

멀티플랫폼용 WebSphere Edge Server



Network Dispatcher 관리 안내서

버전 2.0

멀티플랫폼용 WebSphere Edge Server



Network Dispatcher 관리 안내서

버전 2.0

주!

이 책과 이 책에서 지원하는 제품을 사용하기 전에, 435 페이지의 『부록1. 주의사항』을 읽으십시오.

목차

표	xi	설치 단계	28
그림	xiii	제3장 Network Dispatcher 소개	31
시작	xv	Network Dispatcher 개념	31
독자 의견을 보내는 방법	xv	Network Dispatcher가 필요한 이유	32
제1장 시작하기	1	새로운 기능	34
필요한 내용	2	Network Dispatcher 구성요소 종류	40
준비 방법	2	Dispatcher 구성요소 개요	40
Dispatcher 구성요소 구성	4	CBR(Content Based Routing) 구성요소	
명령행을 사용한 구성	4	개요	44
구성 마법사를 사용한 구성	5	Mailbox Locator 구성요소 개요	46
GUI(Graphical User Interface)를 사용한		Site Selector 구성요소 개요	48
구성	6	Consultant for Cisco CSS Switches 구성	
구성 검사	8	요소 개요	50
클러스터, 포트, 서버 구성의 유형	8	고가용성 정보	53
제2장 Network Dispatcher 설치	13	Dispatcher	53
AIX의 요구사항	15	CBR, Mailbox Locator, Site Selector	53
AIX용 설치	16	제4장 Dispatcher 구성요소 계획	55
설치하기 전에	17	하드웨어 및 소프트웨어 요구사항	55
설치 단계	17	계획 고려사항	56
Red Hat Linux 또는 SuSE Linux 요구사항	20	고가용성	58
Linux에 설치	21	단순 고가용성	58
설치하기 전에	21	상호 고가용성	59
설치 단계	21	Dispatcher의 MAC 레벨 경로 지정(MAC 전	
Solaris 요구사항	23	달 방법)	60
Solaris에 설치	24	Dispatcher의 NAT/NAPT(nat 전달 방법)	61
설치하기 전에	24	Dispatcher content-based routing(cbr 전달	
설치 단계	24	방법)	63
Windows 2000 요구사항	26	제5장 Dispatcher 구성요소 구성	67
Windows 2000에 설치	27	구성 태스크 개요	67
설치 패키지	27	구성 방법	67
설치하기 전에	27	명령행	68
		스크립트	68

GUI	69	구성 마법사.	100
구성 마법사	70	CBR 시스템 설정	101
Dispatcher 시스템 설정	70	1단계. CBR을 사용하기 위해 Caching	
1단계. 서버 기능 시작	73	Proxy 구성.	101
2단계. 실행 프로그램 기능 시작	73	2단계. 서버 기능 시작	104
3단계. 비전달 주소 정의(호스트 이름과 다		3단계. 실행 프로그램 기능 시작.	104
른 경우)	73	4단계. 클러스터 정의 및 클러스터 옵션	
4단계. 클러스터 정의 및 클러스터 옵션 설		설정	104
정	73	5단계. 네트워크 인터페이스 카드 별명 지	
5단계. 네트워크 인터페이스 카드의 별명 지		정(선택적)	105
정	74	6단계. 포트 정의 및 포트 옵션 설정	106
6단계. 포트 정의 및 포트 옵션 설정.	76	7단계. 로드 밸런스 서버 시스템 정의	106
7단계. 로드 밸런스 서버 시스템 정의	76	8단계. 구성에 규칙 추가	106
8단계. 관리 프로그램 기능 시작(선택)	77	9단계. 규칙에 서버 추가	107
9단계. 권고자 기능 시작(선택)	77	10단계. 관리 프로그램 기능 시작(선택적)	107
10단계. 필요한 클러스터 비율 설정	78	11단계. 권고자 기능 시작(선택적)	107
로드 밸런스에 대한 서버 시스템 설정	78	12단계. 필요한 클러스터 비율 설정.	107
1단계. 루프백 장치에 별명 지정	78	13단계. Caching Proxy 시작.	107
2단계. 여분의 라우트 확인	81	CBR 구성 예제	108
3단계. 여분의 라우트 삭제	82	제8장 Mailbox Locator 구성요소에 대한	
4단계. 서버의 올바른 구성 여부 확인	82	계획	109
Linux 커널 패치 설치(루프백 인터페이스에		하드웨어 및 소프트웨어 요구사항	109
서 arp 응답 억제)	83	계획 고려사항	110
제6장 Content Based Routing 구성요소 계		연관 관계 기능 사용.	112
획	89	POP3/IMAP 활동 해제 타이머 덮어쓰기	112
하드웨어 및 소프트웨어 요구사항	89	제9장 Mailbox Locator 구성요소 구성	113
계획 고려사항	89	구성 태스크 개요	113
완전 보안(SSL) 연결의 로드 밸런스	92	구성 방법	114
SSL의 클라이언트-투-프록시 및 HTTP의		명령행	114
프록시-투-서버의 로드 밸런스	92	스크립트.	115
제7장 Content Based Routing 구성요소 구		GUI	115
성	95	구성 마법사.	117
구성 태스크 개요	95	Mailbox Locator 시스템 설정	117
구성 방법.	95	1단계. 서버 기능 시작	118
명령행	96	2단계. 클러스터 정의 및 클러스터 옵션	
스크립트	98	설정	118
GUI	98	3단계. 포트 정의 및 포트 옵션 설정	118

4단계. 로드 밸런스 서버 시스템 정의	119	명령행	142
5단계. 관리 프로그램 기능 시작(선택)	119	스크립트.	143
6단계. 권고자 기능 시작(선택)	119	GUI	143
7단계. 필요한 클러스터 비율 설정	119	Consultant for Cisco CSS Switches 시스템 설정	145
제10장 Site Selector 구성요소에 대한 계획	121	1단계. 서버 기능 시작	145
하드웨어 및 소프트웨어 요구사항	121	2단계. 실행 프로그램 기능 구성	145
계획 고려사항	122	3단계. 클러스터 정의 및 클러스터 옵션 설정	145
TTL 고려사항	124	4단계. 포트 정의 및 포트 옵션 설정	145
네트워크 근접 기능 사용	125	5단계. 로드 밸런스 서버 시스템 정의	146
제11장 Site Selector 구성요소 구성	127	6단계. 관리 프로그램 기능 시작.	146
구성 태스크 개요.	127	7단계. 권고자 기능 시작(선택)	146
구성 방법	127	8단계. 필요한 클러스터 비율 설정	147
명령행	128	9단계. Metric Server 시작(선택)	147
스크립트.	129	구성 검사	147
GUI	129	제14장 고급 Network Dispatcher 기능	149
구성 마법사.	130	Network Dispatcher가 제공하는 로드 밸런 스 최적화	152
Site Selector 시스템 설정.	131	상태 정보에 제공되는 중요성 비율	152
1단계. 서버 기능 시작	131	가중치	154
2단계. 이름 서버 시작	132	관리 프로그램 간격	155
3단계. 사이트 이름 정의 및 사이트 이름 옵션 설정	132	감도 임계치.	156
4단계. 로드 밸런스 서버 시스템 정의	132	스무스 색인.	156
5단계. 관리 프로그램 기능 시작(선택적)	133	스크립트를 사용하여 정보나 레코드 서버 장애 생성	157
6단계. 권고자 기능 시작(선택적).	133	권고자	157
7단계. 시스템 메트릭 정의(선택적)	133	권고자 작동 방법.	158
8단계. 필요한 사이트 이름 비율 설정	133	권고자 시작 및 정지.	158
서버 시스템의 시스템 설정.	133	권고자 간격.	159
제12장 Consultant for Cisco CSS Switches 구성요소에 대한 계획	135	권고자 보고서 시간 종료	160
하드웨어 및 소프트웨어 요구사항	135	권고자 연결 시간 종료 및 서버의 수신 시 간 종료	160
계획 고려사항	136	권고자 목록.	161
제13장 Consultant for Cisco CSS Switches 구성요소 구성	141	조정(사용자 정의 기능) 권고자 작성	163
구성 태스크 개요.	141	WAS(WebSphere Application Server) 권고자	165
구성 방법	142		

이름 지정 규칙	165	클라이언트 IP 주소에 따라 규칙 사용	198
컴파일	165	시간에 따라 규칙 사용	199
실행	166	포트의 초당 연결 수에 따라 규칙 사용	199
필수 루틴	166	포트의 총 작동 중인 연결 수에 따라 규칙 사용	199
탐색 순서	167	클라이언트 포트에 따라 규칙 사용	200
이름 지정 및 경로	167	서비스 유형(TOS)에 기반하여 규칙 사용	200
예제 권고자.	168	예약된 대역폭 및 공유 대역폭에 따라 규칙 사용	201
작업로드 관리 프로그램 권고자	168	Metric all 규칙	203
Metric Server 제한사항	169	Metric average 규칙	204
Metric Server.	169	항상 참인 규칙 사용.	204
WLM 제한사항	170	요청 내용에 따라 규칙 사용	205
전제조건.	170	사용자 구성에 규칙 추가	206
Metric Server 사용 방법	170	규칙에 대한 서버 평가 옵션	206
서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소).	172	명시적 링크 사용.	207
HTTP 권고자 요청/응답(URL) 옵션	174	사설 네트워크 구성 사용	208
결합 배치된 서버 사용	175	와일드 카드 클러스터를 사용하여 서버 구성 조합	209
Dispatcher 구성요소의 경우	175	와일드 카드 클러스터를 사용하여 방화벽 로드 밸런스 수행	210
CBR 구성요소.	176	투명 프록시의 경우 Caching Proxy가 있는 와일드 카드 클러스터 사용.	211
Mailbox Locator 구성요소의 경우	176	와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정	212
Site Selector 구성요소의 경우	177	Network Dispatcher에 대한 친근성 기능 사용법	212
Cisco Consultant 구성요소의 경우	177	친근성이 사용 가능하지 않을 때의 행동	212
광역 Dispatcher 지원 구성	177	친근성이 사용 가능할 때의 행동.	212
명령 구문	178	클라이언트 서버 유사성을 제어하는 서버 지정 유사 API	213
광역 지원으로 원격 권고자 사용.	179	포트간 친근성	214
구성 예제	182	친근성 주소 마스크	215
주.	184	규칙 친근성 무시.	216
GRE(일반 경로 지정 캡슐화) 지원	185	결합 연결 처리 작업중지	216
상하단부 WAND 구성에서 자가 권고자 사용.	186	규칙에 대한 연관 관계 옵션	217
고가용성.	187	활성 쿠키 연관 관계.	218
고가용성 구성	188	수동 쿠키 연관 관계.	219
핵심 메시지 및 도달 목표를 사용하는 고장 검색 기능	191		
복구 전략	192		
스크립트 사용	193		
규칙 기반 로드 밸런스 구성	196		
규칙 평가 방법	197		

URI 연관 관계	220	Metric Server 시작 및 정지	245
서비스 거부 중지 감지	222	Metric Server 로그 사용	245
서버 통제를 분석하기 위해 2진 로그 사용	223	제16장 문제점 해결	247
고급 Cisco Consultant 기능에 대한 추가 정		문제점 해결 테이블	247
보	225	Dispatcher 포트 번호 확인	252
Cisco Consultant가중치	227	CBR 포트 번호 확인	253
제15장 Network Dispatcher 작동 및 관리	229	Mailbox Locator 포트 번호 확인	254
원격 인증 관리	229	Site Selector 포트 번호 확인	255
Network Dispatcher 로그 사용	231	Cisco Consultant 포트 번호 확인	255
로그 파일 경로 변경	232	공통 문제점 해결--Dispatcher	256
Dispatcher 구성요소 사용	233	문제점: Dispatcher가 실행되지 않음	256
Dispatcher 시작 및 정지	233	문제점: Dispatcher 및 서버가 응답하지	
활동해제 제한 시간 값 사용	233	않음	256
FIN 계수를 사용하여 가비지 컬렉션 제어	234	문제점: Dispatcher 요청이 백런스를 이루	
GUI 보고 -- 모니터 메뉴 옵션	235	지 않음	256
Dispatcher 구성요소에 Simple Network		문제점: Dispatcher 고가용성 기능이 작동	
Management Protocol 사용	235	하지 않음	257
ipchain 또는 iptable을 사용하여		문제점: 핵심 메시지를 추가할 수 없음	
Network Dispatcher (Linux에서) box를		(Windows 2000).	257
굳히는 모든 통신량을 거절함	241	문제점: 추가 라우트(Windows 2000)	258
Content Based Routing 구성요소 사용	242	문제점: 권고자가 제대로 작동하지 않음	258
CBR 시작 및 정지	242	문제점: SNMPD가 올바르게 실행되지 않	
CBR 제어	242	음(Windows 2000)	258
CBR 로그 사용	243	문제점: Dispatcher, Microsoft IIS 및	
Mailbox Locator 구성요소 사용	243	SSL이 작동하지 않음(Windows 2000)	258
Mailbox Locator 시작 및 정지	243	문제점: 원격 시스템에 대한 Dispatcher	
Mailbox Locator 제어	243	연결	258
Mailbox Locator 로그 사용	243	문제점: ndcontrol 또는 ndadmin 명령 실행	
Site Selector 구성요소 사용	244	패	259
Site Selector 시작 및 정지	244	온라인 도움말을 보려고 할 때 문제점: “	
Site Selector 제어	244	파일을 찾을 수 없습니다...”라는 오류 메	
Site Selector 로그 사용	244	시지 발생(Windows 2000)	260
Cisco Consultant 구성요소 사용	244	문제점: Solaris 2.7에서 ndserver를 시작	
Cisco Consultant 시작 및 정지	244	할 때 유사 오류 메시지 발생	260
Cisco Consultant 제어	244	문제점: GUI(Graphical User Interface)가	
Cisco Consultant 로그 사용	245	올바로 시작되지 않음	260
Metric Server 구성요소 사용	245		

문제점: Caching Proxy가 설치된 Dispatcher 실행 중 오류	260	문제점: Site Selector가 실행되지 않음	268
문제점: GUI(Graphical User Interface)가 올바르게 표시되지 않음.	261	문제점: Site Selector가 Solaris 클라이언트로부터 통신을 라운드 로빈 (round-robin)하지 않음.	269
문제점: Windows 2000의 경우, 도움말 창이 다른 열린 창 뒤로 사라짐	261	문제점: sscontrol 또는 ndadmin 명령 실패	269
문제점: Network Dispatcher가 프레임을 처리하고 전달할 수 없음	261	문제점: ssserver가 Windows 2000에서 시작에 실패함	269
문제점: Network Dispatcher 실행 프로그램 램을 시작할 때 파란색 화면이 표시됨	262	문제점: Site Selector가 중복 라우트를 통해 올바르게 로드 밸런스하지 않음	270
문제점: Discovery 경로로 인해 Network Dispatcher와의 리턴 통신이 발생하지 못함	262	공통 문제점 해결--Consultant for Cisco CSS Switches.	270
문제점: 권고지는 모든 서버가 중단되었음을 표시함	263	문제점: lbcserver가 시작되지 않음	270
문제점: Network Dispatcher의 광역 모드에서 고가용성이 작동되지 않음	263	문제점: lbccontrol 또는 ndadmin 명령 실패	270
문제점: GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.	264	문제점: 포트 14099에서 레지스트리를 작성할 수 없음	270
공통 문제점 해결--CBR.	265	공통 문제점 해결--Metric Server	271
문제점: CBR이 실행되지 않음	265	문제점: .bat 또는 .cmd 사용자 메트릭 파일을 실행하는 Windows 2000의 Metric Server IOException.	271
문제점: cbrcontrol 또는 ndadmin 명령 실패	265	문제점: Metric Server가 Network Dispatcher 시스템에 로드를 보고하지 않음.	271
문제점: 요청이 로드 밸런스되지 않음	266	문제점: Metric Server 로그가 "에이전트에 액세스하려면 서명이 필요합니다"라고 보고합니다.	272
문제점: Solaris에서 cbrcontrol executor start 명령 실패	266	부록A. 구문 다이어그램 읽는 방법	273
문제점: 구문 또는 구성 오류	266	기호 및 구두점	273
공통 문제점 해결--Mailbox Locator	267	매개변수.	273
문제점: Mailbox Locator가 실행되지 않음.	267	구문 예제	274
문제점: mlserver 명령이 정지됨	267	부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서.	277
문제점: mlcontrol 또는 ndadmin 명령 실패	267	CBR, Mailbox Locator 및 Dispatcher 간의 구성 차이점.	278
문제점: 포트를 추가할 수 없음	268	ndcontrol advisor -- 권고자 제어	280
문제점: 포트 추가를 시도할 때 프록시 오류 발생	268	ndcontrol cluster -- 클러스터 구성.	287
공통 문제점 해결--Site Selector	268		

ndcontrol executor -- 실행 프로그램 제어	293
ndcontrol file -- 구성 파일 관리	298
ndcontrol help -- 이 명령의 도움말 표시 또는 인쇄	300
ndcontrol highavailability -- 고가용성 제어	302
ndcontrol host -- 원격 시스템 구성	307
ndcontrol log -- 2진 로그 파일 제어	308
ndcontrol manager -- 관리 프로그램 제어	309
ndcontrol metric -- 시스템 메트릭 구성	317
ndcontrol 포트 -- 포트 구성	319
ndcontrol rule -- 규칙 구성	327
ndcontrol server -- 서버 구성	335
ndcontrol set -- 서버 로그 구성	342
ndcontrol status -- 관리 프로그램 및 권고자가 실행 중인지 여부 표시	343
ndcontrol subagent -- SNMP 서브에이전트 구성	344
부록C. 콘텐츠 규칙(패턴) 구문	347
콘텐츠 규칙(패턴) 구문:	347
예약된 키워드	347
부록D. Site Selector 명령어 참조서	351
sscontrol advisor - 권고자 제어	352
sscontrol file - 구성 파일 관리	358
sscontrol help - 이 명령의 도움말 표시 또는 인쇄	360
sscontrol manager - 관리 프로그램 제어	361
sscontrol metric - 시스템 메트릭 구성	366
sscontrol nameserver - NameServer 제어	368
sscontrol rule - 규칙 구성	369
sscontrol server - 서버 구성	373
sscontrol set - 서버 로그 구성	375
sscontrol sitename - 사이트 이름 구성	376
sscontrol status - 관리 프로그램 및 권고자가 실행 여부 표시	380
부록E. Consultant for Cisco CSS Switches 명령어 참조서	381

lbcccontrol advisor -- 권고자 제어	382
lbcccontrol cluster -- 클러스터 구성	388
lbcccontrol executor -- 실행 프로그램 제어	390
lbcccontrol file -- 구성 파일 관리	392
lbcccontrol help -- 이 명령의 도움말 표시 또는 인쇄	394
lbcccontrol host -- 원격 시스템 구성	395
lbcccontrol log -- 2진 로그 파일 제어	396
lbcccontrol manager -- 관리 프로그램 제어	397
lbcccontrol metric -- 시스템 메트릭 구성	404
lbcccontrol port -- 포트 구성	406
lbcccontrol server -- 서버 구성	408
lbcccontrol set -- 서버 로그 구성	411
lbcccontrol status -- 관리 프로그램 및 권고자의 실행 여부 표시	412

부록F. 예제 구성 파일	413
예제 Network Dispatcher 구성 파일	413
Dispatcher 구성 파일--AIX, Red Hat Linux, 및 Solaris	413
Dispatcher 구성 파일--Windows 권고자 예제	421

부록G. Dispatcher, CBR 및 Caching Proxy를 사용하는 2단 고가용성 구성 샘플	427
서버 시스템 설정	427

부록H. 기타 자원	433
명령행 액세스	433
온라인 도움말 받기	433
참조서 정보	433

부록I. 주의사항	435
상표	436
용어집	439
색인	451

표

1. AIX installp 이미지	16	11. 컨설턴트 구성에 맵핑한 Cisco CSS	
2. AIX 설치 명령	18	Switch 구성에 대한 예제	139
3. Dispatcher 기능의 구성 태스크.	67	12. Consultant for Cisco CSS Switches	
4. Dispatcher의 루프백 장치(IoO)에 별명		구성요소의 구성 태스크	141
지정 명령	79	13. Network Dispatcher의 고급 구성 태스	
5. Dispatcher에 대한 여분의 라우트 삭제		크	149
명령	82	14. Dispatcher 문제점 해결 테이블	247
6. CBR 구성요소의 태스크 구성	95	15. CBR 문제점 해결 테이블	250
7. NIC 별명 명령.	105	16. Mailbox Locator 문제점 해결 테이블	250
8. Mailbox Locator 구성요소에 대한 구		17. Site Selector 문제점 해결 테이블	251
성 태스크	113	18. Consultant for Cisco CSS Switches	
9. Site Selector 구성요소 구성 태스크	127	문제점 해결 테이블	251
10. 컨설턴트 및 Cisco CSS Switch 구성		19. Metric Server 문제점 해결 테이블	251
용어	137		

그림

1. 간단한 로컬 Dispatcher 구성 1	15. Dispatcher 시스템에 대해 필요한 IP 주소 예제 72
2. GUI(Graphical User Interface) 6	16. AIX용 CBR 구성 파일 102
3. 단일 클러스터와 두 개의 포트 로 구성된 Dispatcher에 대한 예제 9	17. Linux용 CBR 구성 파일 103
4. 각각 단일 포트인 두 개의 클러스터 로 구성된 Dispatcher에 대한 예제 10	18. Solaris용 CBR 구성 파일 103
5. 각각 포트가 두 개인 두 개의 클러스터 로 구성된 Dispatcher에 대한 예제 11	19. Windows 2000용 CBR 구성 파일 103
6. Dispatcher를 사용하여 로컬 서버를 관리 하는 사이트의 물리적 표현 예제 42	20. DNS 환경의 예제 122
7. Dispatcher 및 Metric Server를 사용하여 서버를 관리하는 사이트 예제 43	21. 각각 포트가 세 개인 두 개의 클러스터 로 구성된 컨설턴트에 대한 예제 138
8. Dispatcher를 사용하여 로컬 및 원격 서버를 관리하는 사이트 예제 44	22. 단일 LAN 세그먼트를 구성하는 구성 예제 177
9. CBR을 사용하여 로컬 서버를 관리하는 예제 46	23. 로컬 및 원격 서버를 사용하는 구성 예제 178
10. Mailbox Locator를 사용하여 로컬 서버 를 관리하는 사이트 예제 47	24. 원격 Network Dispatcher의 광역 구성 예제 182
11. Site Selector 및 Metric Server를 사용하여 로컬 및 원격 서버를 관리하는 사이트 예제 49	25. GRE를 지원하는 서버 플랫폼의 광역 구성 예제 186
12. Cisco Consultant 및 Metric Server를 사용하여 로컬 서버를 관리하는 사이트 예제 52	26. 자가 권고자를 사용하는 상하단부 WAND 구성의 예제 187
13. 고가용성을 사용하는 Dispatcher 예제 58	27. Dispatcher를 사용하는 사설 네트워크 예제 209
14. 상호 고가용성을 사용하는 Dispatcher 예제 59	28. AIX 및 Solaris용 SNMP 명령 237
	29. Windows 2000용 SNMP 명령 238
	30. Dispatcher, CBR 및 Caching Proxy를 사용하는 2단 고가용성 구성 예제 427

시작

이 책에서는 IBM WebSphere Edge Server Network Dispatcher AIX, Linux, Solaris 및 Windows 2000용의 설치, 구성, 사용 및 문제점 해결에 대한 계획 방법을 설명합니다. 이전에는 이 제품을 SecureWay Network Dispatcher, eNetwork Dispatcher 및 Interactive Network Dispatcher라고 했습니다.

이 책의 최신 버전은 WebSphere Edge Server 웹 사이트에서 HTML과 PDF 형식으로 구할 수 있습니다. 온라인 서적을 액세스하려면, 다음 URL로 이동하십시오.

<http://www.ibm.com/software/webservers/edgeserver/library.html>

WebSphere Edge Server 웹 사이트에서는 Network Dispatcher를 사용하여 서버의 성능을 최대화하는 방법에 대한 최신 정보를 제공합니다. 구성 예제 및 시나리오가 나와 있습니다. 이 웹 사이트를 액세스하려면 다음 URL로 이동하십시오.

<http://www.ibm.com/software/webservers/edgeserver>

Network Dispatcher에 관한 최신 갱신 사항과 사용법 힌트를 보려면 WebSphere Edge Server 지원 웹 페이지를 방문하여 *Network Dispatcher Hints and Tips* 를 누르십시오. 이 웹 페이지를 액세스하려면 다음 URL로 이동하십시오.

<http://www.ibm.com/software/webservers/edgeserver/support.html>

독자 의견을 보내는 방법

여러분의 피드백은 정확하고 고품질의 정보를 제공하는 데 중요한 역할을 합니다. 이 책이나 WebSphere Edge Server 문서에 대한 의견이 있는 경우 다음을 참조하십시오.

- 여러분의 의견을 전자 우편으로 fsdoc@us.ibm.com에 보내 주십시오. 의견에는 책 이름, 책의 권 번호, WebSphere Edge Server 버전 및 가능한 경우, 설명하고 있는 텍스트의 특정 위치(예: 페이지 번호 또는 테이블 번호)를 명시하십시오.

제1장 시작하기

Network Dispatcher를 빨리 작동시키는 방법입니다. 다음을 참조하십시오.

사용자가 Intersplash Corporation의 웹 마스터라고 가정합니다. 로컬 웹 사이트는 두 개의 HTTP 서버를 사용하여 관리됩니다. 라운드 로빈(round-robin) 접근 방법을 사용하여 두 서버의 로드를 관리하고 있지만, 최근에 업무를 배정받았으며 고객은 사이트에 액세스할 수 없다고 불평하기 시작하고 있습니다. 어떻게 하시겠습니까?

<http://www.ibm.com/software/webserver/edgeserver>로 이동하여 최신의 Network Dispatcher 버전을 다운로드하십시오. 이 제품에는 Dispatcher, CBR(Content Based Routing), Mailbox Locator, Site Selector 및 Consultant for Cisco CSS Switches(Cisco Consultant) 등의 구성요소가 5개 있습니다. 여기에서는 **Dispatcher** 구성요소에 대해서만 논의합니다.

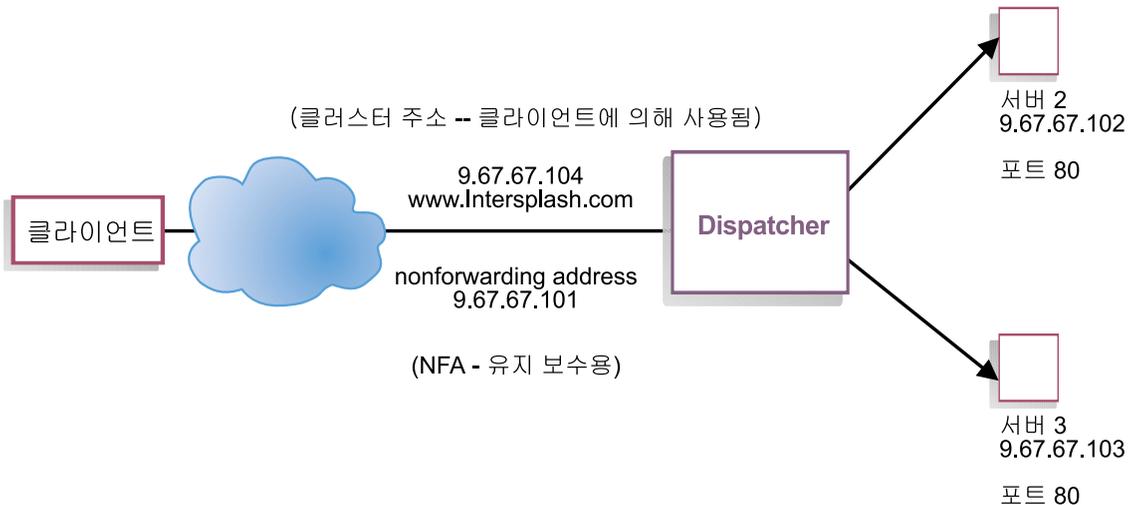


그림 1. 간단한 로컬 Dispatcher 구성

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

주: Dispatcher의 AIX, Linux 또는 Solaris 버전의 경우, 웹 서버 워크스테이션 중 하나에 위치한 Dispatcher가 있는 두 대의 워크스테이션만 사용하여 구성을 완료할 수 있습니다. 이것은 결합 배치 구성을 나타냅니다. 보다 복잡한 구성 설정의 절차는 70 페이지의 『Dispatcher 시스템 설정』에 나와 있습니다.

필요한 내용

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

준비 방법

1. 13 페이지의 『제2장 Network Dispatcher 설치』에 나열되어 있는 조건이 전제되어야 합니다.
2. 동일한 LAN 세그먼트에 있도록 워크스테이션을 설정하십시오. 세 시스템 간의 네트워크 통신량이 라우터 또는 브리지를 통과해서는 안 됩니다.
3. 세 워크스테이션의 네트워크 어댑터를 구성하십시오. 이 예제에서는 다음과 같이 네트워크가 구성되어 있다고 가정합니다.

워크스테이션	이름	IP 주소
1	server1.intersplash.com	9.67.67.101
2	server2.intersplash.com	9.67.67.102
3	server3.intersplash.com	9.67.67.103
Netmask = 255.255.255.0		

각 워크스테이션에는 표준 이더넷 네트워크 인터페이스 카드가 하나만 있습니다.

4. server1.intersplash.com이 server2.intersplash.com과 server3.intersplash.com을 둘 다 ping할 수 있어야 합니다.
5. server2.intersplash.com과 server3.intersplash.com이 server1.intersplash.com을 ping할 수 있어야 합니다.
6. 두 개의 웹 서버(서버 2와 서버 3)에서 내용이 동일한지 확인하십시오. 이는 두 워크스테이션 모두의 데이터를 복제하고, NFS, AFS 또는 DFS와 같은 공유 파일 시스템을 사용하거나 사용자 사이트에 적절한 다른 방법으로 수행될 수 있습니다.
7. server2.intersplash.com과 server3.intersplash.com의 웹 서버가 작동하는지 확인하십시오. 웹 브라우저를 사용하여 **http://server2.intersplash.com**과 **http://server3.intersplash.com**에서 페이지를 직접 요청하십시오.
8. 이 LAN 세그먼트에 유효한 또다른 IP 주소를 확보하십시오. 이 주소는 사용자 사이트에 액세스할 클라이언트에 제공할 주소입니다. 이 예제에서는 다음을 사용합니다.

Name= www.intersplash.com
IP=9.67.67.104

9. www.intersplash.com의 통신을 승인하는 두 개의 웹 서버 워크스테이션을 구성하십시오.

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

- AIX의 경우

ifconfig lo0 alias www.intersplash.com netmask 255.255.255.0

- Solaris 7의 경우

ifconfig lo0:1 www.intersplash.com 127.0.0.1 up

- 기타 운영체제의 경우 79 페이지의 표4를 참조하십시오.

10. 루프백 인터페이스의 별명 지정으로 인해 작성되었을 수 있는 추가 라우트를 삭제하십시오. 81 페이지의 『2단계. 여러분의 라우트 확인』을 참조하십시오. 두 대의 웹 서버 워크스테이션에 필요한 모든 구성 단계를 완료했습니다.

Dispatcher 구성요소 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 ndserver를 시작하십시오.
 - AIX, Linux 또는 Solaris의 경우, **ndserver** 명령을 루트 사용자로서 실행하십시오.
 - Windows 2000의 경우, ndserver는 자동으로 시작되는 서비스로서 실행됩니다.
2. Dispatcher의 실행 프로그램 기능을 시작하십시오.
ndcontrol executor start
3. 클러스터 주소를 Dispatcher 구성에 추가하십시오.
ndcontrol cluster add www.intersplash.com
4. Http 프로토콜 포트를 Dispatcher 구성에 추가하십시오.
ndcontrol port add www.intersplash.com:80
5. 각 웹 서버를 Dispatcher 구성에 추가하십시오.
ndcontrol server add www.intersplash.com:80:server2.intersplash.com
ndcontrol server add www.intersplash.com:80:server3.intersplash.com
6. 클러스터 주소의 통신량을 승인하려면, 다음과 같이 워크스테이션을 구성하십시오.
ndcontrol cluster configure www.intersplash.com
7. Dispatcher의 관리 프로그램 기능을 시작하십시오.
ndcontrol manager start
Dispatcher는 서버 성능에 따라 로드 밸런스를 수행합니다.

8. Dispatcher의 권고자 기능을 시작하십시오.

ndcontrol advisor start http 80

Dispatcher에서는 실패한 웹 서버로 클라이언트 요청이 전송되지 않았음을 확인합니다.

로컬로 연결된 서버의 기본 구성을 완료했습니다.

구성 마법사를 사용한 구성

구성 마법사를 사용 중이면, 다음 단계를 수행하십시오.

1. Dispatcher에서 ndserver를 시작하십시오.

- AIX, Linux 또는 Solaris의 경우, 다음을 루트 사용자로서 실행하십시오.
ndserver
- Windows 2000의 경우, ndserver는 자동으로 시작되는 서비스로서 실행됩니다.

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

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

구성 마법사를 사용하면, 다음과 같은 패널이 표시됩니다.

- 마법사 소개
- 발생 내용
- 설치 준비
- 구성할 호스트 선택(필요한 경우)
- 클러스터 정의
- 포트 추가
- 서버 추가
- 권고자 시작
- 서버 시스템 설치

GUI(Graphical User Interface)를 사용한 구성

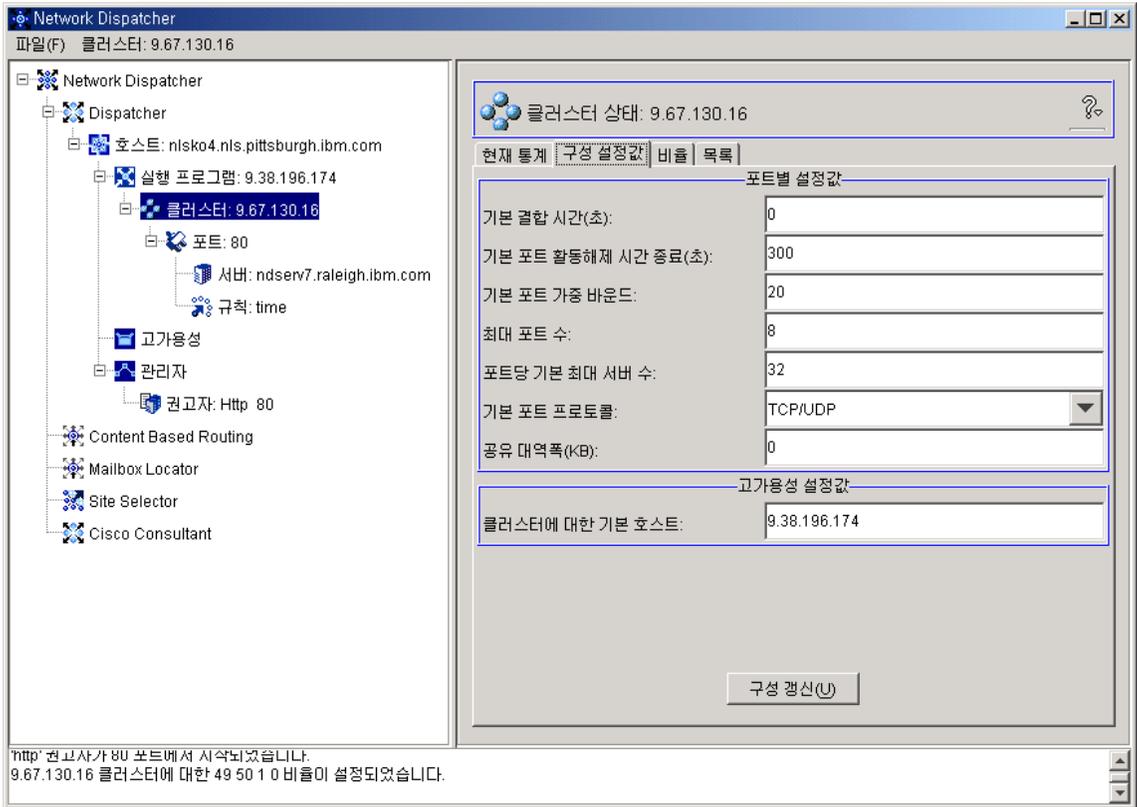


그림 2. GUI(Graphical User Interface)

그래픽 사용자 인터페이스를 시작하려면 다음 단계를 수행하십시오.

1. ndserver가 실행 중인지 확인하십시오.

- AIX, Linux 또는 Solaris의 경우, 다음을 루트로서 실행하십시오.

ndserver

- Windows 2000의 경우, ndserver는 자동으로 시작되는 서비스로서 실행됩니다.

2. 그런 후 다음 중 하나를 수행하십시오.

- AIX, Linux 또는 Solaris의 경우, **ndadmin**을 입력하십시오.

- Windows 2000의 경우, 시작, 프로그램, **IBM WebSphere, Edge Server, IBM Network Dispatcher** 및 **Network Dispatcher**를 차례로 누르십시오.

GUI 사용에 대한 일반 지침

패널의 왼쪽에는 맨 위 레벨에 Network Dispatcher가 있고 구성요소로서 Dispatcher, Content Based Routing, Mailbox Locator, Site Selector 및 Cisco Consultant가 있는 트리 구조가 표시됩니다. 6 페이지의 그림2를 참조하십시오.

모든 구성요소를 GUI에서 구성할 수 있습니다. 마우스 단추(일반적으로, 왼쪽 단추)를 한 번 눌러 트리 구조에서 요소를 선택한 다음, 마우스 단추(일반적으로, 오른쪽 단추)를 두 번 눌러 팝업 메뉴를 표시할 수 있습니다. 트리 요소에 대한 팝업 메뉴는 창의 맨 위에 위치한 메뉴 표시줄에서도 액세스할 수 있습니다.

더하기 또는 빼기 부호를 눌러서 트리 구조의 항목을 펼치거나 압축할 수 있습니다.

창의 오른쪽에는 현재 선택한 요소의 상태 지시자 탭이 표시됩니다.

- 현재 통계 탭은 요소에 대한 통계 정보를 표시합니다.
- 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 구성요소의 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 149 페이지의 『제14장 고급 Network Dispatcher 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 누르면 강조표시된 항목이 삭제됩니다.

도움말에 액세스하려면, Network Dispatcher 창의 오른쪽 상단 모서리에서 물음표를 누르십시오.

- 필드 도움말 -- 각 필드 및 기본값을 설명합니다.
- 수행 방법 -- 해당 화면에서 수행할 수 있는 태스크가 나열되어 있습니다.
- 목차 -- 모든 도움말 정보의 목차입니다.
- 색인 -- 도움말 항목의 영문자 색인입니다.

구성 검사

구성이 작동되고 있는지 확인하십시오.

1. 웹 브라우저에서 **http://www.intersplash.com** 위치로 이동하십시오. 페이지가 표시되면, 모두 작동되고 있는 것입니다.
2. 웹 브라우저에서 페이지를 재로드하십시오.
3. **ndcontrol server report www.intersplash.com:80:** 명령의 결과를 확인하십시오. 두 서버의 총 연결 컬럼은 “2”까지 추가해야 합니다.

클러스터, 포트, 서버 구성의 유형

사용자 사이트를 지원하기 위해 Network Dispatcher를 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 Network Dispatcher가 통신할 포트를 구성합니다. 9 페이지의 그림3을 참조하십시오.

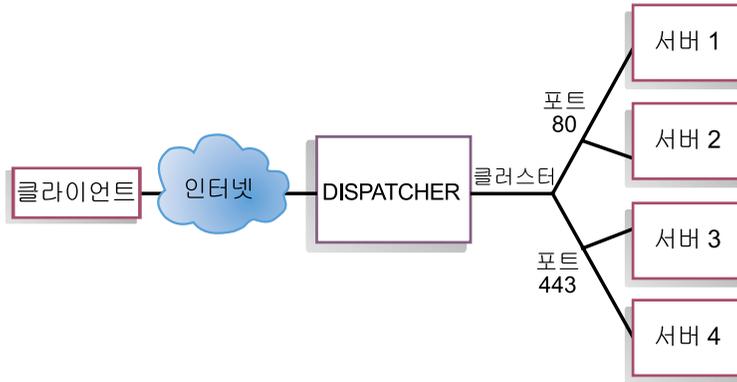


그림 3. 단일 클러스터와 두 개의 포트에 구성된 Dispatcher에 대한 예제

Dispatcher 구성요소 예제에서는 하나의 클러스터가 www.productworks.com에 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 Network Dispatcher 구성 방법이 적합합니다. 이 경우, 10 페이지의 그림4에 표시된 대로 단일 포트이지만 많은 서버가 있는 각 프로토콜마다 클러스터를 정의하고자 할 수 있습니다.

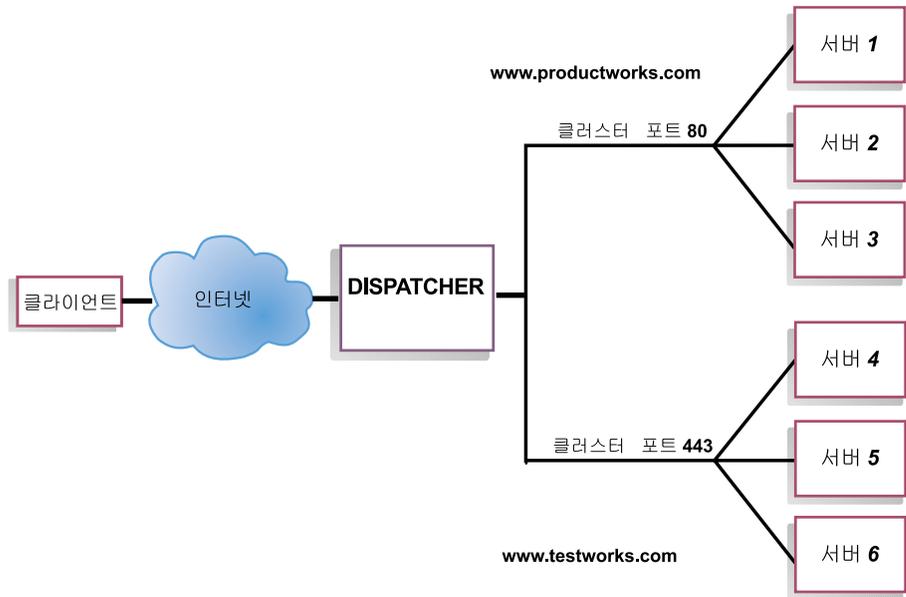


그림 4. 각각 단일 포트인 두 개의 클러스터로 구성된 Dispatcher에 대한 예제

Dispatcher 구성요소 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 Network Dispatcher를 구성하기 위한 세 번째 방법이 필요합니다. 이 경우에는 11 페이지의 그림5에 표시된 것처럼 회사나 부서의 클러스터를 각각 정의한 다음, 해당 URL에서 연결을 받을 포트를 정의할 수 있습니다.

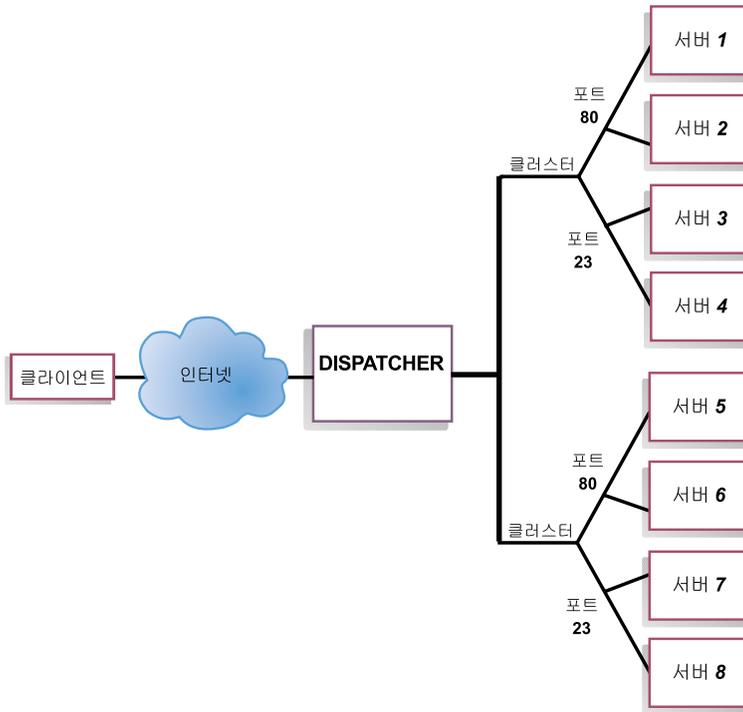


그림 5. 각각 포트가 두 개인 두 개의 클러스터로 구성된 Dispatcher에 대한 예제

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

제2장 Network Dispatcher 설치

이 장에서는 AIX, Linux, Solaris 및 Windows 2000에 Network Dispatcher를 설치하는 방법과 하드웨어 요구사항을 설명합니다. 다음의 지침을 참조하십시오.

- 15 페이지의 『AIX의 요구사항』
- 20 페이지의 『Red Hat Linux 또는 SuSE Linux 요구사항』
- 23 페이지의 『Solaris 요구사항』
- 26 페이지의 『Windows 2000 요구사항』

주:

1. 이전 버전에서 이주하는 경우, Network Dispatcher 설치 디렉토리 구조가 변경되었음에 주의하십시오. 사용자 전용 구성 파일을 **...nd/servers/configurations/component** 디렉토리로 이동해야 합니다(여기서 **component**는 dispatcher, cbr, ml, ss 또는 lbc입니다). 또한 고유 스크립트(예: goIdle, goStandby)는 **...nd/servers/bin** 디렉토리로 이동해야 합니다.
2. Network Dispatcher가 설치된 후 시스템에서 로그오프한 경우 다시 로그인할 때 모든 Network Dispatcher 서비스를 다시 시작해야 합니다.
3. Network Dispatcher 릴리스 2.0용 필수 Java 레벨은 1.3.0 이상입니다. Network Dispatcher 상자에 있는 일부 응용프로그램은 다른 버전의 Java를 요구할 수도 있으므로 업그레이드할 경우 상자에 정확한 Java 버전이 설치되어 있어야 합니다.

여러 버전의 Java가 설치되어 있는 경우 Network Dispatcher 구성요소가 정확한 Java 버전을 사용하도록 다음을 수행하십시오.

- a. 이 장의 요구사항 섹션에 지정되어 있는 것처럼, 운영 체제에 맞는 정확한 버전의 Java 1.3을 설치하십시오.
- b. Java 1.3을 사용하도록 Network Dispatcher 스크립트 파일을 편집하십시오. 기본적으로 스크립트 파일은 다음 디렉토리에 있습니다.

Unix 기반

`/usr/bin/<scriptfile>`

Windows

C:\WINNT\System32*<scriptfile.cmd>*

업그레이드하는 Network Dispatcher의 각 구성요소에 해당하는 스크립트 파일을 편집하십시오. 각 구성요소에 대한 스크립트 파일은 다음과 같습니다.

Administration

ndadmin

Dispatcher

ndserver, ndcontrol, ndwizard, ndkeys

CBR(Content Based Routing)

cbrserver, cbrcontrol, cbrwizard, cbrkeys

Site Selector

sssserver, ssscontrol

Cisco Consultant

lbserver, lbcontrol

주: 기본적으로 이들 파일은 읽기 전용입니다. 따라서 변경사항을 저장하기 전에 이들 파일의 사용 권한을 변경하십시오.

- c. 스크립트 파일에 java 또는 javaw 명령이 있는 경우, Java 1.3 설치 디렉토리에 있는 명령을 표시하는 접두부를 경로에 추가하십시오.

예를 들어, Windows 2000에서 Java 1.3이 C:\Program Files\IBM\Java13\jre\bin에 설치되어 있는 경우 ndserver.cmd의 행을 다음과 같이 변경하십시오.

변경 전:

```
javaw %END_ACCESS% -DEND_INSTALL_PATH=  
%IBMNDPATH% ..
```

변경 후:

```
C:\Program Files\IBM\Java13\jre\bin\javaw %END_ACCESS%  
-DEND_INSTALL_PATH=%IBMNDPATH% ...
```

AIX의 요구사항

- IBM RS/6000 기반 시스템
- IBM AIX 5.1(APAR IY19177). 32비트 파워 PC(64비트 커널이 아님)를 지원합니다.

IBM AIX 4.3.3.10 + apars(Java 1.3 지원용). 필수 AIX apars 목록은 IBM AIX Developer Kit용 README를 참조하십시오.

- 설치에 필요한 50MB의 빈 디스크 공간

주: 로그에 추가 디스크 공간이 필요합니다.

- 다음 네트워크 인터페이스 카드(NIC)가 지원됩니다.
 - 16Mb 토큰링
 - 10Mb 이더넷
 - 100Mb 이더넷
 - 1Gb 이더넷
 - FDDI(Fiber distributed data interface)
 - 멀티포트 이더넷 NIC

주: 멀티포트 NIC 구현은 공급 업체마다 다릅니다. 따라서 일부 멀티포트 NIC에 대해서만 지원될 수 있습니다.

- Java 런타임 환경용 IBM AIX Developer Kit, Java 2 Technology Edition, 버전 1.3.0 이상(여러 버전의 Java 실행에 대한 정보는 주 번호 13 페이지의 3 참조)
- HTTP 또는 SSL 통신량을 로드 밸런싱하기 위해 CBR 구성요소를 사용 중인 경우, Edge Server Caching Proxy V2.0
- 온라인 도움말을 보기 위한 Netscape Navigator 4.07 이상 또는 Netscape Communicator 4.61 이상
- Consultant for Cisco CSS Switches의 경우 Cisco CSS 11000 Series Switch가 설치 및 구성되어야 합니다.

AIX용 설치

테이블 1에는 AIX용 Network Dispatcher의 installp 이미지가 나열되어 있습니다.

표 1. AIX installp 이미지

Dispatcher(구성요소, Administrator, License 및 메시지)	intnd.nd.driver intnd.nd.rte intnd.msg.nd.<language>.nd intnd.admin.rte intnd.msg.<language>.admin
Administrator(전용)	intnd.admin.rte intnd.msg.<language>.admin
Documentation	intnd.doc.rte
License	intnd.nd.license
Metric Server	intnd.ms.rte

여기서 <language>는 다음 중 하나입니다.

- en_US
- de
- es_ES
- fr
- it
- ja_JP
- Ja_JP
- ko_KR
- pt_BR
- zh_CN
- zh_TW
- Zh_TW

웹 사이트에서 제품 평가판을 다운로드하는 중이면 (<http://www.ibm.com/software/webservers/edgeserver/download.html>)의 설치 지침을 따르십시오.

설치하기 전에

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

- ND Administration
- ND Dispatcher 장치 드라이버(ND Dispatcher에 필수)
- ND License(ND Dispatcher에 필수)
- ND Documentation
- ND Metric Server
- 사용권

설치 단계

주: 이미 이전 버전이 설치되어 있으면, 현재 버전을 설치하기 전에 이전 버전의 설치를 제거해야 합니다. 먼저, 모든 실행 프로그램과 모든 서버가 정지되었는지 확인하십시오. 전체 제품을 설치 제거하려면, **installp -u intnd**를 입력하십시오. 특정 파일 세트를 설치 제거하려면, 패키지 이름을 지정하지 말고 명확하게 해당 파일 세트를 나열하십시오.

AIX용 Network Dispatcher를 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 설치되어야 하므로, SMIT를 사용하여 AIX용 Network Dispatcher를 설치하는 것이 바람직합니다.

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. **SMITTY**를 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
```

필요한 AIX용 Network Dispatcher 패키지를 설치하기 위해 입력할 명령을 판별하려면 다음 테이블을 참조하십시오.

표 2. AIX 설치 명령

Network Dispatcher(메시지). Dispatcher, CBR, Mailbox Locator, Site Selector 및 Cisco Consultant가 포함됩니다.	<code>installp -acXgd device intnd.nd.rte intnd.admin.rte intnd.nd.driver intnd.msg.<language>.nd intnd.msg.<language>.admin</code>
Documentation	<code>installp -acXgd device intnd.doc.rte intnd.msg.<language>.doc</code>
Administrator(전용)	<code>installp -acXgd device intnd.admin.rte intnd.msg.<language>.admin</code>
License	<code>installp -acXgd device intnd.nd.license</code>
Metric Server	<code>installp -acXgd device intnd.ms.rte intnd.msg.<language>.admin</code>

여기서 *device*는 다음과 같습니다.

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

요약의 결과 열에 설치할 Network Dispatcher의 각 파트마다 SUCCESS가 표시되어 있는지 확인하십시오(APPLYing). 설치할 모든 파트가 적용될 때까지 계속 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 intnd
```

전체 제품이 설치되고 나면 이 명령은 다음을 표시합니다.

```
intnd.admin.rte
intnd.doc.rte
intnd.ms.rte
intnd.msg.en_US.admin.rte
intnd.msg.en_US.doc
intnd.msg.en_US.nd.rte
intnd.nd.driver
intnd.nd.license
intnd.nd.rte
```

Network Dispatcher 설치 경로는 다음과 같습니다.

- Administrator - **/usr/lpp/nd/admin**
- Network Dispatcher 구성요소 - **/usr/lpp/nd/servers**
- Metric Server - **/usr/lpp/nd/ms**
- Documentation(관리 안내서) - **/usr/lpp/nd/documentation**

Red Hat Linux 또는 SuSE Linux 요구사항

- Red Hat Linux 버전 7.1(Linux 커널 버전 2.4.2-2) 또는 SuSE Linux 버전 7.1(Linux 커널 버전 2.4.0-4GB). 단일 프로세서 및 멀티프로세서 커널 모두 지원됩니다.

주: 고가용성 및 결합 배치와 함께 Dispatcher의 MAC 전달 방법을 사용 중인 경우 Linux 커널 패치를 설치해야 합니다. 83 페이지의 『Linux 커널 패치 설치(루프백 인터페이스에서 arp 응답 억제)』에서 패치 다운로드 및 설치 방법에 대한 정보를 참조하십시오.

- 설치에 필요한 50MB의 빈 디스크 공간

주: 로그에 추가 디스크 공간이 필요합니다.

- 다음 네트워크 인터페이스 카드(NIC)가 지원됩니다.
 - 10Mb 이더넷
 - 100Mb 이더넷
 - 1Gb 이더넷
 - 멀티포트 이더넷 NIC(모드 1만 지원, 결합 허용(모드 2) 및 포트 집약(모드 3)은 지원되지 않음)

주: 멀티포트 NIC 구현은 공급 업체마다 다릅니다. 따라서 일부 멀티포트 NIC에 대해서만 제한적으로 지원될 수 있습니다.

- Korn Shell(ksh)이 설치되어 있어야 합니다.
- Linux용 IBM Runtime Environment, Java 2 Technology Edition, 버전 1.3.0 이상(여러 버전의 Java 실행에 대한 정보는 주 번호 13 페이지의 3 참조)
- JAVA_HOME 및 PATH 환경 변수는 **export** 명령을 사용하여 설정되어야 합니다. JAVA_HOME 변수의 내용은 사용자의 시스템에서 Java가 설치된 위치에 따라 달라집니다. 다음을 예로 들 수 있습니다.
 - JAVA_HOME=/opt/IBMJava2-13/jre
 - PATH=\$JAVA_HOME/bin:\$PATH
- HTTP 또는 SSL 통신량을 로드 밸런싱하기 위해 CBR 구성요소를 사용 중인 경우, Edge Server Caching Proxy V2.0

- 온라인 도움말을 보기 위한 Netscape Navigator 4.07 이상 또는 Netscape Communicator 4.61 이상
- Consultant for Cisco CSS Switches의 경우 Cisco CSS 11000 Series Switch가 설치 및 구성되어야 합니다.

Linux에 설치

여기서는 제품 CD나 웹 사이트에서 다운로드한 제품 평가 사본을 사용하여 Red Hat Linux 또는 SuSE Linux에 Network Dispatcher를 설치하는 방법을 설명합니다. 웹 사이트(<http://www.ibm.com/software/webservers/edgeserver/download.html>)에 설치 지침이 있습니다.

설치하기 전에

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

설치 단계

주: 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.

Network Dispatcher를 설치하려면 다음과 같이 하십시오.

1. 설치를 준비하십시오.

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

주: Red Hat Linux용 설치 패키지와 SuSE Linux용 설치 패키지는 다른 Linux 제품 버전에서 실행할 수 없습니다.

설치 이미지는 **ndlinux-version.tar** 형식의 파일입니다.

- **tar -xf ndlinux-version.tar** 명령을 입력하여 임시 디렉토리에 tar 파일을 해제하십시오. 그러면 .rpm 확장자를 가진 파일 집합이 생성됩니다. 다음은 RPM 설치 가능 패키지의 목록입니다.
 - **ibmnd-adm-release-version.i386.rpm**(ND Administration)
 - **ibmnd-doc-release-version.i386.rpm**(Documentation)
 - **ibmnd-ms-release-version.i386.rpm**(Metric Server)
 - **ibmnd-srv-release-version.i386.rpm** (Network Dispatcher Runtime)
 - **ibmnd-lic-release-version.i386.rpm**(License)
- 패키지가 설치되는 순서는 중요합니다. 다음은 각 구성요소에 필요한 패키지 목록과 준수해야 하는 설치 순서입니다.
 - Administration(adm)
 - License(lic)
 - Network Dispatcher 구성요소(srv)
 - Metric Server(ms)
 - Documentation(doc)

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

주: 적어도 하나 이상의 RPM 파일이 Java가 설치되고 RPM 데이터베이스에 등록되도록 요구합니다. Java가 설치되었으나 RPM 데이터베이스에 등록되지 않으면 다음과 같이 'no dependencies' 옵션과 함께 install 명령을 사용하십시오.

rpm -i --nodeps package.rpm

- Network Dispatcher 설치 경로는 다음과 같습니다.
 - 관리 - **/opt/nd/admin**
 - Network Dispatcher 구성요소 - **/opt/nd/servers**
 - Metric Server - **/opt/nd/ms**
 - 문서(관리 안내서) - **/opt/nd/documentation**

- 패키지를 설치 제거할 때는 패키지 설치에 사용되는 순서를 반대로 따라 작업하여 관리 패키지가 맨 마지막에 설치 제거하십시오.
2. 제품이 설치되어 있는지 확인하십시오. 다음 명령을 입력하십시오.

rpm -qa | grep ibmnd

전체 제품을 설치하면 다음과 같은 목록이 생성됩니다.

- *ibmnd-adm-release-version*
- *ibmnd-doc-release-version*
- *ibmnd-ms-release-version*
- *ibmnd-srv-release-version*
- *ibmnd-lic-release-version*

Solaris 요구사항

- Solaris 버전 7 또는 Solaris 버전 8에서 지원하는 SPARC 워크스테이션 또는 Ultra 60 서버. Network Dispatcher는 Solaris 플랫폼에서 32비트 모드만 지원됩니다.
- 설치에 필요한 50MB의 빈 디스크 공간
 - 주: 로그에 추가 디스크 공간이 필요합니다.
- 다음 네트워크 인터페이스 카드(NIC)가 지원됩니다.
 - 10Mb 이더넷
 - 100Mb 이더넷
 - 1Gb 이더넷(Ultra 60 서버에서만 지원됨)
 - 멀티포트 이더넷 NIC(모드 1만 지원, 결합 허용(모드 2) 및 포트 집약(모드 3)은 지원되지 않음)
 - 주: 멀티포트 NIC 구현은 공급 업체마다 다릅니다. 따라서 일부 멀티포트 NIC에 대해서만 지원될 수 있습니다.
- Java 2 JRE, Standard Edition, 버전 1.3.0 이상(여러 버전의 Java 실행에 대한 자세한 정보는 주 번호 13 페이지의 3 참조)

- HTTP 또는 SSL 통신량을 로드 밸런싱하기 위해 CBR 구성요소를 사용 중인 경우, Edge Server Caching Proxy V2.0
- Solaris 7의 경우, 온라인 도움말을 보려면 Sun Microsystems HotJava Browser 1.0.1 이상
Solaris 8의 경우, 온라인 도움말을 보려면 Netscape Navigator 4.07 이상 또는 Netscape Communicator 4.61 이상
- Consultant for Cisco CSS Switches의 경우 Cisco CSS 11000 Series Switch가 설치 및 구성되어야 합니다.

Solaris에 설치

이 절에서는 제품 CD를 사용하여 Solaris에 Network Dispatcher를 설치하는 방법에 대해 설명합니다. 인터넷에서 제품 평가판을 다운로드하는 중이면, 웹 사이트 (<http://www.ibm.com/software/webservers/edgeserver/download.html>)의 설치 지침을 따르십시오.

설치하기 전에

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

설치 단계

주: 이미 이전 버전이 설치되어 있으면, 현재 버전을 설치하기 전에 이전 버전의 설치를 제거해야 합니다. 먼저 실행 프로그램과 서버를 모두 중단하십시오. 그런 다음 Network Dispatcher를 설치 제거하려면 **pkgrm pkgname**을 입력하십시오.

Network Dispatcher를 설치하려면 다음을 수행하십시오.

1. 설치를 준비하십시오.
 - 루트 사용자로 로그인하십시오.
 - Network Dispatcher 소프트웨어가 들어 있는 CD-ROM을 해당 드라이브에 넣으십시오.

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

설치할 패키지 목록이 제공됩니다. 목록은 다음과 같습니다.

- Solaris용 **ibmdsp** IBM ND(Network Dispatcher 구성요소)
- Solaris용 **ibmndadm** IBM ND 기본 Administration
- Solaris용 **ibmnddoc** IBM ND Documentation
- Solaris용 **ibmndms** IBM ND Metric Server
- Solaris용 **ibmdsplic** License

모든 패키지를 설치하려면 “all”을 입력한 후 리턴 키만 누르면 됩니다. 일부 구성요소만 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.

모든 패키지는 공통 패키지인 **ibmndadm**에 기초합니다. 이 공통 패키지는 다른 패키지와 함께 설치되어야 합니다.

전체 Network Dispatcher 구성요소를 설치하려면 **ibmdsp**, **ibmdsplic**, **ibmndadm**, **ibmnddoc** 및 **ibmndms**를 설치해야 합니다. 원격 Administration을 설치하려면 **ibmndadm**만 설치하면 됩니다.

Network Dispatcher 구성요소는 **/opt/nd/servers** 설치 디렉토리에 있습니다.

2. 설치된 Administration **/opt/nd/admin** 디렉토리에 있습니다.
3. 설치된 Metric Server는 **/opt/nd/ms** 디렉토리에 있습니다.
4. 설치된 Documentation(관리 안내서)은 **/opt/nd/documentation** 디렉토리에 있습니다.
5. 제품이 설치되어 있는지 확인하십시오. **pkginfo | grep ibm** 명령을 실행하십시오.

전체 제품을 설치하면 다음과 같은 목록이 생성됩니다.

- ibmdsp
- ibmndadm
- ibmnddoc
- ibmndms
- ibmdsplic

Windows 2000 요구사항

- Microsoft Windows 2000이 지원하는 Intel x86 PC
- Windows 2000 Professional, Server 또는 Advanced Server
- 설치에 필요한 50MB의 빈 디스크 공간

주: 로그에 추가 디스크 공간이 필요합니다.

- 다음 네트워크 인터페이스 카드(NIC)가 지원됩니다.
 - 16Mb 토큰링
 - 10Mb 이더넷
 - 100Mb 이더넷
 - 1Gb 이더넷
 - 멀티포트 이더넷 NIC

주: 멀티포트 NIC 구현은 공급 업체마다 다릅니다. 따라서 일부 멀티포트 NIC에 대해서만 지원될 수 있습니다.

- Windows v2.0용 IBM Cross Platform Technologies(SDK 1.3.0 이상)
InstallShield 프로그램을 실행하기 전에 Developer Kit 설치 가능 패키지와 런타임 환경 설치 가능 패키지를 모두 다운로드해야 합니다(여러 버전의 Java 실행에 대한 자세한 정보는 주 번호 13 페이지의 3 참조).
- HTTP 또는 SSL 통신량을 로드 밸런싱하기 위해 CBR 구성요소를 사용 중인 경우, Edge Server Caching Proxy V2.0

- 기본 브라우저는 Navigator 4.07 이상, Netscape Communicator 4.61 이상 또는 Internet Explorer 4.0 이상이어야 합니다. 기본 브라우저는 온라인 도움말을 보는 데 사용됩니다.
- Consultant for Cisco CSS Switches의 경우, Cisco CSS 11000 Series Switch가 설치 및 구성되어야 합니다.

Windows 2000에 설치

여기서는 제품 CD를 사용하여 Windows 2000에 Network Dispatcher를 설치하는 방법을 설명합니다. 웹 사이트에서 제품 평가판을 다운로드하는 중이면 웹 사이트(<http://www.ibm.com/software/webservers/edgeserver/download.html>)의 설치 지침을 따르십시오.

설치 패키지

설치할 패키지 선택사항이 제공됩니다.

목록은 다음과 같습니다.

- Runtime
- Administration
- License
- Documentation
- Metric Server

설치하기 전에

다음에서 Windows 2000 버전의 Network Dispatcher를 사용할 수 있습니다.

- Windows 2000 Professional
- Windows 2000 Server
- Windows 2000 Advanced Server

주: Network Dispatcher Windows 2000 버전은 다른 Windows 버전에서는 실행되지 않습니다.

제한사항: Network Dispatcher Windows 2000 버전을 IBM Firewall을 사용하여 동일한 시스템에 설치할 수 없습니다.

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

설치 단계

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->제어판을 누르십시오.
2. 프로그램 추가/제거를 두 번 누르십시오.
3. *Network Dispatcher*를 선택하십시오.
4. 변경/제거 단추를 누르십시오.

*Network Dispatcher*를 설치하려면 다음과 같이 하십시오.

1. CD-ROM 드라이브에 *Network Dispatcher* CD-ROM을 넣으면 설치 창이 자동으로 표시됩니다.
2. 다음 단계는 사용자 컴퓨터에서 CD의 자동 실행이 작동되지 않는 경우에만 필요합니다. 마우스 왼쪽 단추를 눌러 다음 태스크를 수행하십시오.
 - 시작을 누르십시오.
 - 실행을 선택하십시오.
 - CD-ROM 디스크 드라이브를 지정한 후, 다음 예와 같이 *setup.exe*를 지정하십시오.

E:\setup

3. 설치할 언어를 선택하십시오.
4. 확인을 누르십시오.
5. 설치 프로그램의 지침을 따르십시오.
6. 드라이브 또는 디렉토리 목적지를 변경하려면, 찾아보기를 누르십시오.
7. “모든 ND 제품” 또는 “구성요소 선택” 중 하나를 선택할 수 있습니다.

8. 설치가 끝나면 Network Dispatcher를 사용하기 전에 시스템을 재부트하라는 메시지가 표시됩니다. 이 작업을 수행해야만 모든 파일이 제대로 설치되고 IBMNDPATH 환경 변수가 레지스트리에 제대로 추가되었는지 확인할 수 있습니다.

Network Dispatcher 설치 경로는 다음과 같습니다.

- Administration - **c:\Program~1\IBM\edge\nd\admin**
- Network Dispatcher 구성요소 - **c:\Program~1\IBM\edge\nd\servers**
- Metric Server - **c:\Program~1\IBM\edge\nd\ms**
- Documentation(관리 안내서) - **c:\Program~1\IBM\edge\nd\documentation**

제3장 Network Dispatcher 소개

이 장에는 Network Dispatcher 개요와 다음 섹션으로 구성되어 있습니다.

- 『Network Dispatcher 개념』
- 32 페이지의 『Network Dispatcher가 필요한 이유』
- 34 페이지의 『새로운 기능』
- 40 페이지의 『Network Dispatcher 구성요소 종류』
- 53 페이지의 『고가용성 정보』

Network Dispatcher 개념

Network Dispatcher는 로드 밸런스 서버용 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. Network Dispatcher는 전자 우편 서버, 월드 와이드 웹 서버, 분산 병렬 데이터베이스 조회 및 다른 TCP/IP 응용프로그램과 같은 응용프로그램에 유용합니다.

웹 서버에 사용하면, Network Dispatcher는 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, Network Dispatcher는 자동으로 최적의 서버를 찾아 수신 요청을 처리하므로 고객 만족도와 사용자의 수익성이 향상됩니다.

Network Dispatcher는 탁월한 로드 밸런스 결과를 제공하기 위해 독립적으로 사용하거나 함께 사용할 수 있는 다섯 개의 구성요소로 이루어져 있습니다.

- **Dispatcher** 구성요소를 사용하여 Dispatcher에서 동적으로 설정된 다수의 가중치와 측정치를 사용하는 근거리 통신망 또는 광역 통신망 내에서 서버의 로드 밸런스를 수행할 수 있습니다. 이 구성요소는 HTTP, FTP, SSL, NNTP,

IMAP, POP3, SMTP 및 텔넷과 같은 특정 서비스 레벨의 로드 밸런스를 제공합니다. 이 구성요소는 도메인 이름을 IP 주소에 맵핑하기 위해 도메인 이름 서버를 사용하지 않습니다.

HTTP 프로토콜의 경우, Dispatcher content-based routing 기능을 사용하여 클라이언트 요청 내용을 기반으로 로드 밸런스를 수행할 수 있습니다. 선택된 서버는 지정된 규칙과 일치하는 URL 결과입니다.

- HTTP 및 HTTPS(SSL) 프로토콜의 경우, **CBR(Content Based Routing)** 구성요소를 사용하여 클라이언트 요청 내용을 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.
- IMAP 및 POP3 프로토콜의 경우, 클라이언트가 제공하는 사용자 ID 및 암호를 기반으로 적절한 서버를 선택하고 프록시로 기능하는 **Mailbox Locator** 구성요소를 사용할 수 있습니다.
- **Site Selector** 구성요소를 사용하여 DNS 라운드 로빈 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 맵핑하는 작업을 합니다.
- **Consultant for Cisco CSS Switches** 구성요소를 사용하여 최적 서버 선택, 로드 최적화 및 결합 허용을 위해 Cisco CSS Switch로 전송된 서버 가중 메트릭을 생성할 수 있습니다.

