은 Dispatcher 컴포넌트에서만 사용할 수 있습니다.

dscontrol rule 명령에는 규칙에 대한 서버 평가 옵션이 있습니다. 평가 옵션을 사용하여 포트에 있는 모든 서버에서 규칙의 조건을 평가하거나 규칙 내의 서버에서만 규칙의 조건을 평가하십시오. (Load Balancer의 이전 버전에서는 포트에 있는 모든 서버에서 각 규칙의 조건을 측정만 가능했습니다.)

주:

1. 서버 평가 옵션은 서버의 특성에 따라 결정하는 규칙(조당 총 연결 규칙, 활성 연결 규칙 및 예약된 대역폭 규칙)에만 유효합니다.
2. "연결" 유형 규칙에는 **upserversonrule**을 선택할 수 있는 추가 평가 옵션이 있습니다. 229 페이지의 『조당 연결에 따라 규칙 사용』에서 자세한 정보를 참조하십시오.

다음은 예약된 대역폭 규칙에서 평가 옵션을 추가 또는 설정하는 예제입니다.

```
dscontrol rule add 9.22.21.3:80:rbweval type reservedbandwidth evaluate level
dscontrol rule set 9.22.21.3:80:rbweval evaluate level
```

평가 레벨은 포트, 규칙 또는 **upserversonrule**로 설정할 수 있습니다. 기본값은 포트입니다.

규칙 내의 서버 평가

규칙 내의 서버에서 규칙의 조건을 측정하는 옵션을 사용하면 다음의 특성을 가진 두 개의 규칙을 구성할 수 있습니다.

- 첫 번째로 평가되는 규칙에는 웹 사이트 콘텐츠를 유지하는 모든 서버가 포함되며, 평가 옵션은 **규칙**으로 설정됩니다(규칙 내의 서버에서 규칙의 조건 평가).
- 두 번째 규칙은 항상 참 규칙이며, "사이트 사용 중" 유형의 응답을 하는 단일 서버가 포함됩니다.

결과적으로 통신량이 첫 번째 규칙 내의 서버 임계치를 초과하는 경우, 통신이 두 번째 규칙 내의 "사이트 사용 중" 서버로 전송됩니다. 통신량이 첫 번째 규칙 내의 서버 임계치 이하이면 새 통신이 첫 번째 규칙의 서버로 다시 전송됩니다.

포트의 서버 평가

이전 예제에서 설명한 두 가지 규칙을 사용할 때, 평가 옵션을 첫 번째 규칙의 포트로 설정한 경우(포트의 모든 서버에서 규칙의 조건 평가) 통신량이 해당 규칙의 임계치를 초과하면 통신이 두 번째 규칙과 연관된 "사이트 사용 중" 서버로 전송됩니다.

첫 번째 규칙은 포트의 모든 서버 통신량("사이트 사용 중" 서버 포함)을 측정하여 통신량이 임계치를 초과하는지 여부를 판별합니다. 첫 번째 규칙과 연관된 서버의 통신량 정체가 감소할 때, 포트의 통신량이 계속 첫 번째 규칙의 임계치를 초과하기 때문에 통신이 계속해서 "사이트 사용 중" 서버로 전송되는 위치에 의도하지 않은 결과가 발생할 수 있습니다.

Load Balancer에 대한 연관 관계 기능 사용법

Dispatcher 및 CBR 컴포넌트: 연관 관계 기능은 사용자가 클러스터 포트를 결합(sticky)으로 구성할 때 사용 가능합니다. 클러스터 포트를 결합으로 구성하면 클라이언트의 연속적인 요구가 동일한 서버로 지정됩니다. 이것은 결합 시간을 실행 프로그램, 클러스터 또는 포트 레벨에서 초 수로 설정하면 이루어집니다. 이 연관 관계 기능은 결합 시간을 0으로 설정하면 해제됩니다.

크로스 포트 연관 관계가 사용 가능하면, 공유된 포트의 결합 값은 같은 값(0이 아닌)이어야 합니다. 237 페이지의 『포트간 연관 관계』에서 자세한 정보를 참조하십시오.

Site Selector 컴포넌트: 연관 관계 기능은 사용자가 사이트 이름을 결합(sticky)으로 구성할 때 사용 가능합니다. 사이트 이름을 결합되도록 구성하면 클라이언트는 여러 이름 서비스 요청에 대해 동일한 서버를 사용할 수 있습니다. 이것은 결합 시간을 사이트 이름에서 초 수로 설정하면 이루어집니다. 이 연관 관계 기능은 결합 시간을 0으로 설정하면 해제됩니다.

서버의 결합 시간 값은 한 연결의 닫기와 첫 번째 연결 중에 사용되는 동일한 서버로 다시 클라이언트가 전송되는 시간 동안 새로운 연결이 열리는 사이의 간격입니다. 결합 시간이 만기된 이후에 클라이언트는 첫 번째 서버와 다른 서버로 전송될 수도 있습니다. 서버의 결합 시간 값이 dscontrol executor, port 또는 cluster 명령을 사용하여 구성됩니다. 서버 오프라인을 위해 서버 down 명령을 사용하는 경우(dscontrol server down), 해당 서버에 대한 결합 시간 값이 0이 아니면 기존 클라이언트는 결합 시간이 만료될 때까지 기존의 클라이언트에 서비스를 제공합니다. 서버는 결합 시간 값이 만료될 때까지 단절되지 않습니다.

연관 관계가 사용 불가능할 때의 작동

연관 관계 기능이 사용 가능하지 않은 상태에서 클라이언트로부터 새로운 TCP 연결을 받을 때마다, Load Balancer는 적시에 올바른 서버를 선택하여 그 패킷을 서버에 전달합니다. 연속되는 연결이 같은 클라이언트에게서 들어오면 []는 그 연속 연결을 연관되지 않은 새로운 연결로 간주하고 다시 적시에 올바른 서버를 선택합니다.

연관 관계가 사용 가능할 때의 작동

연관 관계 기능이 사용 가능한 상태에서 클라이언트가 연속된 요구를 해오면, 요구는 동일한 서버로 지정됩니다.

시간이 지남에 따라 클라이언트는 트랜잭션 전송을 끝내며 그 연관 관계 기록은 사라지게 됩니다. 따라서 결합 "시간의 의미는 다음과 같습니다." 각각의 연관 관계 기록은 수 초의 "결합 시간"을 위해 존재합니다. 결합 시간 내에 연속적인 연결이 이루어지면, 연관 관계 기록은 여전히 유효하며 클라이언트의 요구는 동일한 서버로 전해집니다. 연속적인 연결이 결합 시간 내에 이루어지지 않으면, 기록은 제거되고 결합 시간 이후에 이루어지는 연결은 새로운 서버를 선택하게 됩니다.

서버 오프라인을 위해 서버 down 명령(dscontrol server down)이 사용되고 서버는 stickytime 값이 만료될 때까지 단절되지 않습니다.

포트간 연관 관계

포트간 연관 관계는 Dispatcher 컴포넌트의 MAC 및 NAT/NATP 전달 메소드에만 적용됩니다.

포트간 연관 관계는 여러 포트에 확장될 수 있는 연관 관계 기능입니다. 예를 들어, 처음에 한 포트에 클라이언트 요청이 수신되고 다른 포트에 다음 요청이 수신되면 포트간 연관 관계를 통해 클라이언트 요청이 동일한 서버로 전송될 수 있습니다. 이 기능을 사용하려면 포트는 다음과 같아야 합니다.

- 동일한 클러스터 주소 공유
- 동일한 서버 공유
- 0이 아닌 동일한 stickytime 값 보유
- 동일한 stickymask 값 보유

둘 이상의 포트가 동일한 crossport에 링크될 수 있습니다. 후속 연결이 동일한 포트나 공유 포트의 동일한 클라이언트에서 시도되면 동일한 서버가 액세스됩니다. 다음은 포트간 연관 관계를 사용하여 포트 10으로 여러 포트를 구성하는 예제입니다.

```
dscontrol port set cluster:20 crossport 10
dscontrol port set cluster:30 crossport 10
dscontrol port set cluster:40 crossport 10
```

포트간 연관 관계가 설정되면 포트에 대한 stickytime 값을 얼마든지 수정할 수 있습니다. 그러나 모든 공유 포트의 stickytime 값을 동일한 값으로 변경하는 것이 좋습니다. 이렇게 하지 않으면 예상치 못한 결과가 발생할 수 있습니다.

포트간 연관 관계를 제거하려면 crossport 값을 다시 자신의 포트 번호로 설정하십시오. **crossport** 옵션에 대한 명령 구문에 대한 자세한 정보는 404 페이지의 『dscontrol port — 포트 구성』을 참조하십시오.

연관 관계 주소 마스크(stickymask)

연관 관계 주소는 Dispatcher 컴포넌트에만 적용됩니다.

연관 관계 주소 마스크는 공통적인 서브넷 주소에 기반하여 클라이언트를 그룹화하는 향상된 결합 기능입니다. **dscontrol port** 명령에 **stickymask**를 지정하면 32비트 IP 주소의 공통된 상위 순서 비트를 마스크할 수 있습니다. 이 기능이 구성되면 클라이언트 요청은 먼저 **포트**로 연결을 수행하고 동일한 서브넷 주소(가려질 주소 부분으로 표시)를 가진 모든 후속 클라이언트 요청이 동일한 서버로 보내집니다.

주: stickymask를 사용하려면 포트 결합 시간이 0이 아닌 값이어야 합니다.

예를 들어, 동일한 네트워크 클래스 A 주소를 사용하는 모든 수신 클라이언트 요청이 동일한 서버로 전송되게 하는 경우, 해당 포트에 대해 stickymask 값을 8(비트)로 설정해야 합니다. 동일한 네트워크 클래스 B 주소를 사용하는 클라이언트 요청을 그룹화하려면 stickymask 값을 16(비트)으로 설정하십시오. 동일한 네트워크 클래스 C 주소를 사용하여 클라이언트 요청을 그룹화하려면 stickymask 값을 24(비트)로 설정하십시오.

최상의 결과를 얻으려면 []를 처음 시작할 때 stickymask 값을 설정하십시오. stickymask 값을 동적으로 변경하면 결과를 예측할 수 없게 됩니다.

포트간 연관 관계를 통한 대화: 포트간 연관 관계를 사용 가능으로 설정하는 경우 공유 포트의 stickymask 값들은 동일해야 합니다. 237 페이지의 『포트간 연관 관계』에서 자세한 정보를 참조하십시오.

연관 관계 주소 마스크를 사용 가능으로 설정하려면 다음과 유사하게 **dscontrol port** 명령을 발행하십시오.

```
dscontrol port set cluster:port stickytime 10 stickymask 8
```

가능한 stickymask 값은 8, 16, 24 및 32입니다. 값 8은 IP 주소(네트워크 클래스 A 주소)의 처음 8개의 상위 순서 비트가 가려지도록 지정합니다. 값 16은 IP 주소(네트워크 클래스 B 주소)의 처음 16개의 상위 순서 비트가 가려지도록 지정합니다. 값 24는 IP 주소(네트워크 클래스 C 주소)의 처음 24개의 상위 순서 비트가 가려지도록 지정합니다. 값 32를 지정하면 연관 관계 주소 마스크 기능을 효과적으로 사용 불가능하게 할 수 있도록 전체 IP 주소를 가리게 됩니다. stickymask의 기본값은 32입니다.

stickymask(연관 관계 주소 마스크 기능)의 명령 구문에 대한 자세한 정보는 404 페이지의 『dscontrol port — 포트 구성』을 참조하십시오.

서버 연결 처리 작업중지

처리 작업중지는 Dispatcher 및 CBR 컴포넌트에 적용됩니다.

어떠한 이유로든(갱신, 업그레이드, 서비스 등) Load Balancer 구성에서 서버를 제거하려면 **dscontrol manager quiesce** 명령을 사용할 수 있습니다. 작업중지 하위명령으로 기존의 연결이 완료될 수 있으며(서버에 전달되지 않고), 연결이 결합으로 지정되고 결합 시간이 만기되지 않았으면 이후의 새 연결만 클라이언트에서 작업중지 서버로 전달됩니다. 작업중지 하위 명령으로 다른 새 연결은 서버에 허용되지 않습니다.

결합 연결 처리 작업중지

결합 시간이 설정되어 있고 결합 시간이 만기되기 전에 새 연결을 다른 서버(작업중지된 서버 대신)로 전송하려면 “지금” 작업중지 옵션을 사용하십시오. 다음은 서버 9.40.25.67을 작업중지하기 위해 지금 옵션을 사용하는 예제입니다.

```
dscontrol manager quiesce 9.40.25.67 now
```

지금 옵션은 다음과 같이 결합 연결이 처리되는 방법을 결정합니다.

- “지금”을 지정하지 않은 경우, 기존의 연결을 완료할 수 있으며, 작업중지된 서버가 결합 시간이 만기되기 전에 새 요청을 받는 동안은 이후의 새 연결을 결합으로 지정된 기존의 연결이 있는 클라이언트에서 작업중지된 서버로 전달합니다(그러나 결합(연관 관계) 기능이 사용 불가능하면 작업중지된 서버가 새 연결을 받을 수 없습니다).

이것은 보다 점진적이며 덜 갑작스러운 서버 작업중지 방법입니다. 예를 들어, 서버를 점진적으로 작업중지한 다음, 통신량이 가장 적은 때(이른 아침)를 기다려서 구성에서 서버를 완전히 제거할 수 있습니다.

- “지금”을 지정하면, 서버를 작업중지하여 서버가 기존의 연결을 완료할 수 있지만 결합으로 지정된 기존의 연결이 있는 클라이언트로부터의 이후의 새 연결을 포함하는 모든 새 연결은 허용하지 않습니다. 이것은 덜 갑작스러운 서버 작업중지 방법으로 이전 버전의 Load Balancer에서는 서버를 처리하는 유일한 방법입니다.

클라이언트 요청의 콘텐츠에 기본적인 규칙에 대한 연관 관계 옵션

dscontrol rule 명령에서 다음과 같은 연관 관계 유형을 지정할 수 있습니다.

- 활성 쿠키 — Load Balancer가 생성한 쿠키를 기준으로 동일한 서버로 연관 관계를 사용하여 웹 통신량을 로드 밸런싱할 수 있습니다.

활성 쿠키 연관 관계는 CBR 컴포넌트에만 적용됩니다.

- 수동 쿠키 — 서버가 생성한 자체 식별 쿠키를 기반으로 동일한 서버로 웹 통신량을 로드 밸런스할 수 있습니다. 또한 수동 쿠키 연관 관계와 관련하여 규칙 명령에서 쿠키 이름 매개변수를 지정해야 합니다.

수동 쿠키는 CBR 컴포넌트와 Dispatcher 컴포넌트의 crb 전달 메소드에 적용됩니다.

- URI — 용량을 효율적으로 증대시키는 방법으로 Caching Proxy 서버로 웹 통신량을 로드 밸런스할 수 있습니다.

URI 연관 관계는 CBR 컴포넌트와 Dispatcher 컴포넌트의 crb 전달 메소드에 적용됩니다.

연관 관계 옵션에 대한 기본값은 "없음"입니다. 규칙 명령에서 연관 관계 옵션을 활성화 쿠키, 수동 쿠키 및 URI로 설정하려면 포트 명령에서 **stickytime** 옵션이 0(사용 불가능)이 되어야 합니다. 규칙에서 연관 관계가 설정되면 포트에서 stickytime을 사용할 수 없습니다.

활성 쿠키 연관 관계

활성 쿠키 연관 관계 기능은 CBR 컴포넌트에만 적용됩니다.

쿠키 연관 관계 기능은 특정 서버에 클라이언트를 “결합하는” 방법을 제공합니다. 이 기능은 규칙의 결합 시간을 양수로 설정하고 연관 관계를 “activecookie”로 설정하면 사용 가능하게 됩니다. 규칙을 추가하거나 규칙 세트 명령을 사용할 때 이 기능이 수행됩니다. 명령 구문에 대한 자세한 정보는 411 페이지의 『dscontrol rule — 규칙 구성』을 참조하십시오.

활성 쿠키 연관 관계에 대해 규칙이 사용 가능하게 설정되면 새 클라이언트 요청이 표준 CBR 알고리즘을 사용하여 로드 밸런스되며, 동일한 클라이언트의 후속 요청은 처음 선택된 서버로 전송됩니다. 선택된 서버는 클라이언트로 보내는 응답에 쿠키로 저장됩니다. 향후 클라이언트 요청에 쿠키가 포함되어 있고 각 요청이 결합 시간 내에 도착하는 한, 클라이언트는 처음 서버와의 연관 관계를 유지합니다.

활성 쿠키 연관 관계는 클라이언트가 일정한 시간 동안 계속 동일한 서버로 로드 밸런스되도록 보장합니다. 이는 클라이언트 브라우저에서 저장된 쿠키를 전송하여 수행됩니다. 쿠키에는 결정할 때 사용된 cluster:port:rule, 로드 밸런스된 서버, 연관 관계가 무효화되는 제한시간 시간 소인이 포함됩니다. 쿠키의 형식은 다음과 같습니다. **IBMCCR=클러스터:포트:규칙+서버-시간!** CBR 구성이 드러나지 않도록 cluster:port:rule 및 server 정보가 인코드됩니다.

활성 쿠키 연관 관계의 작동 방법

활성 쿠키 연관 관계가 설정된 규칙을 사용할 때마다 클라이언트가 보낸 쿠키가 검사됩니다.

- 사용된 cluster:port:rule에 대한 ID를 포함하는 쿠키가 발견되면 로드 밸런스된 서버 및 시간 소인 만기가 쿠키에서 추출됩니다.
- 규칙에서 사용한 세트에 서버가 계속 있고 가중치가 양수(positive)이거나 작업중지된 서버이고 시간 소인 만기가 지금 이후이면 쿠키 내의 서버가 로드 밸런스될 서버로 선택됩니다.
- 이전 글머리표 중 만족하지 않는 조건이 있으면 표준 알고리즘을 사용하여 서버를 선택합니다.
- 두 가지 방법 중 하나를 사용하여 서버를 선택하면 IBMCBR, cluster:port:rule, server_chosen 정보 및 시간 소인이 포함된 새 쿠키가 구성됩니다. 시간 소인은 연관 관계가 만기되는 시간입니다. “cluster:port:rule 및 server_chosen”은 CBR 구성 정보가 드러나지 않도록 인코드됩니다.
- “expires” 매개변수도 쿠키에 삽입됩니다. 이 매개변수는 브라우저가 이해할 수 있는 형식이며, 시간 소인 만기부터 7일후 쿠키가 유효하지 않게 합니다. 따라서 클라이언트 쿠키 데이터베이스는 정돈되어 있습니다.

그리고 나서, 새 쿠키는 클라이언트로 돌아가는 헤더에 삽입되며, 클라이언트 브라우저가 쿠키를 승인하도록 구성된 경우, 후속 요청을 다시 전송합니다.

쿠키의 각 연관 관계 인스턴스는 길이가 65바이트이고 느낌표로 끝납니다. 그 결과, 4096바이트 쿠키는 도메인 당 대략 60개의 개별 활성 쿠키 규칙을 보유할 수 있습니다. 쿠키가 완전히 채워지면 모든 만기 연관 관계 인스턴스가 제거됩니다. 모든 인스턴스가 여전히 유효할 경우, 가장 오래된 인스턴스가 삭제되며 현재 규칙에 대한 새 인스턴스가 추가됩니다.

주: 이전 형식의 IBMCBR 쿠키가 프록시에 나타나면 CBR이 바꿉니다.

포트 결합 시간이 0인 경우(사용 불가능), 규칙 명령에 대한 활성 쿠키 연관 관계 옵션은 activecookie로만 설정할 수 있습니다. 활성 쿠키 연관 관계가 규칙에 활성화되어 있으면 포트에서 결합 시간은 사용 불가능합니다.

활성 쿠키 연관 관계를 사용 가능하게 하는 방법

특정 규칙에 대한 활성 쿠키 연관 관계를 사용 가능하게 하려면 다음과 같은 규칙 세트 명령을 사용합니다.

```
rule set cluster:port:rule stickytime 60
rule set cluster:port:rule affinity activecookie
```

활성 쿠키 연관 관계를 사용하는 이유

규칙 결합은 일반적으로 서버의 클라이언트 상태를 저장하는 CGI 또는 servlets에 대해 사용합니다. 쿠키 ID는 상태를 식별합니다(이는 서버 쿠키임). 클라이언트 상태는 선택된 서버에만 있으므로 클라이언트는 요청 사이에서 해당 상태를 유지하기 위해 해당 서버의 쿠키를 필요로 합니다.

활성 쿠키 연관 관계 만기 시간 대체

활성 쿠키 연관 관계는 현재 서버 시간, 더하기 결합 시간 간격, 더하기 24시의 기본 만기 시간을 사용합니다. 클라이언트(요청을 CBR 시스템에 전송 중인 시스템)의 시간이 해당 시스템에서 부정확할 경우(예를 들어, 서버 시간보다 하루 빠를 경우), 클라이언트 시스템이 쿠키가 이미 만기되었다고 가정하므로 CBR의 쿠키를 무시합니다. 보다 긴 만기 시간을 설정하려면, cbrserver 스크립트를 수정하십시오. 스크립트 파일에서, javaw 행을 편집하여 LB_SERVER_KEYS 뒤에 다음과 같은 매개변수를 추가하십시오. -DCOOKIEEXPIREINTERVAL=X 여기서 X는 만기 시간에 추가할 일 수입니다.

AIX, Solaris 및 Linux 시스템에서 cbrserver 파일은 /usr/bin 디렉토리에 있습니다.

Windows 시스템에서 cbrserver 파일은 \winnt\system32 디렉토리에 있습니다.

수동 쿠키 연관 관계

수동 쿠키 연관 관계는 Dispatcher 컴포넌트의 Content Based Routing(CBR) 전달 메소드 및 CBR 컴포넌트에 적용됩니다. Dispatcher의 cbr 전달 메소드 구성에 대한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』을 참조하십시오.

수동 쿠키 연관 관계는 특정 서버에 클라이언트를 결합시키는 방법을 제공합니다. 규칙의 연관 관계를 "passivecookie"로 사용 가능하게 하면 수동 쿠키 연관 관계를 사용하여 서버에서 생성한 자가 식별 쿠키에 따라 동일한 서버에 대한 연관 관계를 통해 웹 통신량의 로드 밸런싱을 수행할 수 있습니다. 규칙 레벨에서 수동 쿠키 연관 관계를 구성할 수 있습니다.

규칙이 실행된 후, 수동 쿠키 연관 관계를 사용 가능하게 하면 Load Balancer에서 클라이언트 요청의 HTTP 헤더에 있는 쿠키 이름에 따라 서버를 선택합니다. Load Balancer에서 클라이언트 HTTP의 쿠키 이름을 각 서버에 구성된 쿠키 값과 비교하기 시작합니다.

Load Balancer에서 쿠키 값에 클라이언트의 쿠키 이름이 포함된 서버를 처음 찾았을 때 []는 요청에 대해 해당 서버를 선택합니다.

주: 변수 파트가 추가된 정적 파트를 가진 쿠키 값을 서버에서 생성하는 경우를 처리하기 위해 Load Balancer에서 이 용통성을 제공합니다. 예를 들어, 서버의 쿠키 값이 시간 소인(변수 값)이 추가된 서버 이름(정적 값)일 수 있습니다.

클라이언트 요청의 쿠키 값이 없거나 서버의 쿠키 값 내용과 일치하지 않을 경우 기존 서버 선택 방법이나 가중치 라운드 로빙 기술을 사용하여 서버를 선택합니다.

수동 쿠키 연관 관계를 구성하려면 다음을 수행하십시오.

- Dispatcher의 경우, 먼저 Dispatcher의 CBR 전달 메소드를 구성하십시오(61 페이지의『Dispatcher content-based routing(cbr 전달 메소드)』참조). 이 단계는 CBR 컴포넌트에는 생략됩니다.
- **dscontrol rule [add|set]** 명령에서 **affinity** 매개변수를 "passivecookie"로 설정합니다. 또한 **cookieName** 매개변수를 Load Balancer가 클라이언트 HTTP 헤더 요청에서 찾아야 하는 쿠키 이름으로 설정해야 합니다.
- **dscontrol server [add|set]** 명령에서 규칙 서버 세트의 각 서버에 대해 **cookieValue** 매개변수를 설정합니다.

포트 결합 시간이 0인 경우(사용 불가능), 규칙 명령에 대한 수동 쿠키 연관 관계 옵션은 passivecookie로만 설정할 수 있습니다. 수동 쿠키 연관 관계가 규칙에 활성화되어 있으면 포트에서 결합 시간은 사용 불가능합니다.

URI 연관 관계

URI 연관 관계는 Dispatcher의 CBR 전달 메소드 및 CBR 컴포넌트에 적용됩니다. CBR 전달 메소드 구성에 대한 정보는 61 페이지의『Dispatcher content-based routing(cbr 전달 메소드)』을 참조하십시오.

URI 연관 관계를 사용하면 고유한 콘텐츠를 각 서버에서 캐시할 수 있는 Caching Proxy 서버에 대한 웹 통신량의 로드 밸런스를 수행할 수 있습니다. 결과적으로 여러 시스템에서 콘텐츠의 중복 캐시를 제거하여 사이트 캐시의 용량을 효과적으로 증가시킵니다. 규칙 레벨에서 URI 연관 관계를 구성하십시오. 규칙이 실행된 후 URI 연관 관계가 사용 가능하게 되고 동일한 서버 세트가 가동하여 응답하면 Load Balancer는 동일한 URI가 있는 새 수신 클라이언트 요청을 동일한 서버로 전달합니다.

일반적으로 Load Balancer는 동일한 콘텐츠를 제공하는 여러 서버로 요청을 분산시킬 수 있습니다. 캐시 서버 그룹이 있는 Load Balancer를 사용할 때 액세스된 콘텐츠가 결국 모든 서버에 캐시되는 경우가 많습니다. 이것은 동일하게 캐시된 콘텐츠를 여러 시스템에 복제하여 매우 높은 클라이언트 로드를 지원합니다. 이것은 특히 볼륨이 큰 웹 사이트에서 유용합니다.

그러나 사용자의 웹 사이트가 매우 다양한 콘텐츠에 대한 중간 정도의 클라이언트 통신량을 지원하고 사용자가 보다 큰 캐시를 여러 서버에 분포시키려는 경우, 각 캐시 서버에 고유한 콘텐츠가 포함되고 Load Balancer가 해당 콘텐츠가 있는 캐시 서버에만 요청을 분산시키면 사이트의 성능이 향상됩니다.

URI 연관 관계와 함께 Load Balancer를 사용하면 캐시된 콘텐츠를 각 서버에 분산시켜 여러 시스템에 있는 콘텐츠의 중복 캐시를 제거할 수 있습니다. Caching Proxy 서버를 사용하는 다양한 콘텐츠 서버 사이트의 성능은 이러한 향상 내용과 함께 향상됩니다. 동일한 요청을 동일한 서버에 전송하므로 단일 서버에만 있는 콘텐츠를 캐시하게 됩니다. 효과적인 캐시 크기는 각각의 새 서버 시스템이 풀에 추가됨에 따라 증가합니다.

URI 연관 관계를 구성하려면 다음을 수행하십시오.

- Dispatcher의 경우, 먼저 Dispatcher의 CBR 전달 메소드를 구성하십시오(61 페이지의『Dispatcher content-based routing(cbr 전달 메소드)』참조). 이 단계는 CBR 컴포넌트에는 생략됩니다.
- **dscontrol rule [addlset]** 또는 **cbrcontrol rule [addlset]** 명령에서 **affinity** 매개 변수를 "uri"로 설정하십시오.

포트 결합 시간이 0인 경우(사용 불가능) 규칙 명령에 대한 URI 연관 관계 옵션은 URI로만 설정될 수 있습니다. URI 연관 관계가 규칙에 활성화되어 있으면 포트에서 결합 시간은 사용 불가능합니다.

광역 Dispatcher 지원 구성

Dispatcher 컴포넌트의 경우에만 이 기능을 사용할 수 있습니다.

Dispatcher의 광역 지원과 nat 전달 메소드를 사용하지 않는 경우, Dispatcher 시스템 및 그 서버가 모두 동일한 LAN 세그먼트에 접속되는 Dispatcher 구성이 요구됩니다(그림 35 참조). 클라이언트의 요청이 Dispatcher 시스템으로 들어왔다가 서버로 보내 집니다. 서버로부터, 응답이 곧바로 클라이언트에게 보내집니다.

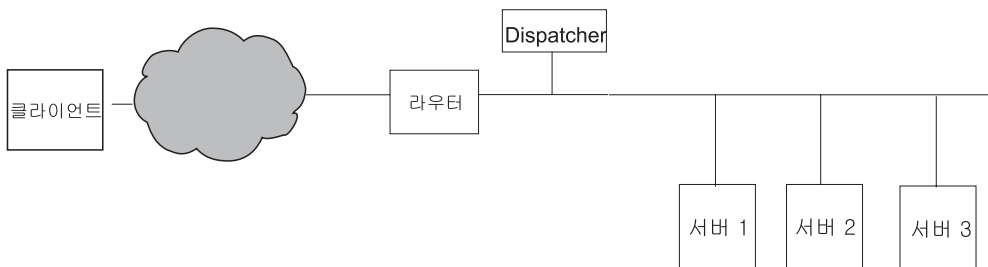


그림 35. 단일 LAN 세그먼트를 구성하는 구성 예제

광역 Dispatcher 기능은 원격 서버로 알려진 오프사이트 서버에 대한 지원을 추가합니다(245 페이지의 그림 36 참조). GRE가 원격 사이트에서 지원되지 않고 Dispatcher의 전달 메소드가 사용 중이지 않은 경우, 원격 사이트는 원격 Dispatcher 시스템(Dispatcher 2)과 이 시스템에 로컬로 접속된 서버(ServerG, ServerH 및 ServerI)로 구성되어야 합니다. 클라이언트 패킷은 인터넷에서 초기 Dispatcher 시스템으로 이동합니다. 초기 Dispatcher 시스템에서, 패킷은 지역적으로 떨어져 있는 원격 Dispatcher 시스템으로 이동한 후 그 지역 내에 접속된 서버 중 하나로 이동합니다.

모든 Dispatcher 시스템(로컬 및 원격)은 광역 구성을 실행하기 위해 동일한 유형의 운영 체제 및 플랫폼에 있어야 합니다.

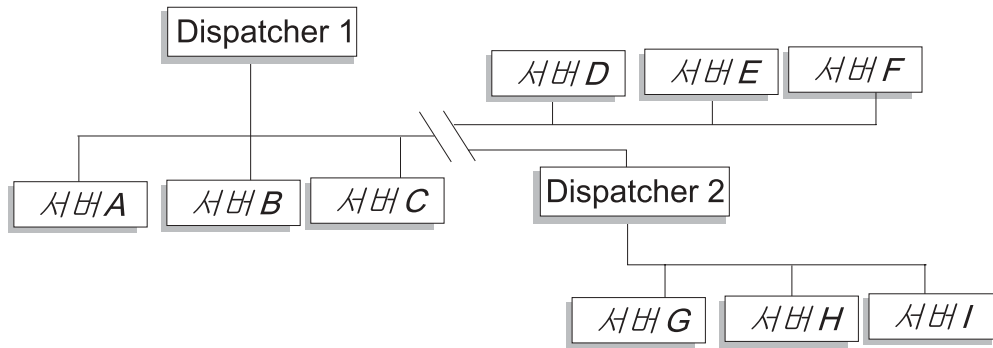


그림 36. 로컬 및 원격 서버를 사용하는 구성 예제

이 경우 하나의 클러스터 주소가 전세계의 모든 클라이언트 요청을 지원하며 로드를 전세계에 있는 서버에 분산시킬 수 있습니다.

처음에 패킷을 수신하는 Dispatcher 시스템에 여전히 접속된 로컬 서버가 있을 수 있으며 그 로컬 서버와 원격 서버간에 로드를 분산시킬 수 있습니다.

명령 구문

광역 지원을 구성하려면 다음을 수행하십시오.

1. 서버를 추가하십시오. Dispatcher에 서버를 추가할 경우, 서버가 로컬 또는 원격인지 정의해야 합니다(위 참조). 서버를 추가하고 로컬로 정의하려면, 라우터를 지정하지 않고 **dscontrol server add** 명령을 실행하십시오. 이것은 기본값입니다. 서버를 원격으로 정의하려면, 원격 서버에 도달하기 위해 Dispatcher가 패킷을 전송해야 하는 라우터를 지정해야 합니다. 서버는 다른 Dispatcher여야 하며 서버의 주소는 Dispatcher의 비전달 주소여야 합니다. 예를 들어, 248 페이지의 그림 37에서 **LB 2**를 **LB 1**의 원격 서버로 추가하는 경우 **router 1**을 라우터 주소로 정의해야 합니다. 일반 구문:

```
dscontrol server add cluster:port:server router address
```

라우터 키워드에 대한 자세한 정보는 418 페이지의 『dscontrol server — 서버 구성』을 참조하십시오.

2. 별명을 구성하십시오. 첫 번째 Dispatcher 시스템(인터넷에서 클라이언트 요청이 도착하는 위치)에서, 클러스터 주소는 **executor configure** 명령을 사용하여 별명이 지정되어야 합니다.(Linux 또는 UNIX 시스템의 경우, **executor configure** 또는 **ifconfig** 명령을 사용할 수 있습니다.) 그러나 원격 Dispatcher 시스템에서는 네트워크 인터페이스 카드에 대해 클러스터 주소의 별명이 지정되지 않습니다.

Dispatcher의 광역 지원으로 원격 어드바이저 사용

시작점 Dispatcher:

AIX, Linux(GRE 사용) 또는 Solaris 플랫폼에서 실행되는 시작점 Dispatcher는 어드바이저 로드를 빠르게 표시합니다. 다른 플랫폼은 광역 네트워크 대신 라운드 로빈 로드 밸런스 또는 Dispatcher nat/cbr 전달 방식을 사용하여 응답할 필요가 있습니다.

AIX 시스템

- 특수 구성 단계는 필요하지 않습니다.

HP-UX 시스템

- HP-UX 플랫폼에서 실행되는 시작점 Dispatcher 사용 시 WAN 구성에서, 원격 어드바이저 사용이 제한됩니다. Dispatcher의 mac 전달 메소드로, HP-UX 어드바이저는 항상 클러스터 대신 서버 주소를 대상으로 직접 설정합니다. 클러스터를 대상으로 하지 않으므로, 원격 Dispatcher는 어드바이저 요청을 원격 서버로 로드 밸런싱하지 않습니다. 그러나, 원격 어드바이저는 Dispatcher의 cbr 또는 nat 전달을 사용할 때 올바르게 작동합니다.

Linux 시스템

- Linux 플랫폼에서 실행되는 시작점 Dispatcher 사용 시 WAN 구성에서, 원격 어드바이저 사용이 제한됩니다. Dispatcher의 mac 전달 메소드로, Linux 어드바이저는 항상 클러스터 대신 서버 주소를 대상으로 직접 설정합니다. 클러스터를 대상으로 하지 않으므로, 원격 Dispatcher는 어드바이저 요청을 원격 서버로 로드 밸런싱하지 않습니다. 그러나, 원격 어드바이저는 Dispatcher의 cbr 또는 nat 전달을 사용할 때 올바르게 작동합니다.
- 구성에 원격 Dispatcher가 존재하지 않고 원격 서버에 통신량을 전송하기 위해 GRE(generic routing encapsulation)를 사용할 경우, Linux 플랫폼에서 Dispatcher의 mac, nat 또는 cbr 전달 메소드 실행 시 어드바이저 사용에 아무 제한이 없습니다. 자세한 정보는 250 페이지의 『GRE(일반 경로 지정 캡슐화) 지원』을 참조하십시오.

Solaris 시스템

- Solaris 플랫폼에서 실행되는 시작점 Dispatcher 사용 시 WAN 구성에서, ifconfig 또는 dscontrol executor 구성 방법 대신 arp 구성 방법을 사용해야 합니다. 예를 들어,

```
arp -s my_cluster_address my_mac_address pub
```
- 다음은 Solaris 플랫폼의 제한사항입니다.
 - WAN 어드바이저는 클러스터 구성의 arp 방법으로만 작동합니다.
 - 특정 바인드 서버의 어드바이저는 클러스터 구성의 arp 방법으로만 작동합니다.
 - 특정 바인드 서버의 어드바이저는 클러스터 구성의 arp 방법으로만 작동합니다. 바인드 고유 서버에 대해 어드바이저 사용 시, 바인드 고유 응용프로그램과 동일한 서버에 Load Balancer를 결합 배치하지 마십시오.

Windows 시스템

- Windows 플랫폼에서 실행되는 시작점 Dispatcher 사용 시 WAN 구성에서, 원격 어드바이저 사용이 제한됩니다. Dispatcher의 mac 전달 메소드로, Windows 어드바이저는 항상 클러스터 대신 서버 주소를 대상으로 직접 설정합니다. 클러스터를 대상으로 하지 않으므로, 원격 Dispatcher는 어드바이저 요청을 원격 서버로 로드 밸런싱하지 않습니다. 그러나, 원격 어드바이저는 Dispatcher의 cbr 또는 nat 전달을 사용할 때 올바르게 작동합니다.

원격 Dispatcher에서: 원격 클러스터 주소마다 다음과 같은 구성 단계를 수행하십시오. 원격 Dispatcher 위치에 있는 고가용성 구성의 경우, 사용자는 두 시스템 모두에서 다음의 단계를 수행해야 합니다.

AIX 시스템

- Dispatcher에는 어드바이저의 올바른 작동을 위해 넷마스크 255.255.255.255의 인터페이스에 구성된 각각의 클러스터가 있어야 합니다. 클러스터 구성을 위해 다음의 구문 형식 중 하나를 사용하십시오.

– `ifconfig interface_name alias cluster_address netmask 255.255.255.255`. 예를 들어,

```
ifconfig en0 alias 10.10.10.99 netmask 255.255.255.255
```

– `dscontrol executor configure interface_address interface_name netmask`. 예를 들어,

```
dscontrol executor configure 204.67.172.72 en0 255.255.255.255
```

주: 로컬 및 원격 Dispatcher 시스템 모두에서 어드바이저가 실행되어야 합니다.

HP-UX 시스템, Linux, Solaris 및 Windows 시스템

- 추가 구성 단계는 필요하지 않습니다.

구성 예제

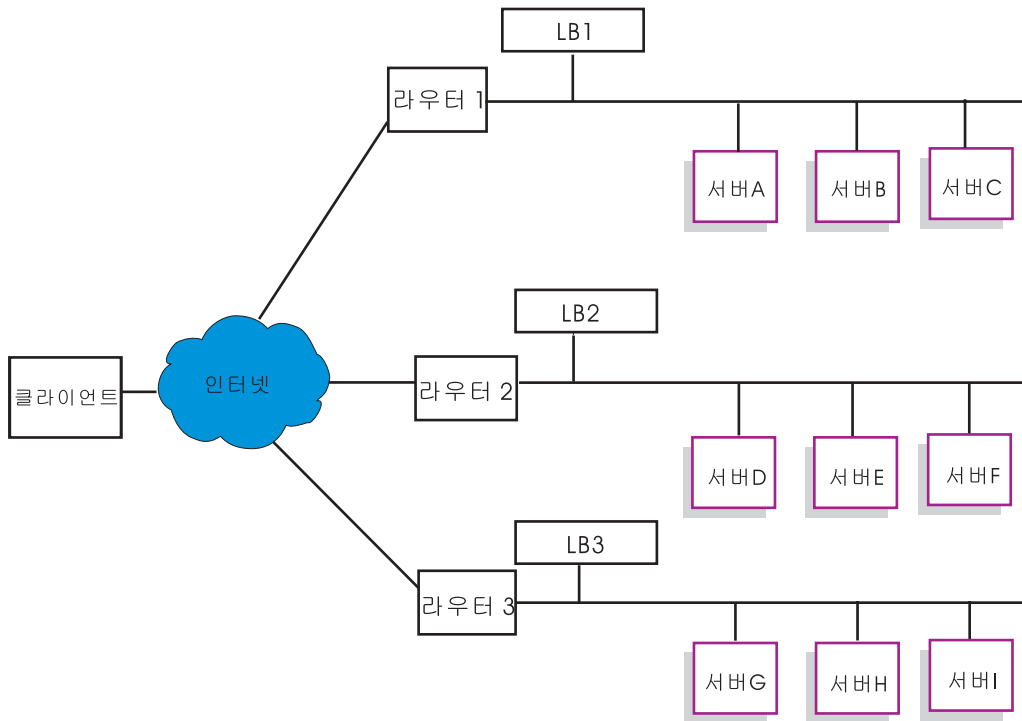


그림 37. 원격 Load Balancer의 광역 구성 예제

이 예제는 그림 37에 설명된 구성에 적용됩니다.

다음은 포트 80에서 클러스터 주소 xebec를 지원하기 위해 Dispatcher 시스템을 구성하는 방법입니다. LB1이 『시작점』 Load Balancer로 정의됩니다. 이더넷 연결로 가정합니다. LB1에 5개의 서버, 즉 3개의 로컬 서버(ServerA, ServerB, ServerC) 및 2개의 원격 서버(LB2 및 LB3)가 정의되어 있다는 점에 유의하십시오. 원격 LB2 및 LB3에는 각각 3개의 로컬 서버가 정의되어 있습니다.

첫 번째 Dispatcher(LB1)의 콘솔에서 다음을 수행하십시오.

1. 실행 프로그램을 시작합니다.

dscontrol executor start

2. Dispatcher 시스템의 비전달 주소를 설정하십시오.

dscontrol executor set nfa LB1

3. 클러스터를 정의하십시오.

dscontrol cluster add xebec

4. 포트를 정의하십시오.

dscontrol port add xebec:80

5. 서버를 정의하십시오.
 - a. **dscontrol server add xebec:80:ServerA**
 - b. **dscontrol server add xebec:80:ServerB**
 - c. **dscontrol server add xebec:80:ServerC**
 - d. **dscontrol server add xebec:80:LB2 router Router1**
 - e. **dscontrol server add xebec:80:LB3 router Router1**
6. 클러스터 주소를 구성합니다.

dscontrol executor configure xebec

두 번째 Dispatcher(LB2)의 콘솔에서 다음을 수행하십시오.

1. 실행 프로그램을 시작합니다.

dscontrol executor start

2. Dispatcher 시스템의 비전달 주소를 설정하십시오.

dscontrol executor set nfa LB2

3. 클러스터를 정의하십시오.

dscontrol cluster add xebec

4. 포트를 정의하십시오.

dscontrol port add xebec:80

5. 서버를 정의하십시오.
 - a. **dscontrol server add xebec:80:ServerD**
 - b. **dscontrol server add xebec:80:ServerE**
 - c. **dscontrol server add xebec:80:ServerF**

세 번째 Dispatcher(LB3)의 콘솔에서 다음을 수행하십시오.

1. 실행 프로그램을 시작합니다.

dscontrol executor start

2. Dispatcher 시스템의 비전달 주소를 설정하십시오.

dscontrol executor set nfa LB3

3. 클러스터를 정의하십시오.

dscontrol cluster add xebec

4. 포트를 정의하십시오.

dscontrol port add xebec:80

5. 서버를 정의하십시오.
 - a. **dscontrol server add xebec:80:ServerG**
 - b. **dscontrol server add xebec:80:ServerH**
 - c. **dscontrol server add xebec:80:ServerI**

주

1. 모든 서버(A-I)에서 루프백에 대해 클러스터 주소의 별명을 지정하십시오.
2. 클러스터 및 포트는 참여 중인 모든 Dispatcher 시스템(시작점 Dispatcher와 모든 원격지)에서 dscontrol을 사용하여 추가됩니다.
3. 광역 지원을 통한 원격 어드바이저 사용에 대한 도움말을 보려면 245 페이지의 『Dispatcher의 광역 지원으로 원격 어드바이저 사용』을 참조하십시오.
4. 광역 지원은 무한 경로 지정 루프를 금지합니다(Dispatcher 시스템이 다른 Dispatcher에서 패킷을 수신하면, 이 패킷은 세 번째 Dispatcher로 전달되지 않습니다). 광역 지원은 한 레벨의 원격지만 지원합니다.
5. 광역 지원은 UDP와 TCP만 지원합니다.
6. 광역 지원은 고가용성으로 작동되며, 각 Dispatcher는 인접한 대기 시스템(같은 LAN 세그먼트에 있는)에서 백업될 수 있습니다.
7. 관리자와 어드바이저는 광역 지원으로 작동되며, 사용할 경우 참여 중인 모든 Dispatcher 시스템에서 시작되어야 합니다.
8. []는 비슷한 운영 체제에 대해서만 WAN을 지원합니다.

GRE(일반 경로 지정 캡슐화) 지원

GRE(일반 경로 지정 캡슐화)는 RFC 1701 및 RFC 1702에 지정된 인터넷 프로토콜입니다. GRE를 사용하면 Load Balancer에서 IP/GRE 패킷 내의 클라이언트 IP 패킷을 캡슐화하여 GRE를 지원하는 OS/390 같은 서버 플랫폼에 전달할 수 있습니다. GRE 지원을 사용하면 Dispatcher 컴포넌트는 하나의 MAC 주소에 연관된 여러 서버 주소로 패킷을 로드 밸런싱할 수 있습니다.

Load Balancer가 WAN 기능의 일부로 GRE를 구현합니다. 그러면 Load Balancer가 GRE 패킷을 열 수 있는 모든 서버 시스템에 광역 로드 밸런스를 직접 제공할 수 있습니다. 원격 서버가 캡슐화 GRE 패킷을 지원하는 경우, 원격 사이트에 []를 설치할 필요가 없습니다. []는 3735928559 10진수 값에 설정된 GRE 키 필드로 WAN 패킷을 캡슐화합니다.

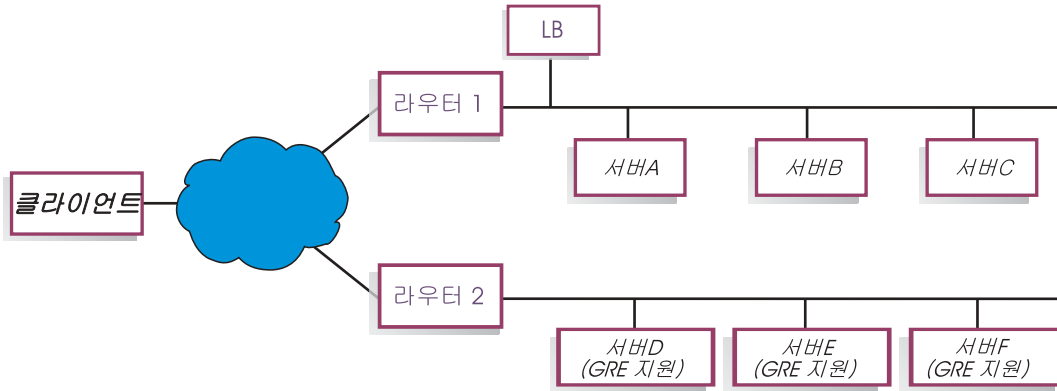


그림 38. GRE를 지원하는 서버 플랫폼의 광역 구성 예제

이 예제(그림 38)에서 GRE를 지원하는 원격 ServerD를 추가하려면 `cluster:port:server` 계층에 WAN 서버를 정의한 것과 마찬가지로 Load Balancer 구성 내에 ServerD를 정의하십시오.

```
dscontrol server add cluster:port:ServerD router Router1
```

Linux 시스템의 경우, WAN용 GRE 캡슐화 구성

Linux 시스템에는 Load Balancer가 S/390용 Linux 서버 이미지(여기서 많은 서버 이미지가 MAC 주소를 공유함)에 대한 로드 밸런스를 유지할 수 있도록 GRE를 캡슐화하는 고유한 기능이 있습니다. 이 기능은 시작점 Load Balancer가 원격 사이트에서 Load Balancer를 경유하지 않고, Linux WAN 서버에 직접 로드 밸런스를 수행할 수 있도록 허용합니다. 또한 시작점 Load Balancer의 어드바이저가 각각의 원격 서버와 직접 작업할 수 있게 해줍니다.

시작점 Load Balancer에서, WAN에 대하여 설명된 대로 구성하십시오.

각각의 Linux 백엔드 서버를 구성하려면, 다음과 같은 명령을 루트로 발행하십시오.(이들 명령을 시스템의 시작 기능에 추가하여 변경사항이 재부팅 시 보존되도록 할 수 있습니다.)

```
# modprobe ip_gre
# ip tunnel add gre-nd mode gre ikey 3735928559
# ip link set gre-nd up
# ip addr add cluster address dev gre-nd
```

주: 이들 명령을 사용하여 구성된 Linux 서버는 시작점 Load Balancer와 동일한 물리적 세그먼트에 위치하지 않아야 합니다. 이는 Linux 서버가 클러스터 주소에 대한 "ARP who-has" 요청에 응답하여, 클러스터 주소의 모든 통신량이 ARP 레이스의 승자에게로만 지정되는 "short-circuit" 레이스 조건을 야기할 수 있습니다.

명시적 링크 사용

일반적으로, Dispatcher의 로드 밸런스 기능은 제품이 사용되는 사이트의 콘텐츠와는 관계없이 작동됩니다. 그러나 사이트 콘텐츠가 중요할 수 있으며 콘텐츠에 대한 결정이 Dispatcher의 효율성에 상당한 영향을 줄 수 있는 하나의 영역이 있습니다. 이 영역은 링크 주소 지정 영역입니다.

사용자 페이지에서 사이트의 개별 서버를 가리키는 연결을 지정할 때 클라이언트가 실제로 특정 시스템으로 이동하게 되므로 다른 경우라면 일어나지 않는 로드 밸런스 기능이 생략됩니다. 이러한 이유로 항상 사용자 페이지에 포함된 링크에서 Dispatcher의 주소를 사용하십시오. 사용자 사이트에서 동적으로 HTML을 작성하는 자동화 프로그램 래밍을 사용할 경우, 사용되는 주소 지정 종류가 항상 명백하지 않을 수도 있다는 점에 유의하십시오. 로드 밸런스를 최대화하려면 명시적 주소 지정에 유의하여 가능한 이를 피해야 합니다.

개인용 네트워크 구성 사용

개인용 네트워크를 사용하여 Dispatcher와 TCP 서버 시스템을 설정할 수 있습니다. 이 구성을 사용하면 성능에 영향을 줄 수 있는 공용 또는 외부 네트워크에서의 회선 경합이 줄어들 수 있습니다.

AIX 시스템의 경우, Dispatcher 및 TCP 서버 시스템을 SP™ 프레임의 노드에서 실행할 경우 이 구성에서는 SP High Performance Switch의 빠른 속도를 이용할 수도 있습니다.

사설 네트워크를 작성하려면 각 시스템에는 최소 두 개의 LAN 카드가 필요하며, 카드 중 하나는 사설 네트워크에 연결되어 있어야 합니다. 다른 서브넷에서 두 번째 LAN 카드도 구성해야 합니다. 그러면 Dispatcher 시스템은 클라이언트 요청을 사설 네트워크를 통해 TCP 서버 시스템에 전송합니다.

Windows 시스템: `executor configure` 명령을 사용하여 비전달 주소를 구성하십시오.

`dscontrol server add` 명령을 사용하여 추가된 서버는 사설 네트워크 주소를 사용하여 추가되어야 합니다. 예를 들어, 253 페이지의 그림 39에 나와 있는 Apple 서버 예제의 경우는 다음과 같이 코딩되어야 합니다.

```
dscontrol server add cluster_address:80:10.0.0.1
```

다음과 같이 코딩해서는 안됩니다.

```
dscontrol server add cluster_address:80:9.67.131.18
```

Site Selector를 사용하여 Dispatcher에 로드 정보를 제공할 경우, Site Selector를 구성하여 개인 주소에 대한 로드를 보고해야 합니다.

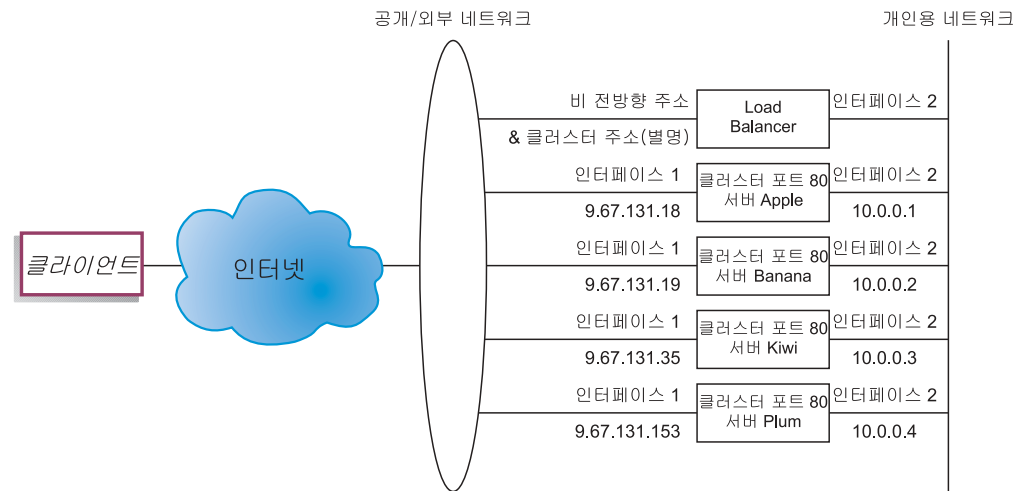


그림 39. Dispatcher를 사용하는 사설 네트워크 예제

Dispatcher 컴포넌트의 경우에만 사설 네트워크 구성이 적용됩니다.

와일드 카드 클러스터를 사용하여 서버 구성 조합

Dispatcher 컴포넌트의 경우에만 와일드 카드 클러스터를 사용한 서버 구성 조합이 적용됩니다.

“와일드 카드”는 여러 개의 IP 주소와 일치할 수 있는 클러스터의 기능(즉, 와일드 카드의 역할을 하는)을 나타냅니다. 와일드 카드 클러스터를 지정하는 데 클러스터 주소 0.0.0.0이 사용됩니다.

로드 밸런스를 수행할 클러스터 주소가 많고 포트/서버 구성이 사용자의 모든 클러스터에 대해 동일한 경우, 클러스터를 하나의 와일드 카드 구성으로 조합할 수 있습니다.

Dispatcher 워크스테이션의 네트워크 어댑터 중 하나에 각 클러스터 주소를 명시적으로 구성해야 합니다. 그러나 `dscontrol cluster add` 명령을 사용하여 Dispatcher 구성에 클러스터 주소 중 어느 것도 추가해서는 안 됩니다.

와일드 카드 클러스터(주소 0.0.0.0)만 추가한 후 로드 밸런스에 필요한 만큼 포트와 서버를 구성하십시오. 어댑터 구성 주소에 대한 통신량이 와일드 카드 구성을 사용하여 로드 밸런스를 수행합니다.

이러한 접근 방법의 이점은 이동할 최상의 서버를 결정할 때 모든 클러스터 주소에 대한 통신량을 고려한다는 점입니다. 하나의 클러스터가 많은 통신량을 확보하고 서버 중 하나에서 다수의 활성 연결 수를 작성했으면, 다른 클러스터 주소에 대한 통신량은 이 정보를 사용하여 로드 밸런스를 수행합니다.

고유 포트/서버 구성이 있는 클러스터 주소와 공통 구성이 있는 클러스터 주소가 몇 가지씩 있는 경우, 실제 클러스터와 와일드 카드 클러스터를 조합할 수 있습니다. 고유 구성은 각각 실제 클러스터 주소에 지정되어야 합니다. 모든 공통 구성은 와일드 카드 클러스터에 지정될 수 있습니다.

와일드 카드 클러스터를 사용하여 방화벽 로드 밸런스 수행

Dispatcher 컴포넌트의 경우에만 와일드 카드 클러스터를 방화벽 로드 밸런스가 적용됩니다. 와일드 카드 클러스터를 지정하는 데 클러스터 주소 0.0.0.0이 사용됩니다.

Dispatcher 워크스테이션의 네트워크 어댑터에서 명시적으로 구성되지 않은 주소에 대한 통신량 로드 밸런스를 수행하는 데 와일드 카드 클러스터를 사용할 수 있습니다. 이렇게 작업하려면, Dispatcher는 최소한 로드 밸런스를 수행할 모든 통신량을 알 수 있어야 합니다. Dispatcher 워크스테이션은 어떤 통신량 세트의 기본 라우트로서 설정되지 않은 경우, 그 네트워크 어댑터 중 하나에서 명시적으로 구성되지 않은 주소에 대한 통신량은 알지 못합니다.

Dispatcher가 기본 라우트로 구성된 후, Dispatcher 시스템을 통한 TCP 또는 UDP 통신량은 와일드 카드 클러스터 구성을 통해 로드 밸런스가 수행합니다.

이 중 하나의 응용프로그램은 방화벽 로드 밸런스를 수행하게 됩니다. 방화벽이 대상 주소와 대상 포트에 대해 패킷을 처리할 수 있으므로 대상 주소 및 포트에 관계없이 통신량 로드 밸런스를 수행할 수 있어야 합니다.

방화벽은 비보안 클라이언트에서 보안 클라이언트까지의 통신량과 보안 서버의 응답, 보안측의 클라이언트에서 비보안측의 서버까지의 통신량과 응답을 처리하는 데 사용됩니다.

두 대의 Dispatcher 시스템을 설정해야 하는데, 하나는 비보안 방화벽 주소에 대한 비보안 통신량의 로드 밸런스를 위한 것이고 또 하나는 보안 방화벽 주소에 대한 보안 통신량의 로드 밸런스를 위한 것입니다. 이들 Dispatcher 모두 서로 다른 서버 주소 세트를 가진 와일드 카드 클러스터와 와일드 카드 포트를 사용해야 하므로, 두 Dispatcher는 별도의 두 개의 워크스테이션에 있어야 합니다.

투명 프록시의 경우 Caching Proxy가 있는 와일드 카드 클러스터 사용

Dispatcher 컴포넌트의 경우에만 투명 프록시를 위해 Caching Proxy가 있는 와일드 카드 클러스터 사용이 적용됩니다. 와일드 카드 클러스터를 지정하는 데 클러스터 주소 0.0.0.0이 사용됩니다.

와일드 카드 클러스터 기능으로 Dispatcher는 Dispatcher와 동일한 시스템에 있는 Caching Proxy 서버의 투명 프록시 기능을 작동시키는 데 사용될 수도 있습니다. 이 기능은 Dispatcher 컴포넌트에서 운영 체제의 TCP 컴포넌트로의 통신이 있어야 하므로 AIX 기능 전용입니다.

이 기능을 사용하려면 포트 80에서 클라이언트 요청을 인식하는 Caching Proxy를 시작해야 합니다. 그런 다음 와일드 카드 클러스터(0.0.0.0)를 구성합니다. 와일드 카드 클러스터에서 포트 80을 구성합니다. 포트 80에서는 유일한 서버로서 Dispatcher 시스템의 NFA를 구성합니다. 이제 포트 80의 주소에 대한 클라이언트 통신량은 Dispatcher 워크스테이션에서 실행 중인 Caching Proxy 서버로 전달됩니다. 그러면 클라이언트 요청은 평소와 같이 프록시되며 응답은 Caching Proxy에서 클라이언트로 다시 전송됩니다. 이 모드에서 Dispatcher 컴포넌트는 어떤 로드 밸런싱도 수행하지 않습니다.

와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정

와일드 카드 포트는 명시적으로 구성된 포트용이 아닌 통신량을 처리하는 데 사용될 수 있습니다. 이 중 하나는 방화벽 로드 밸런싱을 위해 사용됩니다. 구성되어 있지 않은 포트에 대한 통신량이 적당하게 처리되도록 두 번째 것이 사용됩니다. 서버 없이 와일드 카드 포트를 정의하여 구성되지 않은 포트에 대한 요청이 운영 체제로 다시 전달되지 않고 버려집니다. 포트 번호 0은 다음과 같은 와일드카드 포트를 지정합니다.

```
dscontrol port add cluster:0
```

FTP 통신량을 처리하기 위한 와일드 카드 포트

수동 FTP를 처리하는 클러스터와 와일드 카드 포트를 구성할 경우 기본적으로 수동 FTP에서는 데이터 연결을 위해 권한이 없는 전체 TCP 포트 범위를 사용합니다. 이것은 로드 밸런싱 클러스터를 통해 FTP 제어 포트에 기존 연결을 갖고 있는 클라이언트가 Load Balancer에 의해 FTP 제어 연결과 동일한 서버로 자동으로 라우트된 동일한 클러스터에 대한 후속 제어 연결과 높은 포트 연결(포트 >1023)을 갖게 됨을 의미합니다.

동일한 클러스터의 와일드 카드 포트와 FTP 포트가 동일한 서버 설정을 갖고 있지 않은 경우 클라이언트가 기존 FTP 제어 연결을 갖고 있으면 높은 포트 응용프로그램(포트 >1023)이 실패할 수 있습니다. 따라서 동일한 클러스터에서 FTP와 와일드 카드 포트에 대해 다른 서버 설정을 하는 것은 권장하지 않습니다. 이런 시나리오를 원할 경우 Load Balancer 구성에 FTP 디먼 수동 포트 범위를 구성해야 합니다.

서비스 거부 중지 감지

Dispatcher 컴포넌트의 경우에만 이 기능을 사용할 수 있습니다.

Dispatcher에서는 잠재적 "서비스 거부" 중지를 감지하여 경보로 관리자에게 통지하는 기능을 제공합니다. Dispatcher는 서버의 방대한 양의 반개방 TCP 연결에 대한 수신 요청, 단순한 서비스 거부 중지의 일반적 특징을 분석하여 이 기능을 수행합니다. 서비스 거부 중지 시, 사이트는 많은 출발지 IP 주소 및 출발지 포트 번호에서 많은 양의 조립된 SYN 패킷을 받지만 TCP 연결에 대한 후속 패킷은 받지 않습니다. 따라서 서버의 반개방 TCP 연결 수가 많아지고 시간이 지나면서 서버가 느려져서 새 수신 연결을 받을 수 없게 됩니다.

주: 서비스 거부 공격의 끝을 확인하기 위해 Dispatcher에 대한 공격을 받고 있는 클러스터와 포트를 통해 들어오는 통신량이 있어야 합니다. 통신량이 다시 호르기 시작할 때까지 Dispatcher는 공격이 정지되었는지 검색할 수 없습니다.

[1]은 관리자에게 가능한 서비스 거부 중지를 경보하도록 사용자 정의할 수 있는 스크립트를 트리거하는 사용자 엑시트를 제공합니다. Dispatcher는 `...ibm/edge/lb/servers/samples` 디렉토리에 다음의 예제 스크립트 파일을 제공합니다.

- `halfOpenAlert` — 가능한 서비스 거부(DoS) 중지가 감지되었습니다.
- `halfOpenAlertDone` — DoS 중지가 완료되었습니다.

파일을 실행하려면 예제 스크립트를 `...ibm/edge/lb/servers/bin` 디렉토리로 이동하여 확장자가 `".sample"`인 파일을 제거하십시오.

DoS 중지 감지를 구현하려면, 다음과 같이 `dscontrol port` 명령에서 `maxhalfopen` 매개변수를 설정하십시오.

```
dscontrol port set 127.40.56.1:80 maxhalfopen 1000
```

위의 예제에서 Dispatcher는 현재의 반개방 총 연결 수(포트 80의 127.40.56.1 클러스터에 있는 모든 서버)를 임계치 값 1000(maxhalfopen 매개변수로 지정됨)과 비교합니다. 현재의 반개방 연결 수가 임계치를 초과하면 경보 스크립트(halfOpenAlert)를 호출합니다. 반개방 연결 수가 임계치 이하로 떨어지면 다른 경보 스크립트(halfOpenAlertDone)를 호출하여 중지가 종료되었음을 표시합니다.

`maxhalfopen` 값 설정 방법을 결정하려면 다음을 수행하십시오. 사이트의 통신량이 보통에서 많아지면, 정기적으로(10분마다) 반개방 연결 보고서(`dscontrol port halfopenaddressreport cluster:port`)를 실행하십시오. 반개방 연결 보고서는 현재의 "수신된 총 반개방 연결"을 리턴합니다. 사용자의 사이트에 연결되는 가장 많은 반개방 연결 수보다 큰 50% - 200%의 범위에 있는 값으로 maxhalfopen을 설정해야 합니다.

보고된 통계 데이터 이외에 halfopenaddressreport는 반개방 연결을 생성하는 서버에 액세스한 모든 클라이언트 주소(최대 8000개의 주소쌍)에 대해 로그(`..ibm/edge/lb/servers/logs/dispatcher/halfOpen.log`)에 항목을 생성합니다.

주: halfOpenAlert 및 halfOpenAlertDone 스크립트에 해당하는 SNMP 트랩이 있습니다. SNMP 서브에이전트가 구성되어 실행 중이면 대응하는 트랩도 스크립트를 트리거하는 조건과 동일한 조건에서 전송됩니다. SNMP 서브에이전트에 대한 자세한 정보는 290 페이지의 『Dispatcher 컴포넌트에 Simple Network Management Protocol 사용』을 참조하십시오.

백엔드 서버에 대한 서비스 거부 중지로부터 추가로 보호하기 위해 와일드 카드 클러스터 및 포트를 구성할 수 있습니다. 특히 구성된 각 클러스터 아래에 서버 없이 와일드 카드 포트를 하나 추가합니다. 또한 서버가 없고 와일드 카드 포트가 하나 있는 와일드 카드 클러스터 하나를 추가합니다. 이렇게 하면 와일드 카드가 없는 클러스터 및 포트 주소가 지정되지 않은 모든 패킷을 버릴 수 있는 효과가 있습니다. 와일드 카드 포트 및 와일드 카드 클러스터에 대한 정보는 253 페이지의 『와일드 카드 클러스터를 사용하여 서버 구성 조합』 및 255 페이지의 『와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정』을 참조하십시오.

서버 통제를 분석하기 위해 2진 로그 사용

주: 2진 로깅 기능이 Dispatcher와 CBR 컴포넌트에 적용됩니다.

2진 로그 기능으로 서버 정보를 2진 파일에 저장할 수 있습니다. 그러면 이 파일은 시간 초과 시 수집되었던 서버 정보를 분석하기 위해 프로세스될 수 있습니다.

다음 정보는 구성에서 정의된 각 서버의 2진 로그에 저장됩니다.

- 클러스터 주소
- 포트 번호
- serverID
- 서버 주소
- 서버 가중치
- 서버 총 연결
- 서버 총 활성화 연결
- 서버 포트 로드
- 서버 시스템 로드

몇몇 정보는 관리자 주기의 부분으로서 실행 프로그램에서 검색합니다. 따라서 관리자는 2진 로그에 로그될 정보에 대해 순서대로 실행되어야 합니다.

dscontrol binlog 명령은 2진 로그 구성을 설정합니다.

- binlog 시작
- binlog 정지

- binlog 세트 간격 <초>
- binlog 세트 보존 <시>
- binlog 상태

이 시작 옵션은 로그 디렉토리에 있는 2진 로그로 서버 정보를 로그하기 시작합니다. 한 로그는 파일의 이름으로 날짜와 시간이 매시 생성됩니다.

중지 옵션은 2진 로그로 서버 정보 로그를 중지합니다. 로그 서비스는 초기값으로 중지됩니다.

설정 간격 옵션은 얼마나 자주 정보가 로그에 씌여질 것인지를 제어합니다. 관리자는 관리자 간격마다 로그 서버로 서버 정보를 보냅니다. 마지막 기록이 로그에 씌여진 이후 지정된 로그 간격 초가 경과된 경우에만 정보가 로그에 기록됩니다. 기본값으로 로그 간격은 60초로 설정됩니다. 관리자 간격과 로그 간격 설정 사이에는 약간의 상호작용이 있습니다. 로그 서버는 관리자 간격 초 설정보다 느린 정보를 제공하므로, 관리자 간격보다 느린 로그 간격은 관리자 간격과 동일하게 정보를 설정합니다. 이 로그 기술을 사용하여 미세한 서버 정보를 캡처할 수 있습니다. 서버 가중치를 계산하는 관리자가 보여주는 서버 정보로 모든 변경사항을 캡처할 수 있습니다. 그러나 이 정보의 양이 서버 사용과 경향을 분석하는 데 요구되는 것은 아닙니다. 60초마다 서버 정보를 로그하는 것은 시간 경과 시 서버 정보의 스냅샷을 제공합니다. 매우 느린 로그 간격 설정은 많은 양의 데이터를 생성할 수 있습니다.

보존 설정 옵션은 로그 파일이 보존되는 기간을 제어합니다. 지정된 보존 시간보다 더 오래된 로그 파일은 로그 서버에 의해 삭제됩니다. 관리자에서 로그 서버를 호출하고 있는 경우에만 발생하므로, 관리자를 정지하면 이전의 로그 파일이 삭제되지 않습니다.

상태 옵션은 로그 서비스의 현재 설정을 리턴합니다. 이러한 설정으로는 서비스 시작 여부, 간격 및 보존 시간 등이 있습니다.

예제 Java 프로그램 및 명령 파일이 **...ibm/edge/lb/servers/samples/BinaryLog** 디렉토리에 제공됩니다. 이 예제는 로그 파일에서 모든 정보를 검색하여 화면에 인쇄하는 방법을 보여줍니다. 이 예제는 데이터로 원하는 유형의 분석을 수행하기 위해 사용자가 정의할 수 있습니다. Dispatcher에 대해 제공된 스크립트 및 프로그램 사용에 관한 예제는 다음과 같습니다.

```
dslogreport 2001/05/01 8:00 2001/05/01 17:00
```

2001년 5월 1일 오전 8시부터 오후 5시까지 Dispatcher 컴포넌트의 서버 정보에 대한 보고서를 작성하기 위한 것입니다. (CBR의 경우, **cbrlogreport**를 사용하십시오.)

결합 배치된 클라이언트 사용

Load Balancer와 동일한 시스템에 있는 클라이언트에 대해서는 Linux 시스템만 구성을 지원합니다.

Load Balancer가 다른 테크닉을 사용하여 지원하는 여러 운영 체제의 수신 패킷을 검사하기 때문에, 결합 배치된 클라이언트 구성이 다른 플랫폼에서 바르게 기능하지 않을 수 있습니다. 대부분의 경우 Linux 이외의 시스템에서는 Load Balancer가 로컬 시스템으로부터 패킷을 수신하지 않습니다. 네트워크에서 오는 패킷만 수신합니다. 이로 인해 Load Balancer가 로컬 시스템로부터 클러스터 주소에 작성된 요청을 수신할 수 없으며 서비스할 수 없습니다.

제 23 장 Cisco CSS Controller 및 Nortel Alteon Controller의 고급 기능

이 장에는 다음과 같은 섹션이 수록되어 있습니다.

- 『결합 배치』
- 『고가용성』
- 264 페이지의 『Load Balancer에서 제공하는 로드 밸런스 최적화』
- 266 페이지의 『어드바이저』
- 272 페이지의 『Metric Server』
- 275 페이지의 『서버 통제를 분석하기 위해 2진 로그 사용』
- 276 페이지의 『스크립트를 사용하여 정보나 레코드 서버 장애 생성』

주: 이 장에서 **xxxcontrol**은 Cisco CSS Controller의 경우에는 **cococontrol**, Nortel Alteon Controller의 경우에는 **nalcontrol**을 선언합니다.

결합 배치

Cisco CSS Controller 또는 Nortel Alteon Controller는 요청의 로드 밸런스를 유지하는 서버와 동일한 시스템에 상주할 수 있습니다. 이것을 보통 서버 **결합 배치**라고 합니다. 추가 구성 단계는 필요하지 않습니다.

주: 결합 배치된 서버는 통신량이 많을 때 I/O와 자원을 차지하기 위해 경쟁합니다. 그러나 과부하된 시스템이 없을 때 결합 배치된 서버를 사용하면 로드 밸런스된 사이트를 설정하는 데 필요한 총 시스템 수가 줄어듭니다.

고가용성

고가용성 기능은 이제 Cisco CSS Controller 및 Nortel Alteon Controller에 대해 사용 가능합니다.

고가용성 기능은 제어기의 결합 허용을 향상시키기 위해 다음과 같은 기능을 포함합니다.

- 상대 제어기의 사용가능성을 판별하기 위한 하트 비트 메커니즘. 하트 비트는 **xxxcontrol highavailability add** 명령에 구성된 주소 간에 교환됩니다. 메시지를 교환하는 간격 및 제어기가 상대로부터 인수하는 간격을 구성할 수 있습니다.
- 가중치를 계산하고 스위치를 갱신하기 위해 제어기가 도달할 수 있어야 하는 도달 목표 목록. 263 페이지의 『장애 발견』에서 자세한 정보를 참조하십시오.

- 사용가능성 및 도달 정보를 기본으로 활성 제어기를 선택하는 논리
- 제어기가 상대로부터 인수하는 방법을 결정하는 데 사용되는 구성 가능한 인수 전략
- 활성 제어기에서의 유지보수를 위한 수동 인수 메커니즘
- 현재 제어기의 역할, 상태, 동기화등을 표시하는 보고서

구성

xxxcontrol highavailability의 전체 구문에 대해서는 466 페이지의 『ccocontrol highavailability — 고가용성 제어』 및 487 페이지의 『nalcontrol highavailability — 고가용성 제어』를 참조하십시오.

제어기 고가용성을 구성하려면 다음을 수행하십시오.

1. 두 대의 제어기 시스템에서 제어기 서버를 시작하십시오.
2. 동일한 구성으로 각 제어기를 구성하십시오.
3. 다음과 같이 로컬 고가용성 역할, 주소 및 상대 주소를 구성하십시오.

```
xxxcontrol highavailability add address 10.10.10.10  
partneraddress 10.10.10.20 port 143 role primary
```

4. 다음과 같이 상대 고가용성 역할, 주소 및 상대 주소를 구성하십시오.

```
xxxcontrol highavailability add address 10.10.10.20  
partneraddress 10.10.10.10 port 143 role secondary
```

address 및 partneraddress 매개변수는 기본 및 보조 시스템에서는 반대입니다.

5. 선택적으로 로컬 및 상대 제어기에 고가용성 매개변수를 구성하십시오. 예를 들면, 다음과 같습니다.

```
xxxcontrol highavailability set beatinterval 1000
```

6. 선택적으로 다음과 같이 로컬 및 상대 제어기에 도달 목표를 구성하십시오.

```
xxxcontrol highavailability usereach 10.20.20.20
```

로컬 및 상대 제어기에 동일한 수의 도달 목표를 구성해야 합니다.

7. 다음과 같이 고가용성 컴포넌트를 시작하고 로컬 및 상대 제어기에 복구 전략을 정의하십시오.

```
xxxcontrol highavailability start auto
```

8. 선택적으로, 다음과 같이 로컬 및 상대 제어기에 고가용성 정보를 표시하십시오.

```
xxxcontrol highavailability report
```

9. 선택적으로 다음과 같이 대기 제어기에 takeover를 지정하여 상대 제어기로부터 인수하십시오.

```
xxxcontrol highavailability takeover
```

이것은 유지보수에서만 필요합니다.

주:

1. 고가용성 없이 단일 제어를 구성하려면 어떠한 고가용성 명령도 발행하지 마십시오.
2. 고가용성 구성의 두 개의 제어를 단일 제어로 변환하려면, 먼저 대기 제어기에서 고가용성을 정지한 후 선택적으로 활성 제어기에서 고가용성을 정지하십시오.
3. 고가용성 구성에서 두 개의 제어를 실행할 경우, 스위치 간에 제어기 등록 정보(예: switchconsultantid, 스위치 주소 등)가 다르면 예상치 않은 결과가 발생할 수 있습니다. 제어기 고가용성 등록 정보(예: 포트, 역할, 도달 목표, beatinterval, takeoverinterval 및 복구 전략)가 일치하지 않으면 예상치 못한 결과가 발생할 수도 있습니다.

장애 발견

하트 비트를 통해 발견되는 활성 및 대기 제어기 간의 연결성 이외에도 도달 가능성이 있는 다른 장애 발견 메커니즘이 있습니다.

제어기 고가용성 구성 시 제어기가 제대로 작동하기 위해 반드시 도달해야 하는 호스트 목록을 제공할 수 있습니다. 제어기 시스템이 사용하는 각 서브넷에 대해 최소 하나의 호스트가 있어야 합니다. 이 호스트는 라우터, IP 서버 또는 기타 호스트 유형일 수 있습니다.

호스트를 ping하는 도달 어드바이저에 의해 호스트 도달 가능성이 확보됩니다. 스위치 전환은 하트 비트를 전달할 수 없거나 도달 가능성 기준이 활성 제어기보다 대기 제어기에 더 적합할 경우에 발생합니다. 모든 사용 가능한 정보에 근거하여 스위치 전환을 결정하기 위해 활성 제어기는 정기적으로 대기 제어기에 도달 가능성 기능을 전송하고 역으로 대기 제어기도 활성 제어기로 도달 가능성 기능을 전송합니다. 제어기는 신뢰성 정보를 상대의 정보와 비교하여 활성화해야 하는 제어를 결정합니다.

복구 전략

두 제어기 시스템의 역할은 기본 및 보조 제어기로 구성되어 있습니다. 시동 시 제어기는 각 시스템이 동기화될 때까지 정보를 교환합니다. 이 시점에서 기본 제어기는 활성 상태로 이동하여 가중치 계산 및 스위치 갱신을 시작하는 반면 보조 시스템은 대기 상태로 이동하여 기본 시스템의 사용 가능성을 모니터링합니다.

어느 시점에서든 대기 시스템이 활성 시스템의 장애를 발견하면, 대기 시스템이 활성 시스템(장애가 발생한 시스템)의 로드 밸런스 기능을 인수하여 활성화됩니다. 기본 시스템이 다시 작동되면 두 시스템은 복구 전략이 구성된 방법에 따라 활성화할 제어를 결정합니다.

두 종류의 복구 전략이 있습니다.

자동 복구

기본 제어기는 다시 작동하는 즉시 활성 상태로 이동하여 가중치를 계산하고 갱신합니다. 보조 시스템은 기본 시스템이 활성화된 후 대기 상태로 이동합니다.

수동 복구

활성화된 보조 제어기는 기본 제어기가 작동된 이후에도 활성 상태로 남아 있습니다.

기본 제어기는 대기 상태로 이동하며 활성 상태가 되려면 수동 개입이 필요합니다.

전략 매개변수는 두 시스템에 동일하게 설정되어야 합니다.

예제

Cisco CSS Controller 고가용성 예에 대해서는 468 페이지의 『예제』를 참조하십시오.

Nortel Alteon Controller 고가용성 예에 대해서는 489 페이지의 『예제』를 참조하십시오.

Load Balancer에서 제공하는 로드 밸런스 최적화

Load Balancer의 제어기 기능은 다음 설정에 따라 로드 밸런스를 수행합니다.

- 『메트릭 정보에 제공된 중요도』
- 265 페이지의 『가중치』
- 266 페이지의 『가중치 계산 휴면 시간』
- 267 페이지의 『어드바이저 휴면 시간』
- 266 페이지의 『감도 임계치』

이 설정을 변경하여 네트워크에 대한 로드 밸런스를 최적화할 수 있습니다.

메트릭 정보에 제공된 중요도

제어기에서는 가중치 결정 시 다음 메트릭 콜렉터를 일부 또는 모두 사용할 수 있습니다.

- **활성 연결:** 스위치에서 검색된 각 로드 밸런스 서버 시스템의 활성 연결 수.
- **연결 비율:** 스위치에서 검색된 각 로드 밸런스 서버 시스템에서 마지막 조회 이후에 이루어진 새로운 연결 수.
- **CPU:** 각 로드 밸런스 서버 시스템에서 사용 중인 CPU 백분율(Metric Server 에이전트에서 입력).
- **메모리:** 각 로드 밸런스 서버에서 사용 중인 메모리 백분율(Metric Server 에이전트가 입력).
- **시스템 메트릭:** Metric Server 또는 WLM과 같은 시스템 모니터링 도구로 입력.
- **응용프로그램 고유:** 포트에서 인식 중인 어드바이저가 입력.

기본 메트릭은 activeconn 및 connrate입니다.

메트릭 값의 상대적인 중요도 비율을 변경할 수 있습니다. 비율은 백분율로 나타내므로 상대 비율의 합은 100%여야 합니다. 기본값으로 활성 연결 및 새 연결 메트릭이 사용되며 해당 비율은 50/50입니다. 사용자 환경에서 최상의 성능을 제공하는 조합을 찾기 위해 다른 메트릭 비율 조합을 시도해야 할 필요가 있습니다.

비율 값을 설정하려면 다음과 같이 수행하십시오.

Cisco CSS Controller

```
cococontrol ownercontent metrics metricName1 proportion1 metricName2  
proportion2
```

Nortel Alteon Controller

```
nalcontrol service metrics metricName1 proportion1 metricName2  
proportion2
```

가중치

가중치는 응용프로그램 응답 시간 및 사용가능성, 어드바이저로부터의 피드백, Metric Server와 같은 시스템 모니터링 프로그램으로부터의 피드백에 따라 설정됩니다. 가중치를 수동으로 설정하려면 서버에 fixedweight 옵션을 지정하십시오. fixedweight 옵션에 대한 설명은 『제어기 고정 가중치』를 참조하십시오.

가중치는 서비스를 제공하는 모든 서버에 적용됩니다. 특정 서비스의 경우, 요청은 서로에 대해 상대적인 가중치에 따라 서버 간에 분산됩니다. 예를 들어, 하나의 서버가 가중치 10으로 설정되고 다른 서버가 가중치 5로 설정되면 10으로 설정된 서버의 요청수는 5로 설정된 서버 요청수의 두 배가 됩니다.

어드바이저가 서버 종료를 발견하면, 서버의 가중치를 -1로 설정합니다. Cisco CSS Controller 및 Nortel Alteon Controller의 경우, 스위치가 서버를 사용할 수 없다는 사실을 알게 되면 스위치는 서버로의 연결 지정을 정지합니다.

제어기 고정 가중치

제어기가 없으면 어드바이저를 실행할 수 없으며 서버가 작동 중단되었는지 검색할 수도 없습니다. 어드바이저를 실행하기로 했으나 제어기가 특정 서버에 대해 사용자가 설정한 가중치를 갱신하지 않기 바라면 Cisco CSS Controller의 **cococontrol service** 명령이나 Nortel Alteon Controller의 **nalcontrol server** 명령에서 **fixedweight** 옵션을 사용하십시오.

fixedweight 명령을 사용하여 원하는 값으로 가중치를 설정하십시오. 서버 가중치는 고정 가중치가 no로 설정된 다른 명령을 발행할 때까지는 서버가 실행 중일 때 고정 가중치로 유지합니다.

가중치 계산 휴면 시간

전반적인 성능을 최적화하기 위해 메트릭 수집 시간을 제한할 수 있습니다.

컨설턴트 휴면 시간은 컨설턴트가 서버 가중치를 갱신하는 빈도를 지정합니다. 컨설턴트 휴면 시간이 너무 작은 경우, 이는 컨설턴트가 항상 스위치를 인터럽트하므로 결과적으로 성능이 저하될 수 있음을 의미합니다. 컨설턴트 휴면 시간이 너무 크면 스위치의 로드 밸런스가 정확한 최신 정보를 기초로 하지 않음을 의미합니다.

예를 들어, 컨설턴트 휴면 시간을 1초로 설정하려면 다음과 같이 하십시오.

```
xxxcontrol consultant set consultantID sleeptime interval
```

감도 임계치

사용자 서버의 로드 밸런스를 최적화하는 다른 방법을 제공합니다. 최고 속도로 작동하도록, 서버의 가중치가 현저하게 변경된 경우에만 가중치를 갱신합니다. 서버 상태가 약간 변경되었거나 변경되지 않은 경우 가중치를 일정하게 갱신하면 불필요한 오버헤드가 작성됩니다. 서비스를 제공하는 모든 서버의 총 가중치에 대한 백분율 가중치 변경값이 감도 임계치를 초과하면, 로드 밸런스가 연결을 분배하는 데 사용한 가중치가 갱신됩니다. 예를 들어, 총 가중치가 100에서 105로 변경된다고 가정하십시오. 변경값은 5%입니다. 기본 감도 임계치가 5이면 백분율 변경값이 임계치를 넘지 않으므로, Load Balancer에서 사용하는 가중치를 갱신하지 않습니다. 그러나 총 가중치가 100에서 106으로 변경되면, 가중치를 갱신합니다. 컨설턴트의 감도 임계치를 기본값이 아닌 다른 값으로 설정하려면 다음 명령을 입력하십시오.

```
xxxcontrol consultant set consultantID sensitivity percentageChange
```

대부분의 경우, 이 값을 변경할 필요가 없습니다.

어드바이저

어드바이저는 Load Balancer 내의 에이전트입니다. 서버 시스템의 상태와 로드를 평가하는 기능을 합니다. 어드바이저는 서버와의 활성 클라이언트와 같은 교환으로 이를 수행합니다. 어드바이저를 Application Server의 경량 클라이언트로 간주하십시오.

주: 어드바이저 목록에 대한 세부사항은 203 페이지의 『어드바이저 목록』을 참조하십시오.

어드바이저 작동 방법

어드바이저는 정기적으로 각 서버와의 TCP 연결을 열어 서버에 요청 메시지를 전송합니다. 메시지 콘텐츠는 서버에서 실행 중인 프로토콜에 따라 고유합니다. 예를 들어, HTTP 어드바이저는 서버에 HTTP 『HEAD』 요청을 전송합니다.

그런 후 어드바이저는 서버의 응답을 인식합니다. 응답을 받은 후, 어드바이저는 서버를 평가합니다. 이 로드 값을 계산하기 위해, 대부분의 어드바이저는 응답할 서버의 시간을 측정한 후 이 값(밀리초 단위)을 로드로 사용합니다.

그런 다음 어드바이저는 로드 값을 컨설턴트 기능에 보고하고 이 값은 컨설턴트 보고서에 나타납니다. 그러면 컨설턴트는 모든 원본으로부터 비율당 총계 가중치 값을 계산하여 이 가중치 값을 스위치로 전송합니다. 스위치는 이 가중치를 수신 클라이언트 연결 로드 밸런스에 사용합니다.

어드바이저가 서버가 제대로 작동하고 있다고 판단하면 0이 아닌 양의 로드 숫자를 컨설턴트에게 보고합니다. 어드바이저가 서버가 활성화되지 않았다고 판단하면 특수 로드 값인 음수 1(-1)을 리턴하여 스위치에게 서버가 중지되었음을 알려줍니다. 이후로 스위치는 서버가 다시 작동할 때까지 해당 서버로 연결을 전달하지 않습니다.

어드바이저 휴면 시간

주: 어드바이저 기본값은 가능한 많은 시나리오에 효율적으로 적용됩니다. 기본값이 아닌 다른 값을 입력할 경우에는 주의를 기울이십시오.

어드바이저 휴면 시간은 어드바이저가 모니터링하는 포트의 서버에게 상태를 묻는 횟수를 설정하고 결과를 컨설턴트에게 보고합니다. 어드바이저 휴면 시간이 너무 작으면, 어드바이저가 항상 서버를 인터럽트하므로 성능이 저하될 수 있습니다. 어드바이저 휴면 시간이 너무 크면 컨설턴트의 가중치 결정이 정확한 최신 정보를 기본으로 하지 않음을 의미할 수 있습니다.

예를 들어, HTTP 어드바이저의 간격을 3초로 설정하려면, 다음 명령을 입력하십시오.

```
xxxcontrol metriccollector set consultantID:HTTP sleeptime 3
```

어드바이저 연결 제한시간 및 서버의 수신 제한시간

서버의 특정 포트 또는 서비스가 실패했는지를 발견하기 위해 어드바이저가 사용하는 시간을 설정할 수 있습니다. 실패한 서버 제한시간 값 connecttimeout 및 receivetimeout은 어드바이저가 연결이나 수신에 실패했음을 보고하기 전에 대기하는 시간을 결정합니다.