Dispatcher, CBR, Mailbox Locator, Site Selector 및 Consultant for Cisco CSS Switches 구성요소에 대한 자세한 정보는 40 페이지의 『Network Dispatcher 구성요소 종류』를 참조하십시오.

Network Dispatcher가 필요한 이유

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

현재, 네트워크 관리자는 다양한 방법을 사용하여 액세스를 최대화하려고 노력하고 있습니다. 이러한 방법 중 일부에서는 사용자가 이전 선택이 느리거나 응답이 없을 때 임의로 다른 서버를 선택할 수 있습니다. 이러한 접근 방식은 다루기 번거로우며 비효율적입니다. 다른 방법으로는 표준 라운드 로빈 방식이 있는 데, 이 방법은 도메인 이름 서버가 요청을 처리하기 위해 차례대로 서버를 선택합니다. 이 방식이 더 낫기는 하지만 서버의 작업 부하를 고려하지 않고 너무 많은 통신량을 전송하므로 여전히 비효율적입니다. 또한 서버가 실패할 경우에도 요청은 계속 전송됩니다.

더 강력한 솔루션의 필요성에 대한 결과물이 Network Dispatcher입니다. 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 로빈 방법을 사용할 때 자주 발생하는 핫 스팟(hot spot)을 최소화하여 하드웨어 사용을 최적화합니다.

용이한 통합

Network Dispatcher는 표준 TCP/IP 프로토콜을 사용합니다. 네트워크를 물리적으로 변경하지 않고도 기존의 네트워크에 이를 추가할 수 있습니다. 설치하고 구성하는 방법도 아주 간단합니다.

낮은 오버헤드

간단한 MAC 레벨 전달 방법을 사용하면, Dispatcher는 클라이언트에서 서버로 들어오는 인바운드 플로우만 살펴야 합니다. 서버에서 클라이언트로의 아웃바운드 흐름은 보지 않아도 됩니다. 이로 인해, 다른 접근 방식에 비해 응용프로그램으로의 영향력이 현저하게 감소되며 네트워크의 성능은 향상될 수 있습니다.

고가용성

Dispatcher는 내장형 고가용성을 제공하며, 기본 Dispatcher 시스템에 오류가 발생할 경우, 로드 밸런스를 언제든지 인계 받도록 계속 준비 상태로 남아 있는 대기 시스템을 이용합니다. 또한 Dispatcher는 두 시스템이 모두 활성 상태이거나 대기 상태가 되도록 허용하는 상호 고가용성 기능도 제공합니다. 53 페이지의 『고가용성 정보』를 참조하십시오.

Content Based Routing(CBR 구성요소 또는 Dispatcher 구성요소 사용)

Caching Proxy와 함께 CBR 구성요소는 요청된 내용을 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 집합 중 최상의 서버로 요청을 지정할 수 있습니다.

Dispatcher 구성요소는 Content Based Routing을 제공하기도 하지만 Caching Proxy는 설치하지 않아도 됩니다. 패킷이 수신될 때 Dispatcher 구성요소의 Content Based Routing이 커널에서 수행되기 때문에 CBR 구성요소에 비해 신속한 Content Based Routing을 제공할 수 있습니다. Dispatcher 구성요소는 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 내용을 기반으로 통신량을 로드 밸런싱할 때 CBR 구성요소만이 HTTPS(SSL)의 콘텐츠 규칙을 사용할 수 있습니다.

새로운 기능

IBM WebSphere Edge Server 버전 2.0용 Network Dispatcher에는 여러 가지 새로운 기능이 있습니다. 여기서는 가장 중요한 기능만 나열합니다.

- **AIX v5.1 지원**

이 기능은 모든 Network Dispatcher 구성요소에 적용됩니다.

이제 Network Dispatcher는 AIX 최신 버전(AIX v5.1)을 지원합니다. 15 페이지의 『AIX의 요구사항』에서 자세한 정보를 참조하십시오.

- **SuSE Linux v7.1 지원**

이 기능은 모든 Network Dispatcher 구성요소에 적용됩니다.

이제 Network Dispatcher는 SuSE Linux v7.1(커널 버전 2.4.0-4GB)을 지원합니다. 이전에 Network Dispatcher는 Red Hat Linux만 지원했습니다. 20 페이지의 『Red Hat Linux 또는 SuSE Linux 요구사항』에서 자세한 정보를 참조하십시오.

- **Red Hat Linux v7.1 지원**

이 기능은 모든 Network Dispatcher 구성요소에 적용됩니다.

이제 Network Dispatcher는 Red Hat Linux 최신 버전인 Red Hat Linux v7.1(커널 버전 2.4.2-2)을 지원합니다. 20 페이지의 『Red Hat Linux 또는 SuSE Linux 요구사항』에서 자세한 정보를 참조하십시오.

- **Linux 및 Solaris NLS(National Language Support)**

이 기능은 모든 Network Dispatcher 구성요소에 적용됩니다.

Linux 및 Solaris 운영 체제에서 Network Dispatcher는 그룹 1 국가에 NLS를 제공합니다.

- **새로운 중국어 NLS 표준 지원**

이 기능은 모든 Network Dispatcher 구성요소에 적용됩니다.

Network Dispatcher는 새로운 중국어 표준 GB 18030에 NLS를 제공합니다.

- **Consultant for Cisco CSS Switches 구성요소(Cisco Consultant)**

이 기능은 Network Dispatcher의 새로운 구성요소입니다.

Cisco 및 CDN(Content Distribution Network)이 공동으로 Network Dispatcher의 추가 구성요소 즉, Cisco Consultant를 개발하였습니다. 이 구성요소(독립형 Preview로 처음 소개됨)를 사용하면 Network Dispatcher가 Cisco CSS Switch에 대해 가중치를 생성하여 로드 밸런스 결정을 내릴 수 있습니다.

자세한 정보는 135 페이지의 『제12장 Consultant for Cisco CSS Switches 구성요소에 대한 계획』 및 141 페이지의 『제13장 Consultant for Cisco CSS Switches 구성요소 구성』을 참조하십시오.

- **Site Selector 구성요소**

이 기능은 Network Dispatcher의 새로운 구성요소입니다.

Site Selector 구성요소는 이름 서비스 요청에 "적합한" 서버의 IP 주소를 선택하여 서버 그룹 사이에서 로드 밸런스를 수행합니다. 따라서 클라이언트는 모든 통신에 대해 서버에 직접 연결할 수 있습니다. Site Selector는 이전 릴리스의 Network Dispatcher 구성요소인 ISS(Interactive Session Support)를 대체합니다. Site Selector는 기능면에서 ISS와 유사하지만 일반적인 DNS 로드 밸런싱 구성 설정 시 요구되는 수동 단계는 더 적습니다.

자세한 정보는 121 페이지의 『제10장 Site Selector 구성요소에 대한 계획』 및 127 페이지의 『제11장 Site Selector 구성요소 구성』을 참조하십시오.

- **Metric Server**

이 기능은 모든 Network Dispatcher 구성요소에 적용됩니다.

Metric Server는 시스템 고유의 메트릭 양식으로 Network Dispatcher에 서버 로드 정보를 제공합니다. Metric Server 에이전트는 Network Dispatcher가 로드 밸런싱을 수행하는 서버에 설치되어 실행될 수 있는 Network Dispatcher의 구성요소입니다. Metric Server는 Linux에서 지원하며, 이전 릴리스의 SMA(System Monitoring Agent)를 대체합니다. Metric Server는 모든 플랫폼에서 지원됩니다. Metric Server를 Site Selector 구성요소와 함께 사용하는 것이 좋습니다.

169 페이지의 『Metric Server』에서 자세한 정보를 참조하십시오.

- **Mailbox Locator 구성요소**

이 기능은 Network Dispatcher의 새로운 구성요소입니다.

Mailbox Locator 구성요소는 이전에는 사용자 ID와 암호를 기반으로 IMAP 및 POP3 메일 서버 간에 로드 밸런싱된 CBR 구성요소 내의 기능이었습니다. CBR을 두 개의 구성요소로 분리하면 Mailbox Locator(이전에는 "IMAP/POP3용 CBR"이라고 함) 및 Caching Proxy가 있는 CBR을 동일한 시스템에서 실행할 수 있습니다.

자세한 정보는 109 페이지의 『제8장 Mailbox Locator 구성요소에 대한 계획』 및 113 페이지의 『제9장 Mailbox Locator 구성요소 구성』을 참조하십시오.

- **CBR(Content Based Routing) 구성요소의 가용성 향상**

CBR 사용을 위한 Caching Proxy 구성 파일(ibmpoxy.conf)을 구성하는 것이 단순해졌고 Caching Proxy의 다중 인스턴스를 CBR과 인터페이스하는 동

안 동일한 시스템에서 동시에 실행할 수 있도록 CBR이 향상되었습니다. Caching Proxy를 사용하여 CBR을 구성하는 방법에 대한 자세한 정보는 101 페이지의 『CBR 시스템 설정』을 참조하십시오.

- **NAT(Network Address Translation) 및 NAPT(Network Address Port Translation) 지원**

이 기능은 Dispatcher 구성요소에 적용됩니다.

NAT/NAPT를 사용하면 백엔드 서버가 로컬로 접속된 네트워크에 위치하도록 하는 제약이 제거됩니다. 또한 Dispatcher가 클라이언트의 TCP 요청을 동일한 실제 시스템에서 실행 중인 다중 서버 디먼으로 로드 밸런스할 수 있도록 합니다. 서로 다른 두 가지 방법으로 다중 디먼이 있는 서버를 구성할 수 있습니다. NAT를 사용하면 서로 다른 IP 주소에 대한 요청에 응답하도록 다중 서버 디먼을 구성할 수 있습니다. 이것을 IP 주소에 서버 디먼을 바인딩한다고 합니다. NAPT를 사용하면 다른 포트 번호를 인식할 수 있도록 다중 서버 디먼을 구성할 수 있습니다.

Dispatcher nat 전달 방법의 장점은 포트 레벨에서 구성되어 보다 나은 세분성을 제공할 수 있다는 점입니다.

주: Network Dispatcher의 경우, NAT/NAPT는 메시지 데이터 부분에 주소 또는 포트 번호를 끼워 넣는 FTP와 같은 응용프로그램 프로토콜에 대해서는 작동하지 않습니다. 이것은 헤더 기반의 NAT/NAPT 제한사항으로 잘 알려진 내용입니다.

61 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 방법)』에서 자세한 정보를 참조하십시오.

- **Dispatcher content-based routing 기능(컨텐츠 규칙 및 SSL 세션 ID 연관 관계 사용)**

이 기능은 Dispatcher 구성요소에 적용됩니다.

이전 Network Dispatcher 릴리스에서 content-based routing은 Caching Proxy와 함께 CBR 구성요소를 사용할 때만 사용 가능합니다. 이제 Dispatcher 구성요소는 HTTP("content" 유형 규칙 사용) 및 HTTPS(SSL 세션 ID 연관 관계 사용)에 대해 Caching Proxy를 사용하지 않고 content-based routing을 수

행할 수 있도록 합니다. HTTP 및 HTTPS 통신량의 경우, Dispatcher 구성요소는 CBR 구성요소에 비해 신속한 content-based routing을 제공할 수 있습니다.

컨텐츠 규칙 및 SSL 세션 ID 연관 관계 사용에 대한 자세한 정보는 63 페이지의 『Dispatcher content-based routing(cbr 전달 방법)』을 참조하십시오.

- 수동 쿠키 연관 관계

이 기능은 Dispatcher 구성요소의 Content Based Routing 기능(cbr 전달 방법) 및 CBR 구성요소에 적용됩니다.

수동 쿠키 연관 관계를 사용하면 서버가 생성한 자체 식별 쿠키를 기반으로 동일한 서버로 웹 통신량을 로드 밸런싱할 수 있습니다. 219 페이지의 『수동 쿠키 연관 관계』에서 자세한 정보를 참조하십시오.

- URI 연관 관계(Caching Proxy로 로드 밸런싱)

이 기능은 Dispatcher 구성요소의 Content Based Routing 기능(cbr 전달 방법) 및 CBR 구성요소에 적용됩니다.

URI 연관 관계를 사용하면 캐시 크기를 효율적으로 증대시키는 방법으로 Caching Proxy 서버로 웹 통신량을 로드 밸런싱할 수 있습니다. 220 페이지의 『URI 연관 관계』에서 자세한 정보를 참조하십시오.

- 클러스터(또는 사이트) 고유 비율

이 기능은 모든 Network Dispatcher 구성요소에 적용됩니다.

이전 릴리스에서 로드 밸런싱 결정을 위한 중요도(활성 연결, 새 연결, 포트 및 시스템 메트릭에 부여된)는 관리 프로그램 기능에서 설정되었습니다. 이런 비율은 구성요소에 구성된 모든 클러스터에 적용되었습니다. 로드 밸런싱 중인 사이트에 관계없이 동일한 비율을 사용하여 모든 클러스터가 측정되었습니다.

이 점이 개선되어 한 클러스터(또는 사이트)마다 중요도를 설정할 수 있게 되었습니다. 152 페이지의 『상태 정보에 제공되는 중요성 비율』에서 자세한 정보를 참조하십시오.

- 서버 파티션

이 기능은 모든 Network Dispatcher 구성요소에 적용됩니다.

이제 Network Dispatcher는 하나의 실제 서버를 몇 개의 논리 서버로 파티션할 수 있는 기능을 제공합니다. 이 기능을 사용하면, 예를 들어 servlet 엔진 또

는 데이터베이스 요청이 고속으로 실행 중인지 전혀 실행되고 있지 않은지 여부를 감지할 수 있도록 시스템에서 특정 서비스를 조회할 수 있습니다. 이런 강화 기능으로 보다 세밀한 서비스 특정 워크로드를 기반으로 로드를 분배할 수 있는 기능을 제공합니다. 172 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』에서 자세한 정보를 참조하십시오.

- **HTTP 권고자 요청/응답(URL) 옵션**

이 기능은 Dispatcher 및 CBR 구성요소에 적용됩니다.

향상된 기능의 HTTP 권고자를 사용하여 서버 내의 개별 서비스 상태를 평가할 수 있습니다. HTTP 포트 아래의 각 논리 서버에 대해 서버에서 조회할 서비스에 특정한 고유 클라이언트 URL 문자열을 지정할 수 있습니다. 174 페이지의 『HTTP 권고자 요청/응답(URL) 옵션』에서 자세한 정보를 참조하십시오.

- **클러스터(또는 사이트) 특정 권고자**

이 기능은 모든 Network Dispatcher 구성요소에 적용됩니다.

Network Dispatcher를 사용하면 동일한 포트에서 실행되지만 서로 다른 클러스터(사이트)에 구성된 서로 다른 권고자를 실행할 수 있습니다. 예를 들어, 이 기능을 사용하면 한 클러스터(사이트)에 포트 80의 HTTP 권고자를 사용하고, 다른 클러스터(사이트)에 포트 80의 조정 권고자를 사용할 수 있습니다. 158 페이지의 『권고자 시작 및 정지』에서 자세한 정보를 참조하십시오.

- **서비스 거부(DoS) 중지 감지**

이 기능은 Dispatcher 구성요소에 적용됩니다.

향상된 기능의 Dispatcher는 사용하여 서비스 거부 중지를 감지하여 경보를 통해 관리자에게 통지하는 기능을 제공합니다. Dispatcher는 서비스 거부 중지의 일반적인 특징인 눈에 떨 정도의 반개방 연결에 대한 수신 요청을 분석하여 이를 수행합니다.

222 페이지의 『서비스 거부 중지 감지』에서 자세한 정보를 참조하십시오.

- **확장 사용자 엑시트**

이 기능은 Consultant for Cisco CSS Switches와 Site Selector를 제외한 모든 구성요소에 적용됩니다.

Network Dispatcher는 사용자 정의할 수 있는 스크립트를 트리거하는 사용자 엑시트를 추가로 제공합니다. 스크립트를 작성하여 자동화된 조치(예: 고가용성

상태가 변경되었을 때 로그 또는 서버가 작동 중지되었을 때 관리자에게 경보를 수행할 수 있습니다. Network Dispatcher는 다음 새 샘플 스크립트 파일을 제공합니다.

- serverDown, serverUp, managerAlert 및 managerClear - (자세한 정보는 157 페이지의 『스크립트를 사용하여 경보나 레코드 서버 장애 생성』 참조)
 - highavailChange - (자세한 정보는 193 페이지의 『스크립트 사용』 참조)
 - halfOpenAlert - 서비스 거부(DoS) 중지 감지(자세한 정보는 222 페이지의 『서비스 거부 중지 감지』 참조)
 - halfOpenAlertDone - 서비스 거부(DoS) 중지 종료(자세한 정보는 222 페이지의 『서비스 거부 중지 감지』 참조)
- **DB2 권고자**

이 기능은 Dispatcher 구성요소에 적용됩니다.

Dispatcher는 DB2 서버와 통신하는 DB2 권고자를 제공합니다. DB2 권고자에 대한 자세한 정보는 161 페이지의 『권고자 목록』을 참조하십시오.

Network Dispatcher 구성요소 종류

Network Dispatcher의 다섯 개 구성요소는 Dispatcher, CBR(Content Based Routing), Mailbox Locator, Site Selector 및 Consultant for Cisco CSS Switches입니다. Network Dispatcher는 사용자의 사이트 구성에 따라 구성요소를 개별적으로 사용하거나 함께 사용할 수 있는 융통성을 제공합니다. 여기서는 이 구성요소에 대한 개요를 제공합니다.

Dispatcher 구성요소 개요

Dispatcher 구성요소는 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, 텔넷과 기타 다른 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 방법(포트에 지정됨)을 제공합니다.

- **MAC 전달 방법(mac)**. 이 전달 방법을 통해 Dispatcher는 수신 요구를 서버로 로드 밸런싱합니다. 서버는 Dispatcher 개입없이 응답을 클라이언트로 직접 리턴합니다.
- **NAT/NAPT 전달 방법(nat)**. Dispatcher의 NAT(Network Address Translation)/NAPT(Network Address Port Translation) 기능을 사용하여 백엔드 서버가 로컬로 접속된 네트워크에 위치하도록 하는 제약을 없앱니다. 원격 위치에 서버를 위치시킬 경우, GRE/WAND 캡슐화 기술을 사용하지 않고 nat 기술을 사용할 수 있습니다. nat 전달 방법을 통해 Dispatcher는 수신 요청을 서버로 로드 밸런싱합니다. 서버는 Dispatcher로 응답을 리턴합니다. 그런 다음, Dispatcher 시스템은 클라이언트로 응답을 리턴합니다.
- **Content Based Routing 전달 방법(cbr)**. Caching Proxy 없이 Dispatcher 구성요소를 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 구성요소는 CBR 구성요소에 비해 신속한 content-based routing을 제공할 수 있습니다. cbr 전달 방법을 통해 Dispatcher는 수신 요구를 서버로 로드 밸런싱합니다. 서버는 Dispatcher로 응답을 리턴합니다. 그런 다음, Dispatcher 시스템은 클라이언트로 응답을 리턴합니다.

Dispatcher 구성요소는 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 나타납니다. Dispatcher는 도메인 이름 서버와 독립적으로 기능하므로, 모든 요청은 Dispatcher 시스템의 IP 주소로 전송됩니다.

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

Dispatcher로 로컬 서버 관리

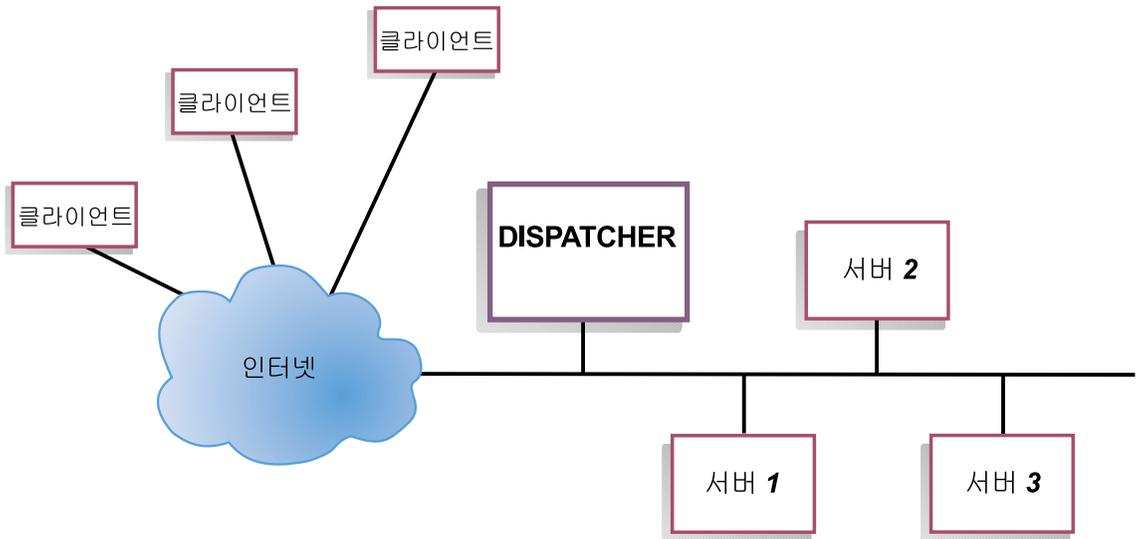


그림 6. Dispatcher를 사용하여 로컬 서버를 관리하는 사이트의 물리적 표현 예제

그림6은 인터넷 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 방법을 사용할 때 Dispatcher가 개입하지 않고 서버에서 클라이언트로 직접 전송됩니다.

Metric Server를 사용하여 서버 관리

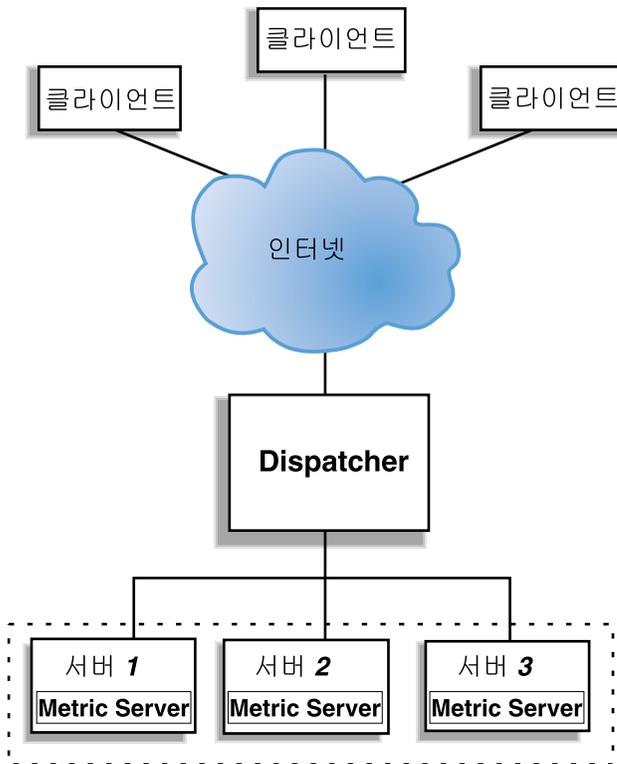


그림 7. Dispatcher 및 Metric Server를 사용하여 서버를 관리하는 사이트 예제

그림 7은 모든 서버가 로컬 네트워크에 있는 사이트를 보여줍니다. Dispatcher 구성요소는 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 Dispatcher 구성요소 또는 다른 Network Dispatcher 구성요소와 함께 사용할 수 있습니다.

Dispatcher로 로컬 및 원격 서버 관리

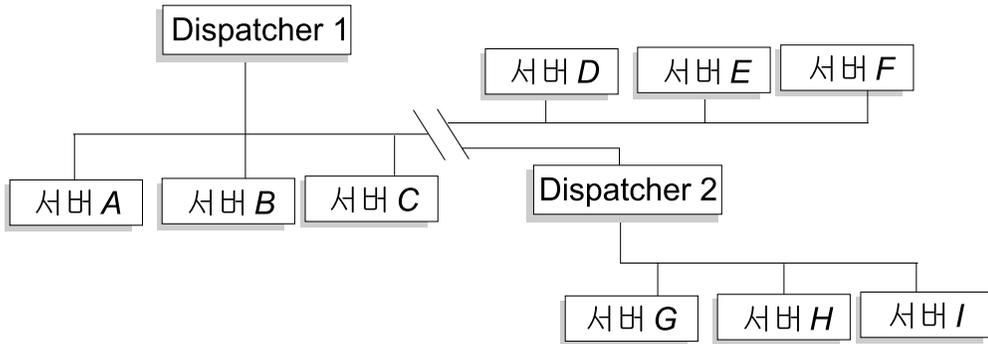


그림 8. Dispatcher를 사용하여 로컬 및 원격 서버를 관리하는 사이트 예제

Dispatcher에서 광역 지원을 사용하여 로컬 및 원격 서버(다른 서브넷에 있는 서버)를 모두 사용할 수 있습니다. 그림8에는 하나의 "로컬" Dispatcher(Dispatcher 1)가 모든 요청의 진입점 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 방법을 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 61 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 방법)』 및 185 페이지의 『GRE(일반 경로 지정 캡슐화) 지원』을 참조하십시오.

CBR(Content Based Routing) 구성요소 개요

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

CBR을 사용하면 요청 내용과 일치하는 일반 표현에 기초하여 요청을 처리하는 서버 집합을 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 집합의 한 서버가 고장난 시기를 감지하여 해당 서버로의 요청 경로지정을 정지합니다. CBR 구성요소가 사용하는 로드 밸런스 알고리즘은 Dispatcher 구성요소가 사용하는 입증된 알고리즘과 동일합니다.

Caching Proxy가 요청을 수신하면 CBR 구성요소에 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, 서브에이전트, 광역을 제외하고는 몇 가지 다른 구성 명령이 있는 Dispatcher와 동일한 기능이 있습니다.

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

CBR로 로컬 서버 관리

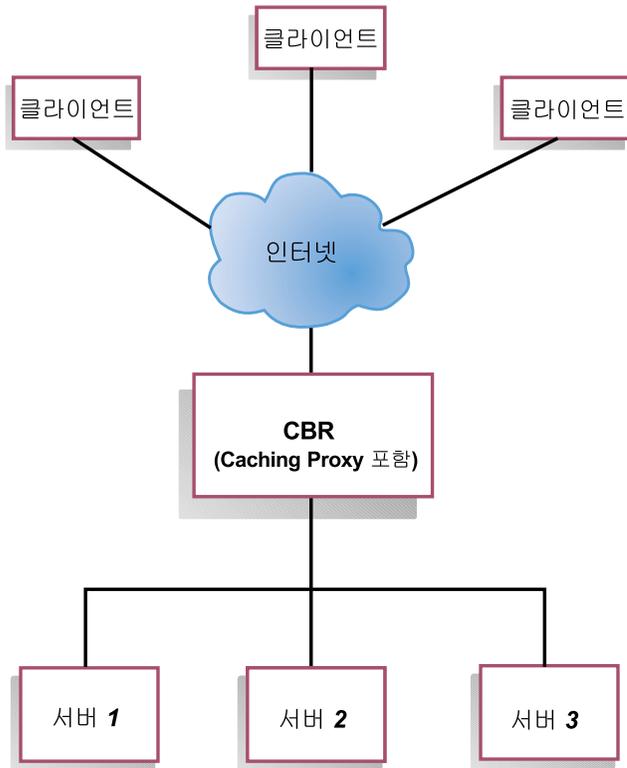


그림 9. CBR을 사용하여 로컬 서버를 관리하는 예제

그림9에서는 CBR을 사용하여 로컬 서버에서 일부 내용을 위임하는 사이트의 논리적 표현을 보여 줍니다. CBR 구성요소는 Caching Proxy를 사용하여 URL 내용을 기반으로 서버로 클라이언트 요청(HTTP 또는 HTTPS)을 전달합니다.

Mailbox Locator 구성요소 개요

Mailbox Locator는 다수의 IMAP 또는 POP3 서버에 대해 단일 연결 지점을 제공할 수 있습니다. 각 서버는 연결 지점에서 처리하는 모든 사용자 우편함의 서브셋을 가질 수 있습니다. IMAP 및 POP3 통신량의 경우 Mailbox Locator는 클라이언트가 제공하는 사용자 ID 및 암호를 기반으로 적절한 서버를 선택하는 프로세스입니다. Mailbox Locator는 규칙 기반 로드 밸런스를 지원하지 않습니다.

주: Mailbox Locator 구성요소는 이전에는 IMAP 및 POP3 메일 서버 간에 로드 밸런스된 CBR 구성요소 내의 기능이었습니다. CBR을 두 개의 구성요소로 분리하면 "IMAP/POP3용 CBR"(Mailbox Locator)과 "HTTP/HTTPS용 CBR"(Caching Proxy를 사용하는 CBR)을 동일한 시스템에서 실행할 수 없도록 하는 제약이 제거됩니다.

Mailbox Locator를 사용하여 로컬 서버 관리

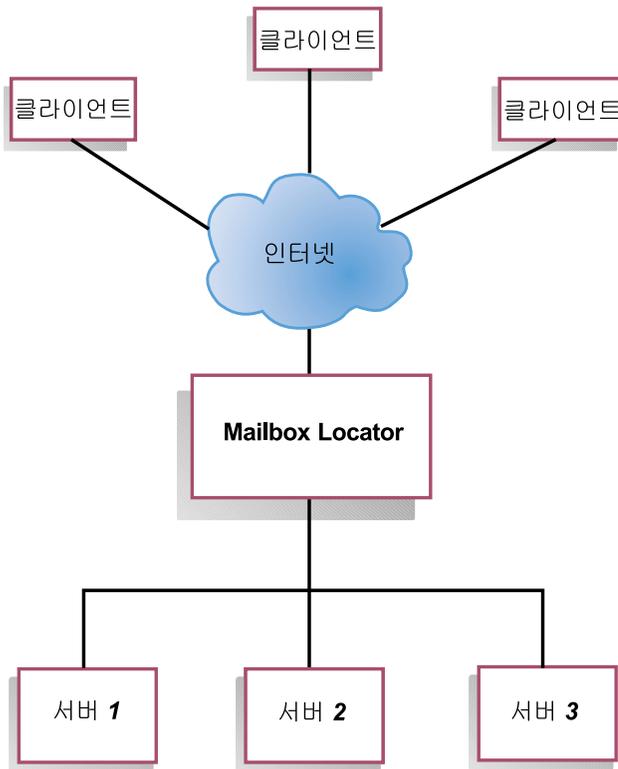


그림 10. Mailbox Locator를 사용하여 로컬 서버를 관리하는 사이트 예제

그림 10에는 사용자 ID 및 암호를 기반으로 클라이언트 요청(IMAP 또는 POP3 프로토콜)을 적절한 서버에 위임하는 데 사용되는 Mailbox Locator의 사이트가 논리적으로 표현되어 있습니다.

Site Selector 구성요소 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. 사이트 구성을 작성하여 클라이언트의 요청에 사용된 도메인 이름을 기반으로 서버 그룹 간 통신량을 로드 밸런스할 수 있습니다.

클라이언트는 도메인 이름 분석 요청을 네트워크 내의 이름 서버에 제출합니다. 이름 서버는 요청을 Site Selector 시스템으로 전달합니다. Site Selector는 도메인 이름을 사이트 이름 아래에 구성된 서버 중 하나의 IP 주소로 분석합니다. Site Selector는 선택된 서버의 IP 주소를 이름 서버에 리턴합니다. 이름 서버는 IP 주소를 클라이언트에 리턴합니다.

Metric Server는 구성 내의 각 로드 밸런스 서버에 설치해야 하는 Network Dispatcher의 시스템 모니터링 구성요소입니다. 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. 액세스 빈도, 총 사용자 수 및 액세스 유형(예: 축약형 조회, 장기 실행 조회 또는 CPU 집약 로드) 등의 요소를 고려하여 사용자 환경에 맞게 Site Selector를 구성할 수 있습니다.

Site Selector 및 Metric Server를 사용하여 로컬 및 원격 서버 관리

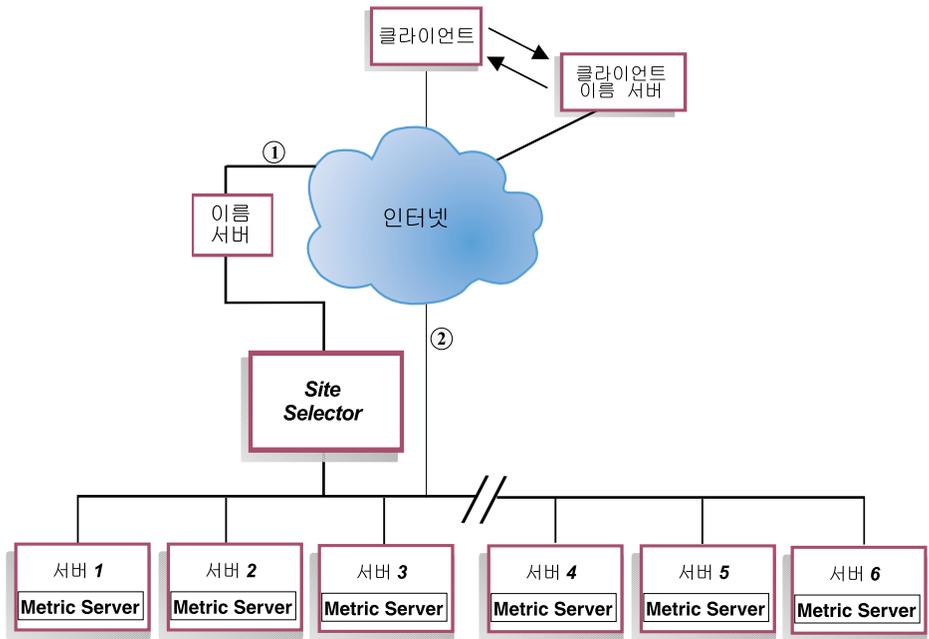


그림 11. Site Selector 및 Metric Server를 사용하여 로컬 및 원격 서버를 관리하는 사이트 예제

그림 11에는 Site Selector 구성요소를 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 후속 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 Selector와 함께 Metric Server를 구성해야 합니다. 그렇지 않을 경우 Site Selector는 로드 밸런스를 위해 라운드 로빈 선택 방법만 사용할 수 있습니다.

Consultant for Cisco CSS Switches 구성요소 개요

Consultant for Cisco CSS Switches는 Cisco의 CSS 11000 Switch 시리즈와 함께 추가 보완 솔루션을 구성합니다. 결합된 솔루션은 CSS 11000 시리즈의 견고한 패킷 전달 방법과 내용 경로 지정 기능을 Network Dispatcher의 정교한 인식 알고리즘과 혼합하여 백엔드 서버, 응용프로그램, 데이터베이스 가용성 및 로드 정보를 판별합니다. Cisco Consultant 기능은 Network Dispatcher의 관리 프로그램, 표준 및 조정 권고자, Metric Server를 활용하여 백엔드 서버, 응용프로그램 및 데이터베이스의 로드, 상태, 메트릭을 판별합니다. 이 정보를 사용하여 Cisco Consultant는 서버 기중 메트릭을 생성하여 최적 서버 선택, 로드 최적화 및 결합 허용을 위해 Cisco CSS Switch로 전송합니다.

Cisco CSS Switch는 사용자 지정 기준을 토대로 로드 밸런스를 결정합니다.

Cisco Consultant는 다음을 포함하여 다수의 기준을 추적합니다.

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

Cisco Consultant를 사용하지 않을 경우, Cisco CSS Switch는 콘텐츠 제공 서버 상태를 판별할 때 콘텐츠 요청 또는 다른 네트워크 측정에 응답 시간을 사용합니다. Cisco Consultant를 사용하면 이런 활동이 Cisco CSS Switch에서 Cisco Consultant로 오프로드됩니다. Cisco Consultant는 콘텐츠 제공 기능 또는 서버 가중치에 영향을 주며, 서버가 가용성을 다시 확보하거나 유실할 때 적절히 서버를 활성화시키거나 일시중단합니다.

Cisco Consultant:

- 공개된 SNMP 인터페이스를 사용하여 Cisco CSS Switch에서 연결 정보 확보
- 권고자 입력을 사용하여 연결 정보 분석
- Metric Server 정보를 사용하여 상대 서버 상태 분석
- 구성에 있어 각 서버에 가중치 생성

가중치는 포트의 모든 서버에 적용됩니다. 특정 포트의 경우, 요청은 각각 상대적인 가중치에 따라 서버 간에 분배됩니다. 예를 들어, 하나의 서버가 가중치 10으로 설정되고 다른 서버가 가중치 5로 설정되면, 10으로 설정된 서버의 요청 수는 5로 설정된 서버 요청 수의 두배가 됩니다. 이런 가중치는 SNMP를 사용하여 Cisco CSS Switch에 제공됩니다. 서버 가중치를 높게 설정할수록 Cisco CSS Switch가 해당 서버로 더 많은 요청을 전달합니다.

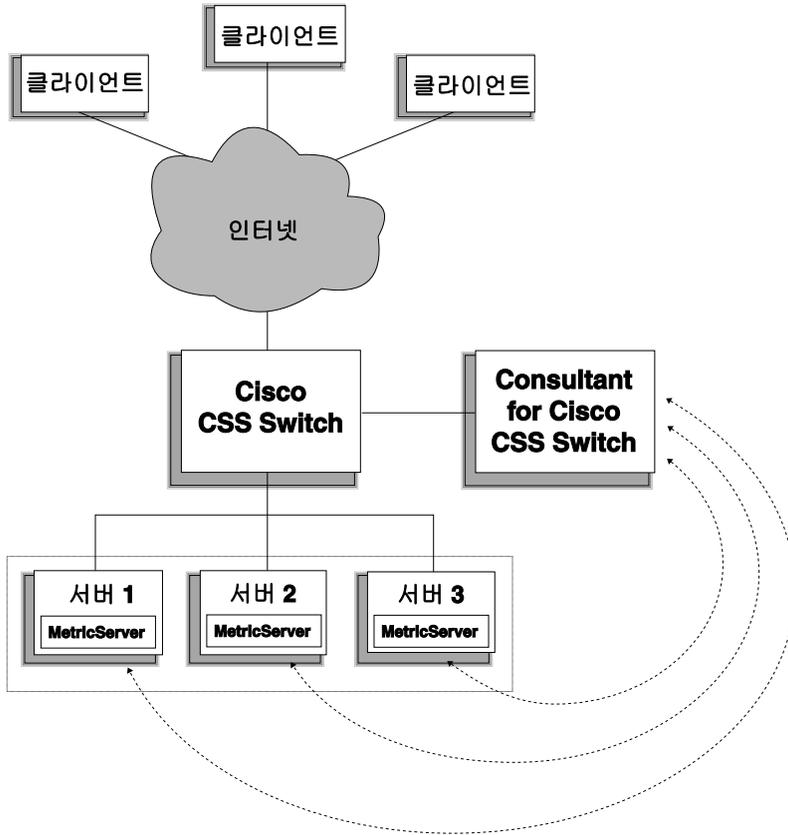


그림 12. Cisco Consultant 및 Metric Server를 사용하여 로컬 서버를 관리하는 사이트 예제

Cisco CSS Switch와 함께 Cisco Consultant는 정교한 응용프로그램 인식으로 우선 속도의 내용 전환, 결합 허용 및 서버 로드 최적화를 결합하는 "최적의" 솔루션을 제공합니다. Cisco Consultant는 Cisco CSS Switch와 IBM WebSphere Edge Server 간의 전반적인 보완 솔루션입니다.

Cisco Consultant 요구사항 목록은 13 페이지의 『제2장 Network Dispatcher 설치』를 참조하십시오.

고가용성 정보

Dispatcher

Dispatcher 구성요소는 내장형 고가용성 기능을 제공합니다. 이 기능은 기본 시스템을 모니터링하는 보조 Dispatcher 시스템의 사용을 포함하며, 기본 시스템이 실패할 경우 언제든지 로드 밸런스 태스크를 넘겨 받도록 대기합니다. Dispatcher 구성요소는 두 시스템이 모두 상호 기본 또는 보조(백업) 시스템이 되도록 허용하는 상호 고가용성 기능도 제공합니다. 188 페이지의 『고가용성 구성』을 참조하십시오.

CBR, Mailbox Locator, Site Selector

상하단부 구성을 CBR, Mailbox Locator 또는 Site Selector가 있는 둘 이상의 서버 시스템으로 통신량을 로드 밸런스하는 Dispatcher 시스템과 함께 사용할 때 Network Dispatcher의 이 구성요소들에 고가용성 레벨을 달성할 수 있습니다.

제4장 Dispatcher 구성요소 계획

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

- Dispatcher의 로드 밸런스 매개변수 구성에 대한 정보는 67 페이지의 『제5장 Dispatcher 구성요소 구성』을 참조하십시오.
- 더 많은 고급 기능을 위한 Network Dispatcher 설치 방법에 대한 정보는 149 페이지의 『제14장 고급 Network Dispatcher 기능』을 참조하십시오.
- 원격 인증 관리, Network Dispatcher 로그 및 Network Dispatcher 구성요소 사용법에 대한 정보는 229 페이지의 『제15장 Network Dispatcher 작동 및 관리』를 참조하십시오.

이 장에는 다음과 같은 섹션이 수록되어 있습니다.

- 『하드웨어 및 소프트웨어 요구사항』
- 56 페이지의 『계획 고려사항』
- 58 페이지의 『고가용성』
- 60 페이지의 『Dispatcher의 MAC 레벨 경로 지정(MAC 전달 방법)』
- 61 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 방법)』
- 63 페이지의 『Dispatcher content-based routing(cbr 전달 방법)』

하드웨어 및 소프트웨어 요구사항

플랫폼 요구사항은 다음과 같습니다.

- AIX의 경우 15 페이지의 『AIX의 요구사항』을 참조하십시오.
- Linux의 경우 20 페이지의 『Red Hat Linux 또는 SuSE Linux 요구사항』을 참조하십시오.
- Solaris의 경우 23 페이지의 『Solaris 요구사항』을 참조하십시오.
- Windows 2000의 경우 26 페이지의 『Windows 2000 요구사항』을 참조하십시오.

계획 고려사항

Dispatcher는 다음과 같은 기능으로 구성됩니다.

- **ndserver**는 명령행에서 실행 프로그램, 관리 프로그램 및 권고자에 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 구성요소가 로드 밸런스에 사용될 때 실행됩니다.
- 관리 프로그램에서는 다음에 따라 실행 프로그램에서 사용하는 가중치를 설정합니다.
 - 실행 프로그램의 내부 카운터
 - 권고자가 제공하는 서버에서의 피드백
 - Metric Server 또는 WLM과 같은 시스템 모니터링 프로그램에서의 피드백관리 프로그램을 사용하는 것은 선택입니다. 그러나 관리 프로그램을 사용하지 않으면, 현재 서버 가중치에 따라 가중된 라운드 로빈(round-robin) 스케줄링을 사용하여 로드 밸런스가 수행되며, 권고자는 사용가능하지 않습니다.
- 권고자는 관리 프로그램을 호출하여 가중치를 적절하게 설정하기 전에 서버를 조회하고 프로토콜별로 결과를 분석합니다. 현재 권고자는 HTTP, FTP, SSL, SMTP, NNTP, IMAP, POP3 및 Telnet 프로토콜에 대해 사용할 수 있습니다.

또한 Dispatcher는 DB2 서버의 상태에 대해 보고하는 DB2 권고자 및 서버가 ping 명령에 응답하는지 여부를 보고하는 Ping 권고자와 같은, 프로토콜 고유 정보를 교환하지 않는 권고자를 제공합니다. 권고자에 대한 전체 목록은 161 페이지의 『권고자 목록』을 참조하십시오.

또한 사용자 고유의 권고자를 작성하는 옵션도 있습니다(163 페이지의 『조정(사용자 정의 가능) 권고자 작성』 참조).

권고자를 사용하는 것은 선택입니다.

- 실행 프로그램, 권고자 및 관리 프로그램을 구성하고 관리하려면, 명령행 (**ndcontrol**) 또는 그래픽 사용자 인터페이스(**ndadmin**)를 사용하십시오.

- Dispatcher 시스템의 구성과 관리에 사용할 예제 구성 파일이 제공됩니다. 413 페이지의 『부록F. 예제 구성 파일』을 참조하십시오. 제품 설치 후, 이 파일은 Network Dispatcher가 있는 **nd/servers/samples** 하위 디렉토리에서 찾을 수 있습니다.
- SNMP 기본 관리 응용프로그램은 **SNMP** 서브에이전트로 Dispatcher의 상태를 모니터링할 수 있습니다.

Dispatcher의 세 가지 핵심 기능(실행 프로그램, 관리 프로그램 및 권고자)은 서버 간의 수신 요청의 밸런스를 조정하고 디스패치하기 위해 상호작용합니다. 요청을 로드 밸런스하면, 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 관리 프로그램에 제공합니다.

관리 프로그램은 실행 프로그램, 권고자 및 Metric Server와 같은 시스템 모니터링 시스템으로부터 정보를 수집합니다. 관리 프로그램에서 수신하는 정보에 따라, 어떻게 서버 시스템이 각 서버 포트에 가중되는지를 조정하고 실행 프로그램에 새로운 연결의 밸런스 조정에 사용할 새로운 가중 방법을 제공합니다.

권고자는 지정된 포트에서 각 서버를 모니터링하여 서버의 응답 시간과 사용 가능성을 판별한 다음, 이 정보를 관리 프로그램에 제공합니다. 또한 권고자는 서버의 연결 또는 단절도 모니터링합니다. 관리 프로그램과 권고자 없이, 실행 프로그램은 현재 서버의 가중치에 따라 라운드 로빈(round-robin) 스케줄링을 수행합니다.

고가용성

단순 고가용성

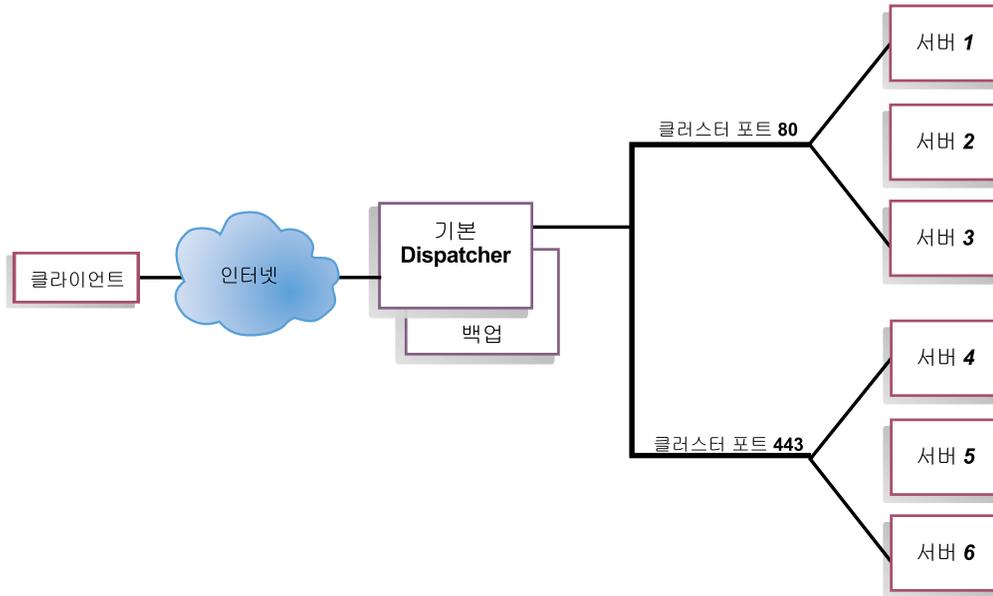


그림 13. 고가용성을 사용하는 Dispatcher 예제

고가용성 기능은 두 번째 Dispatcher 시스템의 사용을 포함합니다. 첫 번째 Dispatcher 시스템은 단일 Dispatcher 구성에 있는 것처럼 모든 클라이언트 통신에 대한 로드 밸런스를 수행합니다. 두 번째 Dispatcher 시스템은 먼저 “상태”를 모니터링하고 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 활성(로드 밸런스 중)인 동안 백업 시스템은 대기 상태에 있다가 계속 갱신되며 필요에 따라 인계를 받을 준비가 됩니다.

두 시스템 간의 통신 세션은 핵심 메시지라고 합니다. 핵심 메시지로 각 시스템은 다른 시스템의 상태를 모니터링할 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 서브넷에 있어야 합니다.

고가용성 구성에 대해서는 187 페이지의 『고가용성』을 참조하십시오.

상호 고가용성

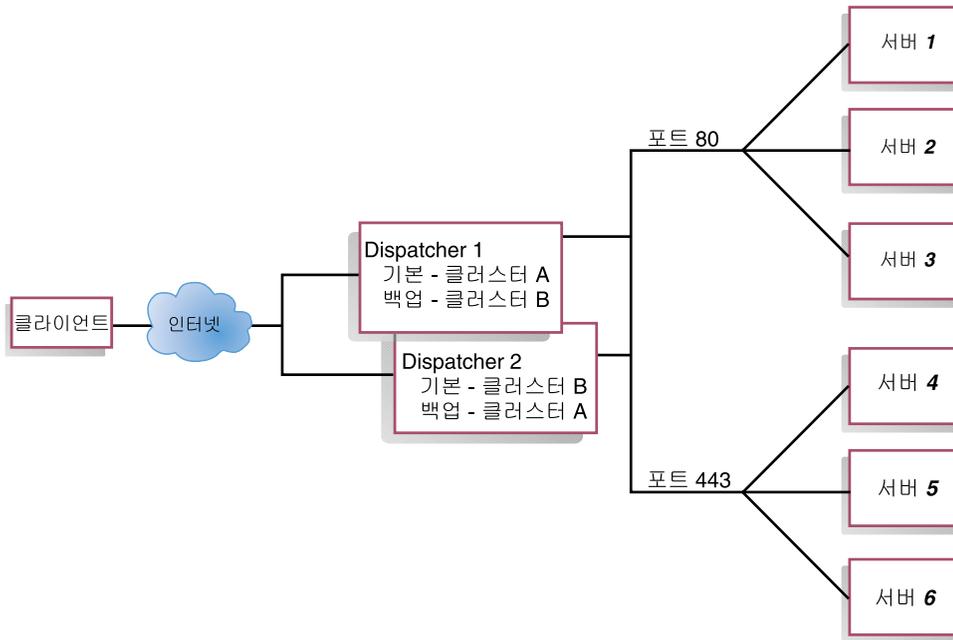


그림 14. 상호 고가용성을 사용하는 Dispatcher 예제

상호 고가용성 기능은 두 Dispatcher 시스템의 사용과 관련됩니다. 두 시스템이 클라이언트 통신에 대한 로드 밸런스를 수행하고 두 시스템이 서로에 대한 백업을 제공합니다. 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 집합 A”와 공유 “클러스터 집합 B”가 있는 상호 고가용성 구성에 대한 설명은 59 페이지의 그림14를 참조하십시오. 각 Dispatcher는 기본 클러스터의 패킷을 활성화적으로 라우트할 수 있습니다. 임의 Dispatcher가 실패했거나 더 이상 기본 클러스터의 패킷을 활성화적으로 라우트할 수 없으면 나머지 Dispatcher가 백업 클러스터의 패킷 경로 지정을 대신합니다.

주: 두 시스템은 공유된 클러스터 설정이 동일하게 구성되어야 합니다.

고가용성 및 상호 고가용성 구성에 대한 자세한 내용은 187 페이지의 『고가용성』을 참조하십시오.

Dispatcher의 MAC 레벨 경로 지정(MAC 전달 방법)

Dispatcher는 Dispatcher의 MAC 전달 방법(기본 전달 방법)을 통해 선택한 서버로 들어오는 요청을 로드 밸런싱하고, 서버는 Dispatcher 없이 클라이언트로 직접 응답을 리턴합니다. Dispatcher는 이 전달 방법을 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 방법은 `ndcontrol port add cluster:port method value` 명령을 사용하여 포트를 추가할 때 선택될 수 있습니다. 기본 전달 방법 값은 `mac`입니다. 방법 매개변수는 포트가 추가될 때만 지정할 수 있습니다. 일단 포트가 추가되면 전달 방법의 설정을 변경할 수 없습니다. 319 페이지의 『ndcontrol 포트 -- 포트 구성』에서 자세한 정보를 참조하십시오.

Dispatcher의 NAT/NAPT(nat 전달 방법)

Dispatcher의 NAT(네트워크 주소 변환) 또는 NAPT(네트워크 주소 포트 변환) 기능을 사용하면 로드 밸런스된 서버가 로컬로 연결된 네트워크에 있어야 한다는 제한사항이 제거됩니다. 서버가 원격 위치에 있을 때는 GRE/WAN 캡슐화 기술을 사용하지 않고 NAT 전달 방법 기술을 사용할 수 있습니다. 또한 각 디먼이 고유한 포트를 인식할 경우, NAPT 기능을 사용하여 로드 밸런스된 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.
- NAPT를 사용하면, 여러 서버 디먼(동일한 물리적 서버에서 실행)을 구성하여 다른 포트 번호를 인식할 수 있습니다.

이 응용프로그램은 HTTP, SSL, IMAP, POP3, NNTP, SMTP, 텔넷 등과 같은 상위 레벨 응용프로그램 프로토콜에서 잘 작동합니다.

제한사항

- Dispatcher의 NAT/NAPT 구현은 이러한 기능의 간단한 구현입니다. TCP/IP 패킷 헤더의 목차에 대해 분석하고 작동합니다. 패킷 데이터 부분의 목차는 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.

NAT/NAPT를 구현하려면 다음을 수행하십시오.

- **ndcontrol executor set** 명령에 **clientgateway** 매개변수를 설정하십시오. Clientgateway는 리턴되는 방향의 통신량을 Network Dispatcher에서 클라이언트로 전달하는 라우터 주소로 사용되는 IP 주소입니다. 이 값은 NAT/NAPT를 사용하기 전에 0이 아닌 IP 주소로 설정되어야 합니다. 293 페이지의 『ndcontrol executor -- 실행 프로그램 제어』에서 자세한 정보를 참조하십시오.

- **ndcontrol port add cluster:port method value** 명령을 사용하여 포트를 추가하십시오. 전달 방법 값은 **nat**로 설정해야 합니다. 방법 매개변수는 포트가 추가될 때만 지정할 수 있습니다. 일단 포트가 추가되면 전달 방법의 설정을 변경할 수 없습니다. 319 페이지의 『ndcontrol 포트 -- 포트 구성』에서 자세한 정보를 참조하십시오.

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

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

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

- **mapport**

Dispatcher가 클라이언트의 요청을 로드 밸런싱하는 데 사용하는 서버의 포트 번호로 클라이언트 요청의 목적지 포트 번호(Dispatcher용)를 맵핑합니다. **mapport**는 Network Dispatcher가 한 포트에서 클라이언트의 요청을 받아서 서버 시스템의 다른 포트에 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 목적지 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, 서버는 패킷을 클라이언트로 직접 전송하지 않고, Dispatcher 시스템으로 패킷을 리턴합니다 (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.

mapport, returnaddress 및 router 매개변수를 사용하는 **ndcontrol server** 명령에 대한 자세한 정보는 335 페이지의 『ndcontrol server -- 서버 구성』을 참조하십시오.

Dispatcher content-based routing(cbr 전달 방법)

이전의 Network Dispatcher 릴리스에서 content-based routing은 CBR 구성요소를 Caching Proxy와 함께 사용할 때만 사용할 수 있었습니다. 이제 Dispatcher 구성요소를 사용하면 Caching Proxy 없이 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 구성요소는 CBR 구성요소보다 content-based routing을 빨리 제공할 수 있습니다.

HTTP: Dispatcher content-based routing에 대해 HTTP 헤더나 URL의 내용을 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 방법).

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

주: 콘텐츠 규칙은 CBR 구성요소에 구성된 것과 동일한 방법으로 Dispatcher 구성요소에 구성됩니다. Dispatcher는 HTTP 통신에 콘텐츠 규칙을 사용할 수 있습니다. 그러나 CBR 구성요소는 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 조정을 제거하면, 서버는 CPU 주기를 저장하고 클라이언트는 보다 빠른 응답을 받게 됩니다. SSL 세션 ID 연관 관계를 가능하게 하기 위해 **stickytime**를 포트하십시오. 은 0이 아닌 값으로 설정해야 합니다. **stickytime**이 초과되면 클라이언트가 이전과 다른 서버로 전송될 수 있습니다.

Dispatcher content-based routing(cbr 전달 방법)을 구현하려면 다음을 수행하십시오.

- **ndcontrol executor set** 명령에 **clientgateway** 매개변수를 설정하십시오. Clientgateway는 리턴되는 방향의 통신량을 Network Dispatcher에서 클라이언트로 전달하는 라우터 주소로 사용되는 IP 주소입니다. clientgateway의 기본 값은 0입니다. content-based routing 전달 방법을 추가하기 전에 이 값을 0이 아닌 IP 주소로 설정해야 합니다. 293 페이지의 『ndcontrol executor -- 실행 프로그램 제어』에서 자세한 정보를 참조하십시오.
- **ndcontrol port add** 명령에 **method** 매개변수를 사용하여 포트를 추가하십시오. 전달 방법 값은 **cbr**로 설정해야 합니다. 319 페이지의 『ndcontrol 포트 -- 포트 구성』에서 자세한 정보를 참조하십시오.

주: 클라이언트 게이트웨이 주소를 0 이외의 값으로 설정하지 않으면, 전달 방법은 **mac** 전달 방법이어야 합니다.

- **mapport**, **returnaddress** 및 **router** 매개변수를 사용하여 서버를 추가하십시오.
ndcontrol server add cluster:port:server mapport value returnaddress rtrnaddress router rtraddress

주: **mapport**, **returnaddress** 및 **router** 매개변수를 사용한 서버 구성에 대한 정보는 62 페이지를 참조하십시오.

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

ndcontrol 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 205 페이지의 『요청 내용에 따라 규칙 사용』을 참

조하십시오. *pattern*의 유효한 표현식에 대한 자세한 정보는 347 페이지의 『부록C. 콘텐츠 규칙(패턴) 구문』을 참조하십시오.

HTTPS(SSL): SSL ID 세션 연관 관계를 구성하려면, 포트의 **stickytime** 매개변수를 0이 아닌 값으로 설정하십시오. 포트 명령에서 **stickytime**에 대한 자세한 정보는 327 페이지의 『ndcontrol rule -- 규칙 구성』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.

제5장 Dispatcher 구성요소 구성

이 장에 나와 있는 단계를 수행하기 전에 55 페이지의 『제4장 Dispatcher 구성요소 계획』을 참조하십시오. 이 장에서는 Network Dispatcher의 Dispatcher 구성요소에 대한 기본 구성을 작성하는 방법에 대해 설명합니다.

- Network Dispatcher 복합 구성에 대한 정보는 149 페이지의 『제14장 고급 Network Dispatcher 기능』을 참조하십시오.
- 원격 인증 관리, Network Dispatcher 로그 및 Network Dispatcher 구성요소 사용법에 대한 정보는 229 페이지의 『제15장 Network Dispatcher 작동 및 관리』를 참조하십시오.

구성 태스크 개요

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

표 3. Dispatcher 기능의 구성 태스크

태스크	설명	관련 정보
Dispatcher 시스템 설정.	로드 밸런스 구성을 설정합니다.	70 페이지의 『Dispatcher 시스템 설정』
로드 밸런스를 수행할 시스템 설정.	루프백 장치에 별명을 지정하고 여분의 라우트를 확인하여 여분의 라우트를 삭제합니다.	78 페이지의 『로드 밸런스에 대한 서버 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.

- 명령 프롬프트에서 **ndserver** 명령을 실행하십시오. Windows 2000의 경우 ndserver는 자동으로 시작하는 서비스로서 실행됩니다.

주: 서비스를 중지하려면 **ndserver stop** 명령을 실행하십시오.

- 그 다음, 구성 설정을 위해 원하는 Dispatcher 제어 명령을 실행하십시오. 이 책의 절차에서는 명령행을 사용한다고 가정합니다. 명령은 **ndcontrol**입니다. 명령에 대한 자세한 내용은 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』를 참조하십시오.

ndcontrol 명령 매개변수의 최소 버전을 입력할 수 있습니다. 매개변수의 고유한 문자만 입력해야 합니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면, **ndcontrol help file** 대신에 **ndcontrol he f**를 입력할 수 있습니다.

명령행 인터페이스를 시작하려면 **ndcontrol** 명령을 실행하여 ndcontrol 명령 프롬프트를 받으십시오.

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

스크립트

Dispatcher 구성 명령은 구성 파일 스크립트에 입력되어 함께 실행될 수 있습니다. 413 페이지의 『예제 Network Dispatcher 구성 파일』을 참조하십시오.

주: 스크립트 파일(예: myscript) 내용을 빨리 실행하려면 다음 명령 중 하나를 사용하십시오.

- 현재 구성을 갱신하려면 다음을 사용하여 스크립트 파일에서 실행 가능 명령을 실행하십시오.

```
ndcontrol file appendload myscript
```

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

```
ndcontrol file newload myscript
```

GUI

GUI(Graphical User Interface) 예제는 6 페이지의 그림2를 참조하십시오.

GUI를 시작하려면 다음 단계를 따르십시오.

1. ndserver가 실행 중인지 확인하십시오.

- AIX, Linux 또는 Solaris의 경우 다음 명령을 루트로 실행하십시오.

```
ndserver
```

- Windows 2000의 경우 ndserver는 자동으로 시작되는 서비스로 실행됩니다.

2. 그리고 나서, 다음 중 하나를 수행하십시오.

- AIX, Linux 또는 Solaris: **ndadmin**을 입력하십시오.
- Windows 2000: 시작, 프로그램, **IBM WebSphere, Edge Server, IBM Network Dispatcher**를 차례로 누른 후 **Network Dispatcher**를 누르십시오.

GUI에서 Dispatcher 구성요소를 구성하려면 우선 트리 구조에서 **Dispatcher**를 선택해야 합니다. 일단 호스트에 연결되면 실행 프로그램과 관리 프로그램을 시작할 수 있습니다. 또한 포트와 서버가 들어 있는 클러스터를 작성하고 관리 프로그램에 대한 권고자를 시작할 수 있습니다.

GUI를 사용하면 **ndcontrol** 명령으로 수행할 수 있는 모든 작업을 수행할 수 있습니다. 예를 들어, 명령행을 사용하여 클러스터를 정의하려면 **ndcontrol cluster add cluster** 명령을 입력하십시오. GUI에서 클러스터를 정의하려면 실행 프로그램을 마우스 오른쪽 단추로 눌러 표시되는 팝업 메뉴에서 클러스터 추가를 누르십시오. 팝업 창에 클러스터 주소를 입력한 후 확인을 누르십시오.

호스트 팝업 메뉴에 표시된 새 구성 로드 옵션(현재 구성을 완전히 바꾸기 위한 옵션) 및 현재 구성에 추가 옵션(현재 구성을 갱신하기 위한 옵션)을 사용하여 기

존 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 Network Dispatcher 구성요소에 걸쳐 기존 파일의 연결을 복원할 수 있습니다.

구성 명령은 원격으로 실행될 수도 있습니다. 자세한 내용은 229 페이지의 『원격 인증 관리』를 참조하십시오.

Network Dispatcher 창 오른쪽 상단 구석의 물음표를 누르면, 도움말이 표시됩니다.

- 필드 도움말 -- 각 필드와 기본값을 설명합니다.
- 수행 방법 -- 해당 화면에서 수행할 수 있는 태스크가 나열되어 있습니다.
- 목차 -- 모든 도움말 정보의 목차입니다.
- 색인 -- 도움말 항목의 영문자 색인입니다.

GUI 사용에 대한 자세한 내용은 7 페이지의 『GUI 사용에 대한 일반 지침』을 참조하십시오.

구성 마법사

구성 마법사 사용에 대한 자세한 내용은 5 페이지의 『구성 마법사를 사용한 구성』을 참조하십시오.

Dispatcher 시스템 설정

Dispatcher 시스템을 설정하려면 루트 사용자(AIX, Linux 또는 Solaris의 경우) 또는 Windows 2000에서는 관리자여야 합니다.

AIX, Linux 및 Solaris의 경우에만 Network Dispatcher에 결합 배치 서버가 있을 수 있습니다. 이것은 Network Dispatcher가 로드 밸런스를 유지하고 있는 서버 시스템에 실제로 상주할 수 있음을 의미할 뿐입니다.

Dispatcher 시스템에 대해 최소한 두 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. 텔넷을 통해 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 사용됩니다. 이 주소는 Dispatcher에서 로드 밸런스된 주소입니다.

Solaris 전용

1. 기본적으로, Dispatcher는 100Mbps 이더넷 네트워크 인터페이스 카드의 통신량을 로드 밸런스하도록 구성됩니다. 기본 설정을 변경하려면, **/opt/nd/servers/ibmnd.conf** 파일을 다음과 같이 편집해야 합니다.

- 기본 100Mbps 이더넷 어댑터는 **ibmnd.conf**에 **hme**로 지정되어 있습니다.
- 10Mbps 이더넷 어댑터를 사용하려면 **hme**를 **1e**로 바꾸십시오.
- 1Gbps 이더넷 어댑터를 사용하려면 **hme**를 **ge**로 바꾸십시오.
- 멀티포트 어댑터를 사용하려면 **hme**를 **qfe**로 바꾸십시오.
- 여러 유형의 어댑터를 지원하려면, **ibmnd.conf** 파일의 해당 행을 복제하여 사용자 장치 유형에 맞도록 각 행을 수정하십시오.

예를 들어, 두 개의 100Mbps 이더넷 어댑터를 사용하려면 **ibmnd.conf** 파일에 **hme** 장치를 지정하는 단일 행이 있어야 합니다. 하나의 10Mbps 이더넷 어댑터와 100Mbps 이더넷 어댑터를 사용하려면, **ibmnd.conf** 파일에 두 행이 있어야 하는데, 한 행에서는 **1e** 장치를 지정하고 다른 한 행에서는 **hme** 장치를 지정합니다.

ibmnd.conf 파일은 Solaris **autopush** 명령에 대한 입력을 제공하며 **autopush** 명령과 호환 가능해야 합니다.

2. Dispatcher 실행 프로그램을 시작하거나 정지하면 **ibmnd.conf** 파일에 나열된 어댑터의 모든 별명이 구성 해제됩니다. 이들 어댑터(Network Dispatcher의 Dispatcher 구성요소에 의해 사용되는 어댑터는 제외)에 대해 별명을 자동으

로 다시 구성하려면 **goAliases** 스크립트 파일을 사용합니다. 샘플 스크립트는 **...nd/servers/samples** 디렉토리에 있으며 실행하기 전에 **...nd/servers/bin**으로 옮겨야 합니다. goAliases 스크립트는 Dispatcher 실행 프로그램이 시작하거나 정지할 때 자동으로 실행됩니다.

예를 들어, 클러스터 X 및 Y가 **ibmnd.conf**에 나열된 어댑터의 Mailbox Locator 구성요소가 사용하도록 구성된 경우 클러스터 X 및 Y는 **ndcontrol executor start** 또는 **ndcontrol 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.

Windows 2000 전용: TCP/IP 프로토콜에 대해 IP 전달이 작동되지 않는지 확인하십시오(Windows 2000 TCP/IP 구성 참조).

그림15는 단일 클러스터, 두 개의 포트 및 세 개의 서버로 설정된 Dispatcher 예를 보여줍니다.

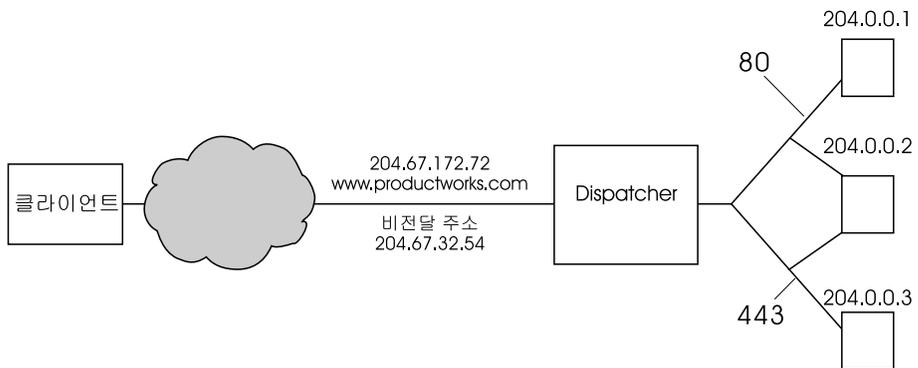


그림 15. Dispatcher 시스템에 대해 필요한 IP 주소 예제

이 절차에서 사용되는 명령에 대한 도움말을 보려면 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』를 참조하십시오.

예제 구성 파일을 보려면, 413 페이지의 『예제 Network Dispatcher 구성 파일』을 참조하십시오.

1단계. 서버 기능 시작

AIX, Linux 및 Solaris: 서버 기능을 시작하려면 **ndserver**를 입력하십시오.

Windows 2000: 서버 기능이 서비스로서 자동으로 시작됩니다.

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

2단계. 실행 프로그램 기능 시작

실행 프로그램 기능을 시작하려면, **ndcontrol executor start** 명령을 입력하십시오. 이때 여러가지 실행 프로그램 설정을 변경할 수도 있습니다. 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』를 참조하십시오.

3단계. 비전달 주소 정의(호스트 이름과 다른 경우)

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

비전달 주소를 정의하려면, **ndcontrol executor set nfa IP_address** 명령을 입력하거나 예제 구성 파일을 편집하십시오. *IP_address*는 기호 이름 또는 점분리 10진수 주소입니다.

4단계. 클러스터 정의 및 클러스터 옵션 설정

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

클러스터는 기호 이름, 점분리 10진수 주소 또는 와일드 카드 클러스터를 정의하는 특수 주소 0.0.0.0입니다. 클러스터를 정의하려면, **ndcontrol cluster add** 명령을 실행하십시오. 클러스터 옵션을 설정하려면, **ndcontrol cluster set** 명령을 실행하거나 GUI를 사용하여 명령을 실행하십시오. 와일드 카드 클러스터는 로드 밸런스를 수행할 수신 패킷에 여러 IP 주소를 대응시키는 데 사용할 수 있습니다. 자세한 정보는 209 페이지의 『와일드 카드 클러스터를 사용하여 서버 구성 조합』,

210 페이지의 『와일드 카드 클러스터를 사용하여 방화벽 로드 밸런스 수행』 및 211 페이지의 『투명 프록시의 경우 Caching Proxy가 있는 와일드 카드 클러스터 사용』을 참조하십시오.