가장 빨리 실패한 서버를 발견하려면 어드바이저 연결 및 수신 제한시간을 최소값(1초)으로 설정하고 어드바이저 및 컨설턴트 휴면 시간을 최소값(1초)으로 설정하십시오.

주: 사용자의 환경에서 통신량 및 서버 응답 시간의 moderate-to-high 볼륨이 증가하면 timeoutconnect 및 timeoutreceive 값을 너무 작게 설정하지 마십시오. 값이 너무 작으면 어드바이저가 너무 일찍 사용량이 많은 서버를 실패한 것으로 표시할 수 있습니다.

HTTP 어드바이저에 대해 `timeoutconnect`를 9초로 설정하려면 다음 명령을 입력하십시오.

```
xxxcontrol metriccollector set consultantID:HTTP timeoutconnect 9
```

연결 및 수신 제한시간의 기본값은 어드바이저 휴면 시간에 지정된 값의 세 배입니다.

어드바이저 재시도

어드바이저는 서버를 다운된 서버로 표시하기 전에 연결을 재시도할 수 있습니다. 어드바이저는 서버 조화가 재시도 횟수 더하기 1회만큼 실패할 때까지 서버를 다운된 서버로 표시하지 않습니다. 설정하지 않은 경우 재시도 값은 기본적으로 0입니다.

Cisco CSS Controller의 경우, `ccocontrol ownercontent set` 명령을 사용하여 재시도 값을 설정하십시오. 자세한 내용은 472 페이지의 『ccocontrol ownercontent — 소수 사용자 이름 및 콘텐츠 규칙 제어』를 참조하십시오.

Nortel Alteon Controller의 경우, `nalcontrol service set` 명령을 사용하여 재시도 값을 설정하십시오. 자세한 내용은 495 페이지의 『nalcontrol service — 서비스 구성』을 참조하십시오.

사용자 정의(사용자 정의 기능) 어드바이저 작성

주: 이 섹션에서 서버는 Cisco CSS Controller용 서비스 또는 Nortel Alteon Controller용 서버를 지칭하는 일반 용어로 사용됩니다.

사용자 정의(사용자 정의 기능) 어드바이저는 클래스 파일로 제공되는 Java 코드의 작은 부분으로, 기본 코드에서 호출합니다. 기본 코드는 다음과 같은 관리 서비스를 모두 제공합니다.

- 사용자 정의 어드바이저의 인스턴스 시작 및 중지
- 상태 또는 보고서 제공
- 로그 파일에 히스토리 정보 레코드

또한 컨설턴트에 결과도 보고합니다. 정기적으로 기본 코드는 개별적으로 그 구성의 모든 서버를 평가하는 어드바이저 주기를 수행합니다. 서버 시스템과의 연결을 열어 시작합니다. 소켓이 열리면, 기본 코드는 사용자 정의 어드바이저의 `getLoad` 메소드(함수)를 호출합니다. 그러면 사용자 정의 어드바이저는 서버의 상태를 평가하기 위한 필수 단계를 수행합니다. 일반적으로 서버에 사용자 정의된 메시지를 전송한 후 응답을 기다립니다. (사용자 정의 어드바이저에게 열린 소켓에 대한 액세스가 제공됩니다.) 그런 다음 기본 코드는 서버와 함께 소켓을 닫고 컨설턴트에게 로드 정보를 보고합니다.

기본 코드 및 사용자 정의 어드바이저는 표준 또는 대체 모드에서 작동될 수 있습니다. 조작 모드의 선택사항은 constructor 메소드의 매개변수로서 사용자 정의 어드바이저 파일에 지정됩니다.

표준 모드에서 사용자 정의 어드바이저는 서버와 데이터를 교환하며, 기본 어드바이저 코드는 교환 시간을 정한 후 로그값을 계산합니다. 그런 다음 기본 코드는 이 로드 값을 컨설턴트에게 보고합니다. 사용자 정의 어드바이저는 0(완료 시) 또는 마이너스 1(오류 시)만 리턴해야 합니다. 표준 모드를 지정하기 위해 구성자의 대체 플래그는 거짓으로 설정됩니다.

대체 모드에서 기본 코드는 타이밍 측정을 수행하지 않습니다. 사용자 정의 어드바이저 코드는 그 고유 요구사항에 필요한 모든 조작을 수행한 후 실제 로드 번호를 리턴합니다. 기본 코드는 번호를 승인하고 이를 컨설턴트에게 보고합니다. 최상의 결과를 위해서는 10과 1000 사이의 로드 번호를 빠른 서버를 나타내는 10과 느린 서버를 나타내는 1000으로 표준화하십시오. 대체 모드를 지정하기 위해 구성자의 대체 모드는 참으로 설정됩니다.

이 기능으로 사용자가 요구하는 서버에 관한 정확한 정보를 제공하는 사용자 고유의 어드바이저를 작성할 수 있습니다. 사용자 정의 어드바이저의 예제인 **ADV_ctrlsample.java**는 제어기에 대해 제공됩니다. Load Balancer를 설치하면 **...ibm/edge/lb/servers/samples/CustomAdvisors** 설치 디렉토리에서 예제 코드를 찾을 수 있습니다.

기본 설치 디렉토리는 다음과 같습니다.

- AIX, HP-UX, Linux, Solaris 시스템: /opt/ibm/edge/lb
- Windows 시스템: C:\Program Files\IBM\ibm\edge\lb

주: 사용자 정의 어드바이저를 Cisco CSS Controller 또는 Nortel Alteon Controller에 추가할 경우, **ccoserver** 또는 **nalserver**(Windows 시스템의 경우, 서비스 사용)를 정지한 후 재시작하여 Java 프로세스가 새 사용자 정의 어드바이저 클래스 파일을 읽을 수 있게 하십시오. 사용자 정의 어드바이저 클래스 파일은 시동 시에만 로드됩니다.

이름 지정 규칙

사용자 정의 어드바이저 파일 이름은 **ADV_myadvisor.java** 양식이어야 합니다. 이 이름은 대문자 **ADV_** 접두부로 시작해야 합니다. 모든 후속 문자는 소문자여야 합니다.

Java 규칙에 따라 파일 내에 정의된 클래스 이름은 파일 이름과 일치해야 합니다. 예제 코드를 복사할 경우, 파일 내의 모든 **ADV_ctrlsample** 인스턴스를 새로운 클래스 이름으로 변경해야 합니다.

컴파일

사용자 정의 어드바이저는 Java 언어로 작성됩니다. Load Balancer와 함께 설치된 Java 컴파일러를 사용하십시오. 다음 파일은 컴파일 시 참조됩니다.

- 사용자 정의 어드바이저 파일
- **...ibm/edge/lb/servers/lib** 설치 디렉토리에 있는 기본 클래스 파일 **ibmlb.jar**

classpath는 컴파일 시 사용자 정의 어드바이저 파일 및 기본 클래스 파일을 모두 연결해야 합니다.

Windows 플랫폼의 경우, 컴파일 명령은 다음과 같습니다.

```
install_dir/java/bin/javac -classpath  
install_dir\lb\servers\lib\ibmlb.jar ADV_pam.java
```

여기서:

- 사용자 어드바이저 파일의 이름은 **ADV_pam.java**입니다.
- 사용자 어드바이저 파일은 현재 디렉토리에 저장됩니다.

컴파일 출력은 클래스 파일입니다. 예를 들어, 다음과 같습니다.

ADV_pam.class

어드바이저를 시작하기 전에 **...ibm/edge/lb/servers/lib/CustomAdvisors** 설치 디렉토리에 클래스 파일을 복사하십시오.

주: 필요에 따라 사용자 정의 어드바이저는 한 운영 체제에서 컴파일되어 다른 운영 체제에서 실행될 수 있습니다. 예를 들어, Windows 시스템에서 어드바이저를 컴파일하고 AIX 시스템에 클래스 파일(2진)을 복사하여 여기에서 사용자 정의 어드바이저를 실행할 수 있습니다.

AIX, HP-UX, Linux 및 Solaris 시스템의 경우, 구문이 유사합니다.

실행

사용자 정의 어드바이저를 실행하려면, 먼저 클래스 파일을 적당한 설치 디렉토리에 복사해야 합니다.

```
...ibm/edge/lb/servers/lib/CustomAdvisors/ADV_pam.class
```

컨설턴트를 시작한 후 이 명령을 발행하여 사용자 정의 어드바이저를 시작하십시오.

Cisco CSS Controller

```
cococontrol ownercontent metrics consultantID:ownerContentID pam 100
```

Nortel Alteon Controller

```
nalcontrol service metrics consultantID:serviceID pam 100
```

여기서:

- pam은 ADV_pam.java의 경우와 같은 어드바이저 이름입니다.
- 100은 이 어드바이저에 제공된 가중치 비율입니다.

필수 루틴

모든 어드바이저와 마찬가지로, 사용자 정의 어드바이저는 어드바이저 기본 기능인 ADV_Base를 확장합니다. 이 기능은 컨설턴트 가중치 알고리즘에서 사용하기 위해 로드를 다시 컨설턴트에 보고하는 작업과 같이 대부분의 어드바이저 기능을 실제로 수행하는 어드바이저 기본 기능입니다. 어드바이저 기본 기능은 소켓 연결 및 닫기 조작을 수행하며 어드바이저가 사용할 송수신 메소드도 제공합니다. 어드바이저 자체는 권고 중인 서버의 포트간에 데이터 송수신에만 사용됩니다. 어드바이저 기본 기능 내의 TCP 메소드는 로드를 계산하기 위해 시간 설정됩니다. ADV_base에서 구성자 내의 플래그는 필요에 따라 기존의 로드 위에 어드바이저로부터 리턴된 새로운 로드를 겹쳐씹니다.

주: 구성자의 일련의 값에 따라, 어드바이저 기본 기능은 지정된 간격으로 가중치 알고리즘에 로드를 제공합니다. 실제 어드바이저가 유효한 로드를 리턴할 수 있도록 완료되지 않은 경우, 어드바이저 기본 기능에서는 이전의 로드를 사용합니다.

다음은 기본 클래스 메소드입니다.

- **construct** 루틴. constructor는 기본 클래스 구성자를 호출합니다(예제 어드바이저 파일 참조).
- **ADV_AdvisorInitialize** 메소드. 이 메소드는 기본 클래스가 초기화를 완료한 후 추가 단계가 수행되어야 할 경우에 후크를 제공합니다.
- **getLoad** 루틴. 기본 어드바이저 클래스는 소켓 열기를 수행하므로, getLoad는 적절한 송수신 요청을 발행하여 권고 주기를 완료해야 합니다.

탐색 순서

제어기는 먼저 제공된 고유 어드바이저 목록을 찾아봅니다. 목록에 제공된 어드바이저가 없으면 사용자 정의 어드바이저 목록을 찾아봅니다.

이름 지정 및 경로

- 사용자 정의 어드바이저 클래스는 Load Balancer 기본 디렉토리의 **...ibm/edge/lb/servers/lib/CustomAdvisors/** 하위 디렉토리 내에 있어야 합니다. 이 디렉토리의 기본값은 운영 체제에 따라 다양합니다.
 - AIX, HP-UX, Linux 또는 Solaris 시스템
/opt/ibm/edge/lb/servers/lib/CustomAdvisors/
 - Windows 시스템
C:\Program Files\IBM\edge\lb\servers\lib\CustomAdvisors
- 소문자의 영문자만 허용됩니다. 따라서 조작원은 명령행에 명령을 입력할 때 대소문자를 구별할 필요가 없습니다. 어드바이저 이름에는 **ADV_** 접두부가 있어야 합니다.

예제 어드바이저

제어기 예제 어드바이저의 프로그램 목록이 518 페이지의 『어드바이저 예제』에 나와 있습니다. 설치 후에 이 예제 어드바이저는 `...ibm/edge/lb/servers/samples/CustomAdvisors` 디렉토리에서 찾을 수 있습니다.

Metric Server

Metric Server는 Load Balancer에 서버 로드 정보를 시스템별 메트릭 양식으로 제공하여 서버의 상태를 보고합니다. Load Balancer consultant는 각 서버에 있는 Metric Server 에이전트를 조회하여 에이전트에서 수집한 메트릭을 사용하여 로드 밸런스 프로세스에 가중치를 지정합니다. 이 결과도 Cisco CSS Controller의 서비스 보고서 또는 Nortel Alteon Controller의 서버 보고서에 저장됩니다.

전제조건

Metric Server 에이전트는 로드 밸런스 중인 모든 서버에서 설치 및 실행되어야 합니다.

Metric Server 사용 방법

다음은 제어기에 대해 Metric Server를 구성하는 단계입니다.

- 제어기측

1. **ccoserver** 또는 **nalserver**를 시작하십시오.
2. Cisco CSS Controller의 경우 스위치 컨설턴트를 추가한 후 **ownercontent**를 추가하십시오.

Nortel Alteon Controller의 경우 스위치 컨설턴트를 추가한 후 서비스를 추가하십시오.

3. Metric Server 에이전트가 인식하는 포트를 지정하십시오. 이 포트는 `metricserver.cmd` 파일에 지정된 정보와 일치해야 합니다. 기본 포트는 10004입니다. 다음 명령을 사용하십시오.

Cisco CSS Controller

```
ccocontrol service set consultantID:ownerContentID:serverID  
metricserverport portNumber
```

Nortel Alteon Controller

```
nalcontrol server set consultantID:serviceID:serverID  
metricserverport portNumber
```

4. 시스템 메트릭 명령을 발행하십시오.

Cisco CSS Controller

```
cococontrol ownercontent metrics consultantID:ownerContentID  
metricName importance
```

Nortel Alteon Controller

```
nalcontrol service metrics consultantID:serviceID metricName  
importance
```

여기서 *metricName*은 메트릭 서버 스크립트의 이름입니다.

시스템 메트릭 스크립트는 백엔드 서버에 상주하며 지정된 ownercontent 또는 서비스 아래의 구성에 있는 각 서버에서 실행합니다. **cpuload**와 **memload** 두 개의 스크립트가 제공되거나 조정 시스템 메트릭 스크립트를 작성할 수 있습니다. 스크립트는 숫자 값을 리턴해야 하는 명령을 포함합니다. 이 숫자 값은 사용가능성 값이 아닌, 로드 조치를 나타냅니다.

제한사항: Windows 시스템의 경우, 시스템 메트릭 스크립트 이름에 .exe 이외의 확장자가 있으면 파일의 전체 이름(예: mySystemScript.bat)을 지정해야 합니다. 이것은 Java 코드 제한사항입니다.

5. 제어기에 대한 명령을 다음과 같이 발행하십시오.

Cisco CSS Controller

```
cococontrol consultant start
```

Nortel Alteon Controller

```
nalcontrol consultant start
```

주: 보안 확인 —

- 제어기 시스템에서 **lbkeys create** 명령을 사용하여 키 파일을 작성하십시오. lbkeys에 대한 자세한 정보는 282 페이지의 『원격 메소드 호출(RMI)』을 참조하십시오.
- 서버 시스템에서 결과 키 파일을 **...ibm/edge/lb/admin/key** 디렉토리에 복사하십시오. 키 파일의 사용 권한이 루트에서 파일을 읽을 수 있도록 허용하는지 확인하십시오.

• Metric Server 에이전트(서버 시스템)

1. Load Balancer 설치에서 Metric Server 패키지를 설치하십시오.
2. **/usr/bin** 디렉토리에서 **metricserver** 스크립트를 확인하여 원하는 RMI 포트가 사용되고 있는지 확인하십시오.(Windows 시스템의 경우, 디렉토리가 C:\WINNT\SYSTEM32입니다.) 기본 RMI 포트는 10004입니다.

주: 지정된 RMI 포트 값은 제어기 시스템에 있는 Metric Server의 RMI 포트 값과 같아야 합니다.

3. 다음의 두 스크립트 **cpuload**(0-100 범위에서 사용 중인 cpu의 비율 리턴) 및 **memload**(0-100 범위에서 사용 중인 메모리 비율 리턴)가 제공되어 있습니다. 이 스크립트는 **...ibm/edge/lb/ms/script** 디렉토리에 있습니다.

선택적으로 Metric Server에서 서버 시스템에 대해 발행할 명령을 정의하는 사용자 정의 메트릭 스크립트 파일을 작성할 수 있습니다. 사용자 정의 스크립트는 모두 실행 가능하고 **...ibm/edge/lb/ms/script** 디렉토리에 위치해야 합니다. 사용자 정의 스크립트는 숫자 로드값을 리턴해야 합니다.

주: 사용자 정의 메트릭 스크립트는 확장자가 **.bat** 또는 **.cmd**인 유효한 프로그램이나 스크립트여야 합니다. 특히 Linux 및 UNIX 시스템의 경우, 스크립트를 쉘 선언으로 시작해야 합니다. 그렇지 않으면 올바르게 실행되지 않을 수 있습니다.

4. **metricserver** 명령을 실행하여 에이전트를 시작하십시오.
5. Metric Server 에이전트를 중지하려면 **metricserver stop**을 입력하십시오.

로컬 호스트 이외의 주소에서 Metric Server를 실행하려면, 로드 밸런싱된 서버 시스템의 **metricserver** 파일을 편집하십시오. **metricserver** 파일에 있는 **java** 뒤에 다음을 삽입하십시오.

```
-Djava.rmi.server.hostname=OTHER_ADDRESS
```

또한, **metricserver** 파일에 있는 "if"문 앞에 **hostname OTHER_ADDRESS**를 추가하십시오.

Windows 시스템의 경우: Microsoft 스택에서 **OTHER_ADDRESS**의 별명을 지정합니다. Microsoft 스택에서 주소의 별명을 지정하려면 225 페이지를 참조하십시오.

작업로드 관리자 어드바이저

WLM은 MVS 메인프레임에서 실행되는 코드입니다. MVS 시스템에 있는 로드를 문의할 때 조회할 수 있습니다.

MVS 작업로드 관리가 OS/390 시스템에 구성되었으면 제어기는 WLM에서 용량 정보를 받아들이며 로드 밸런싱 프로세스에서 이 정보를 사용할 수 있습니다. 제어기는 WLM 어드바이저를 사용하여 정기적으로 컨설턴트 호스트 테이블에 있는 각 서버의 WLM 포트를 통해 연결을 열고 리턴된 용량 정수를 승인합니다. 이 정수는 여전히 사용 가능한 용량을 나타내고 컨설턴트에는 각 시스템의 로드를 나타내는 값이 필요하므로 용량 정수는 어드바이저에 의해 변환되어 로드 값으로 표준화됩니다. (예를 들어, 대용량 정수이지만 작은 로드 값은 둘다 보다 안정적인 서버를 나타냅니다.) WLM 어드바이저와 기타 제어기 어드바이저 사이의 중요한 몇 가지 차이점은 다음과 같습니다.

1. 기타 어드바이저는 표준 클라이언트 통신량이 이동되는 동일한 포트를 사용하여 서버에 대한 연결을 엽니다. WLM 어드바이저는 표준 통신량과는 다른 포트를 사용

하여 서버에 대한 연결을 엽니다. 각 서버 시스템의 WLM 에이전트는 제어기 WLM 어드바이저가 시작되는 동일한 포트에서 인식되도록 구성되어야 합니다. 기본 WLM 포트는 10007입니다.

2. WLM 어드바이저와 함께 프로토콜 고유 어드바이저를 모두 사용할 수 있습니다. 프로토콜 고유 어드바이저는 그 표준 통신 포트에서 서버를 폴링하고 WLM 어드바이저는 WLM 포트를 사용하여 시스템 로드를 폴링합니다.

서버 통제를 분석하기 위해 2진 로그 사용

2진 로그 기능으로 서버 정보를 2진 파일에 저장할 수 있습니다. 그러면 이 파일은 시간 초과 시 수집되었던 서버 정보를 분석하기 위해 프로세스될 수 있습니다.

다음 정보는 구성에서 정의된 각 서버의 2진 로그에 저장됩니다.

- 상위(Cisco CSS Controller의 경우에는 ownercontentID, Nortel Alteon Controller의 경우에는 serviceID)
- 서버 ID
- 서버 주소
- 서버 포트
- 서버 가중치
- 이 서버에 구성된 매트릭 수
- 매트릭 값의 목록

2진 로그에 정보를 로그하려면 컨설턴트가 실행 중이어야 합니다.

2진 로그 구성을 설정하기 위해 **xxxcontrol consultant binarylog** 명령을 사용합니다.

- binarylog start
- binarylog stop
- binarylog report
- binarylog set interval <seconds>
- binarylog set retention <hours>

이 시작 옵션은 로그 디렉토리에 있는 2진 로그로 서버 정보를 로그하기 시작합니다. 한 로그는 파일의 이름으로 날짜와 시간이 메시 생성됩니다.

중지 옵션은 2진 로그로 서버 정보 로그를 중지합니다. 로그 서비스는 초기값으로 중지됩니다.

설정 간격 옵션은 얼마나 자주 정보가 로그에 쓰여질 것인지를 제어합니다. 컨설턴트는 컨설턴트 간격마다 로그 서버로 서버 정보를 전송합니다. 마지막 기록이 로그에 쓰여진 이후 지정된 로그 간격 추가 경과된 경우에만 정보가 로그에 기록됩니다. 기본값으로 로그 간격은 60초로 설정됩니다.

컨설턴트 간격과 로그 간격 설정 사이에는 약간의 상호작용이 있습니다. 로그 서버에는 컨설턴트 간격(초)보다 느린 정보가 제공되므로 컨설턴트 간격 미만의 로그 간격을 설정하면 효과적으로 컨설턴트 간격과 동일한 간격을 설정할 수 있습니다.

이 로그 기술을 사용하여 미세한 서버 정보를 캡처할 수 있습니다. 서버 가중치를 계산하기 위해 컨설턴트가 표시하는 서버 정보에 대한 모든 변경사항을 캡처할 수 있습니다. 그러나 이 정보의 양이 서버 사용과 경향을 분석하는 데 필요하지는 않습니다. 60초마다 서버 정보를 로그하는 것은 시간 경과 시 서버 정보의 스냅샷을 제공합니다. 매우 느린 로그 간격 설정은 많은 양의 데이터를 생성할 수 있습니다.

보존 설정 옵션은 로그 파일이 보존되는 기간을 제어합니다. 지정된 보존 시간보다 더 오래된 로그 파일은 로그 서버에 의해 삭제됩니다. 이는 로그 서버가 컨설턴트에 의해 호출될 경우에만 발생하므로 컨설턴트를 정지하면 이전의 로그 파일은 삭제되지 않습니다.

예제 Java 프로그램 및 명령 파일이 **...ibm/edge/lb/servers/samples/BinaryLog** 디렉토리에 제공됩니다. 이 예제는 로그 파일에서 모든 정보를 검색하여 화면에 인쇄하는 방법을 보여줍니다. 이 예제는 데이터로 원하는 유형의 분석을 수행하기 위해 사용자가 정의할 수 있습니다.

다음은 제공된 스크립트 및 프로그램 사용에 관한 예제입니다.

```
xxxlogreport 2002/05/01 8:00 2002/05/01 17:00
```

이 예제는 2002년 오전 8시부터 오후 5시까지 제어기의 서버 정보 보고서를 작성합니다.

스크립트를 사용하여 경보나 레코드 서버 장애 생성

[]는 사용자 정의할 수 있는 스크립트를 트리거하는 사용자 엑시트를 제공합니다. 스크립트를 작성하여 서버의 단절을 표시할 때나 서버가 장애 이벤트를 기록할 때, 관리자에게 경보를 보내는 것과 같은 자동 조치를 수행할 수 있습니다. 사용자 정의할 수 있는 예제 스크립트는 **...ibm/edge/lb/servers/samples** 설치 디렉토리에 있습니다. 파일을 실행하려면 **...ibm/edge/lb/servers/bin** 디렉토리에서 파일을 복사한 다음 스크립트에 포함된 지시에 따라 각 파일의 이름을 변경하십시오.

다음과 같은 예제 스크립트가 제공됩니다. 여기서 **xxx**는 Cisco CSS Controller의 **cco** 및 Nortel Alteon Controller의 **nal**입니다.

- **xxxserverdown** — 제어기에서 서버를 단절 표시합니다.

- **xxxserverUp** — 제어기에서 서버가 가동됨을 표시합니다.
- **xxxallserversdown** — 특정 서비스에 대해 모든 서버를 단절 표시합니다.

제 8 부 Load Balancer 관리 및 문제점 해결

이 파트에서는 Load Balancer 관리 및 문제점 해결 정보를 제공합니다. 다음 장을 포함합니다.

- 281 페이지의 제 24 장 『Load Balancer 작동 및 관리』
- 303 페이지의 제 25 장 『문제점 해결』

제 24 장 Load Balancer 작동 및 관리

주: 이 장을 읽을 때, 컴포넌트에 특정되지 않은 일반 절에서 Dispatcher 컴포넌트를 사용하지 않는 경우에는 "dscontrol"과 "dsserver"를 다음으로 대체하십시오.

- CBR의 경우 **cbrcontrol** 및 **cbrserver** 사용
- Site Selector의 경우 **sscontrol** 및 **ssserver** 사용
- Cisco CSS Controller의 경우, **ccocontrol**과 **ccoserver**를 사용하십시오.
- Nortel Alteon Controller의 경우, **nalcontrol**과 **nalserver**를 사용하십시오.

중요: 해당 제품의 IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, 이 장의 내용을 보기 전에 제한사항 및 구성 차이점에 대해 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』 페이지를 참조하십시오.

이 장에서는 Load Balancer의 작동 및 관리 방법을 설명하며 다음 절이 수록되어 있습니다.

- 『Load Balancer의 원격 관리』
 - 282 페이지의 『원격 메소드 호출(RMI)』
 - 283 페이지의 『웹 기반 관리』
- 285 페이지의 『Load Balancer 로그 사용』
 - 285 페이지의 『Dispatcher, CBR 및 Site Selector』
 - 287 페이지의 『Cisco CSS Controller 및 Nortel Alteon Controller』
- 288 페이지의 『Dispatcher 컴포넌트 사용』
 - 290 페이지의 『Dispatcher 컴포넌트에 Simple Network Management Protocol 사용』
- 298 페이지의 『Content Based Routing 컴포넌트 사용』
- 299 페이지의 『Site Selector 컴포넌트 사용』
- 299 페이지의 『Cisco CSS Controller 컴포넌트 사용』
- 300 페이지의 『Nortel Alteon Controller 컴포넌트 사용』

Load Balancer의 원격 관리

[[는 Load Balancer가 상주하는 시스템과 별도의 시스템에서 구성 프로그램을 실행하는 두 가지 다른 방법을 제공합니다. 구성 프로그램(dscontrol, cbrcontrol, sscontrol, ccocontrol, nalcontrol)과 서버(dsserver, cbrserver 등) 사이의 통신은 다음 방법 중 하나를 사용하여 수행될 수 있습니다.

- Java RMI(Remote Method Invocation)

- 웹 기반 관리

RMI를 사용한 원격 관리의 장점은 웹 기반 관리보다 성능이 빠르다는 것입니다.

웹 기반 관리 사용의 장점은 보안되고, 인증된 원격 관리를 제공하고, 방화벽이 존재하더라도 Load Balancer 시스템과 통신할 수 있는 것입니다. 또한 이 관리 방법의 경우, Load Balancer 시스템과 통신하는 원격 클라이언트 시스템에 인증 키(lbkeys)를 설치하여 사용할 필요가 없습니다.

원격 메소드 호출(RMI)

RMI의 경우, 원격 관리를 위한 Load Balancer 시스템에 연결하기 위한 명령은 **dscontrol host:remote_host**입니다.

RMI 호출이 로컬 시스템이 아닌 시스템에서 수신되면, 공용 키/개인용 키 인증 순서는 구성 명령을 승인하기 전에 발생해야 합니다.

컴포넌트 서버와 같은 시스템에서 실행 중인 제어 프로그램간의 통신은 인증되지 않습니다.

다음 명령을 사용하여 원격 인증에 사용할 공용 키와 개인용 키를 생성하십시오.

lbkeys [creatdelete]

이 명령은 []와 동일한 시스템에서만 실행됩니다.

작성 옵션을 사용하면 서버 키 디렉토리(...ibm/edge/lb/servers/key/)에 개인용 키가 작성되고, 각 Load Balancer 컴포넌트의 관리 키 디렉토리(...ibm/edge/lb/admin/keys/)에 공용 키가 작성됩니다. 공용 키의 파일 이름은 *component-ServerAddress-RMIport*입니다. 이들 공용 키는 원격 클라이언트로 전송되고 관리 키 디렉토리에 저장되어야 합니다.

각 컴포넌트에 기본 RMI 포트를 사용하는 호스트 이름 주소가 10.0.0.25인 Load Balancer 시스템의 경우, **lbkeys create** 명령은 다음 파일을 생성합니다.

- 공용 키: ...ibm/edge/lb/servers/key/authorization.key
- 공용 키:
 - ...ibm/edge/lb/admin/keys/dispatcher-10.0.0.25-10099.key
 - ...ibm/edge/lb/admin/keys/cbr-10.0.0.25-11099.key
 - ...ibm/edge/lb/admin/keys/ss-10.0.0.25-12099.key
 - ...ibm/edge/lb/admin/keys/cco-10.0.0.25-13099.key
 - ...ibm/edge/lb/admin/keys/na1-10.0.0.25-14099.key

관리 파일 세트는 다른 시스템에 설치되었습니다. 공용 키 파일은 원격 클라이언트 시스템의 `...ibm/edge/lb/admin/keys` 디렉토리에 저장해야 합니다.

이제 10.0.0.25에 []를 구성할 수 있는 권한이 원격 클라이언트에게 부여됩니다.

이들 동일한 키는 10.0.0.25에 Load Balancer의 구성 권한을 부여할 모든 원격 클라이언트에 사용되어야 합니다.

lbkeys create 명령을 다시 실행하게 되면, 새로운 공용/개인용 키 세트가 생성됩니다. 이것은 이전 키를 사용하여 연결하려고 시도했던 원격 클라이언트에게 권한이 부여되지 않았음을 의미합니다. 새로운 키는 다시 권한을 부여할 클라이언트의 올바른 디렉토리에 저장되어야 합니다.

lbkeys delete 명령은 서버 시스템에서 개인용 및 공용 키를 삭제합니다. 이러한 키가 삭제되면, 원격 클라이언트에는 서버를 연결할 권한이 없습니다.

lbkeys 작성과 lbkeys 삭제에는 모두 강제 옵션이 있습니다. 강제 실행 옵션은 기존 키를 삭제하거나 겹쳐쓰려고 하는 경우, 질문 명령 프롬프트를 억제합니다.

RMI 접속을 구축한 후, 명령 프롬프트에서 dscontrol, cbrcontrol, sscontrol, ccocontrol, nalcontrol, dswizard, cbrwizard 및 sswizard 명령을 사용하여 구성 프로그램 간에 통신할 수 있습니다. 또한 명령 프롬프트에서 lbadm을 입력하여 GUI를 사용한 Load Balancer를 구성할 수도 있습니다.

주: Java 버전의 보안 패키지 변경사항 때문에 v5.1.1 이전 릴리스용으로 생성된 Load Balancer 키는 현재 릴리스의 키와 호환되지 않을 수 있으므로, 새 릴리스를 설치할 때 키를 다시 생성해야 합니다.

웹 기반 관리

요구사항

웹 기반 관리를 사용하려면 원격 관리를 수행하는 클라이언트 시스템에 다음이 필요합니다.

- JRE 1.3.0(또는 이상)
- 지원되는 브라우저에 대한 정보는 다음 웹 페이지 <http://www.ibm.com/support/docview.wss?rs=180&uid=swg27006921>을 참조하십시오.

주: Netscape를 사용 중일 경우, Load Balancer GUI가 나타나는 Netscape 브라우저 창의 크기를 조정하지 마십시오(최소화, 최대화, 복원 등). Netscape는 브라우저 창의 크기를 조정할 때마다 페이지를 재로드하므로 이로 인해 호스트로부터 연결이 끊어질 수 있습니다. 창의 크기를 조정할 때마다 호스트에 다시 연결해야 합니다.

원격 웹 기반 관리를 수행하기 위해 액세스하는 호스트 시스템에는 다음이 필요합니다.

- Caching Proxy V6
- Perl 5.5(또는 이상)

Caching Proxy 구성

- Caching Proxy의 경우 SSL 서버 인증서를 작성하려면 IBM 키 관리 유틸리티 (iKeyman) 또는 기타 유틸리티가 필요합니다. (인증 작성 방법에 관한 정보는 *Caching Proxy* 관리 안내서를 참조하십시오.)
- Caching Proxy 구성 파일(ibmproxy.conf)의 "Load Balancer 웹 기반 관리" 섹션에서 보호 도메인이 정의된 다음 맵핑 규칙 이전에 다음과 같은 지시문을 추가하십시오.

Windows 시스템의 경우,

```
Protect /lb-admin/lbwebaccess PROT-ADMIN
Exec /lb-admin/lbwebaccess C:\PROGRA~1\IBM\edge\lb\admin\lbwebaccess.pl
Pass /lb-admin/help/* C:\PROGRA~1\IBM\edge\lb\admin\help\*
Pass /lb-admin/*.jar C:\PROGRA~1\IBM\edge\lb\admin\lib\*.jar
Pass /lb-admin/* C:\PROGRA~1\IBM\edge\lb\admin\*
Pass /documentation/lang/* C:\PROGRA~1\IBM\edge\lb\documentation\lang\*
```

여기서 **lang**은 언어 서브디렉토리입니다(예: ko_KR).

Linux 및 UNIX 시스템의 경우,

```
Protect /lb-admin/lbwebaccess PROT-ADMIN
Exec /lb-admin/lbwebaccess /opt/ibm/edge/lb/admin/lbwebaccess.pl
Pass /lb-admin/help/* /opt/ibm/edge/lb/admin/help/*
Pass /lb-admin/*.jar /opt/ibm/edge/lb/admin/lib/*.jar
Pass /lb-admin/* /opt/ibm/edge/lb/admin/*
Pass /documentation/lang/* /opt/ibm/edge/lb/documentation/lang/*
```

주: HP-UX 시스템에서 lbwebaccess.pl 스크립트는 Perl 2진이 /usr/bin/ 디렉토리에 있는 것으로 가정합니다. (스크립트의 첫 번째 행에는 #!/usr/bin/perl이 포함되어 있습니다.) Perl 응용프로그램이 있는 위치로 디렉토리 경로를 갱신하십시오. 다른 옵션은 기호 링크를 작성하는 것입니다. 예를 들어, installed at /opt/perl/bin/perl에 Perl을 설치한 경우 다음 명령을 실행하십시오.

```
ln -s /opt/perl/bin/perl /usr/bin/perl
```

웹 기반 관리 실행 및 액세스

웹 기본 관리를 실행하려면 Load Balancer 호스트 시스템에서 시작해야 합니다. 호스트 시스템의 명령 프롬프트에서 **lbwebaccess**를 발행하십시오.

또한 원격으로 액세스하는 호스트 시스템에 대한 사용자 ID와 암호도 필요합니다. 사용자 ID와 암호는 Caching Proxy 관리 사용자 ID 및 암호와 동일합니다.

Load Balancer의 웹 기반 관리를 실행하려면 원격 위치의 웹 브라우저에서 다음 URL에 액세스하십시오.

`http://host_name/lb-admin/lbadmin.html`

여기서 `host_name`은 Load Balancer와 통신하기 위해 액세스하는 시스템의 이름입니다.

웹 페이지가 로드되면 원격 웹 기반 관리를 수행할 수 있는 Load Balancer GUI가 브라우저 창에 나타납니다.

Load Balancer GUI에서, 구성 제어 명령도 발행할 수 있습니다. GUI에서 명령을 발행하려면 다음을 수행하십시오.

1. GUI 트리에서 호스트 노드를 강조표시하십시오.
2. 호스트 팝업 메뉴에서 **명령 전송...**을 선택하십시오.
3. 명령 입력 필드에, 실행하려는 명령을 입력하십시오. 예: **executor report**. 현재 세션에서 실행되는 명령의 결과 및 히스토리가 제공된 창에 나타납니다.

원격으로 구성 새로 고치기

원격 웹 기반 관리의 경우, 다른 위치에서 Load Balancer 구성을 변경하는 여러 관리자가 있을 경우, 다른 관리자가 추가한(또는 삭제한) 클러스터, 포트 또는 서버를 보려면 구성을 새로 고쳐야 합니다. 원격 웹 기반 관리 GUI는 구성 새로 고침 및 모든 구성 새로 고침 기능을 제공합니다.

웹 기반 GUI에서, 구성을 새로 고치십시오.

- 단일 호스트의 경우: GUI 트리 구조에서 호스트 노드를 마우스 오른쪽 단추로 클릭한 다음 구성 새로 고침을 선택하십시오.
- 모든 호스트의 경우: 메뉴에서 **파일**을 선택한 다음 모든 구성 새로 고침을 선택하십시오.

Load Balancer 로그 사용

Dispatcher, CBR 및 Site Selector

[[는 서버 로그, 관리자 로그, 메트릭 모니터 로그(Metric Server 에이전트와의 통신 로그) 및 사용하는 각 어드바이저 로그에 항목을 게시합니다.

주: 또한 Dispatcher 컴포넌트에 대해서만 항목을 서브에이전트(SNMP) 로그로 만들 수 있습니다.

주: CBR(Content Based Routing) 컴포넌트는 64비트 JVM 실행 플랫폼을 제외한 모든 지원 플랫폼에서 사용 가능합니다. 혹은 Load Balancer에 있는 Dispatcher 컴

포넌트의 cbr 전달 메소드를 사용하여 Caching Proxy를 사용하지 않고도 CBR을 제공할 수 있습니다. 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 페이지를 참조하십시오.

로그 레벨을 설정하여 로그에 기록되는 메시지들의 확장성을 정의할 수 있습니다. 레벨 0에서는, 오류가 기록되고 Load Balancer도 단 한번 발생한 이벤트의 헤더 및 레코드를 기록합니다(예: 관리자 로그에 기록되기 시작한 어드바이저에 관한 메시지). 레벨 1에는 진행 중인 정보가 포함되고, 레벨 5에는 필요할 때 문제점의 디버깅을 도와주는 모든 생성 메시지가 포함됩니다. 관리자, 어드바이저, 서버 또는 서버에이전트 로그의 기본값은 1입니다.

로그의 최대 크기를 설정할 수도 있습니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서 기록된 후속 항목을 겹쳐씹니다. 로그 크기를 현재 값보다 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다.

로그 레벨을 높게 설정할 수록, 더 주의하여 로그 크기를 선택해야 합니다. 레벨 0에서는 로그 크기를 1MB의 기본값으로 두는 것이 좋습니다. 그러나 레벨 3 이상으로 로그에 기록할 경우 너무 작지 않게 크기를 제한하십시오.

- 서버 로그의 로그 레벨 또는 최대 로그 크기를 구성하려면 **dscontrol set** 명령을 사용하십시오. (서버 로그 설정을 표시하려면 **dscontrol logstatus** 명령을 사용하십시오.)
- 관리자 로그의 로그 레벨 또는 최대 로그 크기를 구성하려면 **dscontrol manager** 명령을 사용하십시오.
- Metric Server 에이전트와의 통신을 기록하는 메트릭 모니터 로그의 로그 레벨 또는 최대 로그 크기를 구성하려면 **dscontrol manager metric set** 명령을 사용하십시오.
- 어드바이저 로그의 로그 레벨 또는 최대 로그 크기를 구성하려면 **dscontrol advisor** 명령을 사용하십시오.
- 서버에이전트 로그의 로그 레벨 또는 최대 로그 크기를 구성하려면 **dscontrol subagent** 명령을 사용하십시오. (Dispatcher 컴포넌트에서만 SNMP 서버에이전트를 사용합니다).

로그 파일 경로 변경

기본적으로, Load Balancer에서 생성된 로드는 Load Balancer 설치의 로그 디렉토리에 저장됩니다. 이 경로를 변경하려면 dsserver 스크립터에서 **lb_logdir** 변수를 설정하십시오.

AIX, HP-UX, Linux 및 Solaris 시스템: dsserver 스크립트는 /usr/bin 디렉토리에 있습니다. 이 스크립트에서, **lb_logdir** 변수가 기본 디렉토리로 설정됩니다. 이 변수를 수정하여 사용자의 로그 디렉토리를 지정할 수 있습니다. 예를 들어,

LB_LOGDIR=/path/to/my/logs/

Windows 시스템: dsserver 파일은 Windows 시스템 디렉토리

C:\WINNT\SYSTEM32에 있습니다(Windows 2003의 경우). dsserver 파일에서, *lb_logdir* 변수가 기본 디렉토리로 설정됩니다. 이 변수를 수정하여 사용자의 로그 디렉토리를 지정할 수 있습니다. 예를 들어,

set LB_LOGDIR=c:\path\to\my\logs

모든 운영 체제에서는 등호 양쪽에 공백이 없어야 하며 경로는 슬래시("/") 또는 "\"로 끝나야 합니다.

2진 로그

주: 2진 로그는 Site Selector 컴포넌트에 적용되지 않습니다.

Load Balancer의 2진 로그 기능은 다른 로그 파일과 같은 로그 디렉토리를 사용합니다. 257 페이지의 『서버 통제를 분석하기 위해 2진 로그 사용』을 참조하십시오

Cisco CSS Controller 및 Nortel Alteon Controller

로그 레벨을 설정하여 로그에 기록되는 메시지들의 확장성을 정의할 수 있습니다. 레벨 0에서는 오류가 기록되고 Load Balancer도 단 한번 발생한 이벤트의 헤더 및 레코드를 기록합니다(예: 컨설턴트 로그에 기록되기 시작한 어드바이저에 관한 메시지). 레벨 1에는 진행 중인 정보가 포함되고, 레벨 5에는 필요할 때 문제점의 디버깅을 도와주는 모든 생성 메시지가 포함됩니다. 로그 기본값은 1입니다.

로그의 최대 크기를 설정할 수도 있습니다. 로그 파일의 최대 크기를 설정하면, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면, 후속 항목은 파일의 맨 위에서 이전 로그 항목 위에 겹쳐 기록됩니다. 로그 크기를 현재 값보다 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다.

로그 레벨을 높게 설정할 수록, 더 주의하여 로그 크기를 선택해야 합니다. 레벨 0에서는 로그 크기를 1MB의 기본값으로 두는 것이 좋습니다. 그러나 레벨 3 이상으로 로그에 기록할 경우 너무 작지 않게 크기를 제한하십시오.

제어기 로그

Cisco CSS Controller 및 Nortel Alteon Controller의 로그는 다음과 같습니다.

- 제어기 로그(**controller set** 명령)
- 컨설턴트 로그(**consultant set** 명령)
- highavailability 로그(**highavailability set** 명령)
- metriccollector 로그(**metriccollector set** 명령)
- 2진 로그(**consultant binarylog** 명령)

다음은 Metric Server 에이전트에 대한 통신을 로그하는 메트릭 모니터 로그의 로그 레벨과 최대 로그 크기를 구성하는 예제입니다.

```
xxxcontrol metriccollector set consultantID:serviceID:metricName  
loglevel x logsize y
```

로그 파일 경로 변경

기본적으로 제어기에서 생성된 로그는 제어기 설치의 로그 디렉토리에 저장됩니다. 이 경로를 변경하려면 xxxserver 스크립터에서 `xxx_logdir` 변수를 설정하십시오.

AIX, HP-UX, Linux 및 Solaris 시스템: xxxserver 스크립트는 `/usr/bin` 디렉토리에 있습니다. 이 스크립트에서, `xxx_logdir` 변수가 기본 디렉토리로 설정됩니다. 이 변수를 수정하여 사용자의 로그 디렉토리를 지정할 수 있습니다. 예를 들어,

```
xxx_LOGDIR=/path/to/my/logs/
```

Windows 시스템: xxxserver 파일은 Windows 시스템 디렉토리, 일반적으로 `C:\WINNT\SYSTEM32`에 있습니다. xxxserver 파일에서, `xxx_logdir` 변수가 기본 디렉토리로 설정됩니다. 이 변수를 수정하여 사용자의 로그 디렉토리를 지정할 수 있습니다. 예를 들어,

```
set xxx_LOGDIR=c:\path\to\my\logs\
```

모든 운영 체제에서는 등호 양쪽에 공백이 없어야 하며 경로는 슬래시("/") 또는 "\"로 끝나야 합니다.

2진 로그

Load Balancer의 2진 로그 기능은 다른 로그 파일과 같은 로그 디렉토리를 사용합니다. 257 페이지의 『서버 통제를 분석하기 위해 2진 로그 사용』을 참조하십시오.

Dispatcher 컴포넌트 사용

이 절에서는 Dispatcher 컴포넌트의 작동 및 관리 방법을 설명합니다.

Dispatcher 시작 및 정지

- Dispatcher를 시작하려면 명령행에서 **dsserver**를 입력하십시오.
- Dispatcher를 정지하려면 명령행에서 **dsserver stop**을 입력하십시오.

활동해제 제한시간 값 사용

Load Balancer의 경우, 활동해제 제한시간에 지정된 시간(초) 동안 접속에 아무 활동이 없으면 접속은 활동해제된 것으로 간주됩니다. 아무 활동 없이 시간(초)이 지나면 Load Balancer는 테이블에서 해당 접속 레코드를 제거하며, 해당 접속에 대한 이후의 통신량은 버려집니다.

포트 레벨에서, 예를 들어, **dscontrol port set staletimeout** 명령에서 활동해제 제한 시간 값을 지정할 수 있습니다.

활동해제 제한시간은 실행 프로그램, 클러스터 및 포트 레벨에 설정할 수 있습니다. 실행 프로그램과 클러스터 레벨에서 기본값은 300초이고 포트로 필터됩니다. 포트 레벨에서 기본값은 포트에 따라 다릅니다. 올바르게 정의된 일부 포트에는 서로 다른 활동해제 제한시간 값이 있습니다. 예를 들어, Telnet 포트 23은 기본값이 259,200초입니다.

일부 서비스에도 자체의 활동해제 제한시간 값이 있을 수 있습니다. 예를 들어, LDAP(Lightweight Directory Access Protocol)에는 **idletimeout**이라는 구성 매개변수가 있습니다. **idletimeout**초를 초과하면 대기 클라이언트 연결이 강제로 닫힙니다. **Idletimeout**은 연결을 강제로 닫을 수 없는 0으로 설정될 수도 있습니다.

Load Balancer의 활동해제 제한시간 값이 서비스의 제한시간 값보다 작을 경우 연결성 문제점이 발생할 수 있습니다. LDAP의 경우, Load Balancer 활동해제 제한시간의 기본값은 300초입니다. 300초 동안 접속에 아무 활동이 없을 경우, []는 테이블에서 접속 레코드를 제거합니다. **idletimeout** 값이 300초 이상(또는 0으로 설정)이면 클라이언트는 아직 서버에 연결이 있다고 간주합니다. 클라이언트가 패킷을 전송하면 Load Balancer가 패킷을 버립니다. 그러면, 서버에 대한 요청이 작성될 때 LDAP가 정지합니다. 이 문제점을 방지하려면 LDAP **idletimeout**을 Load Balancer 활동해제 제한시간 값과 같거나 작은 0이 아닌 값으로 설정하십시오.

fintimeout 및 staletimeout을 사용한 연결 레코드 정리 제어

클라이언트는 서버가 트랜잭션의 종료를 알 수 있도록 모든 패킷을 전송한 후 FIN 패킷을 전송합니다. Dispatcher가 FIN 패킷을 수신하면, 트랜잭션을 활성 상태에서 완료 상태로 표시합니다. 트랜잭션이 FIN으로 표시되면, 연결에 예약된 메모리를 지울 수 있습니다.

연결 레코드 할당 및 재사용 성능을 향상시키려면, **executor set fintimeout** 명령을 사용하여 Dispatcher의 FIN 상태 연결, Dispatcher 테이블에서 활성화 및 통신량 허용 유지 기간을 제어하십시오. FIN 상태 연결이 **fintimeout**을 초과하면 Dispatcher 테이블에서 연결이 해제되고 재사용 준비 상태가 됩니다. **dscontrol executor set fincount** 명령을 사용하여 FIN 제한시간을 변경할 수 있습니다.

dscontrol executor set staletimeout 명령을 사용하여 Dispatcher 테이블 통신량 비 활성화 시의 Established 상태 연결 유지 기간 및 통신량 허용 유지 기간을 제어하십시오. 288 페이지의 『활동해제 제한시간 값 사용』에서 자세한 정보를 참조하십시오.

GUI 보고 — 모니터 메뉴 옵션

실행 프로그램에서 확인하고 관리자로 릴레이하는 정보를 기반으로 다양한 도표를 표시할 수 있습니다. (GUI 모니터 메뉴 옵션은 관리자 프로그램 기능이 실행 중일 것을 요구합니다.)

- 초당 서버 연결 수(여러 서버가 같은 그래프에 표시될 수 있음)
- 특정 포트에서 서버당 상대 가중치
- 특정 포트에서 서버당 평균 연결 기간

Dispatcher 컴포넌트에 Simple Network Management Protocol 사용

네트워크 관리 시스템은 연속적으로 실행되는 프로그램으로서, 네트워크를 모니터링하고 그 상태를 반영하며 제어하는 데 사용됩니다. 네트워크의 장치와 통신하는 데 필요한 일반적인 프로토콜인 SNMP(Simple Network Management Protocol)이 현재 네트워크 관리 표준입니다. 네트워크 장치에는 일반적으로 하나의 SNMP 에이전트와 하나 이상의 서브에이전트가 있습니다. SNMP 에이전트는 **네트워크 관리 스테이션**과 대화하거나 명령행 SNMP 요청에 응답합니다. SNMP 서브에이전트는 데이터를 검색하고 갱신하며 SNMP 에이전트에 해당 데이터를 제공하여 다시 리퀘스터와 통신합니다.

Dispatcher에서는 SNMP 관리 정보 데이터베이스(ibmNetDispatcherMIB) 및 SNMP 서브에이전트를 제공합니다. 이를 통해 Tivoli® NetView®, Tivoli Distributed Monitoring 또는 HP OpenView와 같은 네트워크 관리 시스템을 사용하고 Dispatcher의 상태, 처리량 및 활동을 모니터링할 수 있습니다. MIB 데이터는 관리 중인 Dispatcher에 대해 설명하고 현재 Dispatcher 상태를 반영합니다. MIB는 **..lb/admin/MIB** 하위 디렉토리에 설치됩니다.

주: MIB, ibmNetDispatcherMIB.02는 Tivoli NetView xnmloadmib2 프로그램을 통해 로드하지 않습니다. 이러한 문제점을 수정하려면, MIB의 NOTIFICATION-GROUP 섹션에 설명을 첨부하십시오. 즉, "indMibNotifications Group NOTIFICATION-GROUP" 행 앞과 그 다음 여섯 행에 "- -" 를 삽입하십시오.

네트워크 관리 시스템은 SNMP GET 명령어를 사용하여 다른 시스템의 MIB값을 검토합니다. 그런 후 지정된 임계치가 초과되었는지 여부를 사용자에게 알릴 수 있습니다. 그러면, Dispatcher의 구성 데이터를 수정함으로써 Dispatcher 성능에 영향을 주어 Dispatcher 또는 웹 서버에 장애가 발생하기 전에 미리 Dispatcher 문제점을 조정 또는 수정할 수 있습니다.

SNMP 명령 및 프로토콜

시스템에서는 대개 각 네트워크 관리 스테이션마다 하나의 SNMP 에이전트를 제공합니다. 사용자는 GET 명령어를 SNMP 에이전트로 전송합니다. 그 다음 이 SNMP 에이전트는 GET 명령을 전송하여 이러한 MIB 변수에 책임이 있는 서브에이전트로부터 지정된 MIB 변수값을 검색합니다.

Dispatcher는 MIB 데이터를 갱신하고 검색하는 서브에이전트를 제공합니다. 서브에이전트는 SNMP 에이전트가 GET 명령을 전송할 때 해당 MIB 데이터로 응답합니다. SNMP 에이전트는 데이터를 네트워크 관리 스테이션에 전달합니다. 네트워크 관리 스테이션에서는 사용자에게 지정된 임계치의 초과 여부를 알릴 수 있습니다.

Dispatcher SNMP 지원에는 DPI®(분산 프로그램 인터페이스) 기능을 사용하는 SNMP 서브에이전트가 포함됩니다. DPI는 SNMP 에이전트와 해당 서브에이전트 사이의 인터페이스입니다. Windows 운영 체제에서는 SNMP 에이전트와 해당하는 서브에이전트 간의 인터페이스로 Windows 확장 에이전트를 사용합니다.

AIX, HP-UX, Linux 및 Solaris 시스템에서 SNMP 사용 가능

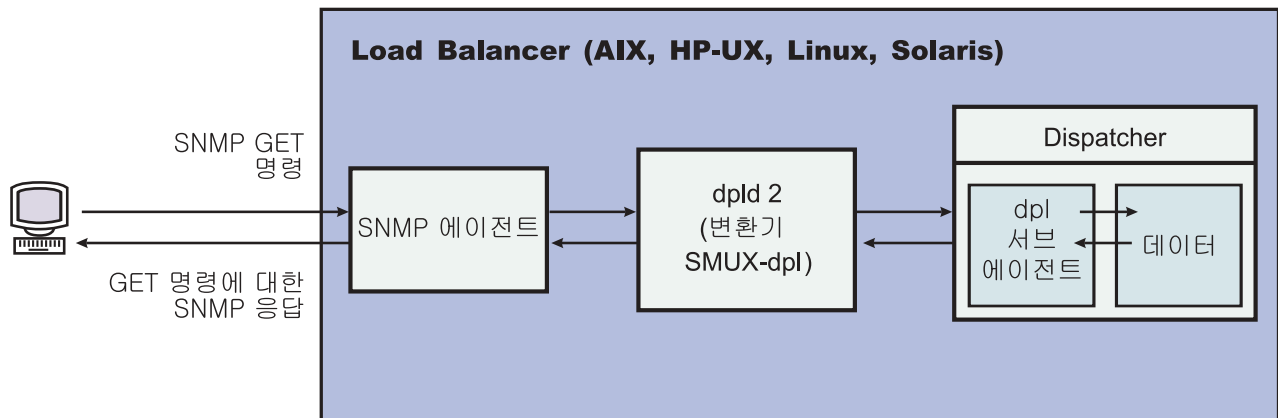


그림 40. Linux 및 UNIX 시스템의 SNMP 명령

AIX 시스템에서는 SNMP 멀티플렉서 프로토콜(SMUX)을 사용하는 SNMP 에이전트를 제공하고, DPI와 SMUX 간의 변환기로서 작동하는 추가 실행 파일 DPID2를 제공합니다.

HP-UX 시스템의 경우, HP-UX에서 SNMP 에이전트를 제공하지 않으므로 SMUX 사용 가능한 SNMP 에이전트를 가져와야 합니다. Load Balancer에서는 HP-UX 시스템용 DPID2를 제공합니다.

Linux 시스템은 SMUX를 사용하는 SNMP 에이전트를 제공합니다. 대부분의 Linux 버전(예: Red Hat)은 UCD SNMP 패키지와 함께 제공됩니다. UCD SNMP 버전 4.1 이상에는 SMUX 사용 가능 에이전트가 있습니다. Load Balancer는 Linux 시스템용 DPID2를 제공합니다.

주: SuSE Linux 시스템의 경우, SuSE가 SNMP 에이전트를 제공하지 않으므로 사용자는 SMUX 사용 가능한 NMP 에이전트를 가져와야 합니다.

Solaris 시스템의 경우, SNMP 에이전트를 제공하지 않으므로 사용자는 SMUX 사용 가능한 SNMP 에이전트를 가져와야 합니다. Load Balancer는 /opt/ibm/edge/lb/servers/samples/SNMP 디렉토리에서 Solaris 시스템용 DPID2를 제공합니다.

DPI 에이전트는 루트 사용자로서 실행되어야 합니다. DPID2 디먼을 실행하기 전에, 다음과 같이 /etc/snmpd.peers와 /etc/snmpd.conf 파일을 갱신하십시오.

AIX 및 Solaris 시스템의 경우:

- /etc/snmpd.peers 파일에서 다음의 dpid 항목을 추가하십시오.

```
"dpid2" 1.3.6.1.4.1.2.3.1.2.2.1.1.2 "dpid_password"
```

- /etc/snmpd.conf에서 다음의 dpid 항목을 추가하십시오.

```
smux 1.3.6.1.4.1.2.3.1.2.2.1.1.2 dpid_password #dpid
```

Linux 시스템의 경우:

- /etc/snmpd.peers 파일에서(시스템에 이 파일이 없을 경우 작성하십시오), dpid에 다음 항목을 추가하십시오.

```
"dpid2" 1.3.6.1.4.1.2.3.1.2.2.1.1.2 "dpid_password"
```

- /etc/snmp/snmpd.conf에서 다음의 dpid 항목을 추가하십시오.

```
smuxpeer .1.3.6.1.4.1.2.3.1.2.2.1.1.2 dpid_password
```

또한 snmpd.conf 파일에서 com2sec, group, view 또는 access로 시작하는 모든 행을 주석으로 지정해야 합니다.

HP-UX 시스템에서 SNMP 사용 가능

HP-UX SNMP 지원을 설치하려면 다음을 수행하십시오.

1. GNU SED 버전을 설치하지 않은 경우 HP 웹 사이트 <http://www.hp.com>에서 가져오십시오.
2. 다음 웹 페이지 http://sourceforge.net/project/showfiles.php?group_id=12694에서 ucd-snmp-4.2.4.tar.gz를 가져오십시오.
3. 시스템에 "gcc" 및 "gmake 또는 make"가 설치되어 있는지 확인하십시오. 설치되지 않은 경우 설치하십시오.
4. ucd-snmp-4.2.4.tar.gz 파일의 압축을 푼 다음 디렉토리의 모든 소스 파일을 untar 하십시오.
5. 소스 파일이 보존되어 있는 디렉토리로 이동하여 다음을 수행하십시오.
 - a. run ./configure --with-mib-modules=smux
 - b. make
 - c. 다음 두 명령을 루트로 실행하십시오.
 - 1) umask 022
 - 2) make install
 - d. export SNMPCONFPATH=/etc/snmp
 - e. start /usr/local/sbin/snmpd -s (SNMP 에이전트가 시작됩니다)
 - f. start dpid2 (DPI 변환기가 시작됩니다)
 - g. dscontrol subagent start (Dispatcher 서브에이전트가 시작됩니다)

SuSE Linux 시스템에서 SNMP 사용 가능

SuSE Linux 시스템에서 Load Balancer SNMP를 사용하려면 다음을 수행해야 합니다.

1. SuSE 시스템에서 설치된 ucd-snmp를 제거하십시오.
2. http://sourceforge.net/project/showfiles.php?group_id=12694에서 ucd-snmp-4.2.4.tar.gz를 확보하십시오.
3. SuSE 시스템에 "gcc" 및 "gmake or make"가 설치되어 있는지 확인하십시오. (설치되어 있지 않으면 설치해야 합니다.)
4. ucd-snmp-4.2.4.tar.gz 파일의 압축을 푼 다음 디렉토리의 모든 소스 파일을 untar 하십시오.
5. 소스 파일이 보존되어 있는 디렉토리로 이동하여 다음을 수행하십시오.
 - a. run ./configure --with-mib-modules=smux
 - b. make
 - c. 다음 두 명령을 루트로 실행하십시오.
 - 1) umask 022 #
 - 2) make install
 - d. export SNMPCONFPATH=/etc/snmp
 - e. start /usr/local/sbin/snmpd -s
 - f. start dpid2

snmpd.conf 파일을 다시 읽도록 snmpd를 새로 고치십시오(이미 실행 중일 경우).

```
refresh -s snmpd
```

DPID SMUX 피어를 시작하십시오.

```
dpid2
```

디먼은 다음 순서로 시작되어야 합니다.

1. SNMP 에이전트
2. DPI 변환기
3. Dispatcher 서브에이전트

Solaris 시스템에서 SNMP 사용 가능

Solaris SNMP 지원을 설치하려면 다음을 수행하십시오.

1. 실행 중인 Solaris SNMP 디먼(snmpdx 및 snmpXdmid)을 종료하십시오.
2. 파일 이름을 다음과 같이 바꾸십시오.

`/etc/rc3.d/S76snmpdx`에서 `/etc/rc3.d/K76snmpdx`로

`/etc/rc3.d/S77dmi`에서 `/etc/rc3.d/K77dmi`로

3. 다음 패키지를 <http://www.sunfreeware.com/>에서 다운로드하십시오.
 - libgcc-3.0.3-sol8-sparc-local(SMClibgcc)
 - openssl-0.9.6c-sol8-sparc-local(SMCssl)
 - popt-1.6.3-sol8-sparc-local(SMCpopt)
4. pkgadd를 사용하여 다운로드한 패키지를 설치하십시오.
5. http://sourceforge.net/project/showfiles.php?group_id=12694에서 ucd-snmp-4.2.3-solaris8.tar.gz를 다운로드하십시오.
6. 루트 디렉토리(/)에서 ucd-snmp-4.2.3-solaris8.tar.gz를 압축 해제하십시오.
7. 다음 명령을 실행하십시오.

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:
/usr/local/lib:/usr/local/ssl/lib:/usr/lib

export PATH=/usr/local/sbin:/usr/local/bin:$PATH

export SNMPCONFPATH =/etc/snmp

export MIBDIRS=/usr/local/share/snmp/mibs

cp /opt/ibm/edge/lb/servers/samples/SNMP/dpid2 /usr/local/sbin/dpid2
```

8. /etc/snmpd.peers가 아직 없으면 작성하십시오. 다음을 snmpd.peers에 삽입하십시오.

```
"dpid2" 1.3.6.1.4.1.2.3.1.2.2.1.1.2 "dpid_password"
```

9. /etc/snmp/snmpd.conf가 아직 없으면 작성하십시오. 다음을 snmpd.conf에 삽입하십시오.

```
smuxpeer 1.3.6.1.4.1.2.3.1.2.2.1.1.2 dpid_password
```

10. /usr/local/sbin/snmpd를 시작하십시오.
11. /usr/local/sbin/dpid2를 시작하십시오.

주:

1. 패키지 형식은 다음과 같습니다.

- libgcc-3.0.3-sol8-sparc-local(SMClibgcc)
- openssl-0.9.6c-sol8-sparc-local(SMCssl)
- popt-1.6.3-sol8-sparc-local(SMCpopt)

<http://sunfreeware.com/> 웹 사이트에서 이름은 .gz 확장자를 가지므로, 압축 해제 하지 마십시오. 대신, pkgadd *packageName*을 사용하십시오.

2. smuxpeer 항목을 /etc/snmp/snmpd.conf에 추가할 때 **dpid_password** 문자열에 공백이 들어가지 않게 하십시오.
3. Load Balancer SNMP 기능은 smux-enabled ucd-snmp 버전 4.2.3에서 테스트되었습니다. smux가 있는 ucd-snmp의 향후 릴리스는 비슷한 설정에서 작동합니다.

Windows 운영 체제에서 SNMP 사용 가능

Windows SNMP 지원을 설치하려면 다음을 수행하십시오.

1. 시작 > 설정 (Windows 2000) > 제어판 > 프로그램 추가/제거를 클릭하십시오.
2. **Windows** 컴포넌트 추가/제거를 클릭하십시오.
3. Windows 컴포넌트 마법사에서 관리 및 모니터링 도구를 클릭한 다음(그러나 해당 선택란을 선택하거나 선택 취소하지 않음) 세부사항을 클릭하십시오.
4. 단순 네트워크 관리 프로토콜(SNMP) 선택란을 선택한 다음 확인을 클릭하십시오.
5. 다음을 클릭하십시오.

SNMP 공동체 이름 제공

실행 프로그램을 실행하면서 **dscontrol subagent start [communityname]** 명령을 사용하여 Windows OS Extension 에이전트와 SNMP 에이전트 간에 사용된 공동체 이름을 정의하십시오.

중요: Windows 2003에서 기본적으로 SNMP는 제공된 공동체 이름에 응답하지 않습니다. 그런 경우 SNMP 서버에이전트가 SNMP 요청에 응답하지 않습니다. SNMP 서버에이전트가 공동체 이름에 응답하도록 하려면 해당하는 공동체 이름과 대상 호스트를 사용하여 SNMP 서비스 등록 정보를 설정하십시오. SNMP 보안 등록 정보를 다음과 같이 구성하십시오.

1. 컴퓨터 관리를 여십시오.
2. 콘솔 트리에서 서비스를 클릭하십시오.
3. 세부사항 분할창에서 **SNMP** 서비스를 클릭하십시오.
4. 조치 메뉴에서 등록 정보를 클릭하십시오.
5. 보안 탭의 승인된 공동체 이름에서 추가를 클릭하십시오.
6. 공동체 권한에서 이 호스트의 권한 레벨을 선택하여 선택된 공동체에서 SNMP 요청을 처리하십시오(최소한 읽기 전용 권한).
7. 공동체 이름에서 Load Balancer 서버에이전트에 제공한 것(기본 공동체 이름: public)과 같은 대소문자를 구분하는 공동체 이름을 입력한 다음 추가를 클릭하십시오.
8. 호스트의 SNMP 패킷을 승인할지 여부를 지정하십시오. 다음 옵션 중 하나를 선택하십시오.
 - 네트워크의 호스트에서 SNMP 요청을 승인하려면 ID와 상관없이 모든 호스트에서 **SNMP** 패킷 승인을 클릭하십시오.(이 옵션을 사용하여 암호 또는 인증서 같은 기준에 따라 인증을 이용하여 개인 또는 엔티티를 검증해야 합니다.)
 - SNMP 패킷 승인을 제한하려면 **SNMP** 패킷 승인 제한, 이 호스트에서 **SNMP** 패킷 승인, 추가를 차례로 클릭하십시오. 적합한 호스트 이름, IP 또는 IPX 주소를 입력하고 각각 입력한 후에 추가를 클릭하십시오.
9. 변경 사항을 적용하려면 SNMP 서비스를 다시 시작하십시오.

트랩

SNMP는 트랩을 송수신하여 통신하며, 예외 조건이나 중대한 이벤트의 발생을 보고하기 위해 관리되는 장치가 전달하는 메시지입니다.

서브에이전트는 다음과 같은 트랩을 사용합니다.

- indHighAvailStatus
- indSrvrGoneDown
- indDOSAttack
- indDOSAttackDone

indHighAvailStatus 트랩은 고가용성 상태의 변수값(hasState)을 변경하여 고가용성 상태의 값이 변경되었다는 것을 알립니다. hasState의 가능한 값은 다음과 같습니다.

-idle 이 시스템이 로드 밸런스를 수행 중이며 상대 Dispatcher와의 접속을 설정하려고 하지 않습니다.

-listen 고가용성이 방금 시작되었으며 Dispatcher가 그 상대를 인식하고 있습니다.

-active

이 시스템이 로드 밸런스를 수행 중입니다.

-standby

이 시스템이 활성화된 시스템을 모니터링하고 있습니다.

-preempt

이 시스템은 기본에서 백업으로 전환되는 동안 임시 상태로 있습니다.

-elect Dispatcher가 기본 또는 백업이 될 상대와 조정 중입니다.

-no_exec

실행 프로그램이 실행 중이 아닙니다.

indSrvrGoneDown 트랩은 오브젝트 ID의 csID(클러스터 ID), psNum(포트 번호) 및 ssID(서버 ID) 부분에 의해 지정된 서버의 가중치가 0이 되었다는 것을 알립니다. 서버에 대한 활성 연결의 마지막으로 알려진 번호가 트랩으로 전송되었습니다. Dispatcher가 판별할 수 있는 경우 이 트랩은 지정된 서버가 단절되었다는 것을 나타냅니다.

indDOSAttack 트랩은 numhalfopen 즉, SYN 패킷만으로 구성된 반개방 연결의 수가 오브젝트 ID의 csID(클러스터 ID) 및 psNum(포트 번호) 부분에 의해 지정된 포트의 maxhalfopen 임계치를 초과했음을 나타냅니다. 포트에서 구성된 서버 수가 트랩에서 전송됩니다. 이 트랩은 Load Balancer에서 서비스 거부 중지가 발생 중임을 나타냅니다.

indDOSAttackDone 트랩은 numhalfopen 즉, SYN 패킷만으로 구성된 반개방 연결의 수가 오브젝트 ID의 csID 및 psNum 부분에 의해 지정된 포트의 maxhalfopen 임

계치보다 작음을 나타냅니다. 포트에서 구성된 서버 수가 트랩에서 전송됩니다. Load Balancer가 가능한 서비스 거부 중지가 종료된 것을 판별하면 indDOSAttack 트랩이 전송된 후 이 트랩이 전송됩니다.

Linux 및 UNIX 시스템의 경우, SMUX API의 제한으로 인해 ibmNetDispatcher 서버 에이전트로부터 트랩에 보고되는 엔터프라이즈 ID가 ibmNetDispatcher, 1.3.6.1.4.1.2.6.144가 아닌 dpid2의 엔터프라이즈 ID일 수 있습니다. 그러나 데이터에 ibmNetDispatcher MIB 내의 오브젝트 ID가 포함되므로, 데이터에 SNMP 관리 유틸리티는 트랩의 소스를 판별할 수 있습니다.

dscontrol 명령으로 SNMP 지원 작동 및 작동 중지

dscontrol subagent start 명령으로 SNMP 지원이 작동됩니다. **dscontrol subagent stop** 명령으로 SNMP 지원이 작동 중지됩니다.

dscontrol 명령에 대한 자세한 정보는 427 페이지의 『dscontrol subagent — SNMP 서버 에이전트 구성』을 참조하십시오.

ipchain 또는 iptable을 사용하여 Load Balancer 시스템(Linux 시스템)을 굳히는 모든 통신량을 거절함

Linux 커널에 ipchain이라는 방화벽 설비가 빌드됩니다. []와 ipchain이 동시에 실행 되면 Load Balancer가 먼저 패킷을 수신한 다음 ipchain이 수신합니다. 이렇게 하면 ipchain을 사용하여 Linux Load Balancer 시스템(예: 방화벽 로드 밸런스에 사용되는 Load Balancer 시스템)을 견고히 할 수 있습니다.

ipchain 또는 iptable이 완전히 제한적으로 구성된 경우(인바운드 또는 아웃바운드 통신량이 허용되지 않는 경우), Load Balancer의 패킷 전달 부분은 정상적으로 작동합니다.

ipchain 및 iptable은 로드 밸런싱되기 전에 수신 통신량을 필터하는데 사용할 수 없습니다.

모든 Load Balancer가 정상적으로 작동하려면 추가 통신이 허용되어야 합니다. 이 통신의 몇 가지 예제가 나와 있습니다.

- 어드바이저는 Load Balancer 시스템과 백엔드 서버 사이에서 통신합니다.
- Load Balancer는 백엔드 서버, 도달 목표 및 고가용성 상대 Load Balancer 시스템을 ping합니다.
- 사용자 인터페이스(그래픽 사용자 인터페이스, 명령행 및 마법사)는 RMI를 사용합니다.
- 백엔드 서버는 Load Balancer 시스템에서 ping에 응답해야 합니다.

일반적으로 Load Balancer 시스템에 대한 적절한 inchain 전략은 백엔드 서버와 연관된 것을 제외한 모든 통신량, 대상 고가용성 Load Balancer, 모든 도달 목표 및 구성 호스트를 허용하지 않는 것입니다.

Linux 커널 버전 2.4.10.x에서 Load Balancer를 실행할 경우 iptable을 활성화하지 않도록 권장합니다. 이 Linux 커널 버전에서 활성화할 경우 시간이 지남에 따라 성능 저하가 발생할 수 있습니다.

iptables을 비활성화하려면, 모듈을 나열하여(lsmod) ip_tables 및 ip_conntrack을 사용 중인 모듈을 본 다음, rmmod ip_tables 및 rmmod ip_conntrack을 발행하여 제거하십시오. 시스템을 재부팅하면 이들 모듈이 다시 추가되며, 따라서 재부팅할 때마다 이들 단계를 반복해야 합니다.

자세한 내용은 346 페이지의 『문제점: Linux iptables가 패킷 라우팅에 간섭할 수 있음』을 참조하십시오.

Content Based Routing 컴포넌트 사용

이 절에서는 Load Balancer의 CBR 컴포넌트 작동 및 관리 방법에 대해 설명합니다.

주: CBR(Content Based Routing) 컴포넌트는 64비트 JVM 실행 플랫폼을 제외한 모든 지원 플랫폼에서 사용 가능합니다. 혹은 Load Balancer에 있는 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 Caching Proxy를 사용하지 않고도 CBR을 제공할 수 있습니다. 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 페이지를 참조하십시오.

CBR 시작 및 중지

- 명령행에 **cbrserver**를 입력하여 CBR을 시작하십시오.
- 명령행에 **cbrserver stop**을 입력하여 CBR을 중지하십시오.

CBR 및 Caching Proxy는 Caching Proxy 플러그인 API를 사용하여 배치되어 HTTP 및 HTTPS(SSL) 요청을 처리합니다. 로드 밸런스 서버를 시작하려면, Caching Proxy가 동일한 시스템에서 실행되어야 합니다. 133 페이지의 『CBR 구성 예제』에서 설명하는 대로 CBR과 Caching Proxy를 설정하십시오.

CBR 제어

CBR을 시작한 후에는 다음 방법을 사용하여 제어할 수 있습니다.

- **cbrcontrol** 명령을 통해 CBR을 구성하십시오. 이 명령의 완전한 구문은 369 페이지의 제 27 장 『Dispatcher 및 CBR 명령어 참조서』에 설명되어 있습니다. 몇 가지의 예제가 여기에 나열되어 있습니다.

- GUI(Graphical User Interface)를 사용하여 CBR을 구성하십시오. GUI를 열려면 명령행에 **ladmin**을 입력하십시오. GUI를 사용한 CBR 구성 방법에 관한 자세한 정보는 126 페이지의 『GUI』를 참조하십시오.

CBR 로그 사용

CBR이 사용하는 로그는 Dispatcher에서 사용하는 로그와 유사합니다. 자세한 내용은 285 페이지의 『Load Balancer 로그 사용』을 참조하십시오.

주:

CBR의 이전 릴리스에서는 Caching Proxy 구성 파일에서 로그 디렉토리 경로를 변경할 수 있습니다. 이제, cbrserver 파일에서 로그가 저장된 디렉토리 경로를 변경할 수 있습니다. 288 페이지의 『로그 파일 경로 변경』을 참조하십시오.

Site Selector 컴포넌트 사용

Site Selector 시작 및 정지

- 명령행에 **ssserver**를 입력하여 Site Selector를 시작하십시오.
- 명령행에 **ssserver stop**을 입력하여 Site Selector를 정지하십시오.

Site Selector 제어

Site Selector를 시작한 후에는 다음 방법을 사용하여 제어할 수 있습니다.

- **sscontrol** 명령을 통해 Site Selector를 구성하십시오. 이 명령의 완전한 구문은 429 페이지의 제 28 장 『Site Selector 명령어 참조서』에 설명되어 있습니다. 몇 가지의 예제가 여기에 나열되어 있습니다.
- GUI(Graphical User Interface)를 사용하여 Site Selector를 구성하십시오. GUI를 열려면 명령행에 **ladmin**을 입력하십시오. GUI를 사용한 Site Selector의 구성 방법에 대한 자세한 정보는 147 페이지의 『GUI』를 참조하십시오.

Site Selector 로그 사용

Site Selector가 사용하는 로그는 Dispatcher에서 사용된 로그와 비슷합니다. 자세한 설명은 285 페이지의 『Load Balancer 로그 사용』을 참조하십시오.

Cisco CSS Controller 컴포넌트 사용

Cisco CSS Controller 시작 및 정지

1. 명령행에 **ccoserver**를 입력하여 Cisco CSS Controller를 시작하십시오.
2. 명령행에 **ccoserver stop**을 입력하여 Cisco CSS Controller를 정지하십시오.

Cisco CSS Controller 제어

Cisco CSS Controller를 시작한 후에는 다음 방법을 사용하여 제어할 수 있습니다.

- **cococontrol** 명령을 통해 Cisco CSS Controller를 구성하십시오. 이 명령의 완전한 구문은 457 페이지의 제 29 장 『Cisco CSS Controller 명령어 참조서』에 설명되어 있습니다. 몇 가지의 예제가 여기에 나열되어 있습니다.
- GUI(Graphical User Interface)를 사용하여 Cisco CSS Controller를 구성하십시오. GUI를 열려면 명령행에 **ladmin**을 입력하십시오. GUI를 사용한 Cisco CSS Controller 구성 방법에 대한 자세한 정보는 165 페이지의 『GUI』의 내용을 참조하십시오.

Cisco CSS Controller 로그 사용

Cisco CSS Controller에서 사용하는 로그는 Dispatcher에서 사용하는 로그와 유사합니다. 자세한 설명은 285 페이지의 『Load Balancer 로그 사용』을 참조하십시오.

Nortel Alteon Controller 컴포넌트 사용

Nortel Alteon Controller 시작 및 중지

1. 명령행에 **nalserver**를 입력하여 Nortel Alteon Controller를 시작하십시오.
2. 명령행에 **nalserver stop**을 입력하여 Nortel Alteon Controller를 중지하십시오.

Nortel Alteon Controller 제어

Nortel Alteon Controller를 시작한 후에는 다음 방법을 사용하여 제어할 수 있습니다.

- **nalcontrol** 명령을 통해 Nortel Alteon Controller를 구성하십시오. 이 명령의 완전한 구문은 477 페이지의 제 30 장 『Nortel Alteon Controller 명령어 참조서』에 설명되어 있습니다. 몇 가지의 예제가 여기에 나열되어 있습니다.
- GUI(Graphical User Interface)를 사용하여 Nortel Alteon Controller를 구성하십시오. GUI를 열려면 명령행에 **ladmin**을 입력하십시오. GUI를 사용한 Nortel Alteon Controller 구성 방법에 관한 자세한 정보는 187 페이지의 『GUI』의 내용을 참조하십시오.

Nortel Alteon Controller 로그 사용

Nortel Alteon Controller에서 사용하는 로그는 Dispatcher에서 사용하는 로그와 유사합니다. 자세한 설명은 285 페이지의 『Load Balancer 로그 사용』을 참조하십시오.

Metric Server 컴포넌트 사용

Metric Server 시작 및 정지

Metric Server는 Load Balancer에 서버 로드 정보를 제공합니다. Metric Server는 로드 밸런싱되는 각 서버에 있습니다.

Linux 및 UNIX 시스템:

- Metric Server가 있는 각각의 서버 시스템의 명령행에서 **metricsserver start**를 입력하여 Metric Server를 시작하십시오.
- Metric Server가 있는 각각의 서버 시스템의 명령행에서 **metricsserver stop**을 입력하여 Metric Server를 중지하십시오.

Windows 시스템:

시작 > 설정(Windows 2000의 경우) > 제어판 > 관리 도구 > 서비스를 클릭하십시오. IBM Metric Server를 마우스 오른쪽 단추로 클릭한 다음 시작을 선택하십시오. 서비스를 중지하려면 동일한 단계를 수행한 후 정지를 선택하십시오.

Metric Server 로그 사용

Metric Server 시동 스크립트의 로그 레벨을 변경하십시오. Load Balancer 로그의 로그 레벨 범위와 마찬가지로 0 - 5의 로그 레벨 범위를 지정할 수 있습니다. 이렇게 하면 **...ms/logs** 디렉토리에 에이전트 로그가 생성됩니다.

제 25 장 문제점 해결

이 장은 []와 관련된 문제점을 발견하고 해결하는 데 도움이 됩니다.

- IBM 서비스를 호출하기 전에 『문제점 해결 정보 집적』을 참조하십시오.
- 308 페이지의 『문제점 해결 테이블』에는 발생할 수 있는 증상이 나와 있습니다.

문제점 해결 정보 집적

이 절의 정보를 사용하여 IBM 서비스에서 요구하는 데이터를 수집하십시오. 정보는 다음 주제로 나뉘어져 있습니다.

- 『일반 정보(항상 필수)』
- 305 페이지의 『고가용성(HA) 문제점』
- 305 페이지의 『어드바이저 문제점』
- 306 페이지의 『Content Based Routing 문제점』
- 307 페이지의 『클러스터를 히트할 수 없음』
- 307 페이지의 『기타 모두 장애』
- 308 페이지의 『업그레이드』
- 308 페이지의 『유용한 링크』

일반 정보(항상 필수)

Dispatcher 컴포넌트의 경우에만, 운영 체제에 고유한 데이터 및 컴포넌트 지정 구성 파일을 자동으로 집적하는 문제점 판별 도구가 있습니다. 이 도구를 실행하려면 적절한 디렉토리에서 **lbpd**를 입력하십시오.

Linux 및 UNIX 시스템의 경우: /opt/ibm/edge/lb/servers/bin/

Windows 시스템의 경우: C:\Program Files\IBM\edge\lb\servers\bin

이 문제점 판별 도구는 데이터를 다음과 같이 파일로 패키징합니다.

Linux 및 UNIX 시스템의 경우: /opt/ibm/edge/lb/**lbpmr.tar.Z**

Windows 시스템의 경우: C:\Program Files\IBM\edge\lb**lbpmr.zip**

주: Windows 시스템용 명령행 zip 유틸리티가 있어야 합니다.

IBM 서비스를 호출하기 전에 다음 정보를 준비하십시오.

- Dispatcher의 경우에만, 위에서 설명한 문제점 판별 도구로 집적된 lbpmr 파일
- 고가용성 환경에서, 두 Load Balancer 시스템의 구성 파일. 모든 운영 체제에서, 구성 로드에서 사용하는 스크립트를 사용하거나 다음 명령을 발행하십시오.

```
dscontrol file save primary.cfg
```

이 명령은 `.../ibm/edge/lb/servers/configuration/component/` 디렉토리에 구성 파일을 배치합니다.

- 실행 중인 운영 체제 및 그 운영 체제의 버전
- Load Balancer의 버전.
 - Load Balancer가 실행 중인 경우, 다음 명령을 발행하십시오.
 - Dispatcher 컴포넌트의 경우: `dscontrol executor report`
 - CBR의 경우: `cbrcontrol executor status`
 - Site Selector의 경우 `.../ibm/edge/lb/servers/logs/ss/`에 위치한 `server.log` 파일의 시작을 확인하십시오.
 - Cisco CSS Controller 및 Nortel Alteon Controller의 경우: `xxxcontrol controller report`
 - 다음 명령을 발행하여 Load Balancer의 설치를 확인하고 Load Balancer의 현재 레벨을 획득하십시오.
 - AIX 시스템: `lspp -l | grep ibmlb`
 - HP-UX 시스템: `swlist | grep ibmlb`
 - Linux 시스템: `rpm -qa | grep ibmlb`
 - Solaris 시스템: `pkginfo | grep ibm`

Windows 시스템에서 Load Balancer의 설치를 확인하려면 다음 메뉴를 사용하십시오. 시작 > 설정 > 제어판 > 프로그램 추가 및 제거입니다.

- Java의 현재 레벨을 얻으려면 다음 명령을 발행하십시오.

```
java -fullversion
```
- 토큰링 또는 이더넷 사용?
- 다음 명령 중 하나를 사용하여 프로토콜 통계 및 TCP/IP 접속 정보를 확보하십시오.

AIX, HP-UX, Linux 및 Solaris 시스템: `netstat -ni`

Windows 시스템: `ipconfig /all`

이것은 모든 서버 및 Load Balancer에서 필수입니다.

- 라우트 테이블 정보를 확보하려면 이들 명령 중 하나를 발행하십시오.

AIX, HP-UX, Linux 및 Solaris 시스템: `netstat -nr`

Windows 시스템: `route print`

이것은 모든 서버 및 Load Balancer에서 필수입니다.

고가용성(HA) 문제점

HA 환경에서 문제점에 대해 다음의 필수 정보를 집적하십시오.

- loglevel 5의 hamon.log 설정: `dscontrol set loglevel 5`.
- loglevel 5의 reach.log 설정: `dscontrol manager reach set loglevel 5`
- 다음과 같은 위치의 스크립트를 확보하십시오.

AIX, HP-UX, Linux 및 Solaris 시스템: `/opt/ibm/edge/lb/servers/bin`

Windows 시스템: `C:\Program Files\ibm\edge\lb\servers\bin`

스크립트 이름은 다음과 같습니다.

`goActive`

`goStandby`

`goIdle` (있을 경우)

`goInOp` (있을 경우)

또한 구성 파일을 포함합니다. 303 페이지의 『일반 정보(항상 필수)』를 참조하십시오.

어드바이저 문제점

어드바이저가 실수로 서버를 작동 중단 상태로 표시할 경우와 같이 어드바이저 문제점에 대해 다음의 필수 정보를 집적하십시오.

- loglevel 5의 어드바이저 로그 설정:

`dscontrol advisor loglevel http 80 5`

또는

`dscontrol advisor loglevel advisorName port loglevel`

또는

`dscontrol advisor loglevel advisorName cluster:port loglevel`

또는

`nalcontrol metriccollector set consultantID:serviceID:metricName
loglevel value`

그러면 이름이 `ADV_advisorName.log` 로그가 작성됩니다. 예를 들어, `ADV_http.log`입니다. 이 로그는 다음 위치에 있습니다.

AIX, HP-UX, Linux 및 Solaris 플랫폼: `/opt/ibm/edge/lb/servers/logs/component`

Windows 플랫폼: `C:\Program Files\ibm\edge\lb\servers\logs\component`

여기서 *component*는 다음과 같습니다.

dispatcher = Dispatcher

cbr = Content Based Routing
cco = Cisco CSS Controller
nal = Nortel Alteon Controller
ss = Site Selector

주: 사용자 정의 어드바이저를 기록할 경우 어드바이저가 제대로 작동하는지 확인하는데 `ADVLOG(loglevel,message)`를 사용하는 것이 도움이 됩니다.

레벨이 어드바이저와 연관된 로깅 레벨보다 작을 경우 `ADVLOG` 호출에서 명령문을 어드바이저 로그 파일에 인쇄합니다. 로깅 레벨이 0이면 명령문을 항상 기록합니다. 생성자에서 `ADVLOG`를 사용할 수 없습니다. 로그 파일 이름에서 생성자에 설정된 정보를 사용하기 때문에 사용자 정의 어드바이저의 생성자가 완료된 후 바로 로그 파일이 작성됩니다.

이 제한사항을 방지하는 사용자 정의 어드바이저를 디버그할 수 있는 또다른 방법이 있습니다. `System.out.println(message)`문을 사용하여 메시지를 창에 인쇄할 수 있습니다. 인쇄문을 창에 표시하려면 `dsserver` 스크립트를 편집하고 `javaw`를 `java`로 변경하십시오. 인쇄를 표시하려면 `dsserver`를 시작하기 위해 사용한 창을 열어두어야 합니다. Windows 플랫폼을 사용할 경우, 서비스로 실행 중인 `Dispatcher`를 정지하고 창에서 수동으로 시작하여 메시지를 표시하십시오.

`ADVLOG`에 대한 자세한 정보는 *Edge Components 프로그래밍 안내서*를 참조하십시오.

Content Based Routing 문제점

Content Based Routing 문제점에 대해 다음 필수 정보를 집적하십시오.

- 다음 명령을 발행하여 버전을 알아보십시오. `cbrcontrol executor status`.
- 다음 파일을 확보하십시오.
 - 다음에 위치한 `ibmproxy.conf`:
 - Linux 및 UNIX 시스템: `/etc/`
 - Windows 시스템: `C:\Program Files\IBM\edge\cp\etc\en_US\`
 - 다음에 위치한 CBR 구성 파일:
 - Linux 및 UNIX 시스템: `/opt/ibm/edge/lb/servers/configurations/cbr`
 - Windows 시스템: `C:\Program Files\IBM\edge\lb\servers\configurations\cbr`
 - 올바른 입력이 `ibmproxy.conf`에 작성되었는지 확인하십시오. 128 페이지의 『1단계. CBR을 사용하기 위해 Caching Proxy 구성』을 참조하십시오.

클러스터를 히트할 수 없음

클러스터를 히트할 수 없을 경우, Load Balancer 시스템 둘 다에서 클러스터의 별명이 지정되었거나 지정되지 않았을 가능성이 있습니다. 클러스터를 소유하는 시스템을 판별하려면 다음과 같이 실행하십시오.

1. 동일한 서브넷의 Load Balancer 시스템 또는 서버가 아닌 곳에서 다음을 실행하십시오.

```
ping cluster  
arp -a
```

Dispatcher의 nat 또는 cbr 전달 메소드를 사용하는 경우, 리턴 주소도 ping하십시오.

2. arp 출력을 보고 MAC(16자리 16진 주소)를 netstat -ni 출력 중 하나에 일치시켜 물리적으로 클러스터를 소유하는 시스템을 판별하십시오.
3. 두 시스템이 모두 클러스터 주소를 가지고 있는지 알아보기 위해 두 시스템의 출력을 해석하려면 다음 명령을 사용하십시오.

AIX 및 HP-UX 시스템: netstat -ni

Linux 및 Solaris 시스템: ifconfig -a

Windows 시스템: ipconfig /all

ping으로부터 응답이 없으며 ULB를 사용하지 않는 경우, 두 시스템 모두 인터페이스에 클러스터 IP 주소의 별명(예: en0, tr0)이 지정되어 있지 않을 가능성이 있습니다.

주: IPv4 및 IPv6용 Load Balancer 설치를 실행 중인 Linux 시스템의 경우 ping에서 응답을 얻지 못하면 백엔드 서버 사용이 불가능하다는 뜻이지만, arp 항목은 계속 갱신되어야 합니다. 그밖에 arping이 사용될 수도 있습니다.

기타 모두 장애

경로 지정 문제점을 해결할 수 없고 기타 모든 시도가 실패할 경우, 다음 명령을 실행하여 네트워크 통신량에 대한 추적을 실행하십시오.

- AIX 시스템의 Load Balancer 시스템:

```
iptrace -a -s failingClientIPAddress -d clusterIPAddress -b iptrace.trc
```

추적을 실행하고, 문제점을 다시 발생시킨 뒤 프로세스를 종료(kill)하십시오.

- HP-UX 시스템:

```
tcpdump -i lan0 host cluster and host client
```

HP-UX GNU 소프트웨어 보존 사이트 중 하나에서 tcpdump를 다운로드할 수 있습니다.

- Linux 시스템:

```
tcpdump -i eth0 host cluster and host client
```

추적을 실행하고, 문제점을 다시 발생시킨 뒤 프로세스를 종료(kill)하십시오.

- Solaris:

```
snoop -v clientIPAddress destinationIPAddress > snooptrace.out
```

- Windows에서는 sniffer가 필수입니다. 필터의 경우와 동일한 입력을 사용하십시오.

다른 로그 레벨(예: 관리자 로그, 어드바이저 로그등)을 증가하고 해당 출력을 검토할 수도 있습니다.

업그레이드

서비스 릴리스 픽스 또는 패치에서 이미 수정된 문제점을 식별하려면 업그레이드를 확인하십시오. 수정된 Edge Components 결함의 목록을 얻으려면 WebSphere Application Server 웹 사이트 지원 페이지를 참조하십시오. 주소는 <http://www.ibm.com/software/webservers/appserv/was/support/>입니다. 지원 페이지에서 수정된 서비스 다운로드 사이트 링크를 따르십시오.

Java 코드

Java 코드의 올바른 버전은 Load Balancer 설치의 일부로 설치됩니다.

유용한 링크

지원 및 라이브러리 웹 페이지 링크에 대해서는 xvii 페이지의 『참조서 정보』 페이지를 참조하십시오. 웹 지원 페이지에는 테크노트 형식의 셀프 도움말 링크가 있습니다.

문제점 해결 테이블

다음을 참조하십시오.

- Dispatcher 문제점 해결 정보 — 표 14
- CBR 문제점 해결 정보 — 313 페이지의 표 15
- Site Selector 문제점 해결 정보 — 314 페이지의 표 16
- Cisco CSS Controller 문제점 해결 정보 — 315 페이지의 표 17
- Nortel Alteon Controller 문제점 해결 정보 — 316 페이지의 표 18
- Metric Server 문제점 해결 정보 — 317 페이지의 표 19

표 14. Dispatcher 문제점 해결 테이블

증상	가능한 원인	이동 위치
Dispatcher가 제대로 실행되지 않음	포트 번호 충돌	319 페이지의 『Dispatcher 포트 번호 확인』
적절히 배치된 서버를 구성했으나 로드 밸런스 요청에 응답하지 않음	주소가 잘못되었거나 충돌함	323 페이지의 『문제점: Dispatcher 및 서버가 응답하지 않음』

표 14. Dispatcher 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
클라이언트 시스템에서의 연결이 제공되지 않거나 연결의 제한시간이 초과됨	<ul style="list-style-type: none"> • 잘못된 경로 지정 구성 • NIC의 별명으로 클러스터 주소가 지정되지 않음 • 서버에 클러스터 주소로 별명이 지정된 루프백 장치가 없음 • 불필요한 라우트가 삭제됨 • 각 클러스터에 대해 포트가 정의되지 않음 	323 페이지의 『문제점: Dispatcher 요청이 밸런스를 이루지 않음』
클라이언트 시스템이 제공되지 않거나 제한시간이 초과되었습니다.	고가용성이 작동하지 않음	324 페이지의 『문제점: Dispatcher 고가용성 기능이 작동하지 않음』
하트비트를 추가할 수 없음 (Windows 플랫폼)	어댑터에서 출발지 주소가 구성되지 않음	324 페이지의 『문제점: 하트비트를 추가할 수 없음(Windows 플랫폼)』
서버가 요청을 처리하지 않음 (Windows 플랫폼)	다른 라우트가 라우팅 테이블에 작성되어 있음	324 페이지의 『문제점: 추가 라우트(Windows 2000)』
어드바이저가 광범위한 영역에서 제대로 작동하지 않음	어드바이저가 원격 시스템에서 실행 중이 아님	324 페이지의 『문제점: 어드바이저가 제대로 작동하지 않음』
Dispatcher, Microsoft IIS 및 SSL이 작동하지 않거나 계속되지 않음	프로토콜을 통해 암호화된 데이터를 전송할 수 없음	325 페이지의 『문제점: Dispatcher, Microsoft IIS 및 SSL이 작동하지 않음(Windows 플랫폼)』
원격 시스템으로의 연결이 거부됨	이전 버전의 키가 계속 사용됨	325 페이지의 『문제점: 원격 시스템에 대한 Dispatcher 연결』
‘서버가 응답하지 않음’ 또는 ‘RMI 서버에 액세스할 수 없음’ 메시지로 dscontrol 또는 lbadm 명령이 실패함	<ol style="list-style-type: none"> 1. 소켓이 연결된 스택으로 인해 명령이 실패했거나 dscontrol이 시작되지 않아서 명령이 실패했음 2. RMI 포트가 올바르게 설정되지 않았음 3. 호스트 파일에 잘못된 로컬 호스트가 있음 	325 페이지의 『문제점: dscontrol 또는 lbadm 명령 실패』
온라인 도움말을 보기 위해 기본 브라우저로서 Netscape를 실행할 때, “파일을 찾을 수 없습니다...”라는 오류 메시지, (Windows 플랫폼)	HTML 파일 연관 설정이 잘못됨	326 페이지의 『온라인 도움말을 보려고 할 때 문제점: “파일을 찾을 수 없습니다...”라는 오류 메시지 발생(Windows 플랫폼)』
그래픽 사용자 인터페이스가 올바르게 시작하지 않음	페이징 공간이 충분하지 않음	326 페이지의 『문제점: GUI (Graphical User Interface)가 올바르게 시작되지 않음』
Caching Proxy가 설치된 Dispatcher 실행 중에 오류 발생	Caching Proxy 파일 종속성	327 페이지의 『문제점: Caching Proxy가 설치된 Dispatcher 실행 중 오류』

표 14. Dispatcher 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
그래픽 사용자 인터페이스가 올바르게 표시되지 않음	해상도가 올바르지 않음	327 페이지의 『문제점: GUI (Graphical User Interface)가 올바르게 표시되지 않음』
때때로 도움말 패널이 다른 창 뒤로 사라짐	Java 한계	327 페이지의 『문제점: Windows 플랫폼에서 때로 도움말 창이 다른 열린 창 뒤로 사라짐』
Load Balancer가 프레임을 처리하고 전달할 수 없음	NIC마다 고유한 MAC 주소가 필요함	327 페이지의 『문제점: Load Balancer가 프레임을 처리하고 전달할 수 없음』
파란색 화면이 나타남	설치 및 구성된 네트워크 카드가 없음	328 페이지의 『문제점: Load Balancer 실행 프로그램을 시작할 때 파란색 화면이 표시됨』
Discovery 경로로 인해 리턴 통신량이 발생하지 못함	루프백에서 클러스터 별명이 지정됨	328 페이지의 『문제점: Discovery 경로로 인해 Load Balancer와의 리턴 통신이 발생하지 못함』
Load Balancer의 광역 모드와 고가용성이 작동되지 않음	원격 Dispatcher가 로컬 Dispatcher의 클러스터에서 서버로 정의되어야 함	329 페이지의 『문제점: Load Balancer의 광역 모드에서 고가용성이 작동되지 않음』
GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.	Java는 GUI의 그러한 큰 변경사항을 처리하는 데 충분한 메모리에 액세스하지 않습니다.	329 페이지의 『문제점: GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.』
원격 연결을 통해 IP 주소가 제대로 분석되지 않습니다.	보안 소켓 구현을 통해 원격 클라이언트를 사용할 경우, 완전한 도메인 이름 또는 호스트 이름이 올바른 IP 주소로 분석되지 않을 수 있습니다.	330 페이지의 『문제점: 원격 연결을 통해 IP 주소가 제대로 분석되지 않음』
한글 Load Balancer 인터페이스가 AIX 및 Linux 시스템에서 겹치거나 원치 않는 글꼴을 표시합니다.	기본 글꼴이 변경되어야 함	330 페이지의 『문제점: 한글 Load Balancer 인터페이스가 AIX 및 Linux 시스템에서 겹치거나 원치 않는 글꼴을 표시함』
Windows 시스템에서 MS 루프백 어댑터의 별명을 지정한 뒤, hostname과 같은 특정 명령을 발행하면 OS가 별명 주소로 잘못 응답합니다.	네트워크 연결 목록에서, 새로 추가된 별명이 로컬 주소 위에 나열되어서는 안됩니다.	331 페이지의 『문제점: Windows 시스템에서 hostname과 같은 명령을 발행하면 로컬 주소 대신 별명 주소가 리턴됨』
Windows 플랫폼을 Matrox AGP 비디오 카드와 함께 사용 시 예상하지 못한 GUI 작동	Load Balancer GUI를 실행하는 중 Matrox AGP 비디오 카드 사용 시 문제점이 발생합니다.	331 페이지의 『문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동』
Linux 시스템에서 "rmmod ibmlb"를 실행할 때의 예상치 못한 작동 (예: 시스템 정지)	Load Balancer 커널 모듈(ibmlb)을 수동으로 제거하면 문제점이 발생함	332 페이지의 『문제점: "rmmod ibmlb"를 실행할 때 예상치 못한 작동(Linux 시스템)』
Dispatcher 시스템에서 명령을 실행할 경우 응답 시간이 느림	대량의 클라이언트 통신량으로부터의 시스템 과부하로 인해 응답 시간이 느려질 수 있음	332 페이지의 『문제점: Dispatcher 시스템에서 명령 실행 중 응답 시간이 느림』

표 14. Dispatcher 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
Dispatcher의 mac 전달 메소드에 대해 SSL 또는 HTTPS 어드바이저가 서버 로드를 등록하지 않음	SSL 서버 응용프로그램이 클러스터 IP 주소로 구성되지 않아 문제점이 발생함	332 페이지의 『문제점: SSL 또는 HTTPS 어드바이저가 서버 로드를 등록하지 않음(mac 전달을 사용할 경우)』
Netscape를 통해 원격 웹 관리를 사용할 때 호스트로부터 연결이 끊어짐	브라우저 창의 크기를 조정할 때 호스트로부터 연결이 끊어질 수 있음	333 페이지의 『문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐』
소켓 풀링이 사용 가능하며 웹 서버가 0.0.0.0으로 바인드됩니다.	Microsoft IIS 서버를 바인드 고유로 구성하십시오.	333 페이지의 『문제점: 소켓 풀링이 사용 가능하며 웹 서버가 0.0.0.0으로 바인드됨』
Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트에 표시됨	명령 프롬프트 창에서 글꼴 등록 정보 변경	334 페이지의 『문제점: Windows 시스템에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨』
HP-UX 플랫폼에서 다음 메시지가 표시됨: java.lang.OutOfMemoryError 새 기본 스레드를 작성할 수 없음	기본적으로 일부 HP-UX 설치에서 프로세스당 64 스레드를 허용합니다. 이것은 충분하지 않습니다.	334 페이지의 『문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생』
Windows 플랫폼에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함	타스크 오프로드가 사용 불가능하거나 ICMP를 사용 가능하게 해야 할 수 있습니다.	335 페이지의 『문제점: Windows 시스템에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함』
Windows 플랫폼에서 어댑터에 대해 둘 이상의 주소를 구성한 경우 호스트 이름에 대한 IP 주소를 확인하는 데 문제가 있음	호스트 이름으로 원하는 IP 주소가 레지스트리에 먼저 표시되어야 합니다.	335 페이지의 『문제점: Windows 플랫폼에서 어댑터에 둘 이상의 주소를 구성할 경우 호스트 이름에 대한 IP 주소를 확인하는 중에 문제가 발생함』
Windows 플랫폼에서 어드바이저가 네트워크 정지 후 고가용성 설정에서 작동하지 않음	시스템에서 네트워크 정지를 발견한 경우 해당하는 ARP(Address Resolution Protocol) 캐시를 지움	336 페이지의 『문제점: Windows 시스템에서 네트워크 정지 후 고가용성 설정에서 어드바이저가 작동하지 않음』
Linux 시스템에서 "IP address add" 명령 및 다중 클러스터 루프백 별명이 호환되지 않음	루프백 장치에서 둘 이상의 주소를 별명 지정할 경우 ip address add가 아니라 ifconfig 명령을 사용해야 함	337 페이지의 『문제점: 루프백 장치에서 여러 클러스터 별명을 지정할 경우 Linux 시스템에서 "IP address add" 명령을 사용하지 않음』
오류 메시지: 서버 추가 시 "포트 메소드에 대해 지정되지 않은 또는 올바르지 않은 라우터 주소"	서버 추가 시 발생한 문제점을 판별하기 위한 정보 점검 목록	337 페이지의 『문제점: "포트 메소드에 대해 지정되지 않은 또는 올바르지 않은 라우터 주소" 오류 메시지』
Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 세션 창을 종료할 때 프로세스가 종료됨	nohup 명령을 사용하여 터미널 세션 종료 시 사용자가 시작한 프로세스가 끊기 신호를 수신하는 것을 방지하십시오.	338 페이지의 『문제점: Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 창을 종료할 때 프로세스가 종료됨』

표 14. Dispatcher 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
Load Balancer 구성 로드 시 속도가 느려지는 현상 발생	지연은 서버 주소를 해석 및 확인하기 위해 작성된 DNS(Domain Name System) 호출이 원인일 수 있습니다.	338 페이지의 『문제점: Load Balancer 구성 로드 시 지연』
Windows 시스템에서 "네트워크의 다른 시스템과 I/P 주소 충돌이 있습니다"라는 오류 메시지가 나타납니다.	고가용성이 구성되는 경우, 해당 오류 메시지를 발생시킨 클러스터 주소가 두 시스템 모두에 잠시 동안 구성될 수 있습니다.	339 페이지의 『문제점: Windows 시스템에서 IP 주소 충돌 오류 메시지가 나타남』
고가용성 구성에서 기본 시스템 및 백업 시스템 모두가 활성화됨	이 문제점은 go 스크립트가 기본 시스템 또는 백업 시스템에서 실행되지 않는 경우 발생할 수 있습니다.	339 페이지의 『문제점: 고가용성 구성에서 기본 시스템 및 백업 시스템 모두가 활성화됨』
Dispatcher가 대형 페이지 응답을 리턴하면 클라이언트 요청이 실패함	nat 또는 cbr 전달 사용 시 최대 전송 단위(MTU)가 Dispatcher 시스템에 바르게 설정되지 않은 경우, 대형 페이지 응답을 초래하는 클라이언트 요청은 제한시간 초과가 됩니다.	339 페이지의 『문제점: 대형 페이지 응답 리턴 시 클라이언트 요청 실패』
Windows 시스템에서 dscontrol 또는 lbadm 명령을 발행하면 "서버가 응답하지 않음" 오류가 발생	두 개 이상의 IP 주소가 Windows 시스템에 존재하고 호스트 파일이 호스트 이름과 연결할 주소를 지정하지 않은 경우입니다.	340 페이지의 『문제점: Windows 시스템에서 dscontrol 또는 lbadm 명령을 발행하면 "서버가 응답하지 않음" 오류가 발생』
고가용성 Dispatcher 시스템의 qeth 장치에 있는 S/390용 Linux에 동기화가 불가능할 수도 있음	qeth 네트워크 드라이버가 있는 S/390용 Linux에 고가용성을 사용하는 경우, 활성화 및 대기 Dispatcher가 동기화에 실패할 수 있습니다.	340 페이지의 『문제점: 고가용성 Dispatcher 시스템의 qeth 드라이버에 있는 S/390 시스템용 Linux에 동기화가 불가능할 수도 있음』
Load Balancer의 고가용성 기능 구성에 대한 팁	다음과 같은 고가용성 문제점 완화를 돕는 팁입니다. • 인계 후에 연결이 끊김 • 상대 시스템과 동기화가 불가능함 • 요청이 백업 상대 시스템으로 잘못 방향 지정됨	340 페이지의 『문제점: 고가용성 구성에 대한 팁』
zSeries 및 S/390 플랫폼의 Dispatcher mac 전달 구성 제한사항	Linux에서는 개방형 시스템 어댑터(OSA) 카드가 있는 zSeries 또는 S/390 서버 사용 시 제한사항이 있습니다. 가능한 해결책이 제공됩니다.	342 페이지의 『문제점: Linux에서는 개방형 시스템 어댑터(OSA) 카드가 있는 zSeries 또는 S/390 서버 사용 시 Dispatcher 구성 제한사항이 있음』
일부 Red Hat Linux 버전에서는 관리자 및 어드바이저로 구성된 Load Balancer 실행 시 메모리 누수가 발생함	Red Hat Enterprise Linux 3.0과 같은 일부 Linux 분배가 실린 NPTL(Native POSIX Thread Library) 및 JVM의 IBM Java SDK 버전은 메모리 누수를 야기할 수 있습니다.	344 페이지의 『문제점: 일부 Linux 버전에서는 관리자 및 어드바이저로 구성된 Dispatcher 실행 시 메모리 누수가 발생함』

표 14. Dispatcher 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
SUSE Linux Enterprise Server 9에서는 Dispatcher 보고서에 패킷이 전달되었음이 표시되지만(패킷 계수 증가), 실제로 패킷이 백엔드 서버에 도달하지는 않음	iptables NAT 모듈이 로드됩니다. iptables의 해당 버전에 Dispatcher와 상호작용 시 특이한 동작을 일으키는 발생 가능하지만 확인되지 않은 오류가 있습니다.	345 페이지의 『문제점: SUSE Linux Enterprise Server 9에서는 Dispatcher가 패킷을 전달하지만 패킷은 백엔드 서버에 도달하지 않음』
Windows 시스템에서 Dispatcher의 고가용성 기능을 사용하는 경우, 인계 중에 문제점이 발생할 수 있음	활성 시스템에 클러스터 IP 주소를 구성하는 goScript가 goScript 전에 실행되어 백업 시스템에 있는 IP 클러스터 주소를 해제하면 문제점이 발생할 수 있습니다.	346 페이지의 『문제점: Windows 시스템에서는 고가용성 인계 중에 IP 주소 충돌 메시지가 나타남』
Linux 시스템에서 iptables가 패킷 라우팅에 간섭할 수 있음	Linux iptables는 통신량의 로드 밸런스를 간섭할 수 있으므로 Load Balancer 시스템에서는 사용 불가능하게 해야 합니다.	346 페이지의 『문제점: Linux iptables가 패킷 라우팅에 간섭할 수 있음』
Solaris 시스템의 경우 Dispatcher 시스템에 IPv6 서버를 구성하려고 하면 "서버를 추가할 수 없음"이라는 메시지가 나타남	해당 오류는 Solaris 운영 체제가 IPv6 주소에 필요한 핑 요청을 핸들하는 방법 때문에 발생할 수 있습니다.	347 페이지의 『문제점: IPv6 서버를 Solaris 시스템의 Load Balancer 구성에 추가할 수 없음』
시스템 패키징 도구를 사용하여 고유하게 설치하거나 서비스 수정사항 설치 시 Java 파일 세트 경고 메시지가 나타남	제품 설치의 동일한 시스템에 설치할 필요가 없는 여러 개의 패키지로 이루어지며, 패키지 각각은 Java 파일 세트를 설치합니다. 동일한 시스템에 설치되는 경우, 다른 파일 세트도 Java 파일 세트를 소유하고 있다는 경고 메시지가 나타납니다.	347 페이지의 『서비스 수정사항 설치 시 Java 경고 메시지가 나타남』
Load Balancer 설치가 공급된 Java 파일 세트 업그레이드	Java 파일 세트에서 문제점이 발견되면 IBM 서비스에 보고하여 Load Balancer 설치가 공급된 Java 파일 세트 업그레이드를 수신할 수 있습니다.	347 페이지의 『Load Balancer 설치가 공급된 Java 파일 세트 업그레이드』

표 15. CBR 문제점 해결 테이블

증상	가능한 원인	이동 위치
CBR이 제대로 실행되지 않습니다.	포트 번호 충돌	319 페이지의 『CBR 포트 번호 확인』
‘서버가 응답하지 않음’ 또는 ‘RMI 서버에 액세스할 수 없음’ 메시지로 cbrcontrol 또는 lbadm 명령이 실패함	소켓이 연결된 스택으로 인해 명령이 실패했거나 cbrserver가 시작되지 않아서 명령이 실패했음	348 페이지의 『문제점: cbrcontrol 또는 lbadm 명령 실패』
요청이 로드 밸런스되지 않음	실행 프로그램이 시작되기 전에 Caching Proxy가 시작됨	348 페이지의 『문제점: 요청이 로드 밸런스되지 않음』
Solaris에서 cbrcontrol executor start 명령이 ‘오류: 실행 프로그램이 시작되지 않았습니다.’라는 메시지를 message	시스템 IPC 기본값이 수정되어야 하거나 라이브러리에 링크가 잘못 되었으므로 명령이 실패함	349 페이지의 『문제점: Solaris 시스템에서 cbrcontrol executor start 명령 실패』

표 15. CBR 문제점 해결 테이블 (계속)

URL 규칙이 작동하지 않음	구문 또는 구성 오류	349 페이지의 『문제점: 구문 또는 구성 오류』
Windows 시스템을 Matrox AGP 비디오 카드와 함께 사용 시 예상하지 못한 GUI 작동	Load Balancer GUI를 실행하는 중 Matrox AGP 비디오 카드 사용 시 문제점이 발생합니다.	349 페이지의 『문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동』
GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.	Java는 GUI의 그러한 큰 변경사항을 처리하는 데 충분한 메모리에 액세스하지 않습니다.	329 페이지의 『문제점: GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.』
Netscape를 통해 원격 웹 관리를 사용할 때 호스트로부터 연결이 끊어짐	브라우저 창의 크기를 조정할 때 호스트로부터 연결이 끊어질 수 있음	349 페이지의 『문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐』
Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트에 표시됨	명령 프롬프트 창에서 글꼴 등록 정보 변경	350 페이지의 『문제점: Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨』
HP-UX 플랫폼에서 다음 메시지가 표시됨: java.lang.OutOfMemoryError 새 기본 스레드를 작성할 수 없음	기본적으로 일부 HP-UX 설치에서 프로세스당 64 스레드를 허용합니다. 이것은 충분하지 않습니다.	350 페이지의 『문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생』
Windows 플랫폼에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함	타스크 오프로드가 사용 불가능하거나 icmp를 사용 가능하게 해야 할 수 있습니다.	350 페이지의 『문제점: Windows 시스템에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함』
Windows 플랫폼에서 어댑터에 대해 둘 이상의 주소를 구성할 경우 호스트 이름에 대한 IP 주소를 확인하는 중 문제점이 발생함	호스트 이름으로 원하는 IP 주소가 레지스트리에 먼저 표시되어야 합니다.	350 페이지의 『문제점: Windows 시스템에서 어댑터에 둘 이상의 주소를 구성할 경우 호스트 이름에 대한 IP 주소를 확인하는 중에 문제가 발생함』
Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 세션 창을 종료할 때 프로세스가 종료됨	nohup 명령을 사용하여 터미널 세션 종료 시 사용자가 시작한 프로세스가 끊기 신호를 수신하는 것을 방지하십시오.	338 페이지의 『문제점: Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 창을 종료할 때 프로세스가 종료됨』

표 16. Site Selector 문제점 해결 테이블

증상	가능한 원인	이동 위치
Site Selector가 올바르게 실행되지 않음	포트 번호 충돌	320 페이지의 『Site Selector 포트 번호 확인』
Site Selector가 Solaris 클라이언트로부터 수신하는 요청을 라운드 로빙하지 않음	Solaris 시스템이 "이름 서비스 캐시 디먼"을 실행함	351 페이지의 『문제점: Site Selector가 Solaris 클라이언트로부터 통신을 라운드 로빙하지 않음』
‘서버가 응답하지 않음’ 또는 ‘RMI 서버에 액세스할 수 없음’ 메시지로 sscontrol 또는 lbadmin 명령이 실패함	소켓이 연결된 스택으로 인해 명령이 실패했거나 ssserver가 시작되지 않아서 명령이 실패했음	351 페이지의 『문제점: sscontrol 또는 lbadmin 명령 실패』

표 16. Site Selector 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
ssserver가 Windows 플랫폼에서 시작되지 않음	Windows 시스템에서는 호스트 이름이 DNS에 있을 필요는 없습니다.	352 페이지의 『문제점: sserver가 Windows 플랫폼에서 시작에 실패함』
시스템이 중복 라우트를 통해 올바르게 로드 밸런스되지 않음 — 이름 분석에 실패	여러 어댑터가 있는 Site Selector가 동일한 서브넷에 연결됨	352 페이지의 『문제점: Site Selector가 중복 라우트를 통해 올바르게 로드 밸런스하지 않음』
Windows 플랫폼을 Matrox AGP 비디오 카드와 함께 사용 시 예상하지 못한 GUI 작동	Load Balancer GUI를 실행하는 중 Matrox AGP 비디오 카드 사용 시 문제점이 발생합니다.	352 페이지의 『문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동』
GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.	Java는 GUI의 그러한 큰 변경사항을 처리하는 데 충분한 메모리에 액세스하지 않습니다.	329 페이지의 『문제점: GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.』
Netscape를 통해 원격 웹 관리를 사용할 때 호스트로부터 연결이 끊어짐	브라우저 창의 크기를 조정할 때 호스트로부터 연결이 끊어질 수 있음	353 페이지의 『문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐』
Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트에 표시됨	명령 프롬프트 창에서 글꼴 등록 정보 변경	353 페이지의 『문제점: Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨』
HP-UX 플랫폼에서 다음 메시지가 표시됨: java.lang.OutOfMemoryError 새 기본 스레드를 작성할 수 없음	기본적으로 일부 HP-UX 설치에서 프로세스당 64 스레드를 허용합니다. 이것은 충분하지 않습니다.	353 페이지의 『문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생』
Windows 플랫폼에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함	타스크 오프로드가 사용 불가능하거나 icmp를 사용 가능하게 해야 할 수 있습니다.	353 페이지의 『문제점: Windows 시스템에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함』
Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 세션 창을 종료할 때 프로세스가 종료됨	nohup 명령을 사용하여 터미널 세션 종료 시 사용자가 시작한 프로세스가 끊기 신호를 수신하는 것을 방지하십시오.	338 페이지의 『문제점: Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 창을 종료할 때 프로세스가 종료됨』

표 17. Controller for Cisco CSS Switches 문제점 해결 테이블

증상	가능한 원인	이동 위치
ccoserver가 시작되지 않음	포트 번호 충돌	321 페이지의 『Cisco CSS Controller 포트 번호 확인』
‘서버가 응답하지 않음’ 또는 ‘RMI 서버에 액세스할 수 없음’ 메시지로 ccocontrol 또는 lbadm 명령이 실패함	소켓이 연결된 스택으로 인해 명령이 실패했거나 ccoserver가 시작되지 않아서 명령이 실패했음	354 페이지의 『문제점: ccocontrol 또는 lbadm 명령 실패』
수신 오류: 포트 13099에서 레지스트리를 작성할 수 없음	제품 사용권이 만기됨	355 페이지의 『문제점: 포트 13099에서 레지스트리를 작성할 수 없음』

표 17. Controller for Cisco CSS Switches 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
Windows 플랫폼을 Matrox AGP 비디오 카드와 함께 사용 시 예상하지 못한 GUI 작동	Load Balancer GUI를 실행하는 중 Matrox AGP 비디오 카드 사용 시 문제점이 발생합니다.	355 페이지의 『문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동』
컨설턴트를 추가할 때 연결 오류를 받음	스위치 또는 제어기의 구성 설정이 올바르지 않음	355 페이지의 『문제점: 컨설턴트를 추가할 때 연결 오류를 받음』
스위치의 가중치가 갱신되지 않음	제어기 또는 스위치 간의 통신이 사용 불가능하거나 인터럽트됨	355 페이지의 『문제점: 스위치의 가중치가 갱신되지 않음』
Refresh 명령이 컨설턴트 구성을 갱신하지 않음	제어기 또는 스위치 간의 통신이 사용 불가능하거나 인터럽트됨	356 페이지의 『문제점: Refresh 명령이 컨설턴트 구성을 갱신하지 않음』
GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.	Java는 GUI의 그러한 큰 변경사항을 처리하는 데 충분한 메모리에 액세스하지 않습니다.	329 페이지의 『문제점: GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.』
Netscape를 통해 원격 웹 관리를 사용할 때 호스트로부터 연결이 끊어짐	브라우저 창의 크기를 조정할 때 호스트로부터 연결이 끊어질 수 있음	356 페이지의 『문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐』
Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트에 표시됨	명령 프롬프트 창에서 글꼴 등록 정보 변경	356 페이지의 『문제점: Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨』
HP-UX 플랫폼에서 다음 메시지가 표시됨: java.lang.OutOfMemoryError 새 기본 스레드를 작성할 수 없음	기본적으로 일부 HP-UX 설치에서 프로세스당 64 스레드를 허용합니다. 이것은 충분하지 않습니다.	356 페이지의 『문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생』
Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 세션 창을 종료할 때 프로세스가 종료됨	nohup 명령을 사용하여 터미널 세션 종료 시 사용자가 시작한 프로세스가 끊기 신호를 수신하는 것을 방지하십시오.	338 페이지의 『문제점: Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 창을 종료할 때 프로세스가 종료됨』

표 18. Nortel Alteon Controller 문제점 해결 테이블

증상	가능한 원인	이동 위치
nalserver가 시작되지 않음	포트 번호 충돌	322 페이지의 『Nortel Alteon Controller 포트 번호 확인』
‘서버가 응답하지 않음’ 또는 ‘RMI 서버에 액세스할 수 없음’ 메시지로 nalcontrol 또는 lbadmin 명령이 실패함	소켓이 연결된 스택으로 인해 명령이 실패했거나 nalserver가 시작되지 않아서 명령이 실패했음	357 페이지의 『문제점: nalcontrol 또는 lbadmin 명령 실패』
수신 오류: 포트 14099에서 레지스트리를 작성할 수 없음	제품 사용권이 만기됨	357 페이지의 『문제점: 포트 14099에서 레지스트리를 작성할 수 없음』

표 18. Nortel Alteon Controller 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
Windows 플랫폼을 Matrox AGP 비디오 카드와 함께 사용 시 예상하지 못한 GUI 작동	Load Balancer GUI를 실행하는 중 Matrox AGP 비디오 카드 사용 시 문제점이 발생합니다.	358 페이지의 『문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동』
GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.	Java는 GUI의 그러한 큰 변경사항을 처리하는 데 충분한 메모리에 액세스하지 않습니다.	329 페이지의 『문제점: GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.』
Netscape를 통해 원격 웹 관리를 사용할 때 호스트로부터 연결이 끊어짐	브라우저 창의 크기를 조정할 때 호스트로부터 연결이 끊어질 수 있음	358 페이지의 『문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐』
컨설팅트를 추가할 때 연결 오류를 받음	스위치 또는 제어기의 구성 설정이 올바르지 않음	358 페이지의 『문제점: 컨설팅트를 추가할 때 연결 오류를 받음』
스위치의 가중치가 갱신되지 않음	제어기 또는 스위치 간의 통신이 사용 불가능하거나 인터럽트됨	358 페이지의 『문제점: 스위치의 가중치가 갱신되지 않음』
Refresh 명령이 컨설팅트 구성을 갱신하지 않음	제어기 또는 스위치 간의 통신이 사용 불가능하거나 인터럽트됨	359 페이지의 『문제점: Refresh 명령이 컨설팅트 구성을 갱신하지 않음』
Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트에 표시됨	명령 프롬프트 창에서 글꼴 등록 정보 변경	359 페이지의 『문제점: Windows 시스템에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨』
HP-UX 플랫폼에서 다음 메시지가 표시됨: java.lang.OutOfMemoryError 새 기본 스택을 작성할 수 없음	기본적으로 일부 HP-UX 설치에서 프로세스당 64 스택을 허용합니다. 이것은 충분하지 않습니다.	359 페이지의 『문제점: HP-UX에서 Java 메모리 부족/스택 오류 발생』
Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 세션 창을 종료할 때 프로세스가 종료됨	nohup 명령을 사용하여 터미널 세션 종료 시 사용자가 시작한 프로세스가 끊기 신호를 수신하는 것을 방지하십시오.	338 페이지의 『문제점: Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 창을 종료할 때 프로세스가 종료됨』

표 19. Metric Server 문제점 해결 테이블

증상	가능한 원인	이동 위치
.bat 또는 .cmd 사용자 메트릭 파일을 실행하는 Windows 플랫폼의 Metric Server IOException	전체 메트릭 이름이 필요함	359 페이지의 『문제점: .bat 또는 .cmd 사용자 메트릭 파일을 실행하는 Windows 플랫폼의 Metric Server IOException』

표 19. Metric Server 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
Metric Server는 Load Balancer 시스템에 로드 정보를 보고하지 않습니다.	가능한 원인은 다음과 같습니다. <ul style="list-style-type: none"> • Metric Server 시스템에 키 파일이 없음 • Metric Server 시스템의 호스트 이름이 로컬 이름 서버에 등록되지 않음 • /etc/hosts 파일에 루프백 주소 127.0.0.1에 대응하는 로컬 호스트 이름이 있음 	360 페이지의 『문제점: Metric Server가 Load Balancer 시스템에 로드를 보고하지 않음』
Metric Server 로그는 키 파일이 서버로 전송될 때 "에이전트에 액세스하려면 서명이 필요합니다"라고 보고합니다.	키 파일이 파손되어 권한 부여에 실패했습니다.	360 페이지의 『문제점: Metric Server 로그에서 "에이전트에 액세스하려면 서명이 필요합니다."라고 보고합니다.』
AIX 시스템에서 멀티프로세서 시스템(AIX 4.3.3 또는 AIX 5.1)이 과부하된 상태로 Metric Server를 실행할 경우 ps -vg 명령 출력이 손상될 수도 있습니다.	APAR IY33804는 이러한 알려진 AIX 문제점을 수정합니다.	360 페이지의 『문제점: AIX 시스템에서 과부하된 상태로 Metric Server를 실행할 경우, ps -vg 명령 출력이 손상될 수도 있음』
고가용성 Dispatcher에서 Site Selector 로드 밸런싱을 사용하여 2계층 구성으로 Metric Server를 구성함	Metric Server(두 번째 계층에 상주)가 새 IP 주소를 인식하도록 구성되지 않았습니다.	361 페이지의 『문제점: 고가용성 Dispatcher에서 Site Selector 로드 밸런싱을 사용하여 2계층 구성으로 Metric Server를 구성함』
여러 개의 CPU가 있는 Solaris 시스템에서 실행되는 스크립트 (metricserver, cpuload, memload)는 원치 않는 콘솔 메시지를 생산함	이 동작은 VMSTAT 시스템 명령을 사용하여 커널에서 CPU 및 메모리 통계를 집계하기 때문입니다.	362 페이지의 『문제점: 여러 개의 CPU가 있는 Solaris 시스템에서 실행 중인 스크립트가 원치 않는 콘솔 메시지를 생산』
Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 세션 창을 종료할 때 프로세스가 종료됨	nohup 명령을 사용하여 터미널 세션 종료 시 사용자가 시작한 프로세스가 끊기 신호를 수신하는 것을 방지하십시오.	338 페이지의 『문제점: Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 창을 종료할 때 프로세스가 종료됨』
Linux 시스템에서는 IPv6용 Load Balancer 실행 시 Metric Server에서 값을 검색할 수 없음	Linux 플랫폼에서 실행 시 소스 IPv6 주소 선택사항의 호환이 불가능합니다. 따라서 Metric Monitor는 잘못된 소스 IP 주소를 통해 Metric Server와 통신하려 합니다.	362 페이지의 『문제점: IPv6용 Load Balancer에서 Linux 시스템의 Metric Server로부터 값을 검색할 수 없음』
Metric Server 시작 후 메트릭 값 -1 리턴	이 문제점은 클라이언트 전송 중에 손실된 키 파일의 무결성에서 기인한 것일 수 있습니다.	363 페이지의 『문제점: Metric Server 시작 후 메트릭 값이 -1을 리턴함』

Dispatcher 포트 번호 확인

Dispatcher를 실행하는데 문제점이 발생한 경험이 있으면, 응용프로그램 중 하나가 Dispatcher가 일반적으로 사용하는 포트 번호를 사용 중일 수 있습니다. Dispatcher 서버는 다음과 같은 포트 번호를 사용합니다.

- 10099: dscontrol에서 명령을 수신할 경우
- 10004: Metric Server에 메트릭 조회를 전송할 경우
- 10199: RMI 서버 포트의 경우

다른 응용프로그램이 Dispatcher의 포트 번호 중 하나를 사용 중일 경우, Dispatcher의 포트 번호를 변경하거나 또는 응용프로그램의 포트 번호를 변경할 수 있습니다.

다음은 수행하여 Dispatcher의 포트 번호를 변경하십시오.

- 명령을 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - dsserver 파일 맨 위에 있는 LB_RMIPORT 변수를 Dispatcher가 명령을 받을 포트로 수정하십시오.
- Metric Server로부터 메트릭 보고서를 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - metricserver 파일의 RMI_PORT 변수를 Dispatcher가 Metric Server와 통신할 포트로 수정하십시오.
 - 관리자가 시작될 때 metric_port 인수를 제공하십시오. **dscontrol manager start** 명령 구문에 대한 설명은 397 페이지의 『dscontrol manager — 관리자 제어』를 참조하십시오.

다음은 수행하여 응용프로그램의 RMI 포트 번호를 변경하십시오.

- 응용프로그램에서 사용하는 포트를 변경하려면
 - dsserver 파일의 LB_RMISERVERPORT 변수를 응용프로그램이 사용할 포트로 수정하십시오. (응용프로그램이 사용하는 RMI 포트의 기본값은 10199입니다.)

주: Windows 플랫폼의 경우 dsserver와 metricserver 파일은 C:\winnt\system32 디렉토리에 있습니다. 다른 플랫폼의 경우, 파일이 /usr/bin/ 디렉토리에 있습니다.

CBR 포트 번호 확인

CBR 실행 중 문제점을 경험한 경우, 응용프로그램 중 하나가 CBR이 일반적으로 사용하는 포트 번호를 사용 중일 수 있습니다. CBR은 다음과 같은 포트 번호를 사용합니다.

- 11099: cbrcontrol에서 명령을 수신할 경우
- 10004: Metric Server에 메트릭 조회를 전송할 경우
- 11199: RMI 서버 포트의 경우

주: CBR(Content Based Routing) 컴포넌트는 64비트 JVM 실행 플랫폼을 제외한 모든 지원 플랫폼에서 사용 가능합니다. 혹은 Load Balancer에 있는 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 Caching Proxy를 사용하지 않고도 CBR을 제공할 수 있습니다. 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 페이지를 참조하십시오.

다른 응용프로그램이 CBR의 포트 번호 중 하나를 사용 중일 경우, CBR의 포트 번호를 변경하거나 또는 응용프로그램의 포트 번호를 변경할 수 있습니다.

다음은 수행하여 CBR'의 포트 번호를 변경하십시오.

- 명령을 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - cbrserver 파일 맨 위의 LB_RMIPORT 변수를 CBR이 명령을 받을 포트로 수정하십시오.
- Metric Server로부터 메트릭 보고서를 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - metricsserver 파일의 RMI_PORT 변수를 CBR이 Metric Server와 통신할 포트 로 수정하십시오.
 - 관리자가 시작될 때 metric_port 인수를 제공하십시오. **manager start** 명령 구문에 대한 설명은 397 페이지의 『dscontrol manager — 관리자 제어』를 참조하십시오.

다음은 수행하여 응용프로그램의 RMI 포트 번호를 변경하십시오.

- 응용프로그램에서 사용하는 포트를 변경하려면
 - cbrserver 파일 맨 위의 LB_RMISERVERPORT 변수를 응용프로그램이 사용할 포트 로 수정하십시오. (응용프로그램이 사용하는 RMI 포트의 기본값은 11199입니다.)

주: Windows 플랫폼의 경우 cbrserver 및 metricsserver 파일은 C:\winnt\system32 디렉토리에 있습니다. 다른 플랫폼의 경우, 파일이 /usr/bin/ 디렉토리에 있습니다.

Site Selector 포트 번호 확인

Site Selector 실행에 관한 문제점이 발생하는 경우, 응용프로그램의 하나가 Site Selector에서 사용하는 포트 번호를 사용하고 있을 수 있습니다. Site Selector는 반드시 다음 포트 번호를 사용해야 합니다.

- 12099: sscontrol에서 명령을 받을 경우
- 10004: Metric Server에 메트릭 조회를 전송할 경우
- 12199: RMI 서버 포트의 경우

다른 응용프로그램이 Site Selector의 포트 번호 중 하나를 사용 중일 경우, Site Selector의 포트 번호를 변경하거나 또는 응용프로그램의 포트 번호를 변경할 수 있습니다.

다음을 수행하여 Site Selector의 포트 번호를 변경하십시오.

- 명령을 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - ssserver 파일 맨 위의 LB_RMI_PORT 변수를 Site Selector가 명령을 받을 포트로 수정하십시오.
- Metric Server로부터 메트릭 보고서를 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - metricserver 파일의 RMI_PORT 변수를 Site Selector가 Metric Server와 통신할 포트로 수정하십시오.
 - 관리자가 시작될 때 metric_port 인수를 제공하십시오. **manager start** 명령 구문에 대한 설명은 439 페이지의 『sscontrol manager — 관리자 제어』를 참조하십시오.

다음을 수행하여 응용프로그램의 RMI 포트 번호를 변경하십시오.

- 응용프로그램에서 사용하는 포트를 변경하려면
 - ssserver 파일 맨 위의 LB_RMISERVERPORT 변수를 응용프로그램이 사용할 포트로 수정하십시오. (응용프로그램이 사용하는 RMI 포트의 기본값은 12199입니다.)

주: Windows 플랫폼의 경우, ssserver 및 metricserver 파일은 C:\winnt\system32 디렉토리에 있습니다. 다른 플랫폼의 경우, 파일이 /usr/bin/ 디렉토리에 있습니다.

Cisco CSS Controller 포트 번호 확인

Cisco CSS Controller 컴포넌트 실행에 대한 문제점이 발생하는 경우, 다른 응용프로그램이 Cisco CSS Controller의 ccserver에서 사용하는 포트 번호 중 하나를 사용하고 있을 수 있습니다. Cisco CSS Controller는 반드시 다음 포트 번호를 사용해야 합니다.

13099: ccocontrol로부터 명령을 받을 경우

10004: Metric Server에 메트릭 조회를 전송할 경우

13199: RMI 서버 포트의 경우

다른 응용프로그램이 Cisco CSS Controller의 포트 번호 중 하나를 사용 중일 경우, Cisco CSS Controller의 포트 번호를 변경하거나 또는 응용프로그램의 포트 번호를 변경할 수 있습니다.

다음을 수행하여 Cisco CSS Controller의 포트 번호를 변경하십시오.

- ccocontrol로부터 명령을 받을 포트를 변경하려면, ccoserver 파일의 CCO_RMIPORT 변수를 수정하십시오. 13099를 Cisco CSS Controller가 ccocontrol 명령을 받을 포트로 변경하십시오.
- Metric Server로부터 메트릭 보고서를 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 1. metricserver 파일의 RMI_PORT 변수를 수정하십시오. 10004를 Cisco CSS Controller가 Metric Server와 통신할 포트로 변경하십시오.
 2. 컨설턴트가 시작될 때 metric_port 인수를 제공하십시오.

다음을 수행하여 응용프로그램의 RMI 포트 번호를 변경하십시오.

- 응용프로그램에서 사용하는 포트를 변경하려면
 - ccoserver 파일 맨 위의 CCO_RMISERVERPORT 변수를 응용프로그램이 사용할 포트로 수정하십시오. (응용프로그램이 사용하는 RMI 포트의 기본값은 13199입니다.)

주: Windows 플랫폼의 경우 ccoserver 및 metricserver 파일은 C:\winnt\system32 디렉토리에 있습니다. 다른 플랫폼의 경우, 파일이 /usr/bin/ 디렉토리에 있습니다.

Nortel Alteon Controller 포트 번호 확인

Nortel Alteon Controller 컴포넌트 실행에 대한 문제점이 발생하는 경우, 다른 응용프로그램이 Nortel Alteon Controller의 nalserver에서 사용하는 포트 번호 중 하나를 사용하고 있을 수 있습니다. Nortel Alteon Controller는 반드시 다음 포트 번호를 사용해야 합니다.

14099: nalcontrol로부터 명령을 받을 경우

10004: Metric Server에 메트릭 조회를 전송할 경우

14199: RMI 서버 포트의 경우

다른 응용프로그램이 Nortel Alteon Controller의 포트 번호 중 하나를 사용 중일 경우, Nortel Alteon Controller의 포트 번호를 변경하거나 또는 응용프로그램의 포트 번호를 변경할 수 있습니다.

다음을 수행하여 Nortel Alteon Controller의 포트 번호를 변경하십시오.

- nalcontrol로부터 명령을 받을 포트를 변경하려면, nalserver 파일의 NAL_RMIPORT 변수를 수정하십시오. 14099를 Nortel Alteon Controller가 nalcontrol 명령을 받을 포트로 변경하십시오.
- Metric Server로부터 메트릭 보고서를 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 1. metricserver 파일의 RMI_PORT 변수를 수정하십시오. 10004를 Nortel Alteon Controller가 Metric Server와 통신할 포트로 변경하십시오.

2. 컨설턴트가 시작될 때 `metric_port` 인수를 제공하십시오.

다음은 수행하여 응용프로그램의 RMI 포트 번호를 변경하십시오.

- 응용프로그램에서 사용하는 포트를 변경하려면
 - `nalserver` 파일 맨 위의 `LB_RMISERVERPORT` 변수를 응용프로그램이 사용할 포트로 수정하십시오. (응용프로그램이 사용하는 RMI 포트의 기본값은 14199입니다.)

주: Windows 플랫폼의 경우, `nalserver` 및 `metricserver` 파일은 `C:\winnt\system32` 디렉토리에 있습니다. 다른 플랫폼의 경우, 파일이 `/usr/bin/` 디렉토리에 있습니다.

공통 문제점 해결—Dispatcher

문제점: Dispatcher가 실행되지 않음

이 문제점은 다른 응용프로그램에서 Dispatcher가 사용하는 포트 중 하나를 사용하는 경우에 발생할 수 있습니다. 자세한 내용은 319 페이지의 『Dispatcher 포트 번호 확인』을 참조하십시오.

문제점: Dispatcher 및 서버가 응답하지 않음

이 문제점은 지정된 주소가 아닌 다른 주소가 사용되고 있을 때 발생합니다. Dispatcher 및 서버를 결합 배치할 때 구성에 사용되는 서버 주소가 NFA 주소이거나 결합 배치 형태로 구성되어 있는지 확인하십시오. 그리고 올바른 주소에 대한 `호스트` 파일도 점검하십시오.

문제점: Dispatcher 요청이 밸런스를 이루지 않음

이 문제점에는 클라이언트 시스템에서의 연결이 제공되지 않거나 연결의 제한시간이 초과되는 등의 증상이 있습니다. 다음을 확인하여 문제점을 진단하십시오.

1. 경로지정에 대해 비전달 주소, 클러스터, 포트 및 서버를 구성했습니까? 구성 파일을 확인하십시오.
2. 네트워크 인터페이스 카드에 클러스터 주소로 별명이 지정되어 있습니까? Linux 및 UNIX 시스템의 경우, `netstat -ni`를 사용하여 확인하십시오.
3. 각 서버에 있는 루프백 장치에 클러스터 주소로 설정된 별명이 있습니까? Linux 및 UNIX 시스템의 경우, `netstat -ni`를 사용하여 확인하십시오.
4. 여분의 라우트가 삭제됩니까? Linux 및 UNIX 시스템의 경우, `netstat -nr`를 사용하여 확인하십시오.
5. `dscontrol cluster status` 명령을 사용하여 사용자가 정의한 각 클러스터에 대한 정보를 확인하십시오. 각 클러스터마다 포트가 정의되어 있어야 합니다.
6. `dscontrol server report:` 명령을 사용하여 서버가 단절되어 있지도 않고 가중치가 0으로 설정되어 있지도 않은지 확인하십시오.

Windows 및 기타 플랫폼의 경우 79 페이지의 『서버 시스템의 시스템 설정』을 참조하십시오.

문제점: Dispatcher 고가용성 기능이 작동하지 않음

이 문제점은 Dispatcher 고가용성 환경이 구성되어 있으나, 클라이언트 시스템으로부터의 연결이 제공되지 않거나 제한시간이 초과되고 있는 경우에 나타납니다. 문제점을 정정하거나 진단하려면, 다음을 확인하십시오.

- goActive, goStandby 및 goInOp 스크립트를 작성하여 이들을 Dispatcher가 설치된 bin 디렉토리에 저장했는지 확인하십시오. 스크립트에 대한 자세한 정보는 223 페이지의 『스크립트 사용』을 참조하십시오.
- AIX, HP-UX, Linux 및 Solaris 시스템의 경우, goActive, goStandby 및 goInOp 스크립트에 실행 권한 세트가 있는지 확인하십시오.
- Windows 시스템의 경우, **executor configure** 명령을 사용하여 비전달 주소를 구성하십시오.

다음 단계는 고가용성 스크립트가 적절히 기능하는지 테스트하는 효과적인 방법입니다.

1. netstat -an 및 ifconfig -a를 발행하여 시스템에서 보고서를 집적
 2. goActive 스크립트 실행
 3. goStandby 스크립트 실행
 4. netstat -an 및 ifconfig -a를 다시 발행하여 보고서 집적
- 스크립트가 바르게 구성되었으면 두 보고서가 일치합니다.

문제점: 하트비트를 추가할 수 없음(Windows 플랫폼)

어댑터에 소스 주소가 구성되지 않은 경우 이 Windows 플랫폼 오류가 발생합니다. 문제점을 정정하거나 진단하려면, 다음을 확인하십시오.

- 토큰링 또는 이더넷 인터페이스를 사용하고 다음 명령 중 하나를 실행하여 비전달 주소를 구성해야 합니다.

```
dscontrol executor configure <ip address>
```

문제점: 추가 라우트(Windows 2000)

서버 시스템을 설정한 후, 부주의로 하나 이상의 라우트를 작성했다는 것을 발견할 수 있습니다. 제거하지 않으면, 이들 추가 라우트로 인해 Dispatcher가 작동되지 않습니다. 이를 확인하여 제거하려면, 79 페이지의 『서버 시스템의 시스템 설정』을 참조하십시오.

문제점: 어드바이저가 제대로 작동하지 않음

광역 지원을 사용 중이고 어드바이저가 제대로 작동하는 것으로 보이지 않으면, 어드바이저가 로컬 및 원격 Dispatcher 모두에서 시작되었는지 확인하십시오.

ICMP 핑이 어드바이저 요청 이전에 서버에 발행됩니다. Load Balancer와 서버 사이에 방화벽이 있는 경우, 방화벽 전반에 핑이 지원되도록 하십시오. 이러한 설정이 네트워크 보안에 위험 요소가 된다면, java 등록 정보를 추가하여 dsserver의 java 명령문을 수정하여 모든 핑을 끄도록 설정하십시오.

```
LB_ADV_NO_PING="true"
java -DLB_ADV_NO_PING="true"
```

245 페이지의 『Dispatcher의 광역 지원으로 원격 어드바이저 사용』을 참조하십시오.

문제점: Dispatcher, Microsoft IIS 및 SSL이 작동하지 않음(Windows 플랫폼)

Dispatcher, Microsoft IIS 및 SSL 사용 시 이들이 함께 작동하지 않으면 SSL 보안을 활성화하는 데 문제가 있을 수 있습니다. 키 쌍 생성, 인증 확보, 키 쌍이 있는 인증 설치, SSL이 필요한 디렉토리 구성에 대한 자세한 정보는 *Microsoft 정보 및 피어 웹 서비스* 문서를 참조하십시오.

문제점: 원격 시스템에 대한 Dispatcher 연결

Dispatcher에서는 원격 시스템에 연결하여 이를 구성할 수 있는 키를 사용합니다. 키는 연결에 RMI 포트를 지정합니다. 보안 이유 또는 충돌로 인해 RMI 포트를 변경할 수 있습니다. RMI 포트를 변경하면, 키의 파일 이름은 달라집니다. 동일한 원격 시스템의 키 디렉토리에 둘 이상의 키가 있고 이들이 서로 다른 RMI 포트를 지정하면, 명령행은 발견된 처음 키만 시도합니다. 이 키가 틀렸으면, 연결은 거부됩니다. 틀린 키를 삭제하지 않는 한, 연결되지 않습니다.

문제점: dscontrol 또는 lbadmin 명령 실패

1. dscontrol 명령이 오류: 서버가 응답하지 않음을 리턴합니다. 또는 lbadmin 명령이 오류: RMI 서버를 액세스할 수 없음을 리턴합니다. 시스템에 소켓에 연결된 스택이 있을 때 이러한 오류가 발생할 수 있습니다. 이 문제점을 수정하려면 다음 행을 포함하도록 socks.cnf 파일을 편집하십시오.

```
EXCLUDE-MODULE java
EXCLUDE-MODULE javaw
```

2. Load Balancer 인터페이스의 관리 콘솔(명령행, 그래픽 사용자 인터페이스 및 마법사)은 RMI(원격 메소드 호출)를 사용하여 dsserver와 통신합니다. 기본 통신은 세 개의 포트를 사용하며, 각각의 포트가 dsserver 시작 스크립트에 다음과 같이 설정됩니다.
 - 10099: dscontrol에서 명령을 수신할 경우
 - 10004: Metric Server에 메트릭 조회를 전송할 경우
 - 10199: RMI 서버 포트의 경우

관리 콘솔 중 하나가 방화벽과 동일한 시스템에서 또는 방화벽을 통해 실행될 경우 이들 포트가 문제점을 야기할 수 있습니다. 예를 들어, []를 방화벽과 동일한 시스템에서 실행하고, dscontrol 명령을 발행하면 오류: 서버가 응답하지 않음과 같은 오류 메시지가 나타날 수 있습니다.

이러한 문제점을 방지하려면, dsserver 스크립트 파일을 편집하여 방화벽(또는 기타 응용프로그램)에 대하여 RMI에서 사용하는 포트를 설정하십시오. LB_RMISERVERPORT=10199 행을 LB_RMISERVERPORT=*yourPort*로 변경하십시오. 여기서 *yourPort*는 다른 포트입니다.

완료되면, dsserver를 재시작하고 포트 10099, 10004, 10199 및 10100 또는 관리 콘솔이 실행될 호스트 주소에 대하여 선택된 포트와 통신을 시작하십시오.

3. 이러한 오류는 **dsserver**가 시작되지 않은 경우에도 발생할 수 있습니다.
4. 시스템에 다중 어댑터가 있는 경우, dsserver 스크립트에 다음을 추가하여 dsserver가 사용할 어댑터를 지정해야 합니다.`java.rmi.server.hostname=<host_name or IPaddress>`

(예: `java -Djava.rmi.server.hostname="10.1.1.1"`)

온라인 도움말을 보려고 할 때 문제점: “파일을 찾을 수 없습니다...”라는 오류 메시지 발생(Windows 플랫폼)

Windows 플랫폼의 경우, 기본 브라우저로 Netscape를 사용할 때 “파일 ‘<filename>.html’(또는 컴포넌트 중 하나)을 찾을 수 없습니다. 경로와 파일 이름이 올바르며 모든 필수 라이브러리가 사용 가능한지 확인하십시오.”라는 오류 메시지가 나타날 수 있습니다.

이 문제점은 HTML 파일 연관 설정이 잘못되어서 발생합니다. 해결책은 다음과 같습니다.

1. 내 컴퓨터를 클릭하고 도구를 클릭한 다음 폴더 옵션을 선택하고 파일 형식 탭을 클릭하십시오.
2. “Netscape 하이퍼텍스트 문서”를 선택하십시오.
3. 고급 단추를 클릭하고 열기를 선택한 다음 편집 단추를 클릭하십시오.
4. 응용프로그램: 필드에서 NSShell을 입력하고(작업 수행: 필드에 사용되는 응용프로그램은 아님) 확인을 클릭하십시오.

문제점: GUI(Graphical User Interface)가 올바르게 시작되지 않음

lbadmin인 GUI(Graphical User Interface)가 올바르게 작동하려면 충분한 페이징 공간이 필요합니다. 페이징 공간이 충분하지 않으면 GUI가 완전히 시작되지 않을 수 있습니다. 이런 경우에는 페이징 공간을 확인하고 필요하면 늘리십시오.

문제점: Caching Proxy가 설치된 Dispatcher 실행 중 오류

[[를 설치 제거하여 다른 버전을 다시 설치하고, Dispatcher 컴포넌트를 시작할 때 오류가 발생한 경우 Caching Proxy가 설치되었는지 확인하십시오. Caching Proxy에는 Dispatcher 파일 중 하나에 대한 종속성이 있습니다. 이 파일은 Caching Proxy가 설치 제거될 때만 설치 제거됩니다.

이 문제를 해결하려면 다음을 수행하십시오.

1. Caching Proxy를 설치 제거하십시오.
2. [[를 설치 제거하십시오.
3. [[와 Caching Proxy를 다시 설치하십시오.

문제점: GUI(Graphical User Interface)가 올바르게 표시되지 않음

Load Balancer GUI의 모양과 관련된 문제점이 발생하면 운영 체제의 데스크탑 해상도에 대한 설정을 확인하십시오. GUI는 해상도 1024x768 픽셀에서 가장 잘 보입니다.

문제점: Windows 플랫폼에서 때로 도움말 창이 다른 열린 창 뒤로 사라짐

Windows 플랫폼에서 처음 도움말 창을 열 때 때때로 도움말 창이 기존 창 뒤의 백그라운드로 사라집니다. 이 경우에는 창을 클릭하여 다시 앞으로 가져오십시오.

문제점: Load Balancer가 프레임을 처리하고 전달할 수 없음

Solaris의 경우, 각 네트워크 어댑터에는 기본적으로 동일한 MAC 주소가 있습니다. 이 주소는 각 어댑터가 다른 IP 서브넷에 있을 때 올바르게 작동합니다. 그러나 전환 환경에서 동일한 MAC와 동일한 IP 서브넷 주소를 가진 여러 개의 NIC가 동일한 스위치와 통신할 경우, 스위치는 단일 MAC(및 두 IP)에 바인딩된 모든 통신을 동일한 와이어로 전송합니다. 프레임을 마지막으로 와이어에 놓은 어댑터에만 두 어댑터에 바인딩된 IP 패킷이 보입니다. s는 "잘못된" 인터페이스에 도착한 유효한 IP 주소의 패킷을 버릴 수 있습니다.

모든 네트워크 인터페이스가 ibmlb.conf에 구성된 것으로 Load Balancer에 지정되지 않고, ibmlb.conf에 정의되지 않은 NIC가 프레임을 받으면 Load Balancer가 프레임을 처리하고 전달할 수 없습니다.

이 문제점을 해결하려면 기본값을 덮어쓰고 인터페이스마다 고유한 MAC 주소를 설정해야 합니다. 다음 명령을 사용하십시오.

```
ifconfig interface ether macAddr
```

예를 들어,

```
ifconfig eri0 ether 01:02:03:04:05:06
```

문제점: Load Balancer 실행 프로그램을 시작할 때 파란색 화면이 표시됨

Windows 플랫폼에서 실행 프로그램을 시작하려면 네트워크 카드가 설치되고 구성되어야 합니다.

문제점: Discovery 경로로 인해 Load Balancer와의 리턴 통신이 발생하지 못함

AIX 운영 체제에는 path MTU discovery라는 네트워크 매개변수가 있습니다. 클라이언트와의 트랜잭션 동안 운영 체제가 전송 패킷에 대해 더 작은 최대 전송 단위(MTU)를 사용해야 한다고 판단하면, 해당 데이터를 기억하기 위해 path MTU discovery를 사용하여 AIX는 라우트를 작성합니다. 새 라우트는 해당되는 특정 클라이언트 IP용이며 이 클라이언트에 도달하기 위해 필요한 MTU를 기록합니다.

라우트가 작성되는 동안 루프백에서 별명이 지정되는 클러스터로 인해 서버에서 문제가 발생할 수 있습니다. 라우트의 게이트웨이 주소가 클러스터/넷마스크의 서브넷 범위에 포함되면 AIX 시스템은 루프백에서 라우트를 작성합니다. 이는 해당 주소가 이 서브넷을 사용하여 별명이 지정된 마지막 인터페이스이기 때문에 발생합니다.

예를 들어, 클러스터가 9.37.54.69이고 255.255.255.0 넷마스크가 사용되며 원하는 게이트웨이가 9.37.54.1인 경우 AIX 시스템은 라우트에 대해 루프백을 사용합니다. 이로 인해 서버 응답이 시스템을 떠날 수 없으며 클라이언트 제한시간이 종료됩니다. 일반적으로 클라이언트는 클러스터가 보내는 하나의 응답만을 받으므로 라우트가 작성되고 클라이언트는 더이상 수신하지 않습니다.

이 문제를 해결할 수 있는 두 가지 해결책이 있습니다.

1. AIX 시스템이 동적으로 라우트를 추가하지 않도록 path MTU discovery를 사용 불가능으로 설정하십시오. 다음 명령을 사용하십시오.

no -a AIX 네트워크 설정을 나열합니다.

no -o option=value

AIX 시스템에서 TCP 매개변수를 설정합니다.

2. 255.255.255.255 넷마스크를 사용하여 루프백의 클러스터 IP의 별명을 지정하십시오. 이는 별명이 지정된 서브넷이 유일한 클러스터 IP임을 의미합니다. AIX 시스템이 동적 라우트를 작성하면 대상 게이트웨이 IP가 해당 서브넷과 일치하지 않으므로 라우트가 정확한 네트워크 인터페이스를 사용하게 됩니다. 그리고 나서, 별명 지정 단계에서 작성된 새 lo0 라우트를 삭제하십시오. 이를 위해 네트워크 대상이 클러스터 IP인 루프백에서 라우트를 찾아서 삭제하십시오. 클러스터의 별명이 지정될 때마다 이를 수행해야 합니다.

주:

1. AIX 4.3.2 이하에서 path MTU discovery는 기본적으로 사용 불가능하며, AIX 4.3.3 이상에서는 기본적으로 사용 가능합니다.

2. 다음 명령은 path MTU discovery를 해제하며 시스템을 부팅할 때마다 수행되어야 합니다. 다음 명령을 /etc/rc.net 파일에 추가하십시오.

- -o udp_pmtu_discover=0
- -o tcp_pmtu_discover=0

문제점: Load Balancer의 광역 모드에서 고가용성이 작동되지 않음

Wide Area []를 설치한 경우, 원격 Dispatcher를 로컬 Dispatcher의 클러스터에서 서버로 정의해야 합니다. 일반적으로 원격 Dispatcher의 전달되지 않는 주소(NFA)를 원격 서버의 대상 주소로 사용합니다. 이 경우 원격 Dispatcher에서 고가용성을 설정하면 실패합니다. 이는 원격 사이트에 액세스할 때 NFA를 사용하면 로컬 Dispatcher는 항상 원격 사이트의 기본값을 가리키기 때문입니다.

이 문제를 해결하려면 다음을 수행하십시오.

1. 원격 Dispatcher에서 추가 클러스터를 정의하십시오. 이 클러스터에 대해 포트나 서버를 정의할 필요는 없습니다.
2. 클러스터 주소를 goActive 및 goStandby 스크립트에 추가하십시오.
3. 로컬 Dispatcher에서 원격 기본 Dispatcher의 NFA 대신 이 클러스터 주소를 서버로 정의하십시오.

원격 기본 Dispatcher가 실행되면, 어댑터에서 이 주소를 별명으로 지정하여 통신을 승인합니다. 실패하면 주소는 백업 시스템으로 이동하고 백업 시스템이 해당 주소의 통신을 승인합니다.

문제점: GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.

lbadmin 또는 웹 관리(lbwebaccess)를 사용하여 큰 구성 파일(대략 200개 또는 그 이상의 **add** 명령)을 로드하려고 하면, GUI는 정지하거나 화면에서 매우 느리게 반응하는 예기치 못한 작동을 나타냅니다.

이러한 현상은 Java가 대형 구성을 처리하는 데 충분한 메모리에 액세스할 수 없기 때문에 발생합니다.

Java에 사용 가능한 메모리 할당 풀을 증가시키기 위해 지정할 수 있는 런타임 환경 옵션이 있습니다.

옵션은 -Xmxn이고, 여기서 n은 메모리 할당 풀의 최대 크기(단위: 바이트)를 나타냅니다. n은 1024의 배수이어야 하며 2MB 보다 커야 합니다. n 값 다음에 KB를 표시하는 k 또는 K 및 MB를 표시하는 m 또는 M이 올 수 있습니다. 예를 들어, -Xmx128M 및 -Xmx81920k는 모두 유효합니다. 기본값은 64M입니다. Solaris 8은 최대값이 4000M입니다.

예를 들어, 이 옵션을 추가하려면 lbadm인 스크립트 파일을 편집하여 다음과 같이 "javaw"를 "javaw -Xmxn"으로 수정하십시오(AIX 시스템의 경우, "java"를 "java -Xmxn"으로 수정).

- **AIX 시스템**

```
javaw -Xmx256m -cp $LB_CLASSPATH $LB_INSTALL_PATH $LB_CLIENT_KEYS  
com.ibm.internet.nd.framework.FWK_Main 1>/dev/null 2>&1 &
```

- **HP-UX 시스템**

```
java -Xmx256m -cp $LB_CLASSPATH $LB_INSTALL_PATH $LB_CLIENT_KEYS  
com.ibm.internet.nd.framework.FWK_Main 1>/dev/null 2>&1 &
```

- **Linux 시스템**

```
javaw -Xmx256m -cp $LB_CLASSPATH $LB_INSTALL_PATH $LB_CLIENT_KEYS  
com.ibm.internet.nd.framework.FWK_Main 1>/dev/null 2>&1 &
```

- **Solaris 시스템**

```
java -Xmx256m -cp $LB_CLASSPATH $LB_INSTALL_PATH $LB_CLIENT_KEYS  
com.ibm.internet.nd.framework.FWK_Main 1>/dev/null 2>&1 &
```

- **Windows 시스템**

```
START javaw -Xmx256m -cp %LB_CLASSPATH% %LB_INSTALL_PATH%  
%LB_CLIENT_KEYS% com.ibm.internet.nd.framework.FWK_Main
```

n에 대한 권장값은 없지만, 기본 옵션보다 커야 합니다. 기본값의 두 배로 시작하는 것이 좋습니다.

문제점: 구성 갱신 후 lbadm이 서버로부터 연결이 끊어짐

구성 갱신 후 Load Balancer 관리(lbadm)가 서버로부터 연결이 끊어질 경우, 구성을 시도하는 서버에서 dsserver의 버전을 확인하고 lbadm 또는 dscontrol의 버전과 동일한지 확인하십시오.

문제점: 원격 연결을 통해 IP 주소가 제대로 분석되지 않음

보안 소켓 구현을 통해 원격 클라이언트 사용 시, 완전한 도메인 이름 또는 호스트 이름이 IP 주소 형식 표기법의 올바른 올바른 IP 주소로 분석되지 못할 수 있습니다. 소켓 구현이 특정, 소켓 관련 데이터를 DNS 분석에 추가할 수 있습니다.

IP 주소가 원격 연결을 통해 제대로 분석되지 않을 경우, IP 주소를 IP 주소 표기법 형식으로 지정하십시오.

문제점: 한글 Load Balancer 인터페이스가 AIX 및 Linux 시스템에서 겹치거나 원치 않는 글꼴을 표시함

한글 Load Balancer 인터페이스에서 겹치거나 원하지 않는 글꼴을 수정하려면 다음을 수행하십시오.

- **AIX 시스템**

1. AIX 시스템에서 모든 Java 프로세스를 정지하십시오.

2. 편집기에서 font.properties.ko 파일을 여십시오. 이 파일은 *home/jre/lib*에 있습니다. 여기서 *home*은 Java 홈입니다.
3. 다음 문자열을 탐색하십시오.

```
-Monotype-TimesNewRomanWT-medium-r-normal
---%d-75-75---ksc5601.1987-0
```

4. 문자열의 모든 인스턴스를 다음으로 바꾸십시오.

```
-Monotype-SansMonoWT-medium-r-normal
---%d-75-75---ksc5601.1987-0
```

5. 파일을 저장합니다.

Linux 시스템

1. 시스템에서 모든 Java 프로세스를 중지하십시오.
2. 편집기에서 font.properties.ko 파일을 여십시오. 이 파일은 *home/jre/lib*에 있습니다. 여기서 *home*은 Java 홈입니다.
3. 다음 문자열(공백 없이)을 탐색하십시오.

```
-monotype-
timesnewromanwt-medium-r-normal---%d-75-75-p---microsoft-symbol
```

4. 문자열의 모든 인스턴스를 다음으로 바꾸십시오.

```
-monotype-sansmonowt-medium-r-normal---%d-75-75-p---microsoft-symbol
```

5. 파일을 저장합니다.

문제점: Windows 시스템에서 hostname과 같은 명령을 발행하면 로컬 주소 대신 별명 주소가 리턴됨

Windows 시스템에서 MS 루프백 어댑터의 별명을 지정한 뒤, hostname과 같은 특정 명령을 발행하면 OS가 로컬 주소 대신 별명 주소로 잘못 응답합니다. 이 문제점을 정정하려면, 네트워크 접속 목록에서 새로 추가되는 별명이 로컬 주소 아래에 나열되어야 합니다. 그러면 루프백 별명 전에 로컬 주소가 액세스됩니다.

네트워크 연결 목록을 확인하려면 다음 명령을 실행하십시오.

1. 시작 -> 설정 -> 네트워크 및 전화 접속을 클릭하십시오.
2. 고급 메뉴 옵션에서 고급 설정...을 선택하십시오.
3. 로컬 영역 접속이 접속 상자에서 첫 번째로 나열되는지 확인하십시오.
4. 필요할 경우, 오른쪽의 순서 지정 단추를 사용하여 목록에서 항목을 위 또는 아래로 이동하십시오.

문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동

Windows 플랫폼에서 Matrox AGP 카드 사용 시, Load Balancer GUI에서 예상하지 못한 작동이 발생할 수 있습니다. 마우스를 클릭하면 마우스 포인터보다 약간 큰 영

역의 블록이 손상되어 역으로 강조표시되거나 이미지가 화면 밖으로 이동하는 원인이 될 수 있습니다. 이전 Matrox 카드에서는 이러한 작동을 보이지 않습니다. Matrox AGP 카드 사용 시 알려진 수정 방법은 없습니다.

문제점: "rmmod ibmlb"를 실행할 때 예상치 않은 작동(Linux 시스템)

Linux 시스템에서, Load Balancer 커널 모듈을 수동으로 제거하는 중에 dsserver가 아직 실행 중이면 시스템 정지 및 javacores와 같은 예상치 않은 작동이 발생할 수 있습니다. Load Balancer 커널 모듈을 수동으로 제거하려면 먼저 dsserver를 중지해야 합니다.

"dsserver stop"이 작동하지 않으면, java process with SRV_KNDConfigServer를 정지시키십시오. `ps -ef | grep SRV_KNDConfigServer` 명령을 사용하여 프로세스 ID를 찾은 후 `kill process_id` 명령으로 프로세스를 종료하여 프로세스를 정지하십시오.

안전하게 "rmmod ibmlb" 명령을 실행하여 커널에서 Load Balancer 모듈을 제거할 수 있습니다.

문제점: Dispatcher 시스템에서 명령 실행 중 응답 시간이 느림

로드 밸런스를 위해 Dispatcher 컴포넌트를 실행 중인 경우 컴퓨터에 클라이언트 트래픽이 과부하될 수 있습니다. Load Balancer 커널 모듈은 높은 우선순위를 갖고 있으며 항상 클라이언트 패킷을 처리 중인 경우 나머지 시스템은 응답을 하지 않게 될 수 있습니다. 사용자 공간에서 명령을 실행하면 완료하는 데 시간이 많이 걸리거나 완료되지 못할 수도 있습니다.

이런 현상이 발생할 경우 Load Balancer 시스템에 통신량이 과부하되지 않도록 설정을 다시 구조화해야 합니다. 다른 방법으로는 로드를 여러 Load Balancer 시스템으로 분산시키거나 시스템을 보다 강력하고 빠른 컴퓨터로 교체하는 방법이 있습니다.

통신량이 많아서 시스템에서의 응답 시간이 느린지를 판별하려면 클라이언트 최대 사용 시간에 이런 현상이 발생하는지 검토하십시오. 루프를 경로 지정하는 잘못 구성된 시스템에서도 동일한 증상이 발생할 수 있습니다. 그러나 Load Balancer 설정을 변경하기 전에 증상이 클라이언트 로드가 많기 때문인지를 판별하십시오.

문제점: SSL 또는 HTTPS 어드바이저가 서버 로드를 등록하지 않음(mac 전달을 사용할 경우)

mac 기본 전달 메소드를 사용할 경우 루프백에 별명이 있는 클러스터 주소를 사용하여 서버로 패킷을 전송합니다. 일부 서버 응용프로그램(예: SSL)에서는 구성 정보(예: 인증)가 IP 주소를 기본으로 해야 합니다. 수신 패킷의 콘텐츠와 일치하려면 IP 주소는

루프백에 구성되어 있는 클러스터 주소이어야 합니다. 서버 응용프로그램을 구성할 때 클러스터의 IP 주소를 사용하지 않으면 클라이언트 요청은 서버로 제대로 전달되지 않습니다.

문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐

원격 웹 관리를 사용하여 Load Balancer를 구성할 경우에는 Load Balancer GUI가 나타나는 Netscape 브라우저 창의 크기를 조정하지 마십시오(최소화, 최대화, 복원 등). Netscape는 브라우저 창의 크기를 조정할 때마다 페이지를 재로드하므로 이로 인해 호스트로부터 연결이 끊어질 수 있습니다. 창의 크기를 조정할 때마다 호스트에 다시 연결해야 합니다. Windows 플랫폼에서 원격 웹 관리를 수행할 경우 Internet Explorer를 사용하십시오.

문제점: 소켓 풀링이 사용 가능하며 웹 서버가 0.0.0.0으로 바인드됨

Windows 백엔드 서버에서 Microsoft IIS 서버를 실행할 경우 Microsoft IIS 서버를 바인드 고유 서버로 구성해야 합니다. 그렇지 않은 경우, 기본적으로 소켓 풀링이 사용 가능하며 웹 서버는 0.0.0.0에 바인드하여 사이트에 대한 복수의 ID로 구성된 가상 IP 주소로 바인딩된 통신량이 아닌 모든 통신량을 인식합니다. 소켓 풀링이 사용 가능할 때 로컬 호스트의 응용프로그램이 중지되면, AIX 또는 Windows ND 서버 어드바이저가 이를 발견합니다. 그러나, 로컬 호스트가 가동 중일 때 가상 호스트의 응용프로그램이 중지되면 어드바이저는 실패를 발견하지 않으며 Microsoft IIS는 중지된 응용프로그램을 비롯하여 모든 통신량에 계속해서 응답합니다.

소켓 풀링이 사용 가능한지와 웹 서버가 0.0.0.0으로 바인드하는지 판별하려면 다음 명령을 실행하십시오.

```
netstat -an
```

Microsoft IIS 서버를 바인드 고유 서버(소켓 풀링을 사용 불가능하게 함)로 구성하는 방법에 대한 명령은 Microsoft 제품 지원 서비스 웹 사이트에 있습니다. 이 정보를 보기 위해 다음 URL 중 하나로 이동할 수도 있습니다.

IIS5: 하드웨어 로드 밸런스가 중지된 웹 사이트를 발견하지 않음(Q300509)

<http://support.microsoft.com/default.aspx?scid=kb;en-us;Q300509>

소켓 풀링을 사용 불가능하게 하는 방법(Q238131)

<http://support.microsoft.com/default.aspx?scid=kb;en-us;Q238131>

문제점: Windows 시스템에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨

Windows 2000 운영 체제의 명령 프롬프트 창에서, Latin-1 계열의 일부 자국 문자가 손상된 채로 표시될 수 있습니다. 예를 들어, tilde가 있는 "a"가 파이 기호로 표시될 수 있습니다. 이를 수정하려면, 명령 프롬프트 창의 글꼴 등록 정보를 변경해야 합니다. 글꼴을 변경하려면, 다음을 수행하십시오.

1. 명령 프롬프트 창 상위 왼쪽 모서리에 있는 아이콘을 클릭하십시오.
2. 등록 정보를 선택한 후, 글꼴 탭을 클릭하십시오.
3. 기본 글꼴은 Raster 글꼴입니다. 이를 Lucida Console로 변경한 다음 확인을 클릭하십시오.

문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생

프로세스당 64 스레드만 허용하도록 일부 HP-UX 11i 설치가 사전 구성되었습니다. 그러나 일부 Load Balancer 구성에서는 이 양보다 더 요구합니다. HP-UX 시스템의 경우 프로세스당 스레드를 최소 256으로 설정하십시오. 이 값을 늘리려면 "sam" 유틸리티를 사용하여 max_thread_proc 커널 매개변수를 설정하십시오. 많이 사용할 경우 max_thread_proc를 256 이상으로 늘려야 합니다.

max_thread_proc를 늘리려면 다음을 수행하십시오.

1. 명령행에서 sam을 입력하십시오.
2. 커널 구성 > 구성 가능한 매개변수를 선택하십시오.
3. 이동 막대에서 max_thread_proc를 선택하십시오.
4. 스페이스바를 눌러 max_thread_proc를 강조표시하십시오.
5. 탭을 한 번 누른 다음, 오른쪽 화살표 키를 눌러 조치를 선택하십시오.
6. Enter 키를 눌러 조치 메뉴를 표시한 다음, M을 눌러 구성 가능한 매개변수 수정을 선택하십시오. (이 옵션이 표시되지 않으면 max_thread_proc를 강조표시하십시오.)
7. 수식/값 필드가 선택될 때까지 탭을 누르십시오.
8. 256 이상의 값을 입력하십시오.
9. 확인을 클릭하십시오.
10. 탭을 한 번 누른 다음, 조치를 선택하십시오.
11. 새 커널 프로세스에 대해 K를 누르십시오.
12. 예를 선택하십시오.
13. 시스템을 재부트하십시오.

문제점: Windows 시스템에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함

Load Balancer 시스템에 어댑터를 구성할 경우 어드바이저가 작동하려면 다음 두 설정이 맞는지 확인해야 합니다.

- 일반적으로 3Com 어댑터 카드에서 사용되는 task 오프로드를 사용 불가능하게 하십시오.
 - task 오프로드를 사용 불가능하게 하려면 시작 > 설정 > 제어판 > 네트워크 및 전화 접속 연결로 이동한 다음, 어댑터를 선택하십시오.
 - 팝업 창에서 등록 정보를 클릭하십시오.
 - 구성을 클릭한 다음, 고급 탭을 선택하십시오.
 - 등록 정보 분할창에서 task 오프로드 등록 정보를 선택한 다음, 값 필드에서 사용 불가능을 선택하십시오.
- TCP/IP 필터링을 사용할 경우 IP 프로토콜에 대해 프로토콜 1(ICMP)을 사용 가능하게 하십시오. ICMP가 사용 불가능할 경우 백엔드 서버에 대한 ping 검사가 성공하지 않습니다. ICMP가 사용 가능한지 확인하려면 다음을 수행하십시오.
 - 시작 > 설정 > 제어판 > 네트워크 및 전화 접속 연결로 이동한 다음, 어댑터를 선택하십시오.
 - 팝업 창에서 등록 정보를 클릭하십시오.
 - 컴포넌트 분할창에서 인터넷 프로토콜(TCP/IP)을 선택한 다음, 등록 정보를 클릭하십시오.
 - 고급을 클릭한 다음, 옵션 탭을 선택하십시오.
 - 옵션 분할창에서 TCP/IP 필터링을 선택한 다음, 등록 정보를 클릭하십시오.
 - TCP/IP 필터링 사용 기능을 선택하고 IP 프로토콜에 대해 허용을 선택한 경우 IP 프로토콜 1을 추가해야 합니다. 사용 가능하게 한 기존 TCP 및 UDP 포트 외에 이를 추가해야 합니다.

문제점: Windows 플랫폼에서 어댑터에 둘 이상의 주소를 구성할 경우 호스트 이름에 대한 IP 주소를 확인하는 중에 문제가 발생함

Windows 플랫폼에서 두 이상의 IP 주소를 가진 어댑터를 구성할 경우 호스트 이름과 관련시킬 IP 주소를 먼저 레지스트리에 구성하십시오.

여러 경우(예: lbkeys 작성)에 Load Balancer가 InetAddress.getLocalHost()에 의존하므로, 단일 어댑터로 별명이 지정된 여러 IP 주소를 사용하면 문제가 생길 수 있습니다. 이 문제를 방지하기 위해 호스트 이름을 분석할 IP 주소를 먼저 레지스트리에 나열하십시오. 예를 들어,

1. Regedit를 시작하십시오.
2. 다음 값을 아래와 같이 수정하십시오.

- HKEY_LOCAL_MACHINE -> SYSTEM -> ControlSet001 -> Services -> *YourInterfaceAddress* -> Parameters -> Tcpip -> IPAddress
- 호스트 이름을 분석할 IP 주소를 먼저 배치하십시오.
- HKEY_LOCAL_MACHINE -> SYSTEM -> ControlSet001 -> Services -> Tcpip -> Parameters -> Interfaces -> *YourInterfaceAddress* -> IPAddress
- 호스트 이름을 분석할 IP 주소를 먼저 배치하십시오.
- HKEY_LOCAL_MACHINE -> SYSTEM -> ControlSet002 -> Services -> *YourInterfaceAddress* -> Parameters -> Tcpip -> IPAddress
- 호스트 이름을 분석할 IP 주소를 먼저 배치하십시오.
- HKEY_LOCAL_MACHINE -> SYSTEM -> ControlSet002 -> Services -> Tcpip -> Parameters -> Interfaces -> *YourInterfaceAddress* -> IPAddress
- 호스트 이름을 분석할 IP 주소를 먼저 배치하십시오.
- HKEY_LOCAL_MACHINE -> SYSTEM -> CurrentControlSet -> Services -> *YourInterfaceAddress* -> Parameters -> Tcpip -> IPAddress
- 호스트 이름을 분석할 IP 주소를 먼저 배치하십시오.
- HKEY_LOCAL_MACHINE -> SYSTEM -> CurrentControlSet -> Services -> Tcpip -> Parameters -> Interfaces -> *YourInterfaceAddress* -> IPAddress
- 호스트 이름을 분석할 IP 주소를 먼저 배치하십시오.

3. 재부트

4. 호스트 이름이 올바른 IP 주소로 분석되는지 확인하십시오. 예: `ping yourhostname`.

문제점: Windows 시스템에서 네트워크 정지 후고가용성 설정에서 어드바이저가 작동하지 않음

기본적으로 Windows 운영 체제에서 네트워크 정지를 발견한 경우 모든 정적 항목을 포함한 해당하는 ARP(Address Resolution Protocol)를 지웁니다. 네트워크가 사용 가능하게 된 후 네트워크에 전송된 ARP 요청에 의해 ARP 캐시가 다시 채워집니다.

고가용성 구성에서는 네트워크 연결성 손실이 한 서버나 두 서버 모두에 영향을 미칠 경우 두 서버 모두가 기본 운영을 인계합니다. ARP 캐시를 다시 채우기 위해 ARP 요청을 전송한 경우 두 서버가 응답하기 때문에 ARP 캐시는 항목이 유효하지 않은 것으로 표시합니다. 따라서 어드바이저는 백업 서버에 대한 소켓을 작성할 수 없습니다.

연결성이 손실되었을 때 Windows 운영 체제에서 ARP 캐시를 지우지 못하게 하면 이 문제가 해결됩니다. Microsoft에서 이 task 완료 방법을 설명하는 기사를 발표했습니다. 이 기사는 Microsoft 웹 사이트에 있습니다. 이 웹 사이트 주소는 Microsoft Knowledge Base이고, 기사 번호는 239924: <http://support.microsoft.com/default.aspx?scid=kb;en-us;239924>입니다.

다음은 시스템이 ARP 캐시를 지우는 것을 방지하기 위해 Microsoft 기사에서 설명한 단계를 요약한 것입니다.

1. 레지스트리 편집기(regedit 또는 regedit32)를 사용하여 레지스트리를 여십시오.
2. 레지스트리에서 다음 키를 보십시오.
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Tcpip\Parameters
3. 값 이름이 DisableDHCPMediaSense이고, 값 유형이 REG_DWORD인 레지스트리 값을 추가하십시오.
4. 키를 추가한 후 값을 편집하고 1로 설정하십시오.
5. 변경 사항을 적용하려면 시스템을 재부트하십시오.

주: 그러면 DHCP 설정과 상관없이 ARP 캐시에 영향을 미칩니다.

문제점: 루프백 장치에서 여러 클러스터 별명을 지정할 경우 Linux 시스템에서 "IP address add" 명령을 사용하지 않음

Linux 커널 2.4.x 서버와 Dispatcher의 MAC 전달 메소드를 사용할 경우 특별히 고려해야 할 사항이 있습니다. 서버에 **ip address add** 명령을 사용하여 루프백 장치에 구성된 클러스터 주소가 있을 경우 하나의 클러스터 주소만 별명을 지정할 수 있습니다.

여러 클러스터를 루프백 장치로 별명을 지정할 경우 다음과 같이 **ifconfig** 명령을 사용하십시오.