5단계. 네트워크 인터페이스 카드의 별명 지정

일단 클러스터가 정의되었으면, 일반적으로 Dispatcher 시스템의 네트워크 인터페이스 카드 중 하나에 클러스터 주소를 구성해야 합니다. 이를 수행하려면, **ndcontrol cluster 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,

```
ndcontrol cluster configure 204.67.172.72
```

클러스터 주소를 구성하지 않는 환경은고가용성 모드의 대기 서버에 추가된 클러스터나 원격 서버의 역할을 하는 광역 Dispatcher로 추가된 클러스터를 위한 것입니다. 또한 독립 실행 모드에서 **goIdle** 스크립트를 사용할 경우 cluster configure 명령을 실행하지 않아도 됩니다. goIdle 스크립트에 대해서는 193 페이지의 『스크립트 사용』에서 자세한 정보를 참조하십시오.

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

몇 가지 예제가 다음에 나와 있습니다.

```
ndcontrol cluster configure 204.67.172.72 en0 255.255.0.0
(AIX)
ndcontrol cluster configure 204.67.172.72 eth0:1 255.255.0.0
(Linux)
ndcontrol cluster configure 204.67.172.72 le0:1 255.255.0.0
(Solaris 7)
ndcontrol cluster configure 204.67.172.72 le0 255.255.0.0
(Solaris 8)
ndcontrol cluster configure 204.67.172.72 en0 255.255.0.0
(Windows 2000)
```

Windows 2000

Windows 2000에서 클러스터 구성 명령의 두 번째 양식을 사용하려면 사용할 인터페이스 이름을 판별해야 합니다.

사용자 시스템에 이더넷 카드가 하나만 있으므로, 인터페이스 이름은 en0입니다. 마찬가지로 토큰링 카드가 하나만 있으면 인터페이스 이름은 tr0이 됩니다. 어느 유형이든 카드가 여러 개 있으면, 카드 맵핑을 판별해야 합니다. 다음 단계를 수행하십시오.

1. 명령 프롬프트에서 **regedit**를 시작하십시오.
2. **HKEY_LOCAL_MACHINE**을 클릭하고 소프트웨어, **Microsoft, Windows NT**를 차례로 클릭하고 현재 버전을 클릭합니다.
3. 그런 다음 네트워크 카드를 클릭합니다.

네트워크 인터페이스 어댑터가 네트워크 카드 아래에 나열됩니다. 인터페이스가 이더넷인지 토큰링인지 판별하려면, 하나씩 눌러 보십시오. 인터페이스 유형이 설명 컬럼에 나열됩니다. **ndconfig**에서 지정된 이름은 인터페이스 유형으로 매핑됩니다. 예를 들면, 목록에서 첫 번째 이더넷 인터페이스가 **ndconfig**에 의해 en0으로 지정되고 두 번째가 en1로 지정되며, 첫 번째 토큰링 인터페이스는 tr0으로 지정되고 두 번째는 tr1로 지정되는 방식으로 계속 진행됩니다.

주: Windows 2000 레지스트리는 **0**이 아니라 **1**부터 어댑터 번호를 시작합니다.

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

ifconfig/ndconfig를 사용하여 클러스터 별명 구성

클러스터 구성 명령은 ifconfig(또는 Windows 2000에서 ndconfig) 명령만 실행하므로 원하면 ifconfig(ndconfig) 명령을 계속 사용할 수 있습니다.

Windows 2000: 명령행을 사용하여 클러스터 변경을 구성할 수 있도록 ndconfig 명령이 Dispatcher 구성요소와 함께 제공됩니다. ndconfig 명령의 구문은 UNIX ifconfig 명령의 구문과 같습니다.

```
ndconfig en0 alias 204.67.172.72 netmask 255.255.0.0
```

주: netmask 매개변수는 필수입니다. 매개변수는 점분리 10진(255.255.0.0) 또는 16진(0xffff0000) 양식입니다.

인터페이스 이름을 결정하려면 클러스터 구성 명령의 두 번째 양식과 동일한 기법을 사용하십시오.

Solaris: 서버의 IP를 포함하지 않는 IP 주소 목록에 바인드되는 바인드 고유 서버 응용프로그램을 사용할 때는 ifconfig 대신 **arp publish** 명령을 사용하여 Network Dispatcher 시스템에 IP 주소를 동적으로 설정하십시오. 예를 들어,

```
arp -s <cluster> <Network Dispatcher MAC address> pub
```

6단계. 포트 정의 및 포트 옵션 설정

포트를 정의하려면, **ndcontrol port add cluster:port** 명령을 입력하고, 샘플 구성 파일을 편집하거나 GUI를 사용하십시오. *cluster*는 기호 이름 또는 점분리 10진수 주소입니다. *Port*는 해당 프로토콜에 사용 중인 포트 번호입니다. 이때, 여러 가지 포트 설정을 변경할 수도 있습니다. 포트에 대한 모든 서버를 정의하고 구성해야 합니다. 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 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 규칙과 서버를 구성하는 데 사용됩니다. 또한 이 기능은 여러 개의 포트에 대해 동일한 서버/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 212 페이지의 『와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정』을 참조하십시오.

주: 와일드 카드 포트는 FTP 통신량을 처리하는 데 사용될 수 없습니다.

7단계. 로드 밸런스 서버 시스템 정의

로드 밸런스 서버 시스템을 정의하려면 **ndcontrol server add cluster:port:server** 명령을 입력하고 샘플 구성 파일을 편집하거나 GUI를 사용하십시오. *Cluster* 및

*server*는 기호 이름 또는 점분리 10진수 주소입니다. *port*는 해당 프로토콜에 사용 중인 포트 번호입니다. 로드 밸런스를 수행하기 위해서는 클러스터의 포트에 둘 이상의 서버를 정의해야 합니다.

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

주: Solaris 및 Linux의 경우 바인드 고유 서버가 결합 배치되어서는 안 됩니다.

다중 주소 결합 배치: 결합 배치 구성에서 배열된 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 구성요소의 경우, 결합 배치 서버 시스템이 **ndcontrol server** 명령을 사용하여 결합 배치로 정의되어야 합니다. 결합 배치된 서버에 대한 자세한 내용은 175 페이지의 『결합 배치된 서버 사용』을 참조하십시오.

ndcontrol server 명령 구문에 대한 자세한 내용은 335 페이지의 『ndcontrol server -- 서버 구성』을 참조하십시오.

8단계. 관리 프로그램 기능 시작(선택)

관리 프로그램 기능은 로드 밸런스를 향상시킵니다. 관리 프로그램을 시작하려면, **ndcontrol manager start** 명령을 입력하고, 예제 구성 파일을 편집하거나 GUI를 사용하십시오.

9단계. 권고자 기능 시작(선택)

권고자는 요청에 응답할 수 있도록 로드 밸런스 서버 시스템의 기능에 대한 자세한 정보를 관리 프로그램에 제공합니다. 권고자는 프로토콜마다 고유합니다. 예를 들어 HTTP 권고자를 시작하려면 다음 명령을 발행하십시오.

```
cbrcontrol advisor start http port
```

권고자 목록 및 기본 포트에 대해서는 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』를 참조하십시오. 각 권고자에 대한 설명은 161 페이지의 『권고자 목록』을 참조하십시오.

10단계. 필요한 클러스터 비율 설정

권고자를 시작하면, 로드 밸런스 결정에 포함되는 권고자 정보에 부여된 중요도를 수정할 수 있습니다. 클러스터 비율을 설정하려면 **ndcontrol cluster set cluster proportions** 명령어를 실행하십시오. 자세한 내용은 152 페이지의 『상태 정보에 제공되는 중요성 비율』을 참조하십시오.

로드 밸런스에 대한 서버 시스템 설정

서버가 결합 배치 서버(Dispatcher가 로드 밸런스하는 동일한 시스템에 있음)이거나 nat 또는 cbr 전달 방법을 사용하고 있는 경우 다음 절차를 수행하지 마십시오.

mac 전달 방법을 사용할 때 Dispatcher는 IP 주소를 추가하여 루프백 어댑터를 구성할 수 있도록 하는 백엔드 서버에 대해서만 작업하기 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 방법을 사용할 때, Dispatcher 구성요소는 TCP 서버 시스템으로 패킷을 전달하기 전에 TCP/IP 패킷의 목적지 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(AIX, Linux, Solaris 또는 Windows 2000 등)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.

주: 루프백 장치의 별명을 지정하려면 패치가 필요한 **Linux** 커널 버전이 몇 개 있습니다. Linux 커널 패치가 필요한지 보려면 83 페이지의 『Linux 커널 패치 설치(루프백 인터페이스에서 arp 응답 억제)』를 참조하십시오.

Linux 커널 버전 2.2.14 이상인 경우 **ifconfig** 명령을 실행하기 전에 다음 명령을 실행하십시오.

```
echo 1 > /proc/sys/net/ipv4/conf/lo/hidden
echo 1 > /proc/sys/net/ipv4/conf/all/hidden
```

HP-UX 및 OS/2와 같이 별명을 지원하지 않는 운영 체제가 있는 서버의 경우, 클러스터 주소에 대해 루프백 장치를 설정해야 합니다.

표4에 나와 있는 대로 운영 체제의 명령을 사용하여 루프백 장치를 설정하거나 그 별명을 지정하십시오.

표4. Dispatcher의 루프백 장치(lo0)에 별명 지정 명령

AIX	ifconfig lo0 alias cluster_address netmask netmask
HP-UX	ifconfig lo0 cluster_address
Linux	ifconfig lo:1 cluster_address netmask 255.255.255.255 up
OS/2	ifconfig lo cluster_address
Solaris 7	ifconfig lo0:1 cluster_address 127.0.0.1 up
Solaris 8	ifconfig lo0:1 plumb cluster_address netmask netmask up

표 4. Dispatcher의 루프백 장치(lo0)에 별명 지정 명령 (계속)

Windows 2000	<ol style="list-style-type: none"> 1. 시작을 누르고 설정을 누른 다음 제어판을 누르십시오. 2. 이미 수행되지 않았으면, MS 루프백 어댑터 드라이버를 추가하십시오. <ol style="list-style-type: none"> a. 하드웨어 추가/제거를 두 번 누르십시오. 이렇게 하면 하드웨어 추가/제거 마법사가 실행됩니다. b. 다음을 누르고 장치 추가/문제 해결을 선택한 후 다음을 누르십시오. c. 화면이 깜박 거린 다음 하드웨어 장치 선택 창이 나타납니다. d. MS 루프백 어댑터가 목록에 있으면 이미 설치되었습니다. 취소를 눌러 종료하십시오. e. MS 루프백 어댑터가 목록에 없으면 새 장치 추가를 선택하고 다음을 누르십시오. f. 목록에서 하드웨어를 선택하려면 새 하드웨어 찾기 창에서 아니오를 누르고 다음을 누르십시오. g. 네트워크 어댑터를 선택하고 다음을 누르십시오. h. 네트워크 어댑터 선택 창에서 제조업체 목록에서 Microsoft를 선택하고 Microsoft 루프백 어댑터를 선택하십시오. i. 다음을 누르고 다시 다음을 눌러 기본 설정을 설치하십시오.(또는 디스크 있음 선택하고 CD를 넣은 다음 CD에서 설치하십시오). j. 마침을 눌러 설치를 끝내십시오. 3. 제어판에서 네트워크를 및 전화 접속 연결을 두 번 누르십시오. 4. 장치 이름 “Microsoft 루프백 어댑터”와의 연결을 선택하고 마우스 오른쪽 단추를 누르십시오. 5. 드롭 다운 목록에서 등록 정보를 선택하십시오. 6. 인터넷 프로토콜(TCP/IP)을 선택한 다음 등록 정보를 누르십시오. 7. IP 주소를 누르십시오. 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 4. Dispatcher의 루프백 장치(lo0)에 별명 지정 명령 (계속)

OS/390	<p>OS/390 시스템에서 루프백 별명 구성</p> <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 1tr1 192.168.252.12 loopback </pre> <ul style="list-style-type: none"> • 루프백에 대해 몇 가지 주소가 정의될 수 있습니다. • 기본적으로 127.0.0.1이 구성됩니다.
--------	--

2단계. 여분의 라우트 확인

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

- 다음 명령으로 Windows 2000에 여분의 라우트가 있는지 확인하십시오.

```
route print
```

- 다음 명령으로 모든 UNIX 시스템에 여분의 라우트가 있는지 확인하십시오.

```
netstat -nr
```

Windows 2000 예제:

1. **route print**를 입력하면 다음과 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. 표5에 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 5. Dispatcher에 대한 여분의 라우트 삭제 명령

HP-UX	route delete cluster_address cluster_address
Windows 2000	route delete network_address cluster_address (MS-DOS 프롬프트에서) 주: 서버를 재부트할 때마다 여분의 라우트를 삭제해야 합니다.

72 페이지의 그림15에 나와 있는 예제를 사용하여 AIX를 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, Network Dispatcher가 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.

주: **Linux** 커널 버전 2.2.12 및 2.2.13의 경우 /proc/sys/net/ipv4/conf/lo/**arp_invisible**에 "1"이 있는지 확인하십시오.

Linux 커널 버전 2.2.14 이상의 경우, /proc/sys/net/ipv4/conf/lo/**hidden** 및 /proc/sys/net/ipv4/conf/all/**hidden**에 "1"이 있는지 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, 단계1부터 단계6을 반복하십시오.

Linux 커널 패치 설치(루프백 인터페이스에서 arp 응답 억제)

Linux 서버의 경우에만 루프백 장치에 별명을 지정하려면 고유 패치(Linux 커널 버전에 근거)가 필요합니다.

패치를 사용하면 ARP 요청에서 요청한 IP 주소를 갖는 네트워크 어댑터 포트에서만 ARP 응답을 전송할 수 있습니다. 이 패치가 없으면 Linux는 루프백 별명에 대해 네트워크에서 ARP 응답을 발행합니다. 또한 패치는 IP 주소가 서로 다른 여러 네트워크 어댑터 포트가 동일한 물리적 네트워크에 있는 경우 ARP race 조건을 정정합니다.

다음 경우에는 패치를 설치해야 합니다.

- **Linux 커널 버전 2.4.x**

- 고가용성 및 결합 배치를 갖는 Dispatcher의 MAC 전달 방법을 사용하는 경우 Dispatcher 상자에 패치를 설치해야 합니다.

주: Dispatcher는 상주하는 동일한 시스템에서 Edge Server의 다른 구성요소(Caching Proxy, Mailbox Locator, CBR 등)를 지금 로드 밸런스한 경우라도 결합 배치되도록 고려될 수 있습니다.

- MAC 전달 방법으로 구성된 Dispatcher가 로드 밸런스하는 백엔드 서버에서 2.4 커널을 사용하는 경우, 백엔드 서버 시스템에 패치를 설치해야 합니다.
- 동일한 물리적 네트워크에서 시스템에 여러 네트워크 어댑터 포트가 있는 경우 시스템에 패치를 설치해야 합니다.

- **Linux 커널 버전 2.2.12 및 2.2.13**

백엔드 서버에서 2.2.12 또는 2.2.13 커널을 사용 중인 경우.

주:

1. Network Dispatcher는 2.2 커널에서 실행되지 않습니다.
2. 패치는 2.2.14 커널에 통합됩니다.
3. Linux 커널 패치는 IBM 제품을 검사하는 데 사용되었으며 IBM 검사 환경에서 적절하다고 밝혀졌습니다. 자체의 환경에서 이 코드의 유용성을 평가하고 여러분의 요구를 충족하는지 확인하십시오. 이 코드는 앞으로 출시될 Linux 기반 소스 코드 버전에는 포함되지 않을 수도 있습니다.

- **Linux 커널 버전 2.4.x**

모든 구성에 커널 패치가 필요한 것은 아닙니다. 다음 조건에서 Linux 커널 2.4.x 버전에 대해 패치를 설치해야 합니다.

- 고가용성 및 결합 배치를 갖는 Dispatcher의 MAC 전달 방법을 사용하는 경우 Dispatcher 상자에 패치를 설치해야 합니다.

주: Dispatcher는 상주하는 동일한 시스템에서 Edge Server의 다른 구성요소 (Caching Proxy, Mailbox Locator, CBR 등)를 지금 로드 밸런스한 경우라도 결합 배치되도록 고려될 수 있습니다.

- MAC 전달 방법으로 구성된 Dispatcher가 로드 밸런스되는 백엔드 서버에서 2.4 커널을 사용하는 경우 백엔드 서버에 패치를 설치해야 합니다.
- 동일한 물리적 네트워크에서 시스템에 여러 네트워크 어댑터 포트가 있는 경우 시스템에 패치를 설치해야 합니다.

<http://oss.software.ibm.com/developerworks/opensource/cvs/naslib>에서 이 패치를 다운로드할 수 있습니다.

다운로드 목록에서 CVS 트리를 선택하십시오.

패치를 적용하려면 다음을 수행하십시오.

1. <http://oss.software.ibm.com/developerworks/opensource/cvs/naslib>에서 루프백 패치를 다운로드할 수 있습니다.
2. 커널 RPM을 설치하십시오.
 - a. **arp.c.2.4.0.patch** 패치 파일을 /usr/src/linux-2.4/net/ipv4에 복사하십시오.
 - b. 다음 명령을 실행하십시오.


```
cd /usr/src/linux-2.4/net/ipv4
patch -p0 -l < arp.c.2.4.0.patch
```

주: Linux 커널 버전 2.4.0 및 2.4.2로 검사를 수행했습니다.

3. /usr/src/linux-2.4 디렉토리로 변경하십시오.
4. Makefile을 편집하여 **-arppatch**를 EXTRAVERSION 값에 추가하십시오.
5. make mrproper 명령을 발행하십시오.
6. make config 명령을 발행하고 시스템에 적절한 값을 선택하십시오. 모듈 지원을 구성했는지 확인하십시오.
7. 다음 명령을 실행하십시오.

```

make dep;make clean;make bzImage;make modules;make modules_install
cd arch/i386/boot
cat bzImage > /boot/vmlinuz-2.4.2-2-arppatch
cd /usr/src/linux-2.4
cp System.map /boot/System.map-2.4.2-2-arppatch
cd /etc

```

8. lilo.conf를 편집하고 **image=** 단락을 복사하십시오. 새 사본에서 다음과 같이 변경하십시오.
 - /boot/vmlinuz-2.4.2-2를 /boot/vmlinuz-2.4.2-2-arppatch로 변경
 - label=linux를 label=linux-arppatch로 변경
 - default=linux를 default=linux-arppatch로 변경
9. /sbin/lilo 명령을 실행하십시오.
10. 새 커널을 재부트하십시오.

Linux 커널 버전 2.2.12 및 2.2.13

Linux 커널 버전 2.2.12 및 2.2.13용 패치를 MAC 전달 방법을 사용하는 서버 상자에 설치해야 합니다. <http://www.ibm.com/developer/linux>에서 패치를 다운로드할 수 있습니다.

패치를 적용하려면 다음을 수행하십시오.

1. <http://www.ibm.com/developer/linux>에서 루프백 패치를 다운로드하십시오.
2. 커널 소스를 설치하십시오. 설치 지침을 보려면 /usr/src/linux 디렉토리의 **README.kernel-sources** 파일을 참조하십시오.
3. /usr/src 디렉토리에서 patch 명령을 발행하여 해당 패치를 적용하십시오. 예를 들어,


```
patch -p0< patchfile
```
4. 커널을 컴파일하십시오. 컴파일 명령은 /usr/src/linux-2.4/ 디렉토리의 **README** 파일을 참조하십시오.
5. 새 커널을 설치하고 **lilo** 명령을 실행하십시오. 지침을 보려면 /usr/src/linux 디렉토리의 **README** 파일을 참조하십시오.
6. 새 커널로 재부트하십시오.
7. 다음 파일이 있는지 확인하십시오. /proc/sys/net/ipv4/conf/lo/**arp_invisible**. 파일이 있고 커널이 패치되었습니다. 파일이 없고, 패치가 실패하거나 패치되지

지 않은 커널이 부트되었습니다. /usr/src/linux/README를 확인하여 모든 설치 단계가 제대로 진행되었는지 확인하십시오.

8. 다음 명령을 실행하십시오.

```
echo 1 > /proc/sys/net/ipv4/conf/lo/arp_invisible
```

이 명령은 시스템이 재부트될 때까지만 남아 있습니다. 재부트되면 이 단계와 후속 단계를 다시 수행해야 합니다.

9. 다음과 같이 넷마스크가 255.255.255.255인 루프백의 별명을 지정하십시오.

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

10. 클러스터에 서버를 추가하십시오.

제6장 Content Based Routing 구성요소 계획

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

- CBR 로드 밸런스 매개변수 구성에 대한 정보는 95 페이지의 『제7장 Content Based Routing 구성요소 구성』을 참조하십시오.
- 더 많은 고급 기능을 위한 Network Dispatcher 설치 방법에 대한 정보는 149 페이지의 『제14장 고급 Network Dispatcher 기능』에서 참조하십시오.
- 원격 인증 관리, Network Dispatcher 로그 및 Network Dispatcher 구성요소 사용법에 대한 정보는 229 페이지의 『제15장 Network Dispatcher 작동 및 관리』를 참조하십시오.

이 장에는 다음과 같은 섹션이 수록되어 있습니다.

- 『하드웨어 및 소프트웨어 요구사항』
- 『계획 고려사항』

하드웨어 및 소프트웨어 요구사항

플랫폼 요구사항은 다음과 같습니다.

- AIX의 경우 15 페이지의 『AIX의 요구사항』을 참조하십시오.
- Linux의 경우 20 페이지의 『Red Hat Linux 또는 SuSE Linux 요구사항』을 참조하십시오.
- Solaris의 경우 23 페이지의 『Solaris 요구사항』을 참조하십시오.
- Windows 2000의 경우 26 페이지의 『Windows 2000 요구사항』을 참조하십시오.

계획 고려사항

CBR 구성요소를 사용하면 Caching Proxy를 통해 HTTP와 SSL 통신을 로드 밸런싱하여 요청을 프록시할 수 있습니다.

주: CBR을 플러그인으로 실행하려면 Caching Proxy의 역방향 프록시 모드를 설치해야 합니다.

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

- **cbrserver**는 실행 프로그램, 관리 프로그램 및 권고자에 대한 명령행의 요청을 처리합니다.
- 실행 프로그램은 클라이언트 요청의 로드 밸런스를 지원합니다. CBR 구성요소를 사용하려면 실행 프로그램을 시작해야 합니다.
- 관리 프로그램은 실행 프로그램에서 사용하는 가중치를 설정합니다.
 - 실행 프로그램의 내부 카운터
 - 권고자가 제공하는 서버에서의 피드백
 - Metric Server와 같은 시스템 모니터링 프로그램 피드백

관리 프로그램을 사용하는 것은 선택입니다. 그러나 관리 프로그램을 사용하지 않으면, 현재 서버 가중치에 따라 가중된 라운드 로빈(round-robin) 스케줄링을 사용하여 로드 밸런스가 수행되며, 권고자는 사용가능하지 않습니다.

- 권고자는 관리 프로그램을 호출하여 가중치를 적절히 설정하기 전에 서버를 조회하고 프로토콜별로 결과를 분석합니다. 일반 구성에서 이들 권고자를 사용하는 것은 바람직하지 않습니다. 또한 사용자 고유의 권고자를 작성하는 옵션도 있습니다. 권고자를 사용하는 것은 선택입니다. Network Dispatcher에서는 Caching Proxy(ibmproxy) 권고자를 제공합니다. 157 페이지의 『권고자』에서 자세한 정보를 참조하십시오.
- 실행 프로그램, 권고자 및 관리 프로그램을 구성하고 관리하려면, 명령행 (**cbrcontrol**) 또는 그래픽 사용자 인터페이스(**ndadmin**)를 사용하십시오.

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

CBR 구성요소는 요청 내용과 일치하는 일반 표현식에 기초하여 요청을 처리하는 서버 집합을 지정할 수 있게 합니다. CBR 구성요소를 사용하면, 서로 다른 내용

이 서로 다른 서버 집합에 의해 처리되도록 사용자 사이트의 파티션을 나눌 수 있습니다. 사이트에 액세스하는 클라이언트는 이러한 파티션 구분 과정을 볼 수 있습니다. 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 내용에 지정되도록 하여 워크스테이션이나 서버가 고장날 경우에 대비할 수 있습니다. CBR은 장애를 인식하고, 서버 집합의 다른 서버로 계속 클라이언트 요청을 전달합니다.

사이트를 나누는 한 가지 방법은 일부 서버는 cgi 요청만을 처리하고 다른 서버 집합은 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 집합에 지정하고 다른 모든 요청을 두 번째 서버 집합에 지정할 수 있습니다. 그러면 등록을 확약한 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.

Caching Proxy는 플러그인 인터페이스를 통해 CBR과 통신합니다. Caching Proxy를 동일한 시스템에 설치해야 합니다. 이제 동일한 시스템에서 실행되는 Caching Proxy의 여러 인스턴스가 CBR과 동시에 통신할 수 있습니다. 이전 릴리스에서는, Caching Proxy의 한 인스턴스만이 CBR과 통신할 수 있었습니다.

Caching Proxy와 함께 CBR은 지정된 규칙 유형을 사용하여 HTTP 요청을 검사합니다. Caching Proxy는 실행 중일 때 클라이언트 요청을 승인하고 최상의 서버에 대한 CBR 구성요소를 조회합니다. 이 조회에서 CBR은 우선순위가 설정된 규칙 집합에 요청을 대응시킵니다. 규칙이 일치하는 경우, 사전 구성된 서버 집합에

서 적절한 서버가 선택됩니다. 마지막으로, CBR은 서버가 선택되고 요청이 프록시되었다는 것을 Caching Proxy에게 알립니다.

로드 밸런스를 조정할 클러스터를 정의했으면, 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 참인 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).

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 밸런스하는 완전 보안 사이트를 유지할 수 있습니다.

프록시 대 서버 측면에서 SSL 암호화를 사용 가능하게 하려면 구성 명령문을 IBM Caching Proxy에 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
```

mapport의 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 권고자 *ssl2http* 가 제공됩니다. 이 권고자는 포트 443(클라이언트에서 수신하는 포트)에서 시작되고 해당 포트에 구성된 서버에 권고합니다. 두 개의 클러스터가 구성되고 클러스터마다 포트 443이 있고, 서버가 서로 다른 mapport로 구성되면 권고자의 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
```

제7장 Content Based Routing 구성요소 구성

이 장의 단계를 수행하기 전에, 89 페이지의 『제6장 Content Based Routing 구성요소 계획』을 참조하십시오. 이 장에서는 Network Dispatcher의 CBR 구성요소에 대한 기본 구성을 작성하는 방법에 대해 설명합니다.

- Network Dispatcher 복합 구성에 대한 정보는 149 페이지의 『제14장 고급 Network Dispatcher 기능』을 참조하십시오.
- 원격 인증 관리, Network Dispatcher 로그 및 Network Dispatcher 구성요소 사용법에 대한 정보는 229 페이지의 『제15장 Network Dispatcher 작동 및 관리』를 참조하십시오.

구성 태스크 개요

주: 이 테이블의 구성 단계를 시작하기 전에, CBR 시스템과 모든 서버 시스템이 네트워크에 연결되어 있고 유효한 IP 주소를 가지며 서로 ping할 수 있어야 합니다.

표 6. CBR 구성요소의 태스크 구성

태스크	설명	관련 정보
CBR 시스템 설정	요구사항에 대해 찾기.	101 페이지의 『CBR 시스템 설정』
로드 밸런스를 수행할 시스템 설정.	로드 밸런스 구성을 설정합니다.	106 페이지의 『7단계. 로드 밸런스 서버 시스템 정의』

구성 방법

다음은 Network Dispatcher의 CBR 구성요소에 대해 기본 구성을 작성하기 위한 4가지 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 정지시켜야 합니다. Caching Proxy 서비스가 자동이 아닌 수동으로 시작되도록 수정하는 것이 좋습니다.

- AIX, Linux 및 Solaris에서는 `ps -ef|grep ibmproxy` 명령을 사용하여 프로세스 ID를 찾은 후 `kill process_id` 명령으로 프로세스를 종료하여 Caching Proxy를 정지시킬 수 있습니다.
- Windows에서는 서비스 패널에서 Caching Proxy를 정지시킬 수 있습니다.

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.

- 루트 사용자로서, 명령 프롬프트에서 **cbrserver** 명령을 실행하십시오.

주: 서비스를 정지하려면, **cbrserver stop**을 실행하십시오.

- 그 다음, 구성을 설정하기 위해 원하는 CBR 제어 명령을 실행하십시오. 이 책의 절차에서는 명령행을 사용한다고 가정합니다. 명령은 **cbrcontrol**입니다. 명령에 대한 자세한 내용은 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』를 참조하십시오.
- Caching Proxy를 시작하십시오. 명령 프롬프트에서 **ibmproxy** 명령을 실행하십시오(Caching Proxy를 시작하기 전에 실행 프로그램을 시작해야 함).

주: Windows 2000의 경우, 서비스 패널(시작 -> 설정 -> 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.

명령행 인터페이스를 시작하려면 **cbrcontrol**을 실행하여 cbrcontrol 명령 프롬프트를 받으십시오.

명령행 인터페이스를 종료하려면 **exit** 또는 **quit**을 실행하십시오.

주:

1. Windows 2000에서 Dispatcher 구성요소의 ndserver는 자동으로 시작됩니다. CBR만을 사용하고 Dispatcher 구성요소를 사용하지 않는 경우 다음과 같이 ndserver가 자동으로 시작되지 않게 할 수 있습니다.
 - a. Windows 2000 서비스 창에서 IBM Dispatcher를 마우스 오른쪽 단추로 누르십시오.
 - b. 등록 정보를 선택하십시오.
 - c. 시동 유형 필드에서 수동을 선택하십시오.
 - d. 확인을 누르고, 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 client=181.0.153.222&uri=http://10.1.203.4/nipoek/*
```

동일한 명령이 운영 체제의 프롬프트에서도 작동하려면 다음과 같이 패턴에 큰 따옴표(" ")가 있어야 합니다.

```
cbrcontrol rule add 10.1.203.4:80:cbr_prod_rule_ek type content
  pattern "client=181.0.153.222&uri=http://10.1.203.4/nipoek/*"
```

큰따옴표를 사용하지 않으면 규칙이 CBR에서 저장될 때 일부 패턴이 잘릴 수 있습니다. 큰따옴표는 `cbrcontrol>>` 명령 프롬프트를 사용할 경우에 지원되지 않습니다.

스크립트

CBR 구성 명령은 구성 스크립트 파일에 입력되어 함께 실행될 수 있습니다.

주: 스크립트 파일(예: `myscript`)의 내용을 빠르게 실행하려면 다음의 한 명령을 사용하십시오.

- 현재 구성을 갱신하려면 다음과 같이 사용자 스크립트 파일에서 실행 가능 명령을 실행하십시오.

```
cbrcontrol file appendload myscript
```

- 현재 구성을 완전히 바꾸려면 다음과 같이 사용자 스크립트 파일에서 실행 가능 명령을 실행하십시오.

```
cbrcontrol file newload myscript
```

GUI

GUI(Graphical User Interface)의 예제는 6 페이지의 그림2를 참조하십시오.

GUI를 시작하려면 다음의 단계를 따르십시오.

1. `cbrserver`가 실행 중인지 확인하십시오. 루트 사용자 또는 관리자로서, 명령 프롬프트에서 `cbrserver`를 실행하십시오.
2. 그리고 나서, 다음 중 하나를 수행하십시오.
 - AIX, Linux 또는 Solaris의 경우, `ndadmin`을 입력하십시오.

- Windows 2000의 경우, 시작, 프로그램을 누른 다음, **IBM WebSphere, Edge Server**를 누르고, **IBM Network Dispatcher**를 누른 다음, **Network Dispatcher**를 누르십시오.
3. Caching Proxy를 시작하십시오(GUI에서 Caching Proxy를 시작하기 전에 먼저 Host에 연결하여 CBR 구성요소에 대해 실행 프로그램을 실행해야 함). 다음 중 하나를 수행하십시오.
- AIX, Linux 또는 Solaris의 경우, Caching Proxy를 시작하려면, **ibmproxy**를 입력하십시오.
 - Windows 2000의 경우, Caching Proxy를 시작하려면, 서비스 패널(시작 -> 설정 -> 제어판 -> 관리 도구 -> 서비스)로 가십시오.

GUI에서 CBR 구성요소를 구성하려면 우선 트리 구조에서 **Content Based Routing**을 선택해야 합니다. 일단 호스트에 연결되면 관리 프로그램을 시작할 수 있습니다. 또한 포트와 서버가 들어 있는 클러스터를 작성하고 관리 프로그램에 대한 권고자를 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 눌러 표시되는 팝업 메뉴에서 클러스터 추가를 마우스 왼쪽 단추로 누르십시오. 팝업 창에 클러스터 주소를 입력한 후 확인을 누르십시오.

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 메뉴에서, 파일에 대한 현재 호스트 연결을 저장하거나 모든 Network Dispatcher 구성요소 전반에 있는 기존 파일의 연결을 복원할 수 있습니다.

Network Dispatcher 창 오른쪽 상단 구석의 물음표를 누르면, 도움말이 표시됩니다.

- 필드 도움말 -- 각 필드와 기본값을 설명합니다.
- 수행 방법 -- 해당 화면에서 수행할 수 있는 태스크가 나열되어 있습니다.
- 목차 -- 모든 도움말 정보의 목차입니다.
- 색인 -- 도움말 항목의 영문자 색인입니다.

GUI 사용에 대한 자세한 내용은 7 페이지의 『GUI 사용에 대한 일반 지침』을 참조하십시오.

구성 마법사

구성 마법사를 사용 중이면, 다음 단계를 수행하십시오.

1. **cbrserver**를 시작하십시오. 루트 사용자나 관리자로 명령 프롬프트에서 **cbrserver**를 실행하십시오.
2. CBR의 마법사 기능을 시작하십시오.
cbrwizard를 실행하여 명령 프롬프트에서 마법사를 실행하십시오. 또는 GUI에 표시된 것처럼 CBR 구성요소 메뉴에서 구성 마법사를 선택하십시오.
3. Caching Proxy를 시작하여 HTTP 또는 HTTPS(SSL) 통신의 로드 밸런스를 수행하십시오.

AIX, Linux 또는 Solaris의 경우, Caching Proxy를 시작하려면, **ibmproxy**를 입력하십시오.

Windows 2000의 경우, Caching Proxy를 시작하려면, 서비스 패널(시작 -> 설정 -> 제어판 -> 관리 도구 -> 서비스)로 가십시오.

CBR 마법사는 CBR 구성요소의 기본 구성 작성 프로세스를 단계별로 안내합니다. 또한 네트워크에 대한 질문을 요청하고 CBR에서 서버 그룹 간의 통신량 로드 밸런스를 유지할 수 있도록 클러스터를 설치하는 단계를 안내합니다.

CBR 구성 마법사를 사용하면, 다음과 같은 패널이 표시됩니다.

- 마법사 소개
- 예상되는 작업
- 시작하기 전에
- 구성할 호스트 선택(필요한 경우)

- 클러스터 정의
- 포트 추가
- 서버 추가
- 규칙 추가
- 권고자 시작

CBR 시스템 설정

CBR 시스템을 설정하려면, 사용자는 루트 사용자(AIX, Linux 또는 Solaris의 경우) 또는 Windows 2000 관리자여야 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 경우만 해당: CBR 구성요소를 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_semume=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 정지시켜야 합니다. Caching Proxy 서비스가 자동이 아닌 수동으로 시작되도록 수정하는 것이 좋습니다.

- AIX, Linux 및 Solaris에서는 `ps -ef|grep ibmproxy` 명령을 사용하여 프로세스 ID를 찾은 후 `kill process_id` 명령으로 프로세스를 종료하여 Caching Proxy를 정지시킬 수 있습니다.
- Windows에서는 서비스 패널에서 Caching Proxy를 정지시킬 수 있습니다.

Caching Proxy 구성 파일(`ibmproxy.conf`)에서 다음과 같이 수정해야 합니다.

수신 URL 지정문 **CacheByIncomingUrl**을 변경하여 "on"으로 지정하십시오.

CBR 플러그인에 대해 편집할 4가지 항목은 다음과 같습니다.

- ServerInit
- PreExit
- 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.

AIX, Linux, Solaris 및 Windows 2000의 구성 파일에 대한 특정 추가사항은 다음과 같습니다.

그림 16. AIX용 CBR 구성 파일

```
ServerInit /usr/lpp/nd/servers/lib/libndcbr.so:ndServerInit
PreExit /usr/lpp/nd/servers/lib/libndcbr.so:ndPreExit
PostExit /usr/lpp/nd/servers/lib/libndcbr.so:ndPostExit
ServerTerm /usr/lpp/nd/servers/lib/libndcbr.so:ndServerTerm
```

그림 17. Linux용 CBR 구성 파일

```
ServerInit /opt/nd/servers/lib/libndcbr.so:ndServerInit
PreExit /opt/nd/servers/lib/libndcbr.so:ndPreExit
PostExit /opt/nd/servers/lib/libndcbr.so:ndPostExit
ServerTerm /opt/nd/servers/lib/libndcbr.so:ndServerTerm
```

그림 18. Solaris용 CBR 구성 파일

```
ServerInit /opt/nd/servers/lib/libndcbr.so:ndServerInit
PreExit /opt/nd/servers/lib/libndcbr.so:ndPreExit
PostExit /opt/nd/servers/lib/libndcbr.so:ndPostExit
ServerTerm /opt/nd/servers/lib/libndcbr.so:ndServerTerm
```

그림 19. Windows 2000용 CBR 구성 파일

일반 설치 디렉토리 경로

```
ServerInit c:\Progra~1\IBM\edge\nd\servers\lib\libndcbr.dll:ndServerInit
PreExit c:\Progra~1\IBM\edge\nd\servers\lib\libndcbr.dll:ndPreExit
PostExit c:\Progra~1\IBM\edge\nd\servers\lib\libndcbr.dll:ndPostExit
ServerTerm c:\Progra~1\IBM\edge\nd\servers\lib\libndcbr.dll:ndServerTerm
```

고유(Native) 설치 디렉토리 경로

```
ServerInit c:\Progra~1\IBM\nd\servers\lib\libndcbr.dll:ndServerInit
PreExit c:\Progra~1\IBM\nd\servers\lib\libndcbr.dll:ndPreExit
PostExit c:\Progra~1\IBM\nd\servers\lib\libndcbr.dll:ndPostExit
ServerTerm c:\Progra~1\IBM\nd\servers\lib\libndcbr.dll:ndServerTerm
```

2단계. 서버 기능 시작

주: Caching Proxy는 기본적으로 설치 후 자동으로 시작하는 서비스입니다. CBR 서버 기능을 시작하기 전에 Caching Proxy를 정지시켜야 합니다. Caching Proxy 서비스가 자동이 아닌 수동으로 시작되도록 수정하는 것이 좋습니다.

- AIX, Linux 및 Solaris에서는 `ps -ef|grep ibmproxy` 명령을 사용하여 프로세스 ID를 찾은 후 `kill process_id` 명령으로 프로세스를 종료하여 Caching Proxy를 정지시킬 수 있습니다.
- Windows에서는 서비스 패널에서 Caching Proxy를 정지시킬 수 있습니다.

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 여러 실행 프로그램 설정을 변경할 수도 있습니다. 293 페이지의 『ndcontrol executor -- 실행 프로그램 제어』를 참조하십시오.

4단계. 클러스터 정의 및 클러스터 옵션 설정

CBR은 해당 클러스터의 포트에 구성된 해당 서버에 대한 클러스터 주소로 전송된 요청의 밸런스를 조정합니다.

클러스터 주소는 기호 이름 또는 점분리 10진수 주소가 됩니다. 이 주소는 URL의 호스트 부분에 위치하게 됩니다.

클러스터를 정의하려면 다음 명령을 발행하십시오.

```
cbrcontrol cluster add cluster
```

클러스터 옵션을 설정하려면 다음 명령을 발행하십시오.

```
cbrcontrol cluster set cluster option value
```

자세한 내용은 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 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 Network Dispatcher 상자의 최소한 하나의 네트워크 인터페이스 카드에 추가해야 합니다. 그렇지 않은 경우에는 이 단계를 생략할 수 있습니다.

AIX, Linux 또는 Solaris의 경우: 클러스터 주소를 네트워크 인터페이스에 추가하려면 다음 `ifconfig` 명령을 사용하십시오. 표7에 나타난 운영 체제에 해당하는 명령을 사용하십시오.

표 7. NIC 별명 명령

AIX	<code>ifconfig interface_name alias cluster_address netmask netmask</code>
Linux	<code>ifconfig interface_name cluster_address netmask netmask up</code>
Solaris 7	<code>ifconfig interface_name cluster_address netmask netmask up</code>
Solaris 8	<code>ifconfig addif interface_name cluster_address netmask netmask up</code>

주: Linux 및 Solaris의 경우, `interface_name`에는 추가된 각각의 클러스터 주소에 해당하는 고유 번호가 있습니다(예: `eth0:1`, `eth0:2` 등).

Windows의 경우: 클러스터 주소를 네트워크 인터페이스에 추가하려면 다음을 수행하십시오.

1. 시작을 누르고 설정을 누른 다음 제어판을 누르십시오.
2. 네트워크 및 전화 접속 연결을 두 번 누르십시오.
3. LAN 접속을 마우스 오른쪽 단추로 누르십시오.
4. 등록 정보를 선택하십시오.
5. 인터넷 프로토콜(TCP/IP)을 선택하고 등록 정보를 누르십시오.
6. 다음 IP 주소 사용을 선택하고 고급을 누르십시오.
7. 추가를 누른 후 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 클러스터로 포트를 정의하려면 다음을 발행하십시오.

```
cbrcontrol port add cluster:port
```

포트 옵션을 설정하려면 다음을 발행하십시오.

```
cbrcontrol port set cluster:port option value
```

자세한 정보는 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 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 CBR w/Caching Proxy를 구성하는 핵심 단계입니다. 규칙은 URL 요청을 구별하고 적절한 서버 집합에 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 비교될 일반 표현식입니다. 패턴 구성법에 대한 자세한 내용은 347 페이지의 『부록C. 콘텐츠 규칙(패턴) 구문』을 참조하십시오.

Dispatcher에 정의된 일부 기타 규칙 유형은 CBR에서도 사용될 수 있습니다. 자세한 내용은 196 페이지의 『규칙 기반 로드 밸런스 구성』을 참조하십시오.

9단계. 규칙에 서버 추가

규칙이 클라이언트 요청에 의해 일치되면, 최적의 서버를 판별하기 위해 규칙의 서버 집합이 조회됩니다. 규칙의 서버 집합은 포트에 정의된 서버의 서브세트입니다. 규칙의 서버 집합에 서버를 추가하려면 다음 명령을 발행하십시오.

```
cbrcontrol rule useserver cluster:port:rule server
```

10단계. 관리 프로그램 기능 시작(선택적)

관리 프로그램 기능은 로드 밸런스를 향상시킵니다. 관리 프로그램을 시작하려면 다음 명령을 발행하십시오.

```
cbrcontrol manager start
```

11단계. 권고자 기능 시작(선택적)

권고자는 요청에 응답하는 로드 밸런스 서버 시스템의 능력에 대한 자세한 정보를 관리 프로그램에 제공합니다. 권고자는 프로토콜마다 고유합니다. 예를 들어 HTTP 권고자를 시작하려면 다음 명령을 발행하십시오.

```
cbrcontrol advisor start http port
```

12단계. 필요한 클러스터 비율 설정

권고자를 시작한 경우, 로드 밸런스 결정 시 포함되는 권고자 정보에 부여된 중요성의 비율을 수정할 수 있습니다. 클러스터 비율을 설정하려면 **cbrcontrol cluster set cluster proportions** 명령을 발행하십시오. 자세한 정보는 152 페이지의 『상태 정보에 제공되는 중요성 비율』을 참조하십시오.

13단계. Caching Proxy 시작

- AIX 플랫폼: LIBPATH 환경 변수에 추가.
/usr/lpp/nd/servers/lib
- Linux 또는 Solaris 플랫폼: LD_LIBRARY_PATH 환경 변수에 추가.
/opt/nd/servers/lib
- Windows 2000 플랫폼: PATH 환경 변수에 추가.

일반 설치 디렉토리 경로

c:\Program Files\IBM\edge\nd\servers\lib

고유(Native) 설치 디렉토리 경로

c:\Program Files\IBM\nd\servers\lib

새로운 환경에서, 명령 프롬프트에서 Caching Proxy를 시작하고 **ibmproxy**를 실행하십시오.

주: Windows 2000의 경우, 서비스 패널(시작 -> 설정 -> 제어판 -> 관리 도구 -> 서비스)에서 Caching Proxy를 시작하십시오.

CBR 구성 예제

CBR을 구성하려면 다음 단계를 따르십시오.

1. CBR을 시작하려면, **cbrserver** 명령을 발행하십시오.
2. 명령행 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 2000의 경우, 서비스 패널에서 Caching Proxy를 시작하십시오.)
5. 브라우저에서 프록시 구성을 모두 제거하십시오.
6. "c"가 위에서 구성한 클러스터인 브라우저로 `http://c/`를 로드하십시오.
 - 서버 's'가 호출됩니다.
 - `http://s/`라는 웹 페이지가 표시됩니다.

제8장 Mailbox Locator 구성요소에 대한 계획

이 장에서는 Mailbox Locator 구성요소를 설치하고 구성하기 전에 네트워크 계획자가 고려해야 할 사항에 대해 설명합니다.

주: Mailbox Locator 구성요소는 원래 사용자 ID와 암호를 기반으로 IMAP 및 POP3 메일 서버를 로드 밸런스한 CBR 구성요소 내의 기능이었습니다. CBR을 두 개의 구성요소로 분리하면 "IMAP/POP3용 CBR"(Mailbox Locator) 및 "HTTP/HTTPS용 CBR"(Caching Proxy가 있는 CBR)을 동일한 시스템에서 실행할 수 없다는 제한사항이 제거됩니다.

- Mailbox Locator의 로드 밸런스 매개변수 구성에 대한 정보는 127 페이지의 『제11장 Site Selector 구성요소 구성』을 참조하십시오.
- 더 많은 고급 기능을 위한 Network Dispatcher 설정 방법에 대한 정보는 149 페이지의 『제14장 고급 Network Dispatcher 기능』을 참조하십시오.
- 원격 인증 관리, Network Dispatcher 로그 및 Network Dispatcher 구성요소 사용법에 대한 정보는 229 페이지의 『제15장 Network Dispatcher 작동 및 관리』를 참조하십시오.

이 장에는 다음과 같은 섹션이 수록되어 있습니다.

- 『하드웨어 및 소프트웨어 요구사항』
- 110 페이지의 『계획 고려사항』

하드웨어 및 소프트웨어 요구사항

- AIX의 경우 15 페이지의 『AIX의 요구사항』을 참조하십시오.
- Linux의 경우 20 페이지의 『Red Hat Linux 또는 SuSE Linux 요구사항』을 참조하십시오.
- Solaris의 경우 23 페이지의 『Solaris 요구사항』을 참조하십시오.
- Windows 2000의 경우 26 페이지의 『Windows 2000 요구사항』을 참조하십시오.

계획 고려사항

Mailbox Locator 구성요소를 사용하면 클라이언트 요청의 사용자 ID와 암호를 기반으로 IMAP 및 POP3 통신을 프록시할 수 있습니다.

Mailbox Locator는 구성요소 구조면에서 Dispatcher와 매우 유사합니다. Mailbox Locator는 다음과 같은 기능으로 구성됩니다

- **mlserver**는 실행 프로그램, 관리 프로그램 및 권고자에 대한 명령행의 요청을 처리합니다.
- 실행 프로그램은 클라이언트 요청의 로드 밸런스를 지원합니다. 실행 프로그램은 Mailbox Locator 구성요소가 실행될 때 항상 실행됩니다.
- 관리 프로그램에서는 다음에 따라 실행 프로그램에서 사용하는 가중치를 설정합니다.
 - 실행 프로그램의 내부 카운터
 - 권고자가 제공하는 서버에서의 피드백
 - Metric Server와 같은 시스템 모니터링 프로그램 피드백

관리 프로그램을 사용하는 것은 선택입니다. 그러나 관리 프로그램을 사용하지 않으면, 현재 서버 가중치에 따라 가중된 라운드 로빈(round-robin) 스케줄링을 사용하여 로드 밸런스가 수행되며, 권고자는 사용가능하지 않습니다.

- 권고자는 가중치를 적절하게 설정하는 관리 프로그램을 호출하기 전에 서버를 조회하고 프로토콜별로 결과를 분석합니다. 일반 구성에서 이들 권고자를 사용하는 것은 바람직하지 않습니다. 또한 사용자 고유의 권고자를 작성하는 옵션도 있습니다. 권고자를 사용하는 것은 선택입니다. 157 페이지의 『권고자』에서 자세한 정보를 참조하십시오.
- 실행 프로그램, 권고자, 관리 프로그램을 구성하고 관리하려면, 명령행(**mlcontrol**) 또는 그래픽 사용자 인터페이스(**ndadmin**)를 사용하십시오.

Mailbox Locator의 핵심적인 세 기능(실행 프로그램, 관리 프로그램, 권고자)은 서버 간의 수신 요청을 밸런스하고 디스패치하기 위해 상호작용합니다. 로드 밸런스 요청과 함께 실행 프로그램은 새로운 연결 및 활성 연결을 모니터링하고, 이러한 정보를 관리 프로그램에 제공합니다.

Mailbox Locator를 시작하려면 명령 프롬프트에서 **mlserver** 명령을 실행하십시오.

Mailbox Locator는 여러 개의 IMAP 또는 POP3 서버에 단일 연결 지점을 제공할 수 있습니다. 각 서버는 연결 지점에서 처리하는 모든 우편함의 서브세트를 가질 수 있습니다. IMAP 및 POP3의 경우, Mailbox Locator는 클라이언트가 제공하는 사용자 ID와 암호를 기반으로 적절한 서버를 선택하는 프록시입니다.

주: Mailbox Locator는 규칙 기반 로드 밸런스를 지원하지 않습니다.

다음은 클라이언트 사용자 ID를 기반으로 요청을 분배하는 방법에 대한 예제입니다. POP3 서버가 두 개(또는 그 이상)이면 우편함을 사용자 ID별로 영문자순으로 나눌 수 있습니다. 사용자 ID가 A-I 문자로 시작되는 클라이언트 요청은 서버 1에 분배할 수 있습니다. 사용자 ID가 문자 J-R로 시작되는 클라이언트 요청은 서버 2에 분배할 수 있습니다.

또한 둘 이상의 서버에 각 우편함이 표시되도록 선택할 수도 있습니다. 이 경우 각 우편함의 내용을 해당 우편함을 가지는 모든 서버에서 사용할 수 있게 됩니다. 서버 장애가 발생해도 다른 서버에서는 여전히 해당 우편함을 액세스할 수 있습니다.

여러 개의 POP3 메일 서버를 단 하나의 주소로 표시하기 위해 Mailbox Locator를 모든 클라이언트의 POP3 메일 서버 주소가 될 단일 클러스터 주소로 구성할 수 있습니다. 이를 구성하기 위한 명령은 다음과 같습니다.

```
mlcontrol cluster add pop3MailServer
mlcontrol port add pop3MailServer:110 protocol pop3
mlcontrol server add pop3MailServer:110:pop3Server1+pop3Server2+pop3Server3
```

이 예에서 *pop3MailServer*는 클러스터 주소를 나타냅니다. 프록시 프로토콜 POP3를 가지는 포트 110이 *pop3MailServer*에 추가됩니다. *Pop3Server1*, *pop3Server2* 및 *pop3Server3*은 포트에 추가되는 POP3 메일 서버를 나타냅니다. 이 구성을 사용할 경우 *pop3MailServer* 클러스터 주소를 사용하여 메일 클라이언트의 수신 POP3 요청을 구성할 수 있습니다.

연관 관계 기능 사용

POP3 또는 IMAP 요청이 프록시에 도착하면 프록시는 클라이언트의 사용자 ID와 암호를 사용하여 포트에 구성된 모든 서버에 접속합니다. 클라이언트의 요청은 응답하는 첫 번째 서버로 보내집니다. 결합/연관 관계 기능은 Mailbox Locator와 함께 IMAP 또는 POP3 서버에 사용해야 합니다. 연관 관계 기능을 사용하면 동일한 클라이언트 사용자 ID의 후속 요청을 동일한 서버로 보낼 수 있습니다. 연관 관계 기능을 사용 가능하게 하려면 포트에 대한 **stickytime**을 0보다 큰 값으로 설정하십시오. 연관 관계 기능에 대한 자세한 내용은 212 페이지의 『Network Dispatcher에 대한 친근성 기능 사용법』을 참조하십시오.

POP3/IMAP 활동 해제 타이머 덮어쓰기

POP3 및 IMAP 프로토콜에 대한 활동 해제 자동 로그아웃 타이머는 각각 최소 값 10분과 30분으로 설정되어 있습니다. 이 제한 시간은 연결이 제거되기 전에 활동 해제 상태로 있을 수 있는 초 단위의 시간을 나타냅니다. 성능을 최적화하기 위해 Mailbox Locator는 활동 해제 시간 종료 값을 60초로 덮어씁니다. 활동 해제 제한 시간을 변경하려면 **mlcontrol port** 명령에서 **staletimeout** 값을 변경하십시오. 이 명령 구성에 대한 자세한 정보는 319 페이지의 『ndcontrol 포트 -- 포트 구성』을 참조하십시오.

제9장 Mailbox Locator 구성요소 구성

이 장에 나와 있는 단계를 수행하기 전에 109 페이지의 『제8장 Mailbox Locator 구성요소에 대한 계획』을 참조하십시오. 여기서는 Network Dispatcher의 Mailbox Locator 구성요소에 대한 기본 구성 작성 방법에 대해 설명합니다.

주: Mailbox Locator 구성요소는 IMAP 및 POP3 메일 서버에서 로드 밸런스된 CBR 구성요소 내의 기능으로서 사용자 ID 및 암호를 기반으로 합니다. CBR을 두 개의 구성요소로 분리하면 "IMAP/POP3용 CBR"(Mailbox Locator)과 "HTTP/HTTPS용 CBR"(Caching Proxy를 사용하는 CBR)을 동일한 시스템에서 실행할 수 없도록 하는 제약이 제거됩니다.

- Network Dispatcher 복합 구성에 대한 정보는 149 페이지의 『제14장 고급 Network Dispatcher 기능』을 참조하십시오.
- 원격 인증 관리, Network Dispatcher 로그 및 Network Dispatcher 구성요소 사용법에 대한 정보는 229 페이지의 『제15장 Network Dispatcher 작동 및 관리』를 참조하십시오.

구성 태스크 개요

주: 이 테이블의 구성 단계를 시작하기 전에, Mailbox Locator 시스템과 모든 서버 시스템은 네트워크에 연결되어 있고 유효한 IP 주소를 가지며 서로 ping할 수 있어야 합니다.

표 8. Mailbox Locator 구성요소에 대한 구성 태스크

태스크	설명	관련 정보
Mailbox Locator 시스템 설정.	요구사항에 대해 찾기.	117 페이지의 『Mailbox Locator 시스템 설정』
로드 밸런스를 수행할 시스템 설정.	로드 밸런스 구성을 설정합니다.	119 페이지의 『4단계. 로드 밸런스 서버 시스템 정의』

구성 방법

다음은 Network Dispatcher의 Mailbox Locator 구성요소에 대해 기본 구성을 작성하기 위한 4가지 기본 방법입니다.

- 명령행
- 스크립트
- GUI(Graphical User Interface)
- 구성 마법사

명령행

이 방법은 Mailbox Locator를 구성하는 가장 직접적인 방법입니다. 명령 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름(예: 클러스터 및 서버 명령에 사용됨) 및 파일 이름은 예외입니다.

명령행에서 Mailbox Locator를 시작하려면 다음을 수행하십시오.

- 명령 프롬프트에서 **mlserver** 명령을 실행하십시오.

주: 서비스를 중지하려면 다음의 **mlserver stop**을 실행하십시오.

- 그 다음, 구성을 설정하기 위해 원하는 Mailbox Locator 제어 명령을 실행하십시오. 이 책의 절차에서는 명령행을 사용한다고 가정합니다. 명령은 **mlcontrol**입니다. 명령에 대한 자세한 내용은 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』를 참조하십시오.

mlcontrol 명령 매개변수의 최소 버전을 입력할 수 있습니다. 매개변수의 고유한 문자만 입력해야 합니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면, **mlcontrol help file** 대신에 **mlcontrol he f**를 입력할 수 있습니다.

명령행 인터페이스를 시작하려면 **mlcontrol**을 실행하여 **mlcontrol** 명령 프롬프트를 받으십시오.

명령행 인터페이스를 종료하려면 **exit** 또는 **quit**을 실행하십시오.

주: Windows 2000에서 Dispatcher 구성요소의 ndserver는 자동으로 시작됩니다. Dispatcher 구성요소가 아닌, Mailbox Locator만을 사용하는 경우, 다음과 같이 ndserver의 자동 시작을 정지시킬 수 있습니다.

1. Windows 2000 서비스 창에서 IBM Dispatcher를 마우스 오른쪽 단추로 누르십시오.
2. 등록 정보를 선택하십시오.
3. 시동 유형 필드에서 수동을 선택하십시오.
4. 확인을 누르고, 서비스 창을 닫으십시오.

스크립트

Mailbox Locator 구성 명령은 구성 스크립트 파일에 입력되어 함께 실행될 수 있습니다.

주: 스크립트 파일(예: myscript)의 내용을 빨리 실행하려면 다음의 한 명령을 사용하십시오.

- 현재 구성을 갱신하려면 다음과 같이 스크립트 파일에서 실행 가능 명령을 실행하십시오.

```
mlcontrol file appendload myscript
```

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

```
mlcontrol file newload myscript
```

GUI

GUI 예제는 6 페이지의 그림2를 참조하십시오.

GUI를 시작하려면 다음의 단계를 따르십시오.

1. mlserver가 실행 중인지 확인하십시오. 루트 사용자 또는 관리자로서, 명령 프롬프트에 **mlserver**를 실행하십시오.
2. 그리고 나서, 다음 중 하나를 수행하십시오.
 - AIX, Linux 또는 Solaris의 경우 **ndadmin**을 입력하십시오.

- Windows의 경우 시작, 프로그램을 누르고 **IBM WebSphere**를 누른 다음, **Edge Server, IBM Network Dispatcher**를 누르고, **Network Dispatcher**를 누르십시오.

GUI에서 Mailbox Locator 구성요소를 구성하려면, 우선 트리 구조에서 **Mailbox Locator**를 선택해야 합니다. 일단 호스트에 연결되면 관리 프로그램을 시작할 수 있습니다. 또한 포트와 서버가 들어 있는 클러스터를 작성하고 관리 프로그램에 대한 권고자를 시작할 수 있습니다.

GUI를 사용하면 **mlcontrol** 명령으로 할 수 있는 모든 작업을 수행할 수 있습니다. 예를 들어, 명령행을 사용하여 클러스터를 정의하려면, **mlcontrol cluster add cluster** 명령을 입력하십시오. GUI에서 클러스터를 정의하려면, 실행 프로그램을 마우스 오른쪽 단추로 눌러 표시되는 팝업 메뉴에서 클러스터 추가를 마우스 왼쪽 단추로 누르십시오. 팝업 창에 클러스터 주소를 입력한 후 확인을 누르십시오.

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 Mailbox Locator 구성 파일을 로드할 수 있습니다. 호스트 팝업 메뉴에 있는 다른 구성 파일 이름으로 저장 옵션을 사용하여 Mailbox Locator 구성을 파일에 주기적으로 저장해야 합니다. GUI의 맨 위에 있는 파일 메뉴는 파일에 대한 현재 호스트 연결을 저장하거나 모든 Network Dispatcher 구성요소 전반에 있는 기존 파일의 연결을 복원하게 합니다.

Network Dispatcher 창 오른쪽 상단 구석의 물음표를 누르면, 도움말이 표시됩니다.

- 필드 도움말 -- 각 필드와 기본값을 설명합니다.
- 수행 방법 -- 해당 화면에서 수행할 수 있는 태스크가 나열되어 있습니다.
- 목차 -- 모든 도움말 정보의 목차입니다.
- 색인 -- 도움말 항목의 영문자 색인입니다.

GUI 사용에 대한 자세한 내용은 7 페이지의 『GUI 사용에 대한 일반 지침』을 참조하십시오.

구성 마법사

구성 마법사를 사용 중이면, 다음 단계를 수행하십시오.

1. 루트 사용자나 관리자로 명령 프롬프트에서 **mlserver** 명령을 실행하십시오.
2. Mailbox Locator의 마법사 기능, **mlwizard**를 시작하십시오.

mlwizard를 실행하여 명령 프롬프트에서 마법사를 실행할 수 있습니다. 또는 GUI에 표시된 것처럼 Mailbox Locator 구성요소 메뉴에서 구성 마법사를 선택하십시오.

Mailbox Locator 마법사는 Mailbox Locator 구성요소의 기본 구성 작성 프로세스를 단계별로 안내합니다. 여기서는 네트워크에 대해 질문하고 Mailbox Locator에서 서버 그룹 간의 통신량 로드 밸런스를 유지하도록 클러스터 설치 단계를 안내합니다.

Mailbox Locator 구성 마법사를 사용하면, 다음의 패널이 표시됩니다.

- 마법사 소개
- 예상되는 작업
- 시작하기 전에
- 구성할 호스트 선택(필요한 경우)
- 클러스터 정의
- 포트 추가
- 서버 추가
- 권고자 시작

Mailbox Locator 시스템 설정

Mailbox Locator 시스템을 설정하려면 사용자는 루트 사용자(AIX, Linux 또는 Solaris) 또는 Windows 2000의 관리자여야 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. 동일한 클러스터 주소에 대한 모든 요청은 Mailbox Locator에 의해 로드 밸런스가 수행됩니다.

1단계. 서버 기능 시작

서버 기능을 시작하려면 명령행에서 **mlserver**를 입력하십시오.

주: 기본 구성 파일(default.cfg)은 mlserver를 시작할 때 자동으로 로드됩니다. 사용자가 default.cfg에 구성을 저장하면, 이 파일에 저장된 모든 내용은 다음에 mlserver가 시작될 때 자동으로 로드됩니다.

2단계. 클러스터 정의 및 클러스터 옵션 설정

Mailbox Locator는 해당 클러스터의 포트에서 구성된 해당 서버에 클러스터 주소로 전송된 요청을 조정합니다.

클러스터 주소는 기호 이름 또는 점분리 10진수 주소가 됩니다.

클러스터를 정의하려면 다음 명령을 실행하십시오.

```
mlcontrol cluster add cluster
```

클러스터 옵션을 설정하려면 다음 명령을 실행하십시오.

```
mlcontrol cluster set cluster option value
```

자세한 정보는 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』를 참조하십시오.

3단계. 포트 정의 및 포트 옵션 설정

포트 번호는 서버 응용프로그램이 인식하는 포트입니다. IMAP 통신의 경우, 일반적으로 포트 143입니다. POP3 통신의 경우, 일반적으로 포트 110입니다.

이전 단계에서 정의한 클러스터에 포트를 정의하려면 다음을 실행하십시오.

```
mlcontrol port add cluster:port protocol [pop3|imap]
```

포트 옵션을 설정하려면 다음을 실행하십시오.

```
mlcontrol port set cluster:port option value
```

주: 포트를 추가할 때, 프록시 프로토콜(pop3 또는 imap)을 지정해야 합니다. 포트를 추가한 후에는 이 포트에 대한 기존의 프로토콜 값을 변경(설정)할 수 없습니다.

자세한 정보는 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』를 참조하십시오.

4단계. 로드 밸런스 서버 시스템 정의

메일 서버는 로드 밸런스를 수행하는 응용프로그램이 실행되는 시스템입니다. *server* 는 서버 시스템의 기호 이름이거나 점분리 10진수 주소입니다. 클러스터에서 서버를 정의하고 3단계에서 포트를 정의하려면 다음 명령을 실행하십시오.

```
mlcontrol server add cluster:port:server
```

로드 밸런스를 수행하기 위해서는 클러스터의 포트별로 두 개 이상의 서버를 정의해야 합니다.

5단계. 관리 프로그램 기능 시작(선택)

관리 프로그램 기능은 로드 밸런스를 향상시킵니다. 관리 프로그램을 시작하려면 다음 명령을 실행하십시오.

```
mlcontrol manager start
```

6단계. 권고자 기능 시작(선택)

권고자는 요청에 응답하는 로드 밸런스 서버 시스템의 능력에 대한 자세한 정보를 관리 프로그램에 제공합니다. 권고자는 프로토콜마다 고유합니다. Network Dispatcher는 IMAP 및 POP3 권고자를 제공합니다. 예를 들어, IMAP 권고자를 시작하려면 다음 명령을 실행하십시오.

```
mlcontrol advisor start imap port
```

권고자 목록 및 기본 포트에 대해서는 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』를 참조하십시오. 각 권고자에 대한 설명은 161 페이지의 『권고자 목록』을 참조하십시오.

7단계. 필요한 클러스터 비율 설정

권고자를 시작할 경우, 로드 밸런스 결정 시 포함되는 권고자 정보에 부여된 중요성의 비율을 수정할 수 있습니다. 클러스터 비율을 설정하려면 **mlcontrol cluster set cluster proportions** 명령을 실행하십시오. 자세한 정보는 152 페이지의 『상태 정보에 제공되는 중요성 비율』을 참조하십시오.

제10장 Site Selector 구성요소에 대한 계획

이 장에서는 Site Selector 구성요소를 설치하고 구성하기 전에 네트워크 계획자가 고려해야 할 사항에 대해 설명합니다.

- Site Selector의 로드 밸런스 매개변수 구성에 대한 정보는 127 페이지의 『제 11장 Site Selector 구성요소 구성』을 참조하십시오.
- 더 많은 고급 기능을 위한 Network Dispatcher 설치 방법에 대한 정보는 149 페이지의 『제14장 고급 Network Dispatcher 기능』을 참조하십시오.
- 원격 인증 관리, Network Dispatcher 로그 및 Network Dispatcher 구성요소 사용법에 대한 정보는 229 페이지의 『제15장 Network Dispatcher 작동 및 관리』를 참조하십시오.

이 장에는 다음과 같은 섹션이 수록되어 있습니다.

- 『하드웨어 및 소프트웨어 요구사항』
- 122 페이지의 『계획 고려사항』

하드웨어 및 소프트웨어 요구사항

- AIX의 경우 15 페이지의 『AIX의 요구사항』을 참조하십시오.
- Linux의 경우 20 페이지의 『Red Hat Linux 또는 SuSE Linux 요구사항』을 참조하십시오.
- Solaris의 경우 23 페이지의 『Solaris 요구사항』을 참조하십시오.
- Windows 2000의 경우 26 페이지의 『Windows 2000 요구사항』을 참조하십시오.

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.

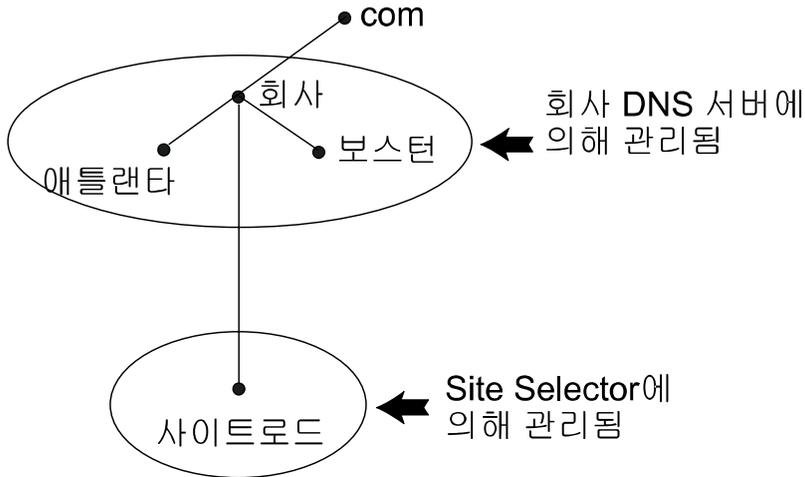


그림 20. DNS 환경의 예제

DNS 환경 내에서 Site Selector에 대한 서브도메인을 설치할 때, Site Selector는 자신의 서브도메인에 대한 권한을 가져야 합니다. 예를 들어(그림 20 참조), **company.com** 도메인에 대한 권한이 사용자 회사에 지정되었습니다. 회사 내에는 여러 개의 서브도메인이 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, AIX에서 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 노드) 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 49 페이지의 그림11을 참조하십시오.

Site Selector는 다음과 같은 기능으로 구성됩니다.

- **ssserver**는 이름 서버, 관리 프로그램 및 권고자에 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.
- 관리 프로그램은 다음에 따라 이름 서버에 사용할 가중치를 설정합니다.

- 권고자가 제공하는 서버에서의 피드백
- Metric Server와 같은 시스템 모니터링 프로그램 피드백

관리 프로그램을 사용하는 것은 선택입니다. 그러나 관리 프로그램을 사용하지 않으면, 현재 서버 가중치에 따라 가중된 라운드 로빈(round-robin) 스케줄링을 사용하여 로드 밸런싱이 수행되며, 권고자는 사용가능하지 않습니다.

- **Metric Server**는 백엔드 서버 시스템에 설치한 Network Dispatcher의 시스템 모니터링 구성요소입니다(로드 밸런싱 서버 시스템에 Network Dispatcher를 배치할 경우, Network Dispatcher 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, 관리 프로그램에서 가중치를 판별하는 데 사용하는 4가지 비율이 있습니다.

- CPU
- 메모리
- 포트
- 시스템

CPU 및 메모리 값은 모두 Metric Server에 의해 제공됩니다. 따라서 Site Selector 구성요소에 Metric Server를 사용하는 것이 권장됩니다.

169 페이지의 『Metric Server』에서 자세한 정보를 참조하십시오.

- 권고자는 관리 프로그램을 호출하여 가중치를 적절하게 설정하기 전에 서버를 조회하고 프로토콜별로 결과를 분석합니다. 일반 구성에서 이들 권고자를 사용하는 것은 바람직하지 않습니다. 또한 사용자 고유의 권고자를 작성하는 옵션도 있습니다. 권고자를 사용하는 것은 선택입니다. 157 페이지의 『권고자』에서 자세한 정보를 참조하십시오.
- 이름 서버, 권고자, Metric Server, 관리 프로그램을 구성하고 관리하려면, 명령 행(sscontrol) 또는 그래픽 사용자 인터페이스(ndadmin)를 사용하십시오.

Site Selector의 핵심적인 네 기능(이름 서버, 관리 프로그램, Metric Server, 권고자)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. 자세한 정보는 376 페이지의 『sscontrol sitename - 사이트 이름 구성』을 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(관리 프로그램 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 시간이 종료될 때까지 대기합니다. 그리고 나서, 서버마다 이름 서버는 관리 프로그램이 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 시간이 종료되기 전에 응답을 시도합니다.)

no로 설정하면, 이름 분석이 현재 관리 프로그램 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 관리 프로그램 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. 351 페이지의 『부록D. Site Selector 명령어 참조서』에서 자세한 정보를 참조하십시오.

제11장 Site Selector 구성요소 구성

이 장의 단계를 수행하기 전에, 121 페이지의 『제10장 Site Selector 구성요소에 대한 계획』을 참조하십시오. 이 장에서는 Network Dispatcher의 Site Selector 구성요소에 대한 기본 구성을 작성하는 방법을 설명합니다.

- Network Dispatcher 복합 구성에 대한 정보는 149 페이지의 『제14장 고급 Network Dispatcher 기능』을 참조하십시오.
- 원격 인증 관리, Network Dispatcher 로그 및 Network Dispatcher 구성요소 사용법에 대한 정보는 229 페이지의 『제15장 Network Dispatcher 작동 및 관리』를 참조하십시오.

구성 태스크 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 구성요소 구성 태스크

태스크	설명	관련 정보
Site Selector 시스템 설정.	요구사항에 대해 찾기.	131 페이지의 『Site Selector 시스템 설정』
로드 밸런스를 수행할 시스템 설정.	로드 밸런스 구성을 설정합니다.	132 페이지의 『4단계. 로드 밸런스 서버 시스템 정의』

구성 방법

Network Dispatcher의 Site Selector 구성요소의 기본 구성을 작성할 경우 네 가지 기본 방법으로 Site Selector 구성요소를 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.

- 명령 프롬프트에서 **ssserver** 명령을 실행하십시오.

주: 서비스를 중지하려면 **ssserver stop**을 실행하십시오.

- 그런 다음, 구성 설정을 위해 원하는 Site Selector 제어 명령을 실행하십시오. 이 책의 절차에서는 명령행을 사용한다고 가정합니다. 명령은 **sscontrol**입니다. 명령에 대한 자세한 내용은 351 페이지의 『부록D. 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.

명령행 인터페이스를 시작하려면 **sscontrol** 명령을 실행하여 sscontrol 명령 프롬프트를 받으십시오.

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

주: Windows 2000에서 Dispatcher 구성요소의 ndserver는 자동으로 시작됩니다. Dispatcher 구성요소가 아닌 Site Selector만을 사용하는 경우 다음과 같이 ndserver의 자동 시작을 정지시킬 수 있습니다.

1. Windows 2000 서비스 창에서 IBM Dispatcher를 마우스 오른쪽 단추로 누르십시오.
2. 등록 정보를 선택하십시오.
3. 시동 유형 필드에서 수동을 선택하십시오.
4. 확인을 누르고, 서비스 창을 닫으십시오.

스크립트

Site Selector 구성 명령을 구성 스크립트 파일에 입력하여 함께 실행할 수 있습니다.

주: 스크립트 파일(예: `myscript`) 내용을 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
```

GUI

GUI 예제는 6 페이지의 그림2를 참조하십시오.