```
ifconfig lo:num clusterAddress netmask 255.255.255.255 up
```

그리고 인터페이스를 구성하기 위한 ifconfig 메소드와 인터페이스를 구성하기 위한 ip 메소드 간에는 비호환성이 있습니다. 가장 좋은 방법은 사이트가 하나의 메소드를 선택하고 해당 메소드를 독점적으로 사용하는 것입니다.

문제점: "포트 메소드에 대해 지정되지 않은 또는 올바르지 않은 라우터 주소" 오류 메시지

Dispatcher 구성에 서버 추가 시, 다음의 오류 메시지가 나타날 수 있습니다. "오류: 포트 메소드에 대해 지정되지 않은 또는 올바르지 않은 라우터 주소"입니다.

문제점을 판별하려면 다음 점검 목록을 사용하십시오.

- 최신 유지보수 레벨을 적용했는지 확인하십시오.
- Java(Solaris 플랫폼 제외)의 IBM 분배를 사용하고 있는지 확인하십시오.
- Windows 시스템에 DHCP를 사용하도록 구성하지 않았는지 확인하십시오.
- 전달 메소드가 mac(기본값)인 경우 서버, 클러스터 및 최소 하나의 지원되는 NIC는 동일한 서브넷에 있어야 합니다. 예를 들어 10.1.1.1인 클러스터와 130.2.3.4인 서버는 동일한 서브넷에 있지 않으므로 정의될 수 없습니다.

주: 전달 메소드가 nat 또는 cbr인 경우, 서버는 클러스터와 동일한 서브넷에 있지 않아도 됩니다.

- 모두 동일한 서브넷에 있고 클러스터에 별명을 부여한 경우, 해당 서브넷으로 라우트되는 NIC의 클러스터에 별명을 지정했는지 확인하십시오. 예를 들어 13.2.3.4에 en0를, 9.1.2.3에 en1을 지정하고 클러스터 정의가 9.5.7.3인 경우 클러스터는 en1에 지정되어야 합니다. 기본 인터페이스는 en0입니다.
- Linux 플랫폼에서는 loadoutput.log 파일에 대해 /usr/lpp/ibm/internet/nd/logs/dispatcher 디렉토리 위치를 점검하여 올바른 커널을 로드했는지 확인하십시오. 보고된 오류 모두에 대해 이 파일을 점검하십시오.

라우터 매개변수의 기본값이 0인 경우는 로컬 서버임을 표시합니다. 서버의 라우터 주소를 0이 아닌 값으로 설정했다면 다른 서브넷에 있는 원격 서버를 표시합니다. server add 명령의 router 매개변수에 대한 자세한 정보는 418 페이지의 『dscontrol server — 서버 구성』을 참조하십시오.

추가 서버가 다른 서브넷에 위치하면, 라우터 매개변수는 원격 서버와의 통신을 위해 로컬 서브넷에서 쓰이는 라우터 주소여야 합니다.

문제점: Solaris 시스템에서는 Load Balancer 프로세스가 시작된 터미널 창을 종료할 때 프로세스가 종료됨

Solaris 시스템에서는 dsserver 또는 lbadmin 같은 Load Balancer 스크립트를 터미널 창에서 시작한 후 같은 창에서 종료하면, Load Balancer 프로세스도 존재합니다.

이 문제점을 해결하려면 **nohup** 명령으로 Load Balancer 스크립트를 시작하십시오. (예: **nohup dsserver**). 해당 명령은 터미널 세션에서 시작된 프로세스가 종료 시 같은 터미널에서 끊기 신호를 수신하지 못하도록 하며, 터미널 세션이 종료된 후에도 프로세스를 계속하도록 합니다. 터미널 세션 종료 후에도 계속 처리하고 싶은 Load Balancer 스크립트 앞에 **nohup** 명령을 사용하십시오.

문제점: Load Balancer 구성 로드 시 지연

Load Balancer 구성 로드 시 시간이 오래 걸리는 것은 서버 주소를 해석 및 확인하기 위해 작성된 DNS(Domain Name System) 호출이 원인일 수 있습니다.

Load Balancer 시스템의 DNS가 부정확하게 구성된 경우 또는 일반 DNS의 시간이 오래 걸린다면, 네트워크에 DNS 요청을 전송하는 Java 프로세스 때문에 구성 로드 속도가 느려질 것입니다.

해결책은 사용자의 /etc/hosts 파일에 서버 주소 및 호스트 이름을 추가하는 것입니다.

문제점: Windows 시스템에서 IP 주소 충돌 오류 메시지가 나타남

고가용성이 구성되는 경우, 클러스터 주소가 잠시 동안 두 시스템 모두에 구성되어 다음의 오류 메시지가 나타날 수 있습니다. "네트워크에 다른 시스템과의 IP 주소 충돌이 있습니다"입니다. 이 경우 해당 메시지를 안전하게 무시할 수 있습니다. 어느 한 쪽의 시스템을 시작할 때 또는 인계가 시작될 때, 클러스터 주소가 잠시 동안 두 고가용성 시스템 모두에 동시 구성될 수 있습니다.

go* 스크립트를 점검하여 클러스터 주소를 바르게 구성하고 해체했는지 확인하십시오. 구성 파일을 호출하여 go* 스크립트를 설치했으면, go* 스크립트의 명령 구성 및 해체와 충돌을 일으키므로 구성 파일의 클러스터 주소에 대해 "executor configure" 명령문이 없음을 확인하십시오.

고가용성 구성 시 go* 스크립트에 대한 자세한 정보는 223 페이지의 『스크립트 사용』을 참조하십시오.

문제점: 고가용성 구성에서 기본 시스템 및 백업 시스템 모두가 활성화됨

이 문제점은 go 스크립트가 기본 시스템 또는 백업 시스템에서 실행되지 않는 경우 발생할 수 있습니다. dsserver가 두 시스템 모두에서 시작되지 않으면 go 스크립트를 실행할 수 없습니다. 두 시스템을 점검하여 dsserver가 실행 중임을 확인하십시오.

문제점: 대형 페이지 응답 리턴 시 클라이언트 요청 실패

최대 전송 단위(MTU)가 Dispatcher 시스템에 바르게 설정되지 않은 경우, 대형 페이지 응답을 초래하는 클라이언트 요청은 제한시간 초과가 됩니다. Dispatcher 컴포넌트의 cbr 및 nat 전달 메소드의 경우, Dispatcher가 값을 조정하지 않고 MTU값을 기본으로 하기 때문에 이런 현상이 발생할 수 있습니다.

MTU가 통신 매체 유형(예: 이더넷 또는 토큰링)에 기반한 각 운영 체제에 설정됩니다. 다른 유형의 통신 매체와 연결된 경우 로컬 세그먼트의 라우터에는 좀더 작은 MTU 세트가 있을 수 있습니다. 표준 TCP 통신량에서는 연결 설정 중에 MTU가 조정되며, 가장 작은 MTU가 시스템 간 데이터 전송에 사용됩니다.

Dispatcher는 TCP 연결 엔드포인트로 활동하기 때문에 cbr 또는 nat 전달 메소드에 대한 MTU 조정을 지원하지 않습니다. cbr 및 nat 전달의 경우 Dispatcher는 MTU 값을 1500으로 기본 설정합니다. 이 값은 표준 이더넷의 일반 MTU 크기이므로 대부분의 고객은 이 설정을 조정할 필요가 없습니다.

Dispatcher의 cbr 또는 nat 전달 메소드 사용 시, 좀더 낮은 MTU가 있는 로컬 세그먼트에 라우터가 있으면 MTU를 Dispatcher 시스템에 설정하여 낮은 MTU와 일치시켜야 합니다.

이 문제점을 해결하려면 dscontrol executor set mss new_value 명령을 사용하여 최대 세그먼트 크기(mss) 값을 설정하십시오.

예를 들어,

```
dscontrol executor set mss 1400
```

mss의 기본값은 1460입니다.

mss 설정은 Dispatcher의 mac 전달 메소드 또는 Load Balancer의 Dispatcher가 아닌 컴포넌트에 대해서는 적용되지 않습니다.

문제점: Windows 시스템에서 dscontrol 또는 lbadmin 명령을 발행하면 "서버가 응답하지 않음" 오류가 발생

두 개 이상의 IP 주소가 Windows 시스템에 존재하고 호스트 파일이 호스트 이름과 연결할 주소를 지정하지 않은 경우, 운영 체제는 호스트 이름과 연결할 제일 작은 주소를 선택합니다.

이 문제점을 해결하려면 호스트 이름과 연결하고자 하는 주소와 함께 `c:\Windows\system32\drivers\etc\hosts` 파일을 갱신하십시오.

중요: IP 주소는 클러스터 주소일 수 없습니다.

문제점: 고가용성 Dispatcher 시스템의 qeth 드라이버에 있는 S/390 시스템용 Linux에 동기화가 불가능할 수도 있음

qeth 네트워크 드라이버가 있는 S/390 시스템용 Linux에 고가용성을 사용하는 경우, 활성 및 대기 Dispatcher가 동기화에 실패할 수 있습니다. 해당 문제점은 Linux Kernel 2.6에만 제한될 수 있습니다.

이 문제점이 발생하면 다음의 해결책을 사용하십시오.

활성 및 대기 Dispatcher 이미지 사이에 채널 대 채널(CTC) 네트워크 장치를 정의하고 두 CTC 엔드포인트 IP 주소 사이에 하트비트를 추가하십시오.

문제점: 고가용성 구성에 대한 팁

상대 기본 시스템이 실패했거나 종료한 경우 상대 시스템은 Load Balancer의 고가용성 기능을 사용하여 로드 밸런스를 인계할 수 있습니다. 고가용성 상대 시스템 간의 연결 유지를 위해 두 시스템 사이에 연결 레코드가 전달됩니다. 상대 백업 시스템이 로드 밸런스 기능을 인계할 경우, 클러스터 IP 주소가 백업 시스템에서 제거되어 새 기본 시스템에 추가됩니다. 해당 인계 조작에 영향을 미치는 타이밍 및 구성 고려사항이 많이 있습니다.

고가용성 구성 문제에서 기인한 다음과 같은 문제점의 완화에 도움이 되는 팁이 이 섹션에 나열되어 있습니다.

- 인계 후에 연결이 끊김
- 상대 시스템과 동기화가 불가능함

- 요청이 백업 상대 시스템으로 잘못 방향 지정됨

다음 팀은 Load Balancer 시스템 고가용성 구성에 유용합니다.

- 스크립트 파일의 고가용성 명령 위치는 현저한 차이를 만들 수 있습니다.

고가용성 명령의 예제는 다음과 같습니다.

```
dscontrol highavailability heartbeat add ...
dscontrol highavailability backup add ...
dscontrol highavailability reach add ...
```

대부분의 경우 고가용성 정의를 파일 끝에 위치해야 합니다. 클러스터, 포트 및 서버 명령은 고가용성 명령문 앞에 위치해야 합니다. 이는 고가용성 동기화 시, 연결 레코드를 수신하면 클러스터, 포트 및 서버 정의를 찾기 때문입니다.

클러스터, 포트 및 서버가 존재하지 않으면 연결 레코드가 삭제됩니다. 인계가 발생하고 연결 레코드가 상대 시스템에 복제되지 않은 경우 연결에 실패합니다.

이 규칙의 예외는 MAC 전달 메소드로 구성된 결합 배치 서버를 사용할 때입니다. 이 경우 고가용성 명령문은 결합 배치된 서버 명령문의 앞에 와야 합니다. 고가용성 명령문이 결합 배치된 서버 명령문 앞에 오지 않으면, Load Balancer는 결합 배치된 서버에 대한 요청을 받지만 클러스터에 대한 수신 요청과 동일하게 나타나며 로드 밸런싱됩니다. 이는 네트워크의 패킷 루프 및 통신량 초과를 가져올 수 있습니다. 고가용성 명령문이 결합 배치된 서버 앞에 오는 경우, Load Balancer는 ACTIVE 상태가 아니면 수신 통신량을 전달해서는 안 된다는 사실을 인지합니다.

- z/OS 또는 OS/390 운영 체제에서는 하이퍼바이저가 인터페이스를 제어하고 게스트 운영 체제 간의 실제 인터페이스를 다중 송신합니다. 하이퍼바이저는 한 번에 한 게스트만을 허용하여 그 자신을 IP 주소로 등록하며 갱신 창이 있습니다. 클러스터 IP가 백업 시스템에서 제거되면 기본 시스템에 클러스터 IP를 추가하기 전에 지연을 추가해야 할 수도 있으며, 그렇지 않은 경우 클러스터 IP가 실패하고 수신 연결이 처리되지 않습니다.

이 동작을 정정하려면 goActive 스크립트에 휴면 지연을 추가하십시오. 휴면에 필요한 시간은 배치 종속적입니다. 휴면 지연 시간 기본값 10을 권장합니다.

- 고가용성 상대는 서로 핑이 가능해야 하며 동일한 서브넷에 있어야 합니다.

시스템이 0.5초마다 서로 통신하고 4번의 시도 실패 후에는 실패를 감지하도록 하는 것이 기본값입니다. 이 설정은 시스템이 사용 중이나 제대로 기능하고 있으면 장애를 극복하도록 할 수 있습니다. 실패할 때까지 다음의 명령을 발행하여 시간을 증가시킬 수 있습니다.

```
dscontrol executor set hatimeout <value>
```

- 상대 시스템 동기화 시, 모든 연결 레코드가 활성 시스템에서 백업 시스템으로 전송됩니다. 동기화는 기본 제한값인 50초 내에 완료되어야 합니다.

제한 시간 내에 동기화를 완료하려면 시간 확장량에 필요한 메모리에 이전 연결이 있으면 안 됩니다. 특히 LDAP 포트 및 대형 staletimeout 기간(하루 초과)에 문제가 있습니다. 대형 staletimeout 기간은 메모리에 이전 연결을 유지하여 동기화 시 전달해야 하는 연결 기록이 더 많아지는 원인이 되고 두 시스템 모두에 더 많은 메모리 소비를 가져 옵니다.

합리적인 staletimeout 기간으로 동기화하는데 실패한 경우 다음의 명령을 발행하여 동기화 제한시간을 늘릴 수 있습니다.

```
e xm 33 5 new_timeout
```

이 명령은 저장 시에 구성 파일에는 저장되지 않으므로, 시스템 종료 간에 이 설정을 유지하려면 수동으로 구성 파일에 추가하여야 합니다.

제한시간 값은 0.5초가 지나면 저장되므로, new_timeout의 기본값은 100(50 초)입니다.

- 상대 시스템은 워크로드 인계 시 무상 ARP 응답을 발행하여 클러스터 IP 주소와 연결할 새 하드웨어 주소의 동일한 서브넷에 있는 시스템을 구분합니다. 라우터가 무상 ARP를 받아서 캐시 갱신을 했는지 확인해야 하며, 그렇지 않으면 요청이 비활성 상대 시스템으로 전송됩니다.

주: 고가용성 기능 구성에 대한 정보는 218 페이지의 『고가용성』을 참조하십시오.

문제점: Linux에서는 개방형 시스템 어댑터(OSA) 카드가 있는 zSeries 또는 S/390 서버 사용 시 Dispatcher 구성 제한사항이 있음

일반적으로 MAC 전달 메소드 사용 시, Load Balancer 구성의 서버는 플랫폼에 관계 없이 동일한 네트워크 세그먼트에 있어야 합니다. 라우터, 브릿지 및 방화벽 같은 활성 네트워크는 Load Balancer를 간섭합니다. 이는 Load Balancer가 특수 라우터로 기능하며 링크 레이어 헤더만을 다음 및 최종 홉에 수정하기 때문입니다. 다음 홉이 최종 홉이 아닌 네트워크 토폴로지는 Load Balancer에 대해 유효하지 않습니다.

주: 채널 대 채널(CTC), 사용자간 통신 장치(IUCV) 같은 터널은 종종 지원됩니다. 그러나 Load Balancer는 터널 전반에 걸쳐 최종 대상에 전달되어야 하며, 네트워크 대 네트워크 터널일 수 없습니다.

해당 어댑터가 대부분의 네트워크 카드와 다르게 작동하므로 OSA 카드를 공유하는 zSeries 및 S/390 서버에 대한 제한사항이 있습니다. OSA 카드에는 뒤에 Linux 및 z/OS 호스트에 표시되는, 이더넷과 관련없는 고유한 가상 링크 레이어 구현이 있습니다. 각각의 OSA 카드는 OSA 호스트가 아니라 이더넷 대 이더넷 호스트처럼 보이며, 카드를 사용하는 호스트는 자신이 이더넷인 것처럼 반응합니다.

OSA는 IP 레이어와 직접적으로 관련 있는 일부 기능도 수행합니다. 주소 결정 프로토콜(ARP) 요청에 응답하는 것이 해당 기능의 한 예입니다. 다른 예로는 공유 OSA가

전환할 레이어로 이더넷 주소에 기반한 IP 패킷 대신 대상 IP 주소에 기반한 IP 패킷을 라우트하는 것을 들 수 있습니다. OSA 카드는 그 자신에 브릿지된 네트워크 세그먼트입니다.

S/390 Linux 또는 zSeries Linux 호스트에서 실행되는 Load Balancer는 이더넷의 호스트 또는 동일한 OSA의 호스트에 전달될 수 있습니다. 동일한 공유 OSA의 호스트 모두는 효과적으로 동일한 세그먼트에 있습니다.

OSA 브릿지의 특성 상 Load Balancer는 공유 OSA에서 외부로 전달 가능합니다. 브릿지는 클러스터 IP를 소유한 OSA 포트를 인지합니다. 브릿지는 이더넷 세그먼트에 직접 연결된 호스트의 MAC 주소를 인지합니다. 그러므로 Load Balancer는 하나의 OSA 브릿지 전반에 MAC 전달을 실행할 수 있습니다.

그러나 Load Balancer는 공유된 OSA로 전달할 수 없습니다. 백엔드 서버가 Load Balancer와 다른 OSA 카드에 있는 경우, S/390 Linux의 Load Balancer가 포함됩니다. 백엔드 서버의 OSA는 서버 IP에 대한 OSA MAC 주소를 광고하지만, 패킷이 서버의 OSA 이더넷 대상 주소 및 클러스터 IP와 함께 도착한 경우 서버의 OSA 카드는 패킷을 수신할 호스트를 인지하지 못합니다. OSA 대 이더넷의 하나의 공유 OSA MAC 전달을 허용하는 동일한 규칙이 공유 OSA로 전달 시 지속되지 않습니다.

해결책:

OSA 카드가 있는 zSeries 또는 S/390 서버를 사용하는 Load Balancer에는 기술된 문제점을 해결하기 위한 두 가지 접근이 가능합니다.

1. 플랫폼 기능 사용

Load Balancer 구성의 서버가 동일한 zSeries 또는 S/390 플랫폼 유형이면, Load Balancer와 각 서버 사이에 지점 대 지점(CTC 또는 IUCV) 연결을 지정할 수 있습니다. 개인용 IP 주소를 사용하여 엔드포인트를 설정하십시오. 지점 대 지점 연결은 Load Balancer 대 서버 통신량에만 사용됩니다. 그리고 터널의 서버 엔드포인트에 있는 IP 주소를 가진 서버를 추가하십시오. 이 구성을 사용하여 클러스터 통신량이 Load Balancer OSA 카드를 통해 전달되며, 고유의 기본 라우트를 통해 서버가 응답하는 지점 대 지점 연결에 전달됩니다. 응답은 남은 서버의 OSA 카드를 사용합니다.

2. Load Balancer의 GRE 기능 사용

주: 참고: IPv4 및 IPv6용 Load Balancer의 듀얼 프로토콜에서는 GRE 기능을 사용할 수 없습니다.

Load Balancer 구성의 서버가 동일한 zSeries 또는 S/390 플랫폼 유형에 있지 않거나 Load Balancer와 각 서버 사이에 지점 대 지점 연결을 정의하는 것이 불가능한 경우, Load Balancer가 라우터 전반에 전달하는 것을 허용하는 프로토콜인 Load Balancer의 GRE(Generic Routing Encapsulation) 기능 사용을 권장합니다.

GRE 사용 시, Load Balancer가 클라이언트 -> 클러스터 IP 패킷을 수신, 캡슐화하여 서버에 전송합니다. 서버에서는 원래의 클라이언트 -> 클러스터 IP 패킷을 풀어 클라이언트에 직접 응답합니다. GRE 사용의 장점은 Load Balancer가 서버 대 클라이언트 통신량을 참조하는 것이 아니라 클라이언트 대 서버 통신량만을 참조한다는 점입니다. 캡슐화된 오버헤드 때문에 TCP의 최대 세그먼트 크기가 줄어드는 단점이 있습니다.

Load Balancer를 GRE 캡슐화와 함께 전달하도록 구성하려면 다음 명령을 사용하여 서버를 추가하십시오.

```
dscontrol server add cluster_add:port:backend_server router
backend_server
```

Load Balancer 및 백엔드 서버가 동일한 IP 서브넷에 있는 경우 라우터 backend_server가 유효한 지점입니다. 동일한 서브넷에 있지 않은 경우 유효한 다음 홉 IP 주소를 라우터로 지정하십시오.

Linux 시스템이 고유한 GRE 캡슐화된 정보 풀기를 수행하도록 구성하려면 각 백엔드 서버에 대해 다음의 명령을 발행하십시오.

```
modprobe ip_gre
ip tunnel add grelb0 mode gre ikey 3735928559
ip link set grelb0 up
ip addr add cluster_addr dev grelb0
```

주: 백엔드 서버의 루프백에 클러스터 주소를 정의하지 마십시오. z/OS 백엔드 서버 사용 시, z/OS 지정 명령을 사용하여 서버가 GRE 캡슐화된 정보 풀기를 수행하도록 구성하십시오.

문제점: 일부 Linux 버전에서는 관리자 및 어드바이저로 구성된 Dispatcher 실행 시 메모리 누수가 발생함

관리자 및 어드바이저로 구성된 Load Balancer 실행 시, 일부 Red Hat Linux 버전에서 대형 메모리 누수가 발생할 수 있습니다. 어드바이저에 대해 시간 간격을 작게 구성하면 Java 메모리 누수가 증가합니다.

Red Hat Enterprise Linux 3.0과 같은 일부 Linux 분배가 실린 NPTL(Native POSIX Thread Library) 및 JVM의 IBM Java SDK 버전은 메모리 누수를 야기할 수 있습니다. 확장된 스레드 라이브러리 NPTL은 NPTL을 지원하는 Red Hat Linux 3.0 같은 일부 Linux 시스템 분배와 함께 실행됩니다.

해당 시스템이 실린 IBM Java SDK 및 Linux 시스템에 대한 최신 정보는 <http://www.ibm.com/developerworks/java/jdk/linux/tested.html>을 참조하십시오.

문제점 판별 도구로 vmstat 또는 ps 명령을 사용하여 메모리 누수를 감지하십시오.

메모리 누수를 수정하려면 Load Balancer 시스템을 실행하기 전에 다음 명령을 발행하여 NPTL 라이브러리 사용을 불가능하게 하십시오.

```
export LD_ASSUME_KERNEL=2.4.10
```

문제점: SUSE Linux Enterprise Server 9에서는 Dispatcher가 패킷을 전달하지만 패킷은 백엔드 서버에 도달하지 않음

Suse Linux Enterprise Server 9에서 MAC 전달 메소드 사용 시 Dispatcher 보고서는 패킷이 전달되었다고 표시하지만(패킷 계수 증가), 실제 패킷은 백엔드 서버에 도달하지 않습니다.

해당 문제점 발생 시 아래 메세지 중 하나 이상을 볼 수 있습니다.

- 시스템측에서 다음 메시지가 표시됩니다.

```
ip_finish_output2: No header cache and no neighbour!
```

- 클라이언트측에서 다음 메시지가 표시됩니다.

```
ICMP Destination unreachable: Fragmentation Needed
```

해당 문제점은 로드된 iptables NAT 모듈 때문에 발생할 수 있습니다. SLES9에는 iptables의 해당 버전에 Dispatcher와 상호작용 시 특이한 동작을 일으키는 발생 기능 하지만 확인되지 않은 오류가 있습니다.

해결방안:

iptables NAT 모듈 및 연결 추적 모듈을 로드 해제하십시오.

예를 들어,

```
# lsmod | grep ip
    iptable_filter          3072  0
    iptable_nat             22060  0
    ip_conntrack            32560  1 iptable_nat
    ip_tables               17280  2
iptables_filter,iptable_nat
    ipv6                   236800  19
    # rmmod iptable_nat
    # rmmod ip_conntrack
```

사용 순서대로 모듈을 제거하십시오. 참조 계수(lsmod 출력의 마지막 컬럼)가 0인 경우에만 모듈을 제거할 수 있습니다. iptables에 구성한 규칙도 제거해야 합니다.

(예: iptables -t nat -F).

iptable_nat 모듈은 ip_conntrack을 사용하므로, iptable_nat 모듈을 먼저 제거한 다음 ip_conntrack 모듈을 제거하십시오.

주: 테이블에 구성된 규칙을 나열하면 해당 모듈이 로드업되는데 예를 들면 다음과 같습니다. iptables -t nat -L입니다. 모듈 제거 후에 이를 실행하지 않도록 하십시오.

문제점: Windows 시스템에서는 고가용성 인계 중에 IP 주소 충돌 메시지가 나타남

Windows 시스템에서 Load Balancer의 고가용성 기능을 사용하는 경우, 인계 발생 시 활성 Load Balancer의 클러스터 IP를 구성하고 백업 시스템에 클러스터 IP를 해체하는데 goScript를 씁니다. 활성 시스템에 클러스터 IP 주소를 구성하는 goScript가 goScript 전에 실행되어 백업 시스템에 있는 IP 클러스터 주소를 해체하면 문제점이 발생할 수 있습니다. 시스템이 IP 주소 충돌을 감지했다는 팝업 창을 볼 수도 있습니다. `ipconfig /all` 명령을 실행하는 경우, 시스템에 IP 주소 0.0.0.0이 존재하는 것을 볼 수도 있습니다.

해결방안:

다음 명령을 발행하여 기본 시스템에서 클러스터 IP 주소를 수동으로 해체하십시오.

```
dscontrol executor unconfigure clusterIP
```

Windows IP 스택에서 0.0.0.0 주소를 제거합니다.

고가용성 상대 시스템이 클러스터 IP 주소를 해제한 후, 다음 명령을 발행하여 클러스터 IP를 수동으로 다시 추가하십시오.

```
dscontrol executor configure clusterIP
```

이 명령을 발행한 후 다음 명령을 발행하여 Windows IP 스택을 다시 확인하십시오.

```
ipconfig /all
```

문제점: Linux iptables가 패킷 라우팅에 간섭할 수 있음

Linux iptables는 통신량의 로드 밸런스를 간섭할 수 있으므로 Dispatcher 시스템에서는 사용 불가능하게 해야 합니다.

다음 명령을 발행하여 iptables가 로드되었는지 판별하십시오.

```
lsmod | grep ip_tables
```

앞의 명령의 결과물은 다음과 유사할 것입니다.

```
ip_tables          22400    3
iptables_mangle,iptable_nat,iptable_filter
```

결과물에 나열된 각 iptable에 대해 다음 명령을 발행하여 테이블 규칙을 표시하십시오.

```
iptables -t <short_name> -L
```

예를 들어,

```
iptables -t mangle -L
iptables -t nat -L
iptables -t filter -L
```


iptables_nat이 로드된 경우, 로드 해제해야 합니다. iptables_nat은 iptables_contrack에 종속되므로, iptables_contrack도 제거되어야 합니다. 다음 명령을 발행하여 두 iptables을 로드 해제 하십시오.

```
rmmod iptables_nat iptables_contrack
```

문제점: IPv6 서버를 Solaris 시스템의 Load Balancer 구성에 추가할 수 없음

Solaris 시스템의 경우 IPv4 및 IPv6용 Load Balancer 설치에 IPv6 서버를 구성하려고 하면 서버를 추가할 수 없음이라는 메시지가 나타납니다. 해당 오류는 Solaris 운영 체제가 IPv6 주소에 필요한 핑 요청을 핸들하는 방법 때문에 발생할 수 있습니다.

Solaris 시스템에서 서버를 구성에 추가 시, Load Balancer가 서버를 핑하여 서버의 MAC 주소를 획득하려 합니다. Solaris 시스템은 시스템의 NFA 주소 대신 구성된 클러스터 주소를 핑 요청의 소스 주소로 선택할 수 있습니다. 클러스터 주소가 서버 루프백에 구성된 경우 핑 응답은 Load Balancer 시스템에 수신되지 않으므로, 구성에 서버를 추가할 필요가 없습니다.

해결방안은 IPv6 클러스터 주소 설정 전이나 후에 Load Balancer 시스템에 다른 IPv6 주소를 구성하는 것입니다. 이 주소는 사용자가 Load Balancer 구성에 추가하려고 하는 백엔드 서버의 루프백에 별명 지정된 주소여야 합니다. 그리고 서버를 Load Balancer 구성에 추가하십시오.

서비스 수정사항 설치 시 Java 경고 메시지가 나타남

Load Balancer는 제품 설치와 함께 설정된 Java 파일을 제공합니다. 제품 설치의 동일한 시스템에 설치할 필요가 없는 여러 개의 패키지로 이루어집니다. Metric Server 패키지, 관리 패키지 및 기본 패키지를 예로 들 수 있습니다. 해당 코드 패키지는 모두는 조작 시 Java 파일 세트가 필요하지만, 세 패키지 각각은 개별 시스템에 설치될 수 있습니다. 패키지 각각은 Java 파일 세트를 설치합니다. 동일한 시스템에 설치된 경우 Java 파일 세트는 해당 파일 세트 각각이 소유합니다. 두 번째 및 세 번째 파일 세트 설치 시, 다른 파일 세트가 Java 파일 세트를 소유했다는 경고 메시지를 받게 됩니다.

고유한 설치 방법(예: AIX의 installp)을 사용하여 코드를 설치할 때, 다른 파일 세트가 Java 파일 세트를 소유했다는 경고 메시지를 무시해야 합니다.

Load Balancer 설치가 공급된 Java 파일 세트 업그레이드

Load Balancer 설치 프로세스 중에 Java 파일 세트도 설치됩니다. Load Balancer는 제품과 함께 설치되는 Java 버전을 사용하는 유일한 응용프로그램입니다. 사용자 단독으로 Java 파일 세트를 업그레이드해서는 안 됩니다. Java 파일 세트 업그레이드가 필요한 문제점의 경우, IBM 서비스에 보고하여 Load Balancer가 실린 Java 파일 세트가 공식 픽스 레벨로 업그레이드되도록 해야 합니다.

공통 문제점 해결—CBR

문제점: CBR이 실행되지 않음

이 문제점은 다른 응용프로그램에서 CBR이 사용하는 포트 중 하나를 사용하는 경우에 발생할 수 있습니다. 자세한 내용은 319 페이지의 『CBR 포트 번호 확인』을 참조하십시오.

문제점: cbrcontrol 또는 lbadmin 명령 실패

1. cbrcontrol 명령이 오류: 서버가 응답하지 않습니다를 리턴합니다. 또는 lbadmin 명령이 오류: RMI 서버를 액세스할 수 없습니다를 리턴합니다. 시스템에 소켓에 연결된 스택이 있을 때 이러한 오류가 발생할 수 있습니다. 이 문제점을 수정하려면 다음 행을 포함하도록 socks.cnf 파일을 편집하십시오.

```
EXCLUDE-MODULE java
EXCLUDE-MODULE javaw
```

2. Load Balancer 인터페이스의 관리 콘솔(명령행, 그래픽 사용자 인터페이스 및 마법사)은 RMI(원격 메소드 호출)를 사용하여 cbrserver와 통신합니다. 기본 통신은 세 개의 포트를 사용하며, 각각의 포트가 cbrserver 시작 스크립트에 다음과 같이 설정됩니다.

- 11099: cbrcontrol에서 명령을 수신할 경우
- 10004: Metric Server에 메트릭 조회를 전송할 경우
- 11199: RMI 서버 포트의 경우

관리 콘솔 중 하나가 방화벽과 동일한 시스템에서 또는 방화벽을 통해 실행될 경우 이들 포트가 문제점을 야기할 수 있습니다. 예를 들어, []를 방화벽과 동일한 시스템에서 실행하고, cbrcontrol 명령을 발행하면 오류: 서버가 응답하지 않음과 같은 오류 메시지가 나타날 수 있습니다.

이러한 문제점을 방지하려면, cbrserver 스크립트 파일을 편집하여 방화벽(또는 기타 응용프로그램)에 대하여 RMI에서 사용하는 포트를 설정하십시오.

LB_RMISERVERPORT=11199 행을 LB_RMISERVERPORT=*yourPort*로 변경하십시오. 여기서 *yourPort*는 다른 포트입니다.

완료되면, cbrserver를 재시작하고 포트 11099, 10004, 11199 및 11100 또는 관리 콘솔이 실행될 호스트 주소에 대하여 선택된 포트와 통신을 시작하십시오.

3. 이러한 오류는 cbrserver가 시작되지 않은 경우에도 발생할 수 있습니다.

문제점: 요청이 로드 밸런스되지 않음

Caching Proxy 및 CBR이 시작되었지만 요청이 로드 밸런스되지 않고 있습니다. 이 오류는 실행 프로그램을 시작하기 전에 Caching Proxy를 시작한 경우 발생할 수 있습니다. 이 오류가 발생하면 Caching Proxy의 stderr 로그에는 "ndServerInit: 실행 프

로그래밍에 접속할 수 없습니다.”라는 오류 메시지가 포함됩니다. 이러한 문제점이 발생하지 않도록 Caching Proxy를 시작하기 전에 실행 프로그램을 시작하십시오.

문제점: Solaris 시스템에서 **cbrcontrol executor start** 명령 실패

Solaris 시스템에서 **cbrcontrol executor start** 명령이 “오류: 실행 프로그램이 시작되지 않았습니다.”라는 메시지를 리턴합니다. 공유 메모리 세그먼트의 최대 크기 및 세마포어 ID의 수가 운영 체제 기본값보다 큰 값을 갖도록 시스템의 프로세스 간 통신(IPC)을 구성하지 않은 경우 이 오류가 발생합니다. 공유 메모리 세그먼트의 크기 및 세마포어 ID 수를 늘리려면 **/etc/system** 파일을 편집해야 합니다. 이 파일의 구성 방법에 대한 자세한 정보는 128 페이지를 참조하십시오.

문제점: 구문 또는 구성 오류

URL 규칙이 작동하지 않으면 구문 또는 구성 오류로 인할 수 있습니다. 이 문제가 발생하면 다음을 확인하십시오.

- 규칙이 제대로 구성되어 있는지 확인하십시오. 507 페이지의 부록 B 『컨텐츠 규칙(패턴) 구문』에서 자세한 정보를 참조하십시오.
- 이 규칙에 대해 **cbrcontrol rule report**를 발행하고 ‘실패한 시간’ 컬럼을 확인하여 수행된 요청 수에 따라 이 항목이 증가되었는지 확인하십시오. 제대로 증가된 경우 서버 구성을 다시 확인하십시오.
- 규칙이 제대로 실행되지 않으면 ‘항상 참’ 규칙을 추가하십시오. ‘항상 참’ 규칙에 대해 **cbrcontrol rule report**를 발행하여 제대로 실행되는지 확인하십시오.

문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동

Windows 플랫폼에서 Matrox AGP 카드 사용 시, Load Balancer GUI에서 예상하지 못한 작동이 발생할 수 있습니다. 마우스를 클릭하면 마우스 포인터보다 약간 큰 영역의 블록이 손상되어 역으로 강조표시되거나 이미지가 화면 밖으로 이동하는 원인이 될 수 있습니다. 이전 Matrox 카드에서는 이러한 작동을 보이지 않습니다. Matrox AGP 카드 사용 시 알려진 수정 방법은 없습니다.

문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐

원격 웹 관리를 사용하여 Load Balancer를 구성할 경우에는 Load Balancer GUI가 나타나는 Netscape 브라우저 창의 크기를 조정하지 마십시오(최소화, 최대화, 복원 등). Netscape는 브라우저 창의 크기를 조정할 때마다 페이지를 재로드하므로 이로 인해 호스트로부터 연결이 끊어질 수 있습니다. 창의 크기를 조정할 때마다 호스트에 다시 연결해야 합니다. Windows 플랫폼에서 원격 웹 관리를 수행할 경우 Internet Explorer를 사용하십시오.

문제점: Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨

Windows 운영 체제의 명령 프롬프트 창에서, Latin-1 계열의 일부 자국 문자가 손상된 채로 표시될 수 있습니다. 예를 들어, 틸드가 있는 "a"가 파이 기호로 표시될 수 있습니다. 이를 수정하려면, 명령 프롬프트 창의 글꼴 등록 정보를 변경해야 합니다. 글꼴을 변경하려면, 다음을 수행하십시오.

1. 명령 프롬프트 창 상위 왼쪽 모서리에 있는 아이콘을 클릭하십시오.
2. 등록 정보를 선택한 후, 글꼴 탭을 클릭하십시오.
3. 기본 글꼴은 Raster 글꼴입니다. 이를 Lucida Console로 변경한 다음 확인을 클릭하십시오.

문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생

프로세스당 64 스레드만 허용하도록 일부 HP-UX 11i 설치가 사전 구성되었습니다. 그러나 일부 Load Balancer 구성에서는 이 양보다 더 요구합니다. HP-UX 시스템의 경우 프로세스당 스레드를 최소 256으로 설정하십시오. 이 값을 늘리려면 "sam" 유틸리티를 사용하여 max_thread_proc 커널 매개변수를 설정하십시오. 많이 사용할 경우 max_thread_proc를 256 이상으로 늘려야 합니다.

max_thread_proc를 늘리려면 334 페이지의 단계를 참조하십시오.

문제점: Windows 시스템에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함

Load Balancer 시스템에 어댑터를 구성할 경우 어드바이저가 작동하려면 다음 두 설정이 맞는지 확인해야 합니다.

- 일반적으로 3Com 어댑터 카드에서 사용되는 task offload를 사용 불가능하게 하십시오.
- TCP/IP 필터링을 사용할 경우 IP 프로토콜에 대해 프로토콜 1(ICMP)을 사용 가능하게 하십시오. ICMP가 사용 불가능할 경우 백엔드 서버에 대한 ping 검사가 성공하지 않습니다.

이 설정 구성에 대한 지시사항은 335 페이지를 참조하십시오.

문제점: Windows 시스템에서 어댑터에 둘 이상의 주소를 구성할 경우 호스트 이름에 대한 IP 주소를 확인하는 중에 문제가 발생함

Windows 플랫폼에서 두 이상의 IP 주소를 가진 어댑터를 구성할 경우 호스트 이름과 관련시킬 IP 주소를 먼저 레지스트리에 구성하십시오.

여러 경우(예: lbkeys 작성)에 Load Balancer가 InetAddress.getLocalHost()에 의존하므로, 단일 어댑터로 별명이 지정된 여러 IP 주소를 사용하면 문제가 생길 수 있습니다. 이 문제를 방지하기 위해 호스트 이름을 분석할 IP 주소를 먼저 레지스트리에 나열하십시오.

레지스트리에 호스트 이름을 먼저 구성하기 위한 단계는 335 페이지를 참조하십시오.

공통 문제점 해결—Site Selector

문제점: Site Selector가 실행되지 않음

이 문제점은 다른 응용프로그램이 Site Selector에서 사용하는 포트 중 하나를 사용하고 있을 때 발생할 수 있습니다. 자세한 내용은 320 페이지의 『Site Selector 포트 번호 확인』을 참조하십시오.

문제점: Site Selector가 Solaris 클라이언트로부터 통신을 라운드 로빙하지 않음

증상: Site Selector 컴포넌트가 Solaris 클라이언트로부터 수신하는 요청을 라운드 로빙하지 않습니다.

가능한 원인: Solaris 시스템에서 이름 서비스 캐시 디먼을 실행합니다. 디먼이 실행되면 Site Selector를 조회하는 대신, 캐시로부터 후속 분석기 요청을 응답합니다.

해결책: Solaris 시스템의 이름 서비스 캐시 디먼을 끄십시오.

문제점: sscontrol 또는 lbadmin 명령 실패

1. sscontrol 명령이 오류: 서버가 응답하지 않습니다를 리턴합니다. 또는 lbadmin 명령이 오류: RMI 서버를 액세스할 수 없습니다를 리턴합니다. 시스템에 소켓에 연결된 스택이 있을 때 이러한 오류가 발생할 수 있습니다. 이 문제점을 수정하려면 다음 행을 포함하도록 socks.cnf 파일을 편집하십시오.

```
EXCLUDE-MODULE java
EXCLUDE-MODULE javaw
```

2. Load Balancer 인터페이스의 관리 콘솔(명령행, 그래픽 사용자 인터페이스 및 마법사)은 RMI(원격 메소드 호출)를 사용하여 ssserver와 통신합니다. 기본 통신은 세 개의 포트를 사용하며, 각각의 포트가 ssserver 시작 스크립트에 다음과 같이 설정됩니다.

- 12099: sscontrol에서 명령을 받을 경우
- 10004: Metric Server에 메트릭 조회를 전송할 경우
- 12199: RMI 서버 포트의 경우
- DNS 통신량 전송 및 수신에 필요한 53

관리 콘솔 중 하나가 방화벽과 동일한 시스템에서 또는 방화벽을 통해 실행될 경우 이들 포트가 문제점을 야기할 수 있습니다. 예를 들어, []를 방화벽과 동일한 시스템에서 실행하고, `sscontrol` 명령을 발행하면 오류: 서버가 응답하지 않음과 같은 오류 메시지가 나타날 수 있습니다.

이러한 문제점을 방지하려면, `ssserver` 스크립트 파일을 편집하여 방화벽(또는 기타 응용프로그램)에 대하여 RMI에서 사용하는 포트를 설정하십시오.

`LB_RMISERVERPORT=10199` 행을 `LB_RMISERVERPORT=yourPort`로 변경하십시오. 여기서 *yourPort*는 다른 포트입니다.

완료되면, `ssserver`를 재시작하고 포트 12099, 10004, 12199 및 12100 또는 관리 콘솔이 실행될 호스트 주소에 대하여 선택된 포트와 통신을 시작하십시오.

3. 이러한 오류는 `ssserver`가 시작되지 않은 경우에도 발생할 수 있습니다.

문제점: `ssserver`가 Windows 플랫폼에서 시작에 실패함

Site Selector는 DNS에 참여할 수 있어야 합니다. 구성에 포함된 모든 시스템도 이 시스템에 참여해야 합니다. Windows 시스템에서는 구성된 호스트 이름이 항상 DNS에 있을 필요는 없습니다. Site Selector가 제대로 시작하려면 그 호스트 이름이 DNS에서 정의되어야 합니다.

이 호스트가 DNS에 정의되어 있는지 확인하십시오. `ssserver.cmd` 파일을 편집하고 "javaw"에서 "w"를 제거하십시오. 그러면 더 많은 오류에 대한 정보가 나타납니다.

문제점: Site Selector가 중복 라우트를 통해 올바르게 로드 밸런스하지 않음

Site Selector의 이름 서버가 시스템의 한 주소에 바인드되지 않습니다. 시스템에서 올바른 IP에 지정된 요청에 응답합니다. Site Selector는 운영 체제에 따라 클라이언트로 응답을 다시 라우트합니다. Site Selector 시스템에 여러 개의 어댑터가 있고 그 중 많은 어댑터가 동일한 서브넷에 연결되면, O/S가 받은 것과 다른 주소의 클라이언트에 응답을 전송할 수 있습니다. 일부 클라이언트 응용프로그램은 전송한 것과 다른 주소로부터 받은 응답을 승인하지 않습니다. 따라서 이름 분석은 실패하게 됩니다.

문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동

Windows 플랫폼에서 Matrox AGP 카드 사용 시, Load Balancer GUI에서 예상하지 못한 작동이 발생할 수 있습니다. 마우스를 클릭하면 마우스 포인터보다 약간 큰 영역의 블록이 손상되어 역으로 강조표시되거나 이미지가 화면 밖으로 이동하는 원인이 될 수 있습니다. 이전 Matrox 카드에서는 이러한 작동을 보이지 않습니다. Matrox AGP 카드 사용 시 알려진 수정 방법은 없습니다.

문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐

원격 웹 관리를 사용하여 Load Balancer를 구성할 경우에는 Load Balancer GUI가 나타나는 Netscape 브라우저 창의 크기를 조정하지 마십시오(최소화, 최대화, 복원 등). Netscape는 브라우저 창의 크기를 조정할 때마다 페이지를 재로드하므로 이로 인해 호스트로부터 연결이 끊어질 수 있습니다. 창의 크기를 조정할 때마다 호스트에 다시 연결해야 합니다. Windows 플랫폼에서 원격 웹 관리를 수행할 경우 Internet Explorer를 사용하십시오.

문제점: Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨

Windows 운영 체제의 명령 프롬프트 창에서, Latin-1 계열의 일부 자국 문자가 손상된 채로 표시될 수 있습니다. 예를 들어, 틸드가 있는 "a"가 파이 기호로 표시될 수 있습니다. 이를 수정하려면, 명령 프롬프트 창의 글꼴 등록 정보를 변경해야 합니다. 글꼴을 변경하려면, 다음을 수행하십시오.

1. 명령 프롬프트 창 상위 왼쪽 모서리에 있는 아이콘을 클릭하십시오.
2. 등록 정보를 선택한 후, 글꼴 탭을 클릭하십시오.
3. 기본 글꼴은 Raster 글꼴입니다. 이를 Lucida Console로 변경한 다음 확인을 클릭하십시오.

문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생

프로세스당 64 스레드만 허용하도록 일부 HP-UX 11i 설치가 사전 구성되었습니다. 그러나 일부 Load Balancer 구성에서는 이 양보다 더 요구합니다. HP-UX 시스템의 경우 프로세스당 스레드를 최소 256으로 설정하십시오. 이 값을 늘리려면 "sam" 유틸리티를 사용하여 max_thread_proc 커널 매개변수를 설정하십시오. 많이 사용할 경우 max_thread_proc를 256 이상으로 늘려야 합니다.

max_thread_proc를 늘리려면 334 페이지의 단계를 참조하십시오.

문제점: Windows 시스템에서 어드바이저와 도달 목표는 모든 다운된 서버를 표시함

Load Balancer 시스템에 어댑터를 구성할 경우 어드바이저가 작동하려면 다음 두 설정이 맞는지 확인해야 합니다.

- 일반적으로 3Com 어댑터 카드에서 사용되는 task offload를 사용 불가능하게 하십시오.
- TCP/IP 필터링을 사용할 경우 IP 프로토콜에 대해 프로토콜 1(ICMP)을 사용 가능하게 하십시오. ICMP가 사용 불가능할 경우 백엔드 서버에 대한 ping 검사가 성공하지 않습니다.

공통 문제점 해결—Cisco CSS Controller

문제점: ccoserver가 시작되지 않음

이 문제는 다른 응용프로그램이 Cisco CSS Controller의 ccoserver에서 사용하는 포트 중 하나를 사용하고 있을 때 발생할 수 있습니다. 자세한 내용은 321 페이지의 『Cisco CSS Controller 포트 번호 확인』을 참조하십시오.

문제점: ccocontrol 또는 lbadm 명령 실패

1. ccocontrol 명령이 오류: 서버가 응답하지 않음을 리턴합니다. 또는 lbadm 명령이 오류: RMI 서버를 액세스할 수 없음을 리턴합니다. 시스템에 소켓에 연결된 스택이 있을 때 이러한 오류가 발생할 수 있습니다. 이 문제점을 수정하려면 다음 행을 포함하도록 socks.cnf 파일을 편집하십시오.

```
EXCLUDE-MODULE java
EXCLUDE-MODULE javaw
```

2. Load Balancer 인터페이스의 관리 콘솔(명령행, 그래픽 사용자 인터페이스 및 마법사)은 RMI(원격 메소드 호출)를 사용하여 ccoserver와 통신합니다. 기본 통신은 세 개의 포트를 사용하며, 각각의 포트가 ccoserver 시작 스크립트에 다음과 같이 설정됩니다.
 - 13099: ccocontrol로부터 명령을 받을 경우
 - 10004: Metric Server에 메트릭 조회를 전송할 경우
 - 13199: RMI 서버 포트의 경우

관리 콘솔 중 하나가 방화벽과 동일한 시스템에서 또는 방화벽을 통해 실행될 경우 이들 포트가 문제점을 야기할 수 있습니다. 예를 들어, []를 방화벽과 동일한 시스템에서 실행하고, ccocontrol 명령을 발행하면 오류: 서버가 응답하지 않음과 같은 오류 메시지가 나타날 수 있습니다.

이러한 문제점을 방지하려면, ccoserver 스크립트 파일을 편집하여 방화벽(또는 기타 응용프로그램)에 대하여 RMI에서 사용하는 포트를 설정하십시오. CCO_RMISERVERPORT=14199 행을 CCO_RMISERVERPORT=*yourPort*로 변경하십시오. 여기서 *yourPort*는 다른 포트입니다.

완료되면, ccoserver를 재시작하고 포트 13099, 10004, 13199 및 13100 또는 관리 콘솔이 실행될 호스트 주소에 대하여 선택된 포트와 통신을 시작하십시오.

3. 이러한 오류는 ccoserver가 시작되지 않은 경우에도 발생할 수 있습니다.

문제점: 포트 13099에서 레지스트리를 작성할 수 없음

이 문제는 유효한 제품 사용권이 없을 때 발생할 수 있습니다. ccoserver를 시작하려고 할 때, 다음 메시지가 나타납니다.

사용권이 만기되었습니다. 현지 IBM
영업대표 또는 허가된 IBM 대리점에 문의하십시오.

이 문제를 해결하려면 다음을 수행하십시오.

1. ccoserver를 이미 시작한 경우 **ccoserver stop**을 입력하십시오.
2. 유효한 사용권을 **...ibm/edge/lb/servers/conf** 디렉토리로 복사하십시오.
3. **ccoserver**를 입력하여 서버를 시작하십시오.

문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동

Windows 플랫폼에서 Matrox AGP 카드 사용 시, Load Balancer GUI에서 예상하지 못한 작동이 발생할 수 있습니다. 마우스를 클릭하면 마우스 포인터보다 약간 큰 영역의 블록이 손상되어 역으로 강조표시되거나 이미지가 화면 밖으로 이동하는 원인이 될 수 있습니다. 이전 Matrox 카드에서는 이러한 작동을 보이지 않습니다. Matrox AGP 카드 사용 시 알려진 수정 방법은 없습니다.

문제점: 컨설턴트를 추가할 때 연결 오류를 받음

컨설턴트를 추가할 때 잘못된 구성으로 인해 연결 오류가 발생할 수 있습니다. 이 문제점을 수정하려면 다음을 수행하십시오.

- 지정된 주소 또는 공동체가 스위치에 구성된 값과 정확히 일치하는지 확인하십시오.
- 제어기와 스위치 간의 연결성이 사용 가능한지 확인하십시오.
- 공동체가 스위치에 대한 읽기/쓰기 권한이 있는지 확인하십시오. 제어기는 쓰기 액세스를 확인하기 위해 연결을 테스트할 때 ApSvcLoadEnable(SNMP) 변수를 사용 가능하게 하려 합니다.

문제점: 스위치의 가중치가 갱신되지 않음

이 문제점을 수정하려면 다음을 수행하십시오.

- 활성 연결 또는 연결 비율 메트릭을 사용 중인 경우 **ccocontrol service SWID:OCID:serviceIO report**를 발행하십시오. 스위치에서의 처리량 통신량에 따라 메트릭 값이 변경되는지 확인하십시오.
- 컨설턴트 로그의 로그 레벨을 늘리고 SNMP 제한시간 어커런스를 찾으십시오. 제한 시간이 발생할 경우 가능한 솔루션은 다음과 같습니다.
 - 스위치에서의 로드를 줄이십시오.
 - 스위치와 제어기 간의 네트워크 지연을 줄이십시오.
- 컨설턴트를 정지시킨 후 재시작하십시오.

문제점: Refresh 명령이 컨설턴트 구성을 갱신하지 않음

컨설턴트 로그 레벨을 늘리고 명령을 재시도하십시오. 다시 명령이 실패하면 로그에서 SNMP 제한시간 또는 기타 SNMP 통신 오류를 탐색하십시오.

문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐

원격 웹 관리를 사용하여 Load Balancer를 구성할 경우에는 Load Balancer GUI가 나타나는 Netscape 브라우저 창의 크기를 조정하지 마십시오(최소화, 최대화, 복원 등). Netscape는 브라우저 창의 크기를 조정할 때마다 페이지를 재로드하므로 이로 인해 호스트로부터 연결이 끊어질 수 있습니다. 창의 크기를 조정할 때마다 호스트에 다시 연결해야 합니다. Windows 플랫폼에서 원격 웹 관리를 수행할 경우 Internet Explorer를 사용하십시오.

문제점: Windows 플랫폼에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨

Windows 운영 체제의 명령 프롬프트 창에서, Latin-1 계열의 일부 자국 문자가 손상된 채로 표시될 수 있습니다. 예를 들어, 틸드가 있는 "a"가 파이 기호로 표시될 수 있습니다. 이를 수정하려면, 명령 프롬프트 창의 글꼴 등록 정보를 변경해야 합니다. 글꼴을 변경하려면, 다음을 수행하십시오.

1. 명령 프롬프트 창 상위 왼쪽 모서리에 있는 아이콘을 클릭하십시오.
2. 등록 정보를 선택한 후, 글꼴 탭을 클릭하십시오.
3. 기본 글꼴은 Raster 글꼴입니다. 이를 Lucida Console로 변경한 다음 확인을 클릭하십시오.

문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생

프로세스당 64 스레드만 허용하도록 일부 HP-UX 11i 설치가 사전 구성되었습니다. 그러나 일부 Load Balancer 구성에서는 이 양보다 더 요구합니다. HP-UX 시스템의 경우 프로세스당 스레드를 최소 256으로 설정하십시오. 이 값을 늘리려면 "sam" 유틸리티를 사용하여 max_thread_proc 커널 매개변수를 설정하십시오. 많이 사용할 경우 max_thread_proc를 256 이상으로 늘려야 합니다.

max_thread_proc를 늘리려면 334 페이지의 단계를 참조하십시오.

공통 문제점 해결—Nortel Alteon Controller

문제점: nalserver가 시작되지 않음

이 문제는 다른 응용프로그램이 Nortel Alteon Controller의 nalserver에서 사용하는 포트 중 하나를 사용하고 있을 때 발생할 수 있습니다. 자세한 내용은 322 페이지의 『Nortel Alteon Controller 포트 번호 확인』을 참조하십시오.

문제점: nalcontrol 또는 lbadmin 명령 실패

1. nalcontrol 명령이 오류: 서버가 응답하지 않습니다를 리턴합니다. 또는 lbadmin 명령이 오류: RMI 서버를 액세스할 수 없습니다를 리턴합니다. 시스템에 소켓에 연결된 스택이 있을 때 이러한 오류가 발생할 수 있습니다. 이 문제점을 수정하려면 다음 행을 포함하도록 socks.cnf 파일을 편집하십시오.

```
EXCLUDE-MODULE java
EXCLUDE-MODULE javaw
```

2. Load Balancer 인터페이스의 관리 콘솔(명령행, 그래픽 사용자 인터페이스 및 마법사)은 RMI(원격 메소드 호출)를 사용하여 nalserver와 통신합니다. 기본 통신은 세 개의 포트를 사용하며, 각각의 포트가 nalserver 시작 스크립트에 다음과 같이 설정됩니다.

- 14099: nalcontrol로부터 명령을 받을 경우
- 10004: Metric Server에 메트릭 조회를 전송할 경우
- 14199: RMI 서버 포트의 경우

관리 콘솔 중 하나가 방화벽과 동일한 시스템에서 또는 방화벽을 통해 실행될 경우 이들 포트가 문제점을 야기할 수 있습니다. 예를 들어, []를 방화벽과 동일한 시스템에서 실행하고, nalcontrol 명령을 발행하면 오류: 서버가 응답하지 않음과 같은 오류 메시지가 나타날 수 있습니다.

이러한 문제점을 방지하려면, nalserver 스크립트 파일을 편집하여 방화벽(또는 기타 응용프로그램)에 대하여 RMI에서 사용하는 포트를 설정하십시오. NAL_RMISERVERPORT=14199 행을 NAL_RMISERVERPORT=*yourPort*로 변경하십시오. 여기서 *yourPort*는 다른 포트입니다.

완료되면, nalserver를 재시작하고 포트 14099, 10004, 14199 및 14100 또는 관리 콘솔이 실행될 호스트 주소에 대하여 선택된 포트와 통신을 시작하십시오.

3. 이러한 오류는 **nalserver**가 시작되지 않은 경우에도 발생할 수 있습니다.

문제점: 포트 14099에서 레지스트리를 작성할 수 없음

이 문제는 유효한 제품 사용권이 없을 때 발생할 수 있습니다. nalserver를 시작하려고 할 때, 다음 메시지가 나타납니다.

사용권이 만기되었습니다. 현지 IBM
영업대표 또는 허가된 IBM 대리점에 문의하십시오.

이 문제를 해결하려면 다음을 수행하십시오.

1. nalserver를 이미 시작한 경우 **nalserver stop**을 입력하십시오.
2. 유효한 사용권을 **...ibm/edge/lb/servers/conf** 디렉토리로 복사하십시오.
3. **nalserver**를 입력하여 서버를 시작하십시오.

문제점: Windows 플랫폼에서, Matrox AGP 비디오 카드 사용 시 예상하지 못한 GUI 작동

Windows 플랫폼에서 Matrox AGP 카드 사용 시, Load Balancer GUI에서 예상하지 못한 작동이 발생할 수 있습니다. 마우스를 클릭하면 마우스 포인터보다 약간 큰 영역의 블록이 손상되어 역으로 강조표시되거나 이미지가 화면 밖으로 이동하는 원인이 될 수 있습니다. 이전 Matrox 카드에서는 이러한 작동을 보이지 않습니다. Matrox AGP 카드 사용 시 알려진 수정 방법은 없습니다.

문제점: 웹 관리를 사용하는 중에 Netscape 브라우저 창의 크기를 조정하면 호스트로부터 연결이 끊어짐

원격 웹 관리를 사용하여 Load Balancer를 구성할 경우에는 Load Balancer GUI가 나타나는 Netscape 브라우저 창의 크기를 조정하지 마십시오(최소화, 최대화, 복원 등). Netscape는 브라우저 창의 크기를 조정할 때마다 페이지를 재로드하므로 이로 인해 호스트로부터 연결이 끊어질 수 있습니다. 창의 크기를 조정할 때마다 호스트에 다시 연결해야 합니다. Windows 플랫폼에서 원격 웹 관리를 수행할 경우 Internet Explorer를 사용하십시오.

문제점: 컨설턴트를 추가할 때 연결 오류를 받음

컨설턴트를 추가할 때 잘못된 구성으로 인해 연결 오류가 발생할 수 있습니다. 이 문제점을 수정하려면 다음을 수행하십시오.

- 지정된 주소 또는 공동체가 스위치에 구성된 값과 정확히 일치하는지 확인하십시오.
- 제어기와 스위치 간의 연결성이 사용 가능한지 확인하십시오.
- 공동체가 스위치에 대한 읽기/쓰기 권한이 있는지 확인하십시오. 제어기는 쓰기 액세스를 확인하기 위해 연결을 테스트할 때 ApSvcLoadEnable(SNMP) 변수를 사용 가능하게 하려 합니다.

문제점: 스위치의 가중치가 갱신되지 않음

이 문제점을 수정하려면 다음을 수행하십시오.

- 활성 연결 또는 연결 비율 메트릭을 사용 중인 경우 ccocontrol service SWID:OCID:serviceIO report를 발행하십시오. 스위치에서의 처리량 통신량에 따라 메트릭 값이 변경되는지 확인하십시오.
- 컨설턴트 로그의 로그 레벨을 늘리고 SNMP 제한시간 어커런스를 찾으십시오. 제한 시간이 발생할 경우 가능한 솔루션은 다음과 같습니다.
 - 스위치에서의 로드를 줄이십시오.
 - 스위치와 제어기 간의 네트워크 지연을 줄이십시오.
- 컨설턴트를 정지시킨 후 재시작하십시오.

문제점: Refresh 명령이 컨설턴트 구성을 갱신하지 않음

컨설턴트 로그 레벨을 늘리고 명령을 재시도하십시오. 다시 명령이 실패하면 로그에서 SNMP 제한시간 또는 기타 SNMP 통신 오류를 탐색하십시오.

문제점: Windows 시스템에서 손상된 Latin-1 자국 문자가 명령 프롬프트 창에 표시됨

Windows 플랫폼 운영 체제의 명령 프롬프트 창에서, Latin-1 계열의 일부 자국 문자가 손상된 채로 표시될 수 있습니다. 예를 들어, tilde가 있는 "a"가 파이 기호로 표시될 수 있습니다. 이를 수정하려면, 명령 프롬프트 창의 글꼴 등록 정보를 변경해야 합니다. 글꼴을 변경하려면, 다음을 수행하십시오.

1. 명령 프롬프트 창 상위 왼쪽 모서리에 있는 아이콘을 클릭하십시오.
2. 등록 정보를 선택한 후, 글꼴 탭을 클릭하십시오.
3. 기본 글꼴은 Raster 글꼴입니다. 이를 Lucida Console로 변경한 다음 확인을 클릭하십시오.

문제점: HP-UX에서 Java 메모리 부족/스레드 오류 발생

프로세스당 64 스레드만 허용하도록 일부 HP-UX 11i 설치가 사전 구성되었습니다. 그러나 일부 Load Balancer 구성에서는 이 양보다 더 요구합니다. HP-UX 시스템의 경우 프로세스당 스레드를 최소 256으로 설정하십시오. 이 값을 늘리려면 "sam" 유틸리티를 사용하여 max_thread_proc 커널 매개변수를 설정하십시오. 많이 사용할 경우 max_thread_proc를 256 이상으로 늘려야 합니다.

max_thread_proc를 늘리려면 334 페이지의 단계를 참조하십시오.

공통 문제점 해결—Metric Server

문제점: .bat 또는 .cmd 사용자 메트릭 파일을 실행하는 Windows 플랫폼의 Metric Server IOException

Windows 플랫폼에서 실행 중인 Metric Servers에서 사용자 작성 메트릭에 대해 전체 메트릭 이름을 사용해야 합니다. 예를 들면, **usermetric** 대신 **usermetric.bat**을 지정해야 합니다. **usermetric** 이름은 명령행에서는 유효하지만 런타임 환경에서 실행할 경우 작동되지 않습니다. 전체 메트릭 이름을 사용하지 않으면 Metric Server IOException을 수신하게 됩니다. metricserver 명령 파일에서 LOG_LEVEL 변수를 3으로 설정한 후 로그 출력을 확인하십시오. 이 예제에서, 다음과 같은 예외가 나타납니다.

```
... java.io.IOException: CreateProcess: usermetric error=2
```

문제점: Metric Server가 Load Balancer 시스템에 로드를 보고하지 않음

Metric Server가 로드 정보를 Load Balancer에 보고하지 않는 데에는 여러 가지 이유가 있을 수 있습니다. 원인을 판별하려면 다음 사항을 확인하십시오.

- 키 파일이 Metric Server로 전송되었는지 확인하십시오.
- Metric Server 시스템의 호스트 이름이 로컬 이름 서버로 등록되었는지 확인하십시오.

metricserver 스크립트의 Java 등록 정보 `java.rmi.server.hostname`에 있는 호스트 이름을 지정하여 이 문제점을 해결할 수 있습니다.

- 더 높은 로그 레벨로 재시작 한 후 오류를 찾으십시오.
- Load Balancer 시스템에서, **dscontrol manager metric set** 명령을 사용하여 메트릭 모니터에 대한 로그 레벨을 늘리십시오. `MetricMonitor.log` 파일에서 오류를 찾으십시오.

문제점: Metric Server 로그에서 "에이전트에 액세스하려면 서명이 필요합니다."라고 보고합니다.

Metric Server 로그는 키 파일이 서버로 전송된 후 이 오류 메시지를 보고합니다.

키의 쌍이 손상되어 키 파일이 해당 키로 권한을 부여 받지 못한 경우, 이 오류가 로그됩니다. 이 문제를 해결하려면 다음을 수행하십시오.

- 2진 전송 방법을 사용하여 키 파일을 다시 FTP로 전송하십시오.
- 새 키를 작성한 후 재분배하십시오.

문제점: AIX 시스템에서 과부하된 상태로 Metric Server를 실행할 경우, **ps -vg** 명령 출력이 손상될 수도 있음

멀티프로세서 AIX 플랫폼(4.3.3, 32비트 5.1 또는 64비트 5.1)에서 과부하된 상태로 Metric Server를 실행할 경우, **ps -vg** 명령의 출력이 손상될 수도 있습니다. 예를 들어,

```
55742 - A 88:19 42 18014398509449680 6396 32768 22 36 2.8 1.0 java -Xms
```

ps 명령의 **SIZE** 및/또는 **RSS** 필드가 사용 중인 메모리에 대하여 초과된 양을 표시할 수도 있습니다.

이것은 알려진 AIX 커널 문제점입니다. `Apar IY33804`는 이러한 문제점을 수정합니다. <http://techsupport.services.ibm.com/server/fixes>의 AIX 지원 센터에서 수정사항을 다운로드하거나, 해당 지역 AIX 지원 담당자에게 문의하십시오.

문제점: 고가용성 Dispatcher에서 Site Selector 로드 밸런싱을 사용하여 2계층 구성으로 Metric Server를 구성함

2계층 Load Balancer 구성에서 Site Selector(첫 번째 계층)가 Dispatcher 고가용성 상대(두 번째 계층)의 쌍에서 로드 밸런싱될 경우, Metric Server 컴포넌트를 구성하기 위한 단계를 수행해야 합니다. 특별히 Metric Server에서 사용하기 위한 새 IP 주소에서 인식하도록 Metric Server를 구성해야 합니다. 두 개의 고가용성 Dispatcher 시스템에서 Metric Server는 활성화된 Dispatcher에서만 작동합니다.

이 설정을 제대로 구성하려면 다음 단계를 수행하십시오.

- 새 로컬 IP에서 인식하도록 Metric Server를 구성하십시오. 로컬 NFA 주소에서 응답하도록 두지 말아야 합니다. 구성 정보는 211 페이지의 『Metric Server』를 참조하십시오.
- Site Selector는 활성화된 Dispatcher와만 통신하므로 고가용성 이동 스크립트에서 Metric Server를 시작 및 중지해야 합니다. Metric Server를 올바르게 시작하거나 정지하려면 시스템에서 새로운 Metric Server 특정 IP 별명을 지정하십시오. goActive 스크립트에서는 Metric Server IP를 루프백에서 물리적 어댑터로 이동하고, goStandby 스크립트에서는 반대로 수행하도록 Metric Server IP 주소를 이동하는(클러스터 주소 이동과 유사함) 이동 스크립트를 수정하십시오. IP 주소를 이동한 후 goActive 스크립트에서 `metricserver` 명령을 실행하여 Metric Server를 시작해야 합니다. 대기 모드에서 Metric Server가 Site Selector와 대화하지 못하도록 goStandby 스크립트에서 `metricserver stop`을 실행해야 합니다.
- Windows 플랫폼에서 Metric Server 특정 IP 주소를 이동하는 것은 223 페이지의 『스크립트 사용』을 참조하십시오.
- goStandby 스크립트 변경사항에는 다음과 같이 운영에 관련된 지시사항도 포함됩니다.
 - **HP-UX, Linux 및 Solaris 시스템:** 클러스터 주소가 루프백으로 이동하는 goStandby 스크립트의 섹션에서 Metric Server 특정 IP를 루프백으로 이동하는 명령을 삽입하십시오. 그 다음, `metricserver stop` 명령을 삽입하여 Metric Server가 Site Selector에 응답하는 것을 중지하십시오.
 - **AIX 시스템:** 클러스터 주소가 루프백으로 이동하는 goStandby 스크립트의 섹션에서 Metric Server 특정 IP를 루프백으로 이동하는 명령을 삽입하십시오. 그 다음, 루프백 별명과 통신할 수 있도록 라우트를 추가하십시오. `route add metricserverIP 127.0.0.1` 명령을 실행하십시오. 그런 다음, `metricserver stop` 명령을 삽입하여 Metric Server가 Site Selector에 응답하는 것을 중지하십시오. Metric Server가 정지된 후 최종 단계는 루프백 라우트를 제거하는 것입니다. 더 이상 혼란을 방지하기 위해 `route delete metricserverIP`를 삽입하십시오.

예를 들어,

```

ifconfig en0 delete 9.27.23.61
ifconfig lo0 alias 9.27.23.61 netmask 255.255.255.0
route add 9.27.23.61 127.0.0.1
metricserver stop
# 최대 60초 또는 Metric Server가 정지될 때까지 휴면합니다
let loopcount=0
while [[ "$loopcount" -lt "60" && 'ps -ef |grep AgentStop|
    grep -c -v gr ep' -eq "1"]]
do
    sleep 1
    let loopcount=$loopcount+1
done
route delete 9.27.23.61

```

- **Windows** 시스템: 먼저 Metric Server 루프백 어댑터(다음 예제에서는 LAN 2라고 함)를 IP 주소를 가진 시스템에 설치하십시오. 사용하지 않은 사설 네트워크 유형의 주소(예: 10.1.1.1)를 추가하십시오. 루프백을 구성한 후 이동 스크립트를 변경하십시오. goStandby 스크립트에는 Metric Server IP를 Metric Server 루프백 어댑터로 이동하는 netsh 명령이 포함됩니다. 그런 다음, **metricserver stop** 명령을 실행하십시오.

예를 들어,

```

call netsh interface ip delete address "Local Area Connection" addr=9.27.23.61
call netsh interface ip add address "Local Area Connection 2" addr=9.27.2.3.61
    mask = 255.255.255.0
sleep 3
metricserver stop

```

문제점: 여러 개의 CPU가 있는 Solaris 시스템에서 실행 중인 스크립트가 원치 않는 콘솔 메시지를 생산

여러 개의 CPU가 있는 Solaris 시스템에서 실행 시에 metricserver, cpuload 및 memload 스크립트가 원치 않는 콘솔 메시지를 생산할 수 있습니다. 이 동작은 VMSTAT 시스템 명령을 사용하여 커널에서 CPU 및 메모리 통계를 집계하기 때문입니다. VMSTAT가 리턴하는 일부 메시지는 커널 상태가 변경되었음을 표시합니다. 스크립트가 해당 메시지를 처리할 수 없으므로 쉘에서 불필요한 콘솔 메시지가 야기됩니다.

이러한 콘솔 메시지의 예제는 다음과 같습니다.

```

/opt/ibm/edge/lb/ms/script/memload[29]: TOTAL=: syntax error
/opt/ibm/edge/lb/ms/script/memload[31]: LOAD=4*100/0: divide by zero
/opt/ibm/edge/lb/ms/script/memload[29]: TOTAL=659664+: more tokens expected

```

해당 메시지는 무시할 수 있습니다.

문제점: IPv6용 Load Balancer에서 Linux 시스템의 Metric Server로부터 값을 검색할 수 없음

Linux 플랫폼에서 실행 시 소스 IPv6 주소 선택사항의 호환이 불가능합니다. 따라서 Metric Monitor는 잘못된 소스 IP 주소를 통해 Metric Server와 통신하려 합니다.

Linux 시스템의 경우, 특수 라우트에 대한 IPv6 소스 주소 선택사항은 라우트의 네트워크 부분과 일치하게 구성된 마지막 주소로 기본값이 지정됩니다.

IPv6 클러스터가 구성된 마지막 인터페이스인 경우, 해당 인터페이스는 라우팅 테이블의 네트워크 부분과 일치하며 라우트 기본 소스 IP 주소로 사용됩니다. 해당 라우트가 Load Balancer와 Metric Server 사이에서 사용되는 경우, 두 노드 사이의 통신은 성립되지 않습니다.

Load Balancer 노드가 클러스터 주소를 소스 IP 주소로 사용하여 Metric Server와의 통신을 시도하기 때문에 통신이 성립되지 않습니다. Metric Server 노드의 루프백에 구성된 클러스터의 경우, Metric Server에서 온 응답은 루프백으로 가고 통신이 성립되지 않습니다.

해결방안:

Linux 노드가 특수 라우트로 사용하는 주소를 판별하고 메트릭 모니터와 Metric Server 사이의 RMI 통신에 사용되는 인터페이스를 판별하려면 다음의 명령을 발행하십시오.

```
ip -6 route get your_ipv6_route
```

예를 들어 이 명령을 발행할 때입니다.

```
ip -6 route get fec0::/64
```

다음에 리턴됩니다.

```
fec0:: via fec0:: dev eth0 src fec0::4 metric 0 cache mtu 1500 advmss 1383
```

fec0::4가 클러스터 주소인 경우 다른 인터페이스가 장치에 추가되어 클러스터가 기본 소스로 사용되는 것을 방지해야 하며, 그렇지 않은 경우 기존의 비클러스터 인터페이스가 제거되고 다시 추가될 수 있습니다.

예를 들어,

```
ip -6 addr add fec0::5/64 dev eth0
```

문제점: Metric Server 시작 후 메트릭 값이 -1을 리턴함

이 문제점은 클라이언트로 전송 중 무결성을 상실한 키 파일의 결과일 수 있습니다.

FTP를 사용하여 키 파일을 Load Balancer 시스템에서 백엔드 서버로 전송하는 경우, 바이너리 모드를 사용하여 FTP 서버에 키 파일을 put하거나 FTP 서버에서 키 파일을 get하도록 하십시오.

제 9 부 명령어 참조서

이 파트에서는 모든 Load Balancer 컴포넌트에 대한 명령어 참조서 정보를 제공합니다. 다음 장을 포함합니다.

- 367 페이지의 제 26 장 『구문 다이어그램 읽는 방법』
- 369 페이지의 제 27 장 『Dispatcher 및 CBR 명령어 참조서』
- 429 페이지의 제 28 장 『Site Selector 명령어 참조서』
- 457 페이지의 제 29 장 『Cisco CSS Controller 명령어 참조서』
- 477 페이지의 제 30 장 『Nortel Alteon Controller 명령어 참조서』

제 26 장 구문 다이어그램 읽는 방법

구문 도표는 사용자가 입력한 내용을 운영 체제에서 올바르게 해석할 수 있도록 명령을 지정하는 방법을 보여줍니다. 왼쪽에서 오른쪽으로, 위에서 아래로 수평선(기본 경로)을 따라 읽도록 하십시오.

기호 및 구두점

구문 도표에는 다음과 같은 기호가 사용됩니다.

기호 설명

▶▶ 명령 구문의 시작을 표시합니다.

◀◀ 명령 구문의 끝을 표시합니다.

구문 도표에 표시되는 콜론, 물음표, 빼기 부호와 같은 모든 구두점을 포함시켜야 합니다.

매개변수

구문 도표에는 다음 유형의 매개변수가 사용됩니다.

매개변수 설명

필수 필수 매개변수는 기본 경로에 표시됩니다.

선택 선택 매개변수는 기본 경로 아래에 표시됩니다.

매개변수는 키워드와 변수로 분류됩니다. 키워드는 소문자로 표시되며, 소문자로 입력할 수 있습니다. 예를 들면, 명령 이름이 키워드입니다. 변수는 이탤릭체로 표시되고, 사용자가 제공하는 이름이나 값을 나타냅니다.

구문 예제

다음 예에서는 `user` 명령이 키워드입니다. 필수 변수는 `user_id`이고, 선택 변수는 `password`입니다. 변수는 사용자 고유의 값으로 바꾸십시오.

▶▶—user—*user_id*—*password*—▶▶

필수 키워드: 필수 키워드와 변수는 기본 경로 선에 나타납니다.

▶▶—required_keyword—▶▶

필수 키워드와 값을 코딩해야 합니다.

스택에서 하나의 필수 항목 선택: 상호 배타적인 둘 이상의 필수 키워드 또는 변수가 있어 이를 선택해야 하는 경우, 이러한 키워드와 변수는 수직의 영숫자순으로 표시됩니다.

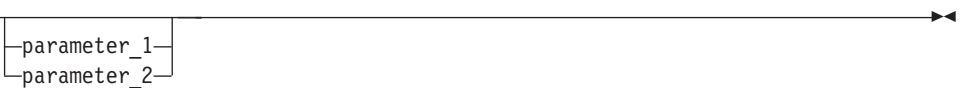
▶▶ 

선택 값: 선택 키워드와 변수는 기본 경로 선 아래에 나타납니다.

▶▶ 

선택 키워드와 변수는 코딩하지 않을 수도 있습니다.

스택에서 하나의 선택 키워드 선택: 상호 배타적인 둘 이상의 선택 키워드 또는 변수가 있어 이를 선택해야 하는 경우, 이러한 키워드와 변수는 기본 경로 선 아래에 수직의 영숫자순으로 표시됩니다.

▶▶ 

변수: 모두 이탤릭체로 되어 있는 단어는 변수입니다. 구문에 변수가 있으면, 이 변수를 텍스트에 정의된 대로 허용되는 이름이나 값 중 하나로 바꾸어야 합니다.

▶▶ 

영숫자가 아닌 문자: 도표에 영숫자가 아닌 문자(예: 콜론, 따옴표 또는 빼기 부호)가 표시되는 경우, 구문의 일부로 해당 문자를 코딩해야 합니다. 이 예제에서는 *cluster:port*를 코딩해야 합니다.

▶▶ 

제 27 장 Dispatcher 및 CBR 명령어 참조서

이 장에서는 Dispatcher **dscontrol** 명령 사용 방법을 설명합니다. 이 부록은 CBR에 대한 명령어 참조서이기도 합니다.

제품이 Network Dispatcher로 알려진 이전 버전의 경우, Dispatcher 제어 명령은 **ndcontrol**이었습니다. Dispatcher 제어 명령 이름은 **dscontrol**입니다. 이전 스크립트 파일들을 모두 갱신하여 Dispatcher 구성을 위해 **dscontrol(ndcontrol이 아님)**을 사용했는지 확인하십시오.

CBR은 이 명령 참조에 나열된 Dispatcher 명령의 서브세트를 사용합니다. **CBR**에 대해 구문 다이어그램을 사용할 경우, **dscontrol**을 **cbrcontrol**로 대체하십시오. 자세한 내용은 370 페이지의 『CBR 및 Dispatcher 간의 구성 차이점』을 참조하십시오.

중요: 이 제품의 IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, Dispatcher 컴포넌트만이 사용 가능합니다. 이 설치 유형의 Dispatcher는 이 명령 참조에 나열된 **dscontrol** 명령의 서브세트를 사용합니다. 해당 구문 다이어그램 사용 시, **dscontrol** 명령의 분리문자로 콜론(:) 대신 **at(@)**을 사용하십시오. 자세한 정보는 IPv4 및 IPv6용 Load Balancer 설치의 101 페이지의 『명령 구문 차이점』 및 101 페이지의 『지원되는 **dscontrol** 명령』을 참조하십시오.

다음 목록에는 이 장에서 참고한 명령이 들어 있습니다.

- 372 페이지의 『**dscontrol advisor** — 어드바이저 제어』
- 378 페이지의 『**dscontrol binlog** — 2진 로그 파일 제어』
- 379 페이지의 『**dscontrol cluster** — 클러스터 구성』
- 383 페이지의 『**dscontrol executor** — 실행 프로그램 제어』
- 388 페이지의 『**dscontrol file** — 구성 파일 관리』
- 390 페이지의 『**dscontrol help** — 이 명령의 도움말 표시 또는 인쇄』
- 391 페이지의 『**dscontrol highavailability** — 고가용성 제어』
- 395 페이지의 『**dscontrol host** — 원격 시스템 구성』
- 396 페이지의 『**dscontrol logstatus** — 서버 로그 설정 표시』
- 397 페이지의 『**dscontrol manager** — 관리자 제어』
- 403 페이지의 『**dscontrol metric** — 시스템 메트릭 구성』
- 404 페이지의 『**dscontrol port** — 포트 구성』
- 411 페이지의 『**dscontrol rule** — 규칙 구성』
- 418 페이지의 『**dscontrol server** — 서버 구성』

- 425 페이지의 『dscontrol set — 서버 로그 구성』
- 426 페이지의 『dscontrol status — 관리자 및 어드바이저가 실행 여부 표시』
- 427 페이지의 『dscontrol subagent — SNMP 서브에이전트 구성』

dscontrol 명령 매개변수의 최소 버전을 입력할 수 있습니다. 매개변수의 고유한 문자만 입력해야 합니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면 **dscontrol help file** 대신에 **dscontrol he f**를 입력할 수 있습니다.

명령 인터페이스를 시작하려면 **dscontrol**을 실행하여 dscontrol 명령 프롬프트를 수신하십시오.

명령 인터페이스를 종료하려면 **exit** 또는 **quit** 명령을 실행하십시오.

명령 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름(클러스터, 서버, 고가용성 명령에 사용)과 파일 이름(파일 명령에 사용)만 예외입니다.

CBR 및 Dispatcher 간의 구성 차이점

CBR 명령 인터페이스는 Dispatcher 명령 인터페이스의 서브세트입니다. CBR의 경우, 컴포넌트를 구성하려면 dscontrol 대신 **cbrcontrol**로 대체하십시오.

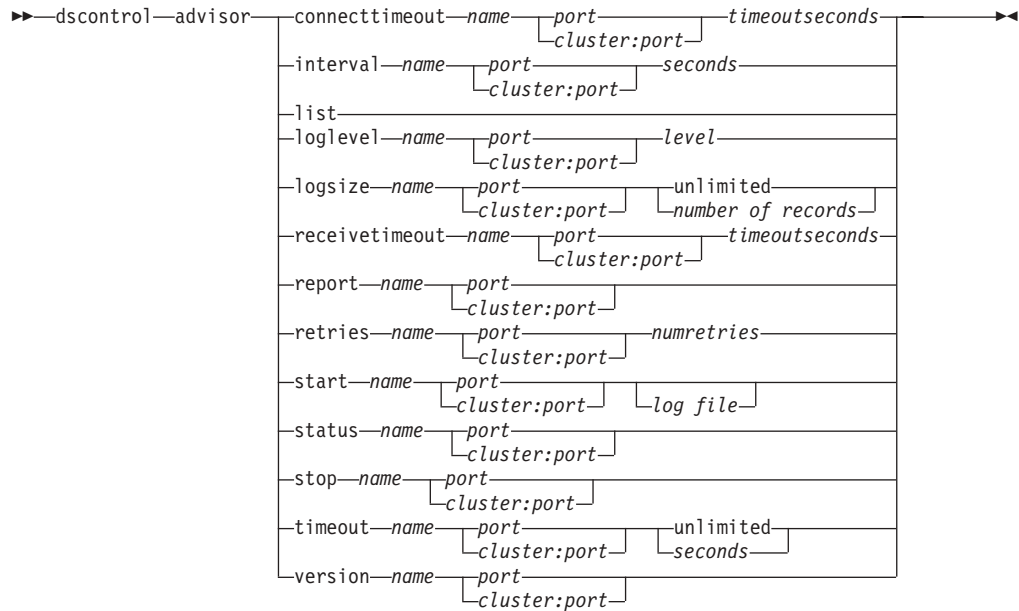
주: CBR(Content Based Routing) 컴포넌트는 64비트 JVM 실행 플랫폼을 제외한 모든 지원 플랫폼에서 사용 가능합니다. 혹은 Load Balancer에 있는 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 Caching Proxy를 사용하지 않고도 CBR을 제공할 수 있습니다. 자세한 정보는 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』 페이지를 참조하십시오.

아래는 CBR에 생략된 일부 명령 목록입니다.

1. 고가용성
2. subagent
3. 실행 프로그램
 - report
 - set nfa <value>
 - set fintimeout <value>
 - set hatimeout <value>
 - set hasynctimeout <value>
 - set porttype <value>
4. 클러스터
 - report {c}
 - set {c} porttype

5. port
 - add {c:p} porttype
 - add {c:p} protocol
 - set {c:p} porttype
6. rule add {c:p:r} type port
7. 서버
 - add {c:p:s} router
 - set {c:p:s} router

dscontrol advisor — 어드바이저 제어



connecttimeout

서버(서비스)의 특정 포트에 대한 서버와의 연결 실패를 보고하기 전에 어드바이저가 대기하는 시간을 설정하십시오. 자세한 정보는 202 페이지의 『어드바이저 연결 제한시간 및 서버의 수신 제한시간』을 참조하십시오.

name

어드바이저 이름. 가능한 값은 **connect**, **db2**, **dns**, **ftp**, **http**, **https**, **cachingproxy**, **imap**, **ldap**, **nntp**, **ping**, **pop3**, **self**, **sip**, **smtp**, **ssl**, **ssl2http**, **telnet** 및 **wlm** 입니다.

Load Balancer가 제공하는 어드바이저에 대한 자세한 정보는 203 페이지의 『어드바이저 목록』을 참조하십시오.

사용자 정의 어드바이저 이름은 **xxxx** 형식이며, 여기서 **ADV_xxxx**는 사용자 정의 어드바이저를 구현하는 클래스의 이름입니다. 207 페이지의 『사용자 정의(사용자 정의 가능) 어드바이저 작성』에서 자세한 정보를 참조하십시오.

port

어드바이저가 모니터링하는 포트의 번호.

cluster:port

클러스터 값은 어드바이저 명령에서 선택할 수 있지만 포트 값은 필수입니다. 클러스터 값을 지정하지 않으면 어드바이저가 모든 클러스터에 대해 포트에서 실행을 시작합니다. 클러스터를 지정하면 어드바이저가 포트에서 실행을 시작하지만 지정한 클러스터에 대해서만 실행됩니다. 200 페이지의 『어드바이저 시작 및 정지』에서 자세한 정보를 참조하십시오.

클러스터는 IP 주소 형식 또는 기호 이름입니다. 포트는 어드바이저가 모니터링하는 포트 번호입니다.

timeoutseconds

서버와의 연결 실패를 보고하기 전에 어드바이저가 대기하는 제한시간을 초 단위로 표시하는 양의 정수. 기본값은 어드바이저 간격에 지정된 값의 3배입니다.

interval

어드바이저가 정보에 대해 서버를 조회하는 간격을 설정합니다.

seconds

서버의 현재 상태를 서버에 요청하는 간격(초 단위)을 나타내는 양의 정수. 기본값은 7입니다.

list

현재 관리자에 정보를 제공하는 어드바이저 목록을 보여줍니다.

loglevel

어드바이저 로그에 대한 로그 레벨을 설정합니다.

level

레벨 번호(0-5). 기본값은 1입니다. 번호가 커질수록 더 자세한 정보가 어드바이저 로그에 기록됩니다. 가능한 값은 다음과 같습니다. 0은 없음, 1은 최소, 2는 기본, 3은 중간, 4는 고급, 5는 자세합니다.

logsize

어드바이저 로그의 최대 크기를 설정합니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서부터 기록된 후속 항목을 겹쳐줍니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목은 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 높은 레벨에서 기록될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

number of records

어드바이저 로그 파일의 최대 크기(바이트 단위). 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 파일은 로그 항목 그 자체의 크기가 변하므로, 겹쳐쓰기 전에는 그 정확한 최대 크기에 도달할 수 없습니다. 기본값은 1MB입니다.

receivetimeout

서버(서비스)의 특정 포트로부터의 수신 실패를 보고하기 전에 어드바이저가 대기하는 시간을 설정하십시오. 자세한 정보는 202 페이지의 『어드바이저 연결 제한시간 및 서버의 수신 제한시간』을 참조하십시오.

timeoutseconds

서버에서 수신 실패를 보고하기 전에 어드바이저가 대기하는 종료 시간을 초 단위로 표시하는 양의 정수. 기본값은 어드바이저 간격에 지정된 값의 3배입니다.

report

어드바이저 상태에 대한 보고서를 표시합니다.

retry

retry는 어드바이저가 서버를 종료된 서버로 표시하기 전에 수행할 수 있는 재시도 횟수를 설정합니다.

numretries

0 이상의 정수. 이 값은 3을 초과하면 안 됩니다. retry 키워드를 구성하지 않을 경우, 재시도 횟수는 기본적으로 0입니다.

start

어드바이저를 시작합니다. 각 프로토콜에 대해 어드바이저가 있습니다. 기본 포트는 다음과 같습니다.

어드바이저 이름	프로토콜	포트
cachingproxy	HTTP(Caching Proxy를 통한)	80
connect	ICMP	12345
db2	개인용	50000
dns	DNS	53
ftp	FTP	21
http	HTTP	80
https	SSL	443
imap	IMAP	143
ldap	LDAP	389
nntp	NNTP	119
ping	PING	0
pop3	POP3	110
self	개인용	12345
sip	SIP	5060
smtp	SMTP	25
ssl	SSL	443
ssl2http	SSL	443
telnet	Telnet	23
WLM	개인용	10,007

주: FTP 어드바이저는 FTP 제어 포트(21)에만 권고합니다. FTP 데이터 포트(20)에서는 FTP 어드바이저를 시작하지 마십시오.

log file

관리 데이터가 기록되는 파일 이름. 로그의 각 레코드에는 시간 소인이 표시됩니다.

기본 파일은 *advisername_port.log*이며, 예로는 **http_80.log**가 있습니다. 로그 파일이 보존되는 디렉토리를 변경하려면 288 페이지의 『로그 파일 경로 변경』을 참

조하십시오. 클러스터(또는 사이트)에 대한 고유 어드바이저의 기본 로그 파일은 클러스터 주소(예: **http_127.40.50.1_80.log**)를 사용하여 작성됩니다.

status

해당 기본값으로 글로벌하게 설정될 수 있는 어드바이저의 모든 값에 대한 현재 상태를 표시합니다.

stop

어드바이저를 정지합니다.

timeout

관리자에서 어드바이저의 정보를 유효한 것으로 간주하는 시간(초)을 설정합니다. 어드바이저 정보가 이 제한시간보다 이전 정보라는 것을 관리자에서 발견하면, 관리자는 어드바이저가 모니터링하는 포트에서 서버에 대한 가중치를 판별할 때 이 정보를 사용하지 않습니다. 이 제한시간에 대한 예외는 어드바이저가 관리자에 특정 서버가 단절되었다는 것을 알린 경우입니다. 관리자에서는 어드바이저 정보의 제한시간이 초과된 후에도 서버에 대한 해당 정보를 사용합니다.

seconds

시간(초)을 나타내는 양수 또는 단어 **unlimited**. 기본값은 unlimited입니다.

version

어드바이저의 현재 버전을 표시합니다.

예제

- 클러스터 127.40.50.1에 대해 포트 80에서 http 어드바이저를 시작하려면 다음 명령을 실행하십시오.
`dscontrol advisor start http 127.40.50.1:80`
- 모든 클러스터에 대해 포트 80에서 http 어드바이저를 시작하려면 다음 명령을 실행하십시오.
`dscontrol advisor start http 88`
- 클러스터 127.40.50.1에 대해 포트 80에서 http 어드바이저를 정지하려면 다음 명령을 실행하십시오.
`dscontrol advisor stop http 127.40.50.1:80`
- 서버와의 연결 실패를 보고하기 전에 포트 80에서 HTTP 어드바이저가 대기하는 시간(30초)을 설정하려면 다음 명령을 실행하십시오.
`dscontrol advisor connecttimeout http 80 30`
- 서버와의 연결 실패를 보고하기 전에 클러스터 127.40.50.1의 포트 80에서 HTTP 어드바이저가 대기하는 시간(20초)을 설정하려면 다음 명령을 실행하십시오.
`dscontrol advisor connecttimeout http 127.40.50.1:80 20`
- FTP 어드바이저 간격(포트 21의 경우)을 6초로 설정하려면 다음 명령을 실행하십시오.

```
dscontrol advisor interval ftp 21 6
```

- 현재 관리자에 정보를 제공하는 어드바이저 목록을 표시하려면 다음 명령을 실행하십시오.

```
dscontrol advisor list
```

이 명령으로 다음과 같은 출력이 작성됩니다.

ADVISOR	CLUSTER:PORT	TIMEOUT
http	127.40.50.1:80	unlimited
ftp	21	unlimited

- 더 나은 성능을 위해 어드바이저 로그의 로그 레벨을 0으로 설정하려면 다음 명령을 실행하십시오.

```
dscontrol advisor loglevel http 80 0
```

- 포트 21의 ftp 어드바이저 로그 크기를 5000 바이트로 변경하려면 다음 명령을 실행하십시오.

```
dscontrol advisor logsize ftp 21 5000
```

- 서버에서 수신 실패를 보고하기 전에 HTTP 어드바이저(포트 80의 경우)가 대기하는 시간(60초)을 설정하려면 다음 명령을 실행하십시오.

```
dscontrol advisor receivetimeout http 80 60
```

- FTP 어드바이저 상태에 대한 보고서를 표시하려면(포트 21의 경우) 다음 명령을 실행하십시오.

```
dscontrol advisor report ftp 21
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Advisor Report:

```
-----
Advisor name ..... Ftp
Port number ..... 21

Cluster address ..... 9.67.131.18
Server address ..... 9.67.129.230
Load ..... 8

Cluster address ..... 9.67.131.18
Server address ..... 9.67.131.215
Load ..... -1
```

- 포트 80의 http 어드바이저와 연관된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

```
dscontrol advisor status http 80
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Advisor Status:

Interval (seconds) 7
Timeout (seconds) Unlimited
Connect timeout (seconds).....21
Receive timeout (seconds).....21
Advisor log filename Http_80.log
Log level 1
Maximum log size (bytes) Unlimited
Number of retries 0

- 포트 21의 ftp 어드바이저 정보에 대한 제한시간 값을 5초로 설정하려면 다음 명령을 실행하십시오.

dscontrol advisor timeout ftp 21 5

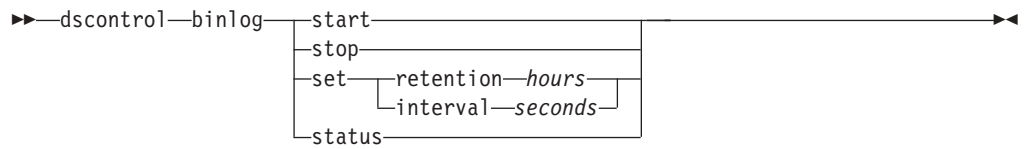
- 포트 443의 ssl 어드바이저의 현재 버전 번호를 표시하려면 다음 명령을 실행하십시오.

dscontrol advisor version ssl 443

이 명령으로 다음과 같은 출력이 작성됩니다.

Version: 04.00.00.00 - 07/12/2001-10:09:56-EDT

dscontrol binlog — 2진 로그 파일 제어



start

2진 로그를 시작합니다.

stop

2진 로그를 중지합니다.

set

2진 로그 필드를 설정합니다. 2진 로그 필드 설정에 대한 자세한 정보는 257 페이지의 『서버 통제를 분석하기 위해 2진 로그 사용』을 참조하십시오.

retention

2진 로그 파일이 보존되는 시간. **retention**의 기본값은 24입니다.

hours

시간.

interval

로그 입력 항목 간의 초 시간. 간격의 기본값은 60입니다.

seconds

초 시간.

status

2진 로그의 보유 및 간격을 표시합니다.

dscontrol cluster — 클러스터 구성



add

이 클러스터를 추가합니다. 최소한 하나의 클러스터를 정의해야 합니다.

cluster

클라이언트가 연결되는 클러스터 이름 또는 주소. 기호 이름 또는 IP 주소 형식으로 된 클러스터 값. 0.0.0.0의 클러스터 값을 사용하여 와일드 카드 클러스터를 지정할 수 있습니다. 253 페이지의 『와일드 카드 클러스터를 사용하여 서버 구성 조합』에서 자세한 정보를 참조하십시오.

dscontrol cluster add 명령을 제외한 콜론(:)을 와일드 카드로 사용할 수 있습니다. 예를 들어, dscontrol cluster set : weightbound 80 명령은 모든 클러스터에 대해 weightbound를 80으로 설정합니다.

주: 추가 클러스터는 더하기 부호(+)로 구분됩니다.

address

호스트 이름 또는 IP 주소 형식으로 된 TCP 시스템의 고유 IP 주소. 클러스터 값을 해석할 수 없는 경우, 실제 시스템의 해당 IP 주소를 제공해야 합니다.

주: 주소는 Dispatcher 컴포넌트에만 적용됩니다.

주소(address)

클러스터 주소의 값.

proportions

클러스터 레벨에서 관리자가 서버 가중치를 설정하는 데 사용되는 활성 연결(active), 새 연결(new), 어드바이저의 정보(port) 및 Metric Server와 같은 시스템 모니터링 프로그램의 정보(system)에 대해 중요도를 설정하십시오. 아래에 설명되어 있는 이러한 값 각각은 총계의 백분율로 표시되므로, 합은 항상 100입니다. 자세한 정보는 194 페이지의 『상태 정보에 제공되는 중요성 비율』을 참조하십시오.

active

활성 연결에 부여될 가중치를 표시하는 0 - 100의 숫자. 기본값은 50입니다.

new

새 연결에 부여될 가중치를 표시하는 0 - 100의 숫자. 기본값은 50입니다.

port

어드바이저의 정보에 부여될 가중치를 표시하는 0 - 100의 숫자. 기본값은 0입니다.

주: 어드바이저가 시작되고 포트 비율이 0이면, []는 자동으로 이 값을 1로 설정하여 관리자에서 서버 가중치를 계산하기 위한 입력으로 어드바이저 정보를 사용할 수 있도록 합니다.

system

Metric Server와 같은 시스템 메트릭 정보에 부여될 가중치를 표시하는 0 - 100의 숫자. 기본값은 0입니다.

maxports

최대 포트 수. maxports의 기본값은 8입니다.

size

허용되는 포트의 수.

maxservers

포트당 기본 최대 서버 수. 이것은 **port maxservers**를 사용하여 각각의 포트마다 대체할 수 있습니다. maxservers의 기본값은 32입니다.

size

포트에서 허용되는 서버의 수.

stickytime

작성되는 포트에 대한 기본 결합 시간. 이것은 **port stickytime**을 사용하여 각각의 포트마다 대체할 수 있습니다. 결합 시간의 기본값은 0입니다.

주: Dispatcher의 cbr 전달 메소드에서 포트가 SSL(HTTP가 아님)인 경우 stickytime(0이 아닌 값으로)을 설정하려면 포트 stickytime이 사용 가능해야 합니다. 포트가 작성될 stickytime이 0이 아닐 때 추가된 새 포트가 SSL이면 SSL ID 연관 관계가 포트에 사용 가능하게 됩니다. 포트에서 SSL ID 연관 관계를 사용하지 않으려면 포트 stickytime을 반드시 0으로 설정해야 합니다.

time

초 단위의 stickytime 값.

weightbound

바인드된 기본 포트 가중치. 이것은 **port weightbound**를 사용하여 각각의 포트마다 대체할 수 있습니다. weightbound의 기본값은 20입니다.

weight

weightbound의 값.

porttype

기본 포트 유형. 이것은 **port porttype**을 사용하여 각각의 포트마다 대체할 수 있습니다.

type

가능한 값은 **tcp**, **udp** 및 **both**입니다.

primaryhost

이 Dispatcher 시스템의 NFA 주소 또는 백업 Dispatcher 시스템의 NFA 주소. 상호 고가용성 구성에서 클러스터는 기본 또는 백업 시스템과 연관됩니다.

기본 및 백업이 이미 시작된 후에 상호 고가용성 기능을 실행하고 있을 때 클러스터의 기본 호스트를 변경하면 새로운 기본 호스트가 강제로 인계를 받게 해야 합니다. 또한 스크립트를 갱신하고 클러스터를 수동으로 구성 해제했다가 다시 구성해야 합니다. 67 페이지의 『상호 고가용성』에서 자세한 정보를 참조하십시오.

address

기본 호스트의 주소 값. 기본값은 이 시스템의 NFA 주소입니다.

staletimeout

연결이 제거되기 전에 연결에서 활동 해제 상태로 있을 수 있는 시간(초 단위). FTP의 기본값은 900이고, Telnet의 기본값은 32,000,000입니다. 모든 프로토콜에 대해 기본값은 300입니다. 이것은 **port staletimeout**을 사용하여 각각의 포트마다 대체할 수 있습니다. 288 페이지의 『활동해제 제한시간 값 사용』에서 자세한 정보를 참조하십시오.

staletimout

활동해제 제한시간 값

sharedbandwidth

클러스터 레벨에서 공유할 수 있는 최대 대역폭(초당 KB 단위). 공유 대역폭에 대한 자세한 정보는 230 페이지의 『예약된 대역폭 및 공유 대역폭에 따라 규칙 사용』 및 231 페이지의 『공유 대역폭 규칙』을 참조하십시오.

주: 공유 대역폭은 Dispatcher 컴포넌트에만 적용됩니다.

size

sharedbandwidth 크기는 정수 값입니다. 기본값은 0입니다. 값이 0이면, 클러스터 레벨에서 대역폭을 공유할 수 없습니다.

set

클러스터의 특성을 설정합니다.

remove

이 클러스터를 제거합니다.

report

클러스터의 내부 필드를 표시합니다.

주: 보고서는 Dispatcher 컴포넌트에 적용됩니다.

status

특정 클러스터의 현재 상태를 보여줍니다.

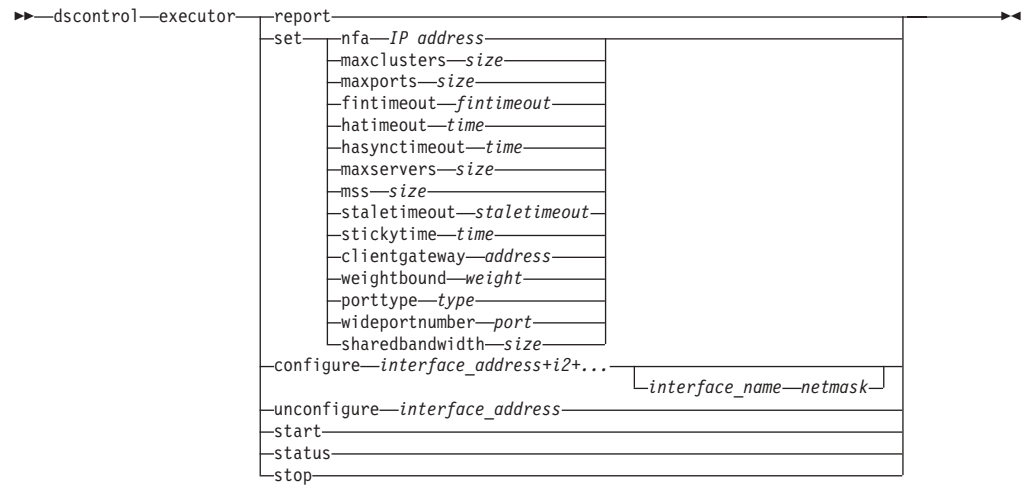
예제

- 클러스터 주소 130.40.52.153을 추가하려면 다음 명령을 실행하십시오.
`dscontrol cluster add 130.40.52.153`
- 클러스터 주소 130.40.52.153을 제거하려면 다음 명령을 실행하십시오.
`dscontrol cluster remove 130.40.52.153`
- 클러스터 9.6.54.12에 있는 서버에 대해 관리자가 수신한 입력(active, new, port, system)에 상대적 중요도를 설정하려면 다음 명령을 실행하십시오.
`dscontrol cluster set 9.6.54.12 proportions 60 35 5 0`
- 와일드 카드 클러스터를 추가하려면 다음 명령을 실행하십시오.
`dscontrol cluster add 0.0.0.0`
- 상호 고가용성 구성의 경우 백업 시스템의 NFA(9.65.70.19)를 기본 호스트로 사용하여 클러스터 주소 9.6.54.12를 설정하십시오.
`dscontrol cluster set 9.6.54.12 primaryhost 9.65.70.19`
- 클러스터 주소 9.67.131.167의 상태를 표시하려면 다음 명령을 실행하십시오.
`dscontrol cluster status 9.67.131.167`

이 명령으로 다음과 같은 출력이 작성됩니다.

```
Cluster Status:
-----
Cluster ..... 9.67.131.167
Address ..... 9.67.131.167
Number of target ports ..... 3
Default sticky time ..... 0
Default stale timeout ..... 30
Default port weight bound ..... 20
Maximum number of ports ..... 8
Default port protocol ..... tcp/udp
Default maximum number of servers ..... 32
Proportion given to active connections... 0.5
Proportion given to new connections..... 0.5
Proportion given specific to the port.... 0
Proportion given to system metrics..... 0
Shared bandwidth (KBytes) ..... 0
Primary Host Address ..... 9.67.131.167
```

dscontrol executor — 실행 프로그램 제어



report

통계 스냅샷 보고서를 표시합니다. 예를 들어, 수신된 패킷 총계, 버린 패킷 총계, 오류로 전송된 패킷 총계 등을 표시합니다.

주: 보고서는 Dispatcher 컴포넌트에 적용됩니다.

set

실행 프로그램의 필드를 설정합니다.

nfa

비전달 주소를 설정합니다. 이 주소로 전송된 모든 패킷이 Dispatcher 시스템에서 전달되지 않습니다.

주: NFA는 Dispatcher 컴포넌트에 적용됩니다.

IP 주소

기호 이름 또는 점분리 10진수 형식으로 된 인터넷 프로토콜 주소

maxclusters

구성될 수 있는 최대 클러스터 수. maxclusters의 기본값은 100입니다.

size

구성될 수 있는 최대 클러스터 수.

maxports

작성되는 클러스터의 maxports 기본값. 이것은 **cluster set** 또는 **cluster add** 명령으로 대체할 수 있습니다. maxports의 기본값은 8입니다.

size

포트의 수.

fintimeout

연결이 완료 상태로 된 후 메모리에 연결을 유지하는 초 시간. 기본 fintimeout 값은 60입니다.

fintimeout

fintimeout 값.

주: Fintimeout은 Dispatcher 컴포넌트에 적용됩니다.

hatimeout

고가용성 하트비트의 제한시간을 초과하기 위해 실행 프로그램에서 사용하는 초 수. 기본값은 2입니다.

주: hatimeout 값은 Dispatcher 컴포넌트에 적용됩니다.

time

hatimeout 값.

hasynctimeout

기본 시스템과 백업 시스템 간의 연결 레코드의 복제를 종료하기 위해 실행 프로그램에서 사용하는 초 수. 기본값은 50입니다.

기본 시스템 및 백업 시스템의 동기화 확인을 위해 타이머가 사용됩니다. 너무 많은 연결이 존재하고 활성 시스템이 대량의 수신 통신량 로드를 계속해서 처리하는 경우, 타이머의 시간이 끝나기 전에 동기화가 완료되지 않을 수도 있습니다. 이 경우 Load Balancer는 재동기화를 계속해서 시도하며 두 시스템은 동기화가 불가능하게 됩니다. 이러한 상황이 발생하면 hasynctimeout을 기본값보다 크게 설정하여 두 시스템이 충분한 시간을 갖고 기존의 연결에 대해 정보를 교환을 할 수 있도록 하십시오. 타이머를 설정하려면 hasynctimeout 명령이 고가용성 명령(dscontrol highavailability) 전에, dscontrol executor start 명령 후에 발행되어야 합니다.

주: hasynctimeout 값은 Dispatcher 컴포넌트에 적용됩니다.

time

hasynctimeout 값

maxservers

포트당 기본 최대 서버 수. 이것은 **cluster** 또는 **port** 명령으로 대체할 수 있습니다. maxservers의 기본값은 32입니다.

mss

TCP/UDP 연결 데이터 세그먼트의 최대 바이트 수입니다. 데이터 세그먼트 및 헤더의 바이트 수는 최대 전송 단위(MTU) 바이트 수보다 적은 값이 되어야 합니다. mss의 기본값은 1460입니다.

주: 최대 세그먼트 크기는 Dispatcher 컴포넌트의 nat 또는 cbr 전달 메소드에만 적용됩니다.

size

서버의 수.

staletimeout

연결이 제거되기 전에 연결에서 활동 해제 상태로 있을 수 있는 시간(초 단위). FTP의 기본값은 900이고, Telnet의 기본값은 32,000,000입니다. 기타 모든 포트에 대해 기본값은 300입니다. 이것은 **cluster** 또는 **port** 명령으로 대체할 수 있습니다. 288 페이지의 『활동해제 제한시간 값 사용』에서 자세한 정보를 참조하십시오.

staletimeout

활동해제 제한시간 값

stickytime

모든 추가될 클러스터의 기본 포트 결합 시간 값. 이것은 **cluster** 또는 **port** 명령으로 대체할 수 있습니다. 기본 stickytime 값은 0입니다.

time

초 단위의 stickytime 값.

clientgateway

clientgateway는 NAT/NAPT 또는 Dispatcher content-based routing에 사용되는 IP 주소입니다. 리턴 방향의 통신량이 Load Balancer에서 클라이언트로 전달되는 라우터 주소입니다. Clientgateway는 NAT/NAPT 또는 Dispatcher content-based routing 전달 메소드를 사용하여 포트를 추가하기 전에 0이 아닌 값으로 설정해야 합니다. 자세한 정보는 59 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 메소드)』 및 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』을 참조하십시오.

주: clientgateway는 Dispatcher 컴포넌트에만 적용됩니다.

address

기호 이름 또는 점분리 10진수 형식으로 된 clientgateway 주소. 기본값은 0.0.0.0입니다.

weightbound

모든 추가 포트의 기본 포트 가중치 범위 값. 이것은 **cluster** 또는 **port** 명령으로 대체할 수 있습니다. 기본 weightbound 값은 20입니다.

weight

weightbound 값.

porttype

모든 추가 포트의 기본 포트 유형 값. 이것은 **cluster** 또는 **port** 명령으로 대체할 수 있습니다.

주: Porttype는 Dispatcher 컴포넌트에만 적용됩니다.

type

가능한 값은 **tcp**, **udp** 및 **both**입니다.

wideportnumber

각 Dispatcher 시스템에서 사용되지 않는 TCP 포트. *wideportnumber*는 모든 Dispatcher 시스템에서 동일해야 합니다. *wideportnumber*의 기본값은 0이며, 이는 광역 지원이 사용되고 있지 않다는 것을 나타냅니다.

주: Wideportnumber는 Dispatcher 컴포넌트에만 적용됩니다.

port

wideportnumber의 값.

sharedbandwidth

실행 프로그램 레벨에서 공유할 수 있는 최대 대역폭(초당 KB 단위). 공유 대역폭에 대한 자세한 정보는 230 페이지의 『예약된 대역폭 및 공유 대역폭에 따라 규칙 사용』 및 231 페이지의 『공유 대역폭 규칙』을 참조하십시오.

주: 공유 대역폭은 Dispatcher 컴포넌트에만 적용됩니다.

size

sharedbandwidth 크기는 정수 값입니다. 기본값은 0입니다. 값이 0이면, 실행 프로그램 레벨에서 대역폭을 공유할 수 없습니다.

configure

Dispatcher 시스템의 네트워크 인터페이스 카드에 주소(예: 클러스터 주소, 리턴 주소 또는 고가용성 하트 비트 주소)를 구성합니다. 이는 또한 Dispatcher 시스템에서 별명을 구성하는 것으로도 알려져 있습니다.

주: 구성은 Dispatcher 컴포넌트에 적용됩니다.

interface_address

기호 이름 또는 IP 주소 형식으로 된 주소.

주: 추가 인터페이스는 더하기 부호(+)로 구분됩니다.

interface_name netmask

주소가 기존 주소의 어떤 서브넷과도 일치하지 않을 경우에만 필요합니다. *interface_name*은 en0, eth1, eri0과 같은 값이 될 수 있습니다. *netmask*는 IP 주소의 호스트 부분에서 서브네트워크 주소 비트를 식별하는 데 사용되는 32비트 마스크입니다.

unconfigure

네트워크 인터페이스 카드로부터 별명 주소를 삭제합니다.

주: 구성 해제는 Dispatcher 컴포넌트에 적용됩니다.

start

실행 프로그램을 시작합니다.

status

실행 프로그램에서 설정할 수 있는 값의 현재 상태와 기본값을 표시합니다.

stop

실행 프로그램을 중지합니다.

주: stop은 Dispatcher 및 CBR에 적용됩니다.

예제

- Dispatcher의 내부 카운터를 표시하려면 다음 명령을 실행하십시오.

```
dscontrol executor status
```

```
Executor Status:
```

```
-----
Nonforwarding address ..... 9.67.131.151
Client gateway address ..... 0.0.0.0
Fin timeout ..... 60
Wide area network port number ..... 0
Shared bandwidth (Kbytes) ..... 0
Default maximum ports per cluster ... 8
Maximum number of clusters ..... 100
Default maximum servers per port .... 32
Default stale timeout ..... 300
Default sticky time ..... 0
Default weight bound ..... 20
Default port type ..... tcp/udp
```

- 비전달 주소를 130.40.52.167로 설정하려면 다음 명령을 실행하십시오.

```
dscontrol executor set nfa 130.40.52.167
```

- 최대 클러스터 수를 설정하려면 다음 명령을 실행하십시오.

```
dscontrol executor set maxclusters 4096
```

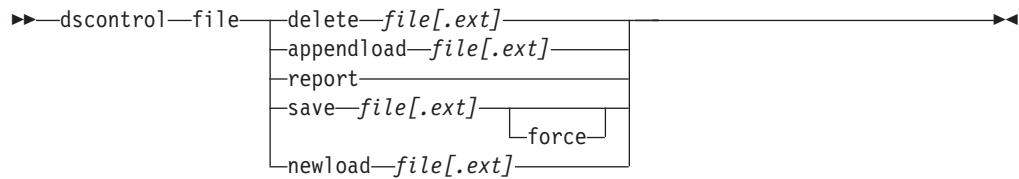
- 실행 프로그램을 시작하려면 다음 명령을 실행하십시오.

```
dscontrol executor start
```

- 실행 프로그램을 정지시키려면 다음을 수행하십시오.

```
dscontrol executor stop
```

dscontrol file — 구성 파일 관리



delete

파일을 삭제합니다.

file[.ext]

dscontrol 명령으로 구성된 구성 파일.

파일 확장자(.ext)는 사용자가 원하는 무엇이든 가능하며 생략할 수도 있습니다.

appendload

현재 구성을 갱신하기 위해 appendload 명령이 스크립트 파일에서 실행 가능 명령을 실행합니다.

report

사용 가능한 파일에 대해 보고합니다.

save

Load Balancer의 현재 구성 파일을 파일에 저장합니다.

주: 파일은 다음 디렉토리에 저장되고 로드됩니다. 여기서, 컴포넌트는 Dispatcher 또는 cbr입니다.

- Linux 및 UNIX 시스템: `/opt/ibm/edge/lb/servers/configurations/component`
- Windows 플랫폼: `C:\Program Files\ibm\edge\lb\servers\configurations\component`

force

파일을 기존 파일과 동일한 이름으로 저장하려면 새 파일을 저장하기 전에 **force**를 사용하여 기존 파일을 삭제하십시오. force 옵션을 사용하지 않으면 기존 파일에 겹쳐질 수 없습니다.

newload

Load Balancer로 새 구성 파일을 로드하여 실행합니다. 새 구성 파일은 현재 구성을 대신합니다.

예제

- 파일을 삭제하려면 다음 명령을 실행하십시오.

```
dscontrol file delete file3
```

```
File (file3) was deleted.
```

- 새 구성 파일을 로드하여 현재 구성을 대체하려면 다음 명령을 실행하십시오.

```
dscontrol file newload file1.sv
```

File (file1.sv) was loaded into the Dispatcher.

- 현재 구성에 구성 파일을 추가하고 로드하려면 다음 명령을 실행하십시오.

```
dscontrol file appendload file2.sv
```

File (file2.sv) was appended to the current configuration and loaded.

- 파일(즉, 이전에 사용자가 저장한 파일)의 보고서를 보려면 다음 명령을 실행하십시오.

```
dscontrol file report
```

FILE REPORT:

file1.save

file2.sv

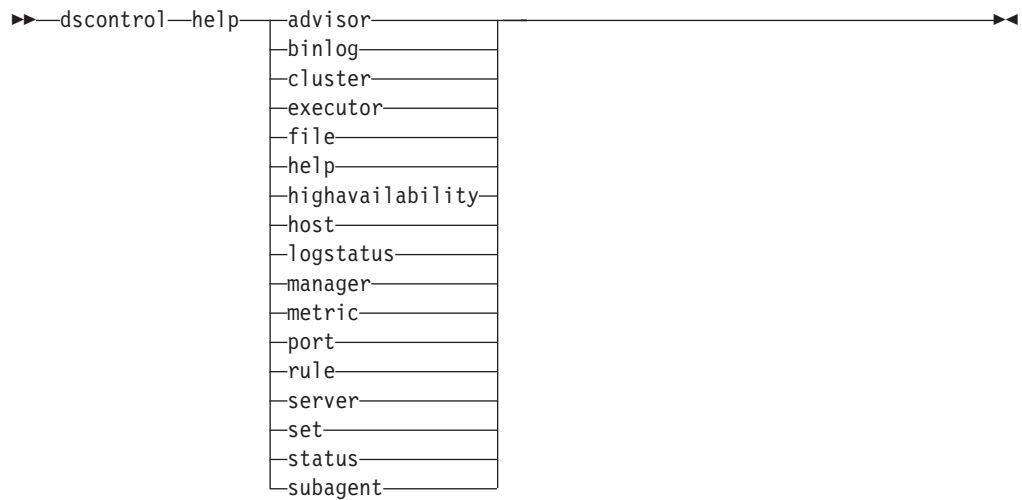
file3

- 구성을 file3 파일에 저장하려면 다음 명령을 실행하십시오.

```
dscontrol file save file3
```

The configuration was saved into file (file3).

dscontrol help — 이 명령의 도움말 표시 또는 인쇄



예제

- dscontrol 명령에 대한 도움말을 보려면 다음 명령을 실행하십시오.

```
dscontrol help
```

이 명령으로 다음과 같은 출력이 작성됩니다.

HELP COMMAND ARGUMENTS:

Usage: help <help option>

Example: help cluster

help	- print complete help text
advisor	- help on advisor command
cluster	- help on cluster command
executor	- help on executor command
file	- help on file command
host	- help on host command
binlog	- help on binary log command
manager	- help on manager command
metric	- help on metric command
port	- help on port command
rule	- help on rule command
server	- help on server command
set	- help on set command
status	- help on status command
logstatus	- help on server log status
subagent	- help on subagent command
highavailability	- help on high availability command

◇ 내의 매개변수가 변수라는 점에 유의하십시오.

- 때로 help는 |로 옵션을 구분하여 변수에 대한 선택사항을 보여줍니다.

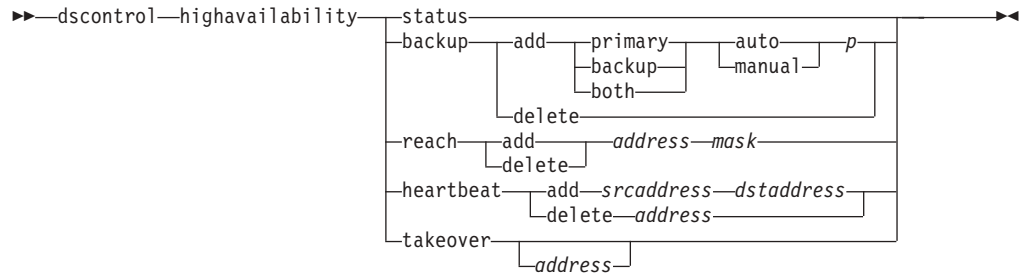
```
fintimeout <cluster address>|all <time>
```

-Change FIN timeout

(모든 클러스터를 변경할 경우 'all' 사용)

dscontrol highavailability — 고가용성 제어

주: dscontrol 고가용성 구문 다이어그램은 Dispatcher 컴포넌트에만 적용됩니다.



status

고가용성에 대한 보고서를 리턴합니다. 시스템은 다음 세 가지 상태 중 하나인 것으로 식별됩니다.

Active 제공된 시스템(기본, 백업 또는 둘다)에서 패킷의 경로를 지정하고 있습니다.

Standby

제공된 시스템(기본, 백업 또는 둘다)이 패킷의 경로를 지정하지 않고, 활성 Dispatcher의 상태를 모니터링합니다.

Idle 제공된 시스템이 패킷의 경로를 지정하고 있으며 그 상대 Dispatcher와의 연결을 설정하려고 하지 않습니다.

또한 **status** 키워드는 다양한 부속 상태에 대한 정보를 리턴합니다.

Synchronized

제공된 시스템이 다른 Dispatcher와의 연결을 설정했습니다.

기타 부속 상태

이 시스템이 그 상대 Dispatcher와의 연결을 설정하려고 하지만 아직 완료되지 않았습니다.

backup

기본 또는 백업 시스템에 대한 정보를 지정합니다.

add

이 시스템의 고가용성 기능을 정의하고 실행합니다.

primary

기본 역할을 하는 Dispatcher 시스템을 나타냅니다.

backup

백업 역할을 하는 Dispatcher 시스템을 나타냅니다.

both

기본 및 백업 역할을 둘다 하는 Dispatcher 시스템을 나타냅니다. 이것은 기본 및

백업 역할이 클러스터 세트 기준으로 연관되어 있는 상호 고가용성 기능에 해당합니다. 67 페이지의 『상호 고가용성』에서 자세한 정보를 참조하십시오.

auto

기본 시스템이 다시 서비스를 시작하자마자 패킷 경로지정을 재개하는 자동 복구 전략어를 지정합니다.

manual

기본 시스템이 관리자가 **takeover** 명령을 발행할 때까지 패킷 경로지정을 재개하지 않는 수동 복구 전략어를 지정합니다.

p[ort]

하트 비트에 대해 Dispatcher에서 사용할 두 시스템 모드에서 사용되지 않는 TCP 포트. *port*는 기본 및 백업 시스템 모두에 동일해야 합니다.

delete

백업이나 기본 시스템으로 더 이상 사용되지 않도록, 고가용성에서 이 시스템을 제거합니다.

reach

기본 및 백업 Dispatcher에 대한 목표 주소를 추가하거나 삭제합니다. 도달 어드바이저는 백업 및 기본 Dispatcher 둘 모두에서 *ping*을 전송하여 목표에 어느 정도 도달 가능한지 판별합니다.

주: 도달 목표를 구성할 때 도달 어드바이저도 시작해야 합니다. 관리자 기능에서 도달 어드바이저를 자동으로 시작합니다.

add

도달 어드바이저에 대한 대상 주소를 추가합니다.

delete

도달 어드바이저의 대상 주소를 제거합니다.

address

대상 노드의 IP 주소(IP 주소 형식 또는 기호)

mask

서브넷 마스크

heartbeat

기본 및 백업 Dispatcher 시스템 간의 통신 세션을 정의합니다.

add

소스 Dispatcher에 상대 주소(대상 주소)를 알립니다.

srcaddress

소스 주소. 해당 Dispatcher 시스템의 주소(IP 또는 기호).

dstaddress

대상 주소. 다른 Dispatcher 시스템의 주소(IP 또는 기호).

주: srcaddress 및 dstaddress는 적어도 하나 이상의 하트 비트 쌍에 대해 시스템의 NFA여야 합니다.

delete

하트 비트 정보에서 주소 쌍을 제거합니다. 하트 비트 쌍의 대상 주소 또는 출발지 주소를 지정할 수 있습니다.

address

대상 주소 또는 출발지 주소(IP 또는 기호).

takeover

단순 고가용성 구성(Dispatcher 시스템의 역할은 기본 또는 백업입니다).

- 인계는 대기 Dispatcher가 활성 상태가 되어 패킷 경로지정을 시작하도록 지시합니다. 이 기능은 현재 활성 상태의 Dispatcher가 강제로 대기 상태가 되게 합니다. 인계 명령은 대기 시스템에 발행해야 하며 수동 방식을 사용할 때만 사용할 수 있습니다. 하위 상태는 동기화됨이어야 합니다.

상호 고가용성 구성(각 Dispatcher 시스템의 역할은 둘다임):

- 상호 고가용성 기능을 가진 Dispatcher 시스템은 그 상대방과 일치하는 두 개의 클러스터를 포함합니다. 한 클러스터는 기본 클러스터(상대방의 백업 클러스터)로 고려되고 다른 클러스터는 백업 클러스터(상대방의 기본 클러스터)로 고려됩니다. 인계는 다른 시스템의 클러스터에 대한 패킷 경로 지정을 시작하도록 Dispatcher 시스템에 지시합니다. 이 인계 명령은 Dispatcher 시스템의 클러스터가 대기 상태에 있고 하위 상태가 동기화됨일 때만 발행할 수 있습니다. 이 명령은 상대방의 현재 활성 상태의 클러스터가 강제로 대기 상태로 변경되게 합니다. 인계 명령은 수동 방식을 사용할 때만 사용할 수 있습니다. 67 페이지의 『상호 고가용성』에서 자세한 정보를 참조하십시오.

주:

1. 시스템의 역할(기본, 백업, 둘다)은 변경되지 않습니다. 그 상대적인 상태(활성 또는 대기)만이 변경됩니다.
2. 세 가지의 가능한 takeover 스크립트인 goActive, goStandby 및 goInOp가 있습니다. 223 페이지의 『스크립트 사용』을 참조하십시오.

address

인계 주소 값은 선택적입니다. 이 값은 시스템의 역할이 기본 및 백업 둘다인 경우에만 사용해야 합니다(상호 고가용성 구성). 지정된 주소는 이 클러스터의 통신량을 정상적으로 라우트하는 Dispatcher 시스템의 NFA입니다. 두 클러스터 간에 인계가 발생하면 Dispatcher 자체의 NFA 주소를 지정하십시오.

예제

- 시스템의 고가용성 상태를 확인하려면 다음 명령을 실행하십시오.

```
dscontrol highavailability status
```

출력:

```
High Availability Status:
```

```
-----
```

```
Role .....primary
Recovery Strategy ..... manual
State ..... Active
Sub-state..... Synchronized
Primary host..... 9.67.131.151
Port .....12345
Preferred Target..... 9.67.134.223
```

```
Heartbeat Status:
```

```
-----
```

```
Count ..... 1
Source/destination ..... 9.67.131.151/9.67.134.223
```

```
Reachability Status:
```

```
-----
```

```
Count ..... 1
Address ..... 9.67.131.1 reachable
```

- 자동 복구 전략어와 포트 80을 사용하여 기본 시스템에 백업 정보를 추가하려면 다음 명령을 실행하십시오.

```
dscontrol highavailability backup add primary auto 80
```

- Dispatcher가 도달할 수 있어야 하는 주소를 추가하려면 다음 명령을 실행하십시오.

```
dscontrol highavailability reach add 9.67.125.18
```

- 기본 및 백업 시스템의 핵심 정보를 추가하려면 다음 명령을 실행하십시오.

```
Primary - highavailability heartbeat add 9.67.111.3 9.67.186.8
Backup - highavailability heartbeat add 9.67.186.8 9.67.111.3
```

- 대기 Dispatcher가 활성 상태가 되도록 지시하고 활성 시스템이 강제로 대기 상태가 되도록 하려면 다음 명령을 실행하십시오.

```
dscontrol highavailability takeover
```

dscontrol host — 원격 시스템 구성

▶▶—dscontrol—host:—remote_host—▶▶

remote_host

구성 중인 원격 Load Balancer 시스템의 이름이 명령을 입력할 때는 다음과 같이 **host:**와 *remote_host* 사이에 공백이 없어야 합니다.

dscontrol host:*remote_host*

명령 프롬프트에서 이 명령을 발생한 후에는 원격 Load Balancer 시스템에 대해 발행한 유효한 dscontrol 명령을 입력하십시오.

dscontrol logstatus — 서버 로그 설정 표시

▶—dscontrol—logstatus—▶

logstatus

서버 로그 설정(로그 파일 이름, 로그 레벨 및 로그 크기)을 표시합니다.

예제

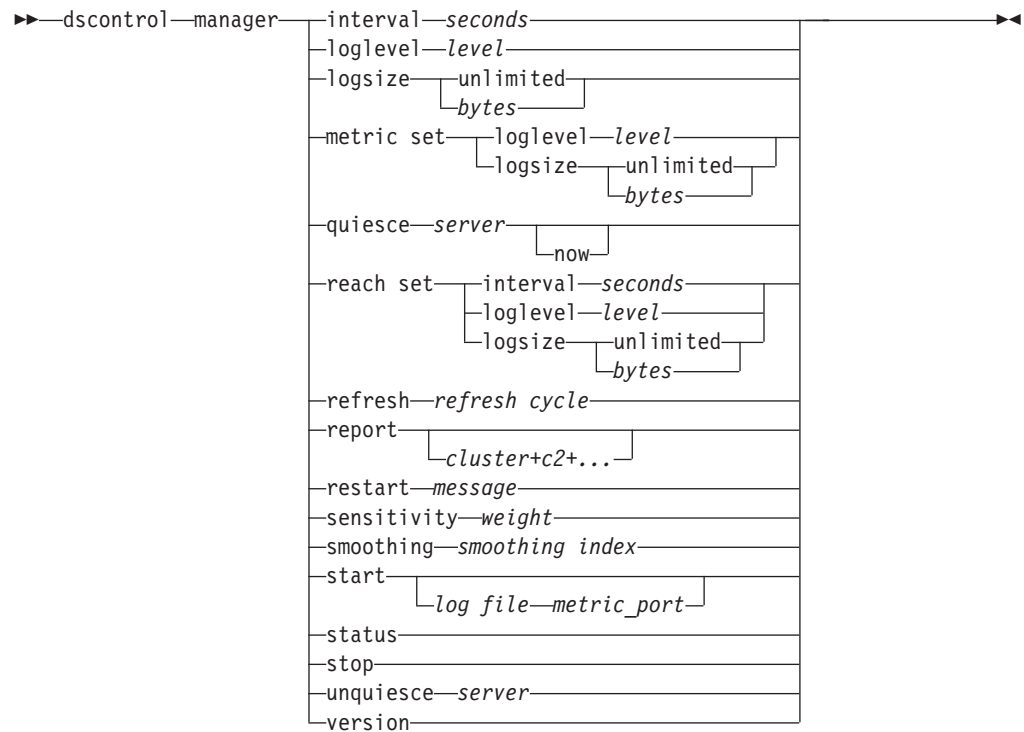
logstatus를 표시하려면 다음 명령을 실행하십시오.

```
dscontrol logstatus
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
Dispatcher Log Status:
-----
Log filename ..... C:\PROGRA~1\IBM\edge\lb\servers\logs\dispatcher
\server.log
Log level ..... 1
Maximum log size (bytes) ... 1048576
```

dscontrol manager — 관리자 제어



interval

관리자가 실행 프로그램에 대한 서버의 가중치를 갱신하는 간격을 설정하여, 실행 프로그램에서 클라이언트 요청을 라우트하는 데 사용하는 기준을 갱신합니다.

seconds

관리자가 실행 프로그램에 대한 가중치를 갱신하는 간격(초)을 나타내는 양수. 기본값은 2입니다.

loglevel

관리자 로그에 대한 로그 레벨을 설정합니다.

level

레벨 번호(0-5). 번호가 커질수록 더 자세한 정보가 관리자 로그에 기록됩니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다. 0은 없음, 1은 최소, 2는 기본, 3은 중간, 4는 고급, 5는 자세합니다.

logsize

관리자 로그의 최대 크기를 설정합니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에 서부터 기록된 후속 항목을 겹쳐씹니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목은 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 높은 레벨에서 기록될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

bytes

관리자 로그 파일의 최대 크기(바이트 단위). 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 파일은 로그 항목 그 자체의 크기가 변하므로, 겹쳐쓰기 전에는 그 정확한 최대 크기에 도달할 수 없습니다. 기본값은 1MB입니다.

metric set

메트릭 모니터 로그에 대한 로그 레벨 및 로그 크기를 설정합니다. 로그 레벨은 메트릭 모니터 로그 레벨입니다(0 - 없음, 1 - 최소, 2 - 기본, 3 - 중간, 4 - 고급 또는 5 - 자세히). 기본 로그 레벨은 1입니다. 로그 크기는 메트릭 모니터 로그 파일에 기록될 최대 바이트 수입니다. 0보다 큰 양의 정수 또는 **unlimited**를 지정할 수 있습니다. 기본 로그 크기는 1MB입니다.

quiesce

연결이 결합 시간으로 지정되고 결합 시간이 만기되지 않은 경우, 클라이언트에서 작업중지된 서버로의 후속 연결을 제외하고 서버로 전송될 연결이 더이상 없음을 지정합니다. 관리자는 서버가 정의되어 있는 모든 포트에 해당 서버의 가중치를 0으로 설정합니다. 서버에서 얼마간의 신속한 유지보수를 수행하기 위해 작업중지를 제한 경우, 이 명령을 사용하십시오. 구성에서 작업중지된 서버를 삭제한 후 다시 추가한 경우, 이전에 작업중지된 상태가 유지되지 않습니다. 자세한 정보는 239 페이지의 『서버 연결 처리 작업중지』를 참조하십시오

server

기호 이름이나 점분리 10진수 형식으로 된 서버의 IP 주소.

또는 서버 파티션을 사용할 경우, 논리 서버의 고유 이름. 64 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』에서 자세한 정보를 참조하십시오.

now

결합 시간을 설정하고 결합 시간이 만기되기 전에 다른 서버(작업중지되지 않은 서버)로 새 연결을 전송할 경우에만 “지금” 작업중지를 사용하십시오. 자세한 정보는 239 페이지의 『서버 연결 처리 작업중지』를 참조하십시오

reach set

도달 어드바이저에 대한 간격, 로그 레벨 및 로그 크기를 설정합니다.

refresh

새로운 연결과 활성 연결에 대한 정보를 갱신하기 위해 실행 프로그램을 조회하기 전의 간격 수를 설정합니다.

refresh cycle

간격 수를 나타내는 양수. 기본값은 2입니다.

report

통계 스냅샷 보고서를 표시합니다.

cluster

보고서에 표시하려는 클러스터의 주소. 주소는 기호 이름 또는 IP 주소 형식일 수 있습니다. 기본값은 모든 클러스터에 대한 관리자 보고서 화면입니다.

주: 추가 클러스터는 더하기 부호(+)로 구분됩니다.

restart

모든 서버(단절되지 않은)를 재시작하여 가중치를 표준화합니다(최대 가중치의 1/2).

message

관리자 로그 파일에 기록할 메시지.

sensitivity

가중치가 갱신되는 최소 감도를 설정합니다. 이 설정은 외부 정보에 따라 서버에 대한 가중치를 관리자에서 변경해야 할 시기를 정의합니다.

weight

가중치 백분율로 사용될 0에서 100 사이의 숫자. 기본값 5를 사용하면 최소 감도는 5%가 됩니다.

smoothing

로드 밸런스 시 가중치의 변화를 평탄화하는 지수를 설정합니다. 스무스 색인이 높으면 네트워크 상태가 변함에 따라 서버 가중치가 상대적으로 낮게 변경됩니다. 스무스 색인이 낮으면 서버 가중치가 상대적으로 높게 변경됩니다.

index

양의 부동 소수점 수. 기본값은 1.5입니다.

start

관리자를 시작합니다.

log file

관리자 데이터가 기록되는 파일 이름. 로그의 각 레코드에는 시간 소인이 표시됩니다.

기본 파일은 **logs** 디렉토리에 설치됩니다. 511 페이지의 부록 C 『예제 구성 파일』을 참조하십시오. 로그 파일이 보존되는 디렉토리를 변경하려면 288 페이지의 『로그 파일 경로 변경』을 참조하십시오.

metric_port

Metric Server가 시스템 로드를 보고하는 데 사용할 포트. 측정 기준 포트를 지정할 경우, 로그 파일 이름을 지정해야 합니다. 기본 측정 기준 포트는 10004입니다.

status

해당 기본값으로 글로벌하게 설정될 수 있는 관리자의 모든 값에 대한 현재 상태를 표시합니다.

stop

관리자를 정지합니다.

unquiesce

관리자가 정의된 모든 포트에서 이전에 작업중지된 서버에 0보다 높은 가중치 제공을 시작할 수 있도록 지정합니다.

server

기호 이름이나 점분리 10진수 형식으로 된 서버의 IP 주소.

version

관리자 현재 버전을 표시합니다.

예제

- 관리자의 갱신 간격을 매 5초로 설정하려면 다음 명령을 실행하십시오.
`dscontrol manager interval 5`
 - 더 나은 성능을 위해 로그 레벨을 0으로 설정하려면 다음 명령을 실행하십시오.
`dscontrol manager loglevel 0`
 - 관리자 로그 크기를 1,000,000 바이트를 설정하려면 다음 명령을 실행하십시오.
`dscontrol manager logsize 1000000`
 - 130.40.52.153에서 서버에 더 이상 연결을 전송하지 않도록 지정하려면 다음 명령을 실행하십시오.
`dscontrol manager quiesce 130.40.52.153`
 - 가중치를 새로 고치기 전의 갱신 간격 수를 3으로 설정하려면 다음 명령을 실행하십시오.
`dscontrol manager refresh 3`
 - 관리자의 통계 스냅샷을 확보하려면 다음 명령을 실행하십시오.
`dscontrol manager report`
- 이 명령으로 다음과 같은 출력이 작성됩니다.

SERVER	IP ADDRESS	STATUS
mach14.dmz.com	10.6.21.14	ACTIVE
mach15.dmz.com	10.6.21.15	ACTIVE

MANAGER REPORT LEGEND	
ACTV	Active Connections
NEWC	New Connections
SYS	System Metric
NOW	Current Weight
NEW	New Weight
WT	Weight
CONN	Connections

www.dmz.com						
10.6.21.100	WEIGHT	ACTV	NEWC	PORT	SYS	
PORT: 21	NOW NEW	49%	50%	1%	0%	
mach14.dmz.com	10 10	0	0	-1	0	
mach15.dmz.com	10 10	0	0	-1	0	

www.dmz.com						
10.6.21.100	WEIGHT	ACTV	NEWC	PORT	SYS	
PORT: 80	NOW NEW	49%	50%	1%	0%	
mach14.dmz.com	10 10	0	0	23	0	
mach15.dmz.com	9 9	0	0	30	0	

ADVISOR	CLUSTER:PORT	TIMEOUT
http	80	unlimited
ftp	21	unlimited

- 표준화된 가중치에 대해 모든 서버를 재시작하고 메시지를 관리자 로그 파일에 기록하려면 다음 명령을 실행하십시오.

```
dscontrol manager restart Restarting the manager to update code
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
320-14:04:54 Restarting the manager to update code
```

- 가중치 변경에 대한 감도를 10으로 설정하려면 다음 명령을 실행하십시오.

```
dscontrol manager sensitivity 10
```

- 스무스 색인을 2.0으로 설정하려면 다음 명령을 실행하십시오.

```
dscontrol manager smoothing 2.0
```

- 관리자를 시작하고 ndmgr.log(경로는 설정할 수 없음) 로그 파일을 지정하려면 다음 명령을 실행하십시오.

```
dscontrol manager start ndmgr.log
```

- 관리자와 관련된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

```
dscontrol manager status
```

이 명령으로 다음 예제와 같은 출력이 작성됩니다.

```
Manager status:
=====
Metric port..... 10004
Manager log filename..... manager.log
Manager log level..... 1
Maximum manager log size (bytes)..... unlimited
Sensitivity level..... 0.05
Smoothing index..... 1.5
Update interval (seconds)..... 2
Weights refresh cycle..... 2
Reach log level..... 1
Maximum reach log size (bytes)..... unlimited
Reach update interval (seconds)..... 7
Metric monitor log file name..... MetricMonitor.log
Metric monitor log level..... 1
Maximum metric monitor log size..... 1048576
```

- 관리자를 중지하려면 다음 명령을 실행하십시오.

```
dscontrol manager stop
```

- 130.40.52.153에서 서버로 더이상 새 연결을 전송하지 않도록 지정하려면 다음 명령을 실행하십시오. (주: 결합 시간을 설정하고 결합 시간이 만기되기 전에 다른 서버로 새 연결을 전송할 경우에만 서버 “지금” 작업중지를 사용하십시오.)

```
dscontrol manager quiesce 130.40.52.153 now
```

- 130.40.52.153에서 서버로 더이상 새 연결을 전송하지 않도록 지정하려면 다음 명령을 실행하십시오. (참고: 결합 시간을 설정한 경우 클라이언트에서 이후의 새 연결은 결합 시간이 만기될 때까지 이 서버로 전송됩니다.)

```
dscontrol manager quiesce 130.40.52.153
```

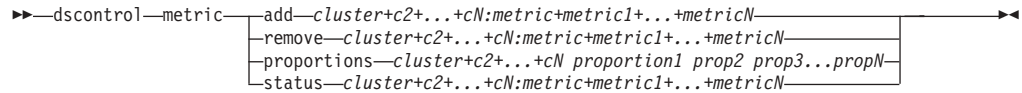
- 관리자에서 이전에 작업 거부된 130.40.52.153에 있는 서버에 0보다 높은 가중치 제공을 시작할 수 있도록 지정하려면 다음을 수행하십시오.

```
dscontrol manager unquiesce 130.40.52.153
```

- 관리자의 현재 버전 번호를 표시하려면 다음 명령을 실행하십시오.

```
dscontrol manager version
```


dscontrol metric — 시스템 메트릭 구성



add

지정된 메트릭을 추가합니다.

cluster

클라이언트가 연결되는 주소. 주소는 시스템의 호스트 이름 또는 IP 주소 표기법 형식일 수 있습니다. 추가 클러스터는 더하기 부호(+)로 구분됩니다.

metric

시스템 메트릭 이름. Metric Server의 스크립트 디렉토리에 있는 스크립트 파일 또는 실행 파일의 이름이어야 합니다.

remove

지정된 메트릭을 제거합니다.

proportions

이 오브젝트와 연관된 모든 메트릭의 비율을 설정하십시오.

status

이 메트릭의 현재 값을 표시합니다.

예제

- 시스템 메트릭을 추가하려면 다음 명령을 실행하십시오.

```
dscontrol metric add site1:metric1
```
- 두 개의 시스템 메트릭을 사용하여 사이트 이름에 비율을 설정하려면 다음 명령을 실행하십시오.

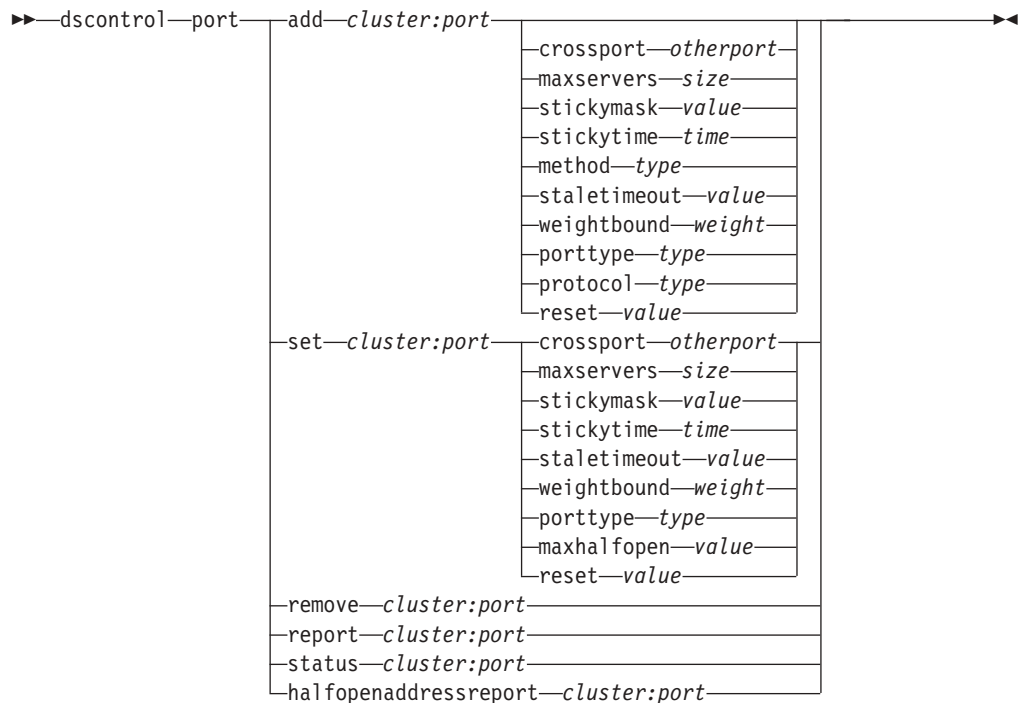
```
dscontrol metric proportions site1 0 100
```
- 지정된 메트릭과 연관된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

```
dscontrol metric status site1:metric1
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
Metric Status:
-----
Cluster ..... 10.10.10.20
Metric name ..... metric1
Metric proportion ..... 50
  Server ..... plm3
  Metric data ..... -1
```

dscontrol port — 포트 구성



add

클러스터에 포트를 추가합니다. 서버를 해당 포트에 추가하기 전에 클러스터에 포트를 추가해야 합니다. 클라이언트에 포트가 없는 경우, 모든 클라이언트 요청은 국지적으로 처리됩니다. 이 명령을 사용하면 한 번에 둘 이상의 포트를 추가할 수 있습니다.

cluster

기호 이름 또는 IP 주소 형식으로 된 클러스터의 주소. 콜론(:)을 와일드 카드로 사용할 수 있습니다. 예를 들어, `dscontrol port add :80` 명령은 모든 클러스터에 포트 80을 추가합니다.

주: 추가 클러스터는 더하기 부호(+)로 구분됩니다.

port

포트의 번호. 와일드 카드 포트를 지정하는 데 사용할 수 있는 포트 번호 값은 0입니다.

주: 추가 포트는 더하기 부호(+)로 구분됩니다.

crossport

Crossport를 사용하면 다른 포트에 수신된 클라이언트 요청이 여전히 후속 요청을 위해 동일한 서버로 전송될 수 있도록 여러 포트 사이로 결합/연관 관계 기능을 확장할 수 있습니다. crossport 값에 대해 포트간 연관 관계 기능을 공유하려는 otherport 수를 지정하십시오. 이 기능을 사용하려면 포트는 다음과 같아야 합니다.

- 동일한 클러스터 주소 공유
- 동일한 서버 공유
- 동일한(0이 아닌) stickytime 값 보유
- 동일한 stickymask 값 보유

포트간 연관 관계를 제거하려면 crossport 값을 다시 자신의 포트 번호로 설정하십시오. 포트간 연관 관계 기능에 대해서는 237 페이지의 『포트간 연관 관계』에서 자세한 정보를 참조하십시오.

주: Crossport는 Dispatcher 컴포넌트의 MAC 및 NAT/NATP 전달 메소드에만 적용됩니다.

otherport

crossport 값. 기본값은 자신의 포트 번호와 동일합니다.

maxservers

최대 서버 수. maxservers의 기본값은 32입니다.

size

maxservers의 값.

stickymask

연관 관계 주소 마스크 기능은 공통 서브넷 주소를 기반으로 수신 클라이언트 요청을 그룹화합니다. 클라이언트 요청이 먼저 포트로 연결되면 동일한 서브넷 주소(가려진 IP 주소의 해당 부분으로 지정됨)를 사용하는 클라이언트의 모든 후속 요청은 동일한 서버로 전송됩니다. stickymask를 사용하려면 port stickytime이 0이 아닌 값이어야 합니다. 238 페이지의 『연관 관계 주소 마스크(stickymask)』에서 자세한 정보를 참조하십시오.

주: stickymask 키워드는 Dispatcher 컴포넌트에만 적용됩니다.

value

stickymask 값은 사용자가 마스크하려는 32비트 IP 주소의 상위 순서 비트 번호입니다. 가능한 값은 8, 16, 24 및 32입니다. 기본값은 32이며, 연관 관계 주소 마스크 기능을 사용 불가능하게 합니다.

stickytime

한 연결의 닫기와 첫 번째 연결 중에 사용되는 동일한 서버로 다시 클라이언트가 전송될 새로운 연결이 열리는 사이의 간격. 결합 시간 이후에 클라이언트는 첫 번째 서버와 다른 서버로 전송될 수도 있습니다.

Dispatcher 컴포넌트의 경우

- Dispatcher cbr 전달 메소드
 - stickytime을 설정하면 SSL ID 연관 관계가 사용 가능하게 되므로 SSL(HTTP가 아님)에서만 stickytime(0이 아닌 값으로)을 설정할 수 있습니다.

- 포트 stickytime을 설정하면, 규칙에 연관 관계 유형이 없어야 합니다(기본값). stickytime이 포트에 설정될 경우 규칙 기반 연관 관계(수동 쿠키 및 URI)는 동시에 존재할 수 없습니다.
- Dispatcher mac 및 nat 전달 메소드
 - 포트 stickytime을(0이 아닌 값으로) 설정하면, 규칙에 연관 관계 유형을 설정할 수 없습니다. stickytime이 포트에 설정될 경우 규칙 기반 연관 관계는 동시에 존재할 수 없습니다.
 - 포트 stickytime 값을 설정하면 IP 주소 연관 관계를 사용할 수 있습니다.

CBR 컴포넌트의 경우: 포트 stickytime을 0이 아닌 값으로 설정하면, 규칙에 연관 관계 유형이 없어야 합니다(기본값). stickytime이 포트에 설정될 경우 규칙 기반 연관 관계(수동 쿠키, URI, 활성 쿠키)는 동시에 존재할 수 없습니다.

time

포트 결합 시간(초 단위). 0은 포트가 결합되지 않음을 의미합니다.

method

전달 메소드. 가능한 전달 메소드에는 mac 전달, nat 전달 또는 cbr(content-based routing) 전달이 있습니다. dscontrol executor 명령의 clientgateway 매개변수에서 0이 아닌 IP 주소를 먼저 지정하지 않으면, nat 또는 cbr 전달 메소드를 추가할 수 없습니다. 자세한 정보는 59 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 메소드)』 및 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』을 참조하십시오.

주:

1. 방법은 Dispatcher 컴포넌트에만 적용됩니다.
2. 백엔드 서버가 리턴 주소와 동일한 서브넷에 있고 cbr 전달 메소드 또는 nat 전달 메소드를 사용하는 경우, 라우터 주소를 백엔드 서버 주소로 정의해야 합니다.
3. mac 전달 메소드를 추가하면 "protocol" 매개변수를 HTTP 또는 SSL로 지정해야 합니다.

type

전달 메소드 유형. 가능한 값은 mac, nat, 또는 cbr입니다. 기본값은 mac 전달입니다.

staletimeout

연결이 제거되기 전에 연결에서 활동 해제 상태로 있을 수 있는 시간(초 단위). Dispatcher 컴포넌트의 경우 포트 21(FTP)에 대한 기본값은 900이며 포트 23(Telnet)에 대한 기본값은 32,000,000입니다. 기타 모든 Dispatcher 포트 및 모

든 CBR 포트의 경우, 기본값은 300입니다. 실행 프로그램 또는 클러스터 레벨에서 staletimeout을 설정할 수도 있습니다. 288 페이지의 『활동해제 제한시간 값 사용』에서 자세한 정보를 참조하십시오.

value

staletimeout의 값(초 단위).

weightbound

이 포트에서 서버의 최대 가중치를 설정합니다. 이것은 실행 프로그램이 각 서버에 제공하는 요청 수 사이의 차이에 영향을 줍니다. 기본값은 20입니다.

weight

최대 가중치 바운드를 표시하는 1 - 100의 숫자

porttype

포트 유형.

주: porttype은 Dispatcher에만 적용됩니다.

type

가능한 값은 **tcp**, **udp** 및 **both**입니다. 기본값은 both(tcp/udp)입니다.

protocol

프로토콜 유형. Dispatcher 컴포넌트에서 포트에 "cbr" 메소드를 지정할 경우 이는 필수 매개변수입니다. 포트 프로토콜 유형 **SSL**을 선택할 경우 **SSL ID** 연관 관계를 사용 가능하게 하려면 0이 아닌 stickytime도 지정해야 합니다. **HTTP** 프로토콜을 선택하면 "컨텐츠" 규칙을 사용하여 서버 연관 관계를 설정할 수 있습니다. 61 페이지의 『Dispatcher content-based routing(cbr 전달 메소드)』에서 자세한 정보를 참조하십시오.

주: 프로토콜은 Dispatcher의 cbr 전달 메소드에 적용됩니다.

type

가능한 값은 **HTTP** 또는 **SSL**입니다.

maxhalfopen

최대 반개방 연결 임계치. 서버에서 TCP 연결의 대다수가 반쯤 열려 있도록 하는 서비스 거부 중지를 감지하려면 이 매개변수를 사용하십시오.

양의 값은 현재 반개방 연결이 임계치를 초과하는지 여부를 판별하기 위한 검사가 수행됨을 표시합니다. 현재 값이 임계치를 초과하면 경보 스크립트가 호출됩니다. 255 페이지의 『서비스 거부 중지 감지』에서 자세한 정보를 참조하십시오.

주: maxhalfopen은 Dispatcher에만 적용됩니다.

value

maxhalfopen 값. 기본값은 0입니다(검사가 수행되지 않음).

reset

재설정을 사용하여 Load Balancer가 TCP 재설정을 포트의 다운된 서버로 전송할 것인지 여부를 지정할 수 있습니다. TCP 재설정은 연결을 즉시 닫습니다. 196 페이지의 『다운된 서버로 TCP 재설정 전송(Dispatcher 컴포넌트 전용)』에서 자세한 정보를 참조하십시오.

주: 재설정은 Dispatcher 컴포넌트에만 적용됩니다. reset 키워드를 사용하려면 dscontrol executor 명령의 clientgateway를 라우터 주소로 설정해야 합니다.

value

reset에 가능한 값은 yes 및 no입니다. 기본값은 no입니다. (다운된 서버에서 TCP 재설정이 수행되지 않습니다.) reset이 yes일 경우, TCP 재설정이 다운된 서버로 전송됩니다.

set

포트의 필드를 설정합니다.

remove

해당 포트를 제거합니다.

report

해당 포트에 대해 보고합니다.

status

해당 포트의 서버 상태를 보여줍니다. 모든 포트의 상태를 보려면, 이 명령에 port를 지정하지 마십시오. 그러나 콜론은 반드시 입력하십시오.

numSeconds

반개방 연결을 재설정하기 전의 시간(초 단위).

halfopenaddressreport

반개방 연결 상태의 서버에 액세스한 모든 클라이언트 주소(최대 8000개 주소 쌍)에 대해 로그(halfOpen.log)에서 항목을 생성합니다. 또한 반개방 연결 총계, 최대 및 평균, 반개방 연결 평균 시간(초 단위) 등의 통계 데이터가 명령행에 다시 보고됩니다. 255 페이지의 『서비스 거부 중지 감지』에서 자세한 정보를 참조하십시오.

예제

- 포트 80과 23을 클러스터 주소 130.40.52.153에 추가하려면 다음 명령을 실행하십시오.

```
dscontrol port add 130.40.52.153:80+23
```

- 와일드 카드 포트를 클러스터 주소 130.40.52.153에 추가하려면 다음 명령을 실행하십시오.

```
dscontrol port set 130.40.52.153:0
```

- 클러스터 주소 130.40.52.153에 있는 포트 80에 최대 가중치 10을 설정하려면 다음 명령을 실행하십시오.

```
dscontrol port set 130.40.52.153:80 weightbound 10
```

- 클러스터 주소 130.40.52.153에서 포트 80 및 포트 23에 대해 stickytime 값을 60 초로 설정하려면 다음 명령을 실행하십시오.

```
dscontrol port set 130.40.52.153:80+23 stickytime 60
```

- 클러스터 주소 130.40.52.153에서 포트 80에서 포트 23으로의 포트간 연관 관계를 설정하려면 다음 명령을 실행하십시오.

```
dscontrol port set 130.40.52.153:80 crossport 23
```

- 클러스터 주소 130.40.52.153에서 포트 23을 제거하려면 다음 명령을 실행하십시오.

```
dscontrol port remove 130.40.52.153:23
```

- 클러스터 주소 9.67.131.153에서 포트 80의 상태를 확보하려면 다음 명령을 실행하십시오.

```
dscontrol port status 9.67.131.153:80
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Port Status:

```
-----
Port number ..... 80
Cluster ..... 9.67.131.153
Stale timeout ..... 300
Weight bound ..... 20
Maximum number of servers ..... 32
Sticky time ..... 0
Port type ..... tcp/udp
Cross Port Affinity ..... 80
Sticky mask bits ..... 32
Max Half Open Connections ..... 0
Send TCP Resets ..... no
```

- 클러스터 주소 9.62.130.157에서 포트 80의 보고서를 확보하려면 다음 명령을 실행하십시오.

```
dscontrol port report 9.62.130.157:80
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Port Report:

```
-----
Cluster address ..... 9.62.130.157
Port number ..... 80
Number of servers ..... 5
Maximum server weight ..... 10
Total active connections ..... 55
Connections per second ..... 12
KBytes per second ..... 298
Number half open ..... 0
TCP Resets sent ..... 0
Forwarding method ..... MAC Based Forwarding
```

- 클러스터 주소 9.67.127.121에서 포트 80에 대한 반개방 주소 보고서를 얻으려면 다음 명령을 실행하십시오.

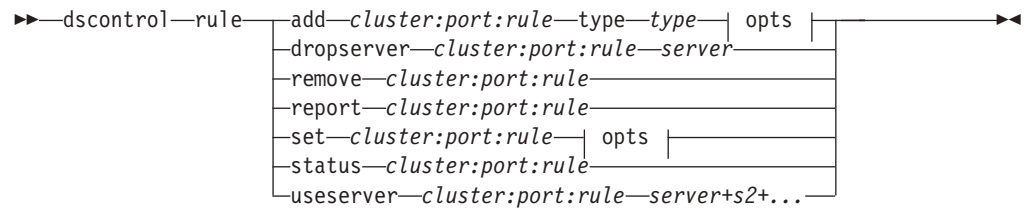
```
dscontrol port halfopenaddressreport 9.67.127.121:80
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Half open connection report successfully created:

```
Half Open Address Report for cluster:port = 9.67.127.121:80
Total addresses with half open connections reported ... 0
Total number of half open connections reported ..... 0
Largest number of half open connections reported ..... 0
Average number of half open connections reported ..... 0
Average half open connection time (seconds) reported .. 0
Total half open connections received ..... 0
```


dscontrol rule — 규칙 구성



opts:

beginrange—low—endrange—high
priority—level
pattern—pattern
tos—value
stickytime—time
affinity—affinity_type
cookie name—value
evaluate—level
sharelevel—level

add

규칙을 포트에 추가합니다.

cluster

기호 이름 또는 IP 주소 형식으로 된 클러스터의 주소. 콜론(:)을 와일드 카드로 사용할 수 있습니다. 예를 들어, `dscontrol rule add :80:RuleA type type` 명령은 모든 클러스터의 포트 80에 RuleA를 추가합니다.

주: 추가 클러스터는 더하기 부호(+)로 구분됩니다.

port

포트의 번호. 콜론(:)을 와일드 카드로 사용할 수 있습니다. 예를 들어, `dscontrol rule add clusterA::RuleA type type` 명령은 ClusterA의 모든 포트에 RuleA를 추가합니다.

주: 추가 포트는 더하기 부호(+)로 구분됩니다.

rule

사용자가 규칙에 대해 선택한 이름. 이 이름에는 영숫자, 밑줄, 하이픈 또는 마침표가 포함될 수 있습니다. 이 이름은 한 개부터 20개까지의 문자가 가능하며 공백이 포함될 수 없습니다.

주: 추가 규칙은 더하기 부호(+)로 구분됩니다.

type

규칙의 유형.

type

*type*의 선택사항은 다음과 같습니다.

ip 규칙이 클라이언트 IP 주소에 기초합니다.

time 규칙이 시간에 기초합니다.

connection

규칙이 포트의 초당 연결 수에 기초합니다. 이 규칙은 관리자가 실행 중인 경우에만 작동합니다.

active 규칙이 포트의 총 작동 중인 연결 수에 기초합니다. 이 규칙은 관리자가 실행 중인 경우에만 작동합니다.

port 규칙이 클라이언트 포트에 기초합니다.

주: 포트는 Dispatcher 컴포넌트에만 적용됩니다.

service

이 규칙은 IP 헤더의 서비스 유형(TOS) 바이트 필드에 기반을 둡니다.

주: Service는 Dispatcher 컴포넌트에만 적용됩니다.

reservedbandwidth

이 규칙은 서버 세트가 전달하는 대역폭(초당 KB 단위)을 기반으로 합니다. 자세한 정보는 230 페이지의 『예약된 대역폭 및 공유 대역폭에 따라 규칙 사용』 및 231 페이지의 『예약된 대역폭 규칙』을 참조하십시오.

주: reservedbandwidth는 Dispatcher 컴포넌트에만 적용됩니다.

sharedbandwidth

이 규칙은 실행 프로그램 또는 클러스터 레벨에서 공유되는 대역폭(초당 KB)을 기반으로 합니다. 자세한 정보는 230 페이지의 『예약된 대역폭 및 공유 대역폭에 따라 규칙 사용』 및 231 페이지의 『공유 대역폭 규칙』을 참조하십시오.

주: sharedbandwidth는 Dispatcher 컴포넌트에만 적용됩니다.

true 이 규칙은 항상 true입니다. 이는 프로그래밍 논리에서 else문으로 간주하십시오.

content

이 규칙은 요청된 URL과 비교될 일반 표현식을 설명합니다. 이 규칙은 Dispatcher 및 CBR에 대해 유효합니다.

beginrange

규칙이 올바른지 여부를 판별하는 데 사용되는 범위 내의 하위값.

low

규칙의 유형을 따릅니다. 값의 종류와 기본값이 규칙의 유형에 따라 다음에 나열되어 있습니다.

ip 기호 이름 또는 IP 주소 형식으로 된 클라이언트의 주소. 기본값은 0.0.0.0입니다.

time 정수. 기본값은 0이며 자정을 나타냅니다.

connection

정수. 기본값은 0입니다.

active 정수. 기본값은 0입니다.

port 정수. 기본값은 0입니다.

reservedbandwidth

정수(초당 KB). 기본값은 0입니다.

endrange

규칙이 올바른지 여부를 판별하는 데 사용되는 범위 내의 상위값.

high

규칙의 유형을 따릅니다. 값의 종류와 기본값이 규칙의 유형에 따라 다음에 나열되어 있습니다.

ip 기호 이름 또는 IP 주소 형식으로 된 클라이언트의 주소. 기본값은 255.255.255.254입니다.

time 정수. 기본값은 24이며 자정을 나타냅니다.

주: 시간 간격의 최소 범위 및 최대 범위를 정의할 때 각 값은 시간의 시 부분만을 나타내는 정수여야 한다는 점에 유의하십시오. 시 부분은 지정되지 않습니다. 이런 이유로 오전 3시와 4시 사이에서 하나의 시간을 지정하려면 최소 범위로 **3**을 지정하고 최대 범위도 **3**을 지정하십시오. 이는 3:00에서 시작하여 3:59에 끝나는 모든 시간(분 단위)를 지정합니다. 최소 범위를 **3**으로 지정하고 최대 범위를 **4**로 지정하면 3:00에서부터 4:59까지의 두 시간을 나타냅니다.

connections

정수. 기본값은 2의 32제곱에서 1을 뺀 값입니다.

active 정수. 기본값은 2의 32제곱에서 1을 뺀 값입니다.

port 정수. 기본값은 65535입니다.

reservedbandwidth

정수(초당 KB). 기본값은 2의 32제곱에서 1을 뺀 값입니다.

priority

규칙이 검토되는 순서

level

정수. 사용자가 추가하는 첫 번째 규칙의 우선순위를 지정하지 않으면, Dispatcher는 이를 기본값 1로 설정합니다. 후속 규칙이 추가되면, 기본적으로 그 규칙의 우선순

위는 기존 규칙의 현재 최하위 우선순위 + 10으로 계산됩니다. 예를 들어, 기존 규칙의 우선순위가 30이라고 가정합니다. 새로운 규칙을 추가하고, 그 우선순위를 25(이는 30보다 높은 우선순위임)로 설정합니다. 그런 다음, 우선순위를 설정하지 않고 세 번째 규칙을 추가합니다. 세 번째 규칙의 우선순위는 40(30 + 10)으로 계산됩니다.

pattern

컨텐츠 유형 규칙에 사용되는 패턴을 지정합니다.

pattern

사용되는 패턴. 유효한 값에 대한 자세한 정보는 507 페이지의 부록 B 『컨텐츠 규칙(패턴) 구문』을 참조하십시오.

tos

service 유형 규칙에 사용되는 “서비스 유형”(TOS) 값을 지정합니다.

주: TOS는 Dispatcher 컴포넌트에만 적용됩니다.

value

tos 값에 사용될 8자 문자열 여기서 유효한 문자는 0(2진 0, 1 (2진 1) 및 x(제한 없음)입니다. 예를 들면 0xx1010x와 같습니다. 자세한 내용은 229 페이지의 『서비스 유형(TOS)에 기반하여 규칙 사용』을 참조하십시오.

stickytime

규칙에 사용될 stickytime을 지정합니다. 규칙 명령에서 affinity 매개변수를 "activecookie"로 설정하는 경우, 해당 affinity 유형을 사용하려면 stickytime을 0이 아닌 값으로 설정하십시오. 규칙에 대한 stickytime은 "passivecookie" 또는 "uri" 연관 관계 규칙 유형에 적용되지 않습니다.

240 페이지의 『활성 쿠키 연관 관계』에서 자세한 정보를 참조하십시오.

주: 규칙 stickytime은 CBR 컴포넌트에 대해서만 적용됩니다.

time

시간(초 단위).

affinity

규칙에 사용할 연관 관계 유형(활성 쿠키, 수동 쿠키, URI 또는 없음)을 지정합니다.

"activecookie" 연관 관계 유형을 사용하면 Load Balancer가 생성한 쿠키를 기준으로 동일한 서버로 연관 관계를 사용하여 웹 통신량을 로드 밸런스할 수 있습니다.

"passivecookie" 연관 관계 유형을 사용하면 서버가 생성한 자체 식별 쿠키를 기준으로 동일한 서버로 연관 관계를 사용하여 웹 통신량을 로드 밸런스할 수 있습니다. 수동 쿠키 연관 관계와 함께 cookienam 매개변수를 사용해야 합니다.

"URI" 연관 관계 유형은 캐시 크기를 효율적으로 증대시키는 방법으로 Caching Proxy 서버로 웹 통신량을 로드 밸런싱할 수 있습니다.

자세한 정보는 240 페이지의 『활성 쿠키 연관 관계』, 242 페이지의 『수동 쿠키 연관 관계』 및 243 페이지의 『URI 연관 관계』를 참조하십시오.

주: 연관 관계는 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 구성된 규칙 및 CBR 컴포넌트에 적용됩니다.

affinity_type

가능한 연관 관계 유형 값은 없음(기본값), activecookie, passivecookie 또는 uri 입니다.

cookieName

Load Balancer ID의 역할을 하는 관리자가 임의로 설정한 이름. Load Balancer가 클라이언트 HTTP 헤더 요청에서 찾아야 하는 이름입니다. 쿠키 값과 함께 쿠키 이름은 Load Balancer ID 역할을 하므로 Load Balancer가 웹 사이트의 후속 요청을 동일한 서버 시스템으로 전송할 수 있습니다. 쿠키 이름은 "수동 쿠키" 연관 관계와 함께만 적용할 수 있습니다.

242 페이지의 『수동 쿠키 연관 관계』에서 자세한 정보를 참조하십시오.

주: 쿠키 이름은 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용하여 구성된 규칙 및 CBR 컴포넌트에 적용됩니다.

value

쿠키 이름 값.

evaluate

이 옵션은 Dispatcher 컴포넌트에서만 사용할 수 있습니다. 규칙 내의 서버 또는 포트 내의 모든 서버에 걸쳐 규칙의 조건을 평가할지 여부를 지정합니다. 이 옵션은 서버 특성을 기반으로 결정하는 규칙(예: connection, active 및 reservedbandwidth 규칙)에만 유효합니다. 자세한 내용은 235 페이지의 『규칙에 대한 서버 평가 옵션』을 참조하십시오.

연결 유형 규칙에 대해서는 upserversonrule 평가 옵션도 지정할 수 있습니다. upserversonrule을 지정하면 서버 세트 내의 서버 중 일부의 작동이 중지되더라도 규칙 내의 나머지 서버가 과부하되지 않음을 보장할 수 있습니다.

level

가능한 값은 포트, 규칙 또는 upserversonrule입니다. 기본값은 포트입니다. upserversonrule은 연결 유형 규칙에서만 사용할 수 있습니다.

sharelevel

이 매개변수는 공유 대역폭 규칙에만 해당합니다. 클러스터 레벨 또는 실행 프로그램 레벨에서 대역폭을 공유할지 여부를 지정합니다. 클러스터 레벨에서 대역폭을 공유하면 포트는 동일한 클러스터 내의 몇 개 포트에 걸쳐 최대 대역폭을 공유할 수

있습니다. 실행 프로그램 레벨에서 대역폭을 공유하면 전체 Dispatcher 구성 내의 클러스터는 최대 대역폭을 공유할 수 있습니다. 자세한 정보는 231 페이지의 『공유 대역폭 규칙』을 참조하십시오.

level

가능한 값은 실행 프로그램 또는 클러스터입니다.

dropserver

규칙 세트에서 서버를 제거합니다.

server

기호 이름 또는 IP 주소 형식으로 된 TCP 서버 시스템의 IP 주소.

또는 서버 파티션을 사용할 경우, 논리 서버의 고유 이름. 64 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』에서 자세한 정보를 참조하십시오.

주: 추가 서버는 더하기 부호(+)로 구분됩니다.

remove

더하기 부호로 구분된 하나 이상의 규칙을 제거합니다.

report

하나 이상의 규칙의 내부값을 표시합니다.

set

해당 규칙의 값을 설정합니다.

status

하나 이상의 규칙의 설정 가능한 값을 표시합니다.

useserver

서버를 규칙 세트에 삽입합니다.

예제

- 항상 참이 되는 규칙을 추가하려면 최소 범위 또는 최대 범위를 지정하지 마십시오.
`dscontrol rule add 9.37.67.100:80:trule type true priority 100`
- IP 주소 범위에 대한 액세스를 금지하는 규칙(이 경우에는 “9”로 시작함)을 작성하려면 다음 명령을 실행하십시오.
`dscontrol rule add 9.37.131.153:80:ni type ip b 9.0.0.0 e 9.255.255.255`
- 오전 11시에서 오후 3시까지 제공된 서버의 사용을 지정하는 규칙을 작성하려면 다음 명령을 실행하십시오.
`dscontrol rule add cluster1:80:timerule type time beginrange 11 endrange 14`
`dscontrol rule useserver cluster1:80:timerule server05`
- IP 헤더에서 TOS 바이트 필드의 콘텐츠에 기반하여 규칙을 작성하려면 다음 명령을 실행하십시오.
`dscontrol rule add 9.67.131.153:80:tosrule type service tos 0xx1001x`

- 최대 초당 100KB로 데이터를 전달하기 위해 일련의 서버(규칙 내에서 평가됨)를 할당할 예약된 대역폭을 기반으로 규칙을 작성하려면 다음 명령을 실행하십시오.

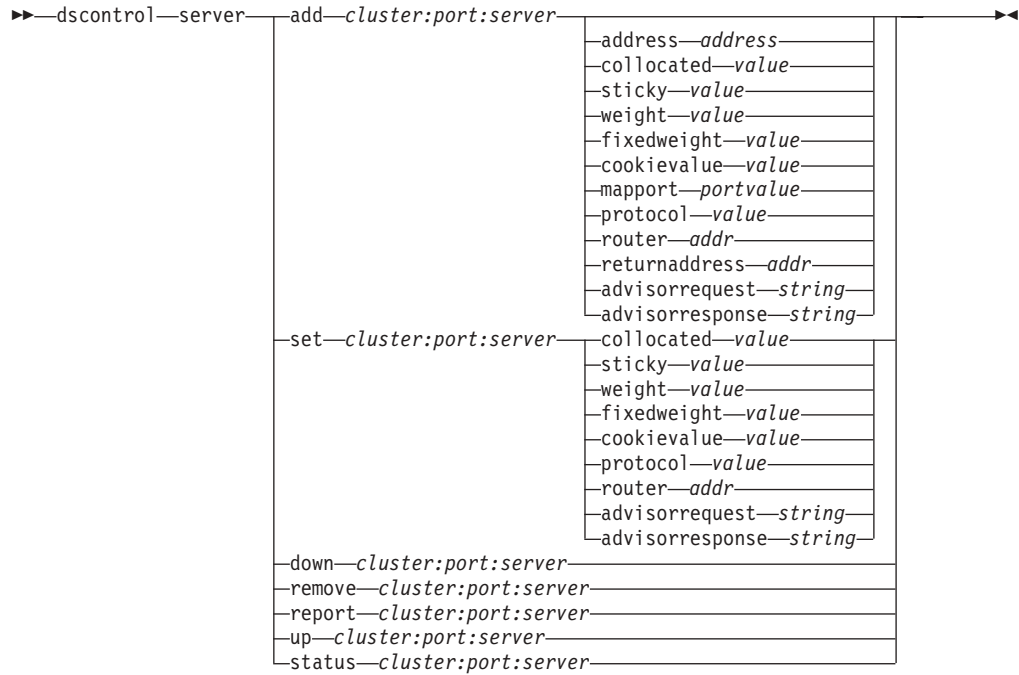
```
dscontrol rule add 9.67.131.153:80:rbwrule type reservedbandwidth
beginrange 0 endrange 100 evaluate rule
```

- 클러스터 레벨에서 사용하지 않은 대역폭을 보충할 공유 대역폭을 기반으로 규칙을 작성하려면 다음 명령을 실행하십시오(주: dscontrol cluster 명령을 사용하여 클러스터 레벨에서 공유할 수 있는 최대 대역폭(초당 KB)을 먼저 지정해야 함).

```
dscontrol cluster set 9.67.131.153 sharedbandwidth 200
```

```
dscontrol rule add 9.67.131.153:80:shbwrule type sharedbandwidth
sharelevel cluster
```

dscontrol server — 서버 구성



add

이 서버를 추가합니다.

cluster

기호 이름 또는 IP 주소 형식으로 된 클러스터의 주소. 콜론(:)을 와일드 카드로 사용할 수 있습니다. 예를 들어, `dscontrol server add :80:ServerA` 명령은 모든 클러스터의 포트 80에 ServerA를 추가합니다.

주: 추가 클러스터는 더하기 부호(+)로 구분됩니다.

port

포트의 번호. 콜론(:)을 와일드 카드로 사용할 수 있습니다. 예를 들어, `dscontrol server add ::ServerA` 명령은 모든 포트의 모든 클러스터에 ServerA를 추가합니다.

주: 추가 포트는 더하기 부호(+)로 구분됩니다.

server

server는 기호 이름 또는 IP 주소 형식으로 된 TCP 서버 시스템의 고유 IP 주소입니다.

IP 주소에 분석되지 않는 고유 이름을 사용하는 경우, **dscontrol server add** 명령에 서버 **address** 매개변수를 제공해야 합니다. 64 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』에서 자세한 정보를 참조하십시오.

주: 추가 서버는 더하기 부호(+)로 구분됩니다.

address

호스트 이름 또는 IP 주소 형식으로 된 TCP 서버 시스템의 고유 IP 주소 서버를 해석할 수 없는 경우, 실제 서버 시스템의 주소를 제공해야 합니다. 64 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』에서 자세한 정보를 참조하십시오.

address

서버의 주소 값.

collocated

collocated를 사용하면 Dispatcher가 로드 밸런싱하고 있는 서버 시스템 중 하나에 설치되었는지 여부를 지정할 수 있습니다.