GUI를 시작하려면 다음 단계를 따르십시오.

1. `ssserver`가 실행되고 있는지 확인하십시오. 루트 사용자 또는 관리자 명령 프롬프트에서 `ssserver` 명령을 실행하십시오.
2. 그런 후 다음 중 하나를 수행하십시오.
 - AIX, Linux 또는 Solaris의 경우 `ndadmin`을 입력하십시오.
 - Windows 2000의 경우 시작, 프로그램, **IBM WebSphere, Edge Server, IBM Network Dispatcher**를 차례로 누른 후 **Network Dispatcher**를 누르십시오.

GUI에서 Site Selector 구성요소를 구성하려면 먼저 트리 구조에서 **Site Selector**를 선택해야 합니다. 일단 호스트에 연결되면 관리 프로그램을 시작할 수 있습니다. 포트와 서버가 들어 있는 사이트 이름을 작성하고 관리 프로그램에 권고자를 시작할 수도 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 누른 후 표시되는 팝업 메뉴에서 사이트 이름 추가를 누르십시오. 팝업 창에 사이트 이름을 입력한 후 확인을 누르십시오.

호스트 팝업 메뉴에 표시된 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 Network Dispatcher 구성요소 전반에 있는 기존 파일의 연결을 복원하게 합니다.

Network Dispatcher 창 오른쪽 상단 구석의 물음표를 누르면, 도움말이 표시됩니다.

- 필드 도움말 -- 각 필드 및 기본값을 설명합니다.
- 수행 방법 -- 해당 화면에서 수행할 수 있는 태스크가 나열되어 있습니다.
- 목차 -- 모든 도움말 정보의 목차입니다.
- 색인 -- 도움말 항목의 영문자 색인입니다.

GUI 사용에 대한 자세한 내용은 7 페이지의 『GUI 사용에 대한 일반 지침』을 참조하십시오.

구성 마법사

구성 마법사를 사용 중이면, 다음 단계를 수행하십시오.

1. 루트 사용자 또는 관리자로 명령 프롬프트에서 **ssserver**를 실행하여 Site Selector에서 **ssserver**를 시작하십시오.
2. Site Selector의 마법사 기능 **sswizard**를 시작하십시오.

sswizard를 실행하면 명령 프롬프트에서 이 마법사를 실행할 수 있습니다. 또는 GUI에 표시된 대로 Site Selector 구성요소 메뉴에서 구성 마법사를 선택하십시오.

Site Selector 마법사는 Site Selector 구성요소의 기본 구성을 작성하는 프로세스를 단계별로 안내합니다. 네트워크에 대한 질문을 하고 Site Selector가 서버 그룹 간에 통신량을 로드 밸런스할 수 있도록 사이트 이름을 설정하는 단계를 안내합니다.

Site Selector 구성 마법사를 사용하면 다음과 같은 패널이 표시됩니다.

- 마법사 소개
- 예상되는 작업
- 시작하기 전에
- 구성할 호스트 선택(필요한 경우)
- 사이트 이름 정의
- 서버 추가
- 권고자 시작
- 네트워크 근접성 설정

Site Selector 시스템 설정

Site Selector 시스템을 설정하려면 루트 사용자 (AIX, Linux 또는 Solaris의 경우) 또는 Windows 2000에서는 관리자여야 합니다.

설정하는 서버 그룹에 사이트 이름으로 사용할 분석할 수 없는 DNS 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 밸런스합니다.

1단계. 서버 기능 시작

AIX, Linux 및 Solaris: 서버 기능을 시작하려면 **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 351 페이지의 『부록D. 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. 관리 프로그램 기능 시작(선택적)

관리 프로그램 기능은 로드 밸런스를 향상시킵니다. 관리 프로그램 기능을 시작하기 전에 로드 밸런스된 모든 시스템에 Metric Server가 설치되었는지 확인하십시오.

관리 프로그램을 시작하려면 다음 명령을 발행하십시오.

```
sscontrol manager start
```

6단계. 권고자 기능 시작(선택적)

권고자는 요청에 응답하는 로드 밸런스 서버 시스템의 능력에 대한 자세한 정보를 관리 프로그램에 제공합니다. 권고자는 프로토콜마다 고유합니다. Network Dispatcher는 다수의 권고자를 제공합니다. 예를 들어, 지정된 사이트 이름에 대해 HTTP 권고자를 시작하려면 다음 명령을 발행하십시오.

```
sscontrol advisor start http sitename:port
```

7단계. 시스템 메트릭 정의(선택적)

시스템 메트릭 및 Metric Server 사용에 대한 정보는 169 페이지의 『Metric Server』를 참조하십시오.

8단계. 필요한 사이트 이름 비율 설정

권고자를 시작하면 로드 밸런스 결정에 포함되는 권고자(포트) 정보에 부여된 중요도를 수정할 수 있습니다. 사이트 이름 비율을 설정하려면 **sscontrol sitename set sitename proportions** 명령을 실행하십시오. 자세한 정보는 152 페이지의 『상태 정보에 제공되는 중요성 비율』을 참조하십시오.

서버 시스템의 시스템 설정

Metric Server를 Site Selector 구성요소와 함께 사용하는 것이 좋습니다. Site Selector가 로드 밸런스하는 모든 서버 시스템에 Metric Server를 설정하는 것에 대한 정보는 169 페이지의 『Metric Server』를 참조하십시오.

제12장 Consultant for Cisco CSS Switches 구성요소에 대한 계획

이 장에서는 Consultant for Cisco CSS Switches 구성요소를 설치하고 구성하기 전에 네트워크 계획자가 고려해야 할 사항에 대해 설명합니다.

- Cisco CSS Switch 구성요소에 대한 Consultant 매개변수의 로드 밸런스 구성에 대한 정보는 141 페이지의 『제13장 Consultant for Cisco CSS Switches 구성요소 구성』을 참조하십시오.
- 더 많은 고급 기능을 위한 Network Dispatcher 설정 방법에 대한 정보는 149 페이지의 『제14장 고급 Network Dispatcher 기능』을 참조하십시오.
- 원격 인증 관리, Network Dispatcher 로그 및 Network Dispatcher 구성요소 사용법에 대한 정보는 229 페이지의 『제15장 Network Dispatcher 작동 및 관리』를 참조하십시오.

이 장에는 다음의 내용이 있습니다.

- 『하드웨어 및 소프트웨어 요구사항』
- 136 페이지의 『계획 고려사항』

하드웨어 및 소프트웨어 요구사항

- AIX의 경우 15 페이지의 『AIX의 요구사항』을 참조하십시오.
- Linux의 경우 20 페이지의 『Red Hat Linux 또는 SuSE Linux 요구사항』을 참조하십시오.
- Solaris의 경우 23 페이지의 『Solaris 요구사항』을 참조하십시오.
- Windows 2000의 경우 26 페이지의 『Windows 2000 요구사항』을 참조하십시오.

계획 고려사항

Cisco Consultant의 구성은 Cisco CSS Switch의 구성에 따라 다릅니다(137 페이지의 표10 참조). Cisco CSS Switch의 계획 및 구성이 완료되면 Cisco Consultant를 구성하고 사용할 수 있습니다. 설계 및 구성 지침은 Cisco CSS Switch 문서를 참조하십시오.

컨설턴트는 다음과 같이 구성됩니다.

- **lbserver**는 구성 정보를 포함하며 Cisco CSS Switch와 상호작용합니다. "lbc" 접두부는 로드 밸런스 컨설턴트를 의미합니다. lbserver는 다음과 같이 구성됩니다.
 - 구성 정보를 보유하고, Cisco CSS Switch에 연결하는 데 필요한 정보가 들어 있는 실행 프로그램.
 - 가중치를 생성하고 Cisco CSS Switch로 전송하기 위해 수집한 정보를 사용하는 관리 프로그램. 관리 프로그램은 다음에서 정보를 수집합니다.
 - Cisco CSS Switch
 - 서버(권고자 사용)

권고자는 관리 프로그램을 호출하여 가중치를 적절하게 설정하기 전에 서버를 조회하고 프로토콜별 결과를 분석합니다. 현재, Cisco Consultant는 HTTP, FTP, SSL, SMTP, NNTP, IMAP, POP3(및 기타)와 같은 권고자를 제공합니다. 또한 사용자 고유의 권고자를 작성하는 옵션도 있습니다(163 페이지의 『조정(사용자 정의 가능) 권고자 작성』 참조). 권고자를 사용하는 것이 권장되지만 선택사항입니다.
 - 서버(Metric Server 사용)

Metric Server는 컨설턴트에 서버 로드 정보를 시스템별 매트릭 양식으로 제공하여 서버의 상태를 보고합니다. 관리 프로그램은 에이전트에서 수집한 매트릭을 통해 각 서버에 있는 Metric Server를 조회하여 가중치를 로드 밸런스 프로세스에 지정합니다. 결과도 관리 프로그램 보고서에 저장됩니다.
- 실행 프로그램, 권고자 및 관리 프로그램을 구성하고 관리하기 위해 명령행과 그래픽 사용자 인터페이스가 제공됩니다.

- **lbcontrol**은 컨설턴트에 대한 명령행 인터페이스입니다.
- **ndadmin**은 컨설턴트를 구성하고 상태를 모니터링하는 데 사용하는 그래픽 사용자 인터페이스입니다.

관리 프로그램은 Cisco CSS Switch, 권고자와 Metric Server로부터 정보를 수집합니다. 관리 프로그램이 받은 정보를 기반으로, 각 포트에서 서버 시스템에 가중치가 지정되는 방법을 조정 하고 새 연결을 밸런스하는 데 사용할 새로운 가중치를 Cisco CSS Switch에 제공합니다. 관리 프로그램이 서버가 다운되었음을 발견하면 서버에 가중치 0이 지정되고 서버가 일시중단됩니다. 그리고 나서, Cisco CSS Switch는 해당 서버로의 통신 전달을 정지시킵니다.

권고자는 지정된 포트에서 각 서버를 모니터링하여 서버의 응답 시간과 사용 가능성을 판별한 다음, 이 정보를 관리 프로그램에 제공합니다. 또한 권고자는 서버의 연결 또는 단절도 모니터링합니다.

컨설턴트를 올바르게 구성하려면 Cisco CSS Switch와 같이 구성해야 합니다. 먼저 *Cisco Services Switch* 시작하기 안내서를 참조하여 Cisco CSS Switch를 구성하십시오. Switch가 올바르게 작동하는지 확인한 다음, 컨설턴트를 구성하십시오.

Cisco CSS Switch 구성은 다음과 같이 소유자, 콘텐츠 규칙 및 컨설턴트 구성에 맵핑되는 서비스로 이루어집니다.

표 10. 컨설턴트 및 Cisco CSS Switch 구성 용어

Cisco CSS Switch	컨설턴트
둘 이상의 소유자에 대한 콘텐츠 규칙의 가상 IP 주소(VIP)	클러스터
콘텐츠 규칙에 포함된 포트	포트
서비스	서버

컨설턴트 구성 트리는 다음과 같이 구성됩니다.

- 클러스터는 분석 가능한 이름 또는 점분리 10진수 주소입니다.
- 포트는 해당 프로토콜에 사용하는 포트의 번호입니다.
- 서버.

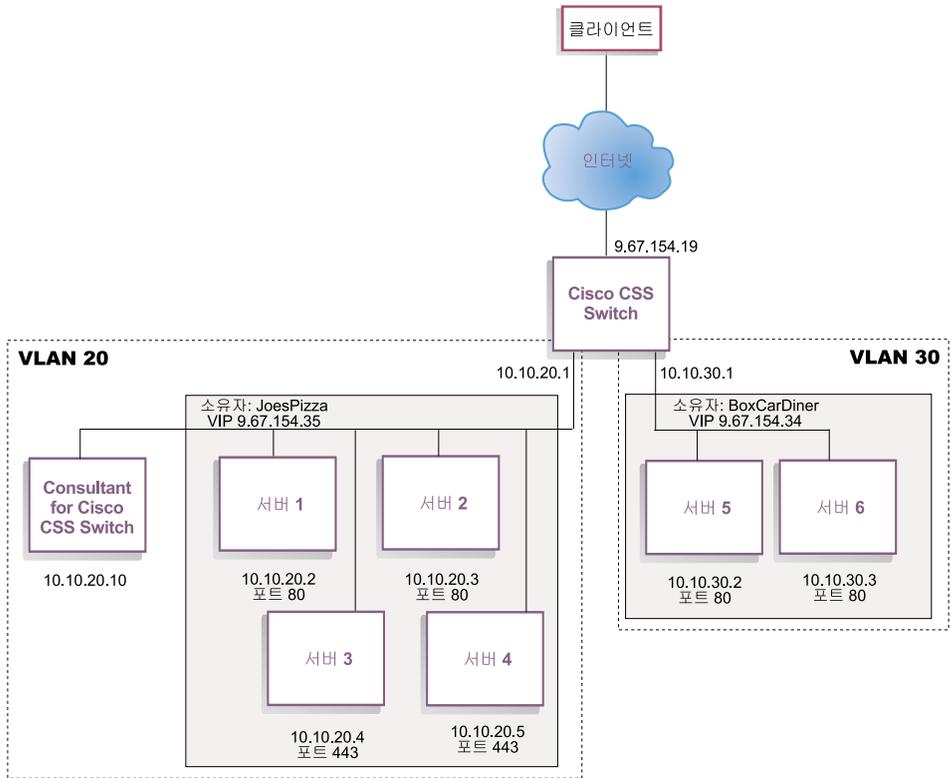


그림 21. 각각 포트가 세 개인 두 개의 클러스터로 구성된 컨설턴트에 대한 예제

그림 21에서,

- 9.67.154.19는 인터넷에 대한 네트워크 연결입니다.
- 두 개의 VLAN(20 및 30)이 구성됩니다.

실행 프로그램을 구성할 때는 주소와 SNMP 공동체를 구성해야 하며, Cisco CSS Switch의 해당 속성과 일치해야 합니다. 실행 프로그램 구성에 대한 정보는 390 페이지의 『Ibcontrol executor -- 실행 프로그램 제어』를 참조하십시오.

표 11. 컨설턴트 구성에 맵핑한 Cisco CSS Switch 구성에 대한 예제

Cisco CSS Switch 구성	컨설턴트 구성
username admin superuser snmp community community private read-write	lbcontrol executor set address 10.10.20.1 lbcontrol executor set communityname community
content rule1 port 80 balance weightedrr add service server1 add service server2 vip address 9.67.154.35 active	lbcontrol cluster add 9.67.154.35 lbcontrol port add 9.67.154.35: 80
content rule 2 protocol tcp port 443 balance weightedrr add service server3 add service server4 vip address 9.67.154.35 active	lbcontrol port add 9.67.154.35: 443
service server1 ip address 10.10.20.2 port 80 weight 4 active	lbcontrol server add 9.67.154.35: 80 :server1 address 10.10.20.2
service server3 ip address 10.10.20.4 port 443 weight 4 active	lbcontrol server add 9.67.154.35: 443 :server3 address 10.10.20.4

제13장 Consultant for Cisco CSS Switches 구성요소 구성

이 장의 단계를 수행하기 전에, 135 페이지의 『제12장 Consultant for Cisco CSS Switches 구성요소에 대한 계획』을 참조하십시오. 여기서는 Network Dispatcher의 Consultant for Cisco CSS Switches 구성요소에 대한 기본 구성을 작성하는 방법에 대해 설명합니다.

- Network Dispatcher 복합 구성에 대한 정보는 149 페이지의 『제14장 고급 Network Dispatcher 기능』을 참조하십시오.
- 원격 인증 관리, Network Dispatcher 로그 및 Network Dispatcher 구성요소 사용법에 대한 정보는 229 페이지의 『제15장 Network Dispatcher 작동 및 관리』를 참조하십시오.

구성 태스크 개요

여기서 구성 방법을 시작하기 전에 다음을 수행하십시오.

1. Cisco CSS Switch 및 모든 서버 시스템이 올바르게 구성되어 있는지 확인하십시오.
2. Cisco Consultant를 구성하여 실행 프로그램의 주소 및 SNMP 공동체 이름이 Cisco CSS Switch의 해당 속성과 일치하는지 확인하십시오. 실행 프로그램 구성에 대한 정보는 390 페이지의 『Ibcontrol executor -- 실행 프로그램 제어』를 참조하십시오.

표 12. Consultant for Cisco CSS Switches 구성요소의 구성 태스크

태스크	설명	관련 정보
Consultant for Cisco CSS Switches 시스템 설정	요구사항 찾기	145 페이지의 『Consultant for Cisco CSS Switches 시스템 설정』
구성 검사	구성의 작동 여부 확인	147 페이지의 『구성 검사』

구성 방법

다음은 Network Dispatcher의 Consultant for Cisco CSS Switches 구성요소에 대해 기본 구성을 작성하기 위한 3가지 방법입니다.

- 명령행
- 스크립트
- GUI(Graphical User Interface)

명령행

이것은 Cisco Consultant를 구성하는 가장 직접적인 방법입니다. 이 책의 절차에서는 명령행을 사용한다고 가정합니다. 명령 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름(예: 클러스터 및 서버 명령에 사용됨) 및 파일 이름은 예외입니다.

명령행에서 Cisco Consultant를 시작하려면 다음을 수행하십시오.

- 명령 프롬프트에서 **lbserver** 명령을 발행하십시오.

주: 서비스를 정지시키려면 **lbserver stop**을 발행하십시오.

- 그 다음, 원하는 Cisco Consultant 제어 명령을 발행하여 구성을 설정하십시오. 명령은 **lbcontrol**입니다. 명령에 대한 자세한 내용은 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』를 참조하십시오.

lbcontrol 명령 매개변수의 축약된 버전을 입력할 수 있습니다. 매개변수의 고유한 문자만 입력해야 합니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보기 위해 **lbcontrol help file** 대신 **lbcontrol he f**를 입력할 수 있습니다.

명령행 인터페이스를 시작하려면 **lbcontrol**을 발행하여 lbcontrol 명령 프롬프트를 시작하십시오.

명령행 인터페이스를 종료하려면 **exit** 또는 **quit** 명령을 발행하십시오.

주: Windows 2000에서 Dispatcher 구성요소의 ndserver는 자동으로 시작됩니다. Cisco Consultant만 사용하고 Dispatcher 구성요소는 사용하지 않는 경우, 다음과 같이 ndserver가 자동으로 시작하는 것을 정지시킬 수 있습니다.

1. Windows 2000 서비스 창에서 IBM Dispatcher를 마우스 오른쪽 단추로 누르십시오.
2. 등록 정보를 선택하십시오.
3. 시동 유형 필드에서 수동을 선택하십시오.
4. 확인을 누르고, 서비스 창을 닫으십시오.

스크립트

Consultant for Cisco CSS Switches 구성 명령은 구성 스크립트 파일에 입력되어 함께 실행될 수 있습니다.

주: 스크립트 파일(예: myscript)의 내용을 빨리 실행하려면 다음의 한 명령을 사용하십시오.

- 현재 구성을 갱신하려면 다음과 같이 스크립트 파일에서 실행 가능 명령을 실행하십시오.

```
lbcontrol file appendload myscript
```

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

```
lbcontrol file newload myscript
```

GUI

GUI(Graphical User Interface)의 예제는 6 페이지의 그림2를 참조하십시오.

GUI를 시작하려면 다음의 단계를 따르십시오.

1. lbserver가 아직 실행되지 않으면 루트로 다음과 같이 실행하여 바로 시작하십시오.

```
lbserver
```

2. 그리고 나서, 다음 중 하나를 수행하십시오.

- AIX, Linux 또는 Solaris의 경우 **ndadmin**을 입력하십시오.
- Windows 2000의 경우 시작, 프로그램을 누르고, **IBM WebSphere**를 누른 다음, **Edge Server**, **IBM Network Dispatcher**를 누르고, **Network Dispatcher**를 누르십시오.

GUI에서 Cisco Consultant 구성요소를 구성하려면 다음을 수행하십시오.

1. 트리 구조에서 마우스 오른쪽 단추로 Cisco Consultant를 누르십시오.
2. 호스트에 연결하십시오.
3. 포트 및 서버를 포함하는 클러스터를 작성하십시오.
4. 관리 프로그램을 시작하십시오.
5. 관리 프로그램의 권고자를 시작하십시오.

GUI를 사용하면 **lbcontrol** 명령으로 할 수 있는 모든 작업을 수행할 수 있습니다. 예를 들어, 명령행을 사용하여 클러스터를 정의하려면, **lbcontrol cluster add cluster** 명령을 입력하십시오. GUI에서 클러스터를 정의하려면, 실행 프로그램을 마우스 오른쪽 단추로 누른 다음, 클러스터 추가를 누르십시오. 팝업 창에 클러스터 주소를 입력한 후 확인을 누르십시오.

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 Cisco Consultant 구성 파일을 로드할 수 있습니다. 다른 구성 파일 이름으로 저장 옵션을 선택하여 Cisco Consultant 구성을 파일에 주기적으로 저장하십시오. 메뉴 표시줄에서 파일을 눌러 파일에 대한 현재 호스트 연결을 저장하거나 모든 Network Dispatcher 구성요소에 있는 기존 파일의 연결을 복원하십시오.

도움말에 액세스하려면 Network Dispatcher 창 오른쪽 상단 구석의 물음표를 누르십시오.

- 필드 도움말 -- 각 필드와 기본값을 설명합니다.
- 수행 방법 -- 해당 화면에서 수행할 수 있는 태스크가 나열되어 있습니다.
- 목차 -- 모든 도움말 정보의 목차입니다.
- 색인 -- 도움말 항목의 영문자 색인입니다.

GUI 사용에 대한 자세한 내용은 7 페이지의 『GUI 사용에 대한 일반 지침』을 참조하십시오.

Consultant for Cisco CSS Switches 시스템 설정

Consultant for Cisco CSS Switches 시스템을 설정하려면 사용자는 루트 사용자(AIX, Linux 또는 Solaris) 또는 Windows 2000 관리자여야 합니다.

컨설턴트는 Cisco CSS Switch에 Cisco CSS Switch 관리자로서 연결될 수 있습니다.

실행 프로그램을 구성할 때, 주소를 구성하고 SNMP 공동체 이름이 Cisco CSS Switch의 해당 속성과 일치해야 합니다.

이 절차에서 사용되는 명령에 대한 도움말을 보려면 381 페이지의 『부록E. Consultant for Cisco CSS Switches 명령어 참조서』를 참조하십시오.

1단계. 서버 기능 시작

lbserver가 아직 실행되지 않으면 루트로 다음과 같이 실행하여 바로 시작하십시오.

```
lbserver
```

2단계. 실행 프로그램 기능 구성

주소 및 SNMP 공동체 이름을 구성해야 합니다. 이 값은 Cisco CSS Switch의 해당 속성과 일치해야 합니다.

3단계. 클러스터 정의 및 클러스터 옵션 설정

클러스터는 분해될 수 있는 이름이거나 점분리 10진수 주소입니다. 클러스터는 소유자 컨텐츠 규칙의 Cisco CSS Switch 가상 IP 주소와 일치합니다.

클러스터를 정의하려면 **lbcontrol cluster add cluster**를 입력하십시오. 클러스터 옵션을 설정하려면 **lbcontrol cluster set**를 입력하십시오.

4단계. 포트 정의 및 포트 옵션 설정

포트를 정의하려면 **lbcontrol port add cluster:port**를 입력하십시오. 포트는 소유자의 Cisco CSS Switch 컨텐츠 규칙에 구성된 포트와 일치합니다.

포트는 Cisco CSS Switch에 대한 소유자 콘텐츠 규칙에 지정된 해당 프로토콜에 사용 중인 포트 번호입니다. 자세한 정보는 406 페이지의 『Ibcontrol port -- 포트 구성』을 참조하십시오.

5단계. 로드 밸런스 서버 시스템 정의

클러스터 및 포트 내의 동일한 서버의 여러 인스턴스를 구성할 수 있습니다(주소 및 SNMP 공동체 이름이 Cisco CSS Switch의 해당 속성과 일치해야 합니다). 동일한 서버의 여러 인스턴스를 구성하는 경우, 동일한 물리적 시스템에 있는 다른 응용프로그램 서버를 구별하고 동일한 포트의 동일한 IP 주소에 응답할 수 있습니다.

로드 밸런스 서버 시스템을 정의하려면 다음을 입력하십시오.

```
lbcontrol server add cluster:port:server address x.x.x.x / hostname
```

서버는 Cisco CSS Switch 서비스 이름과 일치합니다.

로드 밸런스를 수행하려면 클러스터의 포트에 서버를 두 개 이상 정의해야 합니다. 그렇지 않으면, 통신이 하나의 서버에만 전달됩니다. 172 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』를 참조하십시오.

Ibcontrol 서버 명령 구문에 대한 자세한 정보는 408 페이지의 『Ibcontrol server -- 서버 구성』을 참조하십시오.

6단계. 관리 프로그램 기능 시작

관리 프로그램을 시작하려면 **lbcontrol manager start** 명령을 입력하십시오. 자세한 정보는 397 페이지의 『Ibcontrol manager -- 관리 프로그램 제어』를 참조하십시오.

7단계. 권고자 기능 시작(선택)

권고자는 요청에 응답하기 위해 로드 밸런스 서버 시스템의 능력에 대한 자세한 정보를 관리 프로그램에 제공합니다. 권고자는 프로토콜마다 고유합니다. 예를 들어 HTTP 권고자를 시작하려면 다음 명령을 실행하십시오.

```
lbcontrol advisor start http port
```

권고자 목록 및 기본 포트에 대해서는 382 페이지의 『lbccontrol advisor -- 권고자 제어』를 참조하십시오. 각 권고자에 대한 설명은 161 페이지의 『권고자 목록』을 참조하십시오.

8단계. 필요한 클러스터 비율 설정

모든 권고자를 시작하려면 권고자 정보가 로드 밸런스 결정 시 포함되도록 클러스터 비율을 반드시 변경하십시오. **lbccontrol cluster proportions** 명령을 사용하십시오. 152 페이지의 『상태 정보에 제공되는 중요성 비율』을 참조하십시오.

주: 권고자를 시작하고 시스템 메트릭에 제공된 비율이 0인 경우, 비율이 1로 증가합니다. 클러스터 비율이 총 100이므로 이 경우 최고값인 비율이 1만큼 줄어듭니다.

9단계. Metric Server 시작(선택)

Metric Server 사용에 대한 정보는 169 페이지의 『Metric Server』를 참조하십시오.

구성 검사

구성이 작동되고 있는지 확인하십시오.

1. 관리 프로그램 loglevel을 4로 설정하십시오.
2. 1분 동안 Cisco CSS Switch에서 서버의 연결을 끊으십시오. 또는 1분 동안 응용프로그램 서버를 종료하십시오.
3. 서버를 다시 연결하거나 응용프로그램 서버를 다시 시작하십시오.
4. 관리 프로그램 loglevel을 원하는 레벨(1)로 다시 설정하십시오.
5. .../nd/servers/logs/lbc 디렉토리에 있는 manager.log 파일을 보고, **setServerWeights setting service**를 찾으십시오.

제14장 고급 Network Dispatcher 기능

이 장에서는 Network Dispatcher의 로드 밸런스 매개변수를 구성하는 방법과 고급 기능의 Network Dispatcher를 설정하는 방법에 대해 설명합니다.

주: 이 장을 읽을 때 Dispatcher 구성요소를 사용하고 있지 않는 경우 "ndcontrol"을 다음 사항으로 대체하십시오.

- CBR의 경우, **cbrcontrol** 사용
- Mailbox Locator의 경우, **mlcontrol** 사용
- Site Selector의 경우, **sscontrol** 사용(351 페이지의 『부록D. Site Selector 명령어 참조서』 참조)
- Cisco Consultant의 경우, **lbcontrol** 사용(381 페이지의 『부록E. Consultant for Cisco CSS Switches 명령어 참조서』 참조)

표 13. Network Dispatcher의 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 기본 비율은 50-50-0-0입니다. 기본값을 사용할 경우, 권고자나 Metric Server의 정보는 사용되지 않습니다. • 가중치 • 관리 프로그램 고정 가중치 • 관리 프로그램 간격 • 감도 임계치 • 스무스 색인(smoothing index) 	152 페이지의 『Network Dispatcher가 제공하는 로드 밸런스 최적화』
관리 프로그램에서 서버의 작동 중지/가동을 표시할 때 스크립트를 사용하여 경보나 레코드 서버 장애헤 생성하십시오.	Network Dispatcher는 관리 프로그램에서 서버의 작동 중지/가동을 표시할 때 사용자 정의할 수 있는 스크립트를 트리거하는 사용자 종료를 제공합니다.	157 페이지의 『스크립트를 사용하여 경보나 레코드 서버 장애 생성』

표 13. Network Dispatcher의 고급 구성 태스크 (계속)

태스크	설명	관련 정보
권고자 사용 및 조정 권고자 작성	사용자 서버의 특정 상태를 보고하기 위해 권고자 및 고유의 조정 권고자를 작성하는 방법에 대해 설명합니다.	157 페이지의 『권고자』 163 페이지의 『조정(사용자 정의 가능) 권고자 작성』
작업로드 관리 프로그램 권고자 (WLM) 사용	WLM 권고자는 Network Dispatcher에 시스템 로드 정보를 제공합니다.	168 페이지의 『작업로드 관리 프로그램 권고자』
Metric Server 에이전트 사용	Metric Server는 Network Dispatcher에 시스템 로드 정보를 제공합니다.	169 페이지의 『Metric Server』
서버 파티션 사용	제공된 서버에 따라 로드를 분산시킬 논리 서버 정의	172 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』
권고자 요청/응답(URL) 옵션 사용	시스템에 조회할 서비스 고유의 클라이언트 HTTP URL 문자열 정의	174 페이지의 『HTTP 권고자 요청/응답(URL) 옵션』
Network Dispatcher를 로드 밸런스 중인 시스템에 결합 배치	결합 배치된 Network Dispatcher 시스템을 설정하십시오.	175 페이지의 『결합 배치된 서버 사용』
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.	177 페이지의 『광역 Dispatcher 지원 구성』
고가용성 또는 상호 고가용성 구성	두 번째 Dispatcher 시스템을 백업 시스템으로 설정합니다.	187 페이지의 『고가용성』
규칙 기반 로드 밸런스 구성	사용자 서버의 서브셋이 사용될 조건을 정의합니다.	196 페이지의 『규칙 기반 로드 밸런스 구성』
명시적 링크 사용	링크에서 Dispatcher를 생략하지 않습니다.	207 페이지의 『명시적 링크 사용』
사설 네트워크 사용	Dispatcher를 구성하여 사설 네트워크 서버의 로드 밸런스를 수행합니다.	208 페이지의 『사설 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.	209 페이지의 『와일드 카드 클러스터를 사용하여 서버 구성 조합』
와일드 카드 클러스터를 사용하여 방화벽 로드 밸런스 수행	방화벽에 대해 모드 통신량의 로드 밸런스를 수행하게 됩니다.	210 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.	211 페이지의 『투명 프록시의 경우 Caching Proxy가 있는 와일드 카드 클러스터 사용』

표 13. Network Dispatcher의 고급 구성 태스크 (계속)

태스크	설명	관련 정보
와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정	특정 포트에 구성되어 있지 않은 통신량을 처리합니다.	212 페이지의 『와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정』
고급 친근성을 사용하여 클러스터의 포트가 엄격하게 유지되도록 구성	클라이언트 요청이 동일한 서버로 전송되도록 허용합니다.	212 페이지의 『Network Dispatcher에 대한 친근성 기능 사용법』
서버 지정 유사 API 사용	외부 에이전트가 Dispatcher 유사 활동에 영향을 줄 수 있는 API를 제공합니다.	213 페이지의 『클라이언트 서버 유사성을 제어하는 서버 지정 유사 API』
포트간 친근성을 사용하여 포트 사이에서 고급 친근성 기능 확장	다른 포트에서 수신된 클라이언트 요청이 동일한 서버로 전송되도록 허용합니다.	214 페이지의 『포트간 친근성』
친근성 주소 마스크를 사용하여 공통 IP 서브넷 주소 지정	동일한 서브넷에서 수신된 클라이언트 요청이 동일한 서버로 전송되도록 허용합니다.	215 페이지의 『친근성 주소 마스크』
규칙 친근성 무시를 사용하여 서버가 포트 친근성 기능을 무시할 수 있는 메커니즘 제공	서버가 포트에 대한 친근성 설정을 무시할 수 있게 허용합니다.	216 페이지의 『규칙 친근성 무시』
활성 쿠키 친근성을 사용하여 CBR에 대한 서버의 로드 밸런스	특정 서버에 대해 세션이 친근성을 유지할 수 있게 하는 규칙 옵션	218 페이지의 『활성 쿠키 연관 관계』
Dispatcher의 Content Based Routing 및 CBR 구성요소의 경우, 수동 쿠키 친근성을 사용하여 서버의 로드 밸런스 수행	쿠키 이름/쿠키 값에 기반한 특정 서버의 경우, 세션이 친근성을 유지할 수 있도록 허용하는 규칙 옵션	219 페이지의 『수동 쿠키 연관 관계』
URI 친근성을 사용하여 개별 서버에 캐시할 고유의 내용이 있는 Caching Proxy 서버에서 로드 밸런스 수행	URI에 기반한 특정 서버의 경우, 세션이 친근성을 유지할 수 있도록 허용하는 규칙 옵션	220 페이지의 『URI 연관 관계』
"서비스 거부 중지" 감지를 사용하여 관리자에게 잠재적 시각 통지(경보 사용)	Dispatcher는 서버의 방대한 양의 반개방 TCP 연결에 대한 수신 요청을 분석합니다.	222 페이지의 『서비스 거부 중지 감지』
2진 로그를 사용하여 서버 통계 분석	서버 정보가 2진 파일에 저장되어 이 파일에서 검색될 수 있습니다.	223 페이지의 『서버 통계를 분석하기 위해 2진 로그 사용』
Cisco Consultant 사용(추가 정보)	Cisco Consultant가 Cisco CSS Switch 및 가중치 구성에 대한 추가 정보와 상호작용하는 방법입니다.	225 페이지의 『고급 Cisco Consultant 기능에 대한 추가 정보』

Network Dispatcher가 제공하는 로드 밸런스 최적화

Network Dispatcher의 관리 프로그램 기능은 다음 설정에 따라 로드 밸런스를 수행합니다.

- 『상태 정보에 제공되는 중요성 비율』
- 154 페이지의 『가중치』
- 155 페이지의 『관리 프로그램 간격』
- 159 페이지의 『권고자 간격』
- 160 페이지의 『권고자 보고서 시간 종료』
- 156 페이지의 『감도 임계치』
- 156 페이지의 『스무스 색인』

이 설정을 변경하여 네트워크에 대한 로드 밸런스를 최적화할 수 있습니다.

상태 정보에 제공되는 중요성 비율

관리 프로그램에서는 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.

또는

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.

또는

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 권고자가 입력.

- **시스템 메트릭:** Metric Server 또는 WLM과 같은 시스템 모니터링 도구로 입력.

관리 프로그램에서는 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, 관리 프로그램이 Metric Server에서 처음 두 개의 값(cpu 및 메모리)을 확보합니다. Cisco Consultant의 경우 관리 프로그램이 Cisco CSS Switch에서 처음 두 개의 값(활성 연결 및 새 연결)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 권고자와 시스템 정보를 무시합니다. 사용자 환경에서 최상의 성능을 제공하는 조합을 찾기 위해 다른 비율을 시도할 필요가 있습니다.

주: 권고자(WLM 이외)를 추가할 때, 포트 비율이 0이면 관리 프로그램에서 이 값을 1로 늘립니다. 상대 비율의 합이 100이어야 하므로 최고값이 1만큼 줄어듭니다.

WLM 권고자를 추가할 때, 시스템 메트릭 비율이 0이면 관리 프로그램에서 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 적어 집니다. 클라이언트 연결이 느려지면(예: 데이터베이스 조회), 활성 연결 수는 많아 집니다.

활성 및 새 연결 비율을 너무 작게 설정하지 마십시오. 처음 두 값을 각각 최소 20으로 설정하지 않으면 Network Dispatcher의 로드 밸런스 및 스무스 색인은 사용 불가능합니다.

중요성 비율의 값을 설정하려면 **ndcontrol** 클러스터 설정 클러스터 비율 명령을 사용하십시오. 287 페이지의 『ndcontrol cluster -- 클러스터 구성』에서 자세한 정보를 참조하십시오.

가중치

주: Cisco Consultant 구성요소를 사용하는 경우, 추가 정보는 227 페이지의 『Cisco Consultant가중치』를 참조하십시오.

가중치는 실행 프로그램의 내부 카운터, 권고자의 피드백 및 Metric Server와 같은 시스템 모니터링 프로그램의 피드백에 따라 관리 프로그램 기능에서 설정됩니다. 관리 프로그램을 실행 중에 수동으로 가중치를 설정하려면 ndcontrol 서버 명령에서 **fixedweight** 옵션을 지정하십시오. **fixedweight** 옵션에 대한 설명은 155 페이지의 『관리 프로그램 고정 가중치』를 참조하십시오.

가중치는 포트의 모든 서버에 적용됩니다. 특정 포트의 경우, 요청은 서로에 대해 상대적인 가중치에 따라 서버간에 분산됩니다. 예를 들어, 하나의 서버가 가중치 10으로 설정되고 다른 서버가 가중치 5로 설정되면 10으로 설정된 서버의 요청수는 5로 설정된 서버 요청수의 두 배가 됩니다.

모든 서버가 가질 수 있는 최대 가중치 경계를 지정하려면 **ndcontrol port set weightbound** 명령을 입력하십시오. 이 명령은 각 서버가 확보할 요청수 사이에 있을 수 있는 차이에 영향을 줍니다. 최대 가중치를 1로 설정할 경우, 모든 서버의 가중치는 1, 작업중지인 경우에는 0, 단절 표시인 경우에는 -1입니다. 이 숫자가 증가할수록 서버의 가중치 차이도 증가합니다. 최대 가중치가 2이면 한 서버가 다른 서버 요청수의 두 배를 확보할 수 있습니다. 최대 가중치가 10이면, 한 서버가 다른 서버 요청수의 10배를 확보할 수 있습니다. 기본 최대 가중치는 20입니다.

권고자가 서버 종료로 발견하면, 이를 관리 프로그램에 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.

관리 프로그램 고정 가중치

관리 프로그램이 없으면 권고자를 실행할 수 없으며 서버가 작동 중단되었는지 검색할 수도 없습니다. 권고자를 실행하기로 했으나 관리 프로그램이 특정 서버에 대해 사용자가 설정한 가중치를 갱신하지 않기 보려면 `ndcontrol server` 명령에서 **fixedweight** 옵션을 사용하십시오.

예제:

```
ndcontrol server set cluster:port:server fixedweight yes
```

고정 가중치가 `yes`로 설정되면 `ndcontrol server set weight` 명령을 사용하여 원하는 값으로 가중치를 설정하십시오. 서버 가중치는 고정 가중치가 `no`로 설정된 다른 `ndcontrol server` 명령을 실행할 때까지는 관리 프로그램이 실행 중일 때 고정 가중치로 유지합니다. 자세한 정보는 335 페이지의 『`ndcontrol server -- 서버 구성`』을 참조하십시오.

관리 프로그램 간격

전체 성능을 최적화하기 위해 관리 프로그램이 실행 프로그램과 상호작용할 수 있는 횟수가 제한됩니다. 이 간격은 `ndcontrol manager interval` 및 `ndcontrol manager refresh` 명령을 입력하여 변경할 수 있습니다.

관리 프로그램 간격은 실행 프로그램이 연결 경로 지정에서 사용하는 서버 가중치를 관리 프로그램이 갱신하는 횟수를 지정합니다. 관리 프로그램 간격이 너무 작으면, 관리 프로그램에서 일정하게 실행 프로그램을 인터럽트한 결과로서 성능이 저하될 수 있다는 것을 나타냅니다. 관리 프로그램 간격이 너무 크면, 실행 프로그램의 요청 경로 지정이 정확한 최신 정보를 기초로 하지 않는다는 것을 나타냅니다.

예를 들어, 관리 프로그램 간격을 1초로 설정하려면 다음 명령을 입력하십시오.

```
ndcontrol manager interval 1
```

관리 프로그램 갱신 주기는 관리 프로그램에서 실행 프로그램에 상태 정보를 묻는 횟수를 지정합니다. 갱신 주기는 간격 시간을 기초로 합니다.

예를 들어, 관리 프로그램 갱신 주기를 3으로 설정하려면 다음 명령을 입력하십시오.

이렇게 하면 관리 프로그램에서는 실행 프로그램에 상태를 묻기 전에 세 개의 간격 동안 기다립니다.

감도 임계치

Network Dispatcher는 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 백분율 가중치 변경값이 감도 임계치보다 큰 경우, 관리 프로그램에서는 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, 백분율 변경값이 임계치를 넘지 않으므로 관리 프로그램은 실행 프로그램에서 사용하는 가중치를 갱신하지 않습니다. 그러나 총 가중치가 100에서 106으로 변경되면, 관리 프로그램에서는 이 가중치를 갱신합니다. 관리 프로그램의 감도 임계치를 기본값(예: 6)이 아닌 다른 값으로 설정하려면, 다음 명령을 입력하십시오.

```
ndcontrol manager sensitivity 6
```

대부분의 경우, 이 값을 변경할 필요가 없습니다.

스무스 색인

관리 프로그램에서는 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. 관리 프로그램에서는 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, 관리 프로그램에서는 스무스 색인을 사용합니다. 스무스 색인은 서버의 가중치가 변경될 수 있는 정도를 제한하여, 요청 분산 시 변

경을 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.

```
ndcontrol manager smoothing 4
```

대부분의 경우, 이 값을 변경할 필요가 없습니다.

스크립트를 사용하여 경보나 레코드 서버 장애 생성

Network Dispatcher는 사용자 정의할 수 있는 스크립트를 트리거하는 사용자 종료를 제공합니다. 스크립트를 작성하여 관리 프로그램에서 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 `...nd/servers/samples` 설치 디렉토리에 있습니다. 파일을 실행하려면 예제 스크립트를 `...nd/servers/bin` 디렉토리로 이동하여 확장자가 `".sample"`인 파일을 제거하십시오. 다음의 예제 스크립트가 제공됩니다.

- **serverDown** - 관리 프로그램에서 서버를 단절 표시합니다.
- **serverUp** - 관리 프로그램에서 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.

권고자

권고자는 Network Dispatcher 내의 에이전트입니다. 서버 시스템의 상태와 로딩을 평가하는 기능을 합니다. 권고자는 서버와의 활성 클라이언트와 같은 교환으로 이를 수행합니다. 권고자는 응용프로그램 서버의 경량 클라이언트로서 간주될 수 있습니다.

Dispatcher는 가장 널리 쓰이는 프로토콜용으로 몇 가지 프로토콜 고유의 권고자를 제공합니다. 그러나 Network Dispatcher의 모든 구성요소와 함께 제공된 모든 권고자를 사용할 수 없습니다(예를 들어, Telnet 권고자를 CBR 구성요소와 함께

사용할 수 없습니다). 또한 Network Dispatcher는 사용자가 고유의 권고자를 작성할 수 있는 『조정 권고자』의 개념을 지원합니다.

Linux에서 특정 바인드 서버 응용프로그램 제한사항 Linux에서 Network Dispatcher는 Mailbox Locator 및 Site Selector와 같은 다른 Network Dispatcher 구성요소를 포함하여 특정 바인드 서버 응용프로그램이 있는 서버가 클러스터 IP 주소에 바인드할 때 해당 서버의 로드 밸런스 시 권고자 사용을 지원하지 않습니다.

권고자 작동 방법

권고자는 정기적으로 각 서버와의 TCP 연결을 열어 서버에 요청 메시지를 전송합니다. 메시지 내용은 서버에서 실행 중인 프로토콜에 따라 고유합니다. 예를 들어, HTTP 권고자는 서버에 HTTP 『HEAD』 요청을 전송합니다.

그런 후 권고자는 서버의 응답을 인식합니다. 응답을 받은 후, 권고자는 서버를 평가합니다. 이 『로드』 값을 계산하기 위해, 대부분의 권고자는 응답할 서버의 시간을 측정한 후 이 값(밀리초 단위)을 로드로 사용합니다.

그러면 권고자는 『포트』 컬럼에서 관리 프로그램 보고서에 표시되는 로드값을 Dispatcher의 관리 프로그램 기능에 보고합니다. 관리 프로그램은 비올당 모든 소스의 집계 가중치를 계산하여 이 가중치를 실행 프로그램 기능으로 설정합니다. 그러면 실행 프로그램에서는 새로운 수신 클라이언트 연결 로드 밸런스에 이들 가중치를 사용합니다.

권고자는 서버가 적당하게 작동된다고 판별한 경우, 관리 프로그램에 0이 아닌 양수의 로드 숫자를 보고합니다. 권고자가 서버 작동이 되지 않는다고 판별한 경우, 특수 로드값인 음수 1(-1)을 리턴합니다. 관리 프로그램과 실행 프로그램은 해당 서버에 더 이상 연결을 전달하지 않습니다.

권고자 시작 및 정지

모든 클러스터(그룹 권고자)에서 특정 포트에 대해 권고자를 시작할 수 있습니다. 또는 동일한 포트이지만 다른 클러스터에서 다른 권고자를 실행하도록 선택할 수 있습니다(클러스터/사이트 고유의 권고자). 예를 들어, 세 개의 클러스터(*clusterA*,

clusterB, *clusterC*)로 정의된 Network Dispatcher가 있는 경우, 포트 80이 있는 각 클러스터에서 다음 사항을 수행할 수 있습니다.

- 클러스터/사이트 고유의 권고자: *clusterA*의 포트 80에서 권고자를 시작하려면 클러스터 및 포트를 모두 지정하십시오.

```
ndcontrol advisor start http clusterA:80
```

이 명령은 *clusterA*의 포트 80에서 http 권고자를 시작합니다. http 권고자는 *clusterA*의 포트 80에 첨부된 모든 서버에서 권고합니다.

- 그룹 권고자: 기타 모든 클러스터의 포트 80에서 조정 권고자를 시작하려면 포트를 지정하십시오.

```
ndcontrol advisor start ADV_custom 80
```

이 명령은 *clusterB* 및 *clusterC*의 포트 80에서 *ADV_custom* 권고자를 시작합니다. 조정 권고자는 *clusterB* 및 *clusterC*의 포트 80에 첨부된 모든 서버에서 권고합니다(조정 권고자에 대한 자세한 정보는 163 페이지의 『조정(사용자 정의 기능) 권고자 작성』을 참조하십시오).

주: 그룹 권고자는 현재 클러스터/사이트 고유의 권고자가 없는 모든 클러스터/사이트에서 권고합니다.

그룹 권고자에 대한 위의 구성 예제를 사용하여 두 클러스터(*clusterB* 및 *clusterC*) 모두 또는 임의 클러스터의 포트 80에서 *ADV_custom* 조정 권고자를 정지하도록 선택할 수 있습니다.

- *clusterB*의 포트 80에서 조정 권고자를 정지하려면 클러스터 및 포트를 지정하십시오.

```
ndcontrol advisor stop ADV_custom clusterB:80
```

- *clusterB* 및 *clusterC*의 포트 80에서 조정 권고자를 정지하려면 포트만 지정하십시오.

```
ndcontrol advisor stop ADV_custom 80
```

권고자 간격

주: 권고자 기본값은 가능한 많은 시나리오에 효율적으로 적용됩니다. 기본값이 아닌 다른 값을 입력할 경우에는 주의하십시오.

권고자 간격은 권고자가 모니터하는 포트의 서버로부터 상태를 묻는 횟수를 설정하고, 관리 프로그램에 결과를 보고합니다. 권고자 간격이 너무 작으면, 권고자가 일정하게 서버를 인터럽트한 결과로 성능이 저하될 수 있다는 것을 나타냅니다. 권고자 간격이 너무 크면, 관리 프로그램의 가중치에 대한 결정이 정확한 최신 정보를 기초로 하지 않는다는 것을 나타냅니다.

예를 들어, 포트 80에 대한 HTTP 권고자의 간격을 3초로 설정하려면, 다음 명령을 입력하십시오.

```
ndcontrol advisor interval http 80 3
```

관리 프로그램 간격보다 적은 권고자 간격을 지정할 수 없습니다. 기본 권고자 간격은 7초입니다.

권고자 보고서 시간 종료

로드 밸런스 결정 시 관리 프로그램에서 이전 정보를 사용하지 않도록 하기 위해, 시간 소인이 권고자 보고서 시간 종료에 설정된 시간보다 이전인 권고자 정보를 사용하지 않습니다. 권고자 보고서 시간 종료는 권고자 폴링 간격보다 커야 합니다. 제한 시간이 더 적으면, 관리 프로그램에서는 논리적으로 사용해야 하는 보고서를 무시합니다. 기본적으로 권고자 보고서는 종료 시간이 없습니다. 기본값은 무제한입니다.

예를 들어, 포트 80에 대한 HTTP 권고자의 권고자 보고서 제한 시간을 30초로 설정하려면, 다음 명령을 입력하십시오.

```
ndcontrol advisor timeout http 80 30
```

권고자 보고서 시간 종료 설정에 대한 자세한 정보는 280 페이지의 『ndcontrol advisor -- 권고자 제어』를 참조하십시오.

권고자 연결 시간 종료 및 서버의 수신 시간 종료

Network Dispatcher의 경우, 권고자가 서버 실패를 발견하는 권고자 시간 종료 값을 설정할 수 있습니다. 실패한 서버 시간 종료 값(connecttimeout 및 receivetimeout)은 권고자가 연결이나 수신에 실패했음을 보고하기 전에 대기하는 시간을 결정합니다.

가장 빨리 실패한 서버를 발견하려면 권고자 연결 및 수신 제한 시간을 최소값(1초)으로 설정하고 권고자 및 관리 프로그램 간격 시간을 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, 권고자가 사용 중인 서버를 실패로 표시할 수 있습니다.

예를 들어, 포트 80에 있는 HTTP 권고자에 대해 connecttimeout 및 receivetimeout을 9초로 설정하려면, 다음 명령을 입력하십시오.

```
ndcontrol advisor connecttimeout http 80 9
ndcontrol advisor receivetimeout http 80 9
```

연결 및 수신 제한 시간의 기본값은 권고자 간격 시간에 지정된 값의 세 배입니다.

권고자 목록

- **HTTP** 권고자는 연결을 열어 기본적으로 HEAD 요청을 전송하고 응답 연결을 기다린 후 로드로서 경과 시간을 리턴합니다. HTTP 권고자가 전송한 요청의 유형을 변경하는 방법에 대한 자세한 정보는 174 페이지의 『HTTP 권고자 요청/응답(URL) 옵션』을 참조하십시오.
- **FTP** 권고자는 연결을 열어 SYST 요청을 전송하고 응답을 기다리며 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **Telnet** 권고자는 연결을 열고 서버의 초기 메시지를 기다리며 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **NNTP** 권고자는 연결을 열어 서버의 초기 메시지를 기다리며 중지 명령을 전송하고 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **IMAP** 권고자는 연결을 열어 서버의 초기 메시지를 기다리며 중지 명령을 전송하고 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **POP3** 권고자는 연결을 열어 서버의 초기 메시지를 기다리며 중지 명령을 전송하고 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **SMTP** 권고자는 연결을 열어 서버의 초기 메시지를 기다리며 중지 명령을 전송하고 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.

- **SSL** 권고자는 연결을 열어 CLIENT HELLO 요청을 전송하고 응답을 기다리며 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.

주: SSL 권고자는 키 관리 또는 인증에 종속되지 않습니다.

- **ssl2http** 권고자가 시작하여 포트 443에 나열된 서버에서 권고하지만, HTTP 요청에 대해 “mapport”에 소켓을 엽니다. 클라이언트 대 프록시 프로토콜이 SSL 이고 프로시 대 서버 프로토콜이 HTTP인 경우, CBR용 ssl2http 권고자만 사용 가능합니다. 자세한 정보는 92 페이지의 『SSL의 클라이언트-투-프록시 및 HTTP의 프록시-투-서버의 로드 밸런스』를 참조하십시오.
- **Caching Proxy(ibmproxy)** 권고자는 연결을 열어 Caching Proxy 고유의 HTTP GET 요청을 전송한 후 Caching Proxy 로드로서 응답을 해석합니다.

주: ibmproxy 권고자를 사용할 때 Caching Proxy는 로드 밸런스 중인 모든 서버에서 실행되어야 합니다. Network Dispatcher가 로드 밸런스 중인 동일한 시스템에 있지 않으면 상주하는 시스템에 Caching Proxy를 설치해야 할 필요는 없습니다.

- **DNS** 권고자는 연결을 열어 DNS에 대한 포인터 조회를 전송하고 응답을 기다리며 연결을 닫은 후 로드로서 경과 시간을 리턴합니다.
- **연결** 권고자는 프로토콜 고유의 데이터를 서버와 교환하지 않습니다. 단순히 서버와의 TCP 연결을 열고 닫는 데 걸리는 시간만 측정합니다. 이 권고자는 TCP를 사용하지만 IBM 제공 또는 조정 권고자를 사용할 수 없는 고급 프로토콜이 있는 서버 응용프로그램에 유용합니다.
- **ping** 권고자는 서버와의 TCP 연결을 열지 않지만 대신, 서버가 ping에 응답하는지 여부를 보고합니다. ping 권고자는 모든 포트에서 사용될 수 있지만, 여러 프로토콜 통신량이 이동할 수 있는 와이어 카드 포트를 사용하는 구성을 위해 설계되었습니다. 또한 UDP와 같이 그 서버에 비 TCP 프로토콜을 사용하는 구성에서도 유용합니다.
- **도달** 권고자는 대상 시스템을 ping합니다. 이 권고자는 그 『도달 목표』의 도달 가능성을 판별하기 위해 Dispatcher의 고가용성 구성요소에 맞게 설계되었습니다. 그 결과 고가용성 구성요소로 이동하며 관리 프로그램 보고서에는 표시되지 않습니다. 다른 권고자와 달리 도달 권고자는 Dispatcher 구성요소의 관리 프로그램 기능에 의해 자동으로 시작됩니다.

- **DB2** 권고자는 DB2 서버와 함께 작동됩니다. Dispatcher에는 고객이 고유의 조정 권고자를 기록할 필요없이 DB2 서버의 상태를 확인할 수 있는 기능이 있습니다. DB2 권고자는 DB2 연결 포트와만 통신합니다. Java 연결 포트와는 통신하지 않습니다.
- **WLM(작업로드 관리 프로그램)** 권고자는 MVS 작업로드 관리 프로그램(WLM) 구성요소를 실행하는 OS/390 메인프레임의 서버와 함께 작동되도록 설계되었습니다. 자세한 정보는 168 페이지의 『작업로드 관리 프로그램 권고자』를 참조하십시오.
- **자가 권고자**는 백엔드 서버에서 로드 상태 정보를 수집합니다. Dispatcher가 자가 권고자에서 상단부 Network Dispatcher로 정보를 공급하는 상하단부 구성의 Dispatcher를 사용할 때 자가 권고자를 사용할 수 있습니다. 자가 권고자는 특히 실행 프로그램 레벨에서 Dispatcher의 백엔드 서버에서 초당 연결 수를 측정합니다. 186 페이지의 『상하단부 WAND 구성에서 자가 권고자 사용』에서 자세한 정보를 참조하십시오.
- Dispatcher는 고객이 조정(사용자 정의 가능) 권고자를 작성할 수 있는 기능을 제공합니다. 이로써 IBM이 특정 권고자를 개발하지 않은 고유 프로토콜(TCP의 맨 위에 있는)의 지원이 가능해졌습니다. 자세한 정보는 『조정(사용자 정의 가능) 권고자 작성』을 참조하십시오.
- **WAS(WebSphere Application Server)** 권고자는 WebSphere Application Server와 함께 작동됩니다. 이 권고자에 대한 사용자 정의 가능 예제 파일은 설치 디렉토리에 제공됩니다. 자세한 정보는 165 페이지의 『WAS(WebSphere Application Server) 권고자』를 참조하십시오.

조정(사용자 정의 가능) 권고자 작성

조정(사용자 정의 가능) 권고자는 기본 코드에서 호출되어 클래스 파일로서 제공하는 Java 코드의 작은 부분입니다. 기본 코드는 조정 권고자의 인스턴스 시작 및 정지, 상태 및 보고서 제공, 로그 파일에 히스토리 정보 기록과 같은 모든 관리 서비스를 제공합니다. 또한 관리 프로그램 구성요소로 결과도 보고합니다. 정기적으로 기본 코드는 개별적으로 그 구성의 모든 서버를 평가하는 권고자 주기를 수행합니다. 서버 시스템과의 연결을 열어 시작합니다. 소켓이 열리면, 기본 코드는 조정 권고자의 『getLoad』 메소드(함수)를 호출합니다. 그러면 조정 권고자는 서버의

상태를 평가하는 데 필요한 모든 단계를 수행합니다. 일반적으로 서버에 사용자 정의된 메시지를 전송한 후 응답을 기다립니다(조정 권고자에게 열린 소켓에 대한 액세스가 제공됩니다). 그러면 기본 코드는 서버와 함께 소켓을 닫고 로드 정보를 관리 프로그램에 보고합니다.

기본 코드 및 조정 권고자는 표준 또는 대체 모드에서 작동될 수 있습니다. 조작 모드의 선택사항은 constructor 메소드의 매개변수로서 조정 권고자 파일에 지정됩니다.

표준 모드에서 조정 권고자는 서버와 데이터를 교환하며, 기본 권고자 코드는 교환 시간을 정한 후 로그값을 계산합니다. 그런 후 기본 코드는 이 로그값을 관리 프로그램에 보고합니다. 조정 권고자는 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. 조정 권고자 코드는 그 고유 요구사항에 필요한 모든 조작을 수행한 후 실제 로드 번호를 리턴합니다. 기본 코드는 번호를 승인하고 이를 관리 프로그램에 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 권고자를 작성할 수 있습니다. 조정 권고자의 한 예로 **ADV_sample.java**는 Network Dispatcher 제품과 함께 제공됩니다. Network Dispatcher를 설치하면 **...nd/servers/samples/CustomAdvisors** 설치 디렉토리에서 예제 코드를 찾을 수 있습니다.

기본 설치 디렉토리는 다음과 같습니다.

- AIX: /usr/lpp/nd
- Linux: /opt/nd
- Sun: /opt/nd
- Windows 2000: c:\Program Files\IBM\nd

WAS(WebSphere Application Server) 권고자

WebSphere Application Server 권고자에 대한 특수한 예제 조정 권고자 파일이 Network Dispatcher 설치 디렉토리에서 제공됩니다.

- ADV_was.java는 Network Dispatcher 시스템에서 컴파일되어 실행될 파일입니다.
- NDAdvisor.java.servlet(NDAdvisor.java로 이름이 변경될 것임)은 WebSphere Application Server 시스템에서 컴파일되어 실행될 파일입니다.

WebSphere Application Server 권고자 예제 파일은 ADV_sample.java 파일과 동일한 예제 디렉토리에 있습니다.

이름 지정 규칙

조정 권고자 파일 이름은 『ADV_myadvisor.java』 양식이어야 합니다. 이 이름은 대문자 『ADV_』 접두부로 시작해야 합니다. 모든 후속 문자는 소문자여야 합니다.

Java 규칙에서와 마찬가지로 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

조정 권고자는 Java 언어로 작성됩니다. 시스템용 Java 1.3 컴파일러를 설치해야 합니다. 이들 파일은 컴파일 시 참조됩니다.

- 조정 권고자 파일
- Network Dispatcher가 설치된 **...nd/servers/lib** 디렉토리에 있는 기본 클래스 파일 **ibmnd.jar**.

classpath는 컴파일 시 조정 권고자 파일 및 기본 클래스 파일을 모두 연결해야 합니다.

Windows 2000의 경우, 컴파일 명령은 다음과 같습니다.

```
javac -classpath <install_dir>\nd\servers\lib\ibmnd.jar ADV_fred.java
```

여기서:

- 사용자 권고자 파일의 이름은 ADV_fred.java입니다.
- 사용자 권고자 파일은 현재 디렉토리에 저장됩니다.

컴파일 출력은 클래스 파일입니다. 예를 들어, 다음과 같습니다.

```
ADV_fred.class
```

권고자를 시작하기 전에, Network Dispatcher가 설치된 **...nd/servers/lib/CustomAdvisors** 디렉토리에 클래스 파일을 복사하십시오.

주: 필요에 따라 조정 권고자는 한 운영 체제에서 컴파일되어 다른 운영 체제에서 실행될 수 있습니다. 예를 들어, Windows 2000에서 권고자를 컴파일하고 AIX 시스템에 클래스 파일(2진)을 복사하여 여기에서 조정 권고자를 실행할 수 있습니다.

AIX, Linux 및 Sun의 경우, 구문이 유사합니다.

실행

조정 권고자를 실행하려면, 먼저 클래스 파일을 적당한 Network Dispatcher 하위 디렉토리에 복사해야 합니다.

```
.../nd/servers/lib/CustomAdvisors/ADV_fred.class
```

Dispatcher를 구성하고, 그 관리 프로그램 기능을 시작한 후 조정 권고자를 시작할 명령을 실행하십시오.

```
ndcontrol advisor start fred 123
```

여기서:

- fred는 ADV_fred.java의 경우와 같은 권고자 이름입니다.
- 123은 권고자가 작동될 포트입니다.

필수 루틴

모든 권고자와 마찬가지로, 조정 권고자는 권고자 기본 기능인 ADV_Base를 확장합니다. 이 기능은 Dispatcher 가중치 알고리즘에서 사용하기 위해 로드를 다시 관리 프로그램에 보고하는 작업과 같이 대부분의 권고자 기능을 실제로 수행하는 권고자 기본 기능입니다. 권고자 기본 기능은 소켓 연결 및 닫기 조작을 수행하며

권고자가 사용할 송수신 메소드도 제공합니다. 권고자 자체는 권고 중인 서버의 포트간에 데이터 송수신에만 사용됩니다. 권고자 기본 기능 내의 TCP 메소드는 로드를 계산하기 위해 시간 설정됩니다. ADV_base에서 구성자 내의 플래그는 필요에 따라 기존의 로드 위에 권고자로부터 리턴된 새로운 로드를 겹쳐씁니다.

주: 구성자의 일련의 값에 따라, 권고자 기본 기능은 지정된 간격으로 가중치 알고리즘에 로드를 제공합니다. 실제 권고자가 유효한 로드를 리턴할 수 있도록 완료되지 않은 경우, 권고자 기본 기능에서는 이전의 로드를 사용합니다.

다음은 기본 클래스 메소드입니다.

- **construct** 루틴. constructor는 기본 클래스 구성자를 호출합니다(예제 권고자 파일 참조).
- **ADV_AdvisorInitialize** 메소드. 이 메소드는 기본 클래스가 초기화를 완료한 후 추가 단계가 수행되어야 할 경우에 후크를 제공합니다.
- **getload** 루틴. 기본 권고자 클래스는 소켓 열기를 수행하므로, getload는 적절한 송수신 요청을 발행하여 권고 주기를 완료해야 합니다.

탐색 순서

먼저 Network Dispatcher는 제공되는 고유한 권고자 목록을 검토합니다. 이 목록에 정해진 권고자가 없으면 고객의 조정 권고자 목록을 검토합니다.

이름 지정 및 경로

- 조정 권고자 클래스는 Network Dispatcher 기본 디렉토리의 `...nd/servers/lib/CustomAdvisors/` 하위 디렉토리 내에 있어야 합니다. 이 디렉토리의 기본값은 운영 체제에 따라 다양합니다.
 - AIX
`/usr/lpp/nd/servers/lib/CustomAdvisors/`
 - Linux
`/opt/nd/servers/lib/CustomAdvisors/`
 - Solaris
`/opt/nd/servers/lib/CustomAdvisors/`
 - Windows 2000

일반 설치 디렉토리 경로

C:\Program Files\IBM\edge\nd\servers\lib\CustomAdvisors

고유(Native) 설치 디렉토리 경로

C:\Program Files\IBM\nd\servers\lib\CustomAdvisors

- 소문자의 영문자만 허용됩니다. 따라서 조작원은 명령행에 명령을 입력할 때 대 소문자를 구별할 필요가 없습니다. 권고자 이름 앞에는 **ADV_**가 와야 합니다.

예제 권고자

예제 권고자의 프로그램 목록이 421 페이지의 『권고자 예제』에 나와 있습니다. 설치 후에 이 예제 권고자는 `...nd/servers/samples/CustomAdvisors` 디렉토리에서 찾을 수 있습니다.

작업로드 관리 프로그램 권고자

WLM은 MVS 메인프레임에서 실행되는 코드입니다. MVS 시스템에 있는 로드 에 대해 문의할 때 조회할 수 있습니다.

MVS 작업로드 관리가 OS/390 시스템에서 구성되었으면, Dispatcher는 WLM에서 용량 정보를 받아들여 로드 밸런스 프로세스에 이 정보를 사용할 수 있습니다. WLM 권고자를 사용하여 Dispatcher는 정기적으로 Dispatcher에 있는 각 서버의 WLM 포트를 통한 연결을 열어 리턴된 용량 정수를 받아들입니다. 이러한 정수는 여전히 사용 가능한 용량을 나타내고 Dispatcher는 각 시스템의 로드를 나타내는 값을 기대하므로, 용량 정수는 권고자에 의해 반전되며 로드값으로 표준화됩니다(예: 대용량 정수이지만 작은 로드값은 모두 더 완전한 서버를 나타냅니다). 결과 로드는 관리 프로그램 보고서의 시스템 컬럼에 저장됩니다.

WLM 권고자와 기타 Dispatcher 권고자 사이의 중요한 몇 가지 차이점은 다음과 같습니다.

1. 기타 권고자는 표준 클라이언트 통신량이 이동되는 동일한 포트를 사용하여 서버에 대한 연결을 엽니다. WLM 권고자는 표준 통신량과는 다른 포트를 사용하여 서버에 대한 연결을 엽니다. 각 서버 시스템의 WLM 에이전트는 Dispatcher WLM 권고자가 시작되는 동일한 포트에서 인식되도록 구성되어야 합니다. 기본 WLM 포트는 10007입니다.

2. 기타 권고자는 서버 포트가 권고자의 포트와 일치하는 Dispatcher cluster:port:server 구성에서 정의된 서버만 평가합니다. WLM 권고자는 Dispatcher cluster:port:server 구성의 모든 서버에 대해 권고합니다. 따라서 WLM 권고자 사용시 비 WLM 서버를 정의하지 않아도 됩니다.
3. 기타 권고자는 해당 로드 정보를 그 『포트』 컬럼 아래의 관리 프로그램 보고서에 저장합니다. WLM 권고자는 해당 로드 정보를 그 시스템 아래의 관리 프로그램 보고서에 저장합니다.
4. WLM 권고자와 함께 프로토콜 고유 권고자를 모두 사용할 수 있습니다. 프로토콜 고유 권고자는 그 표준 통신 포트에서 서버를 폴링하고 WLM 권고자는 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 관리 프로그램 보고서의 시스템 컬럼에 저장합니다. 결과적으로 WLM 권고자와 Metric Server를 동시에 실행할 수 없습니다.

Metric Server

이 기능은 모든 Network Dispatcher 구성요소에 사용할 수 있습니다.

Metric Server는 시스템 고유의 메트릭 양식으로 Network Dispatcher에 서버 로드 정보를 제공하고 서버의 상태를 보고합니다. Network Dispatcher 관리 프로그램은 각 서버에 있는 Metric Server를 조회하여 에이전트에서 수집한 메트릭을 사용하여 로드 밸런스 프로세스에 가중치를 지정합니다. 결과도 관리 프로그램 보고서에 저장됩니다.

주: 각 서버에 대해 두 개 이상의 메트릭이 수집되어 단일 시스템 로드값으로 표준화될 때 반올림 오류가 발생할 수 있습니다.

49 페이지의 그림11에서 구성 예제를 참조하십시오.

WLM 제한사항

WLM 권고자와 마찬가지로 Metric Server는 개별적인 프로토콜 고유 서버 디먼
에서가 아니라 서버 시스템에서 전체로서 보고합니다. WLM 및 Metric Server 모
두는 결과를 관리 프로그램 보고서의 시스템 컬럼에 저장합니다. 결과적으로 WLM
권고자와 Metric Server를 동시에 실행할 수 없습니다.

전제조건

Metric Server 에이전트는 Network Dispatcher가 로드 밸런스 중인 서버에서 설
치 및 실행되어야 합니다.

Metric Server 사용 방법

다음은 Dispatcher에 대해 Metric Server를 구성하는 단계입니다. Network
Dispatcher의 다른 구성요소에 대해 Metric Server를 구성할 때도 이와 비슷한 단
계를 사용합니다.

- Network Dispatcher 관리 프로그램(Network Dispatcher)

1. **ndserver**를 시작하십시오.

2. 명령 발행: **ndcontrol manager start *manager.log port***

*port*는 모든 Metric Server 에이전트를 실행하기 위해 선택된 RMI 포트입
니다. `metricserver.cmd` 파일에 설정된 기본 RMI 포트는 10004입니다.

3. 명령 발행: **ndcontrol 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 메
트릭 스크립트를 작성할 수 있습니다. 이 스크립트에는 0-100 범위의 숫자
값을 리턴해야 하는 명령이 들어 있습니다. 이 숫자 값은 사용가능성 값이
아니라 로드 측정값을 나타내야 합니다.

주: Site Selector의 경우, cpuload 및 memload가 자동으로 실행됩니다.

제한사항: Windows 2000의 경우, 시스템 메트릭 스크립트 이름에 ".exe"
이외의 확장자가 있으면 파일의 전체 이름을 지정해야 합니다
(예: "mysystemscript.bat"). 이것은 Java 제한사항 때문입니다.

4. `metricsserver.cmd` 파일에 지정된 포트에서 실행하는 Metric Server 에이전트를 포함하는 서버만 구성에 추가하십시오. 포트는 **manager start** 명령에 지정된 포트 값과 같아야 합니다.

주: 보안 확인 -

- Network Dispatcher 시스템에서 실행 중인 구성요소에 대해 키 파일을 작성하십시오(**ndkeys create** 명령 사용). `ndkeys`에 대한 자세한 정보는 229 페이지의 『원격 인증 관리』를 참조하십시오.
- 서버 시스템에서 결과 키 파일을 `.../nd/admin/key` 디렉토리에 복사하십시오. 키 파일의 사용 권한이 루트에서 파일을 읽을 수 있도록 허용하는지 확인하십시오.

• Metric Server 에이전트(서버 시스템)

1. Network Dispatcher 설치에서 Metric Server 패키지를 설치하십시오.
2. `/usr/bin` 디렉토리에서 **metricsserver** 스크립트를 확인하여 원하는 RMI 포트가 사용되고 있는지 확인하십시오(Windows 2000의 경우, 디렉토리가 `C:\WINNT\SYSTEM32`입니다). 기본 RMI 포트는 10004입니다.

주: 지정된 RMI 포트 값은 Network Dispatcher 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 `...nd/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 `...nd/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 UNIX 기반 플랫폼의 경우, 스크립트가 셸 선언으로 시작되어야 합니다. 그렇지 않으면 올바르게 실행되지 않습니다.

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" 뒤에 다음을 삽입하십시오.

```
-Djava.rmi.server.hostname=OTHER_ADDRESS
```

또한 `metricsserver` 파일에서 "if" 명령문 앞에 다음 행, 즉 `hostname OTHER_ADDRESS`를 추가하십시오.

Windows 2000에서는 Microsoft 스택에서 `OTHER_ADDRESS`의 별명을 지정해야 합니다. Microsoft 스택에서 주소의 별명을 지정하려면 195 페이지를 참조하십시오.

서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)

Network Dispatcher 구성에서 서버를 정의할 때, 전체 서버의 상태(Metric Server 에이전트 사용) 및/또는 특정 포트 응용프로그램의 상태(권고자 기능 사용)에 따라 로드를 분산시킬 수 있습니다.

서버를 분할하면 특정 URL과 고유 응용프로그램을 판별할 수 있습니다. 예를 들어, 하나의 웹 서버에서 JSP 페이지, HTML 페이지, 데이터베이스 요청 등을 제공할 수 있습니다. 이제 Network Dispatcher는 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, Network Dispatcher에서 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 작업로드를 기반으로 로드를 분산시킬 수 있습니다.

Network Dispatcher 구성에서 *cluster:port:server* 계층을 사용하여 물리적 서버 또는 논리 서버를 표시할 수 있습니다. 서버는 기호 이름이나 점분리 10진수 형식으로 된 시스템(물리적 서버)의 고유한 IP 주소입니다. 또는 분할된 서버를 나타내도록 서버를 구성할 경우, **ndcontrol server add** 명령의 **address** 매개변수에서 물리적 서버에 대한 해석 가능한 서버 주소를 제공해야 합니다. 335 페이지의 『ndcontrol server -- 서버 구성』에서 자세한 정보를 참조하십시오.

아래는 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
규칙 1: \*.htm
    서버: A
    서버: C
규칙 2: \*.jsp
    서버: D
    서버: F
규칙 3: \*.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(gif 요청 처리) 및 F(jsp 요청 처리)로 분할됩니다.

주: SDA에서는 구성에서 검색 기능에 대해 서버 주소가 고유해야 하므로 Server Directed Affinity가 서버 파티션 기능과 함께 작동하지 않는 제한사항이 있

습니다. 213 페이지의 『클라이언트 서버 유사성을 제어하는 서버 지정 유사 API』에서 자세한 정보를 참조하십시오.

HTTP 권고자 요청/응답(URL) 옵션

HTTP 권고자의 URL 옵션은 Dispatcher 및 CBR 구성요소에 대해 사용 가능합니다.

HTTP 권고자를 시작한 후, 서버에 조회할 서비스 고유의 클라이언트 HTTP URL 문자열을 정의할 수 있습니다. 이 기능으로 HTTP 권고자는 서버 내의 개별 서비스의 상태를 평가할 수 있습니다. 동일한 물리적 IP 주소를 가진 고유의 서버 이름으로 논리 서버를 정의하여 이 기능을 수행할 수 있습니다. 172 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 권고자는 **advisorrequest** 문자열을 사용하여 서버의 상태를 조회합니다. 기본값은 HEAD / HTTP/1.0입니다. **advisorresponse** 문자열은 HTTP 권고자가 HTTP 응답에서 스캔하는 권고자 응답입니다. HTTP 권고자는 **advisorresponse** 문자열을 사용하여 서버에서 수신한 실제 응답과 비교합니다. 기본값은 null입니다.

중요: HTTP URL 문자열에 공백이 포함된 경우에는 다음에 주의하십시오.

- **ndcontrol>>** 셸 프롬프트에서 명령을 발행할 경우, 공백이 문자열에 포함되어 있으면 문자열을 따옴표(')로 묶어야 합니다.

예제:

```
server set cluster:port:server advisorrequest "head / http/2.0"  
server set cluster:port:server advisorresponse "HTTP 200 OK"
```

- 운영 체제 프롬프트에서 **ndcontrol** 명령을 발행할 경우, 텍스트 앞에 "\"를, 텍스트 뒤에 "\"를 표시해야 합니다.

예제:

```
ndcontrol server set cluster:port:server advisorrequest "\"head / http/2.0\""  
ndcontrol server set cluster:port:server advisorresponse "\"HTTP 200 OK\""
```

주: 지정된 HTTP 포트 번호에 대해 HTTP 권고자를 시작한 후, 권고자 요청/응답 값은 해당 HTTP 포트 번호에 대해 사용 가능해집니다.
335 페이지의 『ndcontrol server -- 서버 구성』에서 자세한 정보를 참조하십시오.

결합 배치된 서버 사용

Network Dispatcher는 요청의 로드 밸런스가 유지되는 서버와 동일한 시스템에 상주할 수 있습니다. 이것을 보통 서버 결합 배치라고 합니다. 결합 배치는 Dispatcher, Site Selector, Mailbox Locator 및 Cisco Consultant 구성요소에 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 Network Dispatcher와 자원을 차지하기 위해 경쟁합니다. 그러나 과부하된 시스템이 없을 때 결합 배치된 서버를 사용하면 로드 밸런스된 사이트를 설정하는 데 필요한 총 시스템 수가 줄어듭니다.

Dispatcher 구성요소의 경우

Red Hat Linux v7.1(Linux 커널 버전 2.4.2-2) 또는 SuSE Linux v7.1(Linux 커널 버전 2.4.0-4GB): mac 전달 방법을 사용하여 Dispatcher 구성요소를 실행할 때 결합 배치와 고가용성을 동시에 구성하려면 Linux 커널 패치를 설치해야 합니다. 패치 설치에 대한 자세한 정보는 83 페이지의 『Linux 커널 패치 설치(루프백 인터페이스에서 arp 응답 억제)』를 참조하십시오. 그러나 이 명령을 따를 때 루프백 어댑터의 별명을 지정하는 단계를 건너뛰십시오. ifconfig 명령을 추가하여 Dispatcher가 대기 상태가 될 때 실행되는 goStandby 고가용성 스크립트 파일의 루프백 어댑터의 별명을 지정해야 합니다.

Solaris: 진입점 Dispatcher가 결합 배치될 때 WAND 권고자를 구성할 수 없는 제한사항이 있습니다. 179 페이지의 『광역 지원으로 원격 권고자 사용』을 참조하십시오.

이전 버전에서는 결합 배치된 서버의 주소를 구성의 비전달 주소(NFA)와 동일하게 지정해야 했습니다. 이러한 제한은 해결되었습니다.

서버를 결합 배치되도록 구성하려면, **ndcontrol server** 명령은 **collocated**라는 옵션을 제공합니다. 이 옵션은 *yes* 또는 *no*로 설정할 수 있습니다. 기본값은 *no*입니다. 서버의 주소는 시스템에 있는 네트워크 인터페이스 카드의 유효한 IP 주소여야 합니다.

주: Windows 2000: Dispatcher를 결합 배치할 수 있지만 결합 배치된 키워드를 사용하지 마십시오. Dispatcher의 nat 및 cbr 전달 방법을 사용할 때는 결합 배치가 지원되지만, Dispatcher의 mac 전달 방법을 사용할 때는 결합 배치가 지원되지 않습니다. Dispatcher의 전달 방법에 대한 자세한 정보는 61 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 방법)』, 63 페이지의 『Dispatcher content-based routing(cbr 전달 방법)』 및 60 페이지의 『Dispatcher의 MAC 레벨 경로 지정(MAC 전달 방법)』을 참조하십시오.

다음 방법 중 하나로 결합 배치된 서버를 구성할 수 있습니다.

- NFA를 결합 배치된 서버 주소로 사용하는 경우, **ndcontrol executor set nfa IP_address** 명령을 사용하여 NFA를 설정하십시오. 그리고 **ndcontrol server add cluster:port:server** 명령과 함께 NFA 주소를 사용하여 서버를 추가하십시오.
- NFA 이외의 주소를 사용하는 경우, 다음과 같이 *yes*로 설정된 **collocated** 매개변수를 사용하여 원하는 IP 주소가 있는 서버를 추가하십시오. **ndcontrol server add cluster:port:server collocated yes**

ndcontrol server 명령 구문에 대한 자세한 정보는 335 페이지의 『ndcontrol server -- 서버 구성』을 참조하십시오.

CBR 구성요소

CBR은 추가로 구성할 필요없이 모든 플랫폼에서 결합 배치를 지원합니다. 그러나 사용하는 웹 서버와 Caching Proxy는 특정 바인드이어야 합니다.

Mailbox Locator 구성요소의 경우

Mailbox Locator는 모든 플랫폼에서 결합 배치를 지원합니다. 그러나 서버는 결합 배치가 작동하도록 Network Dispatcher 이외의 주소에 바인드되어야 합니다.

POP3 또는 IMAP 서버를 동일한 시스템에 결합 배치하려면 서버가 클러스터 주소와 다른 IP 주소에 바인드되어야 합니다. 이것은 루프백 주소를 사용하여 완료할 수 있습니다.

Site Selector 구성요소의 경우

Site Selector는 추가로 구성할 필요없이 모든 플랫폼에서 결합 배치를 지원합니다.

Cisco Consultant 구성요소의 경우

Cisco Consultant는 추가로 구성할 필요없이 모든 플랫폼에서 결합 배치를 지원합니다.

광역 Dispatcher 지원 구성

Dispatcher 구성요소의 경우에만 이 기능을 사용할 수 있습니다.

Dispatcher의 광역 지원과 nat 전달 방법을 사용하지 않는 경우, Dispatcher 시스템 및 그 서버가 모두 동일한 LAN 세그먼트에 접속되는 Dispatcher 구성이 요구됩니다(그림22 참조). 클라이언트의 패킷이 ND 시스템으로 가서 서버로 전송된 다음, 서버에서 다시 클라이언트로 직접 전송됩니다.

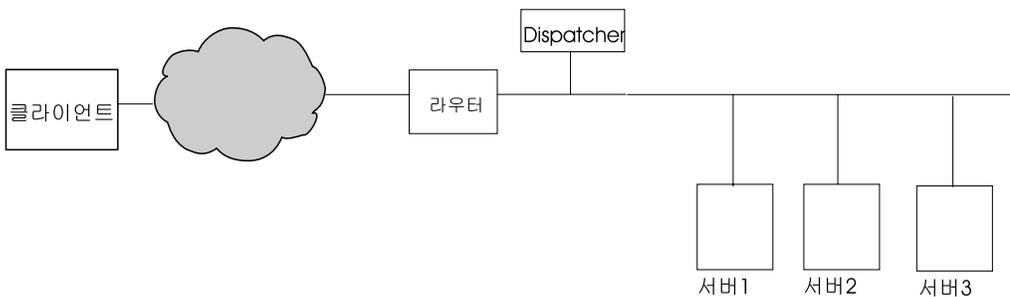


그림 22. 단일 LAN 세그먼트를 구성하는 구성 예제

광역 Dispatcher 확장은 원격 서버로 알려진 오프사이트 서버에 대한 지원을 추가합니다(178 페이지의 그림23 참조). GRE가 원격 사이트에서 지원되지 않고 Dispatcher의 nat 전달 방법을 사용하지 않는 경우, 원격 사이트는 반드시 원격

Dispatcher 시스템(Dispatcher 2)과 이 시스템에 로컬로 접속된 서버(ServerG, ServerH 및 ServerI)로 구성되어야 합니다. 모든 Dispatcher 시스템은 같은 운영 체제를 사용해야 합니다. 이제 클라이언트 패킷은 인터넷에서 Dispatcher 시스템으로 이동해서 여기에서 지역적으로 떨어져 있는 원격 Dispatcher 시스템으로, 다음엔 그 지역 내에 접속된 서버 중 하나로 이동합니다.

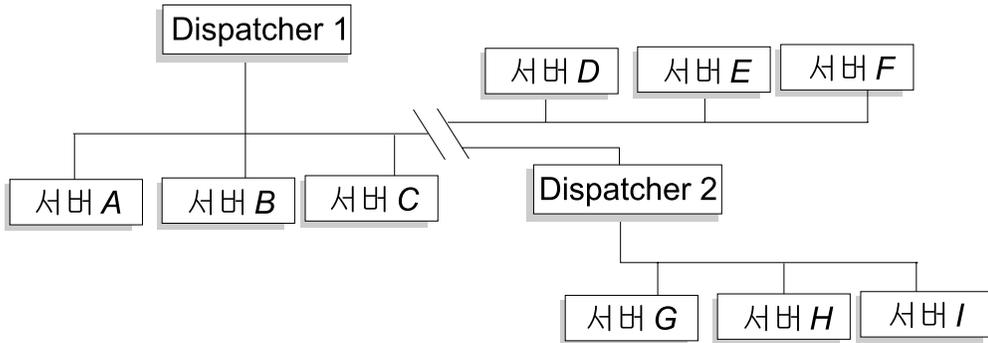


그림 23. 로컬 및 원격 서버를 사용하는 구성 예제

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 명령은 복잡하지 않습니다. 광역 지원을 구성하려면 다음과 같이 하십시오.

1. 서버를 추가하십시오. Dispatcher에 서버를 추가할 경우, 서버가 로컬 또는 원격인지 정의해야 합니다(위 참조). 서버를 추가하고 로컬으로 정의하려면, 라우터를 지정하지 않고 **ndcontrol 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, 182 페이지의 그림24에서 ND 2를 ND 1의 원격 서버로 추가하는 경우 *router 1*을 라우터 주소로 정의해야 합니다. 일반 구문:

```
ndcontrol server add cluster:port:server router address
```

라우터 키워드에 대한 자세한 정보는 335 페이지의 『ndcontrol server -- 서버 구성』을 참조하십시오.

2. 별명을 구성하십시오. 첫 번째 Dispatcher 시스템(인터넷에서 클라이언트 요청이 도착하는 위치)에서, 클러스터 주소는 이전과 마찬가지로 **cluster configure**, **ifconfig** 또는 **ndconfig**를 사용하여 별명이 지정되어야 합니다. 그러나 원격 Dispatcher 시스템에서는 네트워크 인터페이스 카드에 대해 클러스터 주소의 별명이 지정되지 않습니다.

광역 지원으로 원격 권고자 사용

진입점 Dispatcher에서 권고자는 특수한 구성 없이 대부분의 플랫폼에 대해 올바르게 작동됩니다.

Linux: 광역 지원 구성으로 원격 권고자를 사용하는 데에는 제한사항이 있습니다. HTTP 권고자와 같이 진입점 Dispatcher 시스템에서 실행되는 특정 프로토콜 권고자는 원격 사이트에서 서버 시스템의 상태를 정확히 판별하지 않습니다. 이 문제점을 해결하려면 다음 중 하나를 수행하십시오.

- 진입점 Dispatcher 시스템에서 프로토콜 독립 ping 권고자를 실행하십시오.
- 진입점 Dispatcher 시스템의 특정 프로토콜 권고자를 이에 일치하는 원격 Dispatcher 시스템의 특정 프로토콜 서버 디먼(예: 웹 서버)과 함께 실행하십시오.

이 옵션 중 하나는 진입점 Dispatcher 시스템에서 실행되는 권고자에게 원격 Dispatcher 시스템 상태 정보를 제공합니다.

Solaris: 진입점 Network Dispatcher에서 ifconfig 또는 cluster 구성 방법 대신 arp 구성 방법을 사용해야 합니다.

예제:

```
arp -s <my_cluster_address> <my_mac_address> pub
```

주: 다음과 같은 Solaris 제한사항이 있습니다.

- WAND 권고자는 클러스터 구성의 arp 방법으로만 작동됩니다.
- 특정 바인드 서버의 권고자는 클러스터 구성의 arp 방법으로만 작동됩니다.

- 결합 배치는 클러스터 구성의 `ifconfig` 방법으로만 작동됩니다.

원격 Dispatcher에서는 원격 클러스터 주소마다 다음 구성 단계를 수행해야 합니다. 원격 Network Dispatcher 위치에 있는 고가용성 구성의 경우, 사용자는 두 시스템 모두에서 다음의 단계를 수행해야 합니다.

AIX

- 클러스터 주소를 루프백 어댑터의 별명으로 지정합니다. 넷마스크 값은 255.255.255.255로 설정해야 합니다.

예제:

```
ifconfig lo0 alias 9.67.34.123 netmask 255.255.255.255
```

주: 로컬 및 원격 Dispatcher 시스템 모두에서 권고자가 실행되어야 합니다.

Linux

- 클러스터 주소를 루프백 어댑터의 별명으로 지정합니다.

예제:

```
ifconfig lo:1 9.67.34.123 netmask 255.255.255.255 up
```

주: 로컬 및 원격 Dispatcher 시스템 모두에서 권고자가 실행되어야 합니다.

Solaris

- 추가 구성 단계는 필요하지 않습니다.

Windows 2000

1. Dispatcher에는 두 개의 IP 주소가 필요합니다. Microsoft TCP/IP 스택용 및 Network Dispatcher 스택용입니다. Network Dispatcher 스택의 IP 주소를 사용하여 NFA를 구성하십시오.

예제:

```
ndconfig en0 alias 9.55.30.45 netmask 255.255.240.0
```

2. 원격 클러스터 주소를 별명으로 사용하여 루프백 어댑터를 구성하십시오. 넷마스크 값은 255.255.255.255로 설정해야 합니다.

예제:

```
ndconfig lo0 alias 9.67.34.123 netmask 255.255.255.255
```

3. 원격 클러스터 주소에 대한 arp 테이블에 있는 모든 항목을 삭제하십시오.
 - a. arp 테이블의 목차를 보려면, 다음을 입력하십시오.

arp -a

- b. 항목이 있는 경우 항목을 삭제하려면, 다음을 입력하십시오.

arp -d 9.67.34.123

주: 사용자 인터페이스의 MAC 주소를 결정하려면, 다음을 입력하십시오.

1) **ping** *your_hostname*

2) **arp -a**

그런 후 사용자 시스템의 주소를 찾으십시오.

4. NFA(Network Dispatcher 스택의 IP 주소)를 사용하여 원격 클러스터 (9.67.34.123)에 라우트를 추가하십시오. 넷마스크 값은 255.255.255.255로 설정해야 합니다.

예제:

route add 9.67.34.123 mask 255.255.255.255 9.55.30.45

구성 예제

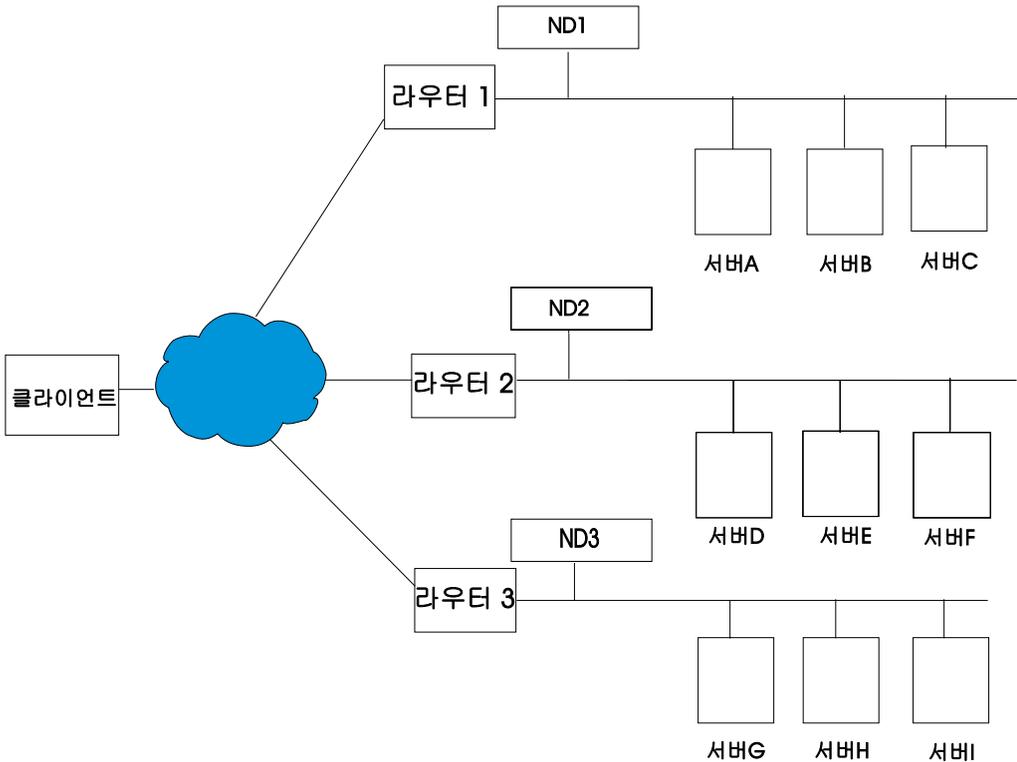


그림 24. 원격 Network Dispatcher의 광역 구성 예제

이 예제는 그림24에 설명된 구성에 적용됩니다.

다음은 포트 80에서 클러스터 주소 xebec를 지원하기 위해 Dispatcher 시스템을 구성하는 방법입니다. ND1이 『진입점』으로 정의됩니다. 이더넷 연결로 가정합니다. ND1에 5개의 서버, 즉 3개의 로컬 서버(ServerA, ServerB, ServerC) 및 2개의 원격 서버(ND2 및 ND3)가 정의되어 있다는 점에 유의하십시오. 원격 ND2 및 ND3에는 각각 3개의 로컬 서버가 정의되어 있습니다.

첫 번째 Dispatcher(ND1)의 콘솔에서 다음을 수행하십시오.

1. 실행 프로그램을 시작합니다.

ndcontrol executor start

2. Dispatcher 시스템의 비전달 주소를 설정하십시오.
ndcontrol executor set nfa ND1
 3. 클러스터를 정의하십시오.
ndcontrol cluster add xebec
 4. 포트를 정의하십시오.
ndcontrol port add xebec:80
 5. 서버를 정의하십시오.
 - a. **ndcontrol server add xebec:80:ServerA**
 - b. **ndcontrol server add xebec:80:ServerB**
 - c. **ndcontrol server add xebec:80:ServerC**
 - d. **ndcontrol server add xebec:80:ND2 router Router1**
 - e. **ndcontrol server add xebec:80:ND3 router Router1**
 6. Windows 2000을 사용하는 경우, Dispatcher LAN 어댑터의 NFA를 구성하십시오.
ndcontrol cluster configure ND1 및 또한 clusteraddr로서 xebec를 구성하십시오.
 7. 클러스터 주소를 구성합니다.
ndcontrol cluster configure xebec
- 두 번째 Dispatcher(ND2)의 콘솔에서 다음을 수행하십시오.
1. 실행 프로그램을 시작합니다.
ndcontrol executor start
 2. Dispatcher 시스템의 비전달 주소를 설정하십시오.
ndcontrol executor set nfa ND2
 3. 클러스터를 정의하십시오.
ndcontrol cluster add xebec
 4. 포트를 정의하십시오.
ndcontrol port add xebec:80
 5. 서버를 정의하십시오.

- a. **ndcontrol server add xebec:80:ServerD**
 - b. **ndcontrol server add xebec:80:ServerE**
 - c. **ndcontrol server add xebec:80:ServerF**
6. Windows 2000을 사용하는 경우, Dispatcher LAN 어댑터의 NFA를 구성하십시오.

ndcontrol cluster configure ND2

세 번째 Dispatcher(ND3)의 콘솔에서 다음을 수행하십시오.

1. 실행 프로그램을 시작합니다.

ndcontrol executor start

2. Dispatcher 시스템의 비전달 주소를 설정하십시오.

ndcontrol executor set nfa ND3

3. 클러스터를 정의하십시오.

ndcontrol cluster add xebec

4. 포트를 정의하십시오.

ndcontrol port add xebec:80

5. 서버를 정의하십시오.

a. **ndcontrol server add xebec:80:ServerG**

b. **ndcontrol server add xebec:80:ServerH**

c. **ndcontrol server add xebec:80:ServerI**

6. Windows 2000을 사용하는 경우, Dispatcher LAN 어댑터의 nfa를 구성하십시오.

ndcontrol cluster configure ND3

주

- 1. 모든 서버(A-I)에서 루프백에 대해 클러스터 주소의 별명을 지정하십시오.
- 2. 클러스터 및 포트는 참여 중인 모든 Dispatcher 시스템(진입점 Dispatcher와 모든 원격지)에서 ndcontrol을 사용하여 추가됩니다.

3. 광역 지원을 통한 원격 권고자 사용에 대한 도움말을 보려면 179 페이지의 『광역 지원으로 원격 권고자 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. 관리 프로그램과 권고자는 광역 지원으로 작동되며, 사용할 경우 참여 중인 모든 Dispatcher 시스템에서 시작되어야 합니다.
8. Network Dispatcher는 비슷한 운영 체제에 대해서만 WAND를 지원합니다.

GRE(일반 경로 지정 캡슐화) 지원

GRE(일반 경로 지정 캡슐화)는 RFC 1701 및 RFC 1702에 지정된 인터넷 프로토콜입니다. Network Dispatcher는 GRE를 사용하여 IP/GRE 패킷 내의 클라이언트 IP 패킷을 캡슐화하여 GRE를 지원하는 OS/390과 같은 서버 플랫폼에 전달할 수 있습니다. GRE 지원을 사용하면 Dispatcher 구성요소는 하나의 MAC 주소에 연관된 여러 서버 주소로 패킷을 로드 밸런싱할 수 있습니다.

Network Dispatcher는 GRE를 WAND(Wide Area Network Dispatcher) 기능의 일부로 구현합니다. 그러면 Network Dispatcher가 GRE 패킷을 열 수 있는 모든 서버 시스템에 광역 로드 밸런스를 직접 제공할 수 있습니다. 원격 서버가 캡슐화 GRE 패킷을 지원하는 경우, 원격 사이트에 Network Dispatcher를 설치할 필요가 없습니다. Network Dispatcher는 3735928559 10진수 값에 설정된 GRE 키 필드로 WAND 패킷을 캡슐화합니다.

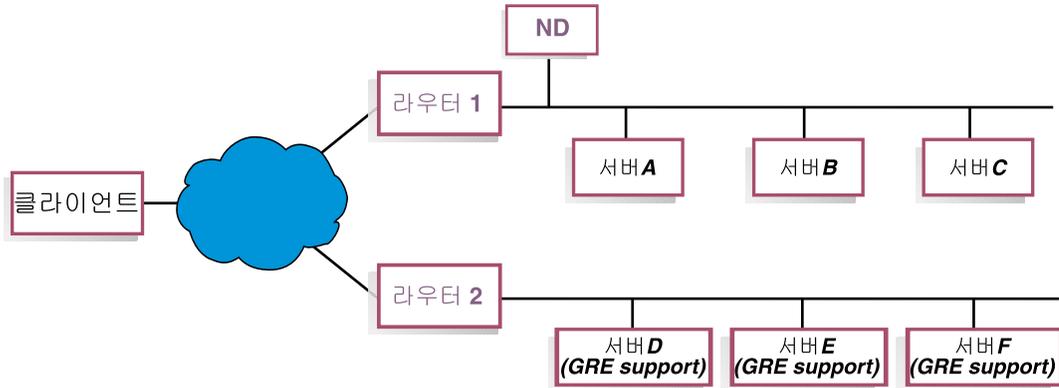


그림 25. GRE를 지원하는 서버 플랫폼의 광역 구성 예제

이 예제(그림25)에서 GRE를 지원하는 원격 ServerD를 추가하려면 cluster:port:server 계층에 WAND 서버를 정의한 것과 마찬가지로 Network Dispatcher 구성 내에 ServerD를 정의하십시오.

```
ndcontrol server add cluster:port:ServerD router Router1
```

상하단부 WAND 구성에서 자가 권고자 사용

자가 권고자는 Dispatcher 구성요소에서 사용 가능합니다.

상하단부 WAND(Wide Area Network Dispatcher) 구성의 Network Dispatcher의 경우, Dispatcher가 백엔드 서버에서 로드 상태 정보를 수집하는 자가 권고자를 제공합니다.

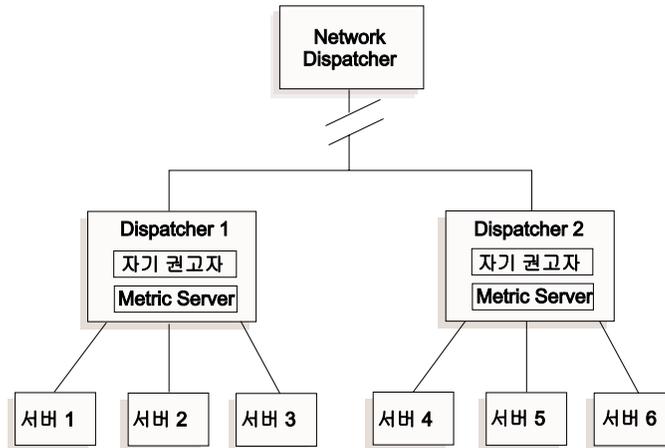


그림 26. 자가 권고자를 사용하는 상단부 WAND 구성의 예제

이 예제에서 자가 권고자는 Metric Server와 함께 상단부 Network Dispatcher가 로드 밸러스 중인 두 대의 Dispatcher 시스템에 있습니다. 자가 권고자는 특히 실행 프로그램 레벨에서 Dispatcher의 백엔드 서버에서 초당 연결 수를 측정합니다.

자가 권고자는 결과를 `ndloadstat` 파일에 기록합니다. 또한 Network Dispatcher는 `ndload`라는 외부 메트릭을 제공합니다. 각 Dispatcher 시스템의 Metric Server 에이전트는 `ndload`라는 외부 메트릭을 호출하는 구성을 실행합니다. `ndload` 스크립트는 `ndloadstat` 파일에서 문자열을 추출하여 Metric Server 에이전트로 리턴합니다. 그 다음에 각 Metric Server 에이전트(각 Dispatcher의)가 클라이언트 요청을 전달할 Dispatcher를 결정하는 데 사용하기 위해 로드 상태 값을 상단부 Network Dispatcher로 리턴합니다.

`ndload` 실행 프로그램은 Network Dispatcher용 `.../nd/ms/script` 디렉토리에 있습니다.

고가용성

고가용성 기능은 Dispatcher 구성요소에 대해서만 사용할 수 있습니다.

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 Dispatcher 모두 동일한 운영 체제를 사용해야 합니다.
- Dispatcher 고장을 발견하기 위한 두 Dispatcher 사이의 『핵심 메시지(heartbeat)』 메커니즘. 적어도 하나 이상의 핵심 메시지 쌍에는 출발지 및 목적지 주소로서 NFA 쌍이 있어야 합니다.

가능하면 핵심 메시지 쌍 중 최소한 하나는 일반 클러스터 통신이 아니라 별도의 서브넷을 통해 전송하는 것이 더 좋습니다. 핵심 메시지를 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 정보는 191 페이지의 『핵심 메시지 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 조작용이 활성 및 대기 상태를 전환하기로 결정한 경우 신속한 IP 인계를 수행하기 위한 메커니즘.

주: 두 클러스터 설정을 공유하는 두 개의 Dispatcher 시스템이 서로 백업을 제공하는 상호 고가용성 구성에 대해서는 59 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

ndcontrol highavailability의 완전한 구문은 302 페이지의 『ndcontrol highavailability -- 고가용성 제어』에 나와 있습니다.

다음의 많은 태스크에 대해 자세히 알려면 70 페이지의 『Dispatcher 시스템 설정』을 참조하십시오.

1. 두 대의 Dispatcher 서버 시스템 모두에서 서버를 시작하십시오.

2. 두 시스템 모두에서 실행 프로그램을 시작하십시오.
3. 각 Dispatcher 시스템의 비전달 주소(NFA)가 구성되어 있고 Dispatcher 시스템의 서버넷에 유효한 IP 주소인지 확인하십시오.

Windows에만 해당: 또 **ndconfig** 명령을 사용하여 각 비전달 주소를 구성하십시오.

예제:

```
ndconfig en0 nfa_addr netmask netmask
```

4. 두 시스템 모두에 클러스터, 포트 및 서버 정보를 설정하십시오.

주: 상호 고가용성 구성(59 페이지의 그림14)의 경우, 두 Dispatcher 간에 공유되는 클러스터 설정을 다음과 같이 구성하십시오.

- Dispatcher 1의 경우 다음을 발행하십시오.

```
ndcontrol cluster set clusterA primaryhost NFAdispatcher1
ndcontrol cluster set clusterB primaryhost NFAdispatcher2
```

- Dispatcher 2의 경우 다음을 발행하십시오.

```
ndcontrol cluster set clusterB primaryhost NFAdispatcher2
ndcontrol cluster set clusterA primaryhost NFAdispatcher1
```

5. 두 시스템 모두에서 관리 프로그램과 권고자를 시작하십시오. 도달 권고자는 관리 프로그램 기능에 의해 자동으로 시작됩니다.
6. 두 대의 Dispatcher 시스템에 별명 스크립트 파일을 작성하십시오. 193 페이지의 『스크립트 사용』을 참조하십시오.
7. 두 시스템 모두에 핵심 메시지를 추가하십시오.

```
ndcontrol highavailability heartbeat add sourceaddress destinationaddress
```

주: *Sourceaddress* 및 *destinationaddress*는 Dispatcher 시스템의 IP 주소(DNSnames 또는 점분리 10진수 주소)입니다. 값은 각 시스템에서 역행될 것입니다.

예제:

```
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 핵심 메시지 쌍에는 출발지 및 목적지 주소로서 NFA 쌍이 있어야 합니다.

가능하면 핵심 메시지 쌍 중 최소한 하나는 일반 클러스터 통신이 아니라 별도의 서브넷을 통해 전송하는 것이 더 좋습니다. 핵심 메시지를 별도로 전송하면 네트워크 로드가 아주 많은 경우 데이터를 가로채지 못하게 막을 수 있으며 장애 발생 후 전체 복구 시간을 줄일 수 있습니다.

8. 두 시스템 모두에서 **reach add** 명령을 사용하여 완전한 서비스를 보장하기 위해 Dispatcher가 도달할 수 있어야 하는 IP 주소의 목록을 구성하십시오. 예제:

```
ndcontrol highavailability reach add 9.67.125.18
```

도달 목표는 권장되지만 필수는 아닙니다. 자세한 정보는 191 페이지의 『핵심 메시지 및 도달 목표를 사용하는 고장 검색 기능』을 참조하십시오.

9. 각 시스템에 백업 정보를 추가하십시오.

- 기본 시스템의 경우:

```
ndcontrol highavailability backup add primary [auto | manual] port
```

- 백업 시스템의 경우:

```
ndcontrol highavailability backup add backup [auto | manual] port
```

- 상호 고가용성의 경우, 각 Dispatcher 시스템은 기본 및 백업 역할이 모두 있습니다.

```
ndcontrol highavailability backup add both [auto | manual] port
```

주: 시스템에서 사용하지 않는 포트를 *port*로 선택하십시오. 두 시스템은 이 포트를 통해 통신합니다.

10. 각 시스템에서 고가용성 상태를 확인하십시오.

```
ndcontrol highavailability status
```

시스템에는 각각 올바른 역할(백업 및 기본 또는 둘다), 상태 및 부속 상태가 있어야 합니다. 기본은 활성화되어 동기화되며, 백업은 대기 모드에 있고 단 시간 내에 동기화되어야 합니다. 전략어는 같아야 합니다.

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(핵심 메시지, 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 ndcontrol 명령(구성을 갱신하기 위해)을 입력한 다음에 활성 시스템에 입력하는 것이 좋습니다.
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. Linux의 경우, Dispatcher 구성요소의 MAC 포트 전달 방법을 사용하여 고가용성 및 결합 배치를 동시에 구성하려면 Linux 커널 패치를 설치해야 합니다. 패치 설치에 대한 자세한 정보는 83 페이지의 『Linux 커널 패치 설치(루프백 인터페이스에서 arp 응답 억제)』를 참조하십시오.

핵심 메시지 및 도달 목표를 사용하는 고장 검색 기능

고장을 발견하는 기본적인 기준 이외에도(핵심 메시지를 통해 발견되는 활성 및 대기 Dispatcher간의 연결성 유실), 도달 가능성 기준이라는 또다른 고장 발견 메커니즘이 있습니다. Dispatcher 구성 시, 올바른 작동을 위해 각 Dispatcher에서 도달할 수 있는 호스트 목록을 제공할 수 있습니다.

Dispatcher 시스템에서 사용하는 서브넷마다 최소한 하나의 호스트를 선택해야 합니다. 호스트는 라우터, IP 서버 또는 다른 유형의 호스트일 수 있습니다. 호스트 도달 가능성은 호스트를 ping하는 도달 권고자가 확보합니다. 핵심 메시지가 전달될 수 없는 경우나 도달 가능성 기준에 기본 Dispatcher가 아닌 대기 Dispatcher가 더 적합한 경우에는 전환이 발생합니다. 사용 가능한 모든 정보에 근거하여 결

정하려면, 활성 Dispatcher는 정기적으로 대기 Dispatcher에 그 도달 가능성 기능을 보냅니다. 그러면 대기 Dispatcher는 그 고유의 기능과 비교하여 전환 여부를 결정합니다.

주: 도달 목표 구성 시, 도달 권고자도 시작해야 합니다. 도달 권고자는 관리 프로그램에 의해 자동으로 시작됩니다. 도달 권고자에 대한 자세한 정보는 162 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.

전략어 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: 텔넷)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).
 - 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.

고장이 발견되면 Dispatcher 시스템의 상태가 변경되므로, 위의 명령은 자동으로 발행되어야 합니다. 이렇게 하기 위해 Dispatcher는 사용자가 작성한 스크립트를 실행합니다. 예제 스크립트는 **...nd/servers/samples** 디렉토리에 있으며, 실행 시 **...nd/servers/bin** 디렉토리로 반드시 이동해야 합니다.

주: 상호 고가용성 구성에서 각 “go” 스크립트는 기본 Dispatcher 주소를 식별하는 매개변수를 사용하여 Dispatcher에서 호출합니다. 스크립트는 이 매개변수를 조회하고 기본 Dispatcher와 연관된 클러스터 주소에 대해 **ifconfig** 명령을 수행해야 합니다(Windows 2000의 경우 **ndconfig** 명령 수행).

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 모니터링하지만 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

goInOp 스크립트는 Dispatcher 실행 프로그램이 정지되고 시작되기 전에 처음 실행합니다.

- 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 않으면 **ndcontrol cluster 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.

주: Windows 2000의 경우: 구성 설치 시, Site Selector가 고가용성 환경에서 작동 중인 두 개의 Dispatcher 시스템을 로드 밸런스하도록 하는 경우, Metirc Server의 Microsoft 스택에 별명을 추가해야 합니다. 이 별명을 goActive 스크립트에 추가해야 합니다.

예제:

```
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.

예제:

```
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.

규칙 기반 로드 밸런스 구성

규칙 기반 로드 밸런스를 사용하여 패킷이 언제 무슨 이유로 어떤 서버로 전송되는지를 조정할 수 있습니다. Network Dispatcher는 사용자가 추가하는 규칙을 첫 번째 우선순위에서 마지막 우선순위까지 검토하며, 처음으로 발견한 올바른 규칙에서 중단한 후 해당 규칙과 연관된 서버 간 내용의 로드 밸런스를 수행합니다. 목적지와 포트에 따라 이미 로드 밸런스를 수행하지만 규칙을 사용하면 사용자의 능력이 확장되어 연결을 분산시킬 수 있습니다.

규칙을 구성하는 대부분의 경우, 우선순위가 높은 다른 규칙을 통해 전달되는 요청을 받기 위해 기본적으로 항상 참인 규칙을 구성하십시오. 이것은 다른 모든 서버가 클라이언트 요청에 실패할 때 "죄송합니다. 사이트가 현재 작동 중지되었습니다. 나중에 다시 시도하십시오."라는 응답이 될 수 있습니다.

어떤 이유로 서버의 서브셋을 사용하려는 경우, Dispatcher 구성요소 및 Site Selector와 함께 규칙 기반 로드 밸런스를 사용해야 합니다. 항상 CBR 구성요소의 규칙을 사용해야 합니다.

주: 규칙을 사용하는 구성은 Mailbox Locator(사용자 ID 및 암호에 따라 IMAP 또는 POP3 요청을 특정 서버에 프록시함) 또는 Cisco Consultant(관리 프로그램 및 권고자 기능을 사용하여 Cisco CSS Switch에 로드 밸런스 정보를 제공함)에 적용되지 않습니다.

다음 유형의 규칙에서 선택할 수 있습니다.

- Dispatcher의 경우:
 - 클라이언트 IP 주소
 - 시간
 - 포트의 초당 연결 수
 - 포트의 총 작동 중인 연결 수
 - 클라이언트 포트
 - 서비스 유형(TOS)
 - 예약된 대역폭
 - 공유 대역폭

- 항상 참
- 요청 내용
- CBR의 경우:
 - 클라이언트 IP 주소
 - 시간
 - 포트의 초당 연결 수
 - 포트의 총 작동 중인 연결 수
 - 항상 참
 - 요청 내용
- Site Selector의 경우:
 - 클라이언트 IP 주소
 - 시간
 - Metric All
 - Metric Average
 - 항상 참

사용자 구성에 규칙 추가를 시작하기 전에 규칙이 수행해야 할 로직의 계획을 세우는 것이 좋습니다.

규칙 평가 방법

모든 규칙에는 이름, 유형, 우선순위가 있으며, 서버 집합과 함께 시작 및 종료 범위가 있을 수도 있습니다. 또한 CBR 구성요소의 내용 유형 규칙에는 규칙과 연관된 일치하는 일반 표현식 패턴이 있습니다(내용 규칙 및 내용 규칙의 유효한 패턴 구문을 사용하는 방법에 대한 예제 및 시나리오는 347 페이지의 『부록C. 콘텐츠 규칙(패턴) 구문』을 참조하십시오).

규칙은 우선순위로 평가합니다. 다시 말해서 우선순위 가 1(낮은 숫자)인 규칙을 우선순위가 2(높은 숫자)인 규칙보다 먼저 평가합니다. 만족시키는 첫 번째 규칙을 사용합니다. 일단 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 내용은 내용 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 이들 중 하나는 패킷을 전달하는 데 사용할 수 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 집합에서 서버를 선택하고 Site Selector는 사이트 이름에서 사용할 수 있는 전체 서버 집합에서 서버를 선택하며 CBR은 Caching Proxy가 오류 페이지를 리턴하도록 합니다.

클라이언트 IP 주소에 따라 규칙 사용

이 규칙 유형은 Dispatcher, CBR 또는 Site Selector 구성요소에서 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 집합에서 오는 클라이언트로부터 지급되지 않아서 원치 않는 통신량이 많아진다는 것을 인식합니다. **ndcontrol rule** 명령을 사용하여 규칙을 작성할 수 있습니다.

```
ndcontrol rule add 9.67.131.153:80:ni type ip beginrange 9.0.0.0 endrange 9.255.255.255
```

이 "ni" 규칙은 IBM 클라이언트로부터의 연결을 화면에 표시합니다. 그런 다음 IBM에 액세스할 서버를 규칙에 추가하거나 서버를 규칙에 추가하지 않을 경우 어떤 서버에서도 9.x.x.x 주소로부터의 요청이 제공되지 않습니다.

시간에 따라 규칙 사용

이 규칙 유형은 Dispatcher, CBR 또는 Site Selector 구성요소에서 사용할 수 있습니다.

용량을 계획하기 위해 시간에 따라 규칙을 사용하고자 할 수 있습니다. 예를 들어, 웹 사이트를 매일 같은 시간대에 작동하는 경우, 전체 시간 동안 5개 서버를 HTTP용으로 전용한 다음 피크 시간 중에는 또다른 5개 서버를 추가하려고 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 설정합니다.

포트의 초당 연결 수에 따라 규칙 사용

이 규칙 유형은 Dispatcher 및 CBR 구성요소에서 사용할 수 있습니다.

주: 다음을 작업하기 위해서는 관리 프로그램이 실행되고 있어야 합니다.

다른 응용프로그램과 사용자 서버 중 일부를 공유해야 하는 경우, 포트의 초당 연결 수에 따라 규칙을 사용하고자 할 수 있습니다. 예를 들어, 다음과 같은 두 가지 규칙을 설정할 수 있습니다.

1. 포트 80의 초당 연결 수 > 100인 경우 이들 두 개 서버 사용
2. 포트 80의 초당 연결 수 > 2000인 경우 이들 10개 서버 사용

아니면 텔넷을 사용 중일 수도 있으며, 초당 연결 수가 특정 레벨 이상으로 증가 되는 경우를 제외하고는 텔넷용으로 5개 서버 중 두 개를 예약하고자 할 수 있습니다. 이 경우, Dispatcher는 피크 시간에 5개의 서버 모두에서 로드 밸런스를 수행합니다.

포트의 총 작동 중인 연결 수에 따라 규칙 사용

이 규칙 유형은 Dispatcher 또는 CBR 구성요소에서 사용할 수 있습니다.

주: 다음을 작업하기 위해서는 관리 프로그램이 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 시간 종료되고 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 **ndcontrol rule** 또는 **cbrcontrol rule** 명령을 사용하여 규칙을 작성할 수 있습니다.

예제:

```
ndcontrol 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.

클라이언트 포트에 따라 규칙 사용

Dispatcher 구성요소의 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 집합을 사용하도록 지시하는 규칙을 작성할 수 있습니다.

서비스 유형(TOS)에 기반하여 규칙 사용

Dispatcher 구성요소의 경우에만 이 규칙 유형을 사용할 수 있습니다.

IP 헤더의 “서비스 유형”(TOS) 필드의 내용에 기반하여 규칙을 사용할 수 있습니다. 예를 들어, 클라이언트 요청에 표준 서비스를 나타내는 하나의 TOS 값이 포함되면 서버 집합 중 하나로 라우트될 수 있습니다. 다른 클라이언트 요청에 더 높은 우선 순위의 서비스를 나타내는 다른 TOS 값이 포함되면 다른 서버 집합으로 라우트될 수 있습니다.

TOS 규칙을 사용하면 **ndcontrol rule** 명령을 통해 TOS 바이트의 각 비트를 완전히 구성할 수 있습니다. TOS 바이트에서 일치시키려는 최상위 비트의 경우, 0 이나 1을 사용하십시오. 그렇지 않은 경우, x 값이 사용됩니다. 다음은 TOS 규칙 추가에 대한 예제입니다.

```
ndcontrol rule add 9.67.131.153:80:tsr type service tos 0xx1010x
```

예약된 대역폭 및 공유 대역폭에 따라 규칙 사용

용량 활용 및 대역폭 규칙은 Dispatcher 구성요소의 경우에만 사용할 수 있습니다.

Dispatcher는 용량 활용 기능을 사용하여 각 서버가 전달한 데이터의 양을 측정합니다. Dispatcher는 서버, 규칙, 포트, 클러스터 및 실행 프로그램 레벨에서 용량을 추적합니다. 각 레벨의 경우, 새로운 바이트 카운터 값(초당 전송 킬로바이트)이 있습니다. 비율 값(초당 전송 킬로바이트)은 60초 간격 이상으로 계산됩니다. 이러한 용량 값은 GUI 또는 명령행 보고서의 출력에서 볼 수 있습니다.

Dispatcher에서는 예약된 대역폭 규칙을 사용하여 사용자의 구성 내의 서버 집합에 지정된 대역폭을 할당할 수 있습니다. 통신량이 예약된 대역폭 임계치를 초과하는 경우, 다음 사항 중 하나를 수행할 수 있습니다.

- 항상 참 규칙을 사용하여 “사이트 사용 중” 유형의 응답을 하는 다른 서버에 통신을 전송하십시오.
- 또는 공유 대역폭 규칙을 사용하여 클러스터 레벨 또는 실행 프로그램 레벨에서 지정된 양의 대역폭을 공유하십시오. 그리고 전체 공유 대역폭 임계치에 도달하면, 항상 참 규칙을 사용하여 “사이트 사용 중” 유형의 응답을 하는 다른 서버에 통신을 전송할 수 있습니다.

위에서 설명한 대로, 예약된 대역폭 규칙과 함께 공유 대역폭 규칙을 사용하면, 기본 클라이언트에게 증가된 서버 액세스를 제공하여 트랜잭션의 성능을 최적화할 수

있습니다. 예를 들어, 공유 대역폭 규칙을 사용하여 사용하지 않는 대역폭을 활용하면, 서버 클러스터에서 거래를 실행하는 온라인 거래 고객이 투자를 연구하는 데 다른 서버 클러스터를 사용하는 고객보다 많은 액세스를 받도록 할 수 있습니다.

다음은 주의하여 대역폭 규칙이 서버에서 클라이언트로 전달되는 응답 통신량을 관리하는 데 도움을 줄 수 있는지 여부를 판별하십시오.

- 대역폭 규칙은 클라이언트 요청에 따라 서버 시스템 집합에서 전달되며 Network Dispatcher를 통해 전달되는 응답 통신량을 관리하는 데 도움을 줄 수 있습니다. 일부 클라이언트 통신량이 서버 시스템으로 직접 이동하여 Network Dispatcher에서 볼 수 없는 경우, 결과를 예측할 수 없습니다.
- 대역폭 규칙은 모든 서버가 네트워크에 대한 동일한 연결을 사용하는 경우, 서버 시스템 집합으로부터 네트워크에 대한 연결에서 전달되는 응답 통신량을 관리하는 데 도움을 줄 수 있습니다. 서버가 다른 연결을 사용하거나 여러 연결을 사용하여 네트워크에 연결되는 경우, 각 개별 연결의 결과를 예측할 수 없습니다.
- 대역폭 규칙은 모든 서버가 Network Dispatcher 시스템과 같은 동일한 로컬 네트워크에 있는 경우에만 유용합니다. 일부 서버는 원격이며 다른 경로로 네트워크에 연결되는 경우에는 결과를 예측할 수 없습니다.

예약된 대역폭 규칙

Dispatcher 구성요소의 경우에만 이 규칙 유형을 사용할 수 있습니다.

예약된 대역폭 규칙을 사용하면 서버 집합에서 전달 중인 초당 킬로바이트 수에 따라 로드 밸런스를 수행할 수 있습니다. 구성 전체의 각 서버 집합에 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.

```
ndcontrol rule add 9.67.131.153:80:rbw type reservedbandwidth  
beginrange 0 endrange 300
```

시작 및 종료 범위는 초당 킬로바이트로 지정됩니다.

공유 대역폭 규칙

Dispatcher 구성요소의 경우에만 이 규칙 유형을 사용할 수 있습니다.

전송된 데이터의 양이 예약된 대역폭 규칙의 한계를 초과하는 경우, 공유 대역폭 규칙에서 사이트에서 사용할 수 있는 미사용 대역폭을 활용하는 기능을 제공합니다. 클러스터나 실행 프로그램 레벨에서 대역폭을 공유하도록 이 규칙을 구성할 수 있습니다. 클러스터 레벨에서 대역폭을 공유하면 하나 또는 여러 개의 포트가 동일한 클러스터 내의 여러 포트에서(응용프로그램/프로토콜) 대역폭의 최대량을 공유할 수 있습니다. 실행 프로그램 레벨에서 대역폭을 공유하면 전체 Dispatcher 구성 내의 하나 또는 여러 개의 클러스터가 대역폭의 최대량을 공유할 수 있습니다.

공유 대역폭 규칙을 구성하기 전에 `sharedbandwidth` 옵션이 있는 `ndcontrol executor` 또는 `ndcontrol cluster` 명령을 사용하여 실행 프로그램 또는 클러스터 레벨에서 공유할 수 있는 대역폭의 최대량(초당 킬로바이트)을 지정해야 합니다. 다음은 명령 구문의 예제입니다.

```
ndcontrol executor set sharedbandwidth size
ndcontrol cluster [add | set] 9.12.32.9 sharedbandwidth size
```

`sharedbandwidth`의 크기는 정수값(초당 킬로바이트)입니다. 기본값은 0입니다. 값이 0이면 대역폭을 공유할 수 없습니다. 사용할 수 있는 총 대역폭(총 서버 용량)을 초과하지 않는 최대 `sharedbandwidth` 값을 지정해야 합니다.

다음은 `sharedbandwidth` 규칙 추가 또는 설정의 예제입니다.

```
ndcontrol rule add 9.20.30.4:80:shbw type sharedbandwidth sharelevel value
ndcontrol rule set 9.20.34.11:80:shrulc sharelevel value
```

`sharelevel`의 값은 실행 프로그램이나 클러스터입니다. `Sharelevel`은 `sharebandwidth` 규칙의 필수 매개변수입니다.

Metric all 규칙

이 규칙 유형은 Site Selector 구성요소에서만 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 집합에 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 구성요소에서만 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 집합에 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 구성요소에서 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 집합을 갖는 “항상 참” 규칙을 가질 수도 있습니다. 사용 가능한 서버가 있는 첫 번째 참 규칙이 선택됩니다. 예를 들어, 6개의 서버가 있다고 가정합니다. 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 집합이 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 집합이 선택됩니다. 둘 모두 단절되면, 두 번째 집합 중 하나가 선택되는 방식으로 진행됩니다.

또다른 예제로서 “항상 참” 규칙은 수신 클라이언트가 설정된 규칙에 맞지 않을 경우, 이들이 제공되지 않는지 확인하는 데 필요할 수 있습니다. 사용자는 **ndcontrol rule** 명령을 사용하여 규칙을 작성합니다.

```
ndcontrol rule add 130.40.52.153:80:jamais 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 내용에 따라 규칙 사용

이 규칙 유형은 Dispatcher 또는 CBR 구성요소에서 사용할 수 있습니다.

사용자는 내용 유형 규칙을 사용하여 사용자 사이트 통신의 일부 서브세트를 처리하도록 특별히 설정된 서버 집합에 요청을 전송하려고 할 수 있습니다. 예를 들어, 한 서버 집합은 모든 *cgi-bin* 요청을 처리하고, 다른 집합은 모든 스트리밍 오디오 요청을 처리하며 또다른 집합은 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 또다른 규칙을 추가하면 됩니다. 그런 다음 각 규칙에 적절한 서버를 추가하십시오.

중요: 내용 규칙 및 내용 규칙의 유효한 패턴 구문을 사용하는 방법에 대한 예제 및 시나리오는 347 페이지의 『부록C. 콘텐츠 규칙(패턴) 구문』을 참조하십시오.

사용자 구성에 규칙 추가

예제 구성 파일을 편집하거나 GUI(Graphical User Interface)로 **ndcontrol 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 얼마나 많이 사용하는지를 추적하고자 합니다. 관리자는 각 부서에 제공된 IP 주소를 알고 있습니다. 먼저 클라이언트 IP 주소에 따라 첫 번째 규칙 집합을 작성하여 각 부서의 로드를 구분합니다.

```
ndcontrol rule add 130.40.52.153:80:div1 type ip b 9.1.0.0 e 9.1.255.255
ndcontrol rule add 130.40.52.153:80:div2 type ip b 9.2.0.0 e 9.2.255.255
ndcontrol 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 측정합니다. 예제:

```
ndcontrol rule useserver 130.40.52.153:80:div1 207.72.33.45
ndcontrol rule useserver 130.40.52.153:80:div2 207.72.33.63
ndcontrol rule useserver 130.40.52.153:80:div3 207.72.33.47
```

규칙에 대한 서버 평가 옵션

서버 평가 옵션은 Dispatcher 구성요소에서만 사용할 수 있습니다.

ndcontrol rule 명령에는 규칙에 대한 서버 평가 옵션이 있습니다. 평가 옵션을 사용하여 포트에 있는 모든 서버에서 규칙의 조건을 평가하거나 규칙 내의 서버에서만 규칙의 조건을 평가하십시오(Network Dispatcher의 이전 버전에서는 포트에 있는 모든 서버에서 각 규칙의 조건을 측정만 가능했습니다).

주: 서버 평가 옵션은 서버의 특성에 따라 결정하는 규칙(초당 총 연결 규칙, 활성 연결 규칙 및 예약된 대역폭 규칙)에만 유효합니다.

다음은 예약된 대역폭 규칙에서 평가 옵션을 추가 또는 설정하는 예제입니다.

```
ndcontrol rule add 9.22.21.3:80:rbweval type reservedbandwidth evaluate level
ndcontrol rule set 9.22.21.3:80:rbweval evaluate level
```

평가 레벨은 포트 또는 규칙으로 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 내용을 유지하는 모든 서버가 포함되며, 평가 옵션은 규칙으로 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 포트에 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.

명시적 링크 사용

일반적으로, Dispatcher의 로드 밸런스 기능은 제품이 사용되는 사이트의 내용과는 관계없이 작동됩니다. 그러나 사이트 내용이 중요할 수 있으며 내용에 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 사용하는 것이 좋습니다. 사용자 사이트에서 동적으로 HTML

을 작성하는 자동화 프로그래밍을 사용할 경우, 사용되는 주소 지정 종류가 항상 명백하지 않을 수도 있다는 점에 유의하십시오. 로드 밸런스를 최대화하려면 명시적 주소 지정에 유의하여 가능한 이를 피해야 합니다.

사설 네트워크 구성 사용

사설 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.

AIX에서 이 구성은 Dispatcher 및 TCP 서버 시스템을 SP 프레임의 노드에서 실행할 경우, SP 고성능 스위치의 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 2000: 다음 명령을 실행하십시오.

```
ndconfig en1 10.0.0.x netmask 255.255.255.0
```

여기서 en1은 Dispatcher 시스템에서 두 번째 인터페이스 카드의 이름이고 10.0.0.x는 두 번째 인터페이스 카드의 네트워크 주소이며 255.255.255.0은 사설 네트워크의 넷마스크입니다.

ndcontrol server add 명령을 사용하여 추가된 서버는 사설 네트워크 주소를 사용하여 추가되어야 합니다. 예를 들어, 209 페이지의 그림27에 나와 있는 Apple 서버 예제의 경우는 다음과 같이 코딩되어야 합니다.

```
ndcontrol server add cluster_address:80:10.0.0.1
```

다음과 같이 코딩해서는 안됩니다.

```
ndcontrol 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.

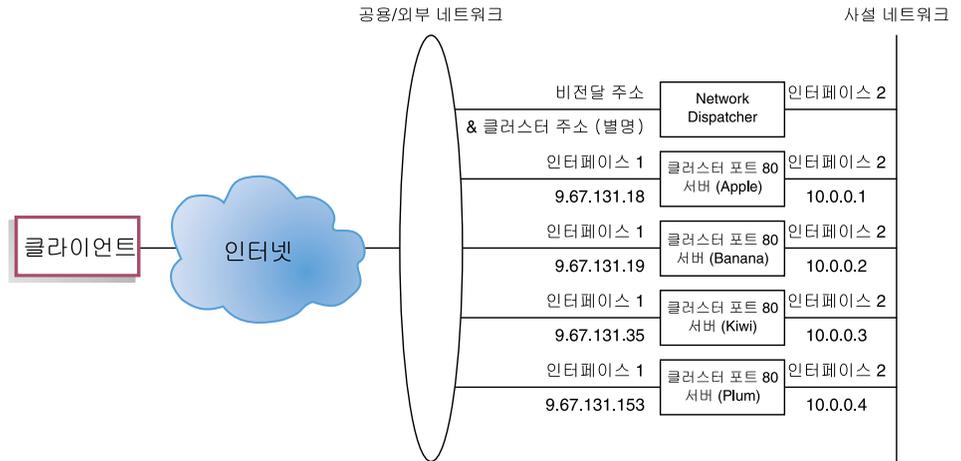


그림 27. Dispatcher를 사용하는 사설 네트워크 예제

Dispatcher 구성요소의 경우에만 사설 네트워크 구성이 적용됩니다.

와일드 카드 클러스터를 사용하여 서버 구성 조합

“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 별 구성으로 조합할 수 있습니다.

Dispatcher 워크스테이션의 네트워크 어댑터 중 하나에 각 클러스터 주소를 명시적으로 구성해야 합니다. 그러나 `ndcontrol 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 수행하게 됩니다.

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 수행하게 됩니다.

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.

Dispatcher 구성요소의 경우에만 와일드 카드 클러스터를 사용한 서버 구성 조합이 적용됩니다.

와일드 카드 클러스터를 사용하여 방화벽 로드 밸런스 수행

Dispatcher 구성요소의 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 집합의 기본 라우트로서 설정되지 않은 경우, 그 네트워크 어댑터 중 하나에서 명시적으로 구성되지 않은 주소에 대한 통신량은 알지 못합니다.

일단 Dispatcher가 기본 라우트로 구성되었으면, Dispatcher 시스템을 통한 TCP 또는 UDP 통신량은 와일드 카드 클러스터 구성을 통해 로드 밸런스가 조정됩니다.

이 중 하나의 응용프로그램은 방화벽 로드 밸런스를 수행하게 됩니다. 방화벽이 목적지 주소와 목적지 포트에 대해 패킷을 처리할 수 있으므로 목적지 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 집합을 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 구성요소의 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 상자에 있는 Caching Proxy 서버의 투명 프록시 기능을 작동시키는 데 사용될 수도 있습니다. 이 기능은 Dispatcher 구성요소에서 운영 체제의 TCP 구성요소로의 통신이 있어야 하므로 AIX 기능 전용입니다.

이 기능을 사용하려면 포트 80에서 클라이언트 요청을 인식하는 Caching Proxy를 시작해야 합니다. 그런 다음 와일드 카드 클러스터를 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 구성요소는 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 밸런스를 위해 사용됩니다. 구성되어 있지 않은 포트에 대한 통신량이 적당하게 처리되도록 두 번째 것이 사용됩니다. 서버없이 와일드 카드 포트를 정의하여 구성되지 않은 포트에 대한 요청이 운영 체제로 다시 전달되지 않고 버려지게 됩니다. 포트 번호 0은 다음과 같은 와일드 카드 포트를 지정합니다.

```
ndcontrol port add cluster:0
```

주: 와일드 카드 포트는 FTP 통신량을 처리하는 데 사용할 수 없습니다.

Network Dispatcher에 대한 친근성 기능 사용법

친근성 기능은 사용자가 클러스터 포트를 결합(sticky)으로 구성할 때 사용 가능합니다. 클러스터 포트를 결합으로 구성하면 클라이언트의 연속적인 요구가 동일한 서버로 지정됩니다. 이것은 “포트 결합 시간”을 초 수로 설정하면 이루어집니다. 이 친근성 기능은 결합 시간을 0으로 설정하면 해제됩니다.

크로스 포트 친근성과의 상호작용: 크로스 포트 친근성이 사용 가능하면, 공유된 포트의 결합 값은 같은 값(0이 아닌)이어야 합니다. 자세한 정보는 214 페이지의 『포트간 친근성』을 참조하십시오.

친근성이 사용 가능하지 않을 때의 행동

친근성 기능이 사용 가능하지 않은 상태에서 클라이언트로부터 새로운 TCP 연결을 받을 때마다, Dispatcher는 적시에 올바른 서버를 선택하여 그 패킷을 서버에 전달합니다. 연속되는 연결이 같은 클라이언트에게서 들어오면 Dispatcher는 그 연속 연결을 연관되지 않은 새로운 연결로 간주하고 다시 적시에 올바른 서버를 선택합니다.

친근성이 사용 가능할 때의 행동

친근성 기능이 사용 가능한 상태에서 클라이언트가 연속된 요구를 해오면, 요구는 동일한 서버로 지정됩니다.

시간이 지남에 따라 클라이언트는 트랜잭션 전송을 끝내며 그 친근성 기록은 사라지게 됩니다. 따라서 결합 ”시간의 의미는 다음과 같습니다. “각각의 친근성 기록은 수 초의 ”결합 시간“을 위해 존재합니다. 결합 시간 내에 연속적인 연결이 이루어지면, 친근성 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.

클라이언트 서버 유사성을 제어하는 서버 지정 유사 API

Dispatcher 구성요소의 경우에만 서버 지정 유사 API가 적용됩니다.

SDA 기능은 외부 에이전트가 Dispatcher 유사 활동에 영향을 줄 수 있는 API를 제공합니다.

주: SDA에서는 구성에서 검색 기능에 대해 서버 주소가 고유해야 하므로 Server Directed Affinity가 서버 파티션 기능과 함께 작동하지 않는 제한사항이 있습니다. 또한 SDA로 서버가 친근성 테이블을 제어하기 때문에 SDA는 SSL ID 친근성 기능과 함께 작동하지 않습니다.

SDA 기능

응용프로그램은 Dispatcher가 할 수 있는 것보다 더 나은 특정 서버 시스템으로 클라이언트 요청을 지정한다는 사실을 그 서버 시스템에서 알도록 지시합니다. Dispatcher 로드 밸런스 선택으로 선택된 것과 동일한 서버에 클라이언트가 ”지정“되는 것이 아니라, 선택한 서버에 클라이언트가 ”지정“되기를 원합니다. SDA 기능은 이 API를 제공합니다. 이제 사용자 고유의 소프트웨어를 작성하여 Dispatcher의 리스너와 통신하는 SDA 에이전트를 구현할 수 있습니다. 그런 후 Dispatcher 유사 테이블을 조작하여 다음과 같이 할 수 있습니다.

- 내용 조회
- 새로운 레코드 삽입
- 레코드 제거

SDA 에이전트에서 유사 테이블에 삽입된 레코드는 테이블에 무한정 남아 있습니다. 이들은 시간 종료되지 않습니다. 이들은 SDA 에이전트가 이들을 제거하거나 Dispatcher 권고자가 서버가 작동되지 않는다는 것을 발견한 경우에만 제거됩니다.

Dispatcher의 SDA 구성요소

Dispatcher는 SDA 에이전트에서 요청을 받아들여 처리할 새로운 소켓 리스너를 구현합니다. SDA 에이전트가 Dispatcher로 연결을 열면 리스너는 이를 승인하고 연결을 연 채로 그대로 둡니다. 여러 요청 및 응답은 이 지속적인 연결을 통해 이동할 수 있습니다. 소켓은 SDA 에이전트가 이를 닫거나 Dispatcher가 복구할 수 없는 오류를 발견한 경우 닫힙니다. Dispatcher 내에서 리스너는 SDA 에이전트로부터 각 요청을 가져오고 Dispatcher 실행 프로그램 커널의 해당 유사 테이블과 통신하며 SDA 에이전트의 응답을 준비합니다.

자세한 정보는 Network Dispatcher의 설치 디렉토리에 표시되는 파일을 참조하십시오.

- API: `...nd/servers/samples/SDA/SDA_API.htm`
- SDA 에이전트의 예제 코드: `...nd/servers/samples/SDA/SDA_SampleAgent.java`.

포트간 친근성

포트간 친근성은 Dispatcher 구성요소에만 적용됩니다.

포트간 친근성은 여러 포트로 확장될 수 있는 친근성 기능입니다. 예를 들어, 처음에 한 포트에 클라이언트 요청이 수신되고 다른 포트에 다음 요청이 수신되면 포트간 친근성을 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** 보유 값
- 동일한 **stickymask** 보유 값

둘 이상의 포트가 동일한 **crossport**에 링크될 수 있습니다. 후속 연결이 동일한 포트나 공유 포트의 동일한 클라이언트에서 시도되면 동일한 서버가 액세스됩니다. 다음은 포트간 친근성을 사용하여 포트 10으로 여러 포트를 구성하는 예제입니다.

```
ndcontrol port set cluster:20 crossport 10
ndcontrol port set cluster:30 crossport 10
ndcontrol port set cluster:40 crossport 10
```

포트간 친근성이 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 친근성을 제거하려면 crossport 값을 다시 자신의 포트 번호로 설정하십시오. **crossport** 옵션에 대한 명령 구문에 대한 자세한 정보는 319 페이지의 『ndcontrol 포트 -- 포트 구성』을 참조하십시오.

친근성 주소 마스크

친근성 주소는 Dispatcher 구성요소에만 적용됩니다.

친근성 주소 마스크는 공통적인 서브넷 주소에 기반하여 클라이언트를 그룹화하는 향상된 결합 기능입니다. **stickymask**를 **ndcontrol port** 명령에서 지정하여 32비트 IP 주소의 공통된 상위 순서 비트를 가릴 수 있습니다. 이 기능이 사용 가능으로 설정되면 클라이언트 요청은 먼저 포트에 연결을 수행하고 동일한 서브넷 주소(가려질 주소 부분으로 표시)를 가진 모든 후속 클라이언트 요청이 동일한 서버로 보내집니다.

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 Network Dispatcher를 처음 시작할 때 stickymask 값을 설정하십시오. stickymask 값을 동적으로 변경하면 결과를 예측할 수 없게 됩니다.

포트간 친근성을 통한 대화: 포트간 친근성을 사용 가능으로 설정하는 경우 공유 포트의 stickymask 값들은 동일해야 합니다. 자세한 정보는 214 페이지의 『포트간 친근성』을 참조하십시오.

친근성 주소 마스크를 사용 가능으로 설정하려면 다음과 유사하게 ndcontrol 포트 명령을 실행하십시오.

```
ndcontrol port set cluster:port 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 친근성 주소 마스크 기능을 효과적으로 사용 불가능하게 할 수 있도록 전체 IP 주소를 가리게 됩니다. stickymask의 기본값은 32입니다.

stickymask(친근성 주소 마스크 기능)의 명령 구문에 대한 자세한 정보는 319 페이지의 『ndcontrol 포트 -- 포트 구성』을 참조하십시오.

규칙 친근성 무시

규칙 친근성 무시 기능을 사용하면 특정 서버에 대한 포트의 결합을 무시할 수 있습니다. 예를 들어, 각 응용프로그램 서버에 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. 규칙 친근성 무시 기능을 사용하여 오버플로우 서버가 해당 포트와 연관된 친근성을 무시하도록 변경할 수 있습니다. 클라이언트가 클러스터를 요청하는 다음 번에 오버플로우 서버가 아니라 사용할 수 있는 응용프로그램 서버로 로드 밸런스가 진행됩니다.

규칙 친근성 무시, 서버 **sticky** 사용의 명령 구문에 대한 자세한 정보는 335 페이지의 『ndcontrol server -- 서버 구성』을 참조하십시오.

결합 연결 처리 작업중지

결합 연결 처리 작업중지는 Dispatcher 및 CBR 구성요소에 적용됩니다.

어떠한 이유로든(갱신, 업그레이드, 서비스 등) Network Dispatcher 구성에서 서버를 제거하려면 **ndcontrol 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 시간이 설정되어 있고 결합 시간이 만기되기 전에 새 연결을 다른 서버(작업 중지된 서버 대신)로 전송하려면 “지금 작업중지”를 사용하십시오. 다음은 서버 9.40.25.67을 작업중지하기 위해 지금 옵션을 사용하는 예제입니다.

```
ndcontrol 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(친근성) 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 Network Dispatcher에서는 서버를 처리하는 유일한 방법입니다.

규칙에 대한 연관 관계 옵션

ndcontrol rule 명령에서 다음과 같은 연관 관계 유형을 지정할 수 있습니다.

- 활성 쿠키 -- Network Dispatcher가 생성한 쿠키를 기준으로 동일한 서버로 친근성을 사용하여 웹 통신량을 로드 밸런스할 수 있습니다.
- 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.
- URI -- 캐시 크기를 효율적으로 증대시키는 방법으로 Caching Proxy 서버로 웹 통신량을 로드 밸런스할 수 있습니다.

친근성 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 stickytime 을 사용할 수 없습니다.

활성 쿠키 친근성은 CBR 구성요소에만 적용됩니다. 수동 쿠키 및 URI 연관 관계는 CBR 구성요소와 Dispatcher 구성요소의 crb 전달 방법에 적용됩니다.

활성 쿠키 연관 관계

활성 쿠키 연관 관계 기능은 CBR 구성요소에만 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 집합 명령을 사용할 때 이 기능이 수행됩니다. 명령 구문에 대한 자세한 정보는 327 페이지의 『ndcontrol 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, 로드 밸런스된 서버, 연관 관계가 무효화되는 시간 종료 시간 소인이 포함됩니다. 활성 쿠키 연관 관계가 설정된 규칙을 사용할 때마다 클라이언트가 보낸 쿠키가 검사됩니다. 사용된 cluster:port 에 대한 ID를 포함하는 쿠키가 발견되면 로드 밸런스된 서버 및 시간 소인 만기가 쿠키에서 추출됩니다. 규칙에서 사용한 집합에 서버가 계속 있고 가중치가 0보다 크며 시간 소인 만기가 지금 이후이면 쿠키 내의 서버가 로드 밸런스될 서버로 선택됩니다. 위 세 조건 중 만족되지 않는 조건이 있으면 표준 알고리즘을 사용하여 서버를 선택합니다. 두 가지 방법 중 하나를 사용하여 서버를 선택하면 IBNCBR, 클러스터, 포트, server_chosen 정보 및 시간 소인이 포함된 새 쿠키가

구성됩니다. 시간 소인은 연관 관계가 만기되는 시간입니다. “cluster:port:server_chosen”은 CBR 구성 정보가 드러나지 않도록 인코딩됩니다. “expires” 매개변수도 쿠키에 삽입됩니다. 이 매개변수는 브라우저가 이해할 수 있는 형식으로 구성되며 이 매개변수를 사용하면 시간 소인 만기부터 두 시간 후 쿠키가 무효화됩니다. 따라서 클라이언트 쿠키 데이터베이스는 정돈되어 있습니다.

그리고 나서, 새 쿠키는 클라이언트로 돌아가는 헤더에 삽입되며, 클라이언트 브라우저가 쿠키를 승인하도록 구성된 경우, 후속 요청을 다시 전송합니다.

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 집합 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 요청들 사이에서 해당 상태를 유지하기 위해 해당 서버의 쿠키를 필요로 합니다.

수동 쿠키 연관 관계

수동 쿠키 연관 관계는 Dispatcher 구성요소의 Content Based Routing(CBR) 전달 방법 및 CBR 구성요소에 적용됩니다. Dispatcher의 cbr 전달 방법 구성에 대한 정보는 63 페이지의 『Dispatcher content-based routing(cbr 전달 방법)』을 참조하십시오.

수동 쿠키 연관 관계는 특정 서버에 클라이언트를 결합시키는 방법을 제공합니다. 규칙의 연관 관계를 “passivecookie”로 사용 가능하게 하면 수동 쿠키 연관 관계로 서버가 생성한 자가 식별 쿠키에 따라 동일한 서버에 대한 연관 관계로 웹 통신량의 로드 밸런스를 수행할 수 있습니다. 규칙 레벨에서 수동 쿠키 연관 관계를

구성할 수 있습니다. 규칙이 실행된 후, 수동 쿠키 연관 관계가 사용 가능하게 되면 Network Dispatcher가 클라이언트 요청의 HTTP 헤더에 있는 쿠키 이름에 따라 서버를 선택합니다. Network Dispatcher는 이전 연결 중 서버가 생성한 쿠키에 따라 새 수신 요청을 서버로 전송합니다. 클라이언트 요청에 쿠키 값이 없거나 서버의 쿠키 값과 일치하지 않는 경우, 가중치 라운드 로빈 기술을 사용하여 서버를 선택합니다.

수동 쿠키 연관 관계를 구성하려면 다음을 수행하십시오.

- Dispatcher의 경우, 먼저 Dispatcher의 cbr 전달 방법을 구성합니다(63 페이지의 『Dispatcher content-based routing(cbr 전달 방법)』 참조). 이 단계는 CBR 구성요소에는 생략됩니다.
- **ndcontrol rule [add|set]** 명령에서 **affinity** 매개변수를 "passivecookie"로 설정합니다. 또한 **cookieName** 매개변수를 Network Dispatcher가 클라이언트 HTTP 헤더 요청에서 찾아야 하는 쿠키 이름으로 설정해야 합니다.
- **ndcontrol server [add|set]** 명령에서 규칙 서버 집합의 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 방법 및 CBR 구성요소에 적용됩니다. CBR 전달 방법 구성에 대한 정보는 63 페이지의 『Dispatcher content-based routing(cbr 전달 방법)』을 참조하십시오.

URI 연관 관계를 사용하면 고유한 내용을 각 서버에서 캐시할 수 있는 Caching Proxy 서버에 대한 웹 통신량의 로드 밸런스를 수행할 수 있습니다. 결과적으로 여러 시스템에서 내용의 중복 캐시를 제거하여 사이트 캐시의 크기를 효과적으로 증가시킵니다. 규칙 레벨에서 URI 연관 관계를 구성하십시오. 규칙이 실행된 후 URI 연관 관계가 사용 가능하게 되고 동일한 서버 집합이 가동하여 응답하면 Network Dispatcher는 동일한 URI가 있는 새 수신 클라이언트 요청을 동일한 서버로 전달합니다.

일반적으로 Network Dispatcher는 동일한 내용을 제공하는 여러 서버로 요청을 분산시킬 수 있습니다. 캐시 서버 그룹이 있는 Network Dispatcher를 사용할 때 액세스된 내용이 결국 모든 서버에 캐시되는 경우가 많습니다. 이것은 동일하게 캐시된 내용을 여러 시스템에 복제하여 매우 높은 클라이언트 로드를 지원합니다. 이것은 특히 볼륨이 큰 웹 사이트에서 유용합니다.

그러나 사용자의 웹 사이트가 매우 다양한 내용에 대한 중간 정도의 클라이언트 통신량을 지원하고 사용자가 보다 큰 캐시를 여러 서버에 분포시키려는 경우, 각 캐시 서버에 고유한 내용이 포함되고 Network Dispatcher가 해당 내용이 있는 캐시 서버에만 요청을 분산시키면 사이트의 성능이 향상됩니다.

URI 연관 관계와 함께 Network Dispatcher를 사용하면 캐시된 내용을 각 서버에 분산시켜 여러 시스템에 있는 내용의 중복 캐시를 제거할 수 있습니다. Caching Proxy 서버를 사용하는 다양한 내용 서버 사이트의 성능은 이러한 향상 내용과 함께 향상됩니다. 동일한 요청을 동일한 서버에 전송하므로 단일 서버에만 있는 내용을 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 방법을 구성하십시오(63 페이지의 『Dispatcher content-based routing(cbr 전달 방법)』 참조). 이 단계는 CBR 구성요소에는 생략됩니다.
- **ndcontrol rule [add|set]** 또는 **cbrcontrol rule [add|set]** 명령에서 **affinity** 매개변수를 "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.

서비스 거부 중지 감지

Dispatcher 구성요소의 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.

Network Dispatcher는 관리자에게 가능한 서비스 거부 중지를 경보하도록 사용자 정의할 수 있는 스크립트를 트리거하는 사용자 종료를 제공합니다. Dispatcher는 **...nd/servers/samples** 디렉토리에 다음의 예제 스크립트 파일을 제공합니다.

- halfOpenAlert - 가능한 서비스 거부(DoS) 중지가 감지되었습니다.
- halfOpenAlertDone - DoS 중지가 완료되었습니다.

파일을 실행하려면 예제 스크립트를 **...nd/servers/bin** 디렉토리로 이동하여 확장자가 ".sample"인 파일을 제거하십시오.

DoS 중지 감지를 구현하려면, 다음과 같이 **ndcontrol port** 명령에서 **maxhalfopen** 매개변수를 설정하십시오.

```
ndcontrol 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(**ndcontrol port halfopenaddressreport cluster:port**)를 실행하십시오. 반개방 연결 보고서는 현

제의 "수신된 총 반개방 연결"을 리턴합니다. 사용자의 사이트에 연결되는 가장 많은 반개방 연결 수보다 큰 50%에서 200% 사이의 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(..nd/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 235 페이지의 『Dispatcher 구성요소에 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 209 페이지의 『와일드 카드 클러스터를 사용하여 서버 구성 조합』 및 212 페이지의 『와일드 카드 포트를 사용하여 구성되어 있지 않은 포트 통신량 지정』을 참조하십시오.

서버 통제를 분석하기 위해 2진 로그 사용

주: 2진 로그 기능은 Site Selector 구성요소에는 적용되지 않습니다.

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 관리 프로그램 주기의 부분으로서 실행 프로그램에서 검색합니다. 따라서 관리 프로그램은 2진 로그에 로그될 정보에 대해 순서대로 실행되어야 합니다.

ndcontrol log 명령은 2진 로그 구성을 설정합니다.

- 로그 시작
- 로그 중지
- 로그 세트 간격 <초>
- 로그 세트 보존 <시>
- 로그 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 쓰여질 것인지를 제어합니다. 관리 프로그램은 관리 프로그램 간격마다 로그 서버로 서버 정보를 보냅니다. 마지막 기록이 로그에 쓰여진 이후 지정된 로그 간격 초가 경과된 경우에만 정보가 로그에 기록됩니다. 기본값으로 로그 간격은 60초로 설정됩니다. 관리 프로그램 간격과 로그 간격 설정 사이에는 약간의 상호작용이 있습니다. 로그 서버는 관리 프로그램 간격 초 설정보다 느린 정보를 제공하므로, 관리 프로그램 간격보다 느린 로그 간격은 관리 프로그램 간격과 동일하게 정보를 설정합니다. 이 로그 기술을 사용하여 미세한 서버 정보를 캡처할 수 있습니다. 서버 가중치를 계산하는 관리 프로그램이 보여주는 서버 정보로 모든 변경사항을 캡처할 수 있습니다. 그러나 이 정보

의 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 서버가 삭제합니다. 관리 프로그램에서 로그 서버를 호출하고 있는 경우에만 발생하므로, 관리 프로그램을 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 **...nd/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.

```
ndlogreport 2001/05/01 8:00 2001/05/01 17:00
```

2001년 5월 1일 오전 8시부터 오후 5시까지 Dispatcher 구성요소의 서버 정보에 대한 보고서를 작성하기 위한 것입니다. (CBR의 경우, **cbrlogreport**를 사용하십시오. Mailbox Locator의 경우, **mllogreport**를 사용하십시오. Cisco Consultant의 경우, **lbcllogreport**를 사용하십시오.)

고급 Cisco Consultant 기능에 대한 추가 정보

Cisco Consultant에서 Cisco CSS Switch는 Dispatcher 구성요소의 실행 프로그램에서 완료된 태스크를 수행합니다. 관리 프로그램에서는 각 서버의 현재 가중치 및 계산에 필요한 일부의 다른 정보와 함께 Cisco CSS Switch에서 활성 및 새 연결값을 확보합니다. 이들 값은 Cisco CSS Switch에서 내부적으로 생성되어 저장되는 정보에 따라 달라집니다.

Cisco Consultant는 Cisco CSS Switch 관리 정보 데이터베이스(MIB)를 조회하여 활성 및 새 연결 정보를 확보하고 다음 사항을 수신합니다.

- **활성 연결의 경우**, Cisco Consultant가 svcExtMIB에서 apSvcConnections를 받습니다. 이 변수는 serviceName으로 색인이 되며 관리 프로그램에 기록된 대로 활성 연결에 직접 맵핑합니다. 다음은 apSvcConnections MIB 항목입니다.

```
apSvcConnections OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "The current number of TCP connections to this service"
DEFVAL { 0 }
--DEFAULT ap-display-name Service Connections
 ::= {apSvcEntry 20}
```

apSvcConnections 오브젝트 ID는 다음과 같습니다.

1.3.6.1.4.1.2467.1.15.2.1.20

활성 연결 수는 로드 밸런스 서버 시스템에서 제공되는 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.

- **새 연결의 경우**, Cisco Consultant가 Cisco CSS Switch cntSvcExtMib에서 apCntsvcHits MIB 변수를 설정합니다. 각 서비스의 경우, Cisco Consultant는 다음을 수행합니다.
 - 색인에 해당 서비스를 가지고 있는 모든 apCntsvcHits의 합을 계산합니다.
 - 수신된 총 apCntsvcHits의 레코드를 보존합니다.
 - 델타를 계산합니다.

이 변수의 색인은 다음과 같습니다.

```
INDEX { apCntsvcOwnName, apCntsvcCntName, apCntsvcSvcName }
```

다음은 MIB 항목입니다.

```
apCntsvcHits OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "Total number of flows placed onto this service for this content rule."
DEFVAL { 0 }
```

```

--DEFAULT ap-display-name Hits
--DEFAULT apjam-popup-ref apCntSvcInst, Statistics
--DEFAULT apjam-chart-def cntSvcHitsChart, pie, apCntInst, "Hit Information Per Service:
--DEFAULT apjam-chart-item cntSvcHitsChart, getnext, apCntsvcSvcName
::= {apSvcEntry 20}

```

apCntsvcHits 오브젝트 ID는 다음과 같습니다.

1.3.6.1.4.1.2467.1.18.2.1.4

Cisco Consultant가중치

Cisco CSS Switch는 가중치 라운드 로빈 로드 밸런스를 사용하도록 구성해야 합니다. 수행 방법에 대한 정보는 *Content Services Switch 기본 구성 안내서*에서 "가중치 구성"을 참조하십시오.

가중치는 Cisco CSS Switch의 내부 카운터 및 권고자와 Metric Server의 피드백에 따른 관리 프로그램 기능에 의해 설정됩니다. 관리 프로그램을 실행하는 중 수동으로 가중치를 설정하려면 **lbcontrol server** 명령에서 **fixedweight** 옵션을 지정하십시오.

모든 서버가 작동 중지된 경우 모든 가중치는 0입니다. 이러한 경우, 모든 가중치가 0이기 때문에 서버가 요청을 처리하지 않으면 가중치가 1/2 가중 바운드로 설정되어 가능한 서버에서 처리 중인 요청의 동일한 기회를 허용합니다. 모니터에서는 올바른 가중치인 0을 표시합니다. 그러나 Cisco Consultant는 모든 다른 위치에서 1/2 가중 바운드의 가중치를 표시합니다.

가중치는 SNMP를 사용하는 Cisco CSS Switch로 전송됩니다. Cisco Consultant는 svcExt.mib에서 apSvcWeight를 설정합니다. 다음은 apSvcWeight 항목입니다.

```

apSvcWeight OBJECT-TYPE
SYNTAX Integer 32(1..10)
MAX-ACCESS read-create
STATUS current
DESCRIPTION
    "The service weight which is used in conjunction with load metrics when
    making load allocation decisions. The weight may be used to bias flows
    towards the specified service."
DEFVAL { 1 }
--DEFAULT ap-display-name Service Weight
--DEFAULT apjam-popup-ref apServicesGroupInst, Properties, Advanced
--DEFAULT apjam-wizard-field 2, normal
::= {apSvcEntry 16}

```

apSvcWeight 오브젝트 ID는 다음과 같습니다.

1.3.6.1.4.1.2467.1.15.2.1.12

가중치는 포트의 모든 서버에 적용됩니다. 특정 포트의 경우, 요청은 서로에 대해 상대적인 가중치에 따라 서버간에 분산됩니다. 예를 들어, 하나의 서버가 가중치 10으로 설정되고 다른 서버가 가중치 5로 설정되면 10으로 설정된 서버의 요청수는 5로 설정된 서버 요청수의 두 배가 됩니다.

모든 서버가 가질 수 있는 최대 가중치 경계를 지정하려면 **lbcontrol port set weightbound** 명령을 사용하십시오. 이 명령은 각 서버가 확보하는 요청수의 차이를 지정합니다. 최대 가중치를 1로 설정할 경우, 모든 서버의 가중치는 1, 일시 중단 경우에는 0, 단절 표시인 경우에는 -1입니다. 이 숫자가 증가할수록 서버의 가중치 차이도 증가합니다. 최대 가중치가 2이면 한 서버가 다른 서버 요청수의 두 배를 확보할 수 있습니다.

서버가 오프라인인 경우...

권고자가 서버가 오프라인임을 발견하면 이를 관리 프로그램에 알려 서버의 가중치를 0으로 설정합니다. 서버의 가중치가 0보다 크면 가중치가 Cisco CSS Switch로 전송되고 서버가 활성화되며, 서버 가중치가 0보다 작거나 같으면 서버가 일시 중단됩니다. 서비스를 활성화하고 일시중단시키는 작업은 Cisco CSS Switch svcExt.mib에 apSvcEnable MIB 변수를 설정하여 수행할 수 있습니다. 다음은 apSvcEnable MIB 항목입니다.

```
apSvcEnable OBJECT-TYPE
SYNTAX Integer
           disable(0)
           enable(1)
MAX-ACCESS read-create
STATUS current
DESCRIPTION
    "The state of the service, either enabled or disabled."
DEFVAL { disable }
--DEFAULT ap-display-name Status
--DEFAULT apjam-popup-ref apServicesGroupInst, Properties
--DEFAULT apjam-wizard-field 2, normal
 ::= {apSvcEntry 12}
```

apSvcEnable 오브젝트 ID는 다음과 같습니다.

1.3.6.1.4.1.2467.1.15.2.1.16

제15장 Network Dispatcher 작동 및 관리

주: 이 장을 읽을 때, 구성요소에 특정되지 않은 일반 절에서 Dispatcher 구성요소를 사용하지 않는 경우에는 "ndcontrol"과 "ndserver"를 다음으로 대체하십시오.

- CBR의 경우 **cbrcontrol** 및 **cbrserver** 사용
- Mailbox Locator의 경우 **mlcontrol** 및 **mlserver** 사용
- Site Selector의 경우 **cbrcontrol** 및 **ssserver** 사용
- Cisco Consultant의 경우 **lbcontrol** 및 **lbserver** 사용

이 장에서는 Network Dispatcher의 작동 및 관리 방법을 설명하며 다음 절이 수록되어 있습니다.

- 『원격 인증 관리』
- 231 페이지의 『Network Dispatcher 로그 사용』
- 233 페이지의 『Dispatcher 구성요소 사용』
 - 235 페이지의 『Dispatcher 구성요소에 Simple Network Management Protocol 사용』
- 242 페이지의 『Content Based Routing 구성요소 사용』
- 243 페이지의 『Mailbox Locator 구성요소 사용』
- 244 페이지의 『Site Selector 구성요소 사용』
- 244 페이지의 『Cisco Consultant 구성요소 사용』

원격 인증 관리

Network Dispatcher는 Network Dispatcher 서버를 실행하지 않는 시스템에서 구성 프로그램을 실행하는 옵션을 제공합니다.

구성 프로그램(ndcontrol, cbrcontrol, mlcontrol, sscontrol, lbcontrol, ndwizard, cbrwizard, mlwizard, sswizard, ndadmin) 간의 통신은 Java RMI(원격 메소드 호출) 호출을 사용하여 수행됩니다. 원격 관리를 위한 Network Dispatcher 시스

템에 연결하기 위한 명령은 **ndcontrol host:remote_host**입니다. RMI 호출이 로컬 시스템이 아닌 시스템에서 수신되면, 공용 키/개인용 키 인증 순서는 구성 명령을 승인하기 전에 발생해야 합니다.

구성요소 서버와 같은 시스템에서 실행 중인 제어 프로그램간의 통신은 인증되지 않습니다.

다음 명령을 사용하여 원격 인증에 사용할 공용 키와 개인용 키를 생성하십시오.

ndkeys [create|delete]

이 명령은 Network Dispatcher와 동일한 시스템에서만 실행됩니다.

작성 옵션을 사용하면 서버 키 디렉토리(**...nd/servers/key/**)에 공용 키가 작성되고, 각 Network Dispatcher 구성요소의 관리 키 디렉토리(**...nd/admin/keys/**)에 개인용 키가 작성됩니다. 개인용 키의 파일 이름은 *component-ServerAddress-RMIport*입니다. 이들 개인용 키는 원격 클라이언트로 전송되고 관리 키 디렉토리에 저장되어야 합니다.

각 구성요소에 기본 RMI 포트를 사용하는 호스트 이름 주소가 10.0.0.25인 Network Dispatcher의 경우, **ndkeys create** 명령은 다음 파일을 생성합니다.

- 공용 키: **.../nd/servers/key/authorization.key**
- 개인용 키:
 - **.../nd/admin/keys/dispatcher-10.0.0.25-10099.key**
 - **.../nd/admin/keys/cbr-10.0.0.25-11099.key**
 - **.../nd/admin/keys/m1-10.0.0.25-13099.key**
 - **.../nd/admin/keys/ss-10.0.0.25-12099.key**
 - **.../nd/admin/keys/lbc-10.0.0.25-14099.key**

관리 파일 세트는 다른 시스템에 설치되었습니다. 개인용 키 파일은 원격 클라이언트 시스템의 **.../nd/admin/keys** 디렉토리에 저장해야 합니다.

이제 10.0.0.25에 Network Dispatcher를 구성할 수 있는 권한이 원격 클라이언트에게 부여됩니다.

이들 동일한 키는 10.0.0.25에 Network Dispatcher의 구성 권한을 부여할 모든 원격 클라이언트에 사용되어야 합니다.

ndkeys create 명령을 다시 실행하게 되면, 새로운 공용/개인용 키 집합이 생성됩니다. 이것은 이전 키를 사용하여 연결하려고 시도했던 원격 클라이언트에게 권한이 부여되지 않았음을 의미합니다. 새로운 키는 다시 권한을 부여할 클라이언트의 올바른 디렉토리에 저장되어야 합니다.

ndkeys delete 명령은 서버 시스템에서 공용 및 개인용 키를 삭제합니다. 이러한 키가 삭제되면, 원격 클라이언트에는 서버를 연결할 권한이 없습니다.

ndkey 작성과 ndkey 삭제에는 모두 강제 옵션이 있습니다. 강제 실행 옵션은 기존 키를 삭제하거나 겹쳐쓰려고 하는 경우, 질문 명령 프롬프트를 억제합니다.

Network Dispatcher 로그 사용

Network Dispatcher는 서버 로그, 관리 프로그램 로그, 메트릭 모니터 로그(Metric Server 에이전트와의 통신 로그) 및 사용하는 각 권고자 로그에 항목을 게시합니다.

주: 또한 Dispatcher 구성요소에 대해서만 항목을 서브에이전트(SNMP) 로그로 만들 수 있습니다.

로그 레벨을 설정하여 로그에 기록되는 메시지들의 확장성을 정의할 수 있습니다. 레벨 0에서 오류가 기록되며, Network Dispatcher는 한 번만 발생하는 이벤트의 레코드와 헤더도 기록합니다(예: 관리 프로그램 로그에 기록되기 시작하는 권고자에 대한 메시지). 레벨 1에는 진행 중인 정보가 포함되고, 레벨 5에는 필요할 때 문제점의 디버깅을 도와주는 모든 생성 메시지가 포함됩니다. 서버 로그의 기본값은 0입니다. 관리 프로그램, 권고자 및 서브에이전트 로그의 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.

- 서버 로그의 로그 레벨이나 최대 로그 크기를 구성하려면 **ndcontrol set** 명령을 사용하십시오.
- 관리 프로그램 로그의 로그 레벨이나 최대 로그 크기를 구성하려면 **ndcontrol manager** 명령을 사용하십시오. 또한 이 명령은 Metric Server 에이전트와의 통신을 로그하는 메트릭 모니터 로그의 로그 레벨을 제어할 수도 있습니다.
- 권고자 로그의 로그 레벨이나 최대 로그 크기를 구성하려면 **ndcontrol advisor** 명령을 사용하십시오.
- 서브에이전트 로그의 로그 레벨이나 최대 로그 크기를 구성하려면 **ndcontrol subagent** 명령을 사용하십시오(Dispatcher 구성요소에서만 SNMP 서브에이전트를 사용합니다).

로그 파일 경로 변경

기본적으로 Network Dispatcher에서 생성한 로그는 Network Dispatcher 설치의 로그 디렉토리에 저장됩니다. 이 경로를 변경하려면, ndserver 스크립트의 *nd_logdir* 변수를 설정하십시오.

AIX, Linux 및 Solaris: ndserver 스크립트는 /usr/bin 디렉토리에 있습니다. 이 스크립트에서 *nd_logdir* 변수는 기본 디렉토리로 설정됩니다. 이 변수를 수정하여 사용자의 로그 디렉토리를 지정할 수 있습니다. 예를 들어,

```
ND_LOGDIR=/path/to/my/logs/
```

Windows 2000: ndserver 파일은 보통 Windows 2000 시스템 디렉토리 C:\WINNT\SYSTEM32에 있습니다. ndserver 파일에서 *nd_logdir* 변수는 기본 디렉토리로 설정됩니다. 이 변수를 수정하여 사용자의 로그 디렉토리를 지정할 수 있습니다. 예를 들어,

```
set ND_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 구성요소에 적용되지 않습니다.

Network Dispatcher의 2진 로그 기능은 기타 로그 파일과 동일한 로그 디렉토리를 사용합니다. 223 페이지의 『서버 통제를 분석하기 위해 2진 로그 사용』을 참조하십시오.

Dispatcher 구성요소 사용

이 절에서는 Dispatcher 구성요소의 작동 및 관리 방법을 설명합니다.

Dispatcher 시작 및 정지

- 명령행에 **ndserver**를 입력하여 Dispatcher를 시작하십시오.
- 명령행에 **ndserver stop**을 입력하여 Dispatcher를 정지하십시오.

활동해제 제한 시간 값 사용

Network Dispatcher의 경우, 활동해제 제한 시간에 지정된 시간(초) 동안 해당 연결에 활동이 없으면 연결이 활동해제된 것으로 간주됩니다. 초 수가 활동 없음을 초과하면 Network Dispatcher가 테이블에서 연결 레코드를 제거하고 해당 연결에 대한 후속 통신이 삭제됩니다.

예를 들어, 포트 레벨에서 **ndcontrol port set staletimeout** 명령에 활동해제 제한 시간 값을 지정할 수 있습니다.

활동해제 제한 시간은 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 제한 시간 값이 있습니다. 예를 들어, 텔넷 포트 23은 기본값이 32,000,000초입니다.

일부 서비스에도 자체의 활동해제 제한 시간 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.

Network Dispatcher의 활동해제 제한 시간 값이 서비스의 시간 종료 값보다 작을 때 연결 가능성 문제점이 발생할 수 있습니다. LDAP의 경우, Network Dispatcher 활동해제 제한 시간의 기본값은 300초입니다. 300초 동안 연결에 활동이 없으면 Network Dispatcher는 연결 레코드를 테이블에서 제거합니다. idletimeout 값이 300초 이상(또는 0으로 설정)이면 클라이언트는 아직 서버에 연결이 있다고 간주합니다. 클라이언트가 패킷을 전송하면 Network Dispatcher가 패킷을 버립니다. 그러면, 서버에 대한 요청이 작성될 때 LDAP가 정지합니다. 이 문제점을 해결하려면 LDAP idletimeout을 Network Dispatcher 활동해제 제한 시간 값과 작거나 같은, 0이 아닌 값으로 설정하십시오.

FIN 계수를 사용하여 가비지 컬렉션 제어

클라이언트는 서버가 트랜잭션의 종료를 알 수 있도록 모든 패킷을 전송한 후 FIN 패킷을 전송합니다. Dispatcher가 FIN 패킷을 수신하면, 그 트랜잭션을 활성 상태에서 완료 상태로 표시합니다. 트랜잭션이 FIN으로 표시되면, 연결에 예약된 메모리는 실행 프로그램에 구축된 가비지 콜렉터가 지울 수 있습니다.

FIN 시간 종료와 계수를 사용하여 실행 프로그램에서 가비지 컬렉션을 수행하는 횟수와 수행해야 할 양을 설정할 수 있습니다. 실행 프로그램에서는 정기적으로 할당된 연결의 목록을 확인합니다. 완료 상태의 연결 수가 FIN 계수보다 크거나 같으면, 실행 프로그램에서는 이 연결 정보를 보유하기 위해 사용되는 메모리를 해제하고자 합니다. **ndcontrol executor set fincount** 명령을 입력하여 FIN 계수를 변경할 수 있습니다.

가비지 콜렉터는 완료 상태에 있고 FIN 시간 종료에 지정된 시간(초 단위) 이전의 연결에 대한 메모리를 해제합니다. FIN 시간 종료는 **ndcontrol executor set fintimeout** 명령을 입력하여 변경할 수 있습니다.

활동해제 제한 시간 값은 연결이 제거되기 전까지 연결에 활동이 없는 시간(초)입니다. 233 페이지의 『활동해제 제한 시간 값 사용』에서 자세한 정보를 참조하십시오. FIN 계수는 “활동해제” 상태의 연결이 제거되는 횟수에도 영향을 줍니다. Dispatcher 시스템에 메모리가 거의 없으면, FIN 수를 더 낮게 설정해야 합니다. 자주 사용하는 웹 사이트가 있으면, FIN 수를 더 높게 설정해야 합니다.

GUI 보고 -- 모니터 메뉴 옵션

실행 프로그램에서 확인하고 관리 프로그램으로 릴레이하는 정보를 기반으로 다양한 도표를 표시할 수 있습니다. (GUI 모니터 메뉴 옵션은 관리자 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 구성요소에 Simple Network Management Protocol 사용

주: Linux의 경우, SNMP가 Network Dispatcher에서 지원되지 않습니다.

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 스테이션과 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 **..nd/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 MIB값
을 검토합니다. 그런 후 지정된 임계치가 초과되었는지 여부를 사용자에게 알릴 수
있습니다. 그러면, Dispatcher의 구성 데이터를 수정함으로써 Dispatcher 성능에 영
향을 주어 Dispatcher 또는 웹 서버에 장애가 발생하기 전에 미리 Dispatcher 문
제점을 조정 또는 수정할 수 있습니다.

SNMP 명령 및 프로토콜

시스템에서는 대개 각 네트워크 관리 스테이션마다 하나의 SNMP 에이전트를 제
공합니다. 사용자는 GET 명령어를 SNMP 에이전트로 전송합니다. 그 다음 이
SNMP 에이전트는 GET 명령을 전송하여 이러한 MIB 변수에 책임이 있는 서브
에이전트로부터 지정된 MIB 변수값을 검색합니다.

Dispatcher는 MIB 데이터를 갱신하고 검색하는 서브에이전트를 제공합니다. 서브
에이전트는 SNMP 에이전트가 GET 명령을 전송할 때 해당 MIB 데이터로 응답
합니다. SNMP 에이전트는 데이터를 네트워크 관리 스테이션에 전달합니다. 네트
워크 관리 스테이션에서는 사용자에게 지정된 임계치의 초과 여부를 알릴 수 있습
니다.

Dispatcher SNMP 지원에는 DPI(분산 프로토콜 인터페이스) 기능을 사용하는
SNMP 서브에이전트가 포함됩니다. DPI는 SNMP 에이전트와 그 서브에이전트간
의 인터페이스입니다.

AIX 및 Solaris에서 SNMP 사용 가능

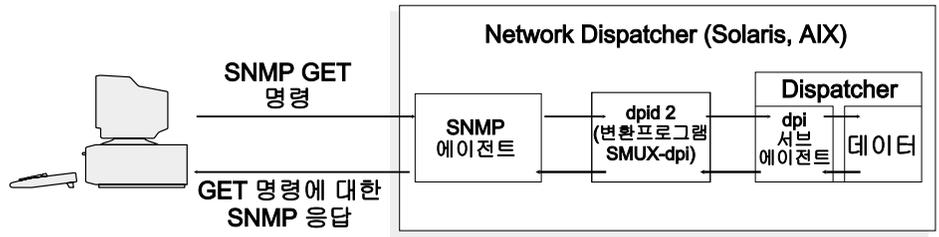


그림 28. AIX 및 Solaris용 SNMP 명령

AIX는 SNMP 멀티플렉서 프로토콜(SMUX)을 사용하는 SNMP 에이전트를 제공하고, DPI와 SMUX 간의 변환기로서 작동하는 추가 실행 파일 DPID2를 제공합니다.

Solaris의 경우, SNMP 에이전트를 제공하지 않으므로 사용자에게는 SMUX 사용 가능한 SNMP 에이전트가 있어야 합니다. Network Dispatcher는 /opt/nd/servers/samples/SNMP 디렉토리에서 Solaris용 DPID2를 제공합니다.

DPI 에이전트는 루트 사용자로서 실행되어야 합니다. DPID2 디먼을 실행하기 전에, 다음과 같이 /etc/snmpd.peers 파일 및 /etc/snmpd.conf 파일을 갱신하십시오.

- /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
```

/etc/snmpd.conf 파일을 다시 읽도록 snmpd를 화면갱신하십시오.

```
refresh -s snmpd
```

DPID SMUX 피어를 시작하십시오.

```
dpid2
```

디먼은 다음 순서로 시작되어야 합니다.

1. SNMP 에이전트
2. DPI 변환기

3. Dispatcher 서브에이전트

Windows 2000에서 SNMP 사용 가능

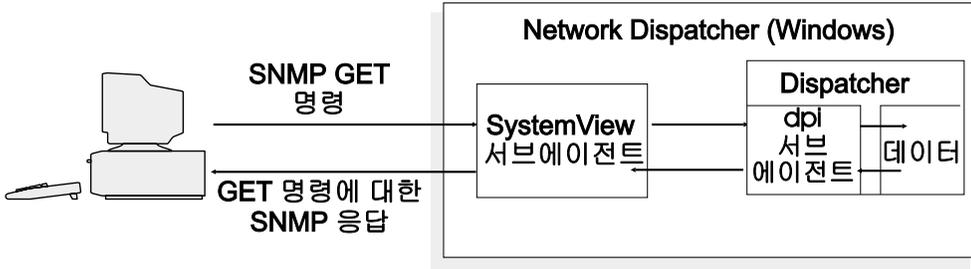


그림 29. Windows 2000용 SNMP 명령

Windows 2000용 DPI 가능 SNMP 에이전트를 받으려면, <http://www.tivoli.com/support/sva>에서 IBM SystemView Agent 툴킷의 Windows NT 버전을 설치하십시오.

SystemView SNMPD 프로세스를 시작하기 전에 Microsoft Windows SNMP 지원을 사용 불가능하게 설정해야 합니다. SystemView snmpd는 DPI 서브에이전트와 Microsoft 준수 에이전트를 지원합니다.

Windows SNMP 지원을 사용하지 않으려면 다음을 수행하십시오.

1. 시작 -> 프로그램 -> 관리 도구 -> 서비스를 누르십시오.
2. **SNMP**를 마우스 오른쪽 단추로 누른 다음 등록 정보를 선택하십시오.
3. 시작 유형:을 "사용 불가능"으로 변경하십시오.

주: Microsoft Windows SNMP 지원을 사용 불가능하게 하지 않으면 Dispatcher SNMP 서브에이전트는 snmpd 에이전트에 연결될 수 없습니다.

SystemView SNMP 에이전트를 구성하려면, 『SNMP 공동체 이름 제공』의 지침을 따르십시오.

SNMP 공동체 이름 제공

SNMP 공동체 이름을 구성하십시오. 기본 SNMP 공동체 이름은 public입니다. UNIX 시스템에서는 /etc/snmpd.conf라는 파일에서 설정됩니다.

공동체 이름은 모든 시스템에서 일관성있게 구성되고 사용되어야 합니다. 즉, Network Dispatcher의 공동체 이름이 SNMP 에이전트 구성에서 "OurCommunity"로 설정되면 서브에이전트 구성에서도 "OurCommunity"로 설정되어야 합니다.

Windows 2000의 경우, 공동체 이름을 작성하기 전에 IBM SystemView SNMP 에이전트를 구성하십시오.

1. 데스크탑에서 **IBM SystemView Agent** 아이콘을 누르십시오.
2. **snmpcfg**를 누르십시오.
3. SNMP 구성 대화 상자에서 공동체 이름을 추가하십시오. 테스트용으로 **public**을 공동체 이름으로 입력하십시오.
이 단계로 임의의 네트워크의 모든 호스트는 SNMP MIB 변수에 액세스할 수 있습니다. 이러한 값이 작동되는지 확인한 후에는 사용자 요구사항에 따라 값을 변경할 수 있습니다.
4. `\sva\dmi\bin\svastart.bat` 파일을 확인하고 **-dpi** 옵션이 지정되었는지 확인하십시오.
5. `\sva\dmi\bin` 하위 디렉토리에서 `svastart.bat`를 사용하여 SNMP 디먼을 시작하십시오.

실행 프로그램을 실행하면서 **ndcontrol subagent start [communityname]** 명령을 사용하여 Dispatcher DPI 서브에이전트와 SNMP 에이전트 간에 사용된 공동체 이름을 정의하십시오. 기본 공동체 이름은 `public`입니다. 이 값을 변경하면 새 공동체 이름도 위와 마찬가지로 `snmpcfg`를 사용하여 SystemView Agent에 추가해야 합니다.

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 csAddr, psNum, ssAddr 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 Object ID의 csAddr, psNum 부분에 의해 지정된 포트의 maxhalfopen 임계치를 초과했음을 나타냅니다. 포트에서 구성된 서버 수가 트랩에서 전송됩니다. 이 트랩은 Network Dispatcher에서 서비스 거부 중지가 발생 중임을 나타냅니다.

indDOSAttackDone 트랩은 numhalfopen, 즉 SYN 패킷만으로 구성된 반개방 연결의 수가 Object Identifier의 csAddr, psNum 부분에 의해 지정된 포트의 maxhalfopen 임계치보다 작음을 나타냅니다. 포트에서 구성된 서버 수가 트랩 내

에서 전송됩니다. Network Dispatcher가 가능한 서비스 거부 중지가 종료된 것을 판별하면 indDOSAttack 트랩이 전송된 후 이 트랩이 전송됩니다.

SMUX API의 제한으로 인해, ibmNetDispatcher 서브에이전트로부터 트랩에 보고되는 기업 ID는 기업 ID ibmNetDispatcher, 1.3.6.1.4.1.2.6.144가 아닌 dpid2의 기업 ID일 수 있습니다. 그러나 데이터에 ibmNetDispatcher MIB 내의 오브젝트 ID가 포함되므로, 데이터에 SNMP 관리 유틸리티는 트랩의 소스를 판별할 수 있습니다.

ndcontrol 명령으로 SNMP 지원 작동 및 작동 중지

ndcontrol subagent start 명령으로 SNMP 지원이 작동됩니다. **ndcontrol subagent stop** 명령으로 SNMP 지원이 작동 중지됩니다.

ndcontrol 명령에 대한 자세한 내용은 344 페이지의 『ndcontrol subagent -- SNMP 서브에이전트 구성』을 참조하십시오.

ipchain 또는 iptable을 사용하여 Network Dispatcher (Linux에서) box를 굳히는 모든 통신량을 거절함

Linux 커널에 ipchain이라는 방화벽 설비가 빌드됩니다. Network Dispatcher와 ipchain이 동시에 실행되면 Network Dispatcher이 먼저 패킷을 수신한 다음 ipchain이 수신합니다. 이렇게 하면 ipchain을 사용하여 Linux Network Dispatcher 상자(예: 방화벽을 로드 밸런싱하는데 사용하는 Network Dispatcher 상자)를 견고히 할 수 있습니다.

ipchain 또는 iptable이 완전히 제한적으로 구성된다면(인바운드 및 아웃바운드 통신량이 허용되지 않는다면), Network Dispatcher의 패킷 전달 부분은 정상적으로 기능합니다.

ipchain 및 iptable은 로드 밸런싱되기 전에 수신 통신량을 필터하는데 사용할 수 없습니다.

모든 Network Dispatcher가 정상적으로 작동하려면 추가 통신이 허용되어야 합니다. 이 통신의 몇 가지 예제가 나와 있습니다.

- 권고자는 Network Dispatcher 상자와 백엔드 서버 사이에서 통신합니다.

- Network Dispatcher는 백엔드 서버, 도달 목표 및 고가용성 파트너 Network Dispatcher 상자를 ping합니다.
- 사용자 인터페이스(그래픽 사용자 인터페이스, 명령행 및 마법사)는 RMI를 사용합니다.
- 백엔드 서버는 Network Dispatcher 상자에서 ping에 응답해야 합니다.

일반적으로 Network Dispatcher 상자에 대한 적절한 inchain 전략은 백엔드 서버와 연관된 것을 제외한 모든 통신량, 파트너 고가용성 Network Dispatcher, 모든 도달 목표 및 구성 호스트를 허용하지 않는 것입니다.

Content Based Routing 구성요소 사용

이 절에서는 Network Dispatcher의 CBR 구성요소를 작동 및 관리하는 방법을 설명합니다.

CBR 시작 및 정지

- 명령행에 **cbrserver**를 입력하여 CBR을 시작하십시오.
- 명령행에 **cbrserver stop**을 입력하여 CBR을 정지하십시오.

CBR과 Caching Proxy는 Caching Proxy 플러그인 API를 통해 배치되어 HTTP와 HTTPS(SSL) 요청을 처리합니다. 로드 밸런스 서버를 시작하려면, Caching Proxy가 동일한 시스템에서 실행되어야 합니다. 108 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 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』에 설명되어 있습니다. 몇 가지의 예제가 여기에 나열되어 있습니다.
- GUI(Graphical User Interface)를 사용하여 CBR을 구성하십시오. 명령행에 **ndadmin**을 입력하여 GUI를 여십시오. GUI를 사용한 CBR 구성 방법에 관한 자세한 정보는 98 페이지의 『GUI』를 참조하십시오.

CBR 로그 사용

CBR이 사용하는 로그는 Dispatcher에서 사용하는 로그와 유사합니다. 자세한 내용은 231 페이지의 『Network Dispatcher 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. 232 페이지의 『로그 파일 경로 변경』을 참조하십시오.

Mailbox Locator 구성요소 사용

Mailbox Locator 시작 및 정지

- 명령행에 **mlserver**를 입력하여 Mailbox Locator를 시작하십시오.
- 명령행에 **mlserver stop**을 입력하여 Mailbox Locator를 정지하십시오.

Mailbox Locator 제어

Mailbox Locator를 시작한 후에는 다음 방법을 사용하여 제어할 수 있습니다.

- **mlcontrol** 명령을 통해 Mailbox Locator를 구성하십시오. 이 명령의 완전한 구문은 277 페이지의 『부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조서』에 설명되어 있습니다. 몇 가지의 예제가 여기에 나열되어 있습니다.
- GUI(Graphical User Interface)를 사용하여 Mailbox Locator를 구성하십시오. 명령행에 **ndadmin**을 입력하여 GUI를 여십시오. GUI를 사용한 Mailbox Locator의 구성 방법에 대한 자세한 정보는 115 페이지의 『GUI』를 참조하십시오.

Mailbox Locator 로그 사용

Mailbox Locator에서 사용하는 로그는 Dispatcher에서 사용하는 로그와 유사합니다. 자세한 설명은 231 페이지의 『Network Dispatcher 로그 사용』을 참조하십시오.

Site Selector 구성요소 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 351 페이지의 『부록D. Site Selector 명령어 참조서』에 설명되어 있습니다. 몇 가지의 예제가 여기에 나열되어 있습니다.
- GUI(Graphical User Interface)를 사용하여 Site Selector를 구성하십시오. 명령행에 **ndadmin**을 입력하여 GUI를 여십시오. GUI를 사용한 Site Selector의 구성 방법에 대한 자세한 정보는 129 페이지의 『GUI』를 참조하십시오.

Site Selector 로그 사용

Site Selector가 사용하는 로그는 Dispatcher에서 사용된 로그와 비슷합니다. 자세한 설명은 231 페이지의 『Network Dispatcher 로그 사용』을 참조하십시오.

Cisco Consultant 구성요소 사용

Cisco Consultant 시작 및 정지

1. 명령행에 **lbcserver**를 입력하여 Cisco Consultant를 시작하십시오.
2. 명령행에 **lbcserver stop**을 입력하여 Cisco Consultant를 정지하십시오.

Cisco Consultant 제어

Cisco Consultant를 시작한 후에는 다음 방법을 사용하여 제어할 수 있습니다.

- **lbcontrol** 명령을 통해 Cisco Consultant를 구성하십시오. 이 명령의 완전한 구문은 381 페이지의 『부록E. Consultant for Cisco CSS Switches 명령어 참조서』에 설명되어 있습니다. 몇 가지의 예제가 여기에 나열되어 있습니다.

- GUI(Graphical User Interface)를 사용하여 Cisco Consultant를 구성하십시오. 명령행에 **ndadmin**을 입력하여 GUI를 여십시오. GUI를 사용한 Cisco Consultant의 구성 방법에 대한 자세한 정보는 129 페이지의 『GUI』를 참조하십시오.

Cisco Consultant 로그 사용

Cisco Consultant에서 사용하는 로그는 Dispatcher에서 사용하는 로그와 유사합니다. 자세한 설명은 231 페이지의 『Network Dispatcher 로그 사용』을 참조하십시오.

Metric Server 구성요소 사용

Metric Server 시작 및 정지

Metric Server는 Network Dispatcher에 서버 로드 정보를 제공합니다. Metric Server는 로드 밸런싱되는 각 서버에 있습니다.

- Metric Server가 있는 각각의 서버 시스템의 명령행에서 **metricserver start**를 입력하여 Metric Server를 시작하십시오.
- Metric Server가 있는 각각의 서버 시스템의 명령행에서 **metricserver stop**을 입력하여 Metric Server를 정지하십시오.

Metric Server 로그 사용

Metric Server 시동 스크립트의 로그 레벨을 변경하십시오. Network Dispatcher 로그의 로그 레벨 범위와 마찬가지로 0부터 5까지 범위에서 로그 레벨을 지정할 수 있습니다. 이렇게 하면 **...ms/logs** 디렉토리에 에이전트 로그가 생성됩니다.

제16장 문제점 해결

이 장은 Network Dispatcher와 관련된 문제점을 발견하고 해결하는 데 도움이 됩니다. 『문제점 해결 테이블』에는 발생할 수 있는 증상이 나와 있습니다.

문제점 해결 테이블

Dispatcher, CBR, Mailbox Locator, Site Selector 및 Consultant for Cisco CSS Switches에 대한 문제점 해결 테이블입니다.

표 14. Dispatcher 문제점 해결 테이블

증상	가능한 원인	이동 위치
Dispatcher가 제대로 실행되지 않음	포트 번호 충돌	252 페이지의 『Dispatcher 포트 번호 확인』
적절히 배치된 서버를 구성했으나 로드 밸런스 요청에 응답하지 않음	주소가 잘못되었거나 충돌함	256 페이지의 『문제점: Dispatcher 및 서버가 응답하지 않음』
클라이언트 시스템에서의 연결이 제공되지 않거나 연결이 시간 종료됨	<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 • 서버가 작동 중단되었거나 0의 가중치로 설정됨 	256 페이지의 『문제점: Dispatcher 요청이 밸런스를 이루지 않음』
클라이언트 시스템이 제공되지 않거나 시간 종료되었습니다.	고가용성이 작동하지 않음	257 페이지의 『문제점: Dispatcher 고가용성 기능이 작동하지 않음』

표 14. Dispatcher 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
핵심 메시지를 추가할 수 없음 (Windows 2000)	어댑터에서 출발지 주소가 구성 되지 않음	257 페이지의 『문제점: 핵심 메 시지를 추가할 수 없음 (Windows 2000)』
서버가 요청을 처리하지 않음 (Window)	다른 라우트가 라우팅 테이블에 작성되어 있음	258 페이지의 『문제점: 추가 라우트(Windows 2000)』
권고자가 광범위한 영역에서 제 대로 작동하지 않음	권고자가 원격 시스템에서 실행 중이 아님	258 페이지의 『문제점: 권고자 가 제대로 작동하지 않음』
SNMP가 시작되지 않거나 계 속 실행되지 않음(Windows 2000)	SNMP 명령에서 패스한 공동 체 이름이 서브에이전트가 시작 할 때 사용한 공동체 이름과 일치하지 않음	258 페이지의 『문제점: SNMP가 올바르게 실행되지 않음(Windows 2000)』
Dispatcher, Microsoft IIS 및 SSL이 작동하지 않거나 계속되 지 않음	프로토콜을 통해 암호화된 데이 터를 전송할 수 없음	258 페이지의 『문제점: Dispatcher, Microsoft IIS 및 SSL이 작동하지 않음 (Windows 2000)』
원격 시스템으로의 연결이 거부 됨	이전 버전의 키가 계속 사용됨	258 페이지의 『문제점: 원격 시스템에 대한 Dispatcher 연 결』
‘서버가 응답하지 않음’ 또는 ‘RMI 서버에 액세스할 수 없음’ 메시지로 ndcontrol 또는 ndadmin 명령이 실패함	1. 소켓이 연결된 스택으로 인해 명령이 실패했거나 ndserver 가 시작되지 않아서 실패했 습니다. 2. RMI 포트가 올바르게 설정 되지 않았음	259 페이지의 『문제점: ndcontrol 또는 ndadmin 명령 실패』
온라인 도움말을 보기 위해 기본 브라우저로서 Netscape를 실행할 때, “파일을 찾을 수 없습니다...” 라는 오류 메시지 발생 (Windows 2000)	HTML 파일 연관 설정이 잘못 됨	260 페이지의 『온라인 도움말을 보려고 할 때 문제점: “파일을 찾을 수 없습니다...”라는 오류 메시지 발생(Windows 2000)』
Solaris 2.7에서 ndserver를 시작 할 때 “stty: : 이러한 장치 또는 주소가 없음”이라는 오류 메시지 발생	이 오류 메시지를 무시하십시오. 문제가 되지 않습니다. ndserver 가 제대로 실행됩니다.	260 페이지의 『문제점: Solaris 2.7에서 ndserver를 시작할 때 유사 오류 메시지 발생』
그래픽 사용자 인터페이스가 울 바르게 시작하지 않음	페이징 공간이 충분하지 않음	260 페이지의 『문제점: GUI (Graphical User Interface)가 울 바로 시작되지 않음』

표 14. Dispatcher 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
Caching Proxy가 설치된 Dispatcher 실행 중에 오류 발생	Caching Proxy 파일 중속성	260 페이지의 『문제점: Caching Proxy가 설치된 Dispatcher 실행 중 오류』
그래픽 사용자 인터페이스가 올바르게 표시되지 않음	해상도가 올바르지 않음	261 페이지의 『문제점: GUI (Graphical User Interface)가 올바르게 표시되지 않음』
때때로 도움말 패널이 다른 창 뒤로 사라짐	Java 한계	261 페이지의 『문제점: Windows 2000의 경우, 도움말 창이 다른 열린 창 뒤로 사라짐』
Network Dispatcher가 프레임을 처리할 수 없고 전달할 수 없음	NIC마다 고유한 MAC 주소가 필요함	261 페이지의 『문제점: Network Dispatcher가 프레임을 처리하고 전달할 수 없음』
파란색 화면이 나타남	설치 및 구성된 네트워크 카드가 없음	262 페이지의 『문제점: Network Dispatcher 실행 프로그램을 시작할 때 파란색 화면이 표시됨』
Discovery 경로로 인해 리턴 통신량이 발생하지 못함	루프백에서 클러스터 별명이 지정됨	262 페이지의 『문제점: Discovery 경로로 인해 Network Dispatcher와의 리턴 통신이 발생하지 못함』
권고자는 모든 서버가 중단되었음을 표시함	TCP 체크섬이 올바르게 계산되지 않음	263 페이지의 『문제점: 권고자는 모든 서버가 중단되었음을 표시함』
Network Dispatcher의 광역 모드의 고가용성이 작동되지 않음	원격 Dispatcher가 로컬 Dispatcher의 클러스터에서 서버로 정의되어야 함	263 페이지의 『문제점: Network Dispatcher의 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.	264 페이지의 『문제점: GUI는 큰 구성 파일을 로드하려고 할 때 정지하거나 예기치 못한 작동을 합니다.』

표 15. CBR 문제점 해결 테이블

증상	가능한 원인	이동 위치
CBR이 제대로 실행되지 않습니다.	포트 번호 충돌	253 페이지의 『CBR 포트 번호 확인』
‘서버가 응답하지 않음’ 또는 ‘RMI 서버에 액세스할 수 없음’ 메시지로 cbrcontrol 또는 ndadmin 명령이 실패함	소켓이 연결된 스택으로 인해 명령이 실패했거나 cbrserver가 시작되지 않아서 명령이 실패했음	265 페이지의 『문제점: cbrcontrol 또는 ndadmin 명령 실패』
요청이 로드 밸런스되지 않음	실행 프로그램이 시작되기 전에 Caching Proxy가 시작됨	266 페이지의 『문제점: 요청이 로드 밸런스되지 않음』
Solaris에서, ‘오류: 실행 프로그램이 시작되지 않습니다.’라는 메시지로 cbrcontrol executor start 명령이 실패함	시스템 IPC 기본값이 수정되어야 하므로 명령이 실패함	266 페이지의 『문제점: Solaris에서 cbrcontrol executor start 명령 실패』
URL 규칙이 작동하지 않음	구문론적 또는 구성 오류	266 페이지의 『문제점: 구문 또는 구성 오류』

표 16. Mailbox Locator 문제점 해결 테이블

증상	가능한 원인	이동 위치
Mailbox Locator가 올바르게 실행되지 않음	포트 번호 충돌	254 페이지의 『Mailbox Locator 포트 번호 확인』
mlserver 명령이 “java.rmi.RMI 보안 예외: security.f.d.read” 예외를 리턴함	파일 설명자에 대한 시스템의 한계가 mlserver가 서비스할 요청수에 비해 너무 작음	267 페이지의 『문제점: mlserver 명령이 정지됨』
‘서버가 응답하지 않음’ 또는 ‘RMI 서버에 액세스할 수 없음’ 메시지로 mlcontrol 또는 ndadmin 명령이 실패함	소켓이 연결된 스택으로 인해 명령이 실패했거나 또는 mlserver가 시작되지 않아서 명령이 실패했습니다.	267 페이지의 『문제점: mlcontrol 또는 ndadmin 명령 실패』
포트를 추가할 수 없음	다른 응용프로그램이 이미 해당 포트를 인식하고 있음	268 페이지의 『문제점: 포트를 추가할 수 없음』
포트를 추가하려고 할 때 프록시 오류 수신	프록시가 시작되기 전에 NIC에서 클러스터 주소가 구성되지 않았거나 다른 응용프로그램이 해당 포트에서 실행 중임	268 페이지의 『문제점: 포트 추가를 시도할 때 프록시 오류 발생』

표 17. Site Selector 문제점 해결 테이블

증상	가능한 원인	이동 위치
Site Selector가 올바르게 실행되지 않음	포트 번호 충돌	255 페이지의 『Site Selector 포트 번호 확인』
Site Selector가 Solaris 클라이언트로부터 수신하는 요청을 라운드 로빈(round-robin)하지 않음	Solaris 시스템이 "이름 서비스 캐시 디먼"을 실행함	269 페이지의 『문제점: Site Selector가 Solaris 클라이언트로부터 통신을 라운드 로빈(round-robin)하지 않음』
'서버가 응답하지 않음' 또는 'RMI 서버에 액세스할 수 없음' 메시지로 sscontrol 또는 ndadmin 명령이 실패함	소켓이 연결된 스택으로 인해 명령이 실패했거나 ssserver가 시작되지 않아서 명령이 실패했음	269 페이지의 『문제점: sscontrol 또는 ndadmin 명령 실패』
sssriver가 Windows 2000에서 시작에 실패함	Windows에서는 호스트 이름이 DNS에 있을 필요가 없습니다.	269 페이지의 『문제점: sssriver가 Windows 2000에서 시작에 실패함』
시스템이 중복 라우트를 통해 올바르게 로드 밸런스되지 않음 - 이름 분석에 실패	여러 어댑터가 있는 Site Selector가 동일한 서브넷에 연결됨	270 페이지의 『문제점: Site Selector가 중복 라우트를 통해 올바르게 로드 밸런스하지 않음』

표 18. Consultant for Cisco CSS Switches 문제점 해결 테이블

증상	가능한 원인	이동 위치
lbcserver가 시작되지 않음	포트 번호 충돌	255 페이지의 『Cisco Consultant 포트 번호 확인』
'서버가 응답하지 않음' 또는 'RMI 서버에 액세스할 수 없음' 메시지로 lbcontrol 또는 ndadmin 명령이 실패함	소켓이 연결된 스택으로 인해 명령이 실패했거나 lbcserver가 시작되지 않아서 명령이 실패했음	270 페이지의 『문제점: lbcontrol 또는 ndadmin 명령 실패』
수신 오류: 포트 14099에서 레지스트리를 작성할 수 없음	제품 사용권이 만기됨	270 페이지의 『문제점: 포트 14099에서 레지스트리를 작성할 수 없음』

표 19. Metric Server 문제점 해결 테이블

증상	가능한 원인	이동 위치
.bat 또는 .cmd 사용자 메트릭 파일을 실행하는 Windows 2000의 Metric Server IOException	전체 메트릭 이름이 필요함	271 페이지의 『문제점: .bat 또는 .cmd 사용자 메트릭 파일을 실행하는 Windows 2000의 Metric Server IOException』

표 19. Metric Server 문제점 해결 테이블 (계속)

증상	가능한 원인	이동 위치
Metric Server는 Network Dispatcher 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 	271 페이지의 『문제점: Metric Server가 Network Dispatcher 시스템에 로드를 보고하지 않음』
Metric Server 로그는 키 파일이 서버로 전송될 때 "에이전트에 액세스하려면 서명이 필요합니다"라고 보고합니다.	키 파일이 파손되어 권한 부여에 실패했습니다.	272 페이지의 『문제점: Metric Server 로그가 "에이전트에 액세스하려면 서명이 필요합니다"라고 보고합니다.』

Dispatcher 포트 번호 확인

Dispatcher를 실행하는 데 문제점이 발생한 적이 있으면, 응용프로그램 중의 하나가 Dispatcher가 일반적으로 사용하는 포트 번호를 사용하고 있을 수 있습니다. Dispatcher 서버는 다음과 같은 포트 번호를 사용합니다.

- 10099: ndcontrol에서 명령을 수신할 경우
- 10004: Metric Server에 메트릭 조회를 전송할 경우
- 10005: SDA 에이전트에서 정보를 수신할 경우

다른 응용프로그램이 Dispatcher 포트 번호 중 하나를 사용하는 경우, 다음을 수행하여 Dispatcher의 포트 번호를 변경할 수 있습니다.

- 명령을 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - ndserver 파일 맨 위에 있는 ND_RMIPORT 변수를 Dispatcher가 명령을 받을 포트 번호로 수정하십시오.
- Metric Server로부터 메트릭 보고서를 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - metricserver 파일의 RMI_PORT 변수를 Dispatcher가 Metric Server와 통신할 포트 번호로 수정하십시오.

- 관리 프로그램이 시작될 때 `metric_port` 인수를 제공하십시오. **ndcontrol manager start** 명령 구문에 대한 설명은 309 페이지의 『ndcontrol manager -- 관리 프로그램 제어』를 참조하십시오.

- SDA 정보를 받는 데 사용할 포트를 변경하려면, `ndserver` 파일의 `ND_AFFINITY_PORT` 변수를 Dispatcher가 SDA 정보를 받는 데 사용할 포트로 변경하십시오.

주: Windows 2000의 경우, `ndserver`와 `metricsserver` 파일은 `c:\winnt\system32` 디렉토리에 있습니다. 다른 플랫폼의 경우, 파일이 `/usr/bin/` 디렉토리에 있습니다.

CBR 포트 번호 확인

CBR 실행에 대한 문제점이 발생한 경우, 응용프로그램 중 하나가 흔히 CBR에서 사용하는 포트 번호를 사용하고 있을 수 있습니다. CBR은 다음과 같은 포트 번호를 사용합니다.

- 11099: `cbrcontrol`에서 명령을 수신할 경우
- 10004: Metric Server에 메트릭 조회를 전송할 경우

다른 응용프로그램에서 CBR 포트 번호 중 하나를 사용 중이면, 다음을 수행하여 CBR의 포트 번호를 변경할 수 있습니다.

- 명령을 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - `cbrserver` 파일 맨 위의 `ND_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 포트로 수정하십시오.
 - 관리 프로그램이 시작될 때 `metric_port` 인수를 제공하십시오. **manager start** 명령 구문에 대한 설명은 309 페이지의 『ndcontrol manager -- 관리 프로그램 제어』를 참조하십시오.

주: Windows 2000의 경우, cbrserver와 metricserver 파일은 c:\winnt\system32 디렉토리에 있습니다. 다른 플랫폼의 경우, 파일이 /usr/bin/ 디렉토리에 있습니다.

Mailbox Locator 포트 번호 확인

Mailbox Locator 실행에 관한 문제점이 발생하는 경우, 응용프로그램의 하나가 Mailbox Locator에서 사용하는 포트 번호를 사용하고 있을 수 있습니다. Mailbox Locator는 반드시 다음 포트 번호를 사용해야 합니다.

- 13099: mlcontrol에서 명령을 받을 경우
- 10004: Metric Server에 메트릭 조회를 전송할 경우

다른 응용프로그램이 Mailbox Locator의 하나를 사용하고 있으면 다음을 수행하여 Mailbox Locator의 포트 번호를 변경할 수 있습니다.

- 명령을 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - mlserver 파일 맨 위의 ND_RMIPORT 변수를 Mailbox Locator가 명령을 받을 포트 Mailbox Locator로 수정하십시오.
- Metric Server로부터 메트릭 보고서를 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - metricserver 파일의 RMI_PORT 변수를 Mailbox Locator가 Metric Server와 통신할 포트로 수정하십시오.
 - 관리 프로그램이 시작될 때 metric_port 인수를 제공하십시오. **manager start** 명령 구문에 대한 설명은 309 페이지의 『ndcontrol manager -- 관리 프로그램 제어』를 참조하십시오.

주: Windows 2000의 경우, mlserver와 metricserver 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: ssscontrol에서 명령을 받을 경우
- 10004: Metric Server에 메트릭 조회를 전송할 경우

다른 응용프로그램이 Site Selector 포트 번호의 하나를 사용하고 있으면 다음을 수행하여 Site Selector의 포트 번호를 변경할 수 있습니다.

- 명령을 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - sssserver 파일 맨 위의 ND_RMIPORT 변수를 Site Selector가 명령을 받을 포트로 수정하십시오.
- Metric Server로부터 메트릭 보고서를 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 - metricsserver 파일의 RMI_PORT 변수를 Site Selector가 Metric Server와 통신할 포트로 수정하십시오.
 - 관리 프로그램이 시작될 때 metric_port 인수를 제공하십시오. **manager start** 명령 구문에 대한 설명은 361 페이지의 『sscontrol manager - 관리 프로그램 제어』를 참조하십시오.

주: Windows 2000의 경우, sssserver와 metricsserver 파일은 c:\winnt\system32 디렉토리에 있습니다. 다른 플랫폼의 경우, 파일이 /usr/bin/ 디렉토리에 있습니다.

Cisco Consultant 포트 번호 확인

Cisco Consultant 구성요소 실행에 대한 문제점이 발생하는 경우, 다른 응용프로그램이 Cisco Consultant의 lbserver에서 사용하는 포트 번호 중 하나를 사용하고 있을 수 있습니다. Cisco Consultant는 다음과 같은 포트 번호를 사용합니다.

- 14099: lbcontrol로부터 명령을 받을 경우
- 10004: Metric Server에 메트릭 조회를 전송할 경우

다른 응용프로그램이 컨설턴트 포트 번호 중 하나를 사용하고 있으면 다음을 수행하여 컨설턴트의 포트 번호를 변경할 수 있습니다.

- lbcontrol로부터 명령을 받을 포트를 변경하려면 lbserver 파일의 ND_RMIPORT 변수를 수정하십시오. 14099를 컨설턴트가 lbcontrol 명령을 받을 포트로 변경하십시오.
- Metric Server로부터 메트릭 보고서를 받는 데 사용할 포트를 변경하려면 다음을 수행하십시오.
 1. metricserver 파일의 RMI_PORT 변수를 수정하십시오. 10004를 컨설턴트가 Metric Server와 통신할 포트로 변경하십시오.
 2. 관리 프로그램이 시작될 때 metric_port 인수를 제공하십시오. lbcontrol manager start 명령 구문에 대한 설명은 397 페이지의 『lbcontrol manager -- 관리 프로그램 제어』를 참조하십시오.

주: Windows 2000의 경우, lbserver와 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 252 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.