주: collocated 매개변수는 Dispatcher의 mac, nat 또는 cbr 을 사용할 때 유효합니다. Site Selector 및 CBR을 모든 플랫폼에서 결합 배치할 수 있지만 이 키워드는 필요하지 않습니다. 자세한 내용은 216 페이지의 『결합 배치된 서버 사용』을 참조하십시오.

value

collocated 값: yes 또는 no. 기본값은 no입니다.

sticky

서버가 포트에 대한 연관 관계 설정을 무시할 수 있게 허용합니다. 기본값 “yes”를 사용하면 서버는 포트에 정의된 대로 표준 연관 관계를 유지합니다. “no” 값을 설정하면 클라이언트는 포트의 stickytime 설정에 관계 없이 해당 포트에 대해 요청을 발행하는 다음 번에 해당 서버로 리턴되지 않습니다. 이것은 규칙을 사용하고 있는 특정 상황에 유용합니다. 자세한 내용은 234 페이지의 『포트 연관 관계 무시』를 참조하십시오.

value

sticky 값: yes 또는 no. 기본값은 yes입니다.

weight

이 서버의 가중치를 표시하는 0 - 100의 숫자(지정된 포트의 가중치 바운드 값을 초과할 수 없음). 가중치를 0으로 설정하면 새로운 요청이 서버로 전송되지 않지만, 해당 서버에 대해 현재 활성 연결은 종료되지 않습니다. 기본값은 지정된 포트의 최대 가중치 바운드 값을 2로 나눈 값입니다. 관리자가 실행되고 있는 경우, 이 설정은 신속하게 겹쳐쓰입니다.

value

서버 가중치 값.

fixedweight

fixedweight 옵션을 사용하여 관리자가 서버 가중치를 수정할 수 있는지 여부를 지

정할 수 있습니다. `fixedweight` 값을 `yes`로 설정하면 관리자가 실행될 때 서버 가중치를 수정할 수 없습니다. 자세한 내용은 196 페이지의 『관리자 고정 가중치』를 참조하십시오.

value

`fixedweight` 값: `yes` 또는 `no`. 기본값은 `no`입니다.

cookievalue

`cookievalue`는 쿠키 이름/쿠키 값 쌍의 서버를 표시하는 임의 값입니다. 쿠키 이름과 함께 쿠키 값은 Load Balancer ID의 역할을 하여 Load Balancer가 후속 클라이언트 요청을 동일한 서버 시스템으로 전송할 수 있도록 합니다. 242 페이지의 『수동 쿠키 연관 관계』에서 자세한 정보를 참조하십시오.

주: `cookievalue`는 Dispatcher(`cbr` 전달 메소드 사용) 및 CBR에 대해 유효합니다.

value

임의 값입니다. 기본값은 쿠키 값 없음입니다.

mapport

클라이언트 요청의 대상 포트 번호(Dispatcher의 경우)를 Dispatcher가 클라이언트의 요청을 로드 밸런싱하는 데 사용하는 서버의 포트 번호에 대응시킵니다. Load Balancer가 하나의 포트에서 클라이언트의 요청을 수신하고 서버 시스템의 다른 포트에 이를 전송할 수 있도록 합니다. `mapport`를 사용하면 클라이언트의 요청을 다중 서버 디먼이 실행 중인 서버로 로드 밸런싱할 수 있습니다.

주: `mapport`는 Dispatcher(`nat` 또는 `cbr` 전달 메소드를 사용하는) 및 CBR에 적용됩니다. Dispatcher의 경우, 59 페이지의 『Dispatcher의 NAT/NAPT(`nat` 전달 메소드)』 및 61 페이지의 『Dispatcher content-based routing(`cbr` 전달 메소드)』을 참조하십시오. CBR의 경우, 120 페이지의 『SSL의 클라이언트-투-프록시 및 HTTP의 프록시-투-서버의 로드 밸런싱』를 참조하십시오.

protocol

프로토콜 유효값은 HTTP 및 HTTPS입니다. 기본값은 HTTP입니다.

주: 프로토콜은 CBR 컴포넌트에만 적용됩니다.

portvalue

맵 포트 번호 값. 기본값은 클라이언트 요청의 대상 포트 번호입니다.

router

광역 네트워크를 설정하고 있는 경우, 원격 서버에 대한 라우터 주소. 기본값은 0으로, 로컬 서버를 나타냅니다. 서버의 라우터 주소가 0(원격 서버를 나타냄)이 아닌 다른 값으로 설정되면, 다시 0으로 설정하여 로컬 서버로 만들 수 없습니다. 대신, 서버를 제거한 다음, 라우터 주소를 지정하지 않고 다시 추가해야 합니다. 마찬가지로, 로컬(라우터 주소 = 0)로 정의된 서버를 라우터 주소를 변경하여 원격 서

버로 만들 수 없습니다. 서버를 제거했다 다시 추가해야 합니다. 244 페이지의 『광역 Dispatcher 지원 구성』에서 자세한 정보를 참조하십시오.

주: 라우터는 Dispatcher에만 적용됩니다. nat 또는 cbr 전달 메소드를 사용하는 경우, 구성에 서버를 추가할 때 라우터 주소를 지정해야 합니다.

addr

라우터 주소의 값.

returnaddress

고유한 IP 주소 또는 호스트 이름. Dispatcher가 서버로 클라이언트 요청을 로드 밸런스할 때 출발지 주소로 사용하는 Dispatcher 시스템에 구성된 주소입니다. 서버가 Dispatcher 시스템으로 패킷을 리턴하여 클라이언트로 직접 패킷을 전송하지 않고 콘텐츠를 처리할 수 있도록 합니다(Dispatcher가 클라이언트로 IP 패킷을 전달합니다). 서버가 추가될 때 리턴 주소 값을 지정해야 합니다. 서버를 제거한 후 다시 추가하는 경우를 제외하고 리턴 주소는 변경할 수 없습니다. 리턴 주소는 클러스터, 서버 또는 NFA 주소와 같을 수 없습니다.

주: Returnaddress는 Dispatcher에만 적용됩니다. nat 또는 cbr 전달 메소드를 사용하는 경우, 구성에 서버를 추가할 때 리턴 주소를 지정해야 합니다.

addr

리턴 주소 값.

advisorrequest

HTTP 또는 HTTPS 어드바이저는 어드바이저 요청 문자열을 사용하여 서버 상태를 조회합니다. HTTP 또는 HTTPS 어드바이저가 권고하는 서버에만 유효합니다. 이 값을 사용하려면 HTTP 또는 HTTPS 어드바이저를 시작해야 합니다. 205 페이지의 『응답 및 요청(URL) 옵션을 사용하여 HTTP 또는 HTTPS 어드바이저 구성』에서 자세한 정보를 참조하십시오.

주: advisorrequest는 Dispatcher 및 CBR 컴포넌트에 적용됩니다.

string

HTTP 또는 HTTPS 어드바이저가 사용하는 문자열 값. 기본값은 HEAD / HTTP/1.0 입니다.

주: 문자열에 공백이 포함된 경우에는 다음에 주의하십시오.

- **dscontrol>>** 셸 프롬프트에서 명령을 발행할 경우, 문자열을 따옴표(')로 묶어야 합니다. 예제: **server set cluster:port:server advisorrequest "head / http/1.0"**
- 운영 체제 프롬프트에서 **dscontrol** 명령을 발행할 경우, 텍스트 앞에 "\"를, 텍스트 뒤에 \"\"를 표시해야 합니다. 예제: **dscontrol server set cluster:port:server advisorrequest "\"head / http/1.0\""**

advisorresponse

HTTP 응답에서 HTTP 또는 HTTPS 어드바이저가 스캔하는 어드바이저 응답 문자열. HTTP 또는 HTTPS 어드바이저가 권고하는 서버에만 유효합니다. 이 값을 사용하려면 HTTP 또는 HTTPS 어드바이저를 시작해야 합니다. 205 페이지의 『응답 및 요청(URL) 옵션을 사용하여 HTTP 또는 HTTPS 어드바이저 구성』에서 자세한 정보를 참조하십시오.

주: advisorresponse는 Dispatcher 및 CBR 컴포넌트에 적용됩니다.

string

HTTP 또는 HTTPS 어드바이저가 사용하는 문자열 값. 기본값은 널입니다.

주: 문자열에 공백이 포함된 경우에는 다음에 주의하십시오.

- **dscontrol>>** 셸 프롬프트에서 명령을 발행할 경우, 문자열을 따옴표(')로 묶어야 합니다.
- 운영 체제 프롬프트에서 **dscontrol** 명령을 발행할 경우, 텍스트 앞에 "\"를, 텍스트 뒤에 \"\"를 표시해야 합니다.

down

이 서버가 단절되었음을 표시합니다. 이 명령은 해당 서버에 대해 모든 활성 연결을 중단하고, 다른 연결이나 패킷이 해당 서버에 전송되지 못하도록 합니다.

서버 오프라인을 위해 서버 down 명령을 사용하는 경우, 해당 서버에 대한 stickytime 값이 0이 아니면 해당 서버는 stickytime이 끝날 때까지 기존의 클라이언트에 서비스를 제공합니다. 서버는 stickytime 값이 만료할 때까지 단절되지 않습니다.

remove

해당 서버를 제거합니다.

report

해당 서버에 대해 보고합니다. 보고서에는 각 서버에 대한 초당 현재 연결 수(CPS), 1초 간격으로 전송된 킬로바이트(KBPS), 연결 총계(Total), 활성 상태에 있는 연결 수(Active), FIN 상태의 연결 수(FINed) 및 완료된 연결 수(Comp) 등과 같은 정보가 포함되어 있습니다.

set

해당 서버의 값을 설정합니다.

status

서버의 상태를 표시합니다.

up 이 서버를 연결할 것을 표시합니다. Dispatcher는 이제 해당 서버에 대한 새로운 연결을 전송합니다.

예제

- 클러스터 주소 130.40.52.153에 있는 포트 80에 27.65.89.42에 있는 서버를 추가하려면 다음 명령을 실행하십시오.

```
dscontrol server add 130.40.52.153:80:27.65.89.42
```

- 27.65.89.42에서 비결합(포트 연관 관계 무시 기능) 상태로 서버를 설정하려면 다음 명령을 실행하십시오.

```
dscontrol server set 130.40.52.153:80:27.65.89.42 sticky no
```

- 27.65.89.42에 있는 서버를 단절되도록 표시하려면 다음 명령을 실행하십시오.

```
dscontrol server down 130.40.52.153:80:27.65.89.42
```

- 모든 클러스터에 있는 모든 포트에서 27.65.89.42에 있는 서버를 제거하려면 다음 명령을 실행하십시오.

```
dscontrol server remove ::27.65.89.42
```

- 27.65.89.42에서 결합 배치(서버가 []와 동일한 시스템에 위치)로 서버를 설정하려면 다음 명령을 실행하십시오.

```
dscontrol server set 130.40.52.153:80:27.65.89.42 collocated yes
```

- 클러스터 주소 130.40.52.153에 있는 포트 80에서 서버 27.65.89.42의 가중치를 10으로 설정하려면 다음 명령을 실행하십시오.

```
dscontrol server set 130.40.52.153:80:27.65.89.42 weight 10
```

- 27.65.89.42에 있는 서버를 연결되도록 표시하려면 다음 명령을 실행하십시오.

```
dscontrol server up 130.40.52.153:80:27.65.89.42
```

- 원격 서버를 추가하려면 다음 명령을 실행하십시오.

```
dscontrol server add 130.40.52.153:80:130.60.70.1 router 130.140.150.0
```

- HTTP 어드바이저가 HTTP 포트 80에서 서버 27.65.89.42에 대해 HTTP URL 요청 HEAD / HTTP/1.0을 조회할 수 있도록 하려면 다음 명령을 실행하십시오.

```
dscontrol server set 130.40.52.153:80:27.65.89.42  
advisorrequest "\"HEAD / HTTP/1.0\""
```

- 포트 80에서 서버 9.67.143.154의 상태를 표시하려면 다음 명령을 실행하십시오.

```
dscontrol server status 9.67.131.167:80:9.67.143.154
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
Server Status:  
-----  
Server ..... 9.67.143.154  
Port number ..... 80  
Cluster ..... 9.67.131.167  
Cluster address ..... 9.67.131.167  
Quiesced ..... N  
Server up ..... Y  
Weight ..... 10  
Fixed weight ..... N
```

Sticky for rule Y
Remote server N
Network Router address 0.0.0.0
Collocated N
Advisor request..... HEAD / HTTP/1.0
Advisor response.....
Cookie value n/a
Clone ID n/a

dscontrol set — 서버 로그 구성



loglevel

dsserver가 활동을 로그하는 레벨.

level

loglevel의 기본값은 0입니다. 범위는 0 - 5입니다. 가능한 값은 다음과 같습니다.
0은 없음, 1은 최소, 2는 기본, 3은 중간, 4는 고급, 5는 자세히입니다.

logsize

로그 파일에 기록할 최대 바이트 수.

size

logsize의 기본값은 1MB입니다.

dscontrol status — 관리자 및 어드바이저가 실행 여부 표시

▶—dscontrol—status—▶

예제

- 수행되고 있는 내용을 보려면 다음 명령을 실행하십시오.

```
dscontrol status
```

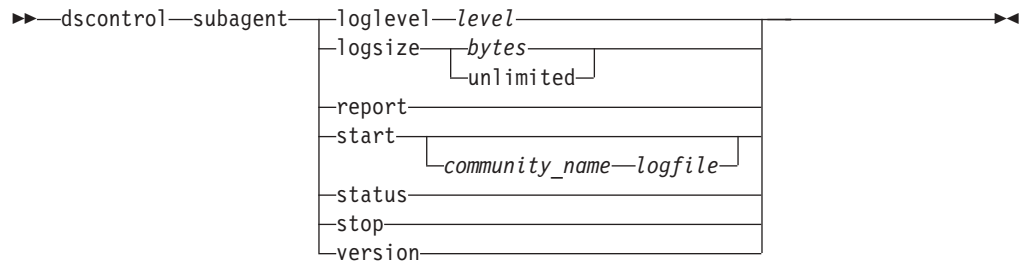
이 명령으로 다음과 같은 출력이 작성됩니다.

```
Executor has been started.  
Manager has been started.
```

ADVISOR	CLUSTER:PORT	TIMEOUT
reach	0	unlimited
http	80	unlimited
ftp	21	unlimited

dscontrol subagent — SNMP 서브에이전트 구성

주: dscontrol 서브에이전트 명령 구문 다이어그램은 Dispatcher 컴포넌트에 적용됩니다.



loglevel

서브에이전트가 활동을 파일에 기록하는 레벨.

level

레벨 번호(0-5). 번호가 커질수록 더 자세한 정보가 관리자 로그에 기록됩니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다. 0은 없음, 1은 최소, 2는 기본, 3은 중간, 4는 고급, 5는 자세합니다.

logsize

서브에이전트 로그에 기록될 최대 바이트 크기를 설정합니다. 기본값은 1MB입니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서부터 기록된 후속 항목을 겹쳐씹니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 높은 레벨에서 기록될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

bytes

서브에이전트 로그 파일의 최대 크기(바이트 단위). 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 파일은 로그 항목 그 자체의 크기가 변하므로, 겹쳐쓰기 전에는 그 정확한 최대 크기에 도달할 수 없습니다. 기본값은 unlimited입니다.

report

통계 스냅샷 보고서를 표시합니다.

start

서브에이전트를 시작합니다.

community_name

보안 암호로 사용할 수 있는 공동체 이름의 SNMP 값 이름. 기본값은 public입니다.

Windows 플랫폼의 경우: 운영 체제의 공동체 이름이 사용됩니다.

log file

SNMP 서브에이전트 데이터가 기록되는 파일 이름. 로그의 각 레코드에는 시간 소인이 표시됩니다. 기본값은 subagent.log입니다. 기본 파일은 **logs** 디렉토리에 설치됩니다. 511 페이지의 부록 C 『예제 구성 파일』을 참조하십시오. 로그 파일이 보존되는 디렉토리를 변경하려면 288 페이지의 『로그 파일 경로 변경』을 참조하십시오.

status

해당 기본값으로 글로벌하게 설정될 수 있는 SNMP 서브에이전트의 모든 값에 대한 현재 상태를 표시합니다.

version

서브에이전트의 현재 버전을 표시합니다.

예제

- 공동체 이름 bigguy로 서브에이전트를 시작하려면 다음 명령을 실행하십시오.
`dscontrol subagent start bigguy bigguy.log`

제 28 장 Site Selector 명령어 참조서

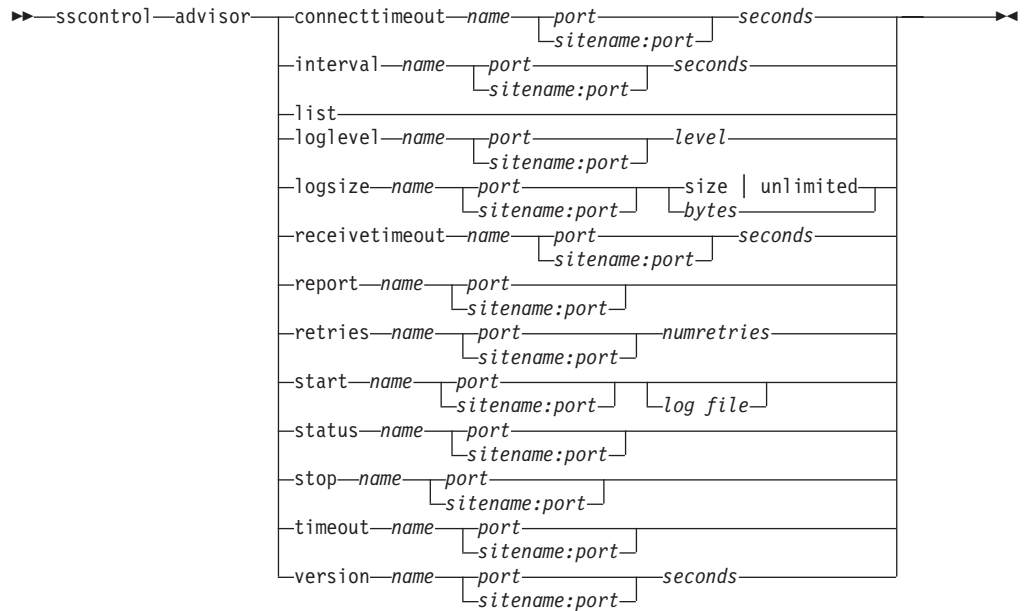
이 장에서는 다음 Site Selector **sscontrol** 명령을 사용하는 방법을 설명합니다.

- 430 페이지의 『sscontrol advisor — 어드바이저 제어』
- 435 페이지의 『sscontrol file — 구성 파일 관리』
- 437 페이지의 『sscontrol help — 이 명령의 도움말 표시 또는 인쇄』
- 438 페이지의 『sscontrol logstatus — 서버 로그 설정 표시』
- 439 페이지의 『sscontrol manager — 관리자 제어』
- 444 페이지의 『sscontrol metric — 시스템 메트릭 구성』
- 445 페이지의 『sscontrol nameserver — NameServer 제어』
- 446 페이지의 『sscontrol rule — 규칙 구성』
- 449 페이지의 『sscontrol server — 서버 구성』
- 451 페이지의 『sscontrol set — 서버 로그 구성』
- 452 페이지의 『sscontrol sitename — 사이트 이름 구성』
- 455 페이지의 『sscontrol status — 관리자 및 어드바이저가 실행 여부 표시』

sscontrol 명령 매개변수의 최소 버전을 입력할 수 있습니다. 매개변수의 고유한 문자만 입력해야 합니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면, **sscontrol help file** 대신에 **sscontrol he f**를 입력할 수 있습니다.

주: 명령 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름(클러스터 및 서버 명령에서 사용)과 파일 이름(파일 명령에 사용)만 예외입니다.

sscontrol advisor — 어드바이저 제어



connecttimeout

서버와의 연결 실패를 보고하기 전에 어드바이저가 대기하는 시간을 설정하십시오. 자세한 정보는 202 페이지의 『어드바이저 연결 제한시간 및 서버의 수신 제한시간』을 참조하십시오.

name

어드바이저 이름. 가능한 값은 **http**, **https**, **ftp**, **sip**, **ssl**, **smtp**, **imap**, **pop3**, **ldap**, **nntp**, **telnet**, **connect**, **ping**, **WLM** 및 **WTE**입니다. 사용자 정의 어드바이저 이름은 xxxx 형식이며, 여기서 ADV_xxxx는 사용자 정의 어드바이저를 구현하는 클래스의 이름입니다.

port

어드바이저가 모니터링하는 포트의 번호.

seconds

서버와의 연결 실패를 보고하기 전에 어드바이저가 대기하는 시간을 초 단위로 표시하는 양의 정수. 기본값은 어드바이저 간격에 지정된 값의 3배입니다.

interval

어드바이저가 정보를 찾아 서버를 조회하는 간격을 설정하십시오.

seconds

서버에 상태 요청을 하는 간격(초 단위)을 나타내는 양의 정수. 기본값은 7입니다.

list

관리자에 현재 정보를 제공하는 어드바이저 목록을 표시합니다.

loglevel

어드바이저 로그에 대한 로그 레벨을 설정합니다.

level

레벨 번호(0-5). 기본값은 1입니다. 번호가 커질수록 더 자세한 정보가 어드바이저 로그에 기록됩니다. 가능한 값은 다음과 같습니다.

- 0은 없음
- 1은 최소
- 2는 기본
- 3은 중간
- 4는 고급
- 5는 자세히

logsize

어드바이저 로그의 최대 크기를 설정합니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 후속 항목을 겹쳐 씁니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 상위 레벨에서 로그될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

size | unlimited

어드바이저 로그 파일의 최대 크기(바이트 단위). 0보다 큰 양의 정수 또는 **unlimited**를 지정할 수 있습니다. 로그 항목 크기가 변하므로 겹쳐쓰기 전에 로그 파일은 정확한 최대 크기에 도달할 수 없습니다. 기본값은 1MB입니다.

receivetimeout

서버에서 수신 실패를 보고하기 전에 어드바이저가 대기하는 시간을 설정합니다. 자세한 정보는 202 페이지의 『어드바이저 연결 제한시간 및 서버의 수신 제한시간』을 참조하십시오.

seconds

서버에서 수신 실패를 보고하기 전에 어드바이저가 대기하는 시간을 초 단위로 표시하는 양의 정수. 기본값은 어드바이저 간격에 지정된 값의 3배입니다.

report

어드바이저 상태에 대한 보고서를 표시합니다.

retries

어드바이저가 서버를 종료된 서버로 표시하기 전에 수행할 수 있는 재시도 횟수.

numretries

0 이상의 정수. 이 값은 3을 초과하면 안 됩니다. **retry** 키워드를 구성하지 않을 경우, 재시도 횟수는 기본적으로 0입니다.

start

어드바이저를 시작합니다. 각 프로토콜에 대해 어드바이저가 있습니다. 기본 포트는 다음과 같습니다.

어드바이저 이름	프로토콜	포트
Connect	n/a	사용자 정의
db2	개인용	50000
ftp	FTP	21
http	HTTP	80
https	SSL	443
imap	IMAP	143
ldap	LDAP	389
nntp	NNTP	119
PING	PING	N/A
pop3	POP3	110
sip	SIP	5060
smtp	SMTP	25
ssl	SSL	443
telnet	Telnet	23

name

어드바이저 이름.

sitename:port

sitename 값은 어드바이저 명령에서 선택적이지만, port 값은 필수입니다. sitename 값이 지정되지 않으면 어드바이저는 구성된 모든 사용 가능한 사이트 이름에서 실행됩니다. 사이트 이름을 지정하면 어드바이저는 지정된 사이트 이름에서만 실행을 시작합니다. 추가 사이트 이름은 더하기 부호(+)로 구분됩니다.

log file

관리 데이터가 기록되는 파일 이름. 로그의 각 레코드에는 시간 소인이 표시됩니다.

기본 파일은 `advisorname_port.log`이며 예를 들어, **http_80.log**입니다. 로그 파일이 저장되는 디렉토리를 변경하려면 288 페이지의 『로그 파일 경로 변경』을 참조하십시오.

각 사이트 이름에 대해 하나의 어드바이저만 시작할 수 있습니다.

status

어드바이저의 모든 글로벌 값에 대한 기본값 및 현재 상태를 표시합니다.

stop

어드바이저를 정지합니다.

timeout

관리자가 어드바이저 정보를 유효한 것으로 간주하는 시간(초 단위)을 설정합니다.

어드바이저 정보가 제한시간 기간보다 이전의 정보라는 것을 관리자가 발견하면, 관리자는 어드바이저가 모니터하는 포트에서 서버에 대한 가중치를 결정할 때 이 정보를 사용하지 않습니다. 이 제한시간에 대한 예외는 어드바이저가 관리자에 특정 서버가 단절되었다는 것을 알린 경우입니다. 관리자는 어드바이저 정보의 제한시간이 초과된 후에도 서버에 대한 해당 정보를 사용합니다.

seconds

초 수를 표시하는 양의 정수 또는 **unlimited**. 기본값은 unlimited입니다.

version

어드바이저의 현재 버전을 표시합니다.

예제

- 서버와의 연결 실패를 보고하기 전에 HTTP 어드바이저가(포트 80의 경우) 대기하는 시간(30초)을 설정하려면 다음 명령을 실행하십시오.

```
sscontrol advisor connecttimeout http 80 30
```

- FTP 어드바이저 간격(포트 21의 경우)을 6초로 설정하려면 다음 명령을 실행하십시오.

```
sscontrol advisor interval ftp 21 6
```

- 현재 관리자에 정보를 제공하는 어드바이저 목록을 표시하려면 다음 명령을 실행하십시오.

```
sscontrol advisor list
```

이 명령으로 다음과 같은 출력이 작성됩니다.

ADVISOR	SITENAME:PORT	TIMEOUT
http	80	unlimited
ftp	21	unlimited

- 성능을 개선하기 위해 mysite라는 사이트의 http 어드바이저 로그의 로그 레벨을 0으로 변경하려면 다음 명령을 실행하십시오.

```
sscontrol advisor loglevel http mysite:80 0
```

- mysite라는 사이트의 ftp 어드바이저 로그 크기를 5000 바이트로 변경하려면 다음 명령을 실행하십시오.

```
sscontrol advisor logsize ftp mysite:21 5000
```

- 서버에서 수신 실패를 보고하기 전에 HTTP 어드바이저(포트 80의 경우)가 대기하는 시간(60초)을 설정하려면 다음 명령을 실행하십시오.

```
sscontrol advisor receivetimeout http 80 60
```

- FTP 어드바이저 상태에 대한 보고서를 표시하려면(포트 21의 경우) 다음 명령을 실행하십시오.

```
sscontrol advisor report ftp 21
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Advisor Report:

Advisor name http

Port number 80

sitename mySite

Server address 9.67.129.230

Load 8

- ftpadv.log 파일을 사용하여 어드바이저를 시작하려면 다음 명령을 실행하십시오.

```
sscontrol advisor start ftp 21 ftpadv.log
```

- http 어드바이저와 연관된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

```
sscontrol advisor status http 80
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Advisor Status:

Interval (seconds) 7

Timeout (seconds) Unlimited

Connect timeout (seconds).....21

Receive timeout (seconds).....21

Advisor log filename Http_80.log

Log level 1

Maximum log size (bytes) Unlimited

Number of retries 0

- 포트 80에서 http 어드바이저를 중지하려면 다음 명령을 실행하십시오.

```
sscontrol advisor stop http 80
```

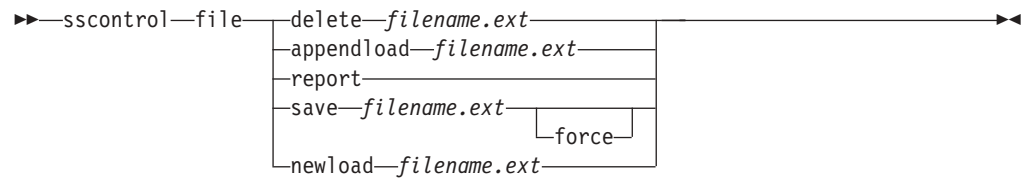
- 어드바이저 정보에 대한 제한시간 값을 5초로 설정하려면 다음 명령을 실행하십시오.

```
sscontrol advisor timeout ftp 21 5
```

- ssl 어드바이저의 현재 버전 번호를 찾으려면 다음 명령을 실행하십시오.

```
sscontrol advisor version ssl 443
```

sscontrol file — 구성 파일 관리



delete

파일을 삭제합니다.

file.ext

구성 파일.

파일 확장자(.ext)는 사용자가 원하는 대로 정할 수 있으며 선택적입니다.

appendload

현재 구성에 구성 파일을 추가하고 Site Selector로 로드합니다.

report

사용 가능한 파일에 대해 보고합니다.

save

Site Selector의 현재 구성 파일을 파일에 저장합니다.

주: 파일은 다음의 디렉토리에 저장되어 로드됩니다.

- Linux 및 UNIX 시스템: **/opt/ibm/edge/lb/servers/configurations/ss**
- Windows 시스템: **C:\Program Files\ibm\edge\lb\servers\configurations\component**

force

파일을 기존 파일과 동일한 이름으로 저장하려면 새 파일을 저장하기 전에 **force**를 사용하여 기존 파일을 삭제하십시오. force 옵션을 사용하지 않으면 기존 파일에 겹쳐 쓸 수 없습니다.

newload

Site Selector로 새 구성 파일을 로드합니다. 새 구성 파일은 현재 구성을 바꿉니다.

예제

- 파일을 삭제하려면 다음 명령을 실행하십시오.

```
sscontrol file delete file3
```

```
File (file3) was deleted.
```

- 새 구성 파일을 로드하여 현재 구성을 대체하려면 다음 명령을 실행하십시오.

```
sscontrol file newload file1.sv
```

File (file1.sv) was loaded into the Dispatcher.

- 현재 구성에 구성 파일을 추가하고 로드하려면 다음 명령을 실행하십시오.

```
sscontrol file appendload file2.sv
```

File (file2.sv) was appended to the current configuration and loaded.

- 파일(즉, 이전에 사용자가 저장한 파일)의 보고서를 보려면 다음 명령을 실행하십시오.

```
sscontrol file report
```

FILE REPORT:

file1.save

file2.sv

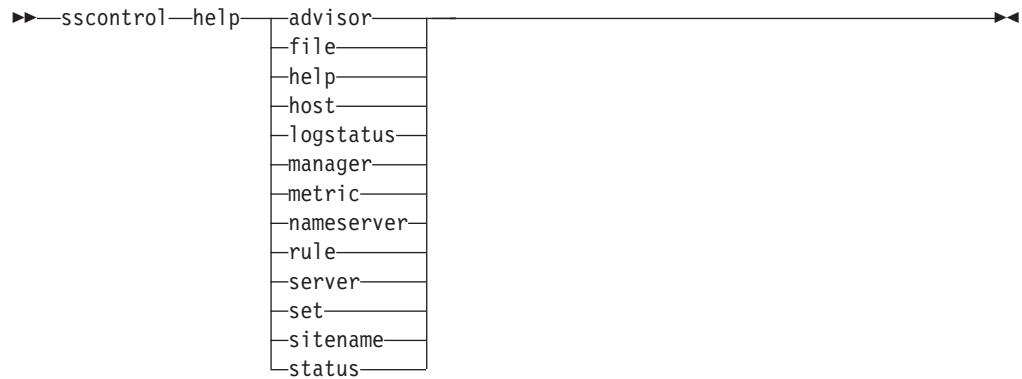
file3

- 구성을 file3 파일에 저장하려면 다음 명령을 실행하십시오.

```
sscontrol file save file3
```

The configuration was saved into file (file3).

sscontrol help — 이 명령의 도움말 표시 또는 인쇄



예제

- sscontrol 명령에 대한 도움말을 보려면 다음 명령을 실행하십시오.

```
sscontrol help
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
HELP COMMAND ARGUMENTS:
```

```
-----
```

```
Usage: help <help option>
```

```
Example: help name
```

```
help          - print complete help text  
adviser       - help on adviser command  
file          - help on file command  
host          - help on host command  
manager       - help on manager command  
metric        - help on metric command  
sitename      - help on sitename command  
nameserver    - help on nameserver command  
rule          - help on rule command  
server        - help on server command  
set           - help on set command  
status        - help on status command  
logstatus     - help on logstatus command
```

< > 내의 매개변수는 변수입니다.

- 다음과 같이 |를 사용하여 옵션을 구분하는 변수에 대한 선택사항이 도움말에 표시됩니다.

```
logsize <number of bytes | unlimited>
```

```
-Set the maximum number of bytes to be logged in the log file
```

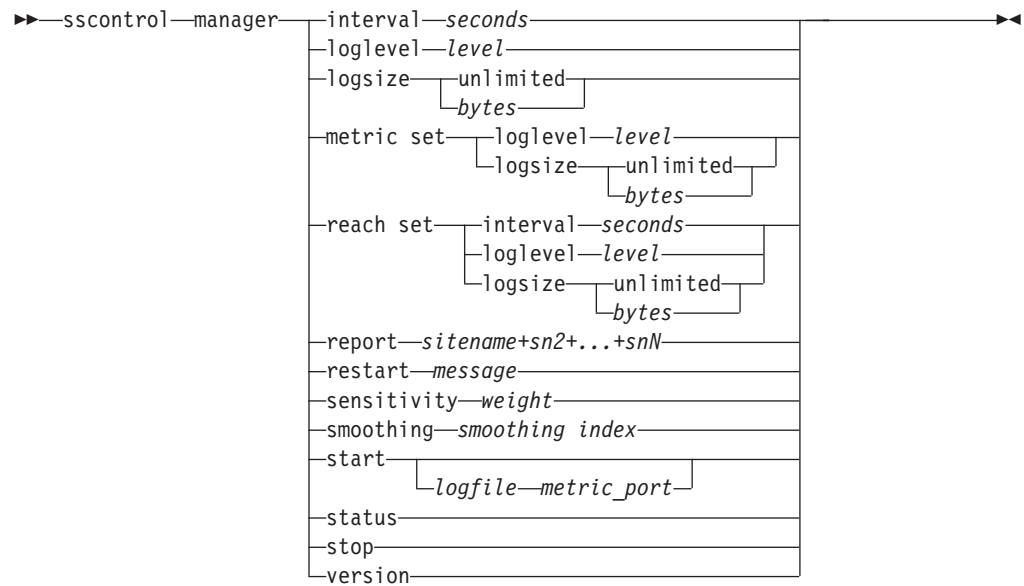
sscontrol logstatus — 서버 로그 설정 표시

▶▶—sscontrol—logstatus—◀◀

logstatus

서버 로그 설정(로그 파일 이름, 로그 레벨 및 로그 크기)을 표시합니다.

sscontrol manager — 관리자 제어



interval

관리자가 서버의 가중치를 갱신하는 빈도를 설정합니다.

seconds

관리자가 가중치를 갱신하는 빈도를 표시하는 양수(초 단위). 기본값은 2입니다.

loglevel

관리자 로그에 대한 로그 레벨을 설정합니다.

level

레벨 번호(0-5). 번호가 커질수록 더 자세한 정보가 관리자 로그에 기록됩니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다.

- 0은 없음
- 1은 최소
- 2는 기본
- 3은 중간
- 4는 고급
- 5는 자세히

logsize

관리자 로그의 최대 크기를 설정합니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서부터 기록된 후속 항목을 겹쳐 씁니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 상위 레벨에서 로그될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

bytes

관리자 로그 파일의 최대 크기(바이트 단위). 0보다 큰 양의 정수 또는 **unlimited**를 지정할 수 있습니다. 로그 항목 크기가 변하므로 겹쳐쓰기 전에 로그 파일은 정확한 최대 크기에 도달할 수 없습니다. 기본값은 1MB입니다.

metric set

메트릭 모니터 로그에 대한 로그 레벨 및 로그 크기를 설정합니다. 로그 레벨은 메트릭 모니터 로그 레벨입니다(0 - 없음, 1 - 최소, 2 - 기본, 3 - 중간, 4 - 고급 또는 5 - 자세히). 기본 로그 레벨은 1입니다. 로그 크기는 메트릭 모니터 로그 파일에 기록될 최대 바이트 수입니다. 0보다 큰 양의 정수 또는 **unlimited**를 지정할 수 있습니다. 기본 로그 크기는 1입니다.

reach set

도달 어드바이저에 대한 간격, 로그 레벨 및 로그 크기를 설정합니다.

report

통계 스냅샷 보고서를 표시합니다.

sitename

보고서에 표시할 사이트 이름. 클라이언트가 요청할, 해석되지 않는 호스트 이름입니다. 사이트 이름은 완전한 도메인 이름이어야 합니다.

주: 추가 사이트 이름은 더하기 부호(+)로 구분됩니다.

restart

모든 서버(단절되지 않은)를 재시작하여 가중치를 표준화합니다(최대 가중치의 1/2).

message

관리자 로그 파일에 기록할 메시지.

sensitivity

가중치가 갱신되는 최소 감도를 설정합니다. 이 설정은 외부 정보에 따라 서버에 대한 가중치를 관리자에서 변경해야 할 시기를 정의합니다.

weight

가중치 백분율로 사용되는 0에서 100 사이의 숫자. 기본값 5를 사용하면 최소 감도는 5%가 됩니다.

smoothing

로드 밸런스 시 가중치의 변화를 평탄화하는 지수를 설정합니다. 스무스 색인을 높게 설정하면 네트워크 조건이 변경됨에 따라 서버 가중치가 덜 변경됩니다. 색인이 낮으면 서버 가중치가 더 변경됩니다.

index

양의 부동 소수점 수. 기본값은 1.5입니다.

start

관리자를 시작합니다.

log file

관리자 데이터가 기록되는 파일 이름. 로그의 각 레코드에는 시간 소인이 표시됩니다.

기본 파일은 **logs** 디렉토리에 설치됩니다. 511 페이지의 부록 C 『예제 구성 파일』을 참조하십시오. 로그 파일이 보존되는 디렉토리를 변경하려면 288 페이지의 『로그 파일 경로 변경』을 참조하십시오.

metric_port

Metric Server가 시스템 로드를 보고하는 데 사용하는 포트측정 기준 포트를 지정할 경우, 로그 파일 이름을 지정해야 합니다. 기본 측정 기준 포트는 10004입니다.

status

관리자의 모든 글로벌 값에 대한 기본값 및 현재 상태를 표시합니다.

stop

관리자를 정지합니다.

version

관리자 현재 버전을 표시합니다.

예제

- 관리자의 갱신 간격을 매 5초로 설정하려면 다음 명령을 실행하십시오.
`sscontrol manager interval 5`
 - 더 나은 성능을 위해 로그 레벨을 0으로 설정하려면 다음 명령을 실행하십시오.
`sscontrol manager loglevel 0`
 - 관리자 로그 크기를 1,000,000 바이트를 설정하려면 다음 명령을 실행하십시오.
`sscontrol manager logsize 1000000`
 - 관리자의 통계 스냅샷을 확보하려면 다음 명령을 실행하십시오.
`sscontrol manager report`
- 이 명령으로 다음과 같은 출력이 작성됩니다.

SERVER	STATUS
9.67.129.221	ACTIVE
9.67.129.213	ACTIVE
9.67.134.223	ACTIVE

MANAGER REPORT LEGEND	
CPU	CPU Load
MEM	Memory Load
SYS	System Metric
NOW	Current Weight
NEW	New Weight
WT	Weight

mySite	WEIGHT	CPU 49%	MEM 50%	PORT 1%	SYS 0%					
	NOW	NEW	WT	LOAD	WT	LOAD	WT	LOAD	WT	LOAD
9.37.56.180	10	10	-99	-1	-99	-1	-99	-1	0	0
TOTALS:	10	10		-1		-1		-1		0

ADVISOR	SITENAME:PORT	TIMEOUT
http	80	unlimited

- 표준화된 가중치에 대해 모든 서버를 재시작하고 메시지를 관리자 로그 파일에 기록하려면 다음 명령을 실행하십시오.

```
sscontrol manager restart Restarting the manager to update code
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
320-14:04:54 Restarting the manager to update code
```

- 가중치 변경에 대한 감도를 10으로 설정하려면 다음 명령을 실행하십시오.

```
sscontrol manager sensitivity 10
```

- 스무스 색인을 2.0으로 설정하려면 다음 명령을 실행하십시오.

```
sscontrol manager smoothing 2.0
```

- 관리자를 시작하고 ndmgr.log(경로는 설정할 수 없음) 로그 파일을 지정하려면 다음 명령을 실행하십시오.

```
sscontrol manager start ndmgr.log
```

- 관리자와 관련된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

```
sscontrol manager status
```


이 명령으로 다음 예제와 같은 출력이 작성됩니다.

```
Manager status:
=====
Metric port..... 10004
Manager log filename..... manager.log
Manager log level..... 1
Maximum manager log size (bytes)..... unlimited
Sensitivity level..... 5
Smoothing index..... 1.5
Update interval (seconds)..... 2
Weights refresh cycle..... 2
Reach log level..... 1
Maximum reach log size (bytes)..... unlimited
Reach update interval (seconds)..... 7
```

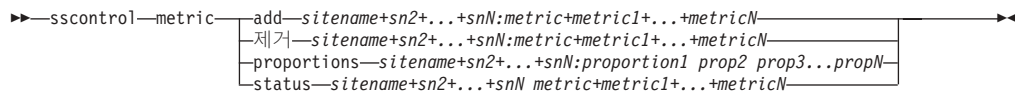
- 관리자를 정지하려면 다음 명령을 실행하십시오.

```
sscontrol manager stop
```

- 관리자의 현재 버전 번호를 표시하려면 다음 명령을 실행하십시오.

```
sscontrol manager version
```

sscontrol metric — 시스템 메트릭 구성



add

지정된 메트릭을 추가합니다.

sitename

구성된 사이트 이름. 추가 사이트 이름은 더하기 부호(+)로 구분됩니다.

metric

시스템 메트릭 이름. Metric Server의 스크립트 디렉토리에 있는 스크립트 파일 또는 실행 파일의 이름이어야 합니다.

remove

지정된 메트릭을 제거합니다.

proportions

비율은 메트릭이 서버에 대한 단일 시스템 로드에서 결합될 때 다른 메트릭과 비교해서 각 메트릭의 중요성을 판별합니다.

status

이 메트릭에 대한 현재 서버 값을 표시합니다.

예제

- 시스템 메트릭을 추가하려면 다음 명령을 실행하십시오.

```
sscontrol metric add site1:metric1
```
- 두 개의 시스템 메트릭을 사용하여 사이트 이름에 비율을 설정하려면 다음 명령을 실행하십시오.

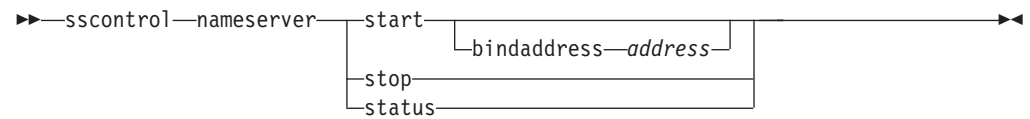
```
sscontrol metric proportions site1 0 100
```
- 지정된 메트릭과 연관된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

```
sscontrol metric status site1:metric1
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
Metric Status:
-----
sitename ..... site1
Metric name ..... metric1
Metric proportion ..... 50
Server ..... 9.37.56.100
Metric data .... -1
```

sscontrol nameserver — NameServer 제어



start

이름 서버를 시작합니다.

bindaddress

지정된 주소에 바인드된 이름 서버를 시작합니다. 이름 서버는 이 주소로 지정된 요청에만 응답합니다.

address

Site Selector 시스템에서 구성된 주소(IP 또는 기호)

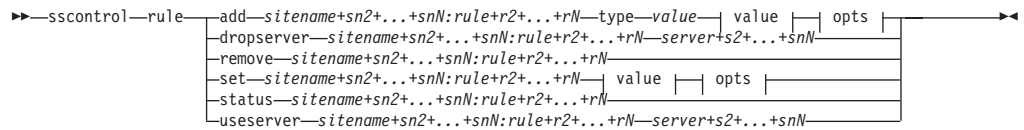
stop

이름 서버를 정지합니다.

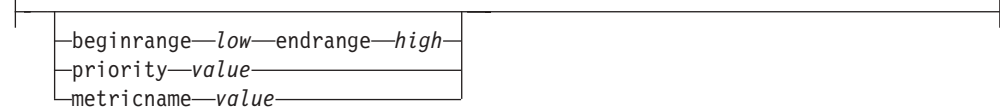
status

이름 서버 상태를 표시합니다.

sscontrol rule — 규칙 구성



opts:



add

이 규칙을 사이트 이름에 추가합니다.

sitename

클라이언트가 요청할 분석되지 않는 호스트 이름. 사이트 이름은 완전한 도메인 이름이어야 합니다. 추가 사이트 이름은 더하기 부호(+)로 구분됩니다.

rule

사용자가 규칙에 대해 선택한 이름. 이 이름에는 영숫자, 밑줄, 하이픈 또는 마침표가 포함될 수 있습니다. 이 이름은 한 개부터 20개까지의 문자가 가능하며 공백이 포함될 수 없습니다.

주: 추가 규칙은 더하기 부호(+)로 구분됩니다.

type

규칙의 유형.

type

type의 선택사항은 다음과 같습니다.

ip 규칙이 클라이언트 IP 주소에 기초합니다.

metricall

규칙은 서버 세트의 모든 서버에 대한 현재 메트릭 값을 기반으로 합니다.

metricavg

규칙은 서버 세트의 모든 서버에 대한 현재 메트릭의 평균값을 기반으로 합니다.

time 규칙이 시간에 기초합니다.

true 이 규칙은 항상 true입니다. 이는 프로그래밍 논리에서 else문으로 간주하십시오.

beginrange

규칙이 올바른지 여부를 판별하는 데 사용되는 범위 내의 하위값.

low

규칙의 유형을 따릅니다. 값의 종류와 기본값이 규칙의 유형에 따라 다음에 나열되어 있습니다.

ip 기호 이름 또는 IP 주소 형식으로 된 클라이언트의 주소. 기본값은 0.0.0.0입니다.

time 정수. 기본값은 0이며 자정을 나타냅니다.

metricall

정수. 기본값은 100입니다.

metricavg

정수. 기본값은 100입니다.

endrange

규칙이 올바른지 여부를 판별하는 데 사용되는 범위 내의 상위값.

high

규칙의 유형을 따릅니다. 값의 종류와 기본값이 규칙의 유형에 따라 다음에 나열되어 있습니다.

ip 기호 이름 또는 IP 주소 형식으로 된 클라이언트의 주소. 기본값은 255.255.255.254입니다.

time 정수. 기본값은 24이며 자정을 나타냅니다.

주: 시간 간격의 최소 범위 및 최대 범위를 정의할 때 각 값은 시간의 시 부분만을 나타내는 정수여야 한다는 점에 유의하십시오. 시 부분은 지정되지 않습니다. 이런 이유로 오전 3시와 4시 사이에서 하나의 시간을 지정하려면 최소 범위로 **3**을 지정하고 최대 범위도 **3**을 지정하십시오. 이는 3:00에서 시작하여 3:59에 끝나는 모든 시간(분 단위)을 지정합니다. 최소 범위를 **3**으로 지정하고 최대 범위를 **4**로 지정하면 3:00에서부터 4:59까지의 두 시간이 포함됩니다.

metricall

정수. 기본값은 2의 32제곱에서 1을 뺀 값입니다.

metricavg

정수. 기본값은 2의 32제곱에서 1을 뺀 값입니다.

priority

규칙이 검토되는 순서

level

정수. 사용자가 추가하는 첫 번째 규칙의 우선순위를 지정하지 않으면, Site Selector는 기본값으로 1을 설정합니다. 후속 규칙을 추가할 때 기본값으로 우선순위는 10 + 기존 규칙의 현재 최저 우선순위로 연산됩니다. 예를 들어, 기존 규칙의 우선순

위가 30이라고 가정합니다. 새 규칙을 추가하고 그 우선순위를 25(이는 30보다 높은 우선순위임)로 설정하십시오. 그런 다음, 우선순위를 설정하지 않고 세 번째 규칙을 추가합니다. 세 번째 규칙의 우선순위는 40(30 + 10)으로 계산됩니다.

metricname

규칙을 측정한 메트릭 이름.

dropserver

규칙 세트에서 서버를 제거합니다.

server

기호 이름 또는 IP 주소 형식으로 된 TCP 서버 시스템의 IP 주소.

주: 추가 사이트 이름은 더하기 부호(+)로 구분됩니다.

remove

더하기 부호로 구분된 하나 이상의 규칙을 제거합니다.

set

해당 규칙의 값을 설정합니다.

status

하나 이상의 규칙 값을 모두 표시합니다.

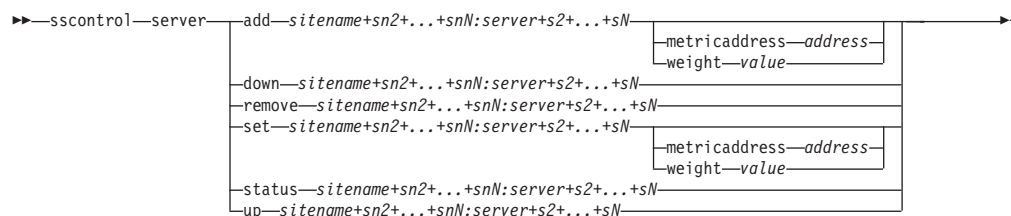
useserver

서버를 규칙 세트에 삽입합니다.

예제

- 항상 참이 되는 규칙을 추가하려면 최소 범위 또는 최대 범위를 지정하지 마십시오.
`sscontrol rule add sitename:rulename type true priority 100`
- IP 주소 범위(이 경우에는 “9”로 시작함)에 대한 액세스를 금지하는 규칙을 작성하려면 다음 명령을 실행하십시오.
`sscontrol rule add sitename:rulename type ip b 9.0.0.0 e 9.255.255.255`
- 오전 11시에서 오후 3시까지 제공된 서버의 사용을 지정하는 규칙을 작성하려면 다음 명령을 실행하십시오.
`sscontrol rule add sitename:rulename type time beginrange 11 endrange 14`
`sscontrol rule useserver sitename:rulename server05`

sscontrol server — 서버 구성



add

이 서버를 추가합니다.

sitename

클라이언트가 요청할 분석되지 않는 호스트 이름. 사이트 이름은 완전한 도메인 이름이어야 합니다. 추가 사이트 이름은 더하기 부호(+)로 구분됩니다.

server

기호 이름 또는 IP 주소 형식으로 된 TCP 서버 시스템의 IP 주소.

주: 추가 서버는 더하기 부호(+)로 구분됩니다.

metricaddress

Metric Server의 주소.

address

기호 이름 또는 IP 주소 형식으로 된 서버의 주소

weight

이 서버의 가중치를 표시하는 0 - 100의 숫자(지정된 사이트 이름의 최대 가중치 바운드 값을 초과할 수 없음). 가중치를 0으로 설정하면 새로운 요청이 서버로 전송되지 않습니다. 기본값은 지정된 사이트 이름의 최대 가중치 바운드 값을 2로 나눈 값입니다. 관리자가 실행되고 있는 경우, 이 설정은 신속하게 겹쳐쓰입니다.

value

서버 가중치 값

down

이 서버가 단절되었음을 표시합니다. 이 명령을 사용하면 다른 요청이 해당 서버로 해석되지 않습니다.

remove

해당 서버를 제거합니다.

set

해당 서버의 값을 설정합니다.

status

서버의 상태를 표시합니다.

up 이 서버를 연결할 것을 표시합니다. Site Selector는 이제 새 요청을 해당 서버로 해석합니다.

예제

- 27.65.89.42의 서버를 사이트 이름 site1에 추가하려면 다음 명령을 실행하십시오.
`sscontrol server add site1:27.65.89.42`
- 27.65.89.42에 있는 서버를 단절되도록 표시하려면 다음 명령을 실행하십시오.
`sscontrol server down site1:27.65.89.42`
- 모든 사이트 이름에 대해 27.65.89.42의 서버를 제거하려면 다음 명령을 실행하십시오.
`sscontrol server remove :27.65.89.42`
- 27.65.89.42에 있는 서버를 연결되도록 표시하려면 다음 명령을 실행하십시오.
`sscontrol server up site1:27.65.89.42`

sscontrol set — 서버 로그 구성



loglevel

ssserver가 활동을 로그하는 레벨.

level

loglevel의 기본값은 0입니다. 가능한 값은 다음과 같습니다.

- 0은 없음
- 1은 최소
- 2는 기본
- 3은 중간
- 4는 고급
- 5는 자세히

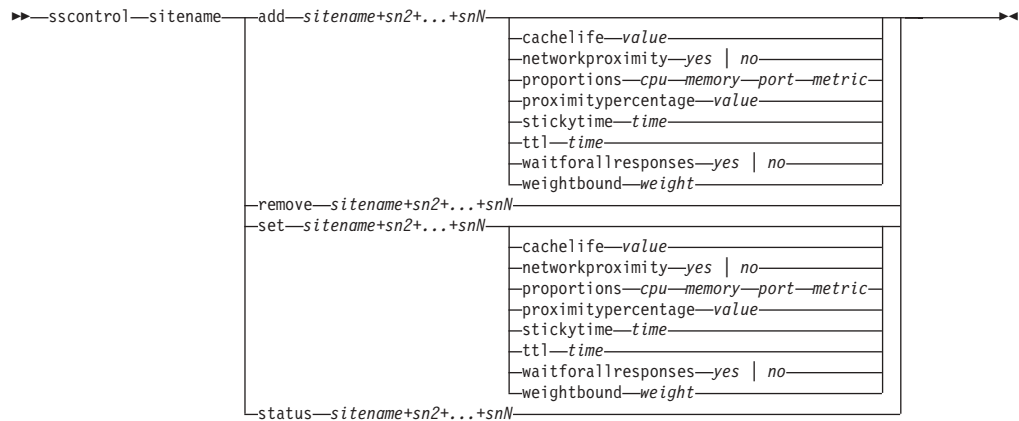
logsize

로그 파일에 기록할 최대 바이트 수.

size

logsize의 기본값은 1MB입니다.

sscontrol sitename — 사이트 이름 구성



add

새 사이트 이름을 추가합니다.

sitename

클라이언트가 요청한 분석할 수 없는 호스트 이름. 추가 사이트 이름은 더하기 부호(+)로 구분됩니다.

cachelife

proximity 응답이 유효하고 캐시에 저장되는 시간. 기본값은 1800입니다. 144 페이지의 『네트워크 근접 기능 사용』에서 자세한 정보를 참조하십시오.

value

proximity 응답이 유효하며 캐시에 저장되었음을 나타내는 양수(초 단위).

networkproximity

요청 클라이언트에 대한 각 서버의 네트워크 근접성을 판별합니다. 로드 밸런스 결정 시 proximity 응답을 사용하십시오. proximity를 on 또는 off로 설정하십시오. 144 페이지의 『네트워크 근접 기능 사용』에서 자세한 정보를 참조하십시오.

value

선택사항은 yes 또는 no입니다. 기본값은 no로 네트워크 근접성이 작동하지 않습니다.

proportions

관리자가 서버 가중치를 설정하는 데 사용하는 Metric Server에 대한 cpu, memory, port(어드바이저의 정보) 및 시스템 메트릭의 중요도 비율을 설정합니다. 각 값은 총계의 백분율로 표시되므로 총계는 항상 100입니다.

cpu 각각의 로드 밸런스된 서버 시스템(Metric Server 에이전트가 보낸 입력)에서 사용되는 CPU 백분율.

memory

각각의 로드 밸런스된 서버에서 사용 중인 메모리 백분율(Metric Server 에이전트가 보낸 입력).

port 포트에서 인식 중인 어드라이저가 보낸 입력.

system Metric Server가 보낸 입력.

proximitypercentage

proximity 응답 vs 서버 상태의 중요도를 설정(관리자 가중치)합니다. 144 페이지의 『네트워크 근접 기능 사용』에서 자세한 정보를 참조하십시오.

value

기본값은 50입니다.

stickytime

클라이언트가 첫 요청에 대해 이전에 리턴된 서버 ID와 동일한 서버 ID를 받는 간격. sticky time의 기본값은 0으로, 사이트 이름이 결합되어 있지 않음을 나타냅니다.

time

클라이언트가 첫 요청에 대해 이전에 리턴된 서버 ID와 동일한 서버 ID를 받는 시간을 나타내는 0이 아닌 양수(초 단위).

ttl 활성 시간을 설정합니다. 이 값은 다른 이름 서버가 해석된 응답을 캐시하는 시간을 나타냅니다. 기본값은 5입니다.

value

이름 서버가 해석된 응답을 캐시하는 시간을 나타내는 양수(초 단위).

waitforallresponses

클라이언트 요청에 응답하기 전에 서버에서 보내는 모든 proximity 응답의 대기 여부를 설정합니다. 144 페이지의 『네트워크 근접 기능 사용』에서 자세한 정보를 참조하십시오.

value

yes나 no 중에서 선택할 수 있습니다. 기본값은 yes입니다.

weightbound

이 사이트 이름의 서버에 설정할 수 있는 최대 가중치를 표시하는 숫자. 사이트에 설정된 가중치 바운드 값에 **server weight**를 사용하여 개별 서버에 대해 덮어쓸 수 있습니다. 사이트 이름 가중치 바운드 기본값은 20입니다.

weight

weightbound의 값.

set

사이트 이름의 등록 정보를 설정합니다.

remove

이 사이트 이름을 제거합니다.

status

고유 사이트 이름의 현재 상태를 표시합니다.

예제

- 사이트 이름을 추가하려면 다음 명령을 실행하십시오.
`sscontrol sitename add 130.40.52.153`
- 네트워크 근저성을 작동하려면 다음 명령을 실행하십시오.
`sscontrol sitename set mySite networkproximity yes`
- 캐시 수명을 1900000초로 설정하려면 다음 명령을 실행하십시오.
`sscontrol sitename set mySite cachelife 1900000`
- 근접성 퍼센트를 45로 설정하려면 다음 명령을 실행하십시오.
`sscontrol sitename set mySite proximitypercentage 45`
- 응답하기 전에 사이트 이름이 모든 응답을 기다리지 않도록 설정하려면 다음 명령을 실행하십시오.
`sscontrol sitename set mySite waitforallresponses no`
- live 시간을 7초로 설정하려면 다음 명령을 실행하십시오.
`sscontrol sitename set mySite ttl 7`
- CpuLoad, MemLoad, Port 및 System Metric에 각각 중요도를 설정하려면 다음 명령을 실행하십시오.
`sscontrol sitename set mySite proportions 50 48 1 1`
- 사이트 이름을 제거하려면 다음 명령을 실행하십시오.
`sscontrol sitename remove 130.40.52.153`
- 사이트 이름 mySite의 상태를 표시하려면 다음 명령을 실행하십시오.
`sscontrol sitename status mySite`

이 명령으로 다음과 같은 출력이 작성됩니다.

```
SiteName Status:
-----
SiteName ..... mySite
WeightBound ..... 20
TTL ..... 5
StickyTime ..... 0
Number of Servers ..... 1
Proportion given to CpuLoad ..... 49
Proportion given to MemLoad ..... 50
Proportion given to Port ..... 1
Proportion given to System metric .. 0
Advisor running on port ..... 80
Using Proximity ..... N
```

sscontrol status — 관리자 및 어드바이저가 실행 여부 표시

▶▶—sscontrol—status—◀◀

예제

- 실행 중인 내용을 보려면 다음 명령을 실행하십시오.

```
sscontrol status
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
      NameServer has been started.  
      Manager has been started.
```

```
-----  
| ADVISOR | SITENAME:PORT | TIMEOUT |  
-----  
|  http  |           80  | unlimited |  
-----
```

제 29 장 Cisco CSS Controller 명령어 참조서

이 장에서는 Cisco CSS Controller에서 다음 **cococontrol** 명령을 사용하는 방법을 설명합니다.

- 458 페이지의 『cococontrol consultant — 컨설턴트 구성 및 제어』
- 461 페이지의 『cococontrol controller — 제어기 관리』
- 463 페이지의 『cococontrol file — 구성 파일 관리』
- 465 페이지의 『cococontrol help — 이 명령의 도움말 표시 또는 인쇄』
- 466 페이지의 『cococontrol highavailability — 고가용성 제어』
- 470 페이지의 『cococontrol metriccollector — 메트릭 콜렉터 구성』
- 472 페이지의 『cococontrol ownercontent — 소유자 이름 및 콘텐츠 규칙 제어』
- 475 페이지의 『cococontrol service — 서비스 구성』

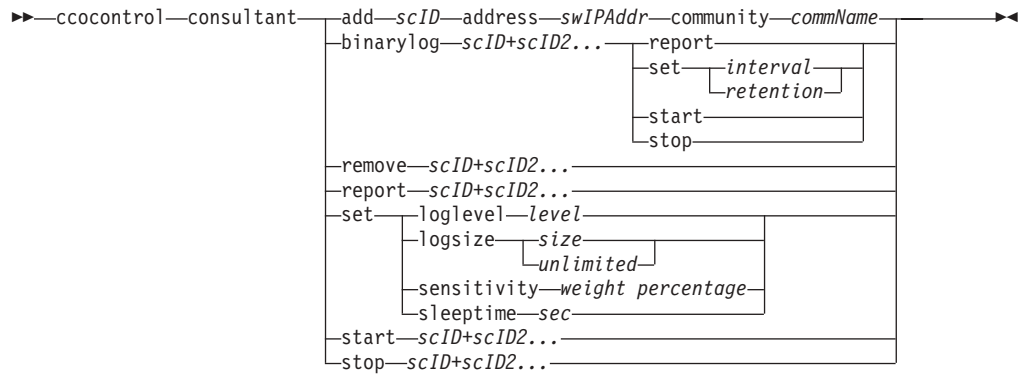
매개변수의 고유한 문자를 입력하여 cococontrol 명령의 축약된 버전을 사용할 수 있습니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면 **cococontrol help file** 대신에 **cococontrol he f**를 입력할 수 있습니다.

cococontrol 명령 프롬프트를 확보하려면 **cococontrol**을 입력하십시오.

명령 인터페이스를 종료하려면 **exit** 또는 **quit** 명령을 입력하십시오.

주: 모든 명령 매개변수 값에는 영어 문자를 사용해야 합니다. 호스트 이름(서버 명령에서 사용)과 파일 이름(파일 명령에 사용)만 예외입니다.

cococontrol consultant — 컨설턴트 구성 및 제어



add

스위치 컨설턴트를 추가합니다.

scID (switchConsultantID)

컨설턴트를 참조하는 사용자 정의 문자열.

address

컨설턴트가 가중치를 제공하는 Cisco CSS Switch의 IP 주소.

swIPAddr (switchIPAddress)

스위치의 IP 주소

community

Cisco CSS Switch에 도달하여 통신을 설정하기 위해 SNMP에 사용되는 이름.

commName

Cisco CSS Switch의 읽기/쓰기 공동체 이름.

binarylog

컨설턴트에 대한 2진 로그를 제어합니다.

report

2진 로그의 특성을 보고합니다.

set

2진 로그에 정보를 기록하는 빈도를 초 단위로 설정합니다. 2진 로그 기능을 사용하면 구성에 정의된 각 서비스에 대한 2진 로그 파일에 서비스 정보를 저장할 수 있습니다. 마지막 기록이 로그에 씌여진 이후 지정된 로그 간격 초가 경과될 때 정보가 로그에 기록됩니다. 기본 2진 로그 간격은 60입니다.

interval

2진 로그의 항목 사이의 시간(초)을 설정합니다.

retention

2진 로그 파일이 보존되는 시간 수를 설정합니다.

start

2진 로그를 시작합니다.

stop

2진 로그를 중지합니다.

remove

스위치 컨설턴트를 제거합니다.

report

스위치 컨설턴트의 특성을 보고합니다.

set

스위치 컨설턴트의 특성을 설정합니다.

loglevel

스위치 컨설턴트가 활동을 기록하는 레벨을 설정합니다. 기본값은 1입니다.

level

레벨 번호는 0 - 5입니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다.

- 0 = 없음
- 1 = 최소
- 2 = 기본
- 3 = 중간
- 4 = 고급
- 5 = 자세히

logsize

로그 파일에 기록되는 최대 바이트 수를 설정합니다. 기본값은 1048576입니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서부터 기록된 후속 항목을 겹쳐씁니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 상위 레벨에서 로그될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

size

컨설턴트 로그에 기록되는 최대 바이트수. 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 항목 크기가 변하므로 겹쳐쓰기 전에는 로그 파일은 정확한 최대 크기에 도달할 수 없습니다.

sensitivity

가중치를 변경하기 위해서 발생해야 하는 이전 가중치와 새 가중치 사이의 변경 양을 표시합니다. 가중치가 변경되려면 새 가중치와 이전 가중치 사이의 차이점이 감도 백분율보다 커야 합니다. 유효한 범위는 0 - 100이고 기본값은 5입니다.

weight percentage

감도 값을 나타내는 0 - 100 사이의 정수입니다.

sleeptime

가중치 설정 주기 사이에 휴면하는 시간(초)을 설정합니다. 기본값은 7입니다.

sec

휴면 시간을 초 단위로 표시하는 정수입니다. 유효한 범위는 0에서 2,147,460 사이입니다.

start

메트릭 수집 및 가중치 설정을 시작합니다.

stop

메트릭 수집 및 가중치 설정을 중지합니다.

예제

- Switch ID가 sc1이고, IP 주소가 9.37.50.17이고, 공동체 이름이 comm1인 스위치 컨설턴트를 추가하려면 다음과 같이 입력하십시오.

```
cococontrol consultant add sc1 address 9.37.50.17 community comm2
```

- 2진 로그를 시작하려면 다음 명령을 수행하십시오.

```
cococontrol consultant binarylog sc1 start
```

- Switch consultant sc1의 특성에 대한 보고서를 보려면 다음과 같이 입력하십시오.

```
cococontrol consultant report sc1
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
Consultant sc1 connected to switch at 9.37.50.1:cn1
Consultant has been started
Sleep time = 7
Sensitivity = 5
Log level = 5
Log size = 1,048,576
ownerContent(s):
ownerContent oc1
```

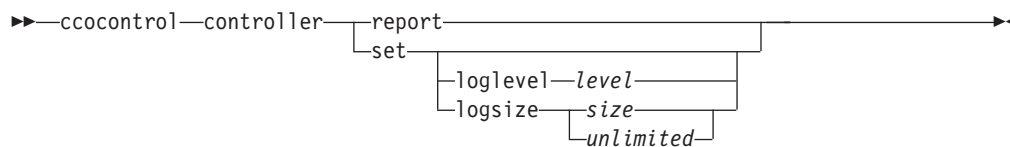
- sc1 switch ID에 대한 가중치 설정 주기 사이의 휴면 시간을 10초로 설정하려면 다음과 같이 입력하십시오.

```
cococontrol consultant set sc1 sleeptime 10
```

- sc1의 consultant ID에 대한 메트릭 수집 및 가중치 설정을 시작하려면 다음과 같이 입력하십시오.

```
cococontrol consultant start sc1
```

cococontrol controller — 제어기 관리



report

제어기의 특성을 표시합니다. 버전 정보는 이 보고서의 일부로서 표시됩니다.

set

제어기의 특성을 설정합니다.

loglevel

제어기가 활동을 기록하는 레벨을 설정합니다. 기본값은 1입니다.

level

레벨 번호는 0 - 5입니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다.

- 0 = 없음
- 1 = 최소
- 2 = 기본
- 3 = 중간
- 4 = 고급
- 5 = 자세히

logsize

로그 파일에 기록되는 최대 바이트 수를 설정합니다. 기본값은 1048576입니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서부터 기록된 후속 항목을 겹쳐씁니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 상위 레벨에서 로그될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

size | unlimited

컨설턴트 로그에 기록되는 최대 바이트수. 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 항목 크기가 변하므로 겹쳐쓰기 전에는 로그 파일은 정확한 최대 크기에 도달할 수 없습니다.

예제

- 제어기에서 보고서를 표시하려면 다음 명령을 실행하십시오.

```
cococontrol controller report
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Controller Report:

Version Version: 05.00.00.00 - 03/21/2002-09:49:57-EST

Logging level 1

Log size. 1048576

Configuration File. . . . config1.xml

Consultants:

Consultant consult1 -Started

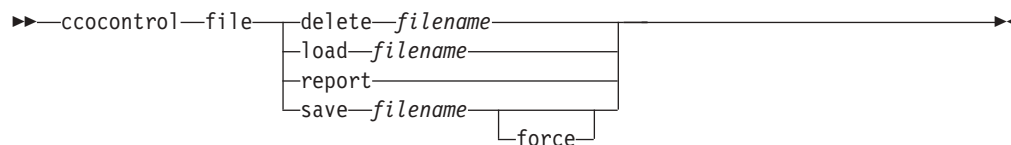
- 더 나은 성능을 위해 로그 레벨을 0으로 설정하려면 다음 명령을 실행하십시오.

cococontrol set loglevel 0

- 제어기 로그 크기를 1,000,000바이트를 설정하려면 다음 명령을 실행하십시오.

cococontrol controller set logsize 1000000

cococontrol file — 구성 파일 관리



delete

지정된 구성 파일을 삭제합니다.

filename

구성 파일. 파일 확장자는 .xml여야 합니다. 이 확장자를 지정하지 않을 경우, 이 확장자로 간주됩니다.

load

지정된 파일에 저장된 구성을 로드합니다.

주: 파일 로드는 해당 파일에 저장된 구성을 실행 중인 구성에 추가합니다. 새 구성을 로드하려면, 파일을 로드하기 전에 서버를 정지시킨 후 재시작해야 합니다.

report

구성 파일을 나열합니다.

save

현재 구성을 지정된 파일에 저장합니다.

주: 파일은 다음의 디렉토리에 저장되어 로드됩니다.

- AIX 시스템: `/opt/ibm/edge/lb/servers/configurations/cco`
- Linux 시스템: `/opt/ibm/edge/lb/servers/configurations/cco`
- Solaris 시스템: `/opt/ibm/edge/lb/servers/configurations/cco`
- Windows 시스템:

설치(기본) 디렉토리: `C:\Program Files\ibm\edge\lb\servers\configurations\cco`

force

기존 파일을 저장합니다.

예제

- file1 파일을 삭제하려면 다음 명령을 실행하십시오.
`cococontrol file delete file1`
- 파일의 구성을 현재 구성에 추가하려면 다음과 명령을 실행하십시오.
`cococontrol file load config2`

- 이전에 저장된 파일의 보고서를 보려면 다음 명령을 실행하십시오.

```
cococontrol file report
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
FILE REPORT:
```

```
-----
```

```
file1.xml
```

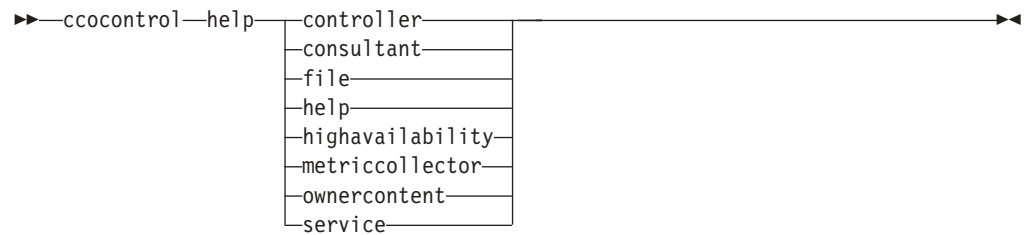
```
file2.xml
```

```
file3.xml
```

- 구성 파일을 config2.xml이라는 파일에 저장하려면 다음 명령을 실행하십시오.

```
cococontrol file save config2
```

cococontrol help — 이 명령의 도움말 표시 또는 인쇄



예제

- cococontrol 명령에 대한 도움말을 보려면 다음 명령을 입력하십시오.

```
cococontrol help
```

이 명령으로 다음과 같은 출력이 작성됩니다.

The following commands are available:

controller	- operate on the controller
consultant	- operate on switch consultants
file	- operate on configuration files
help	- operate on help
highavailability	- operate on high availability
metriccollector	- operate on metric collectors
ownerContent	- operate on ownerContents
service	- operate on services

- 온라인 도움말 구문에는 다음과 같은 기호가 사용됩니다.

< > 중괄호는 매개변수 또는 일련의 문자를 묶습니다.

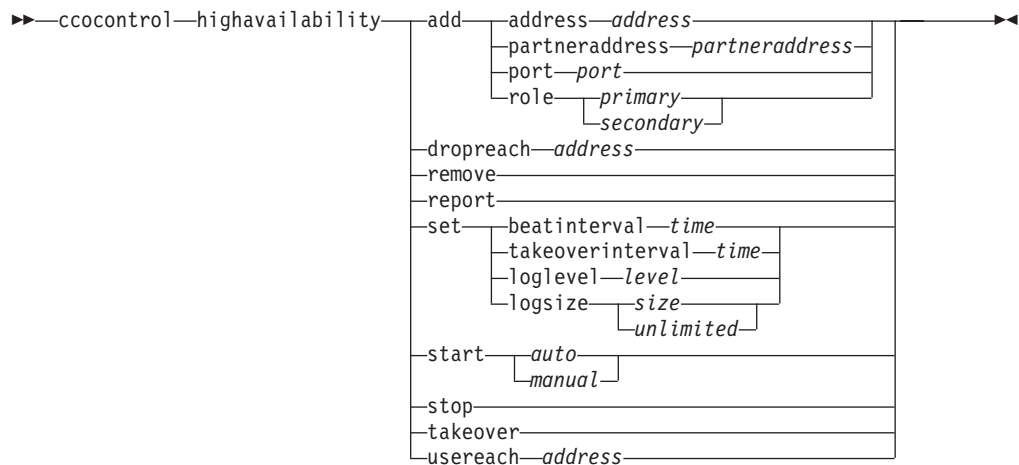
[] 대괄호는 선택적 항목을 묶습니다.

| 수직 막대는 중괄호 및 대괄호 내의 대안을 분리합니다.

:

콜론은 이름 사이의 분리자입니다(**consultant1:ownercontent1**).

cococontrol highavailability — 고가용성 제어



add

고가용성 노드, 상대 및 도달 목표를 구성합니다.

address

하트 비트를 받을 주소입니다.

address

고가용성 노드의 IP 주소입니다.

partneraddress

하트 비트를 보낼 주소입니다. 이 주소는 상대 노드에 대해 구성된 IP 주소 또는 호스트 이름입니다. 이 주소는 상대 고가용성 시스템과의 통신에 사용됩니다.

address

상대의 IP 주소입니다.

port

상대와의 통신에 사용되는 포트입니다. 기본값은 12345입니다.

port

포트 번호.

role

고가용성 역할.

primary | secondary

기본 또는 2차 역할.

dropreach

고가용성 기준에서 이 도달 목표를 제거합니다.

address

도달 목표의 IP 주소입니다.

remove

고가용성 구성에서 노드, 상대 및 도달 목표를 제거합니다. 이 명령을 사용하려면 먼저 고가용성을 정지해야 합니다.

report

고가용성 정보를 표시합니다.

set

고가용성의 특성을 설정합니다.

beatinterval

하트 비트가 상대에게 전송되는 빈도를 밀리초 단위로 설정합니다. 기본값은 500입니다.

time

비트 간격을 밀리초로 표시하는 양의 정수입니다.

takeoverinterval

takeover가 발생하기 전에 반드시 경과해야 하는(하트 비트를 받지 않는 동안) 시간의 양을 밀리초 단위로 설정합니다. 기본값은 2000입니다.

time

takeover 간격을 밀리초로 표시하는 양의 정수입니다.

loglevel

활동이 기록되는 레벨을 설정합니다. 기본값은 1입니다.

level

레벨 번호는 0 - 5입니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다.

- 0 = 없음
- 1 = 최소
- 2 = 기본
- 3 = 중간
- 4 = 고급
- 5 = 자세히

logsize

고가용성 로그 파일에 기록되는 최대 바이트 수를 설정합니다. 기본값은 1048576입니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서부터 기록된 후속 항목을 겹쳐씁니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 상위 레벨에서 로그될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

size | unlimited

고가용성 로그에 기록되는 최대 바이트 수. 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 항목 크기가 변하므로 겹쳐쓰기 전에는 로그 파일은 정확한 최대 크기에 도달할 수 없습니다.

start

고가용성을 사용하여 시작합니다. 이 명령을 사용하려면 먼저 고가용성 노드, 상대 및 도달 목표를 구성해야 합니다.

auto | manual

고가용성을 자동 또는 수동 복구 전략으로 시작할지 결정합니다.

stop

고가용성을 사용하여 정지합니다.

takeover

활성 고가용성 노드에서 제어를 인수합니다.

usereach

고가용성을 사용하여 시작할 도달 목표 주소입니다. 고가용성 상대가 대상에 도달 가능한 거리를 판별할 수 있도록 ping할 수 있는 도달 목표를 추가하십시오.

address

도달 목표의 IP 주소입니다.

예제

- IP 주소가 9.37.50.17이고, 포트 12345에 대한 기본 역할을 가지며, 상대 주소가 9.37.50.14인 고가용성 노드를 추가하려면 다음과 같이 입력하십시오.

```
cococontrol highavailability add  
address 9.37.50.17 role primary port 12345 partneraddress 9.37.50.14
```

- 9.37.50.9의 도달 목표를 추가하려면 다음 명령을 실행하십시오.

```
cococontrol highavailability usereach 9.37.50.9
```

- 9.37.50.9의 도달 목표를 제거하려면 다음 명령을 실행하십시오.

```
cococontrol highavailability dropreach 9.37.50.9
```

- 고가용성을 수동 복구로 시작하려면 다음과 같이 입력하십시오.

```
cococontrol highavailability start manual
```

- 고가용성의 통계 스냅샷을 확보하려면 다음 명령을 실행하십시오.

```
cococontrol highavailability report
```

이 명령으로 다음과 같은 출력이 작성됩니다.

High Availability Status:

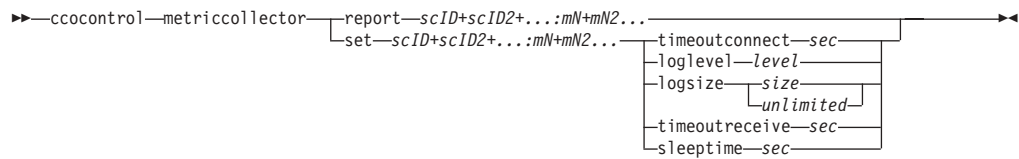
Node primary

Node Address 9.37.50.17

```
Port . . . . . 12345
Partner Address. . . . . 9.37.50.14
Recovery Strategy. . . . . manual
Heartbeat Interval . . . . . 500
Takeover Interval. . . . . 2000
State. . . . . idle
Sub-state. . . . . unsynchronized
```

```
Reachability Status : Node/Partner
-----
No reach targets configured
```

cococontrol metriccollector — 메트릭 콜렉터 구성



report

메트릭 콜렉터의 특성을 표시합니다.

scID(switch consultant ID)

컨설턴트를 참조하는 사용자 정의 문자열.

mN(메트릭 이름)

제공된 또는 조정 메트릭을 식별하는 이름입니다.

set

메트릭 콜렉터의 특성을 설정합니다.

timeoutconnect

연결이 실패했음을 보고하기 전에 메트릭 콜렉터가 대기하는 시간을 설정합니다.

sec

메트릭 콜렉터가 서비스에 대한 연결이 실패했음을 보고하기 전에 대기하는 시간의 양을 초 단위로 표시하는 양의 정수입니다.

loglevel

지정된 컨설턴트가 활동을 기록하는 레벨을 설정합니다. 기본값은 1입니다.

level

레벨의 수. 기본값은 1입니다. 번호가 커질수록 더 자세한 정보가 컨설턴트 로그에 기록됩니다. 가능한 값은 다음과 같습니다.

- 0 = 없음
- 1 = 최소
- 2 = 기본
- 3 = 중간
- 4 = 고급
- 5 = 자세히

logsize

로그 파일에 기록되는 최대 바이트 수를 설정합니다. 기본값은 1048576입니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서부터 기록된 후속 항목을 겹쳐씹습니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록

된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 상위 레벨에서 로그될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

size | unlimited

컨설턴트 로그에 기록되는 최대 바이트수. 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 항목 크기가 변하므로 겹쳐쓰기 전에는 로그 파일은 정확한 최대 크기에 도달할 수 없습니다.

timeoutreceive

서비스로부터의 수신에 실패했음을 보고하기 전에 컨설턴트가 대기하는 시간을 설정합니다.

sec

서비스로부터의 수신에 실패했음을 보고하기 전에 컨설턴트가 대기하는 시간의 양을 초 단위로 표시하는 양의 정수입니다.

sleeptime

메트릭 수집 주기 간에 메트릭 콜렉터가 휴면하는 시간을 초 단위로 설정합니다.

휴면 시간의 초 수를 나타내는 양의 정수.

예제

- 메트릭 콜렉터의 특성에 대한 보고서를 보려면 다음과 같이 입력하십시오.

```
cococontrol metriccollector report sc1:http
```

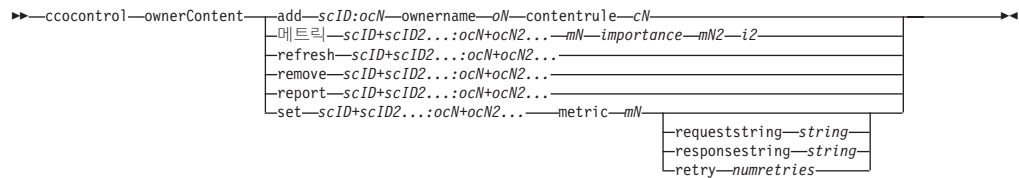
이 명령으로 다음과 같은 출력이 작성됩니다.

```
MetricCollector sc1:http
  collected metric(s).... http
  loglevel..... 5
  logSize..... 1048576
  sleepTimeSeconds..... 7
  timeoutConnectSeconds.. 21
  timeoutReceiveSeconds.. 21
```

- sc1 switch consultant 및 http 메트릭에 대해 timeoutconnect를 15초로, 로그 크기를 제한 없음으로 설정하려면 다음과 같이 입력하십시오.

```
cococontrol metriccollector set sc1:http timeoutconnect 15 logsize unlimited
```

cococontrol ownercontent — 소유자 이름 및 콘텐츠 규칙 제어



add

지정된 컨설턴트에 ownercontent를 추가합니다.

scID(switch consultant ID)

컨설턴트를 나타내는 사용자 정의 문자열.

OCName(ownercontent name)

스위치의 소유자 이름 및 콘텐츠 규칙을 나타내는 사용자 정의 문자열.

ownername

소유자 구성을 식별하는 스위치에 구성된 이름입니다.

oN (ownername)

공백 없는 고유한 텍스트 문자열입니다. ownername은 Cisco switch에 지정된 것과 동일해야 합니다.

contentrule

소유자의 콘텐츠 규칙 구성을 식별하는 스위치에 구성된 이름입니다.

cN (contentname)

공백 없는 고유한 텍스트 문자열입니다. contentname은 Cisco switch에 지정된 것과 동일해야 합니다.

metrics

가중치 및 각 메트릭의 중요도의 계산에 사용된 메트릭 세트를 지정합니다. 중요도는 총계에 대한 백분율로 표현됩니다. 중요도 값의 총계는 100이어야 합니다. 메트릭은 연결 데이터 메트릭, 응용프로그램 어드바이저 메트릭 및 Metric Server 메트릭의 임의의 조합이 될 수 있습니다. 기본값은 활성 연결(activeconn) 및 연결 비율(connrate) 메트릭이며 중요도는 50/50입니다.

mN (metricname)

서버의 가중치를 판별하는 조치를 수집할 메트릭 콜렉터를 식별하는 이름.

다음은 유효한 메트릭 이름 및 연관된 포트 목록입니다.

어드바이저 이름	프로토콜	포트
connect	ICMP	12345
db2	개인용	50000
dns	DNS	53
ftp	FTP	21

어드바이저 이름	프로토콜	포트
http	HTTP	80
https	SSL	443
cachingproxy	HTTP(Caching Proxy를 통한)	80
imap	IMAP	143
ldap	LDAP	389
nntp	NNTP	119
ping	PING	0
pop3	POP3	110
sip	SIP	5060
smtp	SMTP	25
ssl	SSL	443
telnet	Telnet	23
WLM	개인용	10,007
activeconn	n/a	n/a
connrate	n/a	n/a
cpuload	n/a	n/a
memload	n/a	n/a

importance

서버 가중치를 계산할 때 이 메트릭의 중요도를 나타내는 0에서 100까지의 숫자입니다.

refresh

Cisco CSS Switch의 구성으로 구성된 서비스를 새로 고칩니다.

remove

ownercontent를 제거합니다.

report

ownercontents의 특성을 보고합니다.

set

ownercontents의 특성을 설정합니다.

metric

메트릭의 특성을 설정합니다.

mN

원하는 메트릭의 이름입니다.

requeststring

지정된 메트릭에 대한 요청 문자열을 설정합니다. 이는 메트릭 정보를 집적하기 위해 메트릭 콜렉터가 보낸 요청을 나타냅니다.

string

메트릭 콜렉터가 서버로 전송하는 요청 문자열.

responsestring

지정된 메트릭에 대한 응답 문자열을 설정합니다. 지정된 응답 문자열은 메트릭 콜렉터가 서버로부터 수신한 응답과 비교하여 서버의 사용가능성을 결정하는 데 사용됩니다.

string

메트릭 콜렉터가 수신된 서버 응답과 비교하는 응답 문자열.

retry

retry는 서버를 종료된 서버로 표시하기 전에 수행할 수 있는 재시도 횟수를 설정합니다.

numretries

0 이상의 정수. 이 값은 3을 초과하면 안 됩니다. retry 키워드를 구성하지 않을 경우, 재시도 횟수는 기본적으로 0입니다.

예제

- 이름이 oc1인 ownerContent를(소유자 이름이 owner1이고 콘텐츠 이름이 content1) ID가 sc1인 스위치 컨설턴트에 추가하려면 다음을 입력하십시오.

```
cococontrol ownerContent add sc1:oc1 ownername owner1 contentrule content1
```

- activeconn 및 http 메트릭에 각기 50의 비례를 지정하려면 다음과 같이 입력하십시오.

```
cococontrol ownerContent metrics sc1:oc1 activeconn 50 http 50
```

- ownercontents의 특성 보고서를 보려면 다음과 같이 입력하십시오.

```
cococontrol ownerContent report sc1:oc1
```

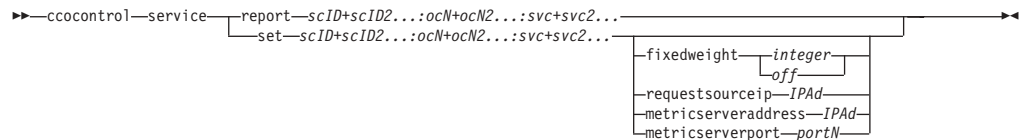
이 명령으로 다음과 같은 출력이 작성됩니다.

```
ownerContent sc1:oc1
  Weightbound = 10
  Metric activeconn has proportion 25
    ResponseString... n/a
    RequestString.... n/a
  Metric http has proportion 50
    ResponseString... n/a
    RequestString.... n/a
  Metric connrate has proportion 25
    ResponseString... n/a
    RequestString.... n/a
  Contains Service t3
  Contains Service t2
  Contains Service t1
```

- HTTP 요청 문자열을 설정하려면 다음과 같이 입력하십시오.

```
cococontrol ownerContent set sc1:oc1 metric http requeststring getCookie
```


cococontrol service — 서비스 구성



report

서비스의 특성을 표시합니다.

scID(switch consultant ID)

컨설턴트를 나타내는 사용자 정의 문자열.

OCName(ownercontent name)

스위치의 소유자 이름 및 콘텐츠 규칙을 나타내는 사용자 정의 문자열.

svc(서비스)

서비스를 나타내는 스위치의 사용자 정의 문자열.

set

서비스의 특성을 설정합니다.

fixedweight

이 서비스에 대한 고정 가중치를 설정합니다. 기본값은 off입니다.

integer | off

이 서비스에 대한 고정 가중치를 나타내는 0 - 10 범위의 양의 정수 또는 고정 가중치가 없음을 나타내는 단어 **off**입니다.

requestsourceip

응용프로그램 요청을 위해 서비스와 접속하려는 주소를 설정합니다.

IPAd(IP 주소)

기호 이름 또는 IP 주소 형식으로, 서비스에 접속하려는 IP 주소

metricserveraddress

Metric Server 요청을 위해 서비스와 접속하려는 주소를 설정합니다.

IPAd(IP 주소)

기호 이름 또는 IP 주소 형식으로 된 Metric Server의 IP 주소

metricserverport

Metric Server 접속에 사용할 포트를 설정합니다.

portN(포트 번호)

Metric Server 접속에 사용되는 포트 번호입니다.

예제

- sc1 consultant에 대한 서비스 t1에 관한 보고서를 표시하려면 다음과 같이 입력하십시오.

```
cococontrol service report scl:ocl:t1
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
Service scl:ocl:ta has weight 10
Fixed weight is off
Request Source Ip..... 9.27.24.156
Application port..... 80
MetricServer address.. 1.0.0.1
MetricServer port..... 10004
Metric activeconn has value -99
Metric http has value -99
Metric connrate has value -99
```

- 서비스 t2에 대한 Metric Server 주소를 설정하려면 다음과 같이 입력하십시오.

```
cococontrol service set scl:ocl:t2 metricserveraddress 9.37.50.17
```

제 30 장 Nortel Alteon Controller 명령어 참조서

이 장에서는 Nortel Alteon Controller에서 다음 **nalcontrol** 명령을 사용하는 방법을 설명합니다.

- 478 페이지의 『nalcontrol consultant — 컨설턴트 구성 및 제어』
- 482 페이지의 『nalcontrol controller — 제어기 관리』
- 484 페이지의 『nalcontrol file — 구성 파일 관리』
- 486 페이지의 『nalcontrol help — 이 명령의 도움말 표시 또는 인쇄』
- 487 페이지의 『nalcontrol highavailability — 고가용성 제어』
- 491 페이지의 『nalcontrol metriccollector — 메트릭 콜렉터 구성』
- 495 페이지의 『nalcontrol service — 서비스 구성』
- 493 페이지의 『nalcontrol server — 서버 구성』

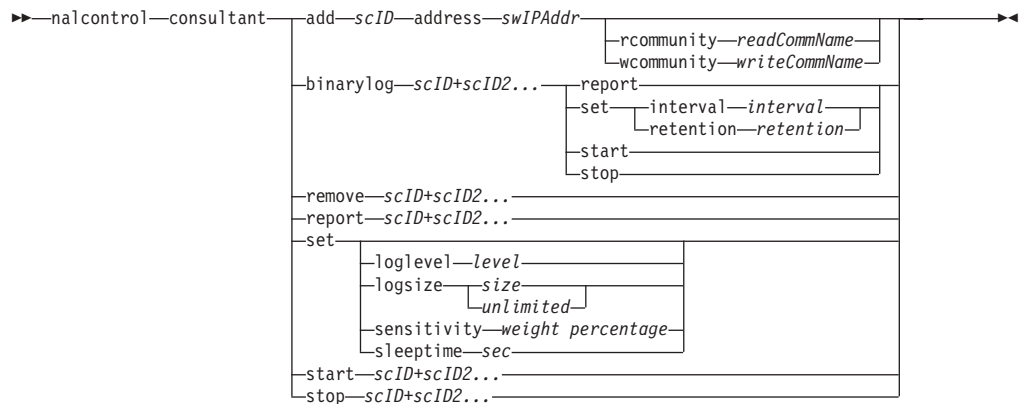
매개변수의 고유한 문자를 입력하여 **nalcontrol** 명령의 축약된 버전을 사용할 수 있습니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면 **nalcontrol help file** 대신에 **nalcontrol he f**를 입력할 수 있습니다.

nalcontrol 명령 프롬프트를 확보하려면 **nalcontrol**을 입력하십시오.

명령 인터페이스를 종료하려면 **exit** 또는 **quit** 명령을 입력하십시오.

주: 모든 명령 매개변수 값에는 영어 문자를 사용해야 합니다. 호스트 이름(서버 명령에서 사용)과 파일 이름(파일 명령에 사용)만 예외입니다.

nalcontrol consultant — 컨설턴트 구성 및 제어



add

스위치 컨설턴트를 추가합니다.

scID

컨설턴트를 참조하는 사용자 정의 문자열.

address

컨설턴트가 가중치를 제공하는 Nortel Alteon Web Switch의 IP 주소.

swIPAddr

스위치의 IP 주소

rcommunity

Nortel Alteon Web Switch와의 SNMP get 통신에 사용되는 읽기 공동체 이름입니다. 기본값은 public입니다.

readCommName

Nortel Alteon Web Switch에 구성되어 있는 대로 읽기 공동체 이름을 나타내는 문자열. 기본값은 public입니다.

wcommunity

SNMP set 통신에 사용되는 기록 공동체 이름입니다.

writeCommName

Nortel Alteon Web Switch에 구성되어 있는 대로 쓰기 공동체 이름을 나타내는 문자열. 기본값은 private입니다.

binarylog

컨설턴트에 대한 2진 로그를 제어합니다.

report

2진 로그의 특성을 보고합니다.

set

2진 로그에 정보를 기록하는 빈도를 초 단위로 설정합니다. 2진 로그 기능을 사용

하면 구성에 정의된 각 서비스에 대한 2진 로그 파일에 서비스 정보를 저장할 수 있습니다. 마지막 기록이 로그에 씌여진 이후 지정된 로그 간격 초과 경과될 때 정보가 로그에 기록됩니다. 기본 2진 로그 간격은 60입니다.

interval

2진 로그 항목 사이의 시간(초)을 설정합니다.

retention

2진 로그 파일이 보존되는 시간 수를 설정합니다.

start

2진 로그를 시작합니다.

stop

2진 로그를 중지합니다.

remove

스위치 컨설턴트를 제거합니다.

report

스위치 컨설턴트의 특성을 보고합니다.

set

스위치 컨설턴트의 특성을 설정합니다.

loglevel

스위치 컨설턴트가 활동을 기록하는 레벨을 설정합니다. 기본값은 1입니다.

level

레벨 번호는 0 - 5입니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다.

- 0 = 없음
- 1 = 최소
- 2 = 기본
- 3 = 중간
- 4 = 고급
- 5 = 자세히

logsize

로그 파일에 기록되는 최대 바이트 수를 설정합니다. 기본값은 1048576입니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서부터 기록된 후속 항목을 겹쳐씹니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 상위 레벨에서 로그될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

size

컨설턴트 로그에 기록되는 최대 바이트수. 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 항목 크기가 변하므로 겹쳐쓰기 전에는 로그 파일은 정확한 최대 크기에 도달할 수 없습니다.

sensitivity

가중치를 변경하기 위해서 발생해야 하는 이전 가중치와 새 가중치 사이의 변경 양을 표시합니다. 가중치가 변경되려면 새 가중치와 이전 가중치 사이의 차이점이 감도 백분율보다 커야 합니다. 유효한 범위는 0 - 100이고 기본값은 5입니다.

weight percentage

감도 값을 나타내는 0 - 100 사이의 정수입니다.

sleeptime

가중치 설정 주기 사이에 휴면하는 시간(초)을 설정합니다. 기본값은 7입니다.

seconds

휴면 시간을 초 단위로 표시하는 정수입니다. 유효한 범위는 0에서 2,147,460 사이입니다.

start

메트릭 수집 및 가중치 설정을 시작합니다.

stop

메트릭 수집 및 가중치 설정을 중지합니다.

예제

- Switch ID가 sc1이고, IP 주소가 9.37.50.17인 스위치 컨설턴트를 추가하려면 다음과 같이 입력하십시오.

```
nalcontrol consultant add sc1 address 9.37.50.17
```

- 2진 로그를 시작하려면 다음 명령을 수행하십시오.

```
nalcontrol consultant binarylog sc1 start
```

- Switch consultant sc1의 특성에 대한 보고서를 보려면 다음과 같이 입력하십시오.

```
nalcontrol consultant report sc1
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
Consultant ID:  sc1  Switch IP addr:  9.37.50.1
Read Community: public
Write Community: private
    Consultant has been started
    Sleep time   = 7
    Sensitivity  = 5
    Log level    = 5
    log size     = 1,048,576
    Service(s):
        Service svc1
```

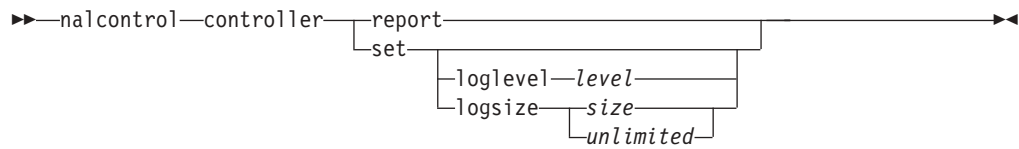
- sc1 switch ID에 대한 가중치 설정 주기 사이의 휴면 시간을 10초로 설정하려면 다음과 같이 입력하십시오.

```
nalcontrol consultant set sc1 sleeptime 10
```

- sc1의 consultant ID에 대한 메트릭 수집 및 가중치 설정을 시작하려면 다음과 같이 입력하십시오.

```
nalcontrol consultant start sc1
```

nalcontrol controller — 제어기 관리



report

제어기의 특성을 표시합니다. 버전 정보는 이 보고서의 일부로서 표시됩니다.

set

제어기의 특성을 설정합니다.

loglevel

제어기가 활동을 기록하는 레벨을 설정합니다. 기본값은 1입니다.

level

레벨 번호는 0 - 5입니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다.

- 0 = 없음
- 1 = 최소
- 2 = 기본
- 3 = 중간
- 4 = 고급
- 5 = 자세히

logsize

로그 파일에 기록되는 최대 바이트 수를 설정합니다. 기본값은 1048576입니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서부터 기록된 후속 항목을 겹쳐씁니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 상위 레벨에서 로그될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

size | unlimited

컨설턴트 로그에 기록되는 최대 바이트수. 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 항목 크기가 변하므로 겹쳐쓰기 전에는 로그 파일은 정확한 최대 크기에 도달할 수 없습니다.

예제

- 제어기에서 보고서를 표시하려면 다음 명령을 실행하십시오.

```
nalcontrol controller report
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Controller Report:

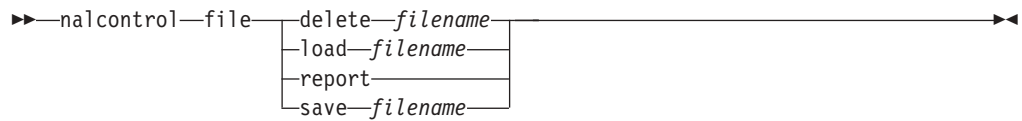
Version Version: 05.00.00.00 - 03/21/2002-09:49:57-EST
Logging level 1
Log size. 1048576
Configuration File. . . . config1.xml

Consultants:

Consultant consult1 -Started

- 더 나은 성능을 위해 로그 레벨을 0으로 설정하려면 다음 명령을 실행하십시오.
nalcontrol set loglevel 0
- 제어기 로그 크기를 1,000,000 바이트를 설정하려면 다음 명령을 실행하십시오.
nalcontrol controller set logsize 1000000

nalcontrol file — 구성 파일 관리



delete

지정된 구성 파일을 삭제합니다.

filename

구성 파일. 파일 확장자는 .xml여야 합니다. 이 확장자를 지정하지 않을 경우, 이 확장자로 간주됩니다.

load

지정된 파일에 저장된 구성을 로드합니다.

주: 파일 로드는 해당 파일에 저장된 구성을 실행 중인 구성에 추가합니다. 새 구성을 로드하려면, 파일을 로드하기 전에 서버를 정지시킨 후 재시작해야 합니다.

report

구성 파일을 나열합니다.

save

현재 구성을 지정된 파일에 저장합니다.

주: 파일은 다음의 디렉토리에 저장되어 로드됩니다.

- AIX 시스템: /opt/ibm/edge/lb/servers/configurations/nal
- Linux 시스템: /opt/ibm/edge/lb/servers/configurations/nal
- Solaris 시스템: /opt/ibm/edge/lb/servers/configurations/nal
- Windows 시스템:

일반 설치 디렉토리 경로 —

C:\Program Files\ibm\edge\lb\servers\configurations\nal

기본 설치 디렉토리 경로 —

C:\Program Files\ibm\lb\servers\configurations\nal

예제

- file1 파일을 삭제하려면 다음 명령을 실행하십시오.
`nalcontrol file delete file1`
- 새 구성 파일을 로드하여 현재 구성을 대체하려면 다음 명령을 실행하십시오.
`nalcontrol file load config2`
- 이전에 저장된 파일의 보고서를 보려면 다음 명령을 실행하십시오.

```
nalcontrol file report
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
FILE REPORT:
```

```
-----
```

```
file1.xml
```

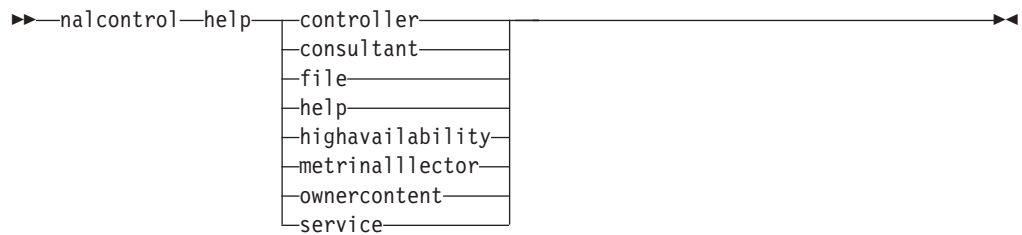
```
file2.xml
```

```
file3.xml
```

- 구성 파일을 config2 파일에 저장하려면 다음 명령을 실행하십시오.

```
nalcontrol file save config2
```

nalcontrol help — 이 명령의 도움말 표시 또는 인쇄



예제

- nalcontrol 명령에 대한 도움말을 보려면 다음 명령을 입력하십시오.

```
nalcontrol help
```

이 명령으로 다음과 같은 출력이 작성됩니다.

The following commands are available:

controller	- operate on the controller
consultant	- operate on switch consultants
file	- operate on configuration files
help	- operate on help
highavailability	- operate on high availability
metriccollector	- operate on metric collectors
server	- operate on servers
service	- operate on services

- 온라인 도움말 구문에는 다음과 같은 기호가 사용됩니다.

< > 중괄호는 매개변수 또는 일련의 문자를 묶습니다.

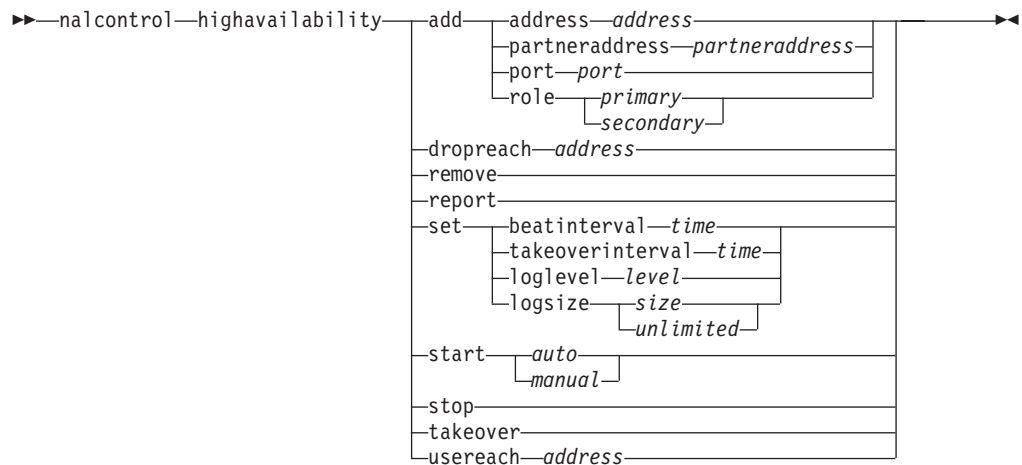
[] 대괄호는 선택적 항목을 묶습니다.

| 수직 막대는 중괄호 및 대괄호 내의 대안을 분리합니다.

:

콜론은 이름 사이의 분리자입니다(예: **consultant1:service1**).

nalcontrol highavailability — 고가용성 제어



add

고가용성 노드, 상대 및 도달 목표를 구성합니다.

address

하트 비트를 받을 주소입니다.

address

고가용성 노드의 IP 주소입니다.

partneraddress

하트 비트를 보낼 주소입니다. 이 주소는 상대 노드에 대해 구성된 IP 주소 또는 호스트 이름입니다. 이 주소는 상대 고가용성 시스템과의 통신에 사용됩니다.

address

상대의 IP 주소입니다.

port

상대와의 통신에 사용되는 포트입니다. 기본값은 12345입니다.

port

포트 번호.

role

고가용성 역할.

primary | secondary

기본 또는 2차 역할.

dropreach

고가용성 기준에서 이 도달 목표를 제거합니다.

address

도달 목표의 IP 주소입니다.

remove

고가용성 구성에서 노드, 상대 및 도달 목표를 제거합니다. 이 명령을 사용하려면 먼저 고가용성을 정지해야 합니다.

report

고가용성 정보를 표시합니다.

set

고가용성의 특성을 설정합니다.

beatinterval

하트 비트가 상대에게 전송되는 빈도를 밀리초 단위로 설정합니다. 기본값은 500입니다.

time

비트 간격을 밀리초로 표시하는 양의 정수입니다.

takeoverinterval

takeover가 발생하기 전에 반드시 경과해야 하는(하트 비트를 받지 않는 동안) 시간의 양을 밀리초 단위로 설정합니다. 기본값은 2000입니다.

time

takeover 간격을 밀리초로 표시하는 양의 정수입니다.

loglevel

활동이 기록되는 레벨을 설정합니다. 기본값은 1입니다.

level

레벨 번호는 0 - 5입니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다.

- 0 = 없음
- 1 = 최소
- 2 = 기본
- 3 = 중간
- 4 = 고급
- 5 = 자세히

logsize

고가용성 로그 파일에 기록되는 최대 바이트 수를 설정합니다. 기본값은 1048576입니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서부터 기록된 후속 항목을 겹쳐씹습니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 상위 레벨에서 로그될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

size | unlimited

고가용성 로그에 기록되는 최대 바이트 수. 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 항목 크기가 변하므로 겹쳐쓰기 전에는 로그 파일은 정확한 최대 크기에 도달할 수 없습니다.

start

고가용성을 사용하여 시작합니다. 이 명령을 사용하려면 먼저 고가용성 노드, 상대 및 도달 목표를 구성해야 합니다.

auto | manual

고가용성을 자동 또는 수동 복구 전략으로 시작할지 결정합니다.

stop

고가용성을 사용하여 정지합니다.

takeover

활성 고가용성 노드에서 제어를 인수합니다.

usereach

고가용성을 사용하여 시작될 도달 목표 주소입니다. 고가용성 상대가 대상에 도달 가능한 거리를 판별할 수 있도록 ping할 수 있는 도달 목표를 추가하십시오.

address

도달 목표의 IP 주소입니다.

예제

- IP 주소가 9.37.50.17이고 포트 12345에 대한 기본 역할을 가지며, 상대 주소가 9.37.50.14인 고가용성 노드를 추가하려면 다음과 같이 입력하십시오.

```
nalcontrol highavailability add  
address 9.37.50.17 role primary port 12345 partneraddress 9.37.50.14
```

- 9.37.50.9의 도달 목표를 추가하려면 다음 명령을 실행하십시오.

```
nalcontrol highavailability usereach 9.37.50.9
```

- 9.37.50.9의 도달 목표를 제거하려면 다음 명령을 실행하십시오.

```
nalcontrol highavailability dropreach 9.37.50.9
```

- 고가용성을 수동 복구로 시작하려면 다음과 같이 입력하십시오.

```
nalcontrol highavailability start manual
```

- 고가용성의 통계 스냅샷을 확보하려면 다음 명령을 실행하십시오.

```
nalcontrol highavailability report
```

이 명령으로 다음과 같은 출력이 작성됩니다.

High Availability Status:

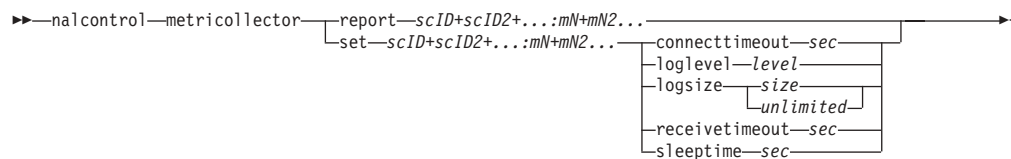
Node primary

Node Address 9.37.50.17

Port 12345
Partner Address. 9.37.50.14
Recovery Strategy. manual
Heartbeat Interval 500
Takeover Interval. 2000
Started. N
State. idle
Sub-state. unsynchronized

Reachability Status : Node/Partner

nalcontrol metriccollector — 메트릭 콜렉터 구성



report

메트릭 콜렉터의 특성을 표시합니다.

scID(switch consultant ID)

컨설턴트를 참조하는 사용자 정의 문자열.

mN(메트릭 이름)

제공된 또는 조정 메트릭을 식별하는 이름입니다.

set

메트릭 콜렉터의 특성을 설정합니다.

connecttimeout

연결이 실패했음을 보고하기 전에 메트릭 콜렉터가 대기하는 시간을 설정합니다.

sec

메트릭 콜렉터가 서비스에 대한 연결이 실패했음을 보고하기 전에 대기하는 시간의 양을 초 단위로 표시하는 양의 정수입니다.

loglevel

지정된 컨설턴트가 활동을 기록하는 레벨을 설정합니다. 기본값은 1입니다.

level

레벨의 수. 기본값은 1입니다. 번호가 커질수록 더 자세한 정보가 컨설턴트 로그에 기록됩니다. 가능한 값은 다음과 같습니다.

- 0 = 없음
- 1 = 최소
- 2 = 기본
- 3 = 중간
- 4 = 고급
- 5 = 자세히

logsize

로그 파일에 기록되는 최대 바이트 수를 설정합니다. 기본값은 1048576입니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 파일의 맨 위에서부터 기록된 후속 항목을 겹쳐씹니다. 로그 크기는 현재의 로그 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록

된 순서를 알 수 있도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 상위 레벨에서 로그될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

size | unlimited

컨설턴트 로그에 기록되는 최대 바이트수. 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 항목 크기가 변하므로 겹쳐쓰기 전에는 로그 파일은 정확한 최대 크기에 도달할 수 없습니다.

receivetimeout

서비스로부터의 수신에 실패했음을 보고하기 전에 컨설턴트가 대기하는 시간을 설정합니다.

sec

서비스로부터의 수신에 실패했음을 보고하기 전에 컨설턴트가 대기하는 시간의 양을 초 단위로 표시하는 양의 정수입니다.

sleeptime

메트릭 수집 주기 간에 메트릭 콜렉터가 휴면하는 시간을 초 단위로 설정합니다.

sec

휴면 시간의 초 수를 나타내는 양의 정수.

예제

- 메트릭 콜렉터의 특성에 대한 보고서를 보려면 다음과 같이 입력하십시오.

```
nalcontrol metrinallector report sc1:http
```

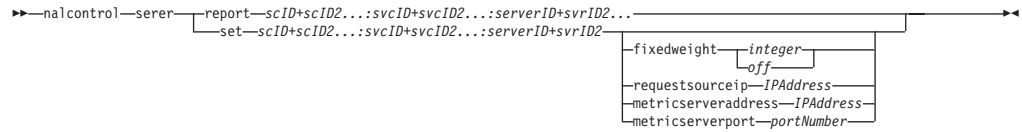
이 명령으로 다음과 같은 출력이 작성됩니다.

```
Metrinallector sc1:http
  collected metric(s).... http
  loglevel..... 5
  logSize..... 1048576
  sleepTimeSeconds..... 7
  timeoutConnectSeconds.. 21
  timeoutReceiveSeconds.. 21
```

- sc1 switch consultant 및 http 메트릭에 대해 connecttimeout을 15초로, 로그 크기를 제한 없음으로 설정하려면 다음과 같이 입력하십시오.

```
nalcontrol metrinallector set sc1:http connecttimeout 15 logsize unlimited
```

nalcontrol server — 서버 구성



report

서버의 특성을 표시합니다.

scID

컨설턴트를 나타내는 사용자 정의 문자열.

svcID

스위치의 가상 서비스 ID 및 가상 포트 번호를 나타내는 사용자 정의 문자열.

serverID

스위치의 서버를 나타내는 정수.

set

서버의 특성을 설정합니다.

fixedweight

이 서버에 대한 고정 가중치를 설정합니다. 기본값은 off입니다. 최대 fixedweight는 48입니다.

integer | off

이 서버에 대한 고정 가중치를 나타내는 양의 정수 또는 고정 가중치가 없음을 나타내는 단어 **off**입니다.

requestsourceip

응용프로그램 요청을 위해 서버와 접속하려는 주소를 설정합니다.

IPAddress

기호 이름 또는 IP 주소 형식으로, 서버에 접속하려는 IP 주소

metricserveraddress

Metric Server 요청을 위해 서버와 접속하려는 주소를 설정합니다.

IPAddress

기호 이름 또는 IP 주소 형식으로 된 Metric Server의 IP 주소

metricserverport

Metric Server 접속에 사용할 포트를 설정합니다.

portNumber

Metric Server 접속에 사용되는 포트 번호입니다.

예제

- sc1 consultant에 대한 서버 1에 관한 보고서를 표시하려면 다음과 같이 입력하십시오.

```
nalcontrol server report sc1:svc1:1
```

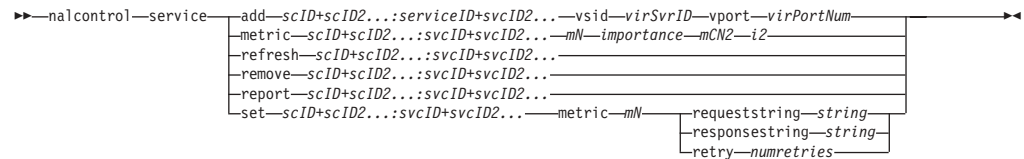
이 명령으로 다음과 같은 출력이 작성됩니다.

```
Server sc1:svc1:1 has weight -99
    Fixed weight is off
    Request Source Ip..... 9.27.24.156
    Application port..... 99
    MetricServer address... 9.99.99.98
    MetricServer port..... 10004
        Metric activeconn has value -99
        Metric connrate has value -99
```

- 서비스 2에 대한 Metric Server 주소를 설정하려면 다음 명령을 실행하십시오.

```
nalcontrol server set sc1:svc1:2 metricserveraddress 9.37.50.17
```

nalcontrol service — 서비스 구성



add

서비스를 지정된 컨설턴트에 추가합니다.

scID (switchConsultantID)

컨설턴트를 참조하는 사용자 정의 문자열.

svcID (serviceID)

서비스를 식별하는 사용자 정의 문자열.

vsid

가상 서비스 식별자 키워드입니다.

virSvrID (virtualServerID)

가상 서버를 표시하는 스위치의 번호입니다.

vport

가상 포트 키워드입니다.

virPortNum (virtualPortNumber)

현재 스위치에 구성된 서비스의 포트 번호.

metrics

가중치 및 각 메트릭의 중요도의 계산에 사용된 메트릭 세트를 지정합니다. 중요도는 총계에 대한 백분율로 표현됩니다. 중요도 값의 총계는 100이어야 합니다. 메트릭은 연결 데이터 메트릭, 응용프로그램 어드바이저 메트릭 및 Metric Server 메트릭의 임의의 조합이 될 수 있습니다. 기본값은 활성 연결(activeconn) 및 연결 비율(connrate) 메트릭이며 중요도는 50/50입니다.

mN(메트릭 이름)

서버의 가중치를 판별하는 조치를 수집할 메트릭 콜렉터를 식별하는 이름.

다음은 유효한 메트릭 이름 및 연관된 포트 목록입니다.

어드바이저 이름	프로토콜	포트
connect	ICMP	12345
db2	개인용	50000
dns	DNS	53
ftp	FTP	21
http	HTTP	80
https	SSL	443
cachingproxy	HTTP(Caching Proxy를 통한)	80

어드바이저 이름	프로토콜	포트
imap	IMAP	143
ldap	LDAP	389
nntp	NNTP	119
ping	PING	0
pop3	POP3	110
sip	SIP	5060
smtp	SMTP	25
ssl	SSL	443
telnet	Telnet	23
WLM	개인용	10,007
activeconn	n/a	n/a
connrate	n/a	n/a
cpuload	n/a	n/a
memload	n/a	n/a

importance

서버 가중치를 계산할 때 이 메트릭의 중요도를 나타내는 0에서 100까지의 숫자입니다.

refresh

Nortel Alteon Web Switch의 정보로 서비스를 새로 고칩니다.

remove

서비스를 제거합니다.

report

서비스의 특성을 보고합니다.

set

서비스의 특성을 설정합니다.

metric

구성된 메트릭의 특성을 설정합니다.

mN(메트릭 이름)

원하는 메트릭의 이름입니다.

requeststring

지정된 메트릭에 대한 요청 문자열을 설정합니다. 이는 메트릭 정보를 집적하기 위해 메트릭 콜렉터가 보낸 요청을 나타냅니다.

string

메트릭 콜렉터가 서버로 전송하는 요청 문자열.

responsestring

지정된 메트릭에 대한 응답 문자열을 설정합니다. 지정된 응답 문자열은 메트릭 콜렉터가 서버로부터 수신한 응답과 비교하여 서버의 사용가능성을 결정하는 데 사용됩니다.

string

메트릭 콜렉터가 수신된 서버 응답과 비교하는 응답 문자열.

retry

retry는 서버를 종료된 서버로 표시하기 전에 수행할 수 있는 재시도 횟수를 설정합니다.

numretries

0 이상의 정수. 이 값은 3을 초과하면 안 됩니다. retry 키워드를 구성하지 않을 경우, 재시도 횟수는 기본적으로 0입니다.

예제

- ID가 sc1인 switch consultant에 이름이 svc1인 서비스를(가상 서버 ID는 1이고 가상 포트는 80) 추가하려면 다음과 같이 입력하십시오.

```
nalcontrol service add sc1:svc1 vsid 1 vport 80
```

- activeconn 및 http 메트릭에 각기 50의 비례를 지정하려면 다음과 같이 입력하십시오.

```
nalcontrol service metrics sc1:svc1 activeconn 50 http 50
```

- ownercontents의 특성 보고서를 보려면 다음과 같이 입력하십시오.

```
nalcontrol service report sc1:svc1
```

이 명령은 다음과 같은 출력을 생성합니다.

```
Service sc1:svc1
  Weightbound = 48
  Metric activeconn has proportion 50
  Metric connrate has rproportion 50
  Contains Server 4
  Contains Server 3
  Contains Server 2
  Contains Server 1
```

- HTTP 요청 문자열을 설정하려면 다음과 같이 입력하십시오.

```
nalcontrol service set sc1:svc1 metric http requeststring getLastErrorCode
```

부록 A. GUI: 일반 명령

Load Balancer GUI(Graphical User Interface)에서 패널의 왼쪽에는 맨 위 레벨에 Load Balancer가 있고 컴포넌트로서 Dispatcher, Content Based Routing(CBR), Site Selector, Cisco CSS Controller 및 Nortel Alteon Controller가 있는 트리 구조가 표시됩니다.

IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, Dispatcher 컴포넌트만이 사용 가능합니다. 자세한 내용은 89 페이지의 제 8 장 『IPv4 및 IPv6용 Load Balancer의 Dispatcher 배치』를 참조하십시오.

여러 컴포넌트를 각기 강조하는 Load Balancer GUI의 그래픽 예에 대해서는, 다음을 참조하십시오.

- Dispatcher에 대해서는 500 페이지의 그림 41의 내용을 참조하십시오.
- CBR에 대해서는 501 페이지의 그림 42의 내용을 참조하십시오.
- Site Selector에 대해서는 502 페이지의 그림 43의 내용을 참조하십시오.
- Cisco CSS Controller에 대해서는 503 페이지의 그림 44의 내용을 참조하십시오.
- Nortel Alteon Controller에 대해서는 504 페이지의 그림 45의 내용을 참조하십시오.

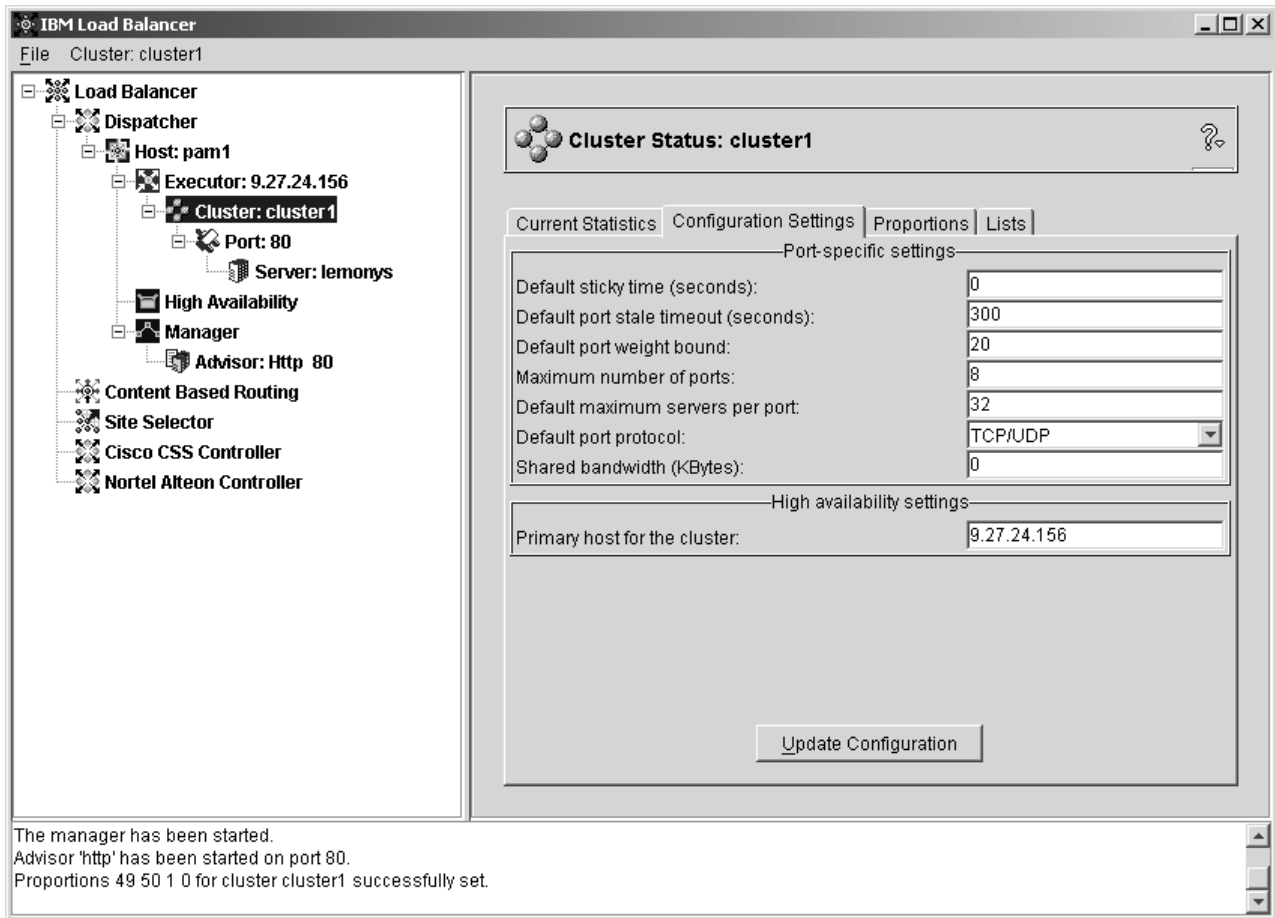


그림 41. GUI(Graphical User Interface)는 Dispatcher 컴포넌트의 GUI 트리 구조 확장을 표시합니다.

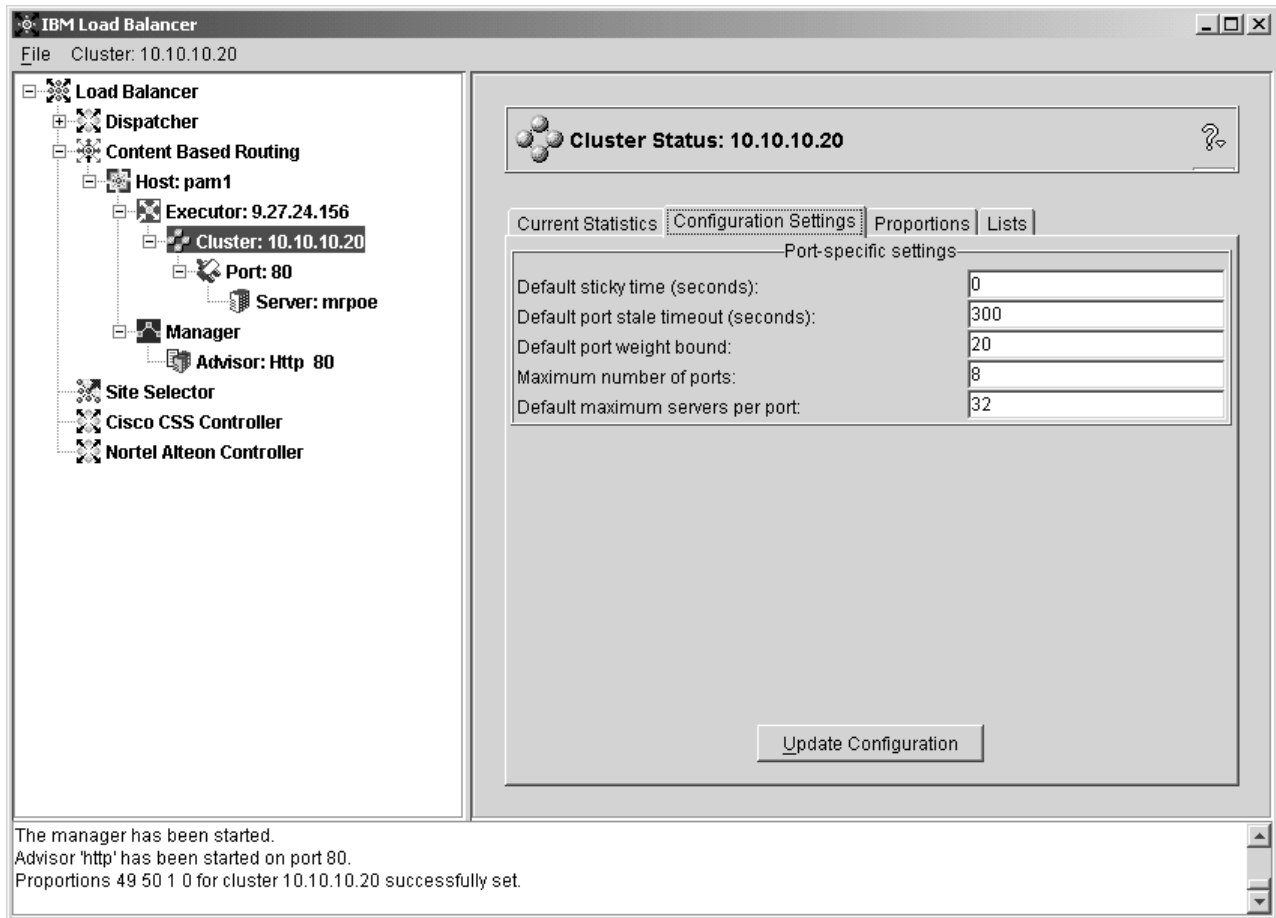


그림 42. GUI(Graphical User Interface)는 CBR 컴포넌트의 GUI 트리 구조 확장을 표시합니다.

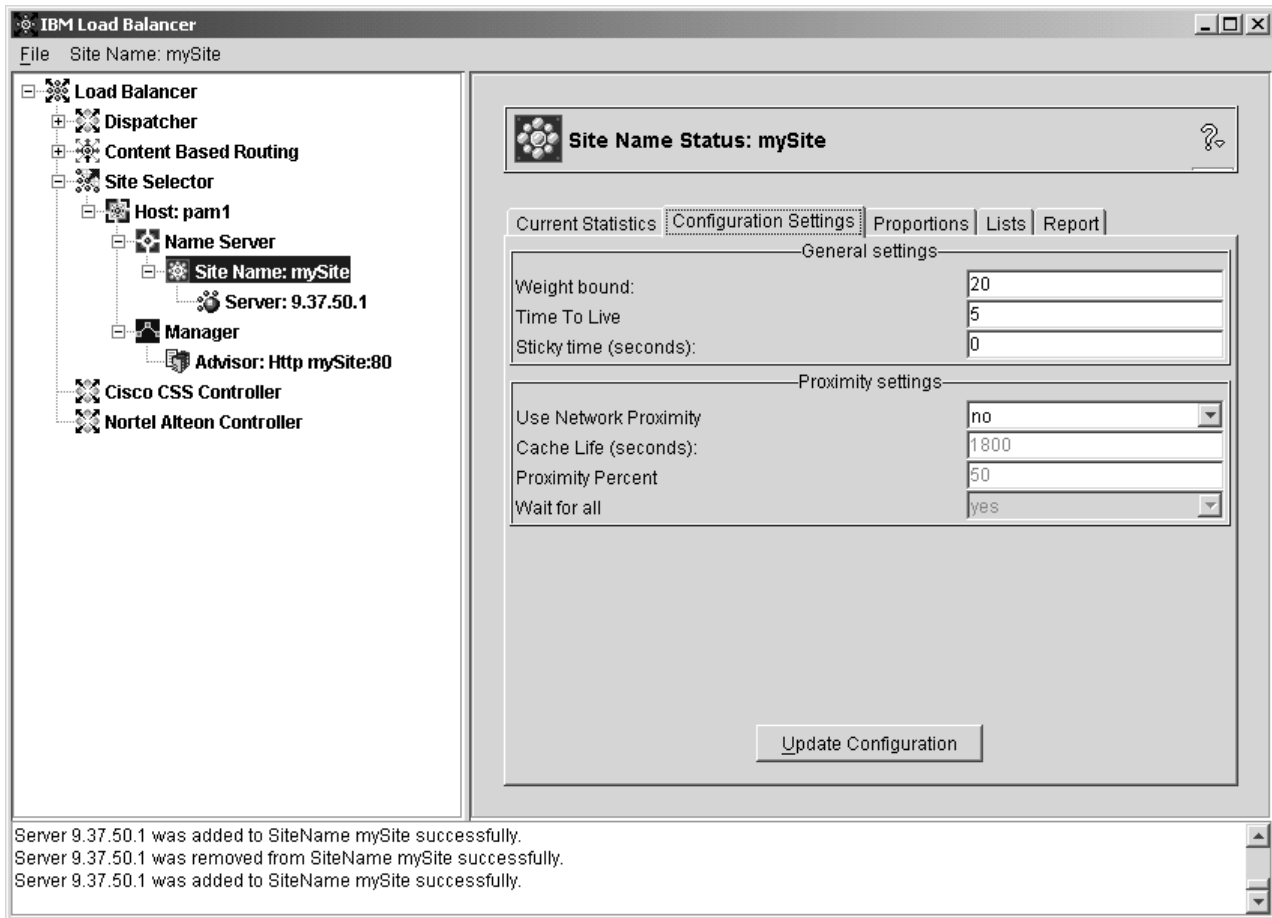


그림 43. GUI(Graphical User Interface)는 Site Selector 컴포넌트의 GUI 트리 구조 확장을 표시합니다.

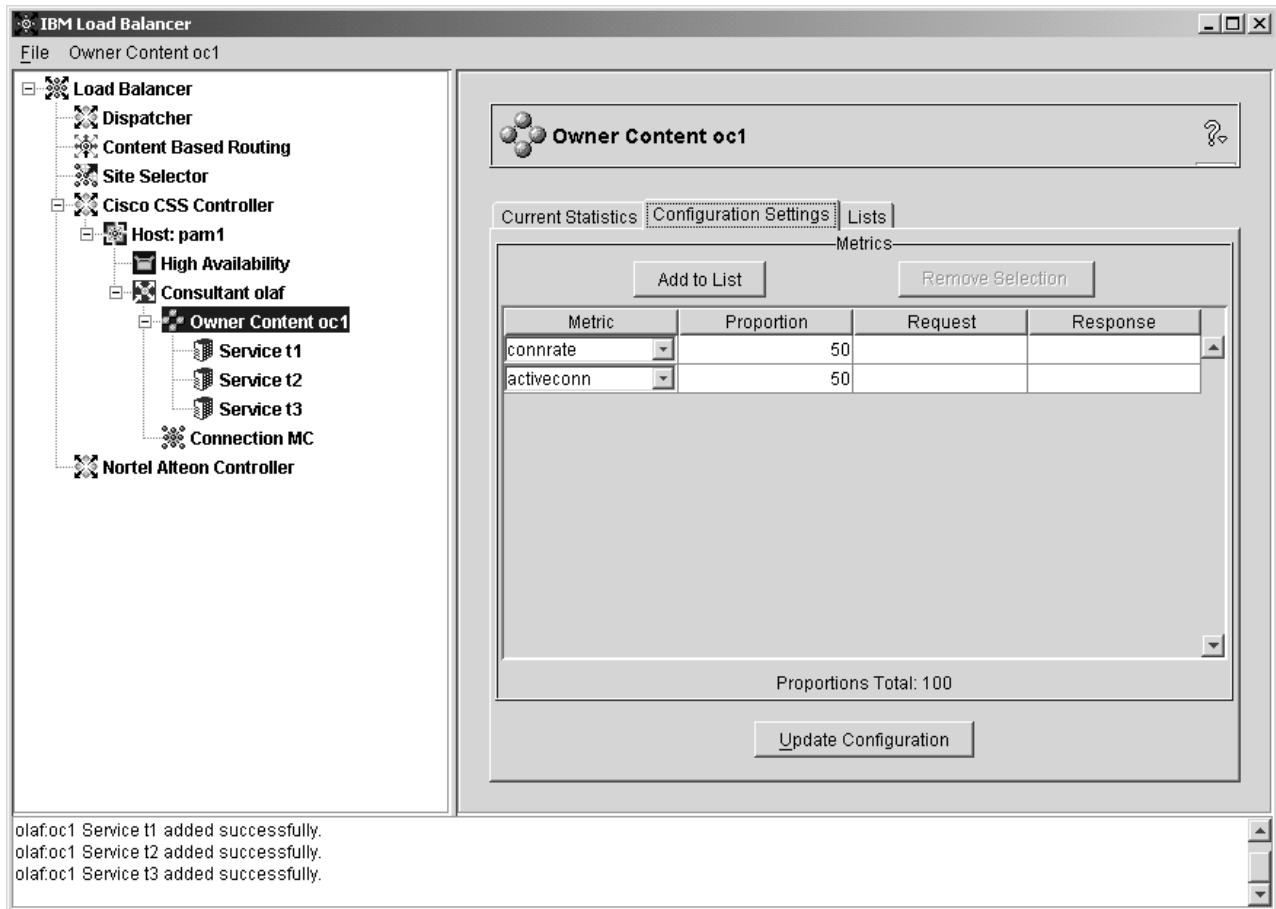


그림 44. GUI(Graphical User Interface)는 Cisco CSS Controller 컴포넌트의 GUI 트리 구조 확장을 표시합니다.

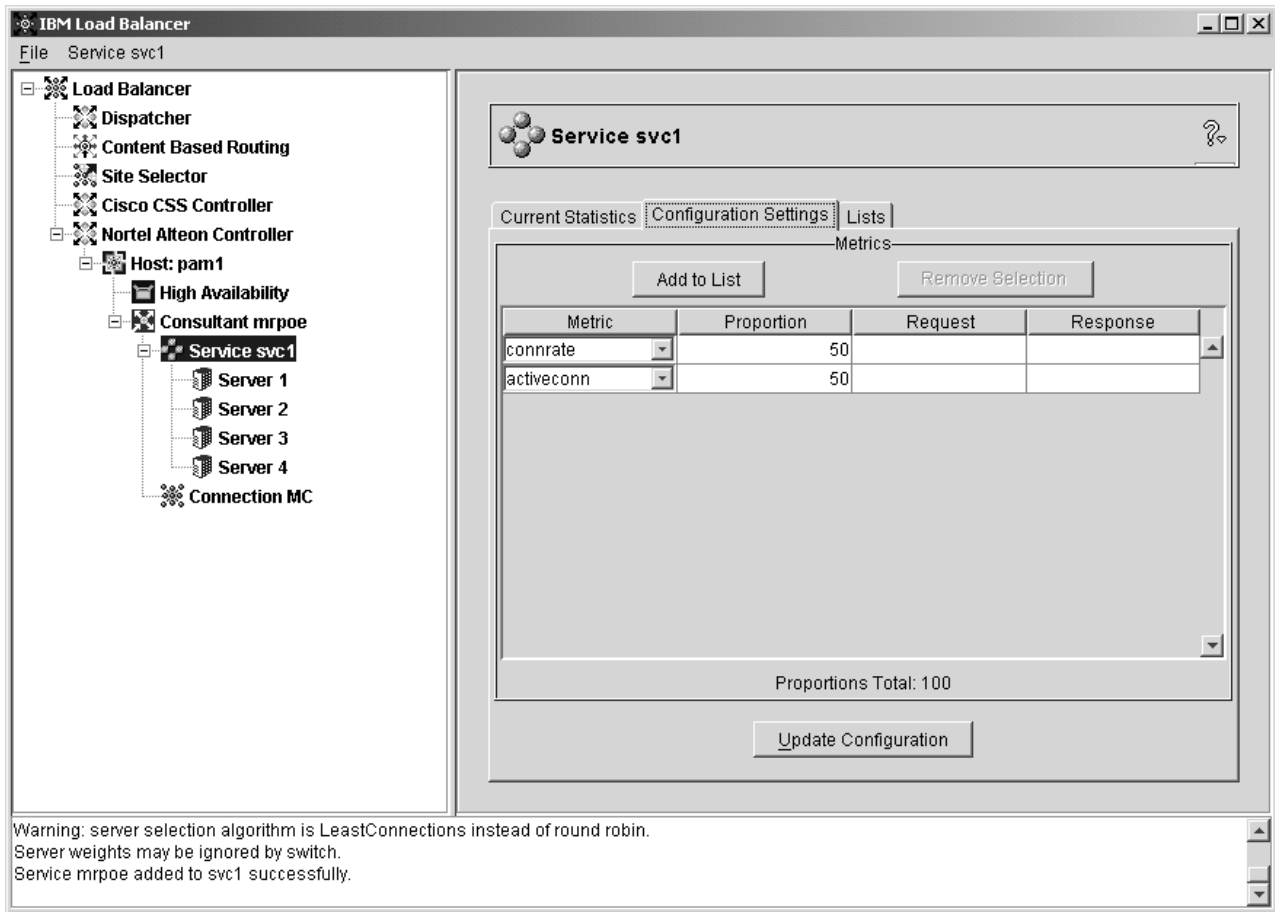


그림 45. GUI(Graphical User Interface)는 Nortel Alteon Controller 컴포넌트의 GUI 트리 구조 확장을 표시합니다.

모든 컴포넌트를 GUI에서 구성할 수 있습니다. 마우스 단추(일반적으로, 왼쪽 단추)를 한 번 클릭하여 트리 구조에서 요소를 선택한 다음, 마우스 단추(일반적으로, 오른쪽 단추)를 두 번 클릭하여 팝업 메뉴를 표시할 수 있습니다. 트리 요소에 대한 팝업 메뉴는 창의 맨 위에 위치한 메뉴 표시줄에서도 액세스할 수 있습니다.

더하기 또는 빼기 부호를 클릭하여 트리 구조의 항목을 펼치거나 압축할 수 있습니다.

GUI에서 명령을 실행하려면 GUI 트리에서 호스트 노드를 강조표시하고 호스트 팝업 메뉴에서 명령 전송....을 선택하십시오. 명령 입력 필드에서는 **executor report**와 같이 실행하려는 명령을 입력하십시오. 현재 세션에서 실행되는 명령의 결과 및 히스토리가 제공된 창에 나타납니다.

창의 오른쪽에는 현재 선택한 요소의 상태 지시자 탭이 표시됩니다.

- 현재 통계 탭은 요소에 대한 통계 정보를 표시합니다. 이 탭은 트리 구조의 모든 요소에 나타나지 않습니다.
- 통계 새로 고침 단추는 최신 통계 데이터를 표시합니다. 통계 새로 고침 단추가 나타나지 않으면 통계가 동적으로 새로 고쳐지고 항상 현재로 유지됩니다.

- 구성 설정값 탭은 각 컴포넌트의 구성 장에 대략적으로 설명되어 있는 프로시저를 사용하여 설정할 수 있는 구성 매개변수를 나타냅니다. 이 탭은 트리 구조의 모든 요소에 나타나지 않습니다.
- 구성 갱신 단추는 현재 실행 중인 구성에 최신 변경사항을 적용합니다.
- 비율 탭은 215 페이지의 제 22 장 『Dispatcher, CBR 및 Site Selector에 대한 고급 기능』의 정보를 사용하여 설정할 수 있는 비율(또는 가중치) 매개변수를 제공합니다. 이 탭은 트리 구조의 모든 요소에 나타나지 않습니다.
- 목록 탭은 선택한 트리 요소에 대한 추가 세부사항을 나타냅니다. 이 탭은 트리 구조의 모든 요소에 나타나지 않습니다.
- 제거 단추를 클릭하면 목록에서 강조표시된 항목이 삭제됩니다.
- 보고서 탭은 요소에 대한 관리자 보고서 정보를 표시합니다. 이 탭은 트리 구조의 모든 요소에 나타나지 않습니다.
- 보고서 새로 고침 단추는 최신 관리자 보고서 데이터를 표시합니다.

도움말에 액세스하려면 Load Balancer 창 오른쪽 상단 구석의 물음표(?)를 클릭하십시오.

- **도움말: 필드 레벨** — 각 필드 및 기본값을 설명합니다.
- **도움말: 수행 방법** — 해당 화면에서 수행할 수 있는 작업이 나열되어 있습니다.
- **InfoCenter** — 새 기능 정보의 개요 및 강조, 제품 웹 사이트에 대한 링크, 온라인 도움말 파일 색인, 용어집 등의 제품 정보에 대한 액세스를 제공합니다.

부록 B. 콘텐츠 규칙(패턴) 구문

이 부록에서는 CBR 컴포넌트의 콘텐츠 규칙 구문 사용법과 Dispatcher 컴포넌트의 cbr 전달 메소드를 사용법 예제 및 시나리오와 함께 설명합니다.

콘텐츠 규칙(패턴) 구문:

규칙 유형에 대한 "콘텐츠"를 선택한 경우에만 해당됩니다.

다음 제한 사항에 따라 사용할 패턴 구문을 입력하십시오.

- 패턴 내에 공백을 사용할 수 없습니다.
- 문자 앞에 백 슬래시(\):를 두는 경우를 제외하면 특수 문자도 사용할 수 없습니다.
 - * 와일드 카드(임의 문자 0-x를 대응시킵니다.)
 - (논리 그룹화에 사용되는 왼쪽 괄호
 -) 논리 그룹화에 사용되는 오른쪽 괄호
 - & 논리적 AND
 - | 논리적 OR
 - ! 논리적 NOT

예약된 키워드

예약 키워드 다음에는 항상 『=』 부호가 옵니다.

메소드 요청 시 HTTP 메소드(예: GET, POST 등)

URI URL 요청 경로(대소문자 구분)

버전 요청 고유 버전(HTTP/1.0 또는 HTTP/1.1)

호스트 호스트의 값: 헤더(대소문자 구분 안함)

주: HTTP/1.0 프로토콜에서는 선택적입니다.

<키> Dispatcher가 탐색할 수 있는 유효한 HTTP 헤더 이름. HTTP 헤더 예에는 User-Agent, Connection, Referer 등이 있습니다.

http://www.company.com/path/webpage.htm을 대상으로 설정한 브라우저 결과 값은 다음과 같습니다.

```
Method=GET
URI=/path/webpage.htm
Version=HTTP/1.1
```

```
Host=www.company.com
Connection=Keep-Alive
Referer=http://www.company.com/path/parentwebpage.htm
```

주: 운영 체제 셸에서 "&"같은 특수 문자를 해석하여 **cbrcontrol**이 평가하기도 전에 이들 문자를 대체 텍스트로 변환할 수 있습니다.

예를 들어, **cbrcontrol>>** 프롬프트를 사용할 경우에만 다음 명령이 유효합니다.

```
rule add 10.1.203.4:80:cbr_prod_rule_ek type content
pattern uri=/nipoe/*
```

특수 문자를 사용할 때, 동일한 명령이 운영 체제의 프롬프트에서(또는 구성 파일에서)도 작동하려면 다음과 같이 패턴에 큰따옴표(" ")가 있어야 합니다.

```
cbrcontrol rule add 10.1.203.4:80:cbr_prod_rule_ek type content
pattern "uri=/nipoe/*"
```

큰따옴표를 사용하지 않으면 규칙이 CBR에서 저장될 때 일부 패턴이 잘릴 수 있습니다. **cbrcontrol>>** 명령 프롬프트를 사용할 경우 따옴표가 지원되지 않습니다.

다음은 패턴 구문 사용 예제 및 가능한 시나리오 모음입니다.

시나리오 1

클러스터 이름을 하나 설정하는 데는 표준 HTML 콘텐츠에 대한 웹 서버 세트, servlet 요청에 대해 WebSphere Application Server에서 사용하는 또다른 웹 서버 세트, NSF 파일에 대한 또다른 Lotus® Notes® 서버 세트 등이 필요합니다. 요청된 페이지를 구별하려면 클라이언트 데이터에 액세스해야 합니다. 또한 요청된 페이지를 해당 서버로 전송해야 합니다. 규칙과 일치하는 콘텐츠 패턴은 이런 작업을 수행하는 데 필요한 분리를 제공해야 합니다. 필수 요청 분리가 자동으로 발생할 수 있도록 일련의 규칙이 구성됩니다. 예를 들어, 다음 명령으로 세 가지로 분할할 수 있습니다.

```
>>rule add cluster1:80:servlets type content pattern uri=*/servlet/*priority 1
>>rule uses cluster1:80:servlets server1+server2

>>rule add cluster1:80:notes type content pattern uri=*.nsf* priority 2
>>rule uses cluster1:80:notes server3+server4

>>rule add cluster1:80:regular type true priority 3
>>rule uses cluster1:80:regular server5+server6
```

NSF 파일 요청이 Load Balancer에 도착하면, servlets 규칙을 먼저 확인하지만 일치하지 않습니다. 그러면, notes 규칙이 요청을 확인하고 일치함을 리턴합니다. 클라이언트는 server3과 server4 간에 로드 밸런스됩니다.

시나리오 2

다른 일반적인 시나리오는 기본 웹 사이트가 몇 개의 별개 내부 그룹을 제어하는 경우입니다. 예를 들어, www.company.com/software에는 www.company.com/hardware 부분과는 서로 다른 서버 세트 및 콘텐츠가 있습니다. 요청은 모두 루트

www.company.com 클러스터를 기반으로 하지 않으므로 URI 차이점을 찾아 로드 밸런스를 완료하려면 콘텐츠 규칙이 필요합니다. 시나리오의 규칙은 다음과 유사합니다.

```
>>rule add cluster1:80:div1 type content pattern uri=/software/* priority 1
>>rule uses cluster1:80:div1 server1+server2

>>rule add cluster1:80:div2 type content pattern uri=/hardware/* priority 2
>>rule uses cluster1:80:div2 server3+server4
```

시나리오 3

특정 조합은 규칙 탐색 순서에 민감합니다. 예를 들어, 시나리오 2에서 클라이언트는 요청 경로의 디렉토리를 기반으로 분할되지만 대상 디렉토리는 경로의 다중 레벨에서 나타나 배치 시 서로 다른 의미를 가질 수 있습니다. 예를 들어, www.company.com/pcs/fixes/software는 www.company.com/mainframe/fixes/software와는 대상이 다릅니다. 이런 가능성을 고려하여 규칙을 정의해야 하며, 동시에 너무 많은 시나리오를 계획해서는 안 됩니다. 예를 들어, 『uri=*/software/*』 검사는 이 경우에는 너무 광범위한 와일드 카드 검색입니다. 다음과 같은 방법으로 대체 규칙을 구성할 수 있습니다.

조합 탐색으로 범위를 좁힐 수 있습니다.

```
>>rule add cluster1:80:pcs type content pattern (uri=/pcs/*)&(uri=*/software/*)
>>rule uses cluster 1:80:pcs server1
```

사용할 조합이 없는 경우, 순서가 중요합니다.

```
>>rule add cluster1:80:pc1 type content pattern uri=/pcs/*
>>rule uses cluster1:80:pc1 server2
```

두 번째 규칙은 『pcs』가 첫 번째 디렉토리가 아닌 이후의 디렉토리 지점에서 나타날 때 적용됩니다.

```
>>rule add cluster1:80:pc2 type content pattern uri=*/pcs/*
>>rule uses cluster1:80:pc2 server3
```

대부분의 경우, 사용자는 기본 **always true** 규칙을 사용하여 규칙을 완료하여 기타 규칙을 통해 실현되지 않는 사항을 적용하려고 합니다. 이것은 기타 모든 서버가 클라이언트에 대해 실패한 시나리오에서 『죄송합니다, 사이트가 현재 작동 중지 상태입니다. 잠시 후 다시 시도하여 주십시오』를 나타내는 서버가 될 수도 있습니다.

```
>>rule add cluster1:80:sorry type true priority 100
>>rule uses cluster1:80:sorry server5
```

부록 C. 예제 구성 파일

이 부록에는 Load Balancer의 Dispatcher 컴포넌트에 대한 예제 구성 파일이 들어 있습니다.

중요: IPv4 및 IPv6용 Load Balancer 설치를 사용하는 경우, 해당 샘플 구성 파일에 있는 dscontrol 명령어의 분리문자로 콜론(:) 대신 앳 마크(@)를 쓰십시오.

예제 Load Balancer 구성 파일

예제 파일은 ...ibm/edge/lb/servers/samples/ 디렉토리에 있습니다.

Dispatcher 구성 파일 — AIX, Linux 및 Solaris 시스템

```
#!/bin/bash
#
# configuration.sample - Sample configuration file for the
# Dispatcher component
#
#
# Ensure the root user is the one executing this script.
#
# iam=`whoami`

# if [ "$iam" != "root" ] if [ "$iam" != "root" ]
# then
#   echo "You must login as root to run this script"
#   exit 2
# fi

#
# First start the server
#
# dsserver start
# sleep 5

#
# Then start the executor
#
# dscontrol executor start

#
# The Dispatcher can be removed at any time using the
# "dscontrol executor stop" and "dsserver stop" commands to
# stop the executor and server respectively prior to removing
# the Dispatcher software.
#
# The next step in configuring the Dispatcher is to set the
# NFA (non-forwarding address) and the cluster address(es).
#
# The NFA is used to remotely access the Dispatcher machine
```

```

# for administration or configuration purposes. This
# address is required since the Dispatcher will forward packets
# to the cluster address(es).
#
# The CLUSTER address is the hostname (or IP address) to
# which remote clients will connect.
#
# Anywhere in this file, you may use hostnames and IP
# addresses interchangeably.
#

# NFA=hostname.domain.name
# CLUSTER=www.yourcompany.com

# echo "Loading the non-forwarding address"
# dscontrol executor set nfa $NFA

#
# The next step in configuring the Dispatcher is to create
# a cluster. The Dispatcher will route requests sent to
# the cluster address to the corresponding server machines
# defined to that cluster. You may configure and server
# multiple cluster address using Dispatcher.

# Use a similar configuration for CLUSTER2, CLUSTER3, etc.
#

# echo "Loading first CLUSTER address "
# dscontrol cluster add $CLUSTER

#
# Now we must define the ports this cluster will use. Any
# requests received by the Dispatcher on a defined port will
# be forwarded to the corresponding port of one of the server
# machines.
#

# echo "Creating ports for CLUSTER: $CLUSTER"

# dscontrol port add $CLUSTER:20+21+80

#
# The last step is to add each of the server machines to the
# ports in this cluster.
# Again, you can use either the hostname or the IP address
# of the server machines.
#

# SERVER1=server1name.domain.name
# SERVER2=server2name.domain.name
# SERVER3=server3name.domain.name

# echo "Adding server machines"
# dscontrol server add $CLUSTER:20+21+80:
# $SERVER1+$SERVER2+$SERVER3

#
# We will now start the load balancing components of the

```

```
# Dispatcher. The main load balancing component is called
# the manager and the second load balancing components are the
# advisors. If the manager and advisors are not running the
# Dispatcher sends requests in a round-robin format. Once the
# manager is started, weighting decisions based on the number
# of new and active connections is employed and incoming
# requests are sent to the best server. The advisors give the
# manager further insight into a servers ability to service
# requests as well as detecting whether a server is up. If
# an advisor detects that a server is down it will be
# marked down (providing the manager proportions have been
# set to include advisor input) and no further requests will be
# routed to the server.
```

```
# The last step in setting up the load balancing components
# is to set the manager proportions. The manager updates the
# weight of each of the servers based on four policies:
# 1. The number of active connections on each server.
# 2. The number of new connections to each server.
# 3. Input from the advisors.
# 4. Input from the system level advisor.
# These proportions must add up to 100. As an example, setting
# the manager proportions to
# dscontrol manager proportions 48 48 0 0
# will give active and new connections 48% input into the
# weighting decision, the advisors will contribute 4% and
# the system input will not be considered.
#
# NOTE: By default the manager proportions are set to 50 50 0 0
#
```

```
# echo "Starting the manager..."
# dscontrol manager start
```

```
# echo "Starting the FTP advisor on port 21 ..."
# dscontrol advisor start ftp 21
# echo "Starting the HTTP advisor on port 80 ..."
# dscontrol advisor start http 80
# echo "Starting the Telnet advisor on port 23 ..."
# dscontrol advisor start telnet 23
# echo "Starting the SMTP advisor on port 25 ..."
# dscontrol advisor start smtp 25
# echo "Starting the POP3 advisor on port 110 ..."
# dscontrol advisor start pop3 110
# echo "Starting the NNTP advisor on port 119 ..."
# dscontrol advisor start nntp 119
# echo "Starting the SSL advisor on port 443 ..."
# dscontrol advisor start ssl 443
#
```

```
# echo "Setting the manager proportions..."
# dscontrol manager proportions 58 40 2 0
```

```
#
# The final step in setting up the Dispatcher machine is to
# alias the Network Interface Card (NIC).
#
# NOTE: Do NOT use this command in a high availability
```

```

# environment. The go* scripts will configure the NIC and
# loopback as necessary.
# dscontrol executor configure $CLUSTER

# If your cluster address is on a different NIC or subnet
# from the NFA use the following format for the cluster configure
# command.
# dscontrol executor configure $CLUSTER tr0 0xfffff800
# where tr0 is your NIC (tr1 for the second token ring card, en0
# for the first ethernet card) and 0xfffff800 is a valid
# subnet mask for your site.
#

#
# The following commands are set to the default values.
# Use these commands as a guide to change from the defaults.
# dscontrol manager loglevel 1
# dscontrol manager logsize 1048576
# dscontrol manager sensitivity 5
# dscontrol manager interval 2
# dscontrol manager refresh 2
#
# dscontrol advisor interval ftp 21 5
# dscontrol advisor loglevel ftp 21 1
# dscontrol advisor logsize ftp 21 1048576
# dscontrol advisor timeout ftp 21 unlimited
# dscontrol advisor interval telnet 23 5
# dscontrol advisor loglevel telnet 23 1
# dscontrol advisor logsize telnet 23 1048576
# dscontrol advisor timeout telnet 23 unlimited
# dscontrol advisor interval smtp 25 5
# dscontrol advisor loglevel smtp 25 1
# dscontrol advisor logsize smtp 25 1048576
# dscontrol advisor timeout smtp 25 unlimited
# dscontrol advisor interval http 80 5
# dscontrol advisor loglevel http 80 1
# dscontrol advisor logsize http 80 1048576
# dscontrol advisor timeout http 80 unlimited
# dscontrol advisor interval pop3 110 5
# dscontrol advisor loglevel pop3 110 1
# dscontrol advisor logsize pop3 110 1048576
# dscontrol advisor timeout pop3 110 unlimited
# dscontrol advisor interval nntp 119 5
# dscontrol advisor loglevel nntp 119 1
# dscontrol advisor logsize nntp 119 1048576
# dscontrol advisor timeout nntp 119 unlimited
# dscontrol advisor interval ssl 443 5
# dscontrol advisor loglevel ssl 443 1
# dscontrol advisor logsize ssl 443 1048576
# dscontrol advisor timeout ssl 443 unlimited
#

```

Dispatcher 구성 파일 — Windows 시스템

다음은 Window에서 사용할 **configuration.cmd.sample**이라는 예제 Load Balancer 구성 파일입니다.


```

@echo off
rem configuration.cmd.sample - Sample configuration file for the
rem Dispatcher component.
rem

rem dsserver must be started by Services

rem

rem
rem Then start the executor
rem
rem call dscontrol executor start

rem

rem The next step in configuring the Dispatcher is to set the
rem NFA (non-forwarding address) and to set the cluster
rem address(es).
rem

rem The NFA is used to remotely access the Dispatcher
rem machine for administration configuration purposes. This
rem address is required since the Dispatcher will forward
rem packets to the cluster address(es).

rem
rem The CLUSTER address is the hostname (or IP address) to which
rem remote clients will connect.
rem

rem Anywhere in this file, you may use hostnames and IP
rem addresses interchangeably.
rem NFA=[non-forwarding address]
rem CLUSTER=[your clustername]
rem

rem set NFA=hostname.domain.name
rem set CLUSTER=www.yourcompany.com

rem echo "Loading the non-forwarding address"
rem call dscontrol executor set nfa %NFA%

rem
rem The following commands are set to the default values.
rem Use these commands to change the defaults

rem call dscontrol executor set fintimeout 30
rem
rem The next step in configuring the Dispatcher is to create
rem a cluster. The Dispatcher will route requests sent to
rem the cluster address to the corresponding server machines
rem defined to that cluster. You may configure and server
rem multiple cluster addresses using Dispatcher.
rem Use a similar configuration for CLUSTER2, CLUSTER3, etc.
rem

rem echo "Loading first CLUSTER address "

```

```

rem call dscontrol cluster add %CLUSTER%

rem
rem Now we must define the ports this cluster will use. Any
rem requests received by the Dispatcher on a defined port
rem will be forwarded to the corresponding
rem port of one of the server machines.
rem

rem echo "Creating ports for CLUSTER: %CLUSTER%"
rem call dscontrol port add %CLUSTER%:20+21+80

rem
rem The last step is to add each of the server machines to
rem the ports in this cluster. Again, you can use either the
rem hostname or the IP address of the server machines.
rem

rem set SERVER1=server1name.domain.name
rem set SERVER2=server2name.domain.name
rem set SERVER3=server3name.domain.name

rem echo "Adding server machines"
rem call dscontrol server add %CLUSTER%:20+21+80:
rem %SERVER1%+%SERVER2%+%SERVER3%

rem
rem We will now start the load balancing components of the
rem Dispatcher. The main load balancing component is called
rem the manager and the second load balancing components are the
rem advisors. If the manager and advisors are not
rem running the Dispatcher sends requests in a round-robin
rem format. Once the manager is started, weighting decisions
rem based on the number of new and active connections is
rem employed and incoming requests are sent to the best
rem server. The advisors give the manager further insight
rem into a servers ability to service requests as well as
rem detecting whether a server is up. If an advisor detects
rem that a server is down it will be marked down (providing the
rem manager proportions have been set to include advisor
rem input) and no further requests will be routed to the server.
rem The last step in setting up the load balancing
rem components is to set the manager proportions. The
rem manager updates the weight of each of the servers based
rem on four policies:

rem 1. The number of active connections on each server
rem 2. The number of new connections for each server
rem 3. Input from the advisors.
rem 4. Input from the system level advisor.
rem
rem These proportions must add up to 100. As an example,
rem setting the cluster proportions using
rem dscontrol cluster set <cluster> proportions 48 48 4 0
rem will give active and new connections 48% input into the
rem weighting decision, the advisor will contribute 4% and
rem the system input will not be considered.
rem

```

```

rem NOTE: By default the manager proportions are set to
rem 50 50 0 0

rem echo "Starting the manager..."
rem call dscontrol manager start

rem echo "Starting the FTP advisor on port 21 ..."
rem call dscontrol advisor start ftp 21
rem echo "Starting the HTTP advisor on port 80 ..."
rem call dscontrol advisor start http 80
rem echo "Starting the Telnet advisor on port 23 ..."
rem call dscontrol advisor start telnet 23
rem echo "Starting the SMTP advisor on port 25 ..."
rem call dscontrol advisor start smtp 25
rem echo "Starting the POP3 advisor on port 110 ..."
rem call dscontrol advisor start pop3 110
rem echo "Starting the NNTP advisor on port 119 ..."
rem call dscontrol advisor start nntp 119
rem echo "Starting the SSL advisor on port 443 ..."
rem call dscontrol advisor start ssl 443
rem

rem echo "Setting the cluster proportions..."
rem call dscontrol cluster set %CLUSTER% proportions 58 40 2 0

rem
rem The final step in setting up the Dispatcher machine is
rem to alias the Network Interface Card (NIC).
rem
rem NOTE: Do NOT use this command in a high availability
rem environment. The go* scripts will configure the NIC and
rem loopback as necessary.
rem
rem dscontrol executor configure %CLUSTER%

rem If your cluster address is on a different NIC or subnet
rem from the NFA use the following format for the cluster
rem configure command.
rem dscontrol executor configure %CLUSTER% tr0 0xfffff800
rem where tr0 is your NIC (tr1 for the second token ring card,
rem en0 for the first ethernet card) and 0xfffff800 is
rem a valid subnet mask for your site.
rem

rem
rem The following commands are set to the default values.
rem Use these commands to guide to change from the defaults.
rem call dscontrol manager loglevel 1
rem call dscontrol manager logsize 1048576
rem call dscontrol manager sensitivity 5
rem call dscontrol manager interval 2
rem call dscontrol manager refresh 2
rem
rem call dscontrol advisor interval ftp 21 5
rem call dscontrol advisor loglevel ftp 21 1
rem call dscontrol advisor logsize ftp 21 1048576
rem call dscontrol advisor timeout ftp 21 unlimited
rem call dscontrol advisor interval telnet 23 5

```

```

rem call dscontrol advisor loglevel telnet 23 1
rem call dscontrol advisor logsize telnet 23 1048576
rem call dscontrol advisor timeout telnet 23 unlimited
rem call dscontrol advisor interval smtp 25 5
rem call dscontrol advisor loglevel smtp 25 1
rem call dscontrol advisor logsize smtp 25 1048576
rem call dscontrol advisor timeout smtp 25 unlimited
rem call dscontrol advisor interval http 80 5
rem call dscontrol advisor loglevel http 80 1
rem call dscontrol advisor logsize http 80 1048576
rem call dscontrol advisor timeout http 80 unlimited
rem call dscontrol advisor interval pop3 110 5
rem call dscontrol advisor loglevel pop3 110 1
rem call dscontrol advisor logsize pop3 110 1048576
rem call dscontrol advisor timeout pop3 110 unlimited
rem call dscontrol advisor interval nntp 119 5
rem call dscontrol advisor loglevel nntp 119 1
rem call dscontrol advisor logsize nntp 119 1048576
rem call dscontrol advisor timeout nntp 119 unlimited
rem call dscontrol advisor interval ssl 443 5
rem call dscontrol advisor loglevel ssl 443 1
rem call dscontrol advisor logsize ssl 443 1048576
rem call dscontrol advisor timeout ssl 443 unlimited
rem

```

어드바이저 예제

다음은 **ADV_sample**이라는 어드바이저 예 파일입니다.

```

/**
 * ADV_sample: The Load Balancer HTTP advisor
 *
 *
 * This class defines a sample custom advisor for Load Balancer. Like all
 * advisors, this custom advisor extends the function of the advisor base,
 * called ADV_Base. It is the advisor base that actually performs most of
 * the advisor's functions, such as reporting loads back to the Load Balancer
 * for use in the Load Balancer's weight algorithm. The advisor base also
 * performs socket connect and close operations and provides send and receive
 * methods for use by the advisor. The advisor itself is used only for
 * sending and receiving data to and from the port on the server being
 * advised. The TCP methods within the advisor base are timed to calculate
 * the load. A flag within the constructor in the ADV_base overwrites the
 * existing load with the new load returned from the advisor if desired.
 *
 * Note: Based on a value set in the constructor, the advisor base supplies
 * the load to the weight algorithm at specified intervals. If the actual
 * advisor has not completed so that it can return a valid load, the advisor
 * base uses the previous load.
 *
 * NAMING
 *
 * The naming convention is as follows:
 *
 * - The file must be located in the following Load Balancer directory:
 *
 *    lb/servers/lib/CustomAdvisors/ (lb\servers\lib\CustomAdvisors on Windows)
 *
 * - The Advisor name must be preceded with "ADV_". The advisor can be
 *   started with only the name, however; for instance, the "ADV_sample"

```

```

*   advisor can be started with "sample".
*
* - The advisor name must be in lowercase.
*
* With these rules in mind, therefore, this sample is referred to as:
*
*   <base directory>/lib/CustomAdvisors/ADV_sample.class
*
*
* Advisors, as with the rest of Load Balancer, must be compiled with the
* prereq version of Java. To ensure access to Load Balancer classes, make
* sure that the ibmlb.jar file (located in the lib subdirectory of the base
* directory) is included in the system's CLASSPATH.
*
* Methods provided by ADV_Base:
*
* - ADV_Base (Constructor):
*
*   - Pargs
*     - String sName = Name of the advisor
*     - String sVersion = Version of the advisor
*     - int iDefaultPort = Default port number to advise on
*     - int iInterval = Interval on which to advise on the servers
*     - String sDefaultName = Unused. Must be passed in as "".
*     - boolean replace = True - replace the load value being calculated
*                               by the advisor base
*                               False - add to the load value being calculated
*                               by the advisor base
*   - Return
*     - Constructors do not have return values.
*
* Because the advisor base is thread based, it has several other methods
* available for use by an advisor. These methods can be referenced using
* the CALLER parameter passed in getLoad().
*
* These methods are as follows:
*
* - send - Send a packet of information on the established socket connection
*         to the server on the specified port.
*   - Pargs
*     - String sDataString - The data to be sent in the form of a string
*   - Return
*     - int RC - Whether the data was sucessfully sent or not: zero indicates
*               data was sent; a negative integer indicates an error.
*
* - receive - Receive information from the socket connection.
*   - Pargs
*     - StringBuffer sbDataBuffer - The data received during the receive call
*   - Return
*     - int RC - Whether the data was successfully received or not; zero
*               indicates data was sent; a negative integer indicates
*               an error.
*
* If the function provided by the advisor base is not sufficient,
* you can create the appropriate function within the advisor and
* the methods provided by the advisor base will then be ignored.
*
* An important question regarding the load returned is whether to apply
* it to the load being generated within the advisor base,
* or to replace it; there are valid instances of both situations.
*
* This sample is essentially the Load Balancer HTTP advisor. It functions
* very simply: a send request--an http head request--is issued. Once a

```

```

* response is received, the getLoad method terminates, flagging the advisor
* base to stop timing the request. The method is then complete. The
* information returned is not parsed; the load is based on the time
* required to perform the send and receive operations.
*/

package CustomAdvisors;
import com.ibm.internet.nd.advisors.*;

public class ADV_sample extends ADV_Base implements ADV_MethodInterface
{
    String COPYRIGHT = "(C) Copyright IBM Corporation 1997, All Rights Reserved.\n";
    static final String ADV_NAME = "Sample";
    static final int ADV_DEF_ADV_ON_PORT = 80;
    static final int ADV_DEF_INTERVAL = 7;

    // Note: Most server protocols require a carriage return ("\r") and line
    // feed ("\n") at the end of messages. If so, include them in
    // your string here.
    static final String ADV_SEND_REQUEST =
        "HEAD / HTTP/1.0\r\nAccept: */*\r\nUser-Agent: " +
        "IBM_Load_Balancer_HTTP_Advisor\r\n\r\n";

    /**
     * Constructor.
     *
     * Parms: None; but the constructor for ADV_Base has several parameters
     * that must be passed to it.
     */
    public ADV_sample()
    {
        super( ADV_NAME,
            "2.0.0.0-03.27.98",
            ADV_DEF_ADV_ON_PORT,
            ADV_DEF_INTERVAL,
            "", // not used
            false);
        super.setAdvisor( this );
    }

    /**
     * ADV_AdvisorInitialize
     *
     * Any Advisor-specific initialization that must take place after the
     * advisor base is started. This method is called only once and is
     * typically not used.
     */
    public void ADV_AdvisorInitialize()
    {
        return;
    }

    /**
     * getLoad()
     *
     * This method is called by the advisor base to complete the advisor's
     * operation, based on details specific to the protocol. In this sample
     * advisor, only a single send and receive are necessary; if more complex
     * logic is necessary, multiple sends and receives can be issued. For
     * example, a response might be received and parsed. Based on the
     * information learned thereby, another send and receive could be issued.

```

```

*
* Parameters:
*
* - iConnectTime - The current load as it refers to the length of time it
*                  took to complete the connection to the server through
*                  the specified port.
*
* - caller - A reference to the advisor base class where the Load
*            Balancer-supplied methods are to perform simple TCP requests,
*            mainly send and receive.
*
* Results:
*
* - The load - A value, expressed in milliseconds, that can either be added
*              to the existing load, or that can replace the existing load, as
*              determined by the constructor's "replace" flag.
*
*              The larger the load, the longer it took the server to respond;
*              therefore, the lower the weight will become within the Load Balancer.
*
*              If the value is negative, an error is assumed. An error from an
*              advisor indicates that the server the advisor is trying to reach is not
*              accessible and has been identified as being down. Load Balancer will
*              not attempt to load balance to a server that is down. Load Balancer will
*              resume load balancing to the server when a positive value is received.
*
*/
public int getLoad(int iConnectTime, ADV_Thread caller)
{
    int iRc;
    int iLoad = ADV_HOST_INACCESSIBLE; // -1

    // Send tcp request
    iRc = caller.send(ADV_SEND_REQUEST);
    if (iRc >= 0)
    {
        // Perform a receive
        StringBuffer sbReceiveData = new StringBuffer("");
        iRc = caller.receive(sbReceiveData);

        /**
         * In the normal advisor mode ("replace" flag is false), the load
         * returned is either 0 or 1 indicating the server is up or down.
         * If the receive is successful, a load of zero is returned
         * indicating that the load built within the base advisor is to be used.
         *
         * Otherwise ("replace" flag is true), return the desired load value.
         */

        if (iRc >= 0)
        {
            iLoad = 0;
        }
    }
    return iLoad;
}

} // End - ADV_sample

```


부록 D. Dispatcher, CBR 및 Caching Proxy를 사용하는 2단 고가용성 구성 예제

이 부록은 두 개의 Load Balancer 컴포넌트(Dispatcher 컴포넌트 및 CBR 컴포넌트)를 Caching Proxy와 결합하는 2단 고가용성 구성을 어떻게 설정하는지를 설명합니다.

서버 시스템 설정

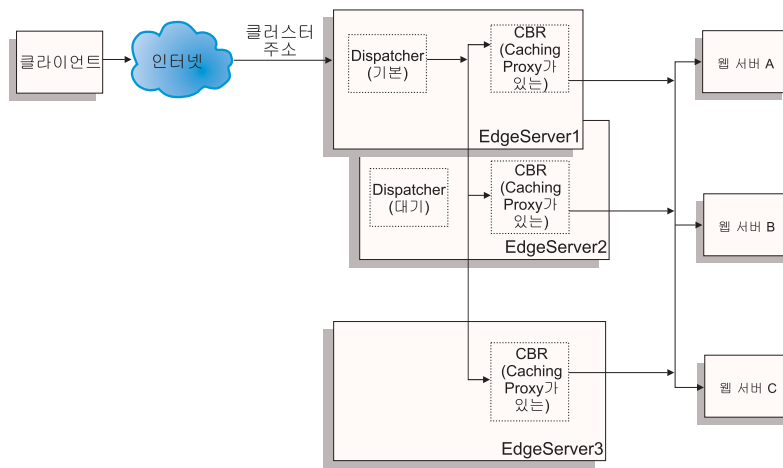


그림 46. Dispatcher, CBR 및 Caching Proxy를 사용하는 2단 고가용성 구성 예제

그림 46에 대해 설정된 서버 시스템은 다음과 같습니다.

- EdgeServer1: 웹 서버를 가로질러 로드 밸런싱하는 CBR 및 Caching Proxy와 함께 위치해 있는 기본(고가용성) Dispatcher 시스템
- EdgeServer2: CBR 및 Caching Proxy와 함께 위치해 있는 대기(고가용성) Dispatcher 시스템
- EdgeServer3: CBR 및 Caching Proxy 시스템
- WebServerA, WebServerB, WebServerC: 백엔드 웹 서버

그림 46에서는 다중 백엔드 웹 서버를 가로질러 로드 밸런싱하는 다중 서버(EdgeServer1, EdgeServer2, EdgeServer3)의 기본 표현을 보여줍니다. CBR 컴포넌트는 Caching Proxy를 사용하여 URL 콘텐츠에 근거한 요청을 백엔드 웹 서버에 전달합니다. Dispatcher 컴포넌트는 Edge Server를 가로질러 CBR 컴포넌트를 로드 밸런싱하는데 사용합니다. Dispatcher 컴포넌트의 고가용성 기능을 사용하면 기본 고가용성 시스템(EdgeServer1)이 작동하지 않을 때 백엔드 서버에 계속해서 요청할 수 있습니다.

기본 구성 지침

- Caching Proxy를 모든 EdgeServers에서 동일하게 설정하십시오. 백엔드 서버에서 웹 페이지에 접근 가능성을 전반적으로 향상시키려면 Caching Proxy를 설정하여 메모리 캐시를 하십시오. 이렇게 하면 EdgeServers는 더 자주 요청되는 웹 페이지를 캐시할 수 있습니다. Caching Proxy 설정에 대한 자세한 정보는 *Caching Proxy 관리 안내서*를 참조하십시오.
- 클러스터 주소 및 포트를 Load Balancer의 Dispatcher 컴포넌트와 CBR에서 모두 동일하게 정의하십시오.
- CBR 컴포넌트를 모든 EdgeServers에서 동일하게 구성하십시오. 클러스터에 대하여 정의하려는 포트의 서버로 웹 서버 A, B 및 C를 사용하십시오. CBR 구성에 대한 자세한 정보는 123 페이지의 제 11 장 『Content Based Routing 구성』을 참조하십시오.
- Dispatcher 컴포넌트를 EdgeServer1 및 EdgeServer2에서 동일하게 구성하십시오. Dispatcher가 로드 밸런싱하는 클러스터에서 정의하려는 포트의 서버로 모든 EdgeServers를 정의하십시오. Dispatcher 구성 방법에 대한 자세한 정보는 69 페이지의 제 7 장 『Dispatcher 구성』을 참조하십시오.
- EdgeServer1은 기본고가용성 시스템으로, EdgeServer2는 대기(백업)고가용성 시스템으로 구성하십시오. 자세한 내용은 218 페이지의 『고가용성』을 참조하십시오.

주:

1. 클라이언트의 URL에 백엔드 서버 주소가 표시되지 않게 하려면 Caching Proxy 구성 파일의 각 백엔드 서버 주소에 대해 ReversePass 지시문을 설정해야 합니다.
2. 웹 메모리가 효과적으로 사용되는지 확인하려면, "Caching" 지시문을 "ON"으로 설정하고 "CacheMemory" 지시문을 Caching Proxy 구성 파일이 요구하는 크기로 증가시키십시오.
3. Sample lines referred to in notes 1-2 (above):

```
Caching                ON
CacheMemory            128000 K
ReversePass /* http://websrvA.company.com/* http://www.company.com/*
```
4. EdgeServer1의 네트워크 인터페이스 카드의 클러스터 주소에 별명을 지정하고 나머지 EdgeServers의 루프백 장치 클러스터 주소에 별명을 지정하십시오.
5. EdgeServers에 Linux 플랫폼을 사용할 경우, Linux 커널에 패치를 설치하거나 대체를 사용하여 루프백 장치의 주소를 지정해야 합니다. 자세한 내용은 85 페이지의 『Load Balancer의 mac 전달 사용 시 Linux 루프백 별명 지정 대안』을 참조하십시오.
6. CBR에 경우 포트 연관 관계(stickytime)는 콘텐츠 규칙이 사용될 때는 사용해서는 안됩니다. 포트 연관 관계를 사용하는 경우, 백엔드 웹 서버로 요청을 처리하는 동안 콘텐츠 규칙이 적용되지 않습니다.