문제점: Dispatcher 요청이 밸런스를 이루지 않음

이 문제점에는 클라이언트 시스템에서의 연결이 제공되지 않거나 연결이 시간 종료되는 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? `netstat -ni`를 사용하여 확인하십시오.
3. 각 서버에 있는 루프백 장치에 클러스터 주소로 설정된 별명이 있습니까? `netstat -ni`를 사용하여 확인하십시오.
4. 여분의 라우트가 삭제됩니까? `netstat -nr`를 사용하여 확인하십시오.
5. **ndcontrol cluster status** 명령을 사용하여 사용자가 정의한 각 클러스터에 대한 정보를 확인하십시오. 각 클러스터마다 포트가 정의되어 있어야 합니다.
6. **ndcontrol server report::** 명령을 사용하여 서버가 단절되어 있지도 않고 가중치가 0으로 설정되어 있지도 않은지 확인하십시오.

문제점: Dispatcher 고가용성 기능이 작동하지 않음

이 문제점은 Dispatcher 고가용성 환경이 구성되어 있으나, 클라이언트 시스템으로부터의 연결이 제공되지 않거나 시간 종료되고 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 193 페이지의 『스크립트 사용』을 참조하십시오.
- AIX, Linux 및 Solaris의 경우, goActive, goStandby, goInOp 스크립트에 실행 권한 세트가 있는지 확인하십시오.
- Windows 2000의 경우, 반드시 비전달 주소를 구성해야 합니다.

문제점: 핵심 메시지를 추가할 수 없음(Windows 2000)

Windows 2000 오류는 출발지 주소가 어댑터에 구성되지 않을 때 발생합니다. 문제점을 정정하거나 진단하려면, 다음을 확인하십시오.

- Windows 2000의 경우, 토큰링이나 이더넷 토큰링을 사용하고 다음 명령을 실행하여 비전달 주소를 구성해야 합니다.

```
ndconfig tr0 <ip address> netmask <netmask> or
ndcontrol cluster configure
```

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, 78 페이지의 『로드 밸런스에 대한 서버 시스템 설정』을 참조하십시오.

문제점: 권고자가 제대로 작동하지 않음

광역 지원을 사용 중이고 권고자가 제대로 작동하는 것으로 보이지 않으면, 권고자가 원격 및 원격 Dispatcher 모두에서 시작되었는지 확인하십시오. 179 페이지의 『광역 지원으로 원격 권고자 사용』을 참조하십시오.

문제점: SNMPD가 올바르게 실행되지 않음(Windows 2000)

SNMP 서브에이전트 사용 시 SystemViewAgent SNMP 디먼이 시작된 채로 남아있지 않으면, snmpcfg 프로그램을 사용하여 SNMP 공동체를 구성했는지 확인하십시오. Dispatcher 서브에이전트에서 SNMP 데이터에 액세스하려면, SNMP 명령에 전달된 공동체 이름은 서브에이전트가 시작된 공동체 이름과 일치해야 합니다.

문제점: Dispatcher, Microsoft IIS 및 SSL이 작동하지 않음(Windows 2000)

Dispatcher, Microsoft IIS 및 SSL 사용시 이들이 함께 작동하지 않으면, SSL 보안을 작동하는 데 문제가 있을 수 있습니다. 키 쌍 생성, 인증 확보, 키 쌍이 있는 인증 설치, SSL이 필요한 디렉토리 구성에 대한 자세한 정보는 Windows 2000 과 함께 제공되는 *Microsoft 정보 피어 웹 서비스 정보 및 계획 안내서*를 참조하십시오. 웹 브라우저에서 볼 수 있는 문서의 로컬 URL은 `file:///C:/WINNT/system32/inetsrv/iisadmin/htmldocs/inetdocs.htm`입니다.

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 포트

를 지정하면, 명령행은 발견된 처음 키만 시도합니다. 이 키가 틀렸으면, 연결은 거부됩니다. 틀린 키를 삭제하지 않는 한, 연결되지 않습니다.

문제점: **ndcontrol** 또는 **ndadmin** 명령 실패

1. **ndcontrol** 명령이 오류: 서버가 응답하지 않습니다를 리턴합니다. 또는 **ndadmin** 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 jre
EXCLUDE-MODULE jrew
EXCLUDE-MODULE javaw
```

2. Network Dispatcher 인터페이스의 관리 콘솔(명령행, 그래픽 사용자 인터페이스 및 마법사)은 **RMI**(원격 메소드 호출)를 사용하여 **ndserver**와 통신합니다. 기본 통신은 두 개의 포트를 사용하는데, 하나는 **ndserver** 시작 스크립트에, 다른 하나는 임의의 포트에 설정됩니다.

관리 콘솔 중 하나가 방화벽과 동일한 시스템 또는 방화벽을 통해 동일한 시스템에서 실행되는 경우, 임의의 포트는 문제를 유발할 수 있습니다. 예를 들어, Network Dispatcher를 방화벽과 동일한 시스템에서 실행하고, **ndcontrol** 명령을 발행하면 오류: 서버가 응답하지 않음과 같은 오류 메시지가 나타날 수 있습니다.

이러한 문제를 피하려면 **PATH**에 위치한 **ndserver** 스크립트를 편집하여 **RMI**가 사용하는 임의의 포트를 설정하십시오. 지정한 포트 *yourPort*가 있는 **END_ACCESS** 문자열에 **-DND_RMI_SERVER_PORT=*yourPort***를 포함시키십시오.

예를 들어,

```
END_ACCESS='-DND_CLIENT_KEYS_DIRECTORY=/usr/lpp/nd/admin/keys/dispatcher
-DND_SERVER_KEYS_DIRECTORY=/usr/lpp/nd/dispatcher/key
-DND_RMI_SERVER_PORT=10100'
ND_RMIPORT=10099
```

완료되면 **ndserver**를 재시작하고 포트 10099와 10100 또는 관리 콘솔이 실행될 호스트 주소에 대해 선택한 포트에 대해 통신을 시작하십시오.

3. 이러한 오류는 **ndserver**가 시작되지 않은 경우에도 발생할 수 있습니다.

온라인 도움말을 보려고 할 때 문제점: “파일을 찾을 수 없습니다...”라는 오류 메시지 발생(Windows 2000)

Windows 2000의 경우, 기본 브라우저로 Netscape를 사용할 때 이 문제점에 대해 “<filename>.html’ 파일(또는 구성요소 중 하나)을 찾을 수 없습니다. 경로와 파일 이름이 올바른지, 필요한 모든 라이브러리가 사용 가능한지 확인하십시오.”라는 오류 메시지가 나타납니다.

이 문제점은 HTML 파일 연관 설정이 잘못되어서 발생합니다. 해결책은 다음과 같습니다.

1. 내 컴퓨터를 누르고 도구를 누른 다음 폴더 옵션을 선택하고 파일 형식 탭을 누르십시오.
2. “Netscape 하이퍼텍스트 문서”를 선택하십시오.
3. 고급 단추를 누르고 열기를 선택한 다음 편집 단추를 누르십시오.
4. 응용프로그램: 필드에서 NSShell을 입력하고(작업 수행: 필드에 사용되는 응용 프로그램은 아님) 확인을 누르십시오.

문제점: Solaris 2.7에서 ndserver를 시작할 때 유사 오류 메시지 발생

Solaris 2.7 플랫폼에서 ndserver를 시작할 때, “stty: : 그러한 장치나 주소가 없음”이라는 유사 오류 메시지가 나타납니다. 이 오류 메시지를 무시하십시오. Ndserv가 올바르게 실행됩니다.

문제점: GUI(Graphical User Interface)가 올바르게 시작되지 않음

ndadmin인 GUI(Graphical User Interface)가 올바르게 작동하려면 충분한 페이지 공간이 필요합니다. 페이지 공간이 충분하지 않으면 GUI가 완전히 시작되지 않을 수 있습니다. 이런 경우에는 페이지 공간을 확인하고 필요하다면 늘리십시오.

문제점: Caching Proxy가 설치된 Dispatcher 실행 중 오류

Network Dispatcher를 설치 제거하여 다른 버전을 다시 설치하고, Dispatcher 구성요소를 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. Network Dispatcher를 설치 제거하십시오.
3. Network Dispatcher와 Caching Proxy를 다시 설치하십시오.

문제점: GUI(Graphical User Interface)가 올바르게 표시되지 않음

Network Dispatcher GUI의 모양과 관련된 문제점이 발생하면 운영 체제의 데스크탑 해상도에 대한 설정을 확인하십시오. GUI는 해상도 1024x768 픽셀에서 가장 잘 보입니다.

문제점: Windows 2000의 경우, 도움말 창이 다른 열린 창 뒤로 사라짐

처음 Windows 2000에서 도움말 창을 열 때, 때때로 기존 창 뒤의 백그라운드로 사라집니다. 이 경우에는 창을 눌러 다시 앞으로 가져오십시오.

문제점: Network Dispatcher가 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 `ibmnd.conf`에 구성된 것으로 Network Dispatcher에 지정되지 않고, `ibmnd.conf`에 정의되지 않은 NIC가 프레임을 받으면 Network Dispatcher가 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 hme0 ether 01:02:03:04:05:06
```

문제점: Network Dispatcher 실행 프로그램을 시작할 때 파란색 화면이 표시됨

Windows 2000에서 실행 프로그램을 시작하려면 네트워크 카드가 설치되고 구성되어야 합니다.

문제점: Discovery 경로로 인해 Network Dispatcher와의 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 AIX는 루프백에서 라우트를 작성합니다. 이는 해당 주소가 이 서브넷을 사용하여 별명이 지정된 마지막 인터페이스이기 때문에 발생합니다.

예를 들어, 클러스터가 9.37.54.69이고 255.255.255.0의 넷마스크가 사용되며 원하는 게이트웨이가 9.37.54.1인 경우 AIX는 라우트에 대해 루프백을 사용합니다. 이로 인해 서버 응답이 상자를 떠날 수 없으며 클라이언트 대기 시간이 종료됩니다. 일반적으로 클라이언트는 클러스터가 보내는 하나의 응답만을 받으므로 라우트가 작성되고 클라이언트는 더이상 수신하지 않습니다.

이 문제를 해결할 수 있는 두 가지 해결책이 있습니다.

1. AIX가 동적으로 라우트를 추가하지 않도록 path MTU discovery를 사용 불가능으로 설정하십시오. 다음 명령을 사용하십시오.

no -a AIX 네트워크 설정을 나열합니다.

no -o option=value

AIX에서 TCP 매개변수를 설정합니다.

2. 255.255.255.255 넷마스크를 사용하여 루프백의 클러스터 IP의 별명을 지정하십시오. 이는 별명이 지정된 서브넷이 유일한 클러스터 IP임을 의미합니다. AIX

가 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 목적지가 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: 권고자는 모든 서버가 중단되었음을 표시함

Windows 2000에는 태스크 오프로드라는 새 기능이 있어서 운영 체제가 아닌 어댑터 카드에서 TCP 체크섬을 계산합니다. 이 경우 시스템의 성능이 향상됩니다. 태스크 오프로드가 사용 가능한 경우, Network Dispatcher 권고자는 서버가 중단되지 않은 경우에도 중단되었다고 보고합니다.

클러스터 주소에서 전송되는 패킷에 대해 TCP 체크섬이 정확하게 계산되지 않는 문제가 발생하는데, 권고자 통신에 의해 이런 문제가 발생합니다.

이러한 문제점을 피하려면 어댑터 카드 설정에서 태스크 오프로드를 사용 불가능으로 설정하십시오.

이 문제점은 Adaptec의 ANA62044 QuadPort Adapter에서 처음 발견되었습니다. 이 어댑터 카드는 체크섬 오프로드 전송이라는 기능을 참조합니다. 이러한 문제점이 발생하지 않도록 체크섬 오프로드 전송을 사용 불가능하게 하십시오.

문제점: Network Dispatcher의 광역 모드에서 고가용성이 작동되지 않음

Wide Area Network Dispatcher를 설치한 경우, 원격 Dispatcher를 로컬 Dispatcher의 클러스터에서 서버로 정의해야 합니다. 일반적으로 원격 Dispatcher

의 전달되지 않는 주소(NFA)를 원격 서버의 목적지 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.

큰 구성 파일(대략 200개 또는 그 이상의 추가 명령)을 로드하려고 하면, GUI는 정지하거나 화면에서 매우 느리게 반응하는 예기치 못한 작동을 나타냅니다.

이러한 현상은 Java가 많은 GUI 변경사항을 처리할 수 있는 데 충분한 메모리에 액세스하지 않기 때문에 발생합니다.

Java에 사용 가능한 메모리 할당 풀을 증가시키기 위해 지정할 수 있는 런타임 환경 옵션이 있습니다.

이 옵션은 `-Xmxn`로 표시되며 여기서 `n`은 메모리 할당 풀의 최대 크기(단위: 바이트)를 나타냅니다. `n`은 1024의 배수이어야 하며 2MB 보다 커야 합니다. 값 `n` 다음에 KB를 표시하는 `k` 또는 K 및 MB를 표시하는 `m` 또는 M이 올 수 있습니다. 예를 들어, `-Xmx128M` 및 `-Xmx81920k`는 모두 유효합니다. 기본값은 64MB입니다. Solaris 7 및 Solaris 8 SPARC 플랫폼의 최대값은 4000m이며, Solaris 2.6 및 x86 플랫폼의 최대값은 2000m입니다.

이 옵션을 추가하려면 다음과 같이 `ndadmin` 스크립트 파일을 수정하십시오.

- **Windows NT 또는 2000**

```
START jrew -mx64m %END_ACCESS% %CONFIG_DIR%  
-DEND_INSTALL_PATH=%IBMNDPATH% -cp %NDCLASSPATH%  
com.ibm.internet.nd.framework.FWK_Framework  
com.ibm.internet.nd.gui.GUI_eNDRootNode1
```

- **Solaris**

```
$JREDIR/$JRE -mx64m $END_ACCESS $CONFIG_DIR  
-DEND_INSTALL_PATH=/opt/&BASEDIR -cp $NDCLASSPATH  
com.ibm.internet.nd.framework.FWK_Framework  
com.ibm.internet.nd.gui.GUI_eNDRootNode &1
```

- **Linux**

```
re -mx64m $END_ACCESS $CONFIG_DIR $NDLOCALE  
-DEND_INSTALL_PATH=/opt/nd -classpath $NDCLASSPATH  
com.ibm.internet.nd.framework.FWK_Framework  
com.ibm.internet.nd.gui.GUI_eNDRootNode 1>/dev/null 2>&1 &1
```

- **AIX**

```
ava -mx64m $END_ACCESS $CONFIG_DIR $NDLOCALE  
-DEND_INSTALL_PATH=/usr/lpp/&BASEDIR -classpath $NDCLASSPATH  
com.ibm.internet.nd.framework.FWK_Framework  
com.ibm.internet.nd.gui.GUI_eNDRootNode 1>/dev/null 2>&1 &
```

n에 대한 권장값은 없지만 기본 옵션보다 커야 합니다. 기본값의 두 배로 시작하는 것이 좋습니다.

공통 문제점 해결--CBR

문제점: CBR이 실행되지 않음

이 문제점은 다른 응용프로그램에서 CBR이 사용하는 포트 중 하나를 사용하는 경우에 발생할 수 있습니다. 자세한 내용은 253 페이지의 『CBR 포트 번호 확인』을 참조하십시오.

문제점: cbrcontrol 또는 ndadmin 명령 실패

cbrcontrol 명령이 “오류: 서버가 응답하지 않습니다.”를 리턴합니다. 또는 ndadmin 명령이 “오류: RMI 서버를 액세스할 수 없습니다.”를 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 jre
EXCLUDE-MODULE jrew
EXCLUDE-MODULE javaw
```

이러한 오류는 **cbrserver**가 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 프로그램에 접속할 수 없습니다."라는 오류 메시지가 포함됩니다. 이러한 문제점이 발생하지 않도록 Caching Proxy를 시작하기 전에 실행 프로그램을 시작하십시오.

문제점: Solaris에서 cbrcontrol executor start 명령 실패

Solaris에서 **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 101을 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. 347 페이지의 『부록C. 콘텐츠 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.

공통 문제점 해결--Mailbox Locator

문제점: Mailbox Locator가 실행되지 않음

이 문제점은 다른 응용프로그램이 Mailbox Locator에서 사용하는 포트 중 하나를 사용하고 있을 때 발생할 수 있습니다. 자세한 내용은 254 페이지의 『Mailbox Locator 포트 번호 확인』을 참조하십시오.

문제점: mlserver 명령이 정지됨

UNIX 플랫폼에서, 이 문제는 **mlserver**가 많은 IMAP/POP3 클라이언트 요청을 로드 밸런싱하는 데 사용되고, 파일 설명자에 대한 시스템의 한계가 mlserver에서 처리하는 요청 수에 비해 너무 작을 때 발생합니다. mlserver가 다음 예외를 작성한 다음, 정지합니다.

```
java.rmi.RMIException: security.fd.read
```

프로토콜 고유의 프록시 로그 파일은 다음을 보고합니다.

```
SocketException=java.net.SocketException: Socket closed
```

해결책은 mlserver가 시작된 셸에서 **nofiles**(AIX, Linux) 또는 **파일 열기**(Solaris) 한계를 수정하는 것입니다. nofiles 한계를 현재 nofiles 한계보다 더 큰 적절한 값으로 늘리십시오. ulimit -a를 사용하여 현재 nofiles 한계를 표시하고, ulimit -n *value*를 사용하여 값을 늘리십시오.

문제점: mlcontrol 또는 ndadmin 명령 실패

mlcontrol 명령이 “오류: 서버가 응답하지 않습니다.”를 리턴합니다. 또는 ndadmin 명령이 “오류: RMI 서버를 액세스할 수 없습니다.”를 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 jre
EXCLUDE-MODULE jrew
EXCLUDE-MODULE javaw
```

이러한 오류는 **mlserver**가 시작되지 않은 경우에도 발생할 수 있습니다.

문제점: 포트를 추가할 수 없음

구성에 포트를 추가하려고 할 때 오류: 포트를 추가할 수 없습니다라는 오류 메시지가 나타날 수 있습니다. 다른 응용프로그램이 이미 해당 포트를 인식했을 수 있습니다. Mailbox Locator가 명령에서 지정한 포트의 클러스터 IP에 바인드하는 프록시를 시작하려고 합니다. 다른 응용프로그램이 해당 IP에 바인드되어 있거나 해당 포트의 모든 IP를 인식하는 경우 프록시 시동이 실패합니다. 해당 포트에서 Mailbox Locator를 사용하려면 충돌하는 응용프로그램을 정지시켜야 합니다.

Linux 플랫폼에서 xinetd 디먼은 POP3 프로그램을 실행하지 않고 리스너를 시작할 수 있습니다. 따라서 **netstat -a**를 확인하여 의도된 포트에서 인식 중인 응용 프로그램이 있는지 판별하는 것이 중요합니다.

문제점: 포트 추가를 시도할 때 프록시 오류 발생

Mailbox Locator의 경우, **mlcontrol port add** 명령이 “클러스터<cluster> , 포트 <port>의 프록시가 시작되지 않았습니다.”라는 오류 메시지를 생성합니다. 해결책은 프록시가 시작되기 전에 먼저 클러스터 주소를 구성하는 것입니다. 또한 클러스터 주소를 인식하는 포트에서 실행되는 응용프로그램이 없는지도 확인하십시오.

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 255 페이지의 『Site Selector 포트 번호 확인』을 참조하십시오.

문제점: Site Selector가 Solaris 클라이언트로부터 통신을 라운드 로빈 (round-robin)하지 않음

증상: Site Selector 구성요소가 Solaris 클라이언트로부터 수신하는 요청을 라운드 로빈하지 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 ndadmin 명령 실패

sscontrol 명령이 “오류: 서버가 응답하지 않습니다.”를 리턴합니다. 또는 ndadmin 명령이 “오류: RMI 서버를 액세스할 수 없습니다.”를 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 jre
EXCLUDE-MODULE jrew
EXCLUDE-MODULE javaw
```

이러한 오류는 ssserver가 시작되지 않은 경우에도 발생할 수 있습니다.

문제점: ssserver가 Windows 2000에서 시작에 실패함

Site Selector는 DNS에 참여할 수 있어야 합니다. 구성에 포함된 모든 시스템도 이 시스템에 참여해야 합니다. Windows에서는 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 오류가 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.

공통 문제점 해결--Consultant for Cisco CSS Switches

문제점: Ibcserver가 시작되지 않음

이 문제는 다른 응용프로그램이 컨설턴트의 Ibcserver에서 사용하는 포트 중 하나를 사용하고 있을 때 발생할 수 있습니다. 자세한 내용은 255 페이지의 『Cisco Consultant 포트 번호 확인』을 참조하십시오.

문제점: Ibccontrol 또는 ndadmin 명령 실패

Ibccontrol 명령이 “오류: 서버가 응답하지 않습니다.”를 리턴합니다. 또는 ndadmin 명령이 “오류: RMI 서버를 액세스할 수 없습니다.”를 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 jre
EXCLUDE-MODULE jrew
EXCLUDE-MODULE javaw
```

이러한 오류는 Ibcserver가 시작되지 않은 경우에도 발생할 수 있습니다.

문제점: 포트 14099에서 레지스트리를 작성할 수 없음

이 문제는 유효한 제품 사용권이 없을 때 발생할 수 있습니다. Ibcserver를 시작하려고 할 때, 다음 메시지가 나타납니다.

사용권이 만기되었습니다. 현지 IBM
영업대표 또는 허가된 IBM 대리점에 문의하십시오.

이 문제를 해결하려면 다음을 수행하십시오.

1. lbcserver를 이미 시작한 경우 **lbcserver stop**을 입력하십시오.
2. 유효한 사용권을 **...nd/servers/conf** 디렉토리로 복사하십시오.
3. **lbcserver**를 입력하여 서버를 시작하십시오.

공통 문제점 해결--Metric Server

문제점: .bat 또는 .cmd 사용자 메트릭 파일을 실행하는 Windows 2000의 Metric Server IOException

Windows 2000 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 Network Dispatcher 시스템에 로드를 보고 하지 않음

Metric Server가 로드 정보를 Network Dispatcher에 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.
- 더 높은 로그 레벨로 재시작 한 후 오류를 찾으십시오.
- Network Dispatcher 시스템에서 관리자 로그 레벨을 늘리십시오. Metric Monitor 로그에서 오류를 찾으십시오.

문제점: Metric Server 로그가 "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.

부록A. 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.

```
▶—required_parameter_1—┐
    └—required_parameter_2—┘
```

선택 값: 선택 키워드와 변수는 기본 경로 선 아래에 나타납니다.

```
▶—┐
   └—keyword—┘
```

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.

```
▶—┐
   └—parameter_1—┘
     └—parameter_2—┘
```

변수: 모두 이탤릭체로 되어 있는 단어는 변수입니다. 구문에 변수가 있으면, 이 변수를 텍스트에 정의된 대로 허용되는 이름이나 값 중 하나로 바꾸어야 합니다.

▶▶—*variable*—▶▶

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.

▶▶—*클러스터: 포트*—▶▶

부록B. Dispatcher, CBR 및 Mailbox Locator 명령어 참조 서

이 부록에서는 Dispatcher **ndcontrol** 명령 사용 방법을 설명합니다. 이 부록은 CBR 및 Mailbox Locator에 대한 명령어 참조서이기도 합니다. CBR 및 Mailbox Locator는 Dispatcher 명령 서브세트를 사용합니다. 278 페이지의 『CBR, Mailbox Locator 및 Dispatcher 간의 구성 차이점』에서 자세한 정보를 참조하십시오.

주: 이 구문 다이어그램 사용 시 -

- CBR의 경우, ndcontrol을 **cbrcontrol**로 대체하십시오.
- Mailbox Locator의 경우, ndcontrol을 **mlcontrol**로 대체하십시오.

아래는 이 부록에 있는 명령 목록입니다.

- 280 페이지의 『ndcontrol advisor -- 권고자 제어』
- 287 페이지의 『ndcontrol cluster -- 클러스터 구성』
- 293 페이지의 『ndcontrol executor -- 실행 프로그램 제어』
- 298 페이지의 『ndcontrol file -- 구성 파일 관리』
- 300 페이지의 『ndcontrol help -- 이 명령의 도움말 표시 또는 인쇄』
- 302 페이지의 『ndcontrol highavailability -- 고가용성 제어』
- 307 페이지의 『ndcontrol host -- 원격 시스템 구성』
- 308 페이지의 『ndcontrol log -- 2진 로그 파일 제어』
- 309 페이지의 『ndcontrol manager -- 관리 프로그램 제어』
- 317 페이지의 『ndcontrol metric -- 시스템 메트릭 구성』
- 319 페이지의 『ndcontrol 포트 -- 포트 구성』
- 327 페이지의 『ndcontrol rule -- 규칙 구성』
- 335 페이지의 『ndcontrol server -- 서버 구성』
- 342 페이지의 『ndcontrol set -- 서버 로그 구성』

- 343 페이지의 『ndcontrol status -- 관리 프로그램 및 권고자가 실행 중인지 여부 표시』
- 344 페이지의 『ndcontrol subagent -- SNMP 서브에이전트 구성』

ndcontrol 명령 매개변수의 최소 버전을 입력할 수 있습니다. 매개변수의 고유한 문자만 입력해야 합니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면, **ndcontrol help file** 대신에 **ndcontrol he f**를 입력할 수 있습니다.

명령행 인터페이스를 시작하려면 **ndcontrol** 명령을 실행하여 ndcontrol 명령 프롬프트를 받으십시오.

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

주: 명령 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름(클러스터, 서버, 고가용성 명령에 사용)과 파일 이름(파일 명령에 사용)만 예외입니다.

CBR, Mailbox Locator 및 Dispatcher 간의 구성 차이점

CBR 및 Mailbox Locator 명령행 인터페이스는 대부분 Dispatcher 명령행 인터페이스의 서브세트입니다. ndcontrol 대신 **cbrcontrol** 명령(CBR 구성요소의 경우)을 사용하거나 **mlcontrol** 명령(Mailbox Locator 구성요소의 경우)을 사용하여 구성요소를 구성하십시오.

아래는 CBR에 생략된 일부 명령 목록입니다.

1. highavailability
2. subagent
3. 실행 프로그램
 - report
 - set nfa <value>
 - set fincount <value>
 - set fintimeout <value>
 - set porttype <value>
4. 클러스터

- report {c}
 - set {c} porttype
5. port add {c:p} porttype
 6. port set {c:p} porttype
 7. rule add {c:p:r} type port
 8. server add {c:p:s} router
 9. server set {c:p:s} router

아래는 Mailbox Locator에 생략된 일부 명령 목록입니다.

1. highavailability
2. rule
3. subagent
4. 실행 프로그램
 - start
 - stop
 - report
 - set nfa <value>
 - set fincount <value>
 - set fintimeout <value>
 - set porttype <value>
5. 클러스터
 - report {c}
 - set {c} porttype
6. port [add|set] {c:p} porttype
7. server [add|set] {c:p:s} router

스터 값을 지정하지 않으면 권고자가 모든 클러스터에 대해 포트에서 실행을 시작합니다. 클러스터를 지정하면 권고자가 포트에서 실행을 시작하지만 지정된 클러스터에 대해서만 실행됩니다. 158 페이지의 『권고자 시작 및 정지』에서 자세한 정보를 참조하십시오.

클러스터는 점분리 10진수 주소 또는 기호 이름입니다. 포트는 권고자가 모니터링하는 포트 번호입니다.

timeoutseconds

서버와의 연결 실패를 보고하기 전에 권고자가 대기하는 제한 시간을 초 단위로 표시하는 양의 정수. 기본값은 권고자 간격에 지정된 값의 3배입니다.

interval

권고자가 정보에 대해 서버를 조회하는 간격을 설정합니다.

seconds

서버의 현재 상태를 서버에 요청하는 간격(초 단위)을 나타내는 양의 정수. 기본값은 7입니다.

list

현재 관리 프로그램에 정보를 제공하는 권고자 목록을 보여줍니다.

loglevel

권고자 로그에 대한 로그 레벨을 설정합니다.

level

레벨 번호(0-5). 기본값은 1입니다. 번호가 높아질수록 더 자세한 정보가 권고자 로그에 기록됩니다. 가능한 값은 다음과 같습니다. 0은 없음, 1은 최소, 2는 기본, 3은 중간, 4는 고급, 5는 자세합니다.

logsize

권고자 로그의 최대 크기를 설정합니다. 로그 파일의 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면, 후속 항목은 파일의 맨 위에서부터, 이전 로그 항목에 겹쳐 기록됩니다. 로그 크기는 로그의 현재 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.

number of records

권고자 로그 파일의 최대 크기(바이트 단위). 0보다 큰 양의 정수이거나 **unlimited** 단어를 지정할 수 있습니다. 로그 파일은 로그 항목 그 자체의 크기가 변하므로, 겹쳐쓰기 전에는 그 정확한 최대 크기에 도달할 수 없습니다. 기본값은 1MB입니다.

receivetimeout

서버에서 수신 실패를 보고하기 전에 권고자가 대기하는 시간을 설정하십시오. 자세한 내용은 160 페이지의 『권고자 연결 시간 종료 및 서버의 수신 시간 종료』를 참조하십시오.

timeoutseconds

서버에서 수신 실패를 보고하기 전에 권고자가 대기하는 종료 시간을 초 단위로 표시하는 양의 정수. 기본값은 권고자 간격에 지정된 값의 3배입니다.

report

권고자 상태에 대한 보고서를 표시합니다.

start

권고자를 시작합니다. 각 프로토콜에 대해 권고자가 있습니다. 기본 포트는 다음과 같습니다.

권고자 이름	프로토콜	포트
connect	ICMP	12345
db2	개인용	50000
dns	DNS	53
ftp	FTP	21
http	HTTP	80
ibmproxy	HTTP(Caching Proxy를 통한)	80
imap	IMAP	143
nntp	NNTP	119
ping	PING	0
pop3	POP3	110
self	개인용	12345
smtp	SMTP	25
ssl	HTTP	443

권고자 이름	프로토콜	포트
ssl2http	SSL	443
telnet	Telnet	23
WLM	개인용	10,007

주: FTP 권고자는 FTP 제어 포트(21)에만 권고합니다. FTP 데이터 포트(20)에서는 FTP 권고자를 시작하지 마십시오.

log file

포트 80에서 관리 데이터가 기록되는 파일 이름. 로그의 각 레코드에는 시간 소인이 표시됩니다.

기본 파일은 *advisorname_port.log*이며, 예로는 **http_80.log**가 있습니다. 로그 파일이 보존되는 디렉토리를 변경하려면, 232 페이지의 『로그 파일 경로 변경』을 참조하십시오. 클러스터(또는 사이트)에 대한 고유 권고자의 기본 로그 파일은 클러스터 주소(예: **http_127.40.50.1_80.log**)를 사용하여 작성됩니다.

status

전역 및 해당 기본값으로 설정될 수 있는 권고자의 모든 값에 대한 현재 상태를 표시합니다.

stop

권고자를 정지합니다.

시간 종료(timeout)

관리 프로그램에서 권고자의 정보를 유효한 것으로 간주하는 시간(초)을 설정합니다. 권고자 정보가 이 시간 종료보다 이전 정보라는 것을 관리 프로그램에서 발견하면, 관리 프로그램은 권고자가 모니터링하는 포트에서 서버에 대한 가중치를 판별할 때 이 정보를 사용하지 않습니다. 이 시간 종료에 대한 예외는, 권고자가 관리 프로그램에 특정 서버가 단절되었다는 것을 알린 경우입니다. 관리 프로그램에서는 권고자 정보가 시간 종료된 후에도 서버에 대한 해당 정보를 사용합니다.

seconds

시간(초)을 나타내는 양수 또는 단어 **unlimited**. 기본값은 unlimited입니다.

version

권고자의 현재 버전을 표시합니다.

예제

- 클러스터 127.40.50.1에 대해 포트 80에서 http 권고자를 시작하려면 다음 명령을 실행하십시오.

```
ndcontrol advisor start http 127.40.50.1:80
```

- 모든 클러스터에 대해 포트 80에서 http 권고자를 시작하려면 다음 명령을 실행하십시오.

```
ndcontrol advisor start http 88
```

- 클러스터 127.40.50.1에 대해 포트 80에서 http 권고자를 정지하려면 다음 명령을 실행하십시오.

```
ndcontrol advisor stop http 127.40.50.1:80
```

- 서버와의 연결 실패를 보고하기 전에 포트 80에서 HTTP 권고자가 대기하는 시간(30초)을 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol advisor connecttimeout http 80 30
```

- 서버와의 연결 실패를 보고하기 전에 클러스터 127.40.50.1의 포트 80에서 HTTP 권고자가 대기하는 시간(20초)을 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol advisor connecttimeout http 127.40.50.1:80 20
```

- FTP 권고자 간격(포트 21의 경우)을 6초로 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol advisor interval ftp 21 6
```

- 현재 관리 프로그램에 정보를 제공하는 권고자 목록을 표시하려면 다음 명령을 실행하십시오.

```
ndcontrol 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 권고자 로그의 로그 레벨을 0으로 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol advisor loglevel http 80 0
```

- 포트 21의 ftp 권고자 로그 크기를 5000 바이트로 변경하려면 다음 명령을 실행하십시오.

```
ndcontrol advisor logsize ftp 21 5000
```

- 서버에서 수신 실패를 보고하기 전에 HTTP 권고자가(포트 80의 경우) 대기하는 시간(60초)을 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol advisor receivetimeout http 80 60
```

- FTP 권고자 상태에 대한 보고서를 표시하려면(포트 21의 경우) 다음 명령을 실행하십시오.

```
ndcontrol 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 권고자와 연관된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

```
ndcontrol 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
```

- 포트 21의 ftp 권고자 정보에 대한 시간 종료 값을 5초로 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol advisor timeout ftp 21 5
```

- 포트 443의 ssl 권고자의 현재 버전 번호를 표시하려면 다음 명령을 실행하십시오.

```
ndcontrol advisor version ssl 443
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
Version: 04.00.00.00 - 07/12/2001-10:09:56-EDT
```


아래에 설명되어 있는 이러한 값 각각은 총계의 백분율로 표시되므로, 합은 항상 100입니다. 자세한 정보는 152 페이지의 『상태 정보에 제공되는 중요성 비율』을 참조하십시오.

active

활성 연결에 부여될 가중치를 표시하는 0에서 100 사이의 숫자. 기본값은 50입니다.

new

새 연결에 부여될 가중치를 표시하는 0에서 100 사이의 숫자. 기본값은 50입니다.

port

권고자의 정보에 부여될 가중치를 표시하는 0에서 100 사이의 숫자. 기본값은 0입니다.

주: 권고자가 시작되고 포트 비율이 0이면, Network Dispatcher는 자동으로 이 값을 1로 설정하여 관리 프로그램에서 서버 가중치를 계산하기 위한 입력으로 권고자 정보를 사용할 수 있도록 합니다.

system

Metric Server와 같은 시스템 메트릭 정보에 부여될 가중치를 표시하는 0에서 100 사이의 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: Dispatcher 전달 방법에서, 포트가 작성될 stickytime이 0이 아니고 새 포트가 추가되면 SSL ID 연관 관계를 포트에 사용할 수 있습니다. 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 값입니다.

porttype

기본 포트 유형. 이것은 **port porttype**을 사용하여 각각의 포트마다 대체할 수 있습니다.

주: porttype은 Dispatcher에 적용됩니다.

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 시작되어 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. 59 페이지의 『상호 고가용성』에서 자세한 정보를 참조하십시오.

address

기본 호스트의 주소 값. 기본값은 이 시스템의 NFA 주소입니다.

staletimeout

연결이 제거되기 전에 연결에서 비활동 상태로 있을 수 있는 시간(초 단위). FTP의 기본값은 900이고, 텔넷의 기본값은 32,000,000입니다. 모든 프로토콜에 대해 기본값은 300입니다. 이것은 **port staletimeout**을 사용하여 각각의 포트마다 대체할 수 있습니다. 자세한 정보는 233 페이지의 『활동해제 제한 시간 값 사용』을 참조하십시오.

주: Mailbox Locator의 경우, staletimeout은 이러한 프로토콜에 대한 비활동 자동 로그아웃 타이머에 해당합니다. Mailbox Locator의 경우 staletimeout 기본값은 60초이며, 이 값은 POP3 및 IMAP에 대한 비활동 제한 시간을 덮어씁니다. Mailbox Locator staletimeout에 대한 자세한 정보는 112 페이지의 『POP3/IMAP 활동 해제 타이머 덮어쓰기』를 참조하십시오.

staletimout

활동해제 제한 시간 값.

sharedbandwidth

클러스터 레벨에서 공유할 수 있는 최대 대역폭(초당 KB 단위). 공유 대역폭에 대한 자세한 정보는 201 페이지의 『예약된 대역폭 및 공유 대역폭에 따라 규칙 사용』 및 202 페이지의 『공유 대역폭 규칙』을 참조하십시오.

주: 공유 대역폭은 CBR 또는 Mailbox Locator에는 적용되지 않습니다.

size

sharedbandwidth 크기는 정수 값입니다. 기본값은 0입니다. 값이 0이면, 클러스터 레벨에서 대역폭을 공유할 수 없습니다.

set

클러스터의 특성을 설정합니다.

remove

클러스터를 제거합니다.

report

클러스터의 내부 필드를 표시합니다.

주: report는 CBR 또는 Mailbox Locator에는 적용되지 않습니다.

status

특정 클러스터의 현재 상태를 보여줍니다.

configure

클러스터 별명을 네트워크 인터페이스 카드에 구성합니다.

주: configure는 CBR 또는 Mailbox Locator에는 적용되지 않습니다.

interfacename netmask

어떤 Dispatcher로부터 처음 찾은 다른 별명이 있는 경우 필요합니다.

unconfigure

네트워크 인터페이스 카드로부터 클러스터 별명을 삭제합니다.

주: unconfigure는 CBR 또는 Mailbox Locator에는 적용되지 않습니다.

예제

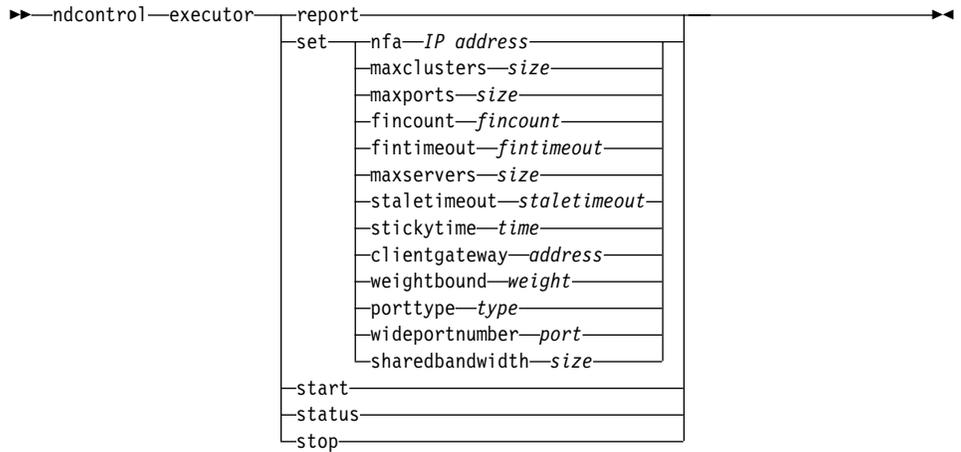
- 클러스터 주소 130.40.52.153을 추가하려면 다음 명령을 실행하십시오.
`ndcontrol cluster add 130.40.52.153`
- 클러스터 주소 130.40.52.153을 제거하려면 다음 명령을 실행하십시오.
`ndcontrol cluster remove 130.40.52.153`
- 클러스터 9.6.54.12에 있는 서버에 대해 관리 프로그램이 수신한 입력(active, new, port, system)에 상대적 중요도를 설정하려면 다음 명령을 실행하십시오.
`ndcontrol cluster set 9.6.54.12 proportions 60 35 5 0`
- 와일드 카드 클러스터를 추가하려면 다음 명령을 실행하십시오.
`ndcontrol 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.
`ndcontrol cluster set 9.6.54.12 primaryhost 9.65.70.19`
- 클러스터 주소 9.67.131.167의 상태를 표시하려면 다음 명령을 실행하십시오.
`ndcontrol cluster status 9.67.131.167`

이 명령으로 다음과 같은 출력이 작성됩니다.

Cluster Status:

```
-----  
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
```

ndcontrol executor -- 실행 프로그램 제어



report

통계 스냅샷 보고서를 표시합니다. 예를 들어, 수신된 패킷 총계, 버린 패킷 총계, 오류로 전송된 패킷 총계 등을 표시합니다.

주: report는 CBR 또는 Mailbox Locator에는 적용되지 않습니다.

set

실행 프로그램의 필드를 설정합니다.

nfa

비전달 주소를 설정합니다. 이 주소로 전송된 모든 패킷이 Dispatcher 시스템에서 전달되지 않습니다.

주: NFA는 CBR 또는 Mailbox Locator에는 적용되지 않습니다.

IP 주소

기호 이름이나 점분리 10진수 형식의 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.

fincount

연결의 가비지 컬렉션이 시작되기 전에 완료 상태에 있어야 하는 연결 수. fincount의 기본값은 4000입니다.

fincount

fincount 값.

주: fincount는 CBR 또는 Mailbox Locator에는 적용되지 않습니다.

fintimeout

연결이 완료 상태로 된 후 메모리에 연결을 유지하는 초 시간. 기본 fintimeout 값은 60입니다.

fintimeout

fintimeout 값.

주: fintimeout은 CBR 또는 Mailbox Locator에는 적용되지 않습니다.

maxservers

포트당 기본 최대 서버 수. 이것은 **cluster** 또는 **port** 명령으로 대체할 수 있습니다. maxservers의 기본값은 32입니다.

size

서버의 수.

staletimeout

연결이 제거되기 전에 연결에서 비활동 상태로 있을 수 있는 시간(초 단위). FTP의 기본값은 900이고, 텔넷의 기본값은 32,000,000입니다. 기타 모든 포트에 대해 기본값은 300입니다. 이것은 **cluster** 또는 **port** 명령으로 대체할 수 있습니다. 자세한 정보는 233 페이지의 『활동해제 제한 시간 값 사용』을 참조하십시오.

주: Mailbox Locator의 경우, `staletimeout`은 이들 프로토콜에 대한 비활동 자동 로그아웃 타이머에 해당합니다. Mailbox Locator의 경우 `staletimeout` 기본값은 60초이며, 이 값은 POP3 및 IMAP에 대한 비활동 제한 시간을 덮어씁니다. Mailbox Locator `staletimeout`에 대한 자세한 정보는 112 페이지의 『POP3/IMAP 활동 해제 타이머 덮어쓰기』를 참조하십시오.

staletimeout

활동해제 제한 시간 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

`clientgateway`는 NAT/NAPT 또는 Dispatcher content-based routing에 사용되는 IP 주소입니다. 리턴 방향의 통신량이 Network Dispatcher에서 클라이언트로 전달되는 라우터 주소입니다. `Clientgateway`는 NAT/NAPT 또는 Dispatcher content-based routing 전달 방법을 사용하여 포트를 추가하기 전에 0이 아닌 값으로 설정해야 합니다. 자세한 정보는 61 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 방법)』 및 63 페이지의 『Dispatcher content-based routing(cbr 전달 방법)』을 참조하십시오.

주: `clientgateway`는 Dispatcher 구성요소에만 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: porttype은 CBR 또는 Mailbox Locator에는 적용되지 않습니다.

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 CBR 또는 Mailbox Locator에는 적용되지 않습니다.

port

wideportnumber의 값.

sharedbandwidth

실행 프로그램 레벨에서 공유할 수 있는 최대 대역폭(초당 KB 단위). 공유 대역폭에 대한 자세한 정보는 201 페이지의 『예약된 대역폭 및 공유 대역폭에 따라 규칙 사용』 및 202 페이지의 『공유 대역폭 규칙』을 참조하십시오.

주: 공유 대역폭은 CBR 또는 Mailbox Locator에는 적용되지 않습니다.

size

sharedbandwidth 크기는 정수 값입니다. 기본값은 0입니다. 값이 0이면, 실행 프로그램 레벨에서 대역폭을 공유할 수 없습니다.

start

실행 프로그램을 시작합니다.

주: start는 Mailbox Locator에는 적용되지 않습니다.

status

실행 프로그램에서 설정할 수 있는 값의 현재 상태와 기본값을 표시합니다.

stop

실행 프로그램을 정지합니다. Dispatcher의 경우, stop은 Windows 2000에서 는 유효한 매개변수가 아닙니다.

주: stop은 Dispatcher 및 CBR에 적용됩니다.

예제

- Dispatcher의 내부 카운터를 표시하려면 다음 명령을 실행하십시오.

```
ndcontrol executor status
```

```
Executor Status:
```

```
-----
```

```
Nonforwarding address ..... 9.67.131.151
Client gateway address ..... 0.0.0.0
Fin count ..... 4,000
Fin timeout ..... 60
Wide area network port number ..... 2,001
Shared bandwidth (Kbytes) ..... 0
Default maximum ports per cluster ... 8
Maximum number of clusters ..... 100
Default maximum servers per port .... 32
Port stale timeout ..... 300
Port sticky time ..... 0
Port weight bound ..... 20
Maximum number of clusters ..... 100
```

- 비전달 주소를 130.40.52.167로 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol executor set nfa 130.40.52.167
```

- 최대 클러스터 수를 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol executor set maxclusters 4096
```

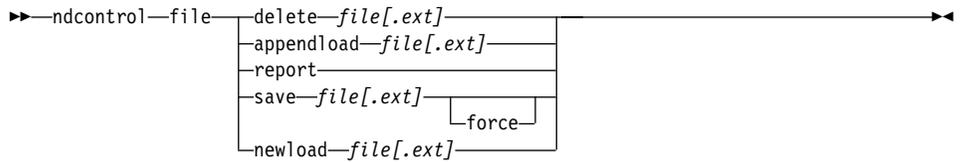
- 실행 프로그램을 시작하려면 다음 명령을 실행하십시오.

```
ndcontrol executor start
```

- 실행 프로그램을 정지하려면(AIX, Linux 및 Solaris만 해당) 다음 명령을 실행하십시오.

```
ndcontrol executor stop
```

ndcontrol file -- 구성 파일 관리



delete

파일을 삭제합니다.

file[.ext]

ndcontrol 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

네트워크 Dispatcher의 현재 구성을 파일에 저장합니다.

주: 파일은 다음 디렉토리에 저장되고 로드됩니다. 여기서, **component**는 Dispatcher, cbr 또는 ml(Mailbox Locator)입니다.

- AIX: **/usr/lpp/nd/servers/configurations/component**
- Linux: **/opt/nd/servers/configurations/component**
- Solaris: **/opt/nd/servers/configurations/component**
- Windows 2000:

일반 설치 디렉토리 경로 -- **c:\Program**

Files\ibm\edge\nd\servers\configurations\ 구성요소

고유(Native) 설치 디렉토리 경로 -- **c:\Program**

Files\ibm\nd\servers\configurations\component

force

파일을 기존 파일과 동일한 이름으로 저장하려면 **force**를 사용하여 새 파일을 저장하기 전에 기존 파일을 삭제하십시오. force 옵션을 사용하지 않으면 기존 파일을 겹쳐쓰지 않습니다.

newload

Network Dispatcher로 새 구성 파일을 로드하여 실행합니다. 새 구성 파일은 현재 구성을 대신합니다.

예제

- 파일을 삭제하려면 다음 명령을 실행하십시오.

```
ndcontrol file delete file3
```

```
File (file3) was deleted.
```

- 새 구성 파일을 로드하여 현재 구성을 대체하려면 다음 명령을 실행하십시오.

```
ndcontrol file newload file1.sv
```

```
File (file1.sv) was loaded into the Dispatcher.
```

- 현재 구성에 구성 파일을 추가하고 로드하려면 다음 명령을 실행하십시오.

```
ndcontrol 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.

```
ndcontrol file report
```

```
FILE REPORT:  
file1.save  
file2.sv  
file3
```

- 구성을 file3 파일에 저장하려면 다음 명령을 실행하십시오.

```
ndcontrol file save file3
```

```
The configuration was saved into file (file3)
```

ndcontrol help -- 이 명령의 도움말 표시 또는 인쇄

```
ndcontrol help help
ndcontrol help host
ndcontrol help executor
ndcontrol help manager
ndcontrol help advisor
ndcontrol help cluster
ndcontrol help port
ndcontrol help rule
ndcontrol help server
ndcontrol help subagent
ndcontrol help highavailability
ndcontrol help file
ndcontrol help set
ndcontrol help status
ndcontrol help log
```

예제

- ndcontrol 명령에 대한 도움말을 보려면 다음 명령을 실행하십시오.

```
ndcontrol 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
log           - help on 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
subagent      - help on subagent command
highavailability - help on high availability command
```

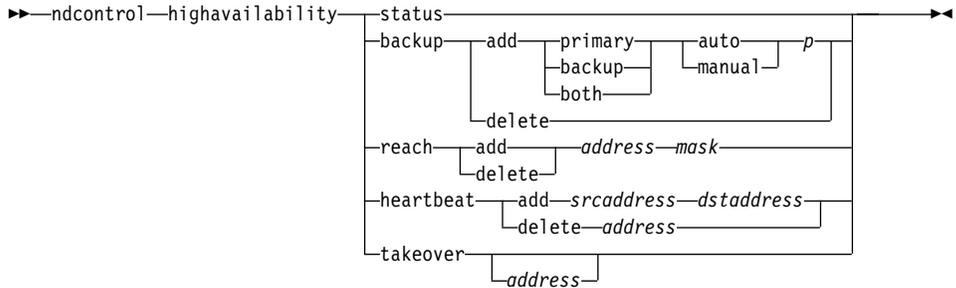
<> 내의 매개변수가 변수라는 점에 유의하십시오.

- 때로 help는 |로 옵션을 구분하여 변수에 대한 선택사항을 보여줍니다.

```
fintimeout <cluster address>|all <time>  
-Change FIN timeout  
(모든 클러스터를 변경할 경우 'all' 사용)
```

ndcontrol highavailability -- 고가용성 제어

주: ndcontrol 고가용성 구문 다이어그램은 CBR 또는 Mailbox Locator에는 적용되지 않습니다.



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)가 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 집합 기준으로 연관되어 있는 상호 고가용성 기능에 해당합니다. 자세한 내용은 59 페이지의 『상호 고가용성』을 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]

핵심 메시지에 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 목표 주소를 추가하거나 삭제합니다. reach 권고지는 ping을 백업 및 기본 Dispatcher 둘 모두에서 전송하여 목표에 어느 정도 도달 가능한지 판별합니다.

주: 도달 목표를 구성할 때 도달 권고자도 시작해야 합니다. reach 권고지는 관리 프로그램에서 자동으로 시작됩니다.

add

도달 권고자에 대한 대상 주소를 추가합니다.

delete

도달 권고자로부터 대상 주소를 제거합니다.

address

대상 노드의 IP 주소(점분리 10진수 또는 기호).

mask

서브넷 마스크.

heartbeat

기본 및 백업 Dispatcher 시스템 간의 통신 세션을 정의합니다.

add

소스 Dispatcher에 상대 주소(목적지 주소)를 알립니다.

srcaddress

소스 주소, 해당 Dispatcher 시스템의 주소(IP 또는 기호).

dstaddress

목적지 주소, 다른 Dispatcher 시스템의 주소(IP 또는 기호).

주: srcaddress 및 dstaddress는 적어도 하나 이상의 핵심 메시지 쌍에 대해 시스템의 NFA여야 합니다.

delete

핵심 메시지 정보에서 주소 쌍을 제거합니다. 핵심 메시지 쌍의 목적지 주소 또는 출발지 주소를 지정할 수 있습니다.

address

목적지 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. 자세한 내용은 59 페이지의 『상호 고가용성』을 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. 193 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.

```
ndcontrol highavailability status
```

출력:

```
High Availability Status:
```

```
-----  
Role .....primary  
Recovery Strategy ..... manual  
State ..... Active  
Sub-state..... Synchronized  
Primary host..... 9.67.131.151  
Port .....12,345
```

Preferred Target..... 9.67.134.223

Heartbeat Status:

Count 1

Reachability Status:

Count 1

- 자동 복구 전략어와 포트 80을 사용하여 기본 시스템에 백업 정보를 추가하려면 다음 명령을 실행하십시오.

```
ndcontrol highavailability backup add primary auto 80
```

- Dispatcher가 도달할 수 있어야 하는 주소를 추가하려면 다음 명령을 실행하십시오.

```
ndcontrol 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.

```
ndcontrol highavailability takeover
```

ndcontrol host -- 원격 시스템 구성

▶▶—ndcontrol—host:—remote_host—▶▶

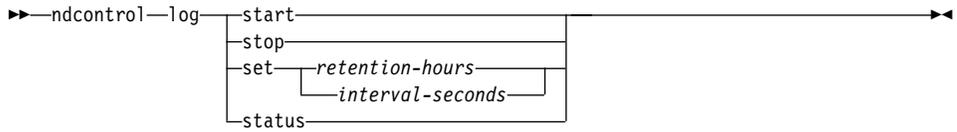
remote_host

구성 중인 원격 Network Dispatcher 시스템의 이름. 이 명령을 입력할 때는 다음과 같이 **host:**와 *remote_host* 사이에 공백이 없어야 합니다.

```
ndcontrol host:remote_host
```

명령 프롬프트에서 이 명령을 발생한 후에는 원격 Network Dispatcher 시스템에 대해 발행한 유효한 ndcontrol 명령을 입력하십시오.

ndcontrol log -- 2진 로그 파일 제어



start

2진 로그를 시작합니다.

stop

2진 로그를 정지합니다.

set

2진 로그 필드를 설정합니다. 2진 로그 필드 설정에 대한 자세한 정보는 223 페이지의 『서버 통제를 분석하기 위해 2진 로그 사용』을 참조하십시오.

retention

2진 로그 파일이 보존되는 시간. *retention*의 기본값은 24입니다.

hours

시간.

intervals

로그 입력 항목 간의 초 시간. 간격의 기본값은 60입니다.

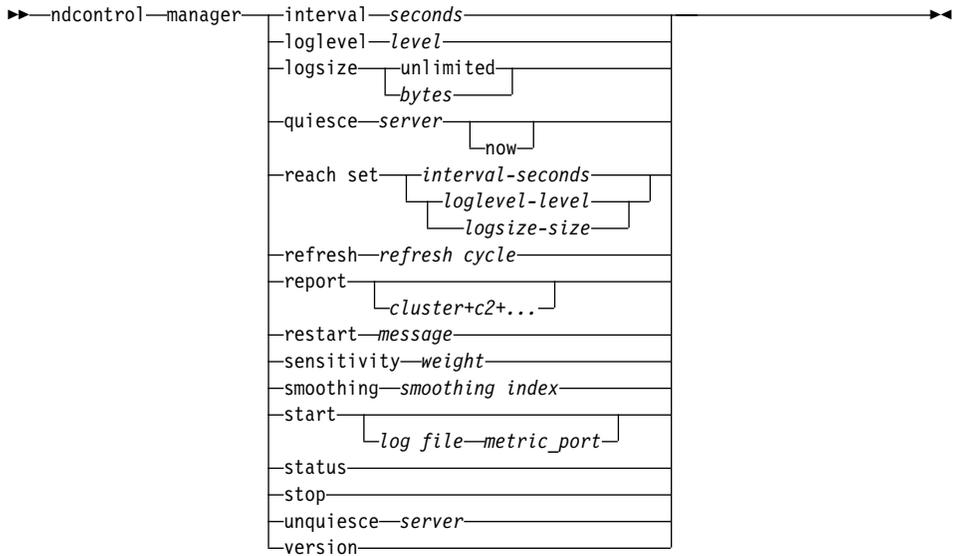
seconds

초 시간.

status

2진 로그의 보유 및 간격을 표시합니다.

ndcontrol manager -- 관리 프로그램 제어



interval

관리 프로그램이 실행 프로그램에 대한 서버의 가중치를 갱신하는 간격을 설정하여, 실행 프로그램에서 클라이언트 요청을 라우트하는 데 사용하는 기준을 갱신합니다.

seconds

관리 프로그램이 실행 프로그램에 대한 가중치를 갱신하는 간격(초)을 나타내는 양수. 기본값은 2입니다.

loglevel

메트릭 모니터 로그 및 관리 프로그램 로그에 대한 로그 레벨을 설정합니다.

level

레벨 번호(0-5). 번호가 커질수록 더 자세한 정보가 관리 프로그램 로그에 기록됩니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다. 0은 없음, 1은 최소, 2는 기본, 3은 중간, 4는 고급, 5는 자세합니다.

logsize

관리 프로그램 로그의 최대 크기를 설정합니다. 로그 파일의 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면, 후속 항목은 파일

의 맨 위에서부터, 이전 로그 항목에 겹쳐 기록됩니다. 로그 크기는 로그의 현재 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

관리 프로그램 로그 파일의 최대 크기(바이트 단위). 0보다 큰 양의 정수이거나 **unlimited** 단어를 지정할 수 있습니다. 로그 파일은 로그 항목 그 자체의 크기가 변하므로, 겹쳐쓰기 전에는 그 정확한 최대 크기에 도달할 수 없습니다. 기본값은 1MB입니다.

quiesce

연결이 결합 시간으로 지정되고 결합 시간이 만기되지 않은 경우, 클라이언트에서 작업중지된 서버로의 후속 연결을 제외하고 서버로 전송될 연결이 더 이상 없음을 지정합니다. 관리 프로그램은 서버가 정의되어 있는 모든 포트에 해당 서버의 가중치를 0으로 설정합니다. 서버에서 얼마간의 신속한 유지보수를 수행하기 위해 작업중지를 해제한 경우, 이 명령을 사용하십시오. 구성에서 작업중지된 서버를 삭제한 후 다시 추가한 경우, 이전에 작업중지된 상태가 유지되지 않습니다. 자세한 정보는 216 페이지의 『결합 연결 처리 작업중지』를 참조하십시오

server

기호 이름이나 점분리 10진수 형식으로 된 서버의 IP 주소.

또는 서버 파티션을 사용할 경우, 논리 서버의 고유 이름. 172 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 216 페이지의 『결합 연결 처리 작업중지』를 참조하십시오

reach set

reach 권고자에 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. 이 주소는 기호 이름이나 점분리 10진수 형식일 수 있습니다. 기본값은 모든 클러스터에 대한 관리 프로그램 보고서 화면입니다.

주: 추가 클러스터는 더하기 부호(+)로 구분됩니다.

restart

모든 서버(단절되지 않은)를 재시작하여 가중치를 표준화합니다(최대 가중치의 1/2).

message

관리 프로그램 로그 파일에 기록할 메시지.

sensitivity

가중치가 갱신되는 최소 감도를 설정합니다. 이 설정은 외부 정보에 따라 서버에 대한 가중치를 관리 프로그램에서 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

관리 프로그램을 시작합니다.

log file

관리 프로그램 데이터가 기록되는 파일 이름. 로그의 각 레코드에는 시간 소인이 표시됩니다.

기본 파일은 **logs** 디렉토리에 설치됩니다. 413 페이지의 『부록F. 예제 구성 파일』을 참조하십시오. 로그 파일이 보존되는 디렉토리를 변경하려면, 232 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

전역 및 해당 기본값으로 설정될 수 있는 관리 프로그램의 모든 값에 대한 현재 상태를 표시합니다.

stop

관리 프로그램을 정지합니다.

unquiesce

관리 프로그램이 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

관리 프로그램 현재 버전을 표시합니다.

예제

- 관리 프로그램의 갱신 간격을 매 5초로 설정하려면 다음 명령을 실행하십시오.
ndcontrol manager interval 5
- 더 나은 성능을 위해 로그 레벨을 0으로 설정하려면 다음 명령을 실행하십시오.
ndcontrol manager loglevel 0

- 관리 프로그램 로그 크기를 1,000,000 바이트를 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol manager logsize 1000000
```

- 130.40.52.153에서 서버에 더 이상 연결을 전송하지 않도록 지정하려면 다음 명령을 실행하십시오.

```
ndcontrol manager quiesce 130.40.52.153
```

- 가중치가 갱신되기 전의 갱신 간격 수를 3으로 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol manager refresh 3
```

- 관리 프로그램의 통계 스냅샷을 확보하려면 다음 명령을 실행하십시오.

```
ndcontrol manager report
```

이 명령으로 다음과 같은 출력이 작성됩니다.

HOST TABLE LIST	STATUS
9.67.129.221	ACTIVE
9.67.129.213	ACTIVE
9.67.134.223	ACTIVE

9.67.131.18	WEIGHT	ACTIVE % 48	NEW % 48	PORT % 4	SYSTEM %					
PORT: 80	NOW	NEW	WT	CONN	WT	CONN	WT	LOAD	WT	LOAD
9.67.129.221	8	8	10	0	10	0	7	29	0	
9.67.134.223	11	11	10	0	10	0	12	17	0	
PORT TOTALS:	19	19		0		0		46		

9.67.131.18	WEIGHT	ACTIVE % 48	NEW % 48	PORT % 4	SYSTEM %					
PORT: 23	NOW	NEW	WT	CONN	WT	CONN	WT	LOAD	WT	LOAD
9.67.129.213	10	10	10	0	10	0	10	71	0	
9.67.134.223	0	0	10	0	10	0	-9999	-1	0	
PORT TOTALS:	10	10		0		0		70		

ADVISOR	PORT	TIMEOUT
reach	0	unlimited
http	80	unlimited
ftp	21	unlimited

- 표준화된 가중치에 대해 모든 서버를 재시작하고 메시지를 관리 프로그램 로그 파일에 기록하려면 다음 명령을 실행하십시오.

```
ndcontrol 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.

ndcontrol manager sensitivity 10

- 스무스 색인을 2.0으로 설정하려면 다음 명령을 실행하십시오.

ndcontrol manager smoothing 2.0

- 관리 프로그램을 시작하고 ndmgr.log(경로는 설정할 수 없음) 로그 파일을 지정하려면 다음 명령을 실행하십시오.

ndcontrol manager start ndmgr.log

- 관리 프로그램과 관련된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

ndcontrol manager status

이 명령으로 다음 예제와 같은 출력이 작성됩니다.

Manager status:

=====

```
Metric port..... 10,004
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
```

- 관리 프로그램을 정지하려면 다음 명령을 실행하십시오.

ndcontrol 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).

ndcontrol manager quiesce 130.40.52.153 now

- 130.40.52.153에서 서버로 더이상 새 연결을 전송하지 않도록 지정하려면 다음 명령을 실행하십시오(주: 결합 시간을 설정한 경우 클라이언트에서 이후의 새 연결은 결합 시간이 만기될 때까지 이 서버로 전송됩니다).

ndcontrol manager quiesce 130.40.52.153

- 관리 프로그램에서 이전에 작업 거부된 130.40.52.153에 있는 서버에 0보다 높은 가중치 제공을 시작할 수 있도록 지정하려면 다음을 수행하십시오.

```
ndcontrol manager unquiesce 130.40.52.153
```

- 관리 프로그램의 현재 버전 번호를 표시하려면 다음을 수행하십시오.

```
ndcontrol manager version
```

ndcontrol metric -- 시스템 메트릭 구성

```
▶▶—ndcontrol—metric—add—cluster+c2+...+cN:metric+metric1+...+metricN—▶▶
—remove—cluster+c2+...+cN:metric+metric1+...+metricN—
—proportions—cluster+c2+...+cN proportion1 prop2 prop3...propN—
—status—cluster+c2+...+cN:metric+metric1+...+metricN—
```

add

지정된 메트릭을 추가합니다.

cluster

클라이언트에 연결되는 주소. 주소는 시스템의 호스트 이름 또는 점분리 10진 수 IP 주소입니다. 추가 클러스터는 더하기 부호(+)로 구분됩니다.

주: Cisco Consultant의 경우, 클러스터 주소는 Cisco CSS Switch 구성에서 소유자 컨텐츠 규칙의 VIP(가상 IP) 주소에 해당합니다.

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

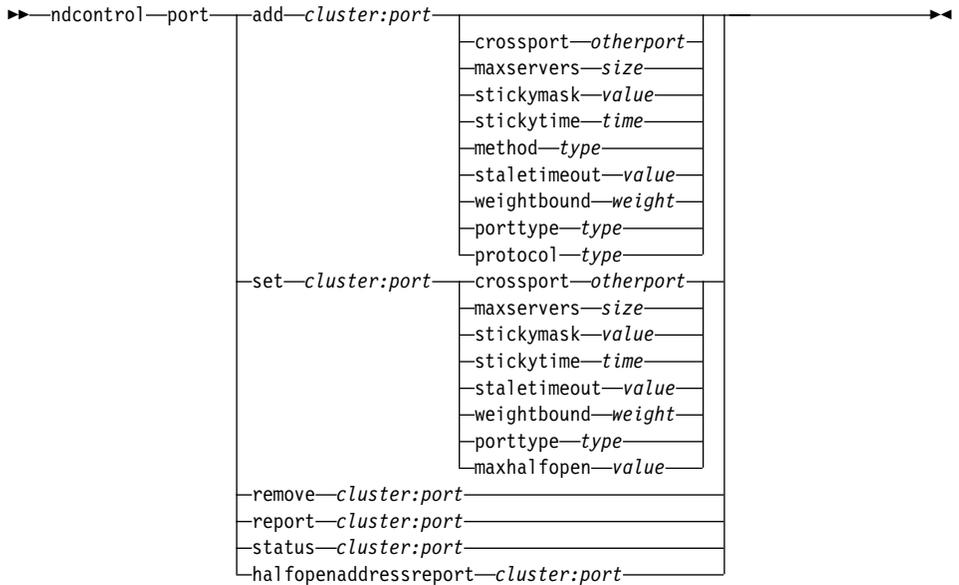
- 시스템 메트릭을 추가하려면 다음 명령을 실행하십시오.
`sscontrol metric add site1:metric1`
- 두 개의 시스템 메트릭을 사용하여 사이트 이름에 비율을 설정하려면 다음 명령을 실행하십시오.
`sscontrol metric proportions site1 0 100`
- 지정된 메트릭과 연관된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.
`sscontrol metric status site1:metric1`

```
sscontrol 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
```

ndcontrol 포트 -- 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.

주: Network Dispatcher의 Mailbox Locator 구성요소인 경우 포트를 추가하기 전에 시스템에서 별명이 지정된 클러스터 IP가 있어야 합니다. **add port** 명령은 클러스터에 바인드하는 Java 프록시를 시작하려고 하므로 IP가 IP 스택에 있어야 합니다.

Windows에서, IP가 Windows 네트워킹 설정에 있어야 함을 의미합니다. **cluster configure** 명령은 IP 별명을 시뮬레이트만 하므로 이 명령만으로는 충분하지 않으며 프록시는 가짜 IP로 바인드할 수 없습니다. 기타 모든 운영 체제의 경우, **cluster configure** 명령이 ifconfig를 사용하여 IP의 별명을 지정하므로 이 명령이 적합합니다.

cluster

기호 이름 또는 점분리 10진수 형식의 클러스터 주소. 콜론(:)을 와일드 카드로 사용할 수 있습니다. 예를 들어, `ndcontrol 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 214 페이지의 『포트간 친근성』에서 자세한 정보를 참조하십시오.

주: Crossport는 Dispatcher 구성요소에만 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. 자세한 내용은 215 페이지의 『친근성 주소 마스크』를 참조하십시오.

주: stickymask 키워드는 Dispatcher 구성요소에만 적용됩니다.

value

stickymask 값은 사용자가 마스크하려는 32비트 IP 주소의 상위 순서 비트 번 호입니다. 가능한 값은 8, 16, 24 및 32입니다. 기본값은 32이며, 연관 관계 주소 마스크 기능을 사용 불가능하게 합니다.

stickytimer

한 연결의 단기와 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 구성요소의 경우

- Dispatcher cbr 전달 방법
 - 포트 stickytimer을 0이 아닌 값으로 설정하면, 규칙에 연관 관계 유형이 없어야 합니다(기본값). stickytimer이 포트에 설정될 경우 규칙 기반 연관 관계(수동 쿠키 및 URI)는 동시에 존재할 수 없습니다.
 - stickytimer 값을 설정하면 SSL ID 연관 관계를 사용할 수 있기 때문에 콘텐츠 규칙을 포트에 추가할 수 없습니다.
- Dispatcher mac 및 nat 전달 방법
 - 포트 stickytimer을 0이 아닌 값으로 설정하면, 규칙에 연관 관계 유형을 설정할 수 없습니다. stickytimer이 포트에 설정될 경우 규칙 기반 연관 관계는 동시에 존재할 수 없습니다.
 - stickytimer 값을 설정하면 IP 주소 연관 관계를 사용할 수 있습니다.

- Server Directed Affinity API를 사용하는 경우 stickytime은 1로 설정되어야 합니다.

CBR 구성요소의 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.

방법

전달 방법. MAC 전달, NAT/NAPT 전달 또는 Content Based Routing 전달 방법이 있습니다. ndcontrol executor 명령의 clientgateway 매개변수에서 0이 아닌 IP 주소를 먼저 지정하지 않으면, NAT/NAPT 또는 Content Based Routing 전달 방법을 추가할 수 없습니다. 자세한 정보는 61 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 방법)』 및 63 페이지의 『Dispatcher content-based routing(cbr 전달 방법)』을 참조하십시오.

주: 백엔드 서버가 리턴 주소와 동일한 서브넷에 있고 content-based routing 전달 방법 또는 NAT/NAPT 전달 방법을 사용하는 경우, 라우터 주소를 백엔드 서버 주소로 정의해야 합니다.

type

전달 방법 유형. 가능한 값은 mac, nat 또는 cbr입니다. 기본값은 mac(MAC 전달)입니다.

staletimeout

연결이 제거되기 전에 연결에서 비활동 상태로 있을 수 있는 시간(초 단위). Dispatcher 또는 CBR 구성요소의 경우, 포트 21(FTP)에 대한 기본값은 900이며 포트 23(텔넷)에 대한 기본값은 32,000,000입니다. 기타 모든 포트의 경우, 기본값은 300입니다. 실행 프로그램 또는 클러스터 레벨에서 staletimeout을 설정할 수도 있습니다. 자세한 정보는 233 페이지의 『활동해제 제한 시간 값 사용』을 참조하십시오.

주: Mailbox Locator의 경우, staletimeout은 이들 프로토콜에 대한 비활동 자동 로그아웃 타이머에 해당합니다. Mailbox Locator의 경우 staletimeout

기본값은 60초이며, 이 값은 POP3 및 IMAP에 대한 비활동 제한 시간을 덮어씁니다. Mailbox Locator stalemtimeout에 대한 자세한 정보는 112 페이지의 『POP3/IMAP 활동 해제 타이머 덮어쓰기』를 참조하십시오.

value

stalemtimeout의 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 가중 바운드를 표시하는 1에서 100 사이의 숫자.

porttype

포트 유형.

주: porttype은 Dispatcher에만 적용됩니다.

type

가능한 값은 **tcp**, **udp** 및 **both**입니다. 기본값은 both(tcp/udp)입니다.

protocol

프록시 프로토콜 유형(POP3 또는 IMAP). 프로토콜 매개변수는 Mailbox Locator에 포트를 추가할 때 필요합니다.

주: protocol은 Mailbox Locator에만 적용됩니다.

type

가능한 값은 **POP3** 또는 **IMAP**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. 222 페이지의 『서비스 거부 중지 감지』에서 자세한 정보를 참조하십시오.

주: maxhalfopen은 Dispatcher에만 적용됩니다.

value

maxhalfopen 값. 기본값은 0입니다(검사가 수행되지 않음).

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 *port*를 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. 222 페이지의 『서비스 거부 중지 감지』에서 자세한 정보를 참조하십시오.

예제

- 포트 80과 23을 클러스터 주소 130.40.52.153에 추가하려면 다음 명령을 실행하십시오.

```
ndcontrol port add 130.40.52.153:80+23
```
- 와일드 카드 포트를 클러스터 주소 130.40.52.153에 추가하려면 다음 명령을 실행하십시오.

```
ndcontrol port set 130.40.52.153:0
```
- Mailbox Locator의 경우 POP3 프로토콜용 포트 20을 클러스터 주소 9.37.60.91에 추가하려면 다음 명령을 실행하십시오.

```
mlcontrol port add 9.37.60.91:20 protocol pop3
```

- 클러스터 주소 130.40.52.153에 있는 포트 80에 최대 가중치 10을 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol port set 130.40.52.153:80 weightbound 10
```

- 클러스터 주소 130.40.52.153에서 포트 80 및 포트 23에 대해 stickytime 값을 60초로 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol 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.

```
ndcontrol port set 130.40.52.153:80 crossport 23
```

- 클러스터 주소 130.40.52.153에서 포트 23을 제거하려면 다음 명령을 실행하십시오.

```
ndcontrol port remove 130.40.52.153:23
```

- 클러스터 주소 9.67.131.153에서 포트 80의 상태를 확보하려면 다음 명령을 실행하십시오.

```
ndcontrol port status 9.67.131.153:80
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Port Status:

```
Port number ..... 80
Cluster address ..... 9.67.131.153
Number of servers ..... 2
Stale timeout ..... 30
Weight bound ..... 20
Maximum number of servers ..... 32
Sticky time ..... 0
Port type ..... tcp/udp
Forwarding method ..... MAC Based Forwarding
Sticky mask bits ..... 32
Cross Port Affinity ..... 80
Max Half Open Connections ..... 0
```

- 클러스터 주소 9.67.127.121에서 포트 80에 대한 반개방 주소 보고서를 얻으려면 다음 명령을 실행하십시오.

```
ndcontrol 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

rule

사용자가 규칙에 대해 선택한 이름. 이 이름에는 영숫자, 밑줄, 하이픈 또는 마침표가 포함될 수 있습니다. 이 이름은 한 개부터 20개까지의 문자가 가능하며 공백을 포함할 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 관리 프로그램이 실행 중인 경우에만 작동합니다.

active 규칙이 포트의 총 작동 중인 연결 수에 기초합니다. 이 규칙은 관리 프로그램이 실행 중인 경우에만 작동합니다.

port 규칙이 클라이언트 포트에 기초합니다.

주: port는 CBR에 적용하지 않습니다.

service

이 규칙은 IP 헤더의 서비스 유형(TOS) 바이트 필드에 기반을 둡니다.

주: Service는 Dispatcher 구성요소에만 적용됩니다.

reservedbandwidth

이 규칙은 서버 집합이 전달하는 대역폭(초당 KB 단위)을 기반으로 합니다. 자세한 정보는 201 페이지의 『예약된 대역폭 및 공유 대역폭에 따라 규칙 사용』 및 202 페이지의 『예약된 대역폭 규칙』을 참조하십시오.

주: reservedbandwidth는 Dispatcher 구성요소에만 적용됩니다.

sharedbandwidth

이 규칙은 실행 프로그램 또는 클러스터 레벨에서 공유될 대역폭(초당 KB)을 기반으로 합니다. 자세한 정보는 201 페이지의 『예약된 대역폭 및 공유 대역폭에 따라 규칙 사용』 및 202 페이지의 『공유 대역폭 규칙』을 참조하십시오.

주: sharedbandwidth는 Dispatcher 구성요소에만 적용됩니다.

true 이 규칙은 항상 참입니다. 이를 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 점분리 10진수 형식의 클라이언트 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 점분리 10진수 형식의 클라이언트 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, 만일 오전 3시와 4시 사이의 한 시간을 지정하려면, 최소 범위로 **3**을 지정하고 최대 범위도 **3**을 지정하십시오. 이는 3:00에서 시작하여 3:59로 끝나는 모든 시간(분 단위)을 지정합니다. 최소 범위를 **3**으로 지정하고 최대 범위를 **4**로 지정하면 3:00에서부터 4:59까지의 두 시간을 나타냅니다.

connections

정수. 기본값은 2의 32 제곱에서 1을 뺀 값입니다.

active 정수. 기본값은 2의 32 제곱에서 1을 뺀 값입니다.

port 정수. 기본값은 65535입니다.

reservedbandwidth

정수(초당 KB). 기본값은 2의 32 제곱에서 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 우선순위는 기존 규칙의 현재 최하위 우선순위 + 10으로 계산됩니다. 예를 들면, 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

내용 유형 규칙에 사용되는 패턴을 지정합니다.

pattern

사용되는 패턴. 유효한 값에 대한 자세한 정보는 347 페이지의 『부록C. 콘텐츠 규칙(패턴) 구문』을 참조하십시오.

tos

service 유형 규칙에 사용되는 “서비스 유형”(TOS) 값을 지정합니다.

주: TOS는 Dispatcher 구성요소에만 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 200 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.

218 페이지의 『활성 쿠키 연관 관계』에서 자세한 정보를 참조하십시오.

주: 규칙 stickytime은 CBR 구성요소에 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 Network Dispatcher가 생성한 쿠키를 기준으로 동일한 서버로 연관 관계를 사용하여 웹 통신량을 로드 밸런싱 할 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 cookiename 매개변수를 사용해야 합니다.

"URI" 연관 관계 유형은 캐시 크기를 효율적으로 증대시키는 방법으로 Caching Proxy 서버로 웹 통신량을 로드 밸런스할 수 있습니다.

자세한 정보는 218 페이지의 『활성 쿠키 연관 관계』, 219 페이지의 『수동 쿠키 연관 관계』 및 220 페이지의 『URI 연관 관계』을 참조하십시오.

주: 연관 관계는 Dispatcher 구성요소의 cbr 전달 방법을 사용하여 구성된 규칙 및 CBR 구성요소에 적용됩니다.

affinity_type

가능한 연관 관계 유형 값은 없음(기본값), activecookie, passivecookie 또는 uri입니다.

cookiename

Network Dispatcher ID의 역할을 하는 관리자가 임의로 설정한 이름. Network Dispatcher가 클라이언트 HTTP 헤더 요청에서 찾아야 하는 이름입니다. 쿠키 값과 함께 쿠키 이름은 Network Dispatcher ID 역할을 하므로 Network Dispatcher가 웹 사이트의 후속 요청을 동일한 서버 시스템으로 전송할 수 있습니다. 쿠키 이름은 "수동 쿠키" 연관 관계와 함께만 적용할 수 있습니다.

219 페이지의 『수동 쿠키 연관 관계』에서 자세한 정보를 참조하십시오.

주: 쿠키 이름은 Dispatcher 구성요소의 cbr 전달 방법을 사용하여 구성된 규칙 및 CBR 구성요소에 적용됩니다.

value

쿠키 이름 값.

evaluate

이 옵션은 Dispatcher 구성요소에서만 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 206 페이지의 『규칙에 대한 서버 평가 옵션』을 참조하십시오

level

가능한 값은 포트 또는 규칙입니다. 기본값은 포트입니다.

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 202 페이지의 『공유 대역폭 규칙』을 참조하십시오.

level

가능한 값은 실행 프로그램 또는 클러스터입니다.

dropserver

규칙 집합에서 서버를 제거합니다.

server

기호 이름 또는 점분리 10진수 형식의 TCP 서버 시스템의 IP 주소.

또는 서버 파티션을 사용할 경우, 논리 서버의 고유 이름. 172 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 집합에 삽입합니다.

예제

- 항상 참이 되는 규칙을 추가하려면, 최소 범위 또는 최대 범위를 지정하지 마십시오.

```
ndcontrol 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.

```
ndcontrol 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.

```
ndcontrol rule add cluster1:80:timerule type time beginrange 11 endrange 14  
ndcontrol rule useserver cluster1:80:timerule server05
```

- IP 헤더에서 TOS 바이트 필드의 내용에 기반하여 규칙을 작성하려면 다음 명령을 실행하십시오.

```
ndcontrol 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.

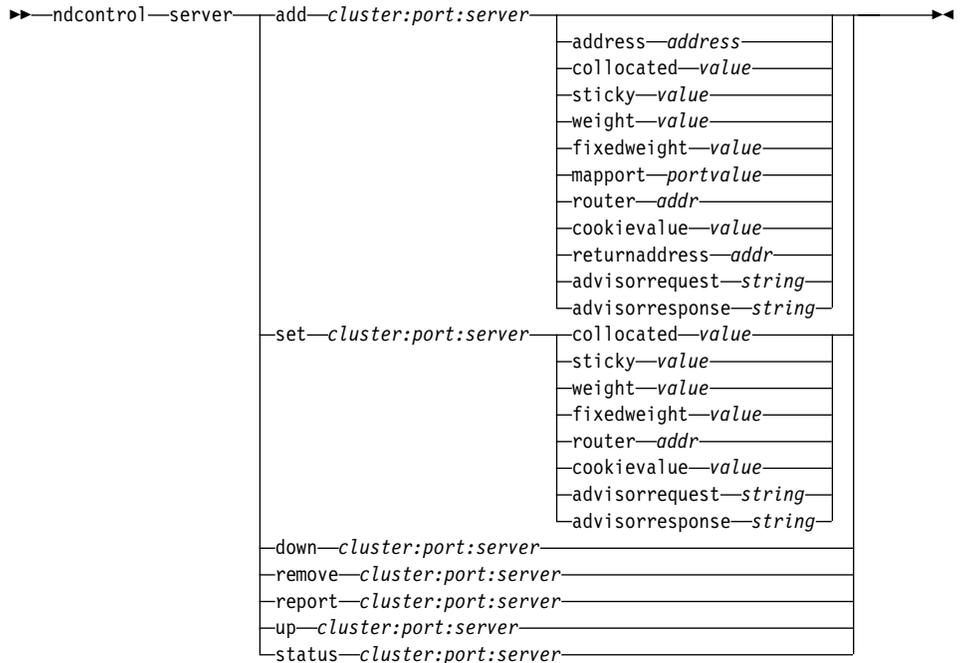
```
ndcontrol rule add 9.67.131.153:80:rbwrule type reservedbandwidth  
beginrange 0 endrange 100 evaluate rule
```

- 클러스터 레벨에서 사용하지 않은 대역폭을 보충할 공유 대역폭을 기반으로 규칙을 작성하려면 다음 명령을 실행하십시오(주: ndcontrol cluster 명령을 사용하여 클러스터 레벨에서 공유할 수 있는 최대 대역폭(초당 KB)을 먼저 지정해야 함).

```
ndcontrol cluster set 9.67.131.153 sharedbandwidth 200
```

```
ndcontrol rule add 9.67.131.153:80:shbwrule type sharedbandwidth  
sharelevel cluster
```

ndcontrol server -- 서버 구성



add

이 서버를 추가합니다.

cluster

기호 이름 또는 점분리 10진수 형식의 클러스터 주소. 콜론(:)을 와일드 카드로 사용할 수 있습니다. 예를 들어, `ndcontrol 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, `ndcontrol server add ::ServerA` 명령은 모든 포트의 모든 클러스터에 ServerA를 추가합니다.

주: 추가 포트는 더하기 부호(+)로 구분됩니다.

server

server는 기호 이름 또는 점분리 10진수 형식의 TCP 서버 시스템의 고유 IP 주소입니다.

IP 주소에 분석되지 않는 고유 이름을 사용하는 경우, **ndcontrol server add** 명령에 서버 **address** 매개변수를 제공해야 합니다. 172 페이지의 『서버 파티션: 물리적 서버에 구성된 논리 서버(IP 주소)』에서 자세한 정보를 참조하십시오.

주: 추가 서버는 더하기 부호(+)로 구분됩니다.

주소(address)

호스트 이름 또는 점분리 10진수 형식의 TCP 서버 시스템의 고유 IP 주소 서버를 해석할 수 없는 경우, 실제 서버 시스템의 주소를 제공해야 합니다. 172 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. collocated 옵션은 Windows 2000 플랫폼에는 적용되지 않습니다.

주: collocated 매개변수는 Dispatcher의 mac 또는 nat 전달 방법을 사용할 때만 유효합니다. Mailbox Locator, Site Selector 및 Cisco Consultant 를 모든 플랫폼에서 결합 배치할 수 있지만 이 키워드는 필요하지 않습니다. 자세한 내용은 175 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 포트

에 대해 요청을 발행하는 다음 번에 해당 서버로 리턴되지 않습니다. 이것은 규칙을 사용하고 있는 특정 상황에 유용합니다. 자세한 내용은 216 페이지의 『규칙 친근성 무시』를 참조하십시오.

value

sticky 값: yes 또는 no. 기본값은 yes입니다.

weight

이 서버의 가중치를 표시하는 0에서 100 사이의 숫자(지정된 포트의 가중 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 가중 바운드 값을 2로 나눈 값입니다. 관리 프로그램이 실행되고 있는 경우, 이 설정은 신속하게 겹쳐씹니다.

value

서버 가중치 값.

fixedweight

fixedweight 옵션을 사용하여 관리 프로그램이 서버 가중치를 수정할 수 있는지 여부를 지정할 수 있습니다. fixedweight 값을 yes로 설정하면 관리 프로그램이 실행될 때 서버 가중치를 수정할 수 없습니다. 자세한 내용은 155 페이지의 『관리 프로그램 고정 가중치』를 참조하십시오.

value

fixedweight 값: yes 또는 no. 기본값은 no입니다.

mapport

클라이언트 요청의 목적지 포트 번호(Dispatcher의 경우)를 Dispatcher가 클라이언트의 요청을 로드 밸런싱하는 데 사용하는 서버의 포트 번호에 대응시킵니다. Network Dispatcher가 하나의 포트에서 클라이언트의 요청을 수신하고 서버 시스템의 다른 포트에 이를 전송할 수 있도록 합니다. mapport를 사용하면 클라이언트의 요청을 다중 서버 디먼이 실행 중인 서버로 로드 밸런싱할 수 있습니다.

주: mapport는 Dispatcher(nat 또는 cbr 전달 방법을 사용하는) 및 CBR에 적용됩니다. Dispatcher의 경우, 61 페이지의 『Dispatcher의 NAT/NAPT(nat 전달 방법)』 및 63 페이지의 『Dispatcher content-based

routing(cbr 전달 방법)』을 참조하십시오. CBR의 경우, 92 페이지의 『SSL의 클라이언트-투-프록시 및 HTTP의 프록시-투-서버의 로드 밸런스』를 참조하십시오.

portvalue

맵 포트 번호 값. 기본값은 클라이언트 요청의 목적지 포트 번호입니다.

router

광역 네트워크를 설정하고 있는 경우, 원격 서버에 대한 라우터 주소. 기본값은 0으로, 로컬 서버를 나타냅니다. 일단 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 서버로 만들 수 없습니다. 서버를 제거했다 다시 추가해야 합니다. 177 페이지의 『광역 Dispatcher 지원 구성』에서 자세한 정보를 참조하십시오.

주: 라우터는 Dispatcher에만 적용됩니다. nat 또는 cbr 전달 방법을 사용하는 경우, 구성에 서버를 추가할 때 라우터 주소를 지정해야 합니다.

addr

라우터 주소의 값.

cookievalue

cookievalue는 쿠키 이름/쿠키 값 쌍의 서버를 표시하는 임의의 값입니다. 쿠키 이름과 함께 쿠키 값은 Network Dispatcher ID의 역할을 하여 Network Dispatcher가 후속 클라이언트 요청을 동일한 서버 시스템으로 전송할 수 있도록 합니다. 219 페이지의 『수동 쿠키 연관 관계』에서 자세한 정보를 참조하십시오.

주: cookievalue는 Dispatcher(cbr 전달 방법 사용) 및 CBR에 대해 유효합니다.

value

임의의 값입니다. 기본값은 쿠키 값 없음입니다.

returnaddress

고유 IP 주소 또는 호스트 이름. Dispatcher가 서버로 클라이언트 요청을 로

드 밸런싱할 때 출발지 주소로 사용하는 Dispatcher 시스템에 구성된 주소입니다. 서버가 Dispatcher 시스템으로 패킷을 리턴하여 클라이언트로 직접 패킷을 전송하지 않고 요청 내용을 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 동일해서는 안 됩니다.

주: Returnaddress는 Dispatcher에만 적용됩니다. nat 또는 cbr 전달 방법을 사용하는 경우, 구성에 서버를 추가할 때 리턴 주소를 지정해야 합니다.

addr

리턴 주소 값.

advisorrequest

HTTP 권고자는 권고자 요청 문자열을 사용하여 서버 상태를 조회합니다. HTTP 권고자가 권고하는 서버에만 유효합니다. 이 값을 사용하려면 HTTP 권고자를 시작해야 합니다. 174 페이지의 『HTTP 권고자 요청/응답(URL) 옵션』에서 자세한 정보를 참조하십시오.

주: advisorrequest는 Dispatcher 및 CBR 구성요소에 적용됩니다.

string

HTTP 권고자가 사용하는 문자열 값. 기본값은 HEAD / HTTP/1.0입니다.

주: 문자열에 공백이 포함된 경우 -

- **ndcontrol**>> 셸 프롬프트에서 명령을 발행할 경우, 문자열을 따옴표(')로 묶어야 합니다. 예제: **server set cluster:port:server advisorrequest "head / http/2.0"**
- 운영 체제 프롬프트에서 **ndcontrol** 명령을 발행할 경우, 텍스트 앞에 \"를, 텍스트 뒤에 \"를 표시해야 합니다.
예제: **ndcontrol server set cluster:port:server advisorrequest \"head / http/2.0\"**

advisorresponse

HTTP 응답에서 HTTP 권고자가 스캔하는 권고자 응답 문자열. HTTP 권고

자가 권고하는 서버에만 유효합니다. 이 값을 사용하려면 HTTP 권고자를 시작해야 합니다. 174 페이지의 『HTTP 권고자 요청/응답(URL) 옵션』에서 자세한 정보를 참조하십시오.

주: `advisoresponse`는 Dispatcher 및 CBR 구성요소에 적용됩니다.

string

HTTP 권고자가 사용하는 문자열 값. 기본값은 널입니다.

주: 문자열에 공백이 포함된 경우 -

- `ndcontrol>>` 셸 프롬프트에서 명령을 발행할 경우, 문자열을 따옴표(')로 묶어야 합니다.
- 운영 체제 프롬프트에서 `ndcontrol` 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.

remove

해당 서버를 제거합니다.

report

해당 서버에 대해 보고합니다.

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.

```
ndcontrol server add 130.40.52.153:80:27.65.89.42
```

- 27.65.89.42에서 비결합(규칙 연관 관계 무시 기능) 상태로 서버를 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol server set 130.40.52.153:80:27.65.89.42 sticky no
```

- 27.65.89.42에 있는 서버를 단절되도록 표시하려면 다음 명령을 실행하십시오.

```
ndcontrol 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.

```
ndcontrol server remove ::27.65.89.42
```

- 27.65.89.42에서 결합 배치(서버가 Network Dispatcher와 동일한 시스템에 위치)로 서버를 설정하려면 다음 명령을 실행하십시오.

```
ndcontrol 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.

```
ndcontrol server set 130.40.52.153:80:27.65.89.42 weight 10
```

- 27.65.89.42에 있는 서버를 연결되도록 표시하려면 다음 명령을 실행하십시오.

```
ndcontrol server up 130.40.52.153:80:27.65.89.42
```

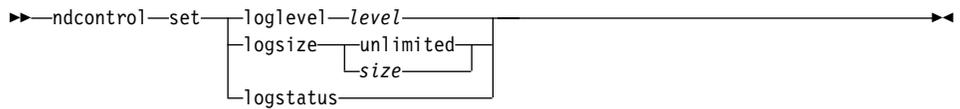
- 원격 서버를 추가하려면 다음 명령을 실행하십시오.

```
ndcontrol server add 130.40.52.153:80:130.60.70.1 router 130.140.150.0
```

- HTTP 권고자가 HTTP 포트 80에서 서버 27.65.89.42에 대해 HTTP URL 요청 HEAD/HTTP/2.0을 조회할 수 있도록 하려면 다음 명령을 실행하십시오.

```
ndcontrol server set 130.40.52.153:80:27.65.89.42  
  advisorrequest "\"HEAD / HTTP/2.0\""
```

ndcontrol set -- 서버 로그 구성



loglevel

ndserver가 그 활동을 기록하는 레벨.

level

loglevel 기본값은 0입니다. 범위는 0에서 5입니다. 가능한 값은 다음과 같습니다. 0은 없음, 1은 최소, 2는 기본, 3은 중간, 4는 고급, 5는 자세합니다.

logsize

로그 파일에 기록할 최대 바이트 수.

size

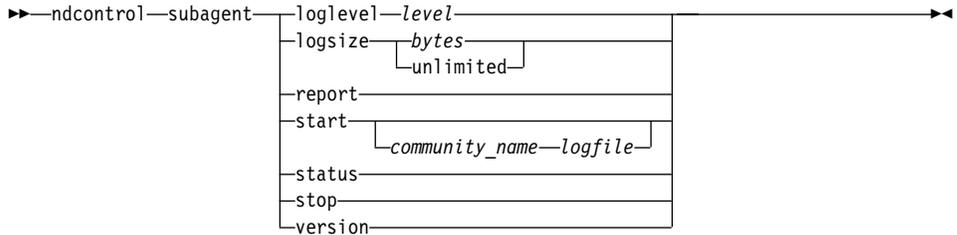
logsize의 기본값은 1MB입니다.

logstatus

서버 로그 설정(로그 레벨 및 로그 크기)을 표시합니다.

ndcontrol subagent -- SNMP 서브에이전트 구성

주: ndcontrol subagent 명령 구문 다이어그램은 CPR 또는 Mailbox Locator에는 적용되지 않습니다.



loglevel

서브에이전트가 활동을 파일에 기록하는 레벨.

level

레벨 번호(0-5). 번호가 커질수록 더 자세한 정보가 관리 프로그램 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 파일의 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면, 후속 항목은 파일의 맨 위에서부터, 이전 로그 항목에 겹쳐 기록됩니다. 로그 크기는 로그의 현재 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 **unlimited** 단어를 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 public 입니다.

log file

SNMP 서브에이전트 데이터가 기록되는 파일 이름. 로그의 각 레코드에는 시간 소인이 표시됩니다. 기본값은 subagent.log입니다. 기본 파일은 **logs** 디렉토리에 설치됩니다. 413 페이지의 『부록F. 예제 구성 파일』을 참조하십시오. 로그 파일이 보존되는 디렉토리를 변경하려면, 232 페이지의 『로그 파일 경로 변경』을 참조하십시오.

status

전역 및 그 기본값으로 설정될 수 있는 SNMP 서브에이전트의 모든 값에 대한 현재 상태를 표시합니다.

version

서브에이전트의 현재 버전을 표시합니다.

예제

- 공동체 이름 **bigguy**로 서브에이전트를 시작하려면 다음 명령을 실행하십시오.
`ndcontrol subagent start bigguy bigguy.log`

부록C. 콘텐츠 규칙(패턴) 구문

이 부록에서는 CBR 구성요소의 콘텐츠 규칙 구문 사용법과 Dispatcher 구성요소의 cbr 전달 방법을 사용법 예제 및 시나리오와 함께 설명합니다.

콘텐츠 규칙(패턴) 구문:

규칙 유형에 대한 "내용"을 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

버전 요청 고유 버전(HTTP/1.0 또는 HTTP/1.1)

호스트 호스트의 값: 헤더.

주: HTTP/1.0 프로토콜에서는 선택적입니다.

<key> 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, 다음 명령은 **cbrcontrol**>> 프롬프트를 사용할 경우에만 유효합니다.

```
rule add 10.1.203.4:80:cbr_prod_rule_ek type content
  pattern client=181.0.153.222&uri=http://10.1.203.4/nipoek/*
```

특수 문자를 사용할 때, 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 "client=181.0.153.222&uri=http://10.1.203.4/nipoek/*"
```

큰따옴표를 사용하지 않으면 규칙이 CBR에서 저장될 때 일부 패턴이 잘릴 수 있습니다. 큰따옴표는 **cbrcontrol**>> 명령 프롬프트를 사용할 경우에 지원되지 않습니다.

다음은 패턴 구문 사용 예제 및 가능한 시나리오 모음입니다.

시나리오 1

클러스터 이름을 하나 설정하는 데 표준 HTML 내용에 대한 웹 서버 집합, WAS(WebSphere Application Server)가 사용하는 servlet 요청에 대한 또다른 웹 서버 집합, NSF 파일에 대한 또다른 Lotus Notes 서버 집합 등이 필요합니다. 요청된 페이지를 구별하려면 클라이언트 데이터에 액세스해야 합니다. 또한 요청된 페이지를 해당 서버로 전송해야 합니다. 규칙과 일치하는 내용 패턴은 이런

태스크를 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 Network Dispatcher에 도착하면, servlets 규칙을 먼저 확인하지만 일치하지 않습니다. 그러면, notes 규칙이 요청을 확인하고 일치함을 리턴합니다. 클라이언트는 server3과 server4 간에 로드 밸런스가 됩니다.

시나리오 2

다른 일반적인 시나리오는 기본 웹 사이트가 몇 개의 별개 내부 그룹을 제어하는 경우입니다. 예를 들어, `www.company.com/software`에는 `www.company.com/hardware` 부분과는 서로 다른 서버 집합 및 내용이 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/software`는 `www.company.com/mainframe/fixes/software`와는 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
```

부록D. Site Selector 명령어 참조서

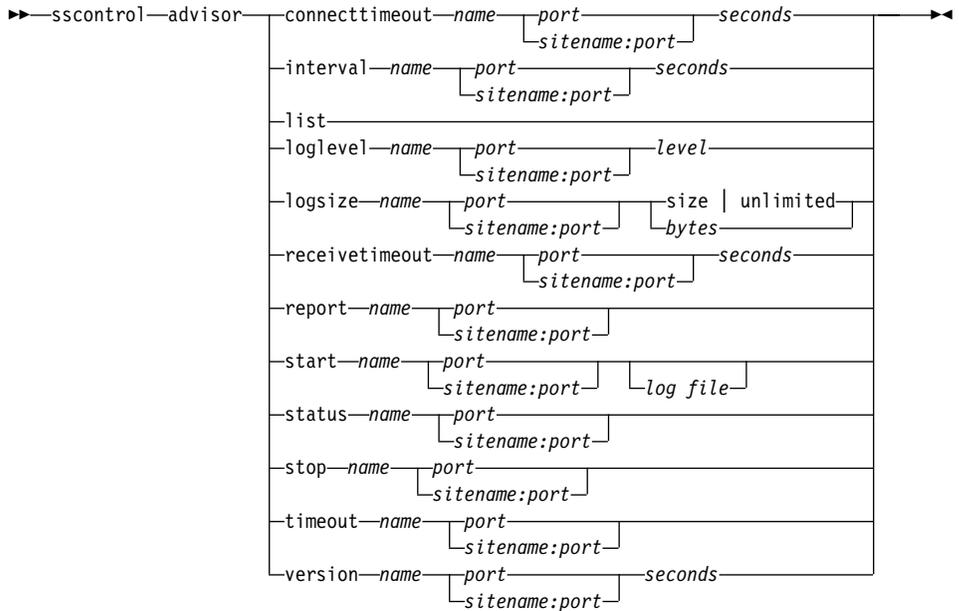
이 부록에서는 다음 Site Selector **sscontrol** 명령 사용 방법을 설명합니다.

- 352 페이지의 『sscontrol advisor - 권고자 제어』
- 358 페이지의 『sscontrol file - 구성 파일 관리』
- 360 페이지의 『sscontrol help - 이 명령의 도움말 표시 또는 인쇄』
- 361 페이지의 『sscontrol manager - 관리 프로그램 제어』
- 366 페이지의 『sscontrol metric - 시스템 메트릭 구성』
- 368 페이지의 『sscontrol nameserver - NameServer 제어』
- 369 페이지의 『sscontrol rule - 규칙 구성』
- 373 페이지의 『sscontrol server - 서버 구성』
- 375 페이지의 『sscontrol set - 서버 로그 구성』
- 376 페이지의 『sscontrol sitename - 사이트 이름 구성』
- 380 페이지의 『sscontrol status - 관리 프로그램 및 권고자가 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 - 권고자 제어



connecttimeout

서버와의 연결 실패를 보고하기 전에 권고자가 대기하는 시간을 설정하십시오. 자세한 정보는 160 페이지의 『권고자 연결 시간 종료 및 서버의 수신 시간 종료』를 참조하십시오.

name

권고자 이름. 가능한 값은 **http**, **ftp**, **ssl**, **smtp**, **imap**, **pop3**, **nntp**, **telnet**, **connect**, **ping**, **WLM** 및 **WTE**입니다. 사용자 정의 권고자 이름은 xxxx 형식이며, 여기서 ADV_xxxx는 조정 권고자를 구현하는 클래스의 이름입니다.

port

권고자가 모니터링하는 포트의 번호.

seconds

서버와의 연결 실패를 보고하기 전에 권고자가 대기하는 시간을 초 단위로 표시하는 양의 정수. 기본값은 권고자 간격에 지정된 값의 3배입니다.

interval

권고자가 정보를 찾아 서버를 조회하는 간격을 설정하십시오.

seconds

서버에 상태 요청을 하는 간격(초 단위)을 나타내는 양의 정수. 기본값은 7입니다.

list

관리 프로그램에 현재 정보를 제공하는 권고자 목록을 표시합니다.

loglevel

권고자 로그에 대한 로그 레벨을 설정합니다.

level

레벨 번호(0-5). 기본값은 1입니다. 번호가 커질수록 더 자세한 정보가 권고자 로그에 기록됩니다. 가능한 값은 다음과 같습니다.

- 0은 없음
- 1은 최소
- 2는 기본
- 3은 중간
- 4는 고급
- 5는 자세히

logsize

권고자 로그의 최대 크기를 설정합니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 이전 로그 항목에 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

권고자 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 권고자가 대기하는 시간을 설정합니다. 자세한 정보는 160 페이지의 『권고자 연결 시간 종료 및 서버의 수신 시간 종료』를 참조하십시오.

seconds

서버에서 수신 실패를 보고하기 전에 권고자가 대기하는 시간을 초 단위로 표시하는 양의 정수. 기본값은 권고자 간격에 지정된 값의 3배입니다.

report

권고자 상태에 대한 보고서를 표시합니다.

start

권고자를 시작합니다. 각 프로토콜에 대해 권고자가 있습니다. 기본 포트는 다음과 같습니다.

권고자 이름	프로토콜	포트
Connect	n/a	사용자 정의
db2	개인용	50000
ftp	FTP	21
http	HTTP	80
imap	IMAP	143
nntp	NNTP	119
PING	PING	0
pop3	POP3	110
smtp	SMTP	25
ssl	SSL	443
telnet	Telnet	23

name

권고자 이름.

sitename:port

sitename 값은 권고자 명령에서 선택적이지만, port 값은 필수입니다. sitename 값이 지정되지 않으면 권고자는 구성된 모든 사용 가능한 사이트 이름에서 실행됩니다. 사이트 이름을 지정하면 권고자는 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 `advisername_port.log`이며 예를 들어, **http_80.log**입니다. 로그 파일이 저장되는 디렉토리를 변경하려면 232 페이지의 『로그 파일 경로 변경』을 참조하십시오.

각 사이트 이름에 대해 하나의 권고자만 시작할 수 있습니다.

status

권고자의 모든 전역값의 기본값 및 현재 상태를 표시합니다.

stop

권고자를 정지합니다.

timeout

관리 프로그램이 권고자 정보를 유효한 것으로 간주하는 시간(초 단위)을 설정합니다. 권고자 정보가 시간 종료 기간보다 이전의 정보라는 것을 관리 프로그램이 발견하면, 관리 프로그램은 권고자가 모니터링하는 포트에서 서버에 대한 가중치를 결정할 때 이 정보를 사용하지 않습니다. 이 시간 종료에 대한 예외는 권고자가 관리 프로그램에 특정 서버가 단절되었다는 것을 알린 경우입니다. 관리 프로그램은 권고자 정보가 시간 종료된 후에도 서버에 대한 해당 정보를 사용합니다.

seconds

초 수를 표시하는 양의 정수 또는 **unlimited**. 기본값은 `unlimited`입니다.

version

권고자의 현재 버전을 표시합니다.

예제

- 서버와의 연결 실패를 보고하기 전에 HTTP 권고자가(포트 80의 경우) 대기하는 시간(30초)을 설정하려면 다음 명령을 실행하십시오.

```
sscontrol advisor connecttimeout http 80 30
```

- FTP 권고자 간격(포트 21의 경우)을 6초로 설정하려면 다음 명령을 실행하십시오.

```
sscontrol advisor interval ftp 21 6
```

- 현재 관리 프로그램에 정보를 제공하는 권고자 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 권고자 로그의 로그 레벨을 0으로 변경하려면 다음 명령을 실행하십시오.

```
sscontrol advisor loglevel http mysite:80 0
```

- mysite라는 사이트의 ftp 권고자 로그 크기를 5000 바이트로 변경하려면 다음 명령을 실행하십시오.

```
sscontrol advisor logsize ftp mysite:21 5000
```

- 서버에서 수신 실패를 보고하기 전에 HTTP 권고자(포트 80의 경우)가 대기하는 시간(60초)을 설정하려면 다음 명령을 실행하십시오.

```
sscontrol advisor receivetimeout http 80 60
```

- FTP 권고자 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 권고자를 시작하려면 다음 명령을 실행하십시오.

```
sscontrol advisor start ftp 21 ftpadv.log
```

- http 권고자와 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
```

- 포트 80에서 http 권고자를 정지하려면 다음 명령을 실행하십시오.

```
sscontrol advisor stop http 80
```

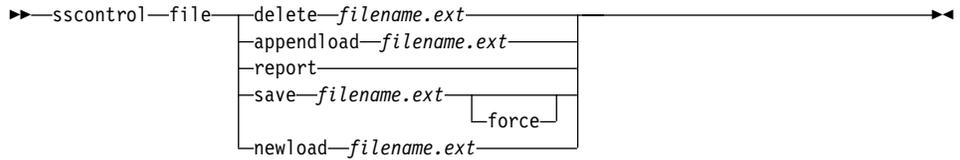
- 권고자 정보에 대한 시간 종료 값을 5초로 설정하려면 다음 명령을 실행하십시오.

```
sscontrol advisor timeout ftp 21 5
```

- ssl 권고자의 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.

- AIX: /usr/lpp/nd/servers/configurations/ss
- Linux: /opt/nd/servers/configurations/ss
- Solaris: /opt/nd/servers/configurations/ss
- Windows 2000:

일반 설치 디렉토리 경로 -- c:\Program

Files\ibm\edge\nd\servers\configurations\구성요소

고유(Native) 설치 디렉토리 경로 -- c:\Program

Files\ibm\nd\servers\configurations\구성요소

force

파일을 기존 파일과 동일한 이름으로 저장하려면 새 파일을 저장하기 전에 **force**를 사용하여 기존 파일을 삭제하십시오. force 옵션을 사용하지 않으면 기존 파일에 겹쳐쓸 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

```
▶▶sscontrol—help—advisor—
file—
help—
host—
manager—
metric—
nameserver—
rule—
server—
set—
sitename—
status—
```

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
advisor       - help on advisor 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
```

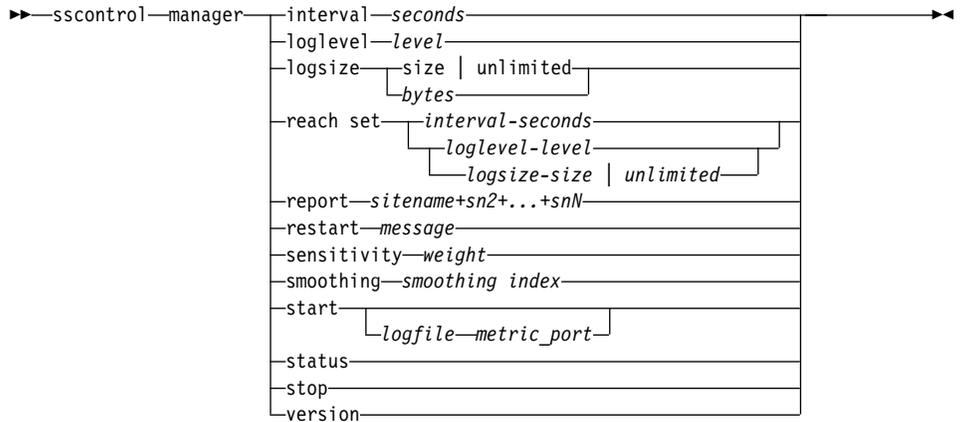
< > 내의 매개변수는 변수입니다.

- 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 manager - 관리 프로그램 제어



interval

관리 프로그램이 서버의 가중치를 갱신하는 빈도를 설정합니다.

seconds

관리 프로그램이 가중치를 갱신하는 빈도를 표시하는 양수(초 단위). 기본값은 2입니다.

loglevel

메트릭 모니터 로그 및 관리 프로그램 로그에 대한 로그 레벨을 설정합니다.

level

레벨 번호(0-5). 번호가 커질수록 더 자세한 정보가 관리 프로그램 로그에 기록됩니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다.

- 0은 없음
- 1은 최소
- 2는 기본
- 3은 중간
- 4는 고급
- 5는 자세히

logsize

관리 프로그램 로그의 최대 크기를 설정합니다. 로그 파일에 최대 크기를 설정

할 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.

bytes

관리 프로그램 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.

reach set

도달 권고자에 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

관리 프로그램 로그 파일에 기록할 메시지.

sensitivity

가중치가 갱신되는 최소 감도를 설정합니다. 이 설정은 외부 정보에 따라 서버에 대한 가중치를 관리 프로그램에서 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

관리 프로그램을 시작합니다.

log file

관리 프로그램 데이터가 기록되는 파일 이름. 로그의 각 레코드에는 시간 소인이 표시됩니다.

기본 파일은 **logs** 디렉토리에 설치됩니다. 413 페이지의 『부록F. 예제 구성 파일』을 참조하십시오. 로그 파일이 보존되는 디렉토리를 변경하려면 232 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

관리 프로그램의 모든 전역값의 현재 상태 및 기본값을 표시합니다.

stop

관리 프로그램을 정지합니다.

version

관리 프로그램 현재 버전을 표시합니다.

예제

- 관리 프로그램의 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`

- 관리 프로그램 로그 크기를 1,000,000 바이트를 설정하려면 다음 명령을 실행하십시오.

```
sscontrol manager logsize 1000000
```

- 관리 프로그램의 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 관리 프로그램 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
```

- 관리 프로그램을 시작하고 ndmgr.log(경로는 설정할 수 없음) 로그 파일을 지정하려면 다음 명령을 실행하십시오.

```
sscontrol manager start ndmgr.log
```

- 관리 프로그램과 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
```

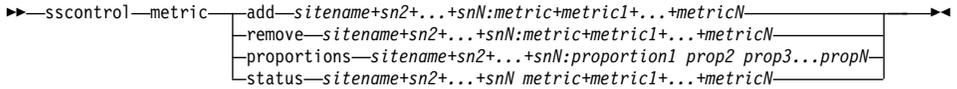
- 관리 프로그램을 정지하려면 다음 명령을 실행하십시오.

```
sscontrol manager stop
```

- 관리 프로그램의 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 sitel:metric1
```
- 두 개의 시스템 메트릭을 사용하여 사이트 이름에 비율을 설정하려면 다음 명령을 실행하십시오.

```
sscontrol metric proportions sitel 0 100
```
- 지정된 메트릭과 연관된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

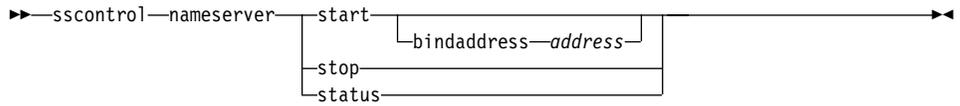
```
sscontrol metric status sitel:metric1
```

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 상자에서 구성된 주소(IP 또는 기호).

stop

이름 서버를 정지합니다.

status

이름 서버 상태를 표시합니다.

sscontrol rule - 규칙 구성

```
►► sscontrol rule add —sitename+sn2+...+snN:rule+r2+...+rN —type—value— value | opts |
dropserver—sitename+sn2+...+snN:rule+r2+...+rN—server+s2+...+sN
remove—sitename+sn2+...+snN:rule+r2+...+rN
set—sitename+sn2+...+snN:rule+r2+...+rN | value | opts |
status—sitename+sn2+...+snN:rule+r2+...+rN
useserver—sitename+sn2+...+snN:rule+r2+...+rN—server+s2+...+sN
```

opts:

```
|
| —beginrange—low—endrange—high—
| —priority—value
| —metricname—value
|
```

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 집합의 모든 서버에 대한 현재 메트릭 값을 기반으로 합니다.

metricavg

규칙은 서버 집합의 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 점분리 10진수 형식의 클라이언트 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 점분리 10진수 형식의 클라이언트 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 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 우선순위가 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 측정할 메트릭 이름.

dropserver

규칙 집합에서 서버를 제거합니다.

서버

기호 이름 또는 점분리 10진수 형식의 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 집합에 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 점분리 10진수 형식의 TCP 서버 시스템의 IP 주소.

주: 추가 서버는 더하기 부호(+)로 구분됩니다.

metricaddress

Metric Server의 주소.

address

기호 이름 또는 점분리 10진수 형식으로 된 서버의 주소.

weight

이 서버의 가중치를 표시하는 0에서 100 사이의 숫자(지정된 사이트 이름의 최대 가중 바운드를 초과해서는 안됨). 가중치를 0으로 설정하면 새로운 요청이 서버로 전송되지 않습니다. 기본값은 지정된 사이트 이름의 최대 가중 바운드 값을 2로 나눈 값입니다. 관리 프로그램이 실행되고 있는 경우, 이 설정은 신속하게 겹쳐쓰입니다.

value

서버 가중 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.

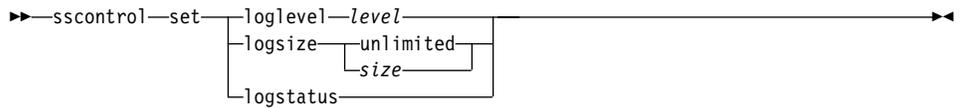
```
sscontrol server add site1:27.65.89.42
```
- 27.65.89.42에 있는 서버를 단절되도록 표시하려면 다음 명령을 실행하십시오.

```
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.

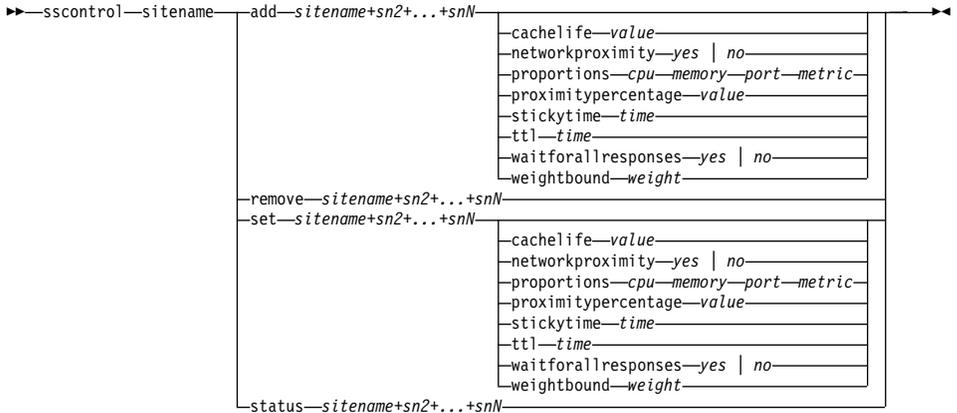
size

logsize의 기본값은 1MB입니다.

logstatus

서버 로그 설정(로그 레벨 및 로그 크기)을 표시합니다.

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 유효하여 캐시에 저장되는 시간. 기본값은 1800입니다. 125 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 밸런싱 결정 시 proximity 응답을 사용하십시오. proximity를 on 또는 off로 설정하십시오. 125 페이지의 『네트워크 근접 기능 사용』에서 자세한 정보를 참조하십시오.

value

선택사항은 yes 또는 no입니다. 기본값은 no로 네트워크 근접성이 작동하지 않습니다.

proportions

관리 프로그램이 서버 가중치를 설정하는 데 사용하는 Metric Server에 대한 `cpu`, `memory`, `port`(권고자의 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 권고자가 보낸 입력.

system Metric Server가 보낸 입력.

proximitypercentage

`proximity` 응답 vs 서버 상태의 중요도를 설정(관리 프로그램 가중치)합니다. 125 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 time`의 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. 125 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. 사이트 이름에 설정된 가중 바운드 값에 **server weight**를 사용하여 개별 서버에 대해 덮어쓸 수 있습니다. 사이트 이름 가중 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 - 관리 프로그램 및 권고자가 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 |  
-----
```

부록E. Consultant for Cisco CSS Switches 명령어 참조 서

이 부록에서는 Consultant for Cisco CSS Switches에서 다음 **lbcontrol** 명령을 사용하는 방법을 설명합니다.

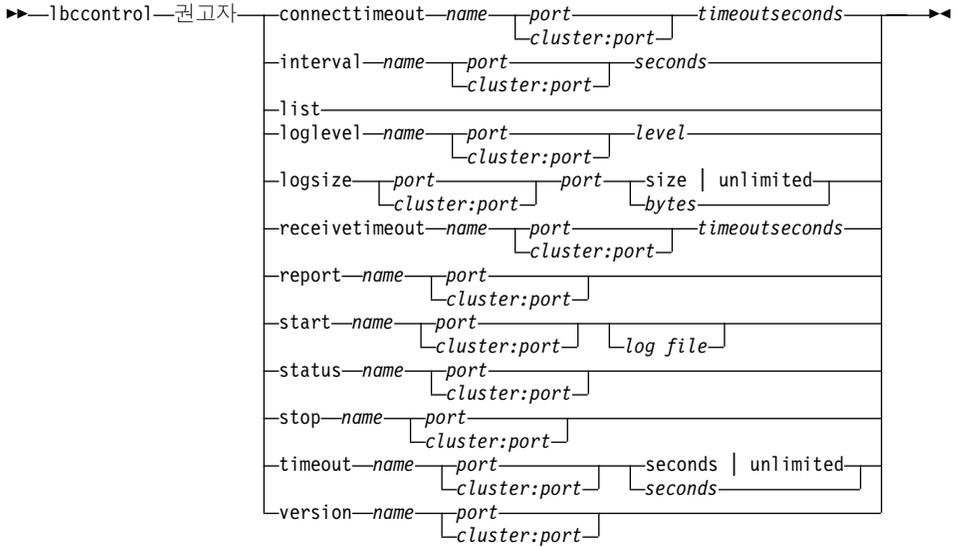
- 382 페이지의 『lbcontrol advisor -- 권고자 제어』
- 388 페이지의 『lbcontrol cluster -- 클러스터 구성』
- 390 페이지의 『lbcontrol executor -- 실행 프로그램 제어』
- 392 페이지의 『lbcontrol file -- 구성 파일 관리』
- 394 페이지의 『lbcontrol help -- 이 명령의 도움말 표시 또는 인쇄』
- 395 페이지의 『lbcontrol host -- 원격 시스템 구성』
- 396 페이지의 『lbcontrol log -- 2진 로그 파일 제어』
- 397 페이지의 『lbcontrol manager -- 관리 프로그램 제어』
- 404 페이지의 『lbcontrol metric -- 시스템 메트릭 구성』
- 406 페이지의 『lbcontrol port -- 포트 구성』
- 408 페이지의 『lbcontrol server -- 서버 구성』
- 411 페이지의 『lbcontrol set -- 서버 로그 구성』
- 412 페이지의 『lbcontrol status -- 관리 프로그램 및 권고자의 실행 여부 표시』

ndcontrol 명령 매개변수의 최소 버전을 입력할 수 있습니다. 매개변수의 고유한 문자만 입력해야 합니다. 예를 들어, 파일 저장 명령에 대한 도움말을 보려면, **lbcontrol help file** 대신에 **lbcontrol he f**를 입력할 수 있습니다.

"lbc" 접두부는 로드 밸런스 컨설턴트를 의미합니다.

주: 명령 매개변수 값은 영문자로 입력해야 합니다. 호스트 이름(클러스터 및 서버 명령에 사용)과 파일 이름(파일 명령에 사용)만 예외입니다.

lbcontrol advisor -- 권고자 제어



connecttimeout

서버와의 연결 실패를 보고하기 전에 권고자가 대기하는 시간을 설정하십시오. 자세한 내용은 160 페이지의 『권고자 연결 시간 종료 및 서버의 수신 시간 종료』를 참조하십시오.

name

권고자의 이름. 가능한 값은 **http**, **ftp**, **ssl**, **smtp**, **imap**, **pop3**, **nntp**, **telnet**, **connect**, **ping** 및 **WTE**입니다. 사용자 정의 권고자 이름은 xxxx 형식이며, 여기서 ADV_xxxx는 조정 권고자를 구현하는 클래스의 이름입니다.

port

권고자가 모니터하는 포트의 번호.

timeoutseconds

서버와의 연결 실패를 보고하기 전에 권고자가 대기하는 시간을 초 단위로 표시하는 양의 정수. 기본값은 권고자 간격에 지정된 값의 3배입니다.

interval

권고자가 서버에 정보를 조회하는 간격을 설정합니다.

seconds

서버의 현재 상태를 서버에 요청하는 간격(초 단위)을 나타내는 양의 정수. 기본값은 15입니다.

list

현재 관리 프로그램에 정보를 제공하는 권고자 목록을 보여줍니다.

loglevel

권고자 로그에 대한 로그 레벨을 설정합니다.

level

레벨 번호(0-5). 기본값은 1입니다. 번호가 높아질수록 더 자세한 정보가 권고자 로그에 기록됩니다. 가능한 값은 다음과 같습니다. 0은 없음, 1은 최소, 2는 기본, 3은 중간, 4는 고급, 5는 자세합니다.

logsize

권고자 로그의 최대 크기를 설정합니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 후속 항목은 파일의 맨 위에서부터 이전 로그 항목에 겹쳐 기록됩니다. 로그 크기는 로그의 현재 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.

number of records

권고자 로그 파일의 최대 크기(바이트 단위). 0보다 큰 양의 정수이거나 단어 **unlimited**를 지정할 수 있습니다. 로그 항목 크기가 변하므로 겹쳐쓰기 전에는 로그 파일은 정확한 최대 크기에 도달할 수 없습니다. 기본값은 1MB입니다.

receivetimeout

서버에서 수신 실패를 보고하기 전에 권고자가 대기하는 시간을 설정하십시오. 자세한 내용은 160 페이지의 『권고자 연결 시간 종료 및 서버의 수신 시간 종료』를 참조하십시오.

timeoutseconds

서버에서 수신 실패를 보고하기 전에 권고자가 대기하는 시간을 초 단위로 표시하는 양의 정수. 기본값은 권고자 간격에 지정된 값의 3배입니다.

report

권고자 상태에 대한 보고서를 표시합니다.

start

권고자를 시작합니다. 각 프로토콜에 대해 권고자가 있습니다. 기본 포트는 다음과 같습니다.

권고자 이름	프로토콜	포트
connect	ICMP	12345
db2	개인용	50000
ftp	FTP	21
http	HTTP	80
ibmproxy	HTTP(Caching Proxy를 통한)	80
imap	IMAP	143
nntp	NNTP	119
ping	PING	0
pop3	POP3	110
smtp	SMTP	25
ssl	SSL	443
telnet	Telnet	23
WLM	개인용	10007

주: FTP 권고자는 FTP 제어 포트(21)에만 권고합니다. FTP 데이터 포트(20)에서는 FTP 권고자를 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, 232 페이지의 『로그 파일 경로 변경』을 참조하십시오.

권고자 정보가 사용되는지 확인하려면 관리 프로그램 비율을 설정하십시오.

status

모든 권고자 전역 값의 현재 상태 및 기본값을 표시합니다.

stop

권고자를 정지합니다.

timeout

관리 프로그램이 권고자의 정보를 유효한 것으로 간주하는 시간(초 단위)을 설정합니다. 권고자 정보가 이 시간 기간보다 이전의 정보라는 것을 관리 프로그램이 발견하면, 관리 프로그램은 권고자가 모니터링하는 포트에서 서버에 대한 가중치를 판별할 때 이 정보를 사용하지 않습니다. 시간 종료의 예외는 권고자가 관리 프로그램에 특정 서버가 작동 중지 상태임을 통지하는 경우입니다. 관리 프로그램은 권고자 정보가 시간 종료된 후에도 서버에 대한 해당 정보를 사용합니다.

seconds

시간(초)을 나타내는 양수 또는 단어 **unlimited**. 기본값은 unlimited입니다.

version

권고자의 현재 버전을 표시합니다.

예제

- 서버와의 연결 실패를 보고하기 전에 HTTP 권고자가(포트 80의 경우) 대기하는 시간(30초)을 설정하려면 다음 명령을 실행하십시오.

```
lbccontrol advisor connecttimeout http 80 30
```

- FTP 권고자 간격(포트 21의 경우)을 6초로 설정하려면 다음 명령을 실행하십시오.

```
lbccontrol advisor interval ftp 21 6
```

- 현재 관리 프로그램에 정보를 제공하는 권고자 목록을 표시하려면 다음 명령을 실행하십시오.

```
lbccontrol advisor list
```

이 명령으로 다음과 같은 출력이 작성됩니다.

ADVISOR	PORT	TIMEOUT
http	80	unlimited
ftp	21	unlimited

- 더 나은 성능을 위해 권고자 로그의 로그 레벨을 0으로 설정하려면 다음 명령을 실행하십시오.

```
lbccontrol advisor loglevel http 80 0
```

- 권고자 로그 크기를 5000 바이트로 변경하려면 다음 명령을 실행하십시오.

```
lbccontrol advisor logsize ftp 21 5000
```

- 서버에서 수신 실패를 보고하기 전에 HTTP 권고자가(포트 80의 경우) 대기하는 시간(60초)을 설정하려면 다음 명령을 실행하십시오.

```
lbccontrol advisor receivetimeout http 80 60
```

- FTP 권고자 상태에 대한 보고서를 표시하려면(포트 21의 경우) 다음 명령을 실행하십시오.

```
lbccontrol 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
```

- ftpadv.log 파일을 사용하여 권고자를 시작하려면 다음 명령을 실행하십시오.

```
lbccontrol advisor start ftp 21 ftpadv.log
```

- http 권고자와 연관된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

```
lbccontrol advisor status http 80
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Advisor Status:

Interval (seconds) 15
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

- 포트 80에서 http 권고자를 정지하려면 다음 명령을 실행하십시오.

```
lbcontrol advisor stop http 80
```

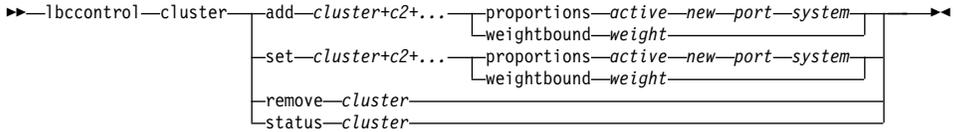
- 권고자 정보에 대한 시간 종료 값을 5초로 설정하려면 다음 명령을 실행하십시오.

```
lbcontrol advisor timeout ftp 21 5
```

- ssl 권고자의 현재 버전 번호를 찾으려면 다음 명령을 실행하십시오.

```
lbcontrol advisor version ssl 443
```

lbcontrol cluster -- 클러스터 구성



add

이 클러스터를 추가합니다. 최소한 하나의 클러스터를 정의해야 합니다.

weightbound

이 포트에서 서버의 최대 가중치를 설정합니다. 이것은 Cisco CSS Switch가 각 서버에 제공할 요청 수 사이의 차이에 영향을 줍니다. 기본값은 10입니다.

weight

weightbound의 값입니다.

set

클러스터의 특성을 설정합니다.

proportions

활성 연결(*active*), 새 연결(*new*), 권고자의 정보(*port*) 및 서버 가중치를 설정하기 위해 관리 프로그램이 사용하는 Metric Server의 정보(*system*)에 대한 중요도를 설정하십시오. 아래에 설명되어 있는 이러한 값 각각은 총계의 백분율로 표시되므로, 합은 항상 100입니다. 자세한 정보는 152 페이지의 『상태 정보에 제공되는 중요성 비율』을 참조하십시오.

active

활성 연결에 부여될 가중치를 표시하는 0에서 100 사이의 숫자. 기본값은 50입니다.

new

새 연결에 부여될 가중치를 표시하는 0에서 100 사이의 숫자. 기본값은 50입니다.

port

권고자의 정보에 부여될 가중치를 표시하는 0에서 100 사이의 숫자. 기본값은 0입니다.

system

시스템 메트릭 정보에 부여될 가중치를 표시하는 0에서 100 사이의 숫자. 기본값은 0입니다.

remove

이 클러스터를 제거합니다.

status

특정 클러스터의 현재 상태를 보여줍니다.

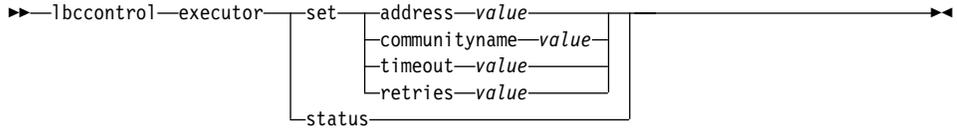
예제

- 클러스터 주소 130.40.52.153을 추가하려면 다음 명령을 실행하십시오.
lbcontrol cluster add 130.40.52.153
- 클러스터 주소 130.40.52.153을 제거하려면 다음 명령을 실행하십시오.
lbcontrol cluster remove 130.40.52.153
- 관리 프로그램에서 수신한 입력에 있는 상대적 중요성을 설정하려면 다음 명령을 실행하십시오.
lbcontrol cluster proportions 60 35 5 0
- 클러스터 주소 9.67.131.167의 상태를 표시하려면 다음 명령을 실행하십시오.
lbcontrol cluster status 9.67.131.167

이 명령으로 다음과 같은 출력이 작성됩니다.

```
Cluster Status:
-----
Address ..... 9.67.131.167
Number of target ports ..... 3
Default port weight bound ..... 10
Proportion given to active connections .. 49
Proportion given to new connections ..... 49
Proportion given specific to the port ... 2
Proportion given to system metrics ..... 0
```

lbcontrol executor -- 실행 프로그램 제어



set

실행 프로그램의 필드를 설정합니다.

address

관리 목적으로 Cisco CSS Switch에 연결하는 데 사용되는 호스트 이름의 IP 주소. 자세한 정보는 *Cisco Content Services Switch 기본 구성 안내서*를 참조하십시오.

value

유효한 IP 주소 또는 호스트 이름.

communityname

Cisco CSS Switch와의 SNMP 통신에서 사용되는 SNMP 공동체 이름. 자세한 정보는 *Cisco Content Services Switch 기본 구성 안내서*를 참조하십시오.

value

기본값은 읽기/쓰기 액세스를 허용하는 공용입니다.

timeout

Cisco Consultant에서 Cisco CSS Switch까지 SNMP 조회가 시간 종료된 후의 시간(초 단위). Cisco Consultant는 SNMP를 사용하여 Cisco CSS Switch에서 정보를 수집합니다. manager.log 메시지가 빈번한 시간 종료를 표시하는 경우, 이 값을 조정할 수 있습니다.

value

기본값은 3입니다.

retries

Cisco Consultant가 Cisco CSS Switch에 발행한 SNMP 조회 재시도 횟수. manager.log 메시지가 빈번한 SNMP 조회 장애를 표시하는 경우, 이 값을 조정할 수 있습니다.

value

기본값은 2입니다.

status

실행 프로그램에서 설정할 수 있는 값의 현재 상태와 기본값을 표시합니다.

예제

- Cisco Consultant의 내부 카운터를 표시하려면 다음 명령을 실행하십시오.

```
lbcontrol executor status
```

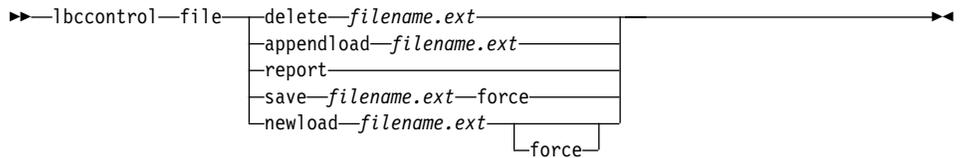
```
Executor Status:
```

```
-----  
address ..... 9.67.131.151  
community name ..... public  
timeout value ..... 3  
retires value ..... 2
```

- 주소를 130.40.52.167로 설정하려면 다음 명령을 실행하십시오.

```
lbcontrol executor set address 130.40.52.167
```

lbccontrol file -- 구성 파일 관리



delete

파일을 삭제합니다.

filename.ext

구성 파일.

파일 확장자(.ext)는 사용자가 원하는 대로 정할 수 있으며 선택적입니다.

appendload

현재 구성에 구성 파일을 추가하고 Cisco Consultant로 로드합니다.

report

사용 가능한 파일에 대해 보고합니다.

save

Cisco Consultant의 현재 구성 파일을 파일에 저장합니다.

주: 파일은 다음 디렉토리에 저장되고 여기서 로드됩니다.

- AIX: /usr/lpp/nd/servers/configurations/lbc
- Linux: /opt/nd/servers/configurations/lbc
- Solaris: /opt/nd/servers/configurations/lbc
- Windows 2000:

일반 설치 디렉토리 경로 -- c:\Program

Files\ibm\edge\nd\servers\configurations\ 구성요소

고유(Native) 설치 디렉토리 경로 -- c:\Program

Files\ibm\nd\servers\configurations\component

force

파일을 기존 파일과 동일한 이름으로 저장하려면 **force**를 사용하여 새 파일을 저장하기 전에 기존 파일을 삭제하십시오. force 옵션을 사용하지 않으면 기존 파일을 겹쳐쓰지 않습니다.

newload

Cisco Consultant로 새 구성 파일을 로드합니다. 새 구성 파일은 현재 구성을 대신합니다.

예제

- 파일을 삭제하려면 다음 명령을 실행하십시오.

```
lbccontrol file delete file3
```

```
File (file3) was deleted.
```

- 새 구성 파일을 로드하여 현재 구성을 대체하려면 다음 명령을 실행하십시오.

```
lbccontrol file newload file1.sv
```

```
File (file1.sv) was loaded into the Dispatcher.
```

- 현재 구성에 구성 파일을 추가하고 로드하려면 다음 명령을 실행하십시오.

```
lbccontrol 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.

```
lbccontrol file report
```

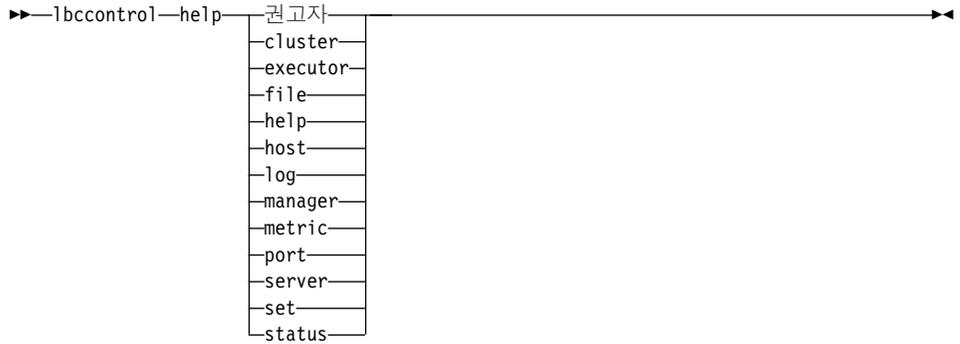
```
FILE REPORT:  
file1.save  
file2.sv  
file3
```

- 구성을 file3 파일에 저장하려면 다음 명령을 실행하십시오.

```
lbccontrol file save file3
```

```
The configuration was saved into file (file3).
```

lbccontrol help -- 이 명령의 도움말 표시 또는 인쇄



예제

- lbccontrol 명령에 대한 도움말을 보려면 다음 명령을 실행하십시오.

```
lbccontrol help
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
HELP COMMAND ARGUMENTS:
```

```
-----
```

```
Usage: help <help option>
```

```
Example: help cluster
```

```
executor      - help on executor command
cluster       - help on cluster command
port          - help on port command
server        - help on server command
manager       - help on manager command
metric        - help on metric command
advisor       - help on advisor command
file          - help on file command
host          - help on host command
log           - help on log command
set           - help on set command
status        - help on status command
help          - print complete help text
```

< > 내의 매개변수는 변수입니다.

lbcontrol host -- 원격 시스템 구성

▶▶—lbcontrol—host:—remote_host—▶▶

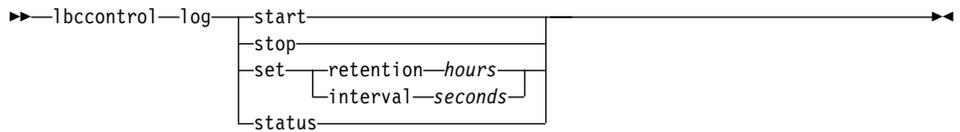
remote_host

구성 중인 원격 Cisco Consultant 시스템의 이름. 이 명령을 입력할 때는 다음과 같이 **host:**와 *remote_host* 사이에 공간이 없어야 합니다.

```
lbcontrol host:remote_host
```

명령 프롬프트에서 이 명령을 실행한 다음, 원격 Cisco Consultant 시스템에 발행할 유효한 lbcontrol 명령을 입력하십시오.

lbccontrol log -- 2진 로그 파일 제어



start

2진 로그를 시작합니다.

stop

2진 로그를 정지합니다.

set

2진 로그 필드를 설정합니다. 2진 로그 필드 설정에 대한 자세한 정보는 223 페이지의 『서버 통제를 분석하기 위해 2진 로그 사용』을 참조하십시오.

retention

2진 로그 파일이 보존되는 시간 수. **retention**의 기본값은 24입니다.

hours

시간.

interval

로그 입력 항목 간의 초 시간. 간격의 기본값은 60입니다.

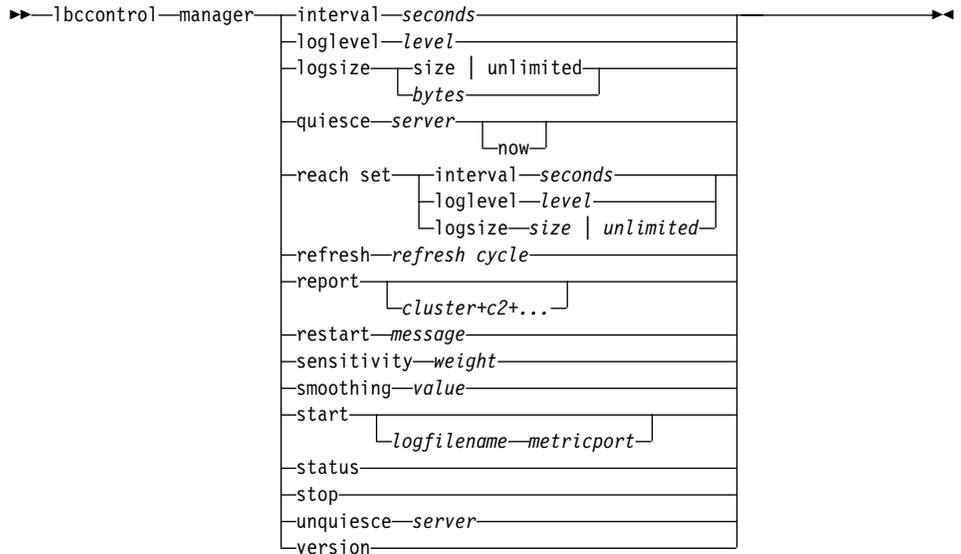
seconds

초 시간.

status

2진 로그의 보유 및 간격을 표시합니다.

lbcontrol manager -- 관리 프로그램 제어



interval

관리 프로그램이 Cisco CSS Switch에 대한 서버 가중치를 갱신하는 빈도를 설정하여 Cisco CSS Switch가 클라이언트 요청을 라우트하는 데 사용하는 기준을 갱신합니다.

seconds

관리 프로그램이 Cisco CSS Switch에 대한 가중치를 갱신하는 빈도를 초 단위로 표시하는 양수. 기본값은 15이며, 최소 간격은 10입니다. 관리 프로그램 간격을 10초 미만으로 설정할 경우, 간격은 10초로 설정됩니다. Cisco CSS Switch가 고속 갱신을 이용하지 않으므로 기본 관리 프로그램 간격을 15초로 설정하는 것이 좋습니다.

loglevel

관리 프로그램 로그에 대한 로그 레벨을 설정합니다.

level

레벨