예제 구성 파일:

다음 예제 구성 파일은 523 페이지의 그림 46에서와 같이 Edge Components 구성 설정 시 작성되는 파일과 유사합니다. 예제 구성 파일은 Load Balancer의 CBR 컴포넌트 및 Dispatcher용 파일을 표시합니다. 예제 구성에서 하나의 이더넷 어댑터는 EdgeServer 시스템 각각에 대하여 사용되고 모든 주소는 개인용 서브넷 내에서 표시됩니다. 예제 구성 파일은 지정된 시스템에 대해 다음 IP 주소를 사용합니다.

- EdgeServer1(기본 고가용성 EdgeServer): 192.168.1.10
- EdgeServer2(백업 고가용성 EdgeServer): 192.168.1.20
- EdgeServer3(웹 캐싱 EdgeServer): 192.168.1.30
- 웹 사이트 클러스터 주소: 192.168.1.11
- WebServersA-C (백엔드 웹 서버): 192.168.1.71, 192.168.1.72 및 192.168.1.73

기본 고가용성 EdgeServer의 Dispatcher 컴포넌트에 대한 샘플 구성 파일

```
dscontrol executor start

dscontrol cluster add 192.168.1.11 primaryhost 192.168.1.10

dscontrol port add 192.168.1.11:80

dscontrol server add 192.168.1.11:80:edgeserver1 address 192.168.1.10
dscontrol server add 192.168.1.11:80:edgeserver2 address 192.168.1.20
dscontrol server add 192.168.1.11:80:edgeserver3 address 192.168.1.30

dscontrol manager start manager.log 10004

dscontrol highavailability heartbeat add 192.168.1.10 192.168.1.20
dscontrol highavailability backup add primary auto 4567
```

EdgeServers의 CBR 컴포넌트에 대한 샘플 구성 파일

```
cbrcontrol set loglevel 1
cbrcontrol executor start

cbrcontrol cluster add 192.168.1.11

cbrcontrol port add 192.168.1.11:80

cbrcontrol server add 192.168.1.11:80:webserverA address 192.168.1.71
cbrcontrol server add 192.168.1.11:80:webserverB address 192.168.1.72
cbrcontrol server add 192.168.1.11:80:webserverC address 192.168.1.73

cbrcontrol rule add 192.168.1.11:80:webA_rule type content
    pattern (URI=*WSA*)|(URI=*wsA*) priority 21
cbrcontrol rule useserver 192.168.1.11:80:webA_rule webserverA

cbrcontrol rule add 192.168.1.11:80:webB_rule type content
```

```
pattern (URI=/WS_B*) priority 22
cbrcontrol rule useserver 192.168.1.11:80:webB_rule webserverB

cbrcontrol rule add 192.168.1.11:80:webC_rule type content
pattern URI=*webC* priority 23
cbrcontrol rule useserver 192.168.1.21:80:webC_rule webserverC
```

부록 E. 주의사항

이 정보는 미국에서 제공되는 제품 및 서비스용으로 작성된 것입니다.

IBM은 다른 국가에서는 이 자료에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 한국 IBM 담당자에게 문의하십시오. 이 책에서 IBM 제품, 프로그램 또는 서비스를 언급하는 것이 해당 IBM 제품, 프로그램 또는 서비스만을 사용할 수 있다는 것을 의미하지는 않습니다. IBM의 지적 재산을 침해하지 않는 한 기능상으로 동등한 제품, 프로그램 또는 서비스를 대신 사용할 수도 있습니다. 그러나 비IBM 제품, 프로그램 또는 서비스 운영에 대한 평가 및 검증은 사용자의 책임입니다.

IBM은 이 책에서 다루고 있는 특정 내용에 대해 특허를 보유하고 있거나 현재 출원 중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 사용권을 부여하는 것은 아닙니다. 사용권 조치는 다음 주소로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

전화번호: 080-023-8080

2바이트(DBCS) 정보에 대한 사용권 문의는 한국 IBM 고객만족센터에 문의하거나 다음 주소로 서면문의하시기 바랍니다.

IBM World Trade Asia Corporation

Licensing

2-31 Roppongi 3-chome, Minato-ku

Tokyo 106, Japan

다음 단락은 현지법과 상충하는 영국이나 기타 국가에서는 적용되지 않습니다.

IBM은 타인이 권리 비침해, 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여 (단, 이에 한하지 않음) 묵시적이든 명시적이든 어떠한 종류의 보증없이 이 책을 현상태대로 제공합니다. 일부 국가에서는 특정 거래에서 명시적 또는 묵시적 보증의 면책사항을 허용하지 않으므로 이 사항이 적용되지 않을 수도 있습니다.

이 정보에는 기술적으로 부정확한 내용이나 인쇄상의 오류가 있을 수 있습니다. 이 정보는 주기적으로 변경되며 이 변경사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및/또는 프로그램을 사전 통지 없이 언제든지 개선 및(또는) 변경할 수 있습니다.

이 정보에서 비IBM의 웹 사이트는 단지 편의상 제공된 것으로 어떤 방식으로든 이들 웹 사이트를 옹호하고자 하는 것은 아닙니다. 해당 웹 사이트의 자료는 본 IBM 제품 자료의 일부분이 아니므로 해당 웹 사이트 사용으로 인한 위험은 사용자 본인이 감수해야 합니다.

IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각되는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

(i) 독자적으로 작성된 프로그램 및 기타 프로그램(이 프로그램 포함)간의 정보 교환 및
(ii) 교환된 정보의 상호 이용을 목적으로 정보를 원하는 프로그램 사용권자는 다음 주소로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

이러한 정보는 해당 조항 및 조건(예를 들어, 사용권 지불 등)에 따라 사용할 수 있습니다.

이 정보에 기술된 사용 허가 프로그램 및 사용 가능한 모든 사용권 자료는 IBM under terms of the IBM이 IBM 기본 계약, IBM 프로그램 사용권 계약(IPLA) 또는 이와 동등한 계약에 따라 제공한 것입니다.

이 책에 포함된 성능 데이터는 제한된 환경에서 산출된 것입니다. 그러므로 다른 운영 환경에서의 결과와 상당히 다를 수 있습니다. 일부 측정은 개발 단계의 시스템에서 이루어진 것이므로 그 측정치가 일반적으로 사용 가능한 환경에서도 동일하다고 보장할 수 없습니다. 또한 일부 측정은 보외법을 통해 이루어졌습니다. 따라서 실제 결과는 다를 수 있습니다. 이 책의 사용자는 본인의 고유 환경에 적용할 수 있는 데이터를 확인해야 합니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료 또는 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 비IBM 제품을 테스트하지 않았으므로, 이들 제품과 관련된 성능의 정확성, 호환성 또는 기타 주장에 대해서는 확신할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

IBM의 향후 방침 또는 의도에 대한 모든 내용은 통지 없이 변경되거나 철회될 수 있으며 목적 및 목표만을 나타냅니다.

이 책에는 일상적인 업무 처리에 사용되는 자료와 보고서의 예가 들어 있습니다. 보다 구체적으로 예를 나타내기 위해 특정 개인, 회사, 상표명이 언급되는 경우도 있습니다. 여기서 언급된 이름은 가상의 것이므로 실제 업무에 사용되는 이름 및 주소와 유사하다면 우연일 뿐입니다.

이 정보를 소프트웨어로 보고 있으면 사진 및 색상이 잘 나타나지 않을 수 있습니다.

상표

다음 용어는 미국 또는 기타 국가에서 사용되는 IBM Corporation의 상표 또는 등록 상표입니다.

AFS

AIX

DFS

IBM

iSeries™

NetView

OS/2

Redbooks™

RS/6000®

SecureWay

ViaVoice

WebSphere

zSeries®

Java 및 모든 Java 기반 상표는 미국 또는 기타 국가에서 사용되는 Sun Microsystems, Inc.의 상표입니다.

Microsoft, Windows, Windows NT 및 Windows 로고는 미국 또는 기타 국가에서 사용되는 Microsoft Corporation의 상표입니다.

Intel™, Intel Inside(로고), MMX™ 및 Pentium®은 미국 또는 기타 국가에서 사용되는 Intel Corporation의 상표입니다.

UNIX는 미국 또는 기타 국가에서 사용되는 The Open Group의 등록상표입니다.

Linux는 미국 또는 기타 국가에서 사용되는 Linus Torvalds의 상표입니다.

기타 회사, 제품 및 서비스 이름은 해당 회사의 상표 또는 서비스표입니다.

용어

가

게이트웨이. 다른 구조를 갖는 두 개의 컴퓨터 네트워크를 상호 연결하는 기능 단위.

결합 시간. 한 연결의 닫기와 첫 번째 연결 중에 사용되는 동일한 서버로 다시 클라이언트가 전송될 새로운 연결이 열리는 사이의 간격. 결합 시간 이후에 클라이언트는 첫 번째 서버와 다른 서버로 전송될 수도 있습니다.

고가용성. 부품이 실패한 경우 다른 부품의 기능을 Load Balancer가 넘겨 받을 수 있는 Load Balancer 기능.

관리 대상 노드. 인터넷 통신에서 네트워크 관리 에이전트를 포함하는 워크스테이션, 서버 또는 라우터. 인터넷 프로토콜(IP)에서 관리 대상 노드는 대개 SNMP(Simple Network Management Protocol) 에이전트를 포함합니다.

관리자. 여러 Load Balancer 기능 중 하나 관리자는 실행 프로그램의 내부 카운터와 어드바이저가 제공하는 피드백에 따라 가중치를 설정합니다. 실행 프로그램에서는 가중치를 사용하여 로드 밸런스를 수행합니다.

규칙. 규칙 기반 로드 밸런스에서, 대상 주소 및 포트가 아닌 정보에 따라 서버를 선택할 수 있도록 서버를 그룹화하는 메커니즘.

규칙 유형. 규칙 기반 로드 밸런스에서 규칙이 만족되는지 여부를 판별하기 위해 평가해야 하는 정보의 지시자.

기본. Dispatcher 고가용성에서 활동적으로 패킷의 경로를 지정하는 시스템으로서 시작하는 시스템. 상대 시스템인 백업 시스템이 기본 시스템의 상태를 모니터링하며 필요에 따라 넘겨 받습니다. 백업 및 고가용성도 참조하십시오.

기본값. 명시적으로 지정되지 않은 경우 가정되는 값, 속성 또는 옵션.

나

네트워크. 하드웨어 및 소프트웨어 데이터 통신 시스템. 네트워크는 종종 지역적 범위(예: LAN(Local Area Network), MAN(Metropolitan Area Network), WAN(Wide Area Network)) 및 사용되는 프로토콜에 의해 분류됩니다.

네트워크 관리 스테이션. SNMP(Simple Network Management Protocol)에서 네트워크 요소를 모니터링하고 제어하는 관리 응용프로그램을 실행하는 스테이션

네트워크 근접성. 라운드 트립 시간을 측정하여 Site Selector가 결정하는 클라이언트와 서버 같은 두 네트워크 엔티티의 근접성.

네트워크 주소 변환. NAT 또는 Network Address Translator, 가상 LAN. 이미 사용 중인 인터넷 주소를 확장하는 데 사용되거나 현재 개발 중인 하드웨어. 중복 IP 주소를 회사 내에 사용하고 고유 주소를 외부에서 사용할 수 있습니다.

네트워크 주소 포트 변환. NAPT, 포트 맵핑이라고도 합니다. 다른 포트 번호에서 인식하도록 여러 서버 디먼을 하나의 물리적 서버 내에 구성할 수 있습니다.

넷마스크. IPv4의 경우, IP 주소의 호스트 부분에서 서브네트워크 주소 비트를 식별하는 데 사용되는 32비트 마스크

다

다중 주소 결합 배치. 다중 주소 결합 배치를 통해 고객은 배열된 서버의 주소를 구성에서 NFA(NonForwarding Address)와 다르게 지정할 수 있습니다. 관련 주제: 결합 배치.

단절 표시. 서버에 대한 모든 활성 연결을 중단하고 새로운 연결이나 패킷이 그 서버에 전송되지 않도록 정지하는 것.

대역폭. 전송 채널의 최고 주파수와 최저 주파수의 차이. 초당 정해진 통신 회로를 통해 전송할 수 있는 데이터의 양.

도달. Dispatcher에서 제공된 대상에 ping을 발행하고 대상이 응답하는지 보고하는 어드바이저.

도달 주소. Dispatcher의 고가용성에서 어드바이저가 대상이 응답 중인지를 알기 위해 ping 명령을 발행하는 대상의 주소.

도메인 이름 서버. DNS. 호스트 이름을 인터넷 주소로 변환하기 위해 주로 인터넷에서 사용하는 일반 용도의 분배 복제 데이터 조회 서비스. 완전한 도메인 이름이라고 하지만, 인터넷에 사용되는 호스트 이름 양식이기도 합니다. DNS는 일치하는 이름을 찾을 때까지 찾을 이름의 도메인에 따라 이름 서버의 순서를 사용하도록 구성할 수 있습니다.

디먼. 디스크 및 실행 모니터. 명시적으로 관련되지 않으나 일부 조건이 발생할 때까지 대기하고 있는 프로그램. 조건 원인자는 디먼이 잠재해 있다는 것을 인식하지 않아도 됩니다(그러나 프로그램은 디먼을 암시적으로 호출할 것을 알기 때문에 조치를 확약함).

라

라우터. 패킷을 네트워크 사이로 전달하는 장치. 전달 결정은 흔히 경로 지정 제품에 의해 작성되는 네트워크 계층 정보와 라우팅 테이블을 기준으로 합니다.

라우트. 기점에서 대상까지의 네트워크 통신 경로.

루트 사용자. 대개 시스템을 관리하는 사용자와 관련된 AIX, Red Hat Linux 또는 Solaris 운영 체제의 일부를 액세스하고 수정할 수 있는 제한되지 않는 권한

루프백 별명. 루프백 인터페이스와 연관된 대체 IP 주소. 대체 주소는 실제 인터페이스에서 눈에 띄지 않는 유용한 부수 영향을 미칩니다.

루프백 인터페이스. 같은 시스템 내에서 정보가 한 엔티티에 대해 주소지정될 때, 불필요한 통신 기능을 생략하는 인터페이스.

리턴 주소. 고유한 IP 주소 또는 호스트 이름. Dispatcher 시스템에 구성되며, 서버에 대한 클라이언트의 요청을 로드 밸런스할 때 Dispatcher가 출발지 주소로 사용합니다.

마

마법사. 특정 타스크를 통해 사용자를 안내하는 단계별 지침을 사용하는 응용프로그램 내의 대화.

메트릭. 네트워크의 로드 밸런스에 사용할 수 있는 수치(예: 현재 로그인한 사용자의 수)을 리턴하는 프로세스 또는 명령.

메트릭 주소. Metric Server가 연결되는 주소.

메트릭 콜렉터. 컨설턴트에 상주하며 메트릭 수집을 담당합니다.

대상 주소. 하트 비트와 응답이 전송되는 고가용성 상대 시스템의 주소.

바

방화벽(Firewall). 업무와 같은 개인 네트워크를 인터넷과 같은 공용 네트워크에 연결하는 컴퓨터. 두 네트워크간의 액세스를 제한하는 프로그램이 포함됩니다. 프록시 게이트웨이를 참조하십시오.

배치(collocate). 동일한 시스템에 Load Balancer를 설치할 때 로드 밸런스됩니다.

백업. Dispatcher의 고가용성에서 기본 시스템의 상태. 기본 시스템의 상태를 모니터하고, 필요에 대신합니다. 고가용성 및 기본을 참조하십시오.

별명. 서버에 지정되는 추가 이름. 별명을 사용하면 서버가 호스트 시스템의 이름과 독립적으로 사용할 수 있게 됩니다. 별명은 도메인 이름 서버에서 정의해야 합니다.

사

사설 네트워크. 성능상의 이유로 Dispatcher가 클러스터된 서버와 통신하는 별도의 네트워크.

사이트 이름. 사이트 이름은 클라이언트가 요청하는 분석할 수 없는 호스트 이름입니다. 예를 들어, 웹 사이트에는 사이트 이름 *www.dnsload.com*으로 구성된 서버가 세 개(1.2.3.4, 1.2.3.5 및 1.2.3.6) 있습니다. 클라이언트가 이 사이트 이름을 요청하면, 세 가지 서버 IP 주소 중 하나가 분석되어 리턴됩니다. 사이트 이름은 *dnsload.com*과 같이 완전한 도메인 이름이어야 합니다. 예를 들어, 완전하지 않은 이름, *dnsload*는 사이트 이름으로 유효하지 않습니다.

상호 고가용성. 상호 고가용성을 통해 두 Dispatcher 시스템은 서로에 대해 기본 및 백업이 될 수 있습니다. 관련 주제: 백업, 고가용성, 기본.

서버. 네트워크를 거쳐 다른 컴퓨터에 공유 서비스를 제공하는 컴퓨터. 예로는 파일 서버, 인쇄 서버 또는 메일 서버 등이 있습니다.

서버 시스템. Dispatcher가 다른 서버를 하나의 가상 서버로 그룹화하는 서버. Dispatcher는 서버 시스템 간의 통신량 밸런스를 조정합니다. 클러스터된 서버의 동의어입니다.

서버 주소. 네트워크에서 공유 서비스를 다른 컴퓨터에 제공하는 각 컴퓨터에 지정되는 고유한 코드. 예로는 파일 서버, 인쇄 서버 또는 메일 서버가 있습니다. 서버 주소는 IP 주소 또는 호스트 이름일 수 있습니다.

서브넷 마스크. IPv4의 경우, IP 주소의 호스트 부분에서 서브네트워크 주소 비트를 식별하는 데 사용되는 32비트 마스크

서비스. (1) 하나 이상의 노드에서 제공되는 기능으로 HTTP, FTP, Telnet 등이 있습니다. (2) Nortel Alteon 제어기의 경우, 서비스는 사이트에서 일반 사용자가 요구하는 기능 또는 정보입니다. 일반 사용자 요청에서는 가상 IP 주소 및 가상 포트 번호로 식별됩니다. 스위치에서는 정수인 가상 서버 ID 및 가상 포트 번호 또는 서비스 이름으로 식별됩니다. (3) Cisco CSS Consultant의 경우, 서비스는 콘텐츠가 실제로 상주하는 대상 위치입니다. 예를 들면, 로컬 또는 원격 서버와 포트입니다.

소스 주소. Dispatcher 고가용성에서 하트 비트를 전송하는 고가용성 상대 시스템의 주소.

소유자 콘텐츠. 소유자 이름 및 소유자에 대한 콘텐츠 규칙을 나타내며 둘 다 Cisco CSS Switch에 정의됩니다.

셸. 사용자의 워크스테이션에서 명령행을 승인하고 처리하는 소프트웨어. bash 셸은 사용 가능한 몇 가지 UNIX 셸 중 하나입니다.

실행 프로그램. 여러 Load Balancer 기능 중 하나 실행 프로그램은 요청을 TCP나 UDP 서버에 라우트하고, 새로운 연결, 활성 연결 및 종료된 연결의 수를 모니터하여 완료되거나 재설정된 연결의 가비지 컬렉션을 수행합니다. 실행 프로그램은 관리자 기능에 새로운 연결과 활성 연결을 제공합니다.

아

어드바이저. 어드바이저는 Load Balancer의 기능입니다. 어드바이저는 각각의 서버의 피드백을 수집하여 분석한 후 관리자 기능에 알립니다.

에이전트. (1) 시스템 관리에서 특정 상호작용을 위해 에이전트 역할이 가정되는 사용자 (2) (a) 오브젝트 관련 통지를 보내고 (b) 오브젝트를 수정 또는 조회하는 관리 조작에 대해 관리자의 요청을 처리함으로써 관리되는 하나 이상의 오브젝트를 나타내는 엔티티.

연결 표시. 서버가 새로운 연결을 수신할 수 있도록 하는 것.

완료 상태(FIN state). 종료된 트랜잭션의 상태. 트랜잭션이 완료 상태가 되면, Load Balancer 가비지 컬렉터는 연결에 예약된 메모리를 지울 수 있습니다.

우선순위. 규칙 기반 로드 밸런싱에서 제공된 규칙에 대한 중요성 레벨. Dispatcher는 첫 번째 우선순위 레벨에서 마지막 우선순위 레벨까지 평가합니다.

웹. 프로그램과 파일, HTTP 서버의 다른 문서에 대한 링크를 포함하는 많은 하이퍼 텍스트 문서를 포함하는 HTTP 서버의 네트워크. 월드 와이드 웹(WWW)이라고 합니다.

이더넷. LAN의 표준 유형. 사전에 조정하지 않아도 여러 스테이션에서 전송 매체에 액세스할 수 있으며 반송자 감지를 사용하여 회선 경합을 피하고 충돌 검출과 전송을 사용하여 회선 경합을 해결합니다. 이더넷 시스템에서 사용되며, TCP/IP를 포함하는 소프트웨어 프로토콜.

인터넷. 프로토콜의 인터넷을 사용하고 공용 액세스를 허용하는 전세계의 상호 연결된 네트워크 모음.

인트라넷(intranet). 인터넷 표준과 응용프로그램(웹 브라우저와 같은)을 조직의 기존 컴퓨터 네트워킹 기본 구조와 통합한 보안된 개인 네트워크.

자

작업중지. 조작이 정상적으로 완료되도록 프로세스를 종료하는 것.

전략어. Dispatcher 고가용성에서 활성 시스템의 장애 후에 수행할 복구 방법을 지정하는 키워드.

점분리 10진수 표기법. 기본 10으로 기록되고 마침표(점)로 구분되는 네 개의 8비트 숫자로 이루어진 32비트 정수의 구문 표시. IPv4 주소를 나타내는 데 사용됩니다.

제어기. 둘 이상의 컨설턴트의 모음.

제한시간. 발생한 조작에 할당된 시간 간격.

주소. 네트워크에 연결되어 있는 각 장치나 워크스테이션에 지정되는 고유한 코드. 표준 IPv4 주소는 두 개의 파트를 포함하는 32비트 주소 필드입니다. 첫 번째 파트는 네트워크 주소이고, 두 번째 파트는 호스트 번호입니다. IPv6 주소는 IPv4보다 훨씬 많은 수의 주소를 지원하는 128비트 주소 필드입니다. 멀티캐스트 및 애니캐스트 주소 지정 같은 추가 기능도 지원합니다.

차

최대 범위. 규칙 기반 로드 밸런싱에서 하나의 규칙에 지정된 상위 값. 이 값의 기본값은 규칙 유형에 따라 달라집니다.

최소 범위. 규칙 기반 로드 밸런싱에서 하나의 규칙에 지정된 하위 값. 이 값의 기본값은 규칙 유형에 따라 달라집니다.

카

클라이언트. 다른 컴퓨터 시스템이나 프로세스의 서비스를 요청하는 컴퓨터 시스템이나 프로세스. 예를 들어, Lotus Domino® Go Webserver로부터 HTML 문서를 요청하는 워크스테이션이나 PC는 해당 서버의 클라이언트입니다.

클러스터. Dispatcher에서 같은 목적으로 사용되며 단일 도메인에 의해 식별되는 TCP 또는 UDP 서버의 그룹. 셀도 참조하십시오.

클러스터 주소. Dispatcher에서 클라이언트가 연결되는 주소.

클러스터된 서버. Dispatcher가 다른 서버를 하나의 가상 서버로 그룹화하는 서버. []는 이 클러스터된 서버 사이에서 TCP 또는 UDP 통신량의 밸런스를 조정합니다.

과

패킷. 인터넷이나 기타 패킷 교환 네트워크에서 기점 및 대상 사이에서 라우트되는 자료의 단위.

포트간 연관 관계. 포트간 연관 관계는 여러 포트로 확장되는 연관 관계 기능입니다. 관련 주제: 결합 시간.

포트(port). 추상적인 통신 장치를 식별하는 번호. 웹 서버는 기본적으로 포트 80을 사용합니다.

하

하트 비트. 활성 Load Balancer의 상태를 모니터링하기 위해 대기 Load Balancer가 사용하는 고가용성 모드의 두 Load Balancer 시스템 간에 전송되는 간단한 패킷.

호스트. 네트워크에 연결되어 있는 컴퓨터로, 해당 네트워크에 대한 액세스 지점을 제공합니다. 호스트는 클라이언트, 서버 또는 동시에 둘 모두가 될 수 있습니다.

호스트 이름. 호스트에 지정되는 기호 이름. 호스트 이름은 도메인 이름 서버를 통해 IP 주소로 분석됩니다.

확장 가능. 사용, 볼륨 또는 요구의 집중을 더 크게 하거나 더 적게 하기 위해 쉽게 적용할 수 있는 시스템의 능력에 관련된 용어. 예를 들면, 확장 가능한 시스템은 다양한 복잡성을 갖는 작업을 수행하는 보다 크거나 보다 작은 네트워크에 대해 작동되도록 효율적으로 적용할 수 있습니다.

숫자

2진 로그. 서버 정보를 2진 파일에 저장하고, 처리하여 계속 수집되는 서버 정보를 분석할 수 있습니다.

A

ACK. 연속 공백이 없는 제어 비트(응답 문자)로서, 이는 이 세그먼트의 응답 필드가 이 세그먼트의 송신자가 수신할 것으로 예상하고 있는 다음 순서 번호를 지정하여 모든 이전 순서 번호들의 수신에 응답함으로써 나타냅니다.

API. 응용프로그램 프로그래밍 인터페이스. 응용프로그램이 운영 체제와 다른 서비스를 액세스할 때 사용하는 인터페이스(규약이라고 함). API는 소스 코드 레벨에서 정의되며 응용프로그램과 커널(또는 권한 있는 다른 유틸리티) 간의 분리 레벨을 제공하여 코드의 이식성을 보장합니다.

C

Caching Proxy. 고효율적인 캐시 설계를 통해 일반 사용자 응답 시간 속도를 빠르게 하는 데 도움을 줄 수 있는 Caching Proxy 서버. 융통성 있는 PICS 필터링은 하나의 중앙 위치에서 웹 기반 정보에 대한 네트워크 관리자 제어 액세스를 돕습니다.

CBR. Content Based Routing. Load Balancer의 컴포넌트. CBR은 지정된 규칙 유형을 사용하는 웹 페이지 콘텐츠에 따라 HTTP 또는 HTTP 서버에 대한 수신 요청을 로드 밸런싱하기 위해 Caching Proxy와 함께 작동합니다.

cbrcontrol. Load Balancer의 콘텐츠 기반 라우터 컴포넌트에 인터페이스를 제공합니다.

cbrserver. 콘텐츠 기반 라우터에서 실행 프로그램, 관리자 및 어드바이저에 대한 명령행의 요청을 처리합니다.

ccocontrol. Cisco CSS Controller에서 Cisco CSS Switch에 인터페이스를 제공합니다.

ccoserver. Cisco CSS Controller에서 컨설턴트에 대한 명령행의 요청을 처리합니다.

CGI. 공통 게이트웨이 인터페이스. 웹 서버와 외부 프로그램 사이의 정보 교환을 위한 표준. 외부 프로그램은 운영 체제에서 지원되는 어떤 언어로도 작성될 수 있으며, 양식 처리와 같이 서버에서 대개 수행되지 않는 작업을 수행합니다.

CGI 스크립트(CGI script). 양식 처리와 같이 대개 서버에서 수행되지 않는 작업을 수행하기 위해 공통 게이트웨이 인터페이스를 사용하는 Perl이나 REXX와 같은 스크립트 언어로 작성되는 CGI 프로그램.

Cisco CSS Controller. IBM Load Balancer의 컴포넌트. Cisco CSS Controller는 Load Balancer 기술을 사용하여 실시간 로드 밸런싱 정보를 Cisco Content Services Switch에 제공합니다.

Cisco CSS Switch. 패킷 전달 및 콘텐츠 경로 지정에 사용되는 Cisco의 CSS Switch 11000 시리즈.

consultant. 로드 밸런싱 중인 서버에서 서버 메트릭을 수집하여 서버 가중치 정보를 로드 밸런싱을 수행하는 스위치로 전송합니다.

D

Dispatcher. 링크된 각각의 서버 그룹 사이에서 TCP 또는 UDP 통신량 밸런싱을 효율적으로 조정하는 Load Balancer의 컴포넌트. Dispatcher 시스템은 Dispatcher 코드를 실행 중인 서버입니다.

dscontrol. Load Balancer의 Dispatcher 컴포넌트에 인터페이스를 제공합니다.

dsserver. Dispatcher에서 실행 프로그램, 관리자 및 어드바이저에 대한 명령행의 요청을 처리합니다.

F

FIN. 하나의 순서 번호를 차지하는 제어 비트(finis)로서, 송신자가 더 이상 데이터를 전송하지 않거나 차지하는 순서 공간을 제어하지 않는다는 것을 나타냅니다.

FQDN. 완전한 도메인 이름. 로컬 호스트 이름과 도메인 이름으로 구성되고 최상위 레벨 도메인(tld)이 포함된 시스템의 전체 이름. 예를 들어, "venera"는 호스트 이름이고 "venera.isi.edu"는 FQDN입니다. FQDN은 인터넷의 호스트에 고유한 인터넷 주소를 판별할 수 있어야 합니다. "이름 분석"이라고 하는 이 프로세스는 DNS(Domain Name System)를 사용합니다.

FTP(File Transfer Protocol). 네트워크 컴퓨터 사이에서 파일을 전송하는 데 사용되는 응용프로그램 프로토콜. FTP에서는 원격 호스트 시스템의 파일에 액세스할 수 있는 사용자 ID와 가끔 암호가 필요합니다.

G

GRE. 일반 경로 지정 캡슐화. 패킷을 GRE 패킷 안에 캡슐화하여 임의의 네트워크 프로토콜 A를 다른 임의 프로토콜 B에서 전송한 다음에 B 패킷 안에 포함되는 프로토콜.

H

HTML(Hypertext Markup Language). 하이퍼 텍스트 문서를 작성하는 데 사용되는 언어. 하이퍼 텍스트 문서에는 강조표시된 용어나 주제에 관한 추가 정보를 포함하는 다른 문서에 대한 링크도 포함됩니다. HTML은 텍스트의 형식과 양식 입력 영역의 위치를 제어하고, 예를 들면 경로 선택 가능한 링크도 제어합니다.

HTTP(Hypertext Transfer Protocol). 하이퍼 텍스트 문서를 전송하고 표시하는 데 사용되는 프로토콜.

HTTPS(Hypertext Transfer Protocol, Secure). SSL을 사용하여 하이퍼 텍스트 문서를 전송하고 표시하는 데 사용되는 프로토콜.

I

ICMP. 인터넷 제어 메시지 프로토콜. 인터넷에 대한 호스트 서버와 게이트웨이 사이의 메시지 제어 및 오류 보고 프로토콜.

IMAP. 인터넷 메시지 액세스 프로토콜, 클라이언트가 서버의 전자 메일 메시지를 액세스하고 처리할 수 있게 하는 프로토콜. 이 프로토콜은 로컬 우편함과 동일한 기능을 함으로써 원격 메시지 폴더(우편함)의 처리를 가능하게 합니다.

IP. 인터넷 프로토콜. 네트워크나 상호 연결된 네트워크를 통해 데이터를 라우트하는 무연결 프로토콜. IP는 상위 프로토콜 계층과 물리적 계층 사이의 중간 매체의 역할을 합니다.

IP 주소(IP address). 인터넷 프로토콜 주소. 네트워크에서 워크스테이션 또는 각 장치의 실제 위치를 지정하는 고유한 주소. 인터넷 주소라고도 합니다.

IPSEC. 인터넷 프로토콜 보안. 네트워크 통신의 네트워크 또는 패킷 처리 계층에서의 보안을 위한 개발 표준.

L

LAN. 근거리 통신망. 통신을 위해 제한된 지리적 영역 내에서 연결되어 있으며 더 큰 네트워크로 연결될 수 있는 컴퓨터 네트워크 장치.

M

MAC 주소. 매체 액세스 제어 주소. 공유 네트워크 매체에 연결된 장치의 하드웨어 주소.

Metric Server. 전에는 SMA(Server Monitor Agent)라고 했습니다. Metric server는 Load Balancer 관리자에게 시스템 고유의 메트릭을 제공합니다.

MIB. (1) 관리 정보 기준. 네트워크 관리 프로토콜을 통해 액세스할 수 있는 오브젝트 모음. (2) 호스트 또는 게이트웨이와 허용되는 조작에 사용 가능한 정보를 지정하는 관리 정보에 대한 정의.

N

nalcontrol. Load Balancer의 Nortel Alteon Controller 컴포넌트에 인터페이스를 제공합니다.

nalserver. Nortel Alteon Controller에서, 컨설턴트에 대한 명령행의 요청을 처리합니다.

nfa(비전달 주소). 관리 및 구성에 사용되는 Load Balancer 시스템의 기본 IP 주소.

NIC. 네트워크 인터페이스 카드. 네트워크에 물리적 연결을 제공하기 위해 컴퓨터에 설치하는 어댑터 회선 보드.

NNTP. 네트워크 뉴스 전송 프로토콜. 뉴스 항목을 전송하기 위한 TCP/IP 프로토콜.

Nortel Alteon Controller. IBM Load Balancer의 컴포넌트. Nortel Alteon Controller는 Load Balancer 기술을 사용하여 실시간 로드 밸런스 정보를 Nortel Alteon Web Switch에 제공합니다.

Nortel Alteon Web Switch. 패킷 전달 및 콘텐츠 경로 지정에 사용되는 Alteon Web Switching 포트폴리오의 Nortel Alteon ACE Director Series Switch 및 Nortel Alteon 180 Series Switch입니다.

P

PICS. 인터넷 콘텐츠 선택을 위한 플랫폼. PICS 작동 클라이언트로 사용자는 사용할 비율 서비스와 각 비율 서비스마다 승인할 수 있는 비율 및 승인할 수 없는 비율을 결정할 수 있습니다.

ping. ICMP(인터넷 제어 메시지 프로토콜) 에코 요청(echo-request) 패킷을 응답 수신을 예상하는 호스트, 게이트웨이 또는 라우터에 전송하는 명령.

POP3. 우편 사무 프로토콜 3. 네트워크 메일을 교환하고 우편함에 액세스하는 데 사용되는 프로토콜.

protocol. 통신 발생시, 통신 시스템의 기능적인 단위에 대한 조작을 관리하는 규칙 세트. 프로토콜은 바이트에서 비트가 전송되는 순서와 같은 시스템 대 시스템 인터페이스의 하위 레벨 세부사항을 결정할 수 있으며, 파일 전송과 같은 응용프로그램간의 상위 레벨 교환도 판별할 수 있습니다.

Q

QoS(Quality of Service). 처리량, 전환 지연 및 우선순위를 포함하여 네트워크 서비스의 성능 등록 정보. 일부 프로토콜을 사용하면 패킷이나 스트림에 QoS 요구사항이 포함될 수 있습니다.

R

RMI. 원격 메소드 호출. 한 컴퓨터에서 실행되는 Java 프로그램이 다른 컴퓨터에서 실행되는 또다른 Java 프로그램의 메소드와 오브젝트를 액세스할 수 있게 해주는 Java 프로그래밍 언어 라이브러리의 부분.

RPM. Red Hat 패키지 관리자.

S

Site Selector. Load Balancer의 DNS 기반 로드 밸런스 컴포넌트. Site Selector는 해당 서버에서 실행되는 Mertric Server 컴포넌트에서 수집한 단위와 가중치를 사용하여 WAN(Wide Area Network) 내에서 서버의 로드 밸런스를 조정합니다.

SMTP. 단순 메일 전송 프로토콜(Simple Mail Transfer Protocol). 프로토콜의 인터넷 모음에서 인터넷 환경의 사용자 간에 메일 전송을 위한 응용프로그램 프로토콜. SMTP는 메일 교환 순서와 메시지 형식을 지정합니다. TCP(전송 제어 프로토콜)이 기본 프로토콜이라고 가정합니다.

SNMP. 단순 네트워크 관리 프로토콜(Simple Network Management Protocol). STD 15, RFC 1157에 정의되고 IP 네트워크에서 노드를 관리하기 위해 개발된 인터넷 표준 프로토콜. SNMP는 TCP/IP로 제한되지 않습니다. 컴퓨터, 라우터, 배선 허브, 토스터 및 주크 박스를 포함한 모든 종류의 장치를 관리하고 모니터링하는 데 사용할 수 있습니다.

SPARC. 확장 가능한 프로세서 구조.

sscontrol. Load Balancer의 Site Selector 컴포넌트에 인터페이스를 제공합니다.

SSL. 보안 소켓 계층. RSA Data Security Inc.와 함께 Netscape Communications Corp.에서 개발된 일반화된 보안 설계. SSL을 사용하여 클라이언트는 서버를 인증하고 모든 데이터 및 요청을 암호화할 수 있습니다. SSL로 보호되는 보안 서버의 URL은 https(HTTP 아님)로 시작합니다.

sssserver. Site Selector에서 사이트 이름, 관리자 및 어드바이저에 대한 명령행의 요청을 처리합니다.

SYN. 하나의 순서 번호를 차지하는 수신 세그먼트 내 하나의 제어 비트로서, 연결 시작시 순서 번호가 시작되는 위치를 나타내는 데 사용됩니다.

T

TCP. 전송 제어 프로토콜. 인터넷에서 사용되는 통신 프로토콜. TCP는 신뢰성이 있는 호스트 대 호스트의 정보 교환을 제공합니다. 기본 프로토콜로 IP를 사용합니다.

TCP 서버 시스템. Load Balancer가 다른 서버에 대해 하나의 가상 서버로 링크하는 서버. []는 TCP 서버 시스템 사이에 TCP 통신량 밸런스를 유지합니다. 클러스터된 서버의 동의어입니다.

TCP/IP. 전송 제어 프로토콜/인터넷 프로토콜. 각 네트워크에서 사용되는 통신 기법에 관계 없이 네트워크 사이의 통신을 허용하도록 설계된 프로토콜 모음.

Telnet. 터미널 에뮬레이션 프로토콜로 원격 연결 서비스를 위한 TCP/IP 응용프로그램 프로토콜. Telnet을 사용하여 사용자의 워크스테이션이 해당 원격 호스트에 직접 연결된 것처럼 한 사이트에서 원격 호스트에 액세스할 수 있습니다.

TOS. 서비스 유형. SYN 패킷의 IP 헤더에 있는 1 바이트 필드.

TTL. DNS TTL(지속 시간)은 클라이언트가 이름 분석 응답을 캐시할 수 있는 시간(초 단위)입니다.

U

UDP. 사용자 데이터그램 프로토콜. 프로토콜의 인터넷 모음에서 신뢰할 수 없는, 무연결 데이터그램 서비스를 제공하는 프로토콜. 이것으로 하나의 시스템이나 프로세스의 응용프로그램이 데이터그램을 다른 시스템이나 프로세스의 응용프로그램으로 전송할 수 있습니다. UDP는 IP(인터넷 프로토콜)를 사용하여 데이터그램을 전달합니다.

URI. 범용 자원 ID. HTML 문서, 이미지, 비디오 클립, 프로그램 등 웹 자원의 인코딩된 주소.

URL. URL(동일 자원 위치 지정자). 인터넷에서 웹 페이지와 같은 오브젝트의 위치를 지정하는 표준 방식. URL은 World Wide Web에 사용되는 주소 양식입니다. 다른 HTML 문서(다른 컴퓨터에 저장되어 있는)에서 하이퍼링크의 목표를 지정하는 데 사용되기도 합니다.

V

VPN. 가상 사설 네트워크(VPN). 둘 이상의 네트워크를 연결하는 하나 이상의 보안 IP 터널로 구성되는 네트워크.

W

WAN. 광역 네트워크. 근거리 통신망 또는 대도시 통신망에서 제공하는 것보다 지리적으로 더 광범위한 지역에 통신 서비스를 제공하고, 공용 통신 설비를 사용하거나 제공하는 네트워크.

WAP. 무선 응용프로그램 프로토콜. 핸드폰의 인터넷 액세스와 같은 무선 통신을 사용하는 응용프로그램의 개방형 국제 표준.

WAS. WebSphere Application Server.

WLM. 작업로드 관리자. Dispatcher와 함께 제공되는 어드바이저. WLM(MVS Workload Manager) 컴포넌트를 실행하는 OS/390 메인프레임의 서버하고만 작동하도록 설계되었습니다.

색인

[가]

가중치

관리자 설정 방법 196

설정

서버용 423, 449

포트상의 모든 서버 경계 195, 409

제어기 265

가중치 갱신에 대한 감도, 설정 197, 401, 440, 442

간격, 빈도 설정

관리자에서 실행 프로그램 조회 197, 400

관리자에서 실행 프로그램에 대한 가중치

갱신 197, 400, 439, 441

어드바이저가 서버 조회 375, 433

감도 임계치 266

개요

CBR 구성 123

Cisco CSS Controller 구성 163

Dispatcher 컴포넌트 구성 69

Nortel Alteon Controller 구성 185

Site Selector 구성 145

개인용 키

원격 인증용 282

검사

구성 168, 190

결합 배치

Cisco CSS Controller 261

IPv6 고려사항 94

Nortel Alteon Controller 261

결합 배치(키워드) 217, 423

결합 배치, Load Balancer 및 서버 72, 78, 216, 246, 419, 423

결합 배치, Load Balancer 및 클라이언트 259

결합(연관 관계)

결합(포트 연관 관계 무시) 234, 419

사용법 236

수동 쿠키 240, 242, 414

연관 관계 주소 마스크 238

지금 작업중지 239, 398, 402

포트 연관 관계 무시 234

포트간 연관 관계 237, 238, 404

활성 쿠키 239, 240, 414

결합(연관 관계) (계속)

stickymask 237, 238, 405

stickytime 62, 236, 237, 405, 414

URI 240, 414

경보

제어기 276

Dispatcher, CBR, Site Selector 198

계획

CBR 117

Cisco CSS Controller 157

Dispatcher 컴포넌트 57

Nortel Alteon Controller 175

Site Selector 141

고가용성 5, 6, 66, 218

구성 168, 190, 219

기본 호스트 381

상호 67, 220, 381, 382, 393

스크립트 223

goActive 224

goldle 224

goInOp 224

goStandby 224

highavailChange 225

Cisco CSS Controller 261

dscontrol 391, 487

IPv6 고려사항 93

nat 전달 224

ndcontrol 466

Nortel Alteon Controller 261

primaryhost 382

S/390용 Linux 225

공용 키

원격 인증용 282

관리자

고정 가중치 196

버전 402, 441, 443

시작 78, 401, 440, 442

정지 402, 441, 443

cbrcontrol 397

dscontrol 397

proportions 194

sscontrol 439

광역 지원 244

구성 예제 248

광역 지원 (계속)

원격 어드바이저 사용 245

원격 Dispatcher 사용 244

GRE 사용 250

Linux 251

구문 다이어그램

구두점 367

기호 367

매개변수 367

예제 367

읽기 367

구성

검사 168, 190

고가용성 168, 190

메트릭 167, 189

방법

마법사(CBR) 127

마법사(Dispatcher) 72

마법사(Site Selector) 148

명령행(CBR) 124

명령행(Cisco CSS Controller) 164

명령행(Dispatcher) 70

명령행(Nortel Alteon

Controller) 185

명령행(Site Selector) 146

스크립트(CBR) 125

스크립트(Cisco CSS Controller) 165

스크립트(Dispatcher) 70

스크립트(Nortel Alteon

Controller) 186

스크립트(Site Selector) 146

GUI(CBR) 126

GUI(Cisco CSS Controller) 165

GUI(Dispatcher) 71

GUI(Nortel Alteon Controller) 187

GUI(Site Selector) 147

서비스 189

스위치 컨설턴트 정의 189

예제 파일 511

컨설턴트 시작 168, 189

타스크, 고급 193, 215

확인 84

cbrwizard 127

Cisco CSS Controller 163

구성 (계속)

Content Based Routing 123
Dispatcher 컴포넌트 69
dswizard 72
Nortel Alteon Controller 185
Site Selector 145
sswizard 148

규칙

cbrcontrol 411
dscontrol 411
sscontrol 446

규칙 기반 로드 밸런스 226

공유 대역폭 230, 231, 412, 417
규칙 선택, 컴포넌트로 226
서버 평가 옵션 235
서비스 유형(TOS) 229, 412, 416
시간 229, 412, 416, 446, 448
예약된 대역폭 230, 231, 412, 417
요청 콘텐츠 61, 234, 412
초당 연결 229, 412
클라이언트 포트 228, 412
클라이언트 IP 주소 228, 412, 416, 446, 448
평가 옵션 235
포트에 대한 활성 연결 230, 412
항상 참 233, 412, 416, 446, 448
Metric All 232
metric average 233
metricall 446
metricavg 446

근접 옵션 144

기본 호스트 220

[나]

내게 필요한 옵션 xvii

네트워크 근접 144

[다]

다중 주소 결합 배치 78

단절, 서버 표시 423, 449, 450

[라]

라우트, 여분 삭제 84

라우트, 여분의 83

로드 밸런스 설정(최적화) 194, 264

로드 밸런스의 중요성 비율, 설정 195, 382

루프백

Linux 별명 지정 대안 85

루프백 장치

별명 79

[마]

마법사, 구성

CBR 127

Dispatcher 72

Site Selector 148

메트릭

구성 167, 189

명령

라우트

여분의 라우트 삭제 83, 84

cbrcontrol

관리자 397

규칙 411

서버 418

advisor 372

binlog 378

cluster 379

executor 383

file 388

help 390

host 395

logstatus 396

metric 403

port 404

set 425

status 426

cococontrol

서버, 구성 475

프롬프트 457

consultant 458, 461

file 463

help 465

host 472

metric 470

Cisco CSS Controller 457

dscontrol

고가용성, 제어 391, 487

관리자 397

관리자 제어 78

규칙 411

비전달 주소 정의 75, 387

명령 (계속)

dscontrol (계속)

서버 418

서버 정의 78

서브에이전트, SNMP 구성 427

어드바이저 제어 78

포트 정의 77

프롬프트 370

advisor 372

binlog 378

cluster 379

executor 383

file 388

help 390

host 395

logstatus 396

metric 403

port 404

set 425

status 426

ifconfig 77, 247

루프백 장치에 별명 지정 80

nalcontrol

서버, 구성 493

컨설턴트 478

프롬프트 477

consultant 482

file 484

help 486

host 495

metric 491

ndcontrol

고가용성, 제어 466

netstat

IP 주소와 별명을 확인 83

Nortel Alteon Controller 477

Site Selector 429

sscontrol

관리자 439

규칙 446

advisor 430

file 435

help 437

logstatus 438

metric 444

nameserver 445

server 449

set 451

명령 (계속)

sscontrol (계속)

sitename 452

status 455

명령어 참조서

읽는 방법 367

명령행

구성 예제

CBR 110

Cisco CSS Controller 154

Dispatcher 51

Nortel Alteon Controller 173

Site Selector 138

명령 전송(GUI) 504

명시적 링크 252

모니터 메뉴 옵션 289

문제점 진단

갱신 후 lbadm이 서버로부터 연결이 끊어
짐 330

고가용성 구성에서 기본 시스템 및 백업 시
스템 활성화 339

고가용성 사용 시 IP 주소 충돌 339

고가용성, 구성 팁 340

공동 문제점 및 해결책 323, 325, 348,
351, 354, 356, 359

구문 또는 구성 오류 349

네트워크 정지 후 고가용성 설정에서 어드
바이저가 작동하지 않음(Windows) 336
대형 페이지 응답 리턴 시 클라이언트 요청
실패 339

도움말 패널이 사라짐 327

로컬 주소 대신 별명 리턴 331

루프백 별명을 지정하기 위해 IP address
add 명령을 사용하지 마십시오
(Linux) 337

서버 로드를 등록하지 않음 332

서비스 수정사항 설치 시 Java 경고 메시지
가 나타남 347

설치가 공급된 Java 업그레이드 347

손상된 Latin-1 자국 문자가 표시됨
(Windows) 334, 350, 353, 356, 359

스위치에 의해 가중치가 갱신되지 않음
355, 358

어드바이저가 작동하지 않음 324

어드바이저와 도달 목표에서 모든 다운된
서버를 표시함(Windows) 335, 350,
353

문제점 진단 (계속)

온라인 도움말을 보려고 할 때 오류 메시지
발생 326

요청이 로드 밸런스되지 않음 348

원격 연결을 통해 분석되지 않는 IP 주소
330

응답 시간이 느림 332

추가 라우트 324

컨설턴트 연결 오류 355, 358

큰 구성 파일을 로드할 때의 예기치 못한
작동 329

포트 메소드에 대해 지정되지 않은 또는 올
바르지 않은 라우터 주소 337

포트 13099에서 레지스트리를 작성할 수
없음 355

포트 14099에서 레지스트리를 작성할 수
없음 357

하트 비트를 추가할 수 없음 324

호스트 이름에 대한 IP 주소를 확인하는 중
에 문제점 발생(Windows) 335, 350
호스트로부터 연결이 끊어짐, 웹 관리 사용
333, 349, 353, 356, 358

0.0.0.0에 웹 서버 바인딩 333

2계층 구성에서 Metric Server를 구성함
361

AIX 및 Linux에서 원치 않는 한글 글꼴
330

AIX에서 ps -vg 명령 출력이 손상됨 360

Caching Proxy가 설치된 Dispatcher 실행
중 오류 327

cbrcontrol 또는 lbadm 명령 실패 348

CBR에서 사용되는 포트 번호 319

CBR이 실행되지 않음 348

cococontrol 또는 lbadm 명령 실패 354
ccoserver가 시작되지 않음 354

Cisco CSS Controller에서 사용하는 포트
번호 321

Discovery 경로로 인해 []와의 리턴 통신이
발생하지 못함 328

Dispatcher 고가용성이 작동되지 않음
324

Dispatcher 및 서버가 응답하지 않음 323

Dispatcher 요청이 라우트되지 않음 323

Dispatcher가 수행되지 않음 323

Dispatcher에서 사용되는 포트 번호 319

Dispatcher, Microsoft IIS 및 SSL이 작동
하지 않음 325

dscontrol 또는 lbadm 명령 실패 325

문제점 진단 (계속)

GUI가 올바르게 시작되지 않음 326

GUI가 올바르게 표시되지 않음 327

Java 메모리/스레드 오류(HP-UX) 334,
359

Java 메모리/스레드 오류(HP-UX) 350,
353, 356

Linux 시스템에서 metric server로부터 값
을 검색할 수 없음 362

Linux에서 관리자 및 어드바이저 사용 시
메모리 누수가 발생 344

Linux에서 HA Dispatcher 동기화가 실패
할 수 있음 340

Linux에서는 Dispatcher가 패킷을 전달하지
만 백엔드 서버는 이를 수신하지 못함
345

Linux의 zSeries 또는 S/390 서버 사용 시
제한사항 342

Load Balancer 구성 로드 시 지연 338

Load Balancer 실행 프로그램을 시작할 때
파란색 화면이 표시됨 328

Load Balancer 프로세스 종료
(Solaris) 338

Load Balancer가 프레임을 처리하고 전달
할 수 없음 327

Load Balancer의 광역 모드에서 고가용성
이 작동되지 않음 329

Matrox AGP 카드에서 예상하지 못한

GUI 작동 331, 349, 352, 355, 358

Metric Server 로그에서 "에이전트에 액세스
하려면 서명이 필요합니다."라고 보고합
니다. 360

Metric Server 시작 후 메트릭 값 -1 리턴
363

Metric Server로드를 보고하지 않음 360

nalcontrol 또는 lbadm 명령 실패 357
nalserver가 시작되지 않음 356

Nortel Alteon Controller에서 사용하는 포
트 번호 322

refresh 명령이 구성을 갱신하지 않음
356, 359

Site Selector가 라운드 로빙하지 않음
(Solaris) 351

Site Selector가 실행되지 않음 351

Site Selector가 올바르게 로드 밸런스하지
않음 352

Site Selector에서 사용하는 포트 번호
320

문제점 진단 (계속)

Solaris에서 스크립트가 원치 않는 콘솔 메시지를 생산 362

Solaris에서 cbrcontrol 실패 349

Solaris에서 IPv6 서버를 구성에 추가할 수 없음 347

sscontrol 또는 lbadm 명령 실패 351

ssserver가 Windows에서 시작에 실패함 352

Windows 시스템의 고가용성 인계 문제 346

Windows에서 "서버가 응답하지 않음" 오류 발생 340

Windows의 Metric Server
IOException 359

"rmmod ibmlb"에서의 예상치 않은 작동 332

문제점 해결 303

갱신 후 lbadm이 서버로부터 연결이 끊어짐 330

고가용성 구성에서 기본 시스템 및 백업 시스템 활성화 339

고가용성 사용 시 IP 주소 충돌 339

고가용성, 구성 팀 340

공동 문제점 및 해결책 323, 325, 348, 351, 354, 356, 359

구문 또는 구성 오류 349

네트워크 정지 후 고가용성 설정에서 어드바이저가 작동하지 않음(Windows) 336

대형 페이지 응답 리턴 시 클라이언트 요청 실패 339

도움말 패널이 사라짐 327

로컬 주소 대신 별명 리턴 331

루프백 별명을 지정하기 위해 IP address add 명령을 사용하지 마십시오 (Linux) 337

서버 로드를 등록하지 않음 332

서비스 수정사항 설치 시 Java 경고 메시지가 나타남 347

설치가 공급된 Java 업그레이드 347

손상된 Latin-1 자국 문자가 표시됨 (Windows) 334, 350, 353, 356, 359

스위치에 의해 가중치가 갱신되지 않음 355, 358

어드바이저가 작동하지 않음 324

어드바이저와 도달 목표에서 모든 다운된 서버를 표시함(Windows) 335, 350, 353

문제점 해결 (계속)

온라인 도움말을 보려고 할 때 오류 메시지 발생 326

요청이 로드 밸런스되지 않음 348

원격 연결을 통해 분석되지 않는 IP 주소 330

응답 시간이 느림 332

추가 라우트 324

컨설턴트 연결 오류 355, 358

큰 구성 파일을 로드할 때의 예기치 못한 작동 329

포트 메소드에 대해 지정되지 않은 또는 올바르게 않은 라우터 주소 337

포트 13099에서 레지스트리를 작성할 수 없음 355

포트 14099에서 레지스트리를 작성할 수 없음 357

하트 비트를 추가할 수 없음 324

호스트 이름에 대한 IP 주소를 확인하는 중에 문제점 발생(Windows) 335, 350

호스트로부터 연결이 끊어짐, 웹 관리 사용 333, 349, 353, 356, 358

0.0.0.0에 웹 서버 바인딩 333

2계층 구성에서 Metric Server를 구성함 361

AIX 및 Linux에서 원치 않는 한글 글꼴 330

AIX에서 ps -vg 명령 출력이 손상됨 360

Caching Proxy가 설치된 Dispatcher 실행 중 오류 327

cbrcontrol 또는 lbadm 명령 실패 348

CBR에서 사용되는 포트 번호 319

CBR이 실행되지 않음 348

cococontrol 또는 lbadm 명령 실패 354

ccoserver가 시작되지 않음 354

Cisco CSS Controller에서 사용하는 포트 번호 321

Discovery 경로로 인해 []와의 리턴 통신이 발생하지 못함 328

Dispatcher 고가용성이 작동되지 않음 324

Dispatcher 및 서버가 응답하지 않음 323

Dispatcher 요청이 라우트되지 않음 323

Dispatcher가 수행되지 않음 323

Dispatcher에서 사용되는 포트 번호 319

Dispatcher, Microsoft IIS 및 SSL이 작동하지 않음 325

dscontrol 또는 lbadm 명령 실패 325

문제점 해결 (계속)

GUI가 올바르게 시작되지 않음 326

GUI가 올바르게 표시되지 않음 327

Java 메모리/스레드 오류(HP-UX) 334, 359

Java 메모리/스레드 오류(HP-UX) 350, 353, 356

Linux 시스템에서 metric server로부터 값을 검색할 수 없음 362

Linux에서 관리자 및 어드바이저 사용 시 메모리 누수가 발생 344

Linux에서 HA Dispatcher 동기화가 실패할 수 있음 340

Linux에서는 Dispatcher가 패킷을 전달하지만 백엔드 서버는 이를 수신하지 못함 345

Linux의 zSeries 또는 S/390 서버 사용 시 제한사항 342

Load Balancer 구성 로드 시 지연 338

Load Balancer 실행 프로그램을 시작할 때 파란색 화면이 표시됨 328

Load Balancer 프로세스 종료 (Solaris) 338

Load Balancer가 프레임을 처리하고 전달할 수 없음 327

Load Balancer의 광역 모드에서 고가용성이 작동되지 않음 329

Matrox AGP 카드에서 예상하지 못한 GUI 작동 331, 349, 352, 355, 358

Metric Server 로그에서 "에이전트에 액세스하려면 서명이 필요합니다."라고 보고합니다. 360

Metric Server 시작 후 메트릭 값 -1 리턴 363

Metric Server로드를 보고하지 않음 360

nalcontrol 또는 lbadm 명령 실패 357

nalserver가 시작되지 않음 356

Nortel Alteon Controller에서 사용하는 포트 번호 322

refresh 명령이 구성을 갱신하지 않음 356, 359

Site Selector가 라운드 로빙하지 않음 (Solaris) 351

Site Selector가 실행되지 않음 351

Site Selector가 올바르게 로드 밸런스하지 않음 352

Site Selector에서 사용하는 포트 번호 320

문제점 해결 (계속)

Solaris에서 스크립트가 원치 않는 콘솔 메시지를 생산 362

Solaris에서 cbrcontrol 실패 349

Solaris에서 IPv6 서버를 구성에 추가할 수 없음 347

sscontrol 또는 lbadm 명령 실패 351

ssserver가 Windows에서 시작에 실패함 352

Windows 시스템의 고가용성 인계 문제 346

Windows에서 "서버가 응답하지 않음" 오류 발생 340

Windows의 Metric Server
IOException 359

"rmmod ibmlb"에서의 예상치 않은 작동 332

문제점 해결 테이블

CBR 313

Cisco CSS Controller 315

Dispatcher 컴포넌트 308

Metric Server 317

Nortel Alteon Controller 316

Site Selector 314

[바]

바인드 고유 서버 77, 78, 199, 246

방화벽(제한) 44

백업, 고가용성 66, 391, 466, 487
구성 219

버전, 표시
관리자 402, 441, 443
advisor 377, 433, 434

별명
루프백 장치 79

NIC 76, 130

보안 소켓 계층 77

비전달 주소
설정 387

정의 75

빠른 시작 예제 49

CBR 109

Cisco CSS Controller 153

Nortel Alteon Controller 171

Site Selector 137

[사]

사설 네트워크, Dispatcher에 사용 252

사용자 영역 지원 90

사용자 정의(사용자 정의 가능) 어드바이저 작성 207, 268
예제 518

사용자 종료 스크립트 198, 276

서비스 거부 감지 256

ccoallserversdown 277

ccoserverdown 276

ccoserverup 277

managerAlert 198

managerClear 198

nalallserversdown 277

naloserverup 277

nalserverdown 276

serverDown 198

serverUp 198

삭제

여분의 라우트 84

클러스터 382, 453

클러스터에서 포트 409

포트에서 서버 423, 449, 450

상태, 표시
특정 포트상의 서버 409

상표 529

상호 고가용성 67, 219, 220

스크립트 224

primaryhost 381, 382

takeover 223

새로운 기능, V6.1

결합 배치된 클라이언트 구성 7

사용자 영역 지원 7

Firefox 브라우저 지원 8

HP-UX Itanium 64비트 지원 7

kernel free 지원 7

Linux zSeries 64비트 지원 7

SIP 어드바이저 7

새로운 연결, 중요성 비율 설정 195, 380

서버

가중치 419

다운된 서버 재설정 196

리턴 주소 421

비결합(포트 연관 관계 무시) 419, 423

작업중지 239, 398, 402

address 419

advisorrequest 421

서버 (계속)

advisorresponse 422

cbrcontrol 418

collocated 419, 423

cookievalue 420

dscontrol 418

fixedweight 419

mapport 420

nat를 사용하여 결합 배치 217

protocol 420

router 420

서버 작업중지 239, 398, 400, 402

서버 통제의 2진 로그 257, 287, 288

제어기 275

서버 표시

단절 423, 449, 450

연결 423, 450

서브에이전트 286, 290

dscontrol 427

서비스

구성 189

서비스 거부 중지 감지 255

halfopenaddressreport 408

maxhalfopen 407

설정

가중치 갱신에 대한 감도 197, 401, 440, 442

간격(시간)

서버를 조회할 어드바이저 375, 433

실행 프로그램을 갱신할 관리자 197, 400, 439, 441

관리자에서 실행 프로그램을 조회하는 간격 197, 400

로그 레벨

관리자용 439

어드바이저용 285, 376, 433

로그 파일의 이름 432

관리자용 441

로그의 최대 크기

관리자용 400, 439, 441

어드바이저용 286, 376, 431, 433

로드 밸런스에서 중요성 비율 382

비전달 주소 73

서버의 가중치 400, 402, 423, 449

최대 가중치

특정 포트상의 서버 195, 409

클러스터 주소 77

smoothing index 198, 401, 440, 442

설정, 모든 글로벌 값 표시
관리자용 402, 441, 442
어드바이저용 376, 432, 434

설치

AIX 34
HP-UX 38
Linux 40
Load Balancer 33
Solaris 42
Windows에 44, 45

설치 계획 3, 9, 57, 141

설치 제거

AIX 35
HP-UX 39
Linux 40
Solaris 42
Windows에 45

소프트웨어 요구사항

Cisco CSS Controller 157
Nortel Alteon Controller 175

수동 쿠키 연관 관계 240, 242, 414

스무스 색인, 설정 198, 401, 440, 442

스위치 컨설턴트

정의 189

스크립트 223

사용자 종료 198, 276
ccoserverdown 276
goActive 224
goIdle 224
goInOp 224
goStandby 224
highavailChange 225

시스템 메트릭

중요성 비율 설정 265

시작

관리자 78, 401, 440, 442

실행 프로그램 75, 387

어드바이저 78, 375, 432, 434

CBR 110

Cisco CSS Controller 154, 299

Dispatcher 51

Metric Server 301

Nortel Alteon Controller 173, 300

server 75

Site Selector 138, 299

시작 및 정지

CBR 298

Dispatcher 288

실행 프로그램

시작 387

정지 387

[아]

어드바이저

목록 374

예제 구성 파일 518

제어기 266

빠른 실패 발견 267

사용자 정의 268

서버 수신 제한시간 267

서버 연결 제한시간 267

서버 재시도 268

sleeptime 267

조정 예제 518

CBR 컴포넌트

ssl2http 어드바이저 204

cbrcontrol 372

Dispatcher 컴포넌트 199

간격 201, 375

목록 203, 376

버전 377

보고서 제한시간 202, 375

빠른 실패 발견 202

사용자 정의 207

상태 보고서 376

서버 수신 제한시간 202, 373, 376

서버 연결 제한시간 202, 372, 375

서버 재시도 196, 203, 374

시작 78, 375

시작/정지 200

이름 372

자가 어드바이저 205, 206

정지 375

포트 380

Caching Proxy 어드바이저 204

report 377

dscontrol 372

HTTP 어드바이저 요청/응답 205

IPv6 고려사항 93

Site Selector

간격 433

목록 432, 433

버전 433, 434

보고서 제한시간 432, 434

빠른 실패 발견 202

어드바이저 (계속)

Site Selector (계속)

상태 보고서 431, 433

서버 수신 제한시간 202, 431, 433

서버 연결 제한시간 202, 430, 433

서버 재시도 203

서버 재시도 횟수 431

시작 432, 434

이름 430

정지 432, 434

포트 372, 430

interval 430

list 430

loglevel 430

Solaris 제한사항 200

sscontrol 430, 437

URL 옵션, HTTP 어드바이저 205

어드바이저, Load Balancer 컴포넌트

시작 78

연결, 서버 표시 423, 450

연결, 중요성 비율 설정 195, 382

연관 관계 주소 마스크 238, 405

연관 관계(결합)

결합(포트 연관 관계 무시) 234, 419

규칙 옵션 239

사용법 236

수동 쿠키 240, 242, 414

연관 관계 주소 마스크 238

지금 작업중지 239, 398, 402

포트 연관 관계 무시 234

포트간 연관 관계 237, 238, 404

활성 쿠키 239, 240, 414

SSL ID (cbr 전달) 62

stickymask 237, 238, 405

stickytime 62, 236, 237, 405, 414

URI 240, 243, 414

예제

로컬 서버 관리 11, 12, 14, 16, 17

빠른 시작 49

CBR 109

Cisco CSS Controller 153

Nortel Alteon Controller 171

Site Selector 137

예제 구성 파일 511

어드바이저 518

Dispatcher 컴포넌트(AIX) 511

Dispatcher 컴포넌트(Windows) 514

와일드 카드 클러스터 76, 382

와일드 카드 클러스터 (계속)

방화벽 로드 밸런스 유지 254

서버 구성 조합 253

투명 프록시의 경우 Caching Proxy 포함
254

와일드 카드 포트 77, 408

구성되지 않은 포트 통신량 경로 지정
255

FTP 통신량 처리 255

ping 어드바이저 204

요구사항

AIX 34

HP-UX 38

Linux 40

Solaris 42

Windows 44

원격 관리 37, 42, 43, 44

웹 기반 관리 281, 283

RMI 281, 282

원격 관리(웹 기반)

refresh 285

원격으로 구성 새로 고치기 285

웹 기반 관리 281, 283

refresh 285

이더넷 NIC

ibmlb.conf

Solaris용으로 구성 73

이주 33

[자]

작업로드 관리자 어드바이저(WLM) 214, 274

전달 메소드

cbr 61, 63

mac 59, 60

mac, nat 또는 cbr 62, 406

NAT 59

nat 63

정보 집계 303

정보, 집계 303

정의

비전달 주소 75, 387

클러스터 382

클러스터에 포트 77, 408

포트에 서버 78, 423, 450

정지

관리자 402, 441, 443

실행 프로그램 387

정지 (계속)

어드바이저 375, 432, 434

Cisco CSS Controller 299

Nortel Alteon Controller 300

제거

여분의 라우트 84

클러스터 382, 453

클러스터에서 포트 409

포트에서 서버 423, 449, 450

제어기

고정 가중치 265

로드 밸런스 설정

가중치 265

감도 임계치 266

메트릭 정보에 제공된 중요도 264

어드바이저 서버 재시도 268

어드바이저 서버 제한시간 267

어드바이저 휴면 시간 267

휴면 시간 266

사용자 정의(사용자 정의 가능) 어드바이저

작성 268

Cisco CSS Controller

loglevel 459

logsize 459

Nortel Alteon Controller

loglevel 479, 480

logsize 479

제품 컴포넌트 57

주소 맵핑 파일

예제 253

주의사항 527

[차]

최대 가중치, 설정

특정 포트상의 서버 195, 409

추가

클러스터 382

클러스터에 포트 77, 408

포트에 서버 78, 423, 450

추가 라우트 83, 84

[카]

컨설턴트

시작 168, 189

nalcontrol 478

컨텐츠 규칙 61, 234

클러스터

비율 설정 79

와일드 카드 76

정의 75, 382

제거 382, 453

주소 구성 76

추가 382

표시

이 클러스터의 상태 382

클러스터 소유의

proportions 452

키

lbkeys 212, 272, 282

[타]

통계 스냅샷 보고서, 표시 400, 440, 441

[파]

파일

cbrcontrol 125

dscontrol 70

sscontrol 146

포트

어드바이저용 372, 430

와일드 카드 77

제거 409

최대 가중치 설정 195, 409

추가 408

클러스터에 대해 정의 77, 408

표시

이 포트상의 서버 상태 409

포트 연관 관계 무시

서버 234, 419, 423

포트간 연관 관계 237, 404

표시

글로벌 값과 해당되는 기본 설정값

관리자용 402, 441, 442

어드바이저용 376, 432, 434

내부 카운터 387

목록

현재 측정 기준을 제공하는 어드바이저

376, 433

버전 번호

관리자의 402, 441, 443

어드바이저의 377, 433, 434

표시 (계속)

상태

포트상의 서버 409

하나의 클러스터 또는 모든 클러스터

382

어드바이저의 상태에 대한 보고서 376,

431, 433

통계 보고서 400, 440, 441

표준화된 가중치에 대해 모든 서버 재시작

401, 440, 442

[하]

하드웨어 요구사항

Cisco CSS Controller 157

Nortel Alteon Controller 175

해상도, GUI 327

확인

여분의 라우트 83

활동해제 제한시간 288, 381, 385, 406

활성 쿠키 연관 관계 239, 240, 414

A

add

Cisco CSS Controller 458

Nortel Alteon Controller 478

AIX

설치 34

요구사항 34

B

binlog

2진 로그, 서버 통계 378

cbrcontrol 378

dscontrol 378

C

Caching Proxy 119

CBR용으로 구성 128

Caching Proxy 어드바이저 204

CBR

계획 117

구문 또는 구성 오류 349

구성

타스크 개요 123

CBR (계속)

구성 (계속)

CBR 시스템 설정 128

로드 밸런스 설정 194

어드바이저 서버 재시도 203

문제점 해결 테이블 313

빠른 시작 예제 109

사용할 기능 결정 26

손상된 Latin-1 자국 문자가 표시됨

(Windows) 350

시작 및 정지 298

실행되지 않음 348

어드바이저와 도달 목표에서 모든 다운된

서버를 표시함(Windows) 350

요청이 로드 밸런스되지 않음 348

호스트 이름에 대한 IP 주소를 확인하는 중

에 문제점 발생(Windows) 350

호스트로부터 연결이 끊어짐, 웹 관리 사용

349

Caching Proxy와 함께

개요 118

구성 133

mapport 키워드 120

SSL 연결 120

ssl2http 어드바이저 120

cbrcontrol 실패 348

Dispatcher 컴포넌트 사용 61

ifconfig 명령 130

Java 메모리/ 스레드 오류(HP-UX) 350

lbadmin 실패 348

Matrox AGP 카드에서 예상하지 못한

GUI 작동 349

NIC 별명 지정 130

Solaris에서 cbrcontrol 실패 349

cbr 전달 메소드 61, 63

stickytime 62

cbrcontrol 명령

관리자 397

규칙 411

서버 418

advisor 372

binlog 378

cluster 379

executor 383

file 388

help 390

host 395

logstatus 396

cbrcontrol 명령 (계속)

metric 403

port 404

set 425

status 426

cbrserver

시작 110

ccocontrol 명령

명령 프롬프트 457

컨설턴트 458

consultant 461

file 463

help 465

host 472

metric 470

server 475

ccoserver

시작 154

시작되지 않음 321, 322, 354

Cisco CSS Controller

결합 배치 261

경보 276

계획 157

고가용성 261

구성

예제 17

타스크 개요 163

CSS 시스템 설정 166

로드 밸런스 설정 264

명령 457

문제점 해결 테이블 315

빠른 시작 예제 153

사용 299

사용할 기능 결정 30

서버 통계의 2진 로그 275

스위치에 의해 가중치가 갱신되지 않음

355

시작 299

시작 및 정지 299

시작되지 않음 354

어드바이저 266

작업로드 관리자 어드바이저 274

컨설턴트 연결 오류 355

포트 13099에서 레지스트리를 작성할 수

없음 355

하드웨어 및 소프트웨어 요구사항 157

호스트로부터 연결이 끊어짐, 웹 관리 사용

356

Cisco CSS Controller (계속)

- ccocontrol 실패 354
- Java 메모리/ 스레드 오류(HP-UX) 356
- lbadm인 실패 354
- Matrox AGP 카드에서 예상하지 못한 GUI 작동 355
- Metric Server 272
- refresh 명령이 구성을 갱신하지 않음 356
- report
 - controller 461

Cisco CSS Controller 컴포넌트

- 손상된 Latin-1 자국 문자가 표시됨 (Windows) 356

cluster

- cbrcontrol 379
- dscontrol 379
- proportions 379

connecttimeout

- Site Selector 430

consultant

- cococontrol 458, 461
- Cisco CSS Controller
 - 2진 로그 458
 - add 458
 - report 458
- nalcontrol 482
- Nortel Alteon Controller
 - 2진 로그 478
 - add 478
 - report 478

Content Based Routing 5

- 계획 117
- 구성
 - 타스크 개요 123
- CBR 시스템 설정 128
- 로드 밸런스 설정 194
- 문제점 해결 테이블 313
- 사용 298
- Dispatcher 컴포넌트 사용 61

controller

- Cisco CSS Controller
 - loglevel 461
 - logsize 461
 - report 461
 - set 461
- Nortel Alteon Controller
 - loglevel 482
 - logsize 482

controller (계속)

- Nortel Alteon Controller (계속)
 - report 482
 - set 482

D

DB2 어드바이저 204

default.cfg 75, 130, 149

Dispatcher

- 구성
 - 백엔드 서버 설정 79
- 사용할 기능 결정 21

Dispatcher 컴포넌트

- 갱신 후 lbadm인 서버로부터 연결이 끊어짐 330
- 계획 57
- 고가용성 구성에서 기본 시스템 및 백업 시스템 활성화 339
- 고가용성 사용 시 IP 주소 충돌 339
- 고가용성이 작동하지 않음 324
- 고가용성, 구성 팁 340
- 구성
 - 개인용 네트워크 설정 252
 - 타스크 개요 69
 - Load Balancer 시스템 설정 72
- 네트워크 정지 후 고가용성 설정에서 어드바이저가 작동하지 않음(Windows) 336
- 다운된 서버 재설정 196, 408
- 대형 페이지 응답 리턴 시 클라이언트 요청 실패 339
- 도움말 창을 열 수 없음 326
- 도움말 창이 사라짐 327
- 로드 밸런스 설정 194
 - 가중치 195
 - 감도 임계치 197
 - 관리자 간격 197
 - 상태 정보에 제공되는 중요성의 비율 194
- 어드바이저 간격 201
- 어드바이저 보고서 제한시간 202
- 어드바이저 서버 재시도 196, 203
- 어드바이저 서버 제한시간 202
- smoothing index 198
- 로컬 주소 대신 별명 리턴 331
- 루프백 별명을 지정하기 위해 IP address add 명령을 사용하지 마십시오 (Linux) 337

Dispatcher 컴포넌트 (계속)

- 문제점 해결 테이블 308
- 사용 288
- 서버 로드를 등록하지 않음 332
- 서버가 응답하지 않음 323
- 서비스 수정사항 설치 시 Java 경고 메시지가 나타남 347
- 설치가 공급된 Java 업그레이드 347
- 손상된 Latin-1 자국 문자가 표시됨 (Windows) 334
- 시작 288
- 실행 프로그램을 시작할 때 파란색 화면이 표시됨 328
- 실행되지 않음 323
- 어드바이저가 작동하지 않음 324
- 어드바이저와 도달 목표에서 모든 다운된 서버를 표시함(Windows) 335
- 요청이 밸런스되지 않음 323
- 원격 시스템에 연결 325
- 원격 연결을 통해 분석되지 않는 IP 주소 330
- 응답 시간이 느림 332
- 추가 라우트(Windows) 324
- 큰 구성 파일을 로드할 때의 예기치 못한 작동 329
- 포트 메소드에 대해 지정되지 않은 또는 올바르게 바르지 않은 라우터 주소 337
- 프레임을 전달할 수 없음 327
- 하트 비트를 추가할 수 없음 324
- 호스트 이름에 대한 IP 주소를 확인하는 중에 문제점 발생(Windows) 335
- 호스트로부터 연결이 끊어짐, 웹 관리 사용 333
- 0.0.0.0에 웹 서버 바인딩 333
- AIX 및 Linux에서 원치 않는 한글 글꼴 330
- Caching Proxy 설치 시 오류 327
- content-based routing 61
- Discovery 경로로 인해 []와의 리턴 통신이 발생하지 못함 328
- dscontrol 실패 325
- GUI가 올바르게 시작되지 않음 326
- GUI가 올바르게 표시되지 않음 327
- IPv6 지원 89
- Java 메모리/ 스레드 오류(HP-UX) 334
- lbadm인 실패 325
- Linux에서 관리자 및 어드바이저 사용 시 메모리 누수가 발생 344

Dispatcher 컴포넌트 (계속)

Linux에서 HA Dispatcher 동기화가 실패
할 수 있음 340

Linux에서는 Dispatcher가 패킷을 전달하지
만 백엔드 서버는 이를 수신하지 못함
345

Linux의 zSeries 또는 S/390 서버 사용 시
제한사항 342

Load Balancer 구성 로드 시 지연 338

Load Balancer 프로세스 종료
(Solaris) 338

Load Balancer의 광역 모드에서 고가용성
이 작동되지 않음 329

MAC 전달 59

Matrox AGP 카드에서 예상하지 못한
GUI 작동 331

MS IIS와 SSL이 작동하지 않음 325

NAT/ NAPT 59

Solaris에서 IPv6 서버를 구성에 추가할 수
없음 347

Windows 시스템의 고가용성 인계 문제
346

Windows에서 "서버가 응답하지 않음" 오
류 발생 340

"rmmod ibmlb"에서의 예상치 않은 작동
332

DPID2 291

dscontrol 명령

고가용성 391, 487

관리자 78, 397

규칙 411

명령 매개변수 최소화 370

명령 프롬프트 370

서버 418

실행 프로그램 75

advisor 78, 372

binlog 378

cluster 379

executor 383

file 388

help 390

host 395

logstatus 396

metric 403

port 77, 404

server 78

set 425

status 426

dscontrol 명령 (계속)

subagent 427

dsserver

시작 51

E

executor

cbrcontrol 383

dscontrol 383

F

file

cbrcontrol 388

ccocontrol 463

dscontrol 388

nalcontrol 484

sscontrol 435

ftp 어드바이저 372, 430

G

goActive 224

goIdle 224

goInOp 224

goStandby 224

GRE(일반 경로 지정 캡슐화)

광역 지원 250

Linux 251

OS/390 250

GUI

일반 명령 499

해상도 327

CBR 126

Cisco CSS Controller 165

Dispatcher 71

Nortel Alteon Controller 187

Site Selector 147

GUI(Graphical User Interface)

일반 명령 499

CBR 126

Cisco CSS Controller 165

Dispatcher 71

Nortel Alteon Controller 187

Site Selector 147

H

help

cbrcontrol 390

ccocontrol 465

dscontrol 390

nalcontrol 486

highavailChange 225

host

cbrcontrol 395

ccocontrol 472

dscontrol 395

nalcontrol 495

HP-UX

설치 38

요구사항 38

arp publish 명령 77

http 어드바이저 372, 430

I

IBM 방화벽(제한) 44

ibmlb.conf

Solaris용으로 구성 73

ibmproxy 120, 128

ifconfig 명령 77, 80, 130, 247

IPv4 및 IPv6용 Load Balancer 89

결합 배치 94

고가용성 93

구성 고려사항 92

루프백 장치의 별명 지정 97

링크 로컬 주소 92

명령 구문 차이점 101

어드바이저, 사용 93

지원되지 않는 기능 92

플랫폼 지원 90

autoconf6, AIX 96

dsconfig 97

dscontrol 명령 101

ifconfig 97

ip addr 97

IPv6 패킷 사용 가능 96

Metric Server 95

modprobe, Linux 96

IPv6 지원 89

결합 배치 94

고가용성 93

구성 고려사항 92

IPv6 지원 (계속)

- 링크 로컬 주소 92
- 명령 구문 차이점 101
- 어드바이저, 사용 93
- 지원되지 않는 기능 92
- 플랫폼 지원 90
- autoconf6, AIX 96
- dsconfig 97
- dscontrol 명령 101
- ifconfig 97
- ip addr 97
- IPv6 패킷 사용 가능 96
- Metric Server 95
- modprobe, Linux 96
- NIC 별명 지정 97

K

kernel free 지원 90

L

lbkeys 212, 273, 282

lbwebaccess 284

Linux

- 설치 40
- 요구사항 40
- S/390의 고가용성 225

Load Balancer

- 개요 3, 9
- 계획 고려사항 57, 141
- 구성

- CBR 123
- Cisco CSS Controller 163
- Dispatcher 컴포넌트 72, 128, 148
- Nortel Alteon Controller 185
- Site Selector 145

구성 태스크, 고급 193, 215

기능 3, 9

문제점 해결 303

빠른 시작 예제 49

- CBR 109
- Cisco CSS Controller 153
- Nortel Alteon Controller 171
- Site Selector 137

설치 33

이점 5

작동 및 관리 281, 299, 300

Load Balancer (계속)

IPv6 지원 89

Load Balancer 관리 281

Load Balancer 작동 281

log

레벨, 설정

관리자용 285, 439

서버 285, 287

서브에이전트 285

어드바이저용 285, 376, 433

컨설턴트용 287

서버 정보는 2진 파일에 257

크기, 설정

관리자용 286, 400, 439, 441

서버 286, 287

서브에이전트 286, 287

어드바이저용 286, 376, 431, 433

컨설턴트용 287

파일, 이름 설정

관리자용 441

어드바이저용 432

CBR 로그 사용 299

Cisco CSS Controller 로그 사용 300

Load Balancer 로그 사용 285

Metric Server 로그 사용 301

Site Selector 로그 사용 299

logstatus

cbrcontrol 396

dscontrol 396

sscontrol 438

M

mac 전달 메소드 59

metric

cbrcontrol 403

ccocontrol 470

dscontrol 403

nalcontrol 491

sscontrol 444

Metric Server

개요 211, 272

문제점 해결 테이블 317

사용 301

시작 및 정지 301

2계층 구성에서 Metric Server를 구성함 361

AIX에서 ps -vg 명령 출력이 손상됨 360

Metric Server (계속)

IPv6 고려사항 95

Linux 시스템에서 metric server로부터 값을 검색할 수 없음 362

Metric Server 로그에서 "에이전트 액세스 하려면 서명이 필요합니다."라고 보고합니다. 360

Metric Server 시작 후 메트릭 값 -1 리턴 363

Metric Server로드를 보고하지 않음 360

Solaris에서 스크립트가 원치 않는 콘솔 메시지를 생산 362

Windows의 Metric Server

IOException 359

N

nalcontrol 명령

명령 프롬프트 477

컨설턴트 478

consultant 482

file 484

help 486

host 495

metric 491

server 493

nalserver

시작 173

시작되지 않음 356

nameserver

sscontrol 445

NAPT(네트워크 주소 포트 변환) 59

NAT 전달 메소드 59

고가용성 스크립트 224

nat 전달 메소드 63

NAT(네트워크 주소 변환) 59

nat를 사용하여 결합 배치 217

nat, 서버 결합 배치 217

ndcontrol 명령

고가용성 466

netstat 명령 83

NIC

매퍼핑(Windows의 경우) 76

별명 76

이더넷(Solaris-용) 73

Nortel Alteon Consultant

사용할 기능 결정 31

Nortel Alteon Controller

- 결합 배치 261
- 경보 276
- 계획 175
- 고가용성 261
- 구성
 - 타스크 개요 185
- Nortel Alteon Controller 시스템 설정 188
- 로드 밸런스 설정 264
- 명령 477
- 문제점 해결 테이블 316
- 빠른 시작 예제 171
- 사용 300
- 서버 통계의 2진 로그 275
- 손상된 Latin-1 자국 문자가 표시됨 (Windows) 359
- 스위치에 의해 가중치가 갱신되지 않음 358
- 시작 및 정지 300
- 시작되지 않음 356
- 어드바이저 266
- 작업로드 관리자 어드바이저 274
- 컨설턴트 연결 오류 358
- 포트 14099에서 레지스트리를 작성할 수 없음 357
- 하드웨어 및 소프트웨어 요구사항 175
- 호스트로부터 연결이 끊어짐, 웹 관리 사용 358
- Java 메모리/ 스레드 오류(HP-UX) 359
- lbadmin 실패 357
- Matrox AGP 카드에서 예상하지 못한 GUI 작동 358
- Metric Server 272
- nalcontrol 실패 357
- refresh 명령이 구성을 갱신하지 않음 359
- report
 - controller 482

O

OS/390

- GRE 지원 250

P

port

- cbrcontrol 404

port (계속)

- dscontrol 404
- primaryhost 382

R

report

- Cisco CSS Controller 461
- Nortel Alteon Controller 482
- RMI(원격 메소드 호출) 37, 42, 43, 44, 281, 282
- route 명령 83, 84

S

server

- 가중치 설정 423, 449
 - 논리 64
 - 단절 표시 423, 449, 450
 - 물리적 64
 - 연결 표시 423, 450
 - 작업중지 400
 - 작업중지 해제 402
 - 제거 423, 449, 450
 - 추가 423, 450
 - 파티션 64
 - 포트에 대해 정의 78, 423, 450
 - 표준화된 가중치에 대해 모두 재시작 401, 440, 442
 - ccocontrol 475
 - mapport 120
 - nalcontrol 493
 - sscontrol 449
- ### set
- cbrcontrol 425
 - dscontrol 425
 - sscontrol 451

Site Selector

- 개요 15
- 계획 141
- 구성
 - 시스템 설정 148
 - 타스크 개요 145
- 구성 예제 16
- 로드 밸런스 설정 194
- 어드바이저 서버 재시도 203
- 어드바이저 서버 제한시간 202
- 명령 429

Site Selector (계속)

- 문제점 해결 테이블 314
- 빠른 시작 예제 137
- 사용 299
- 사용할 기능 결정 28
- 손상된 Latin-1 자국 문자가 표시됨 (Windows) 353
- 시작 및 정지 299
- 실행되지 않음 351
- 어드바이저와 도달 목표에서 모든 다운된 서버를 표시함(Windows) 353
- 중복 라우트를 통해 올바르게 로드 밸런스 하지 않음 352
- 호스트로부터 연결이 끊어짐, 웹 관리 사용 353
- HA Dispatcher 로드 밸런스 225
- Java 메모리/ 스레드 오류(HP-UX) 353
- lbadmin 실패 351
- Matrox AGP 카드에서 예상하지 못한 GUI 작동 352
- Solaris 클라이언트로부터 통신을 라운드 로 방하지 않음 351
- sscontrol 실패 351
- ssserver가 Windows에서 시작에 실패함 352
- sitename
 - sscontrol 452
- SNMP 286, 290
- SNMP(Simple Network Management Protocol) 290
- Solaris
 - 설치 42
 - 요구사항 42
 - arp publish 명령 77
 - Dispatcher 시스템 설정 73
- sscontrol 명령
 - 관리자 439
 - 규칙 446
 - advisor 430
 - file 435
 - help 437
 - logstatus 438
 - metric 444
 - nameserver 445
 - server 449
 - set 451
 - sitename 452
 - status 455

SSL 77

SSL 연결

사용 가능에 대한 문제점 325

CBR 120

HTTPS 어드바이저 203

ibmproxy 구성 120

SSL 어드바이저 204

ssl2http 어드바이저 120, 204

ssserver

시작 138

status

cbrcontrol 426

dscontrol 426

system metrics

구성 403, 444, 470, 491

중요성 비율 설정 195, 379, 380

U

URI 연관 관계 240, 243, 414

W

WAS 어드바이저 205, 208

WAS(WebSphere Application Server)

WAS 어드바이저 205, 208

Windows

설치 44

실행 프로그램 구성 명령 76

요구사항 44

Dispatcher 시스템 설정 74



GA30-2917-00



Spine information:



**WebSphere Application
Server**

Load Balancer 관리 안내서

버전 6.1

GA30-2917-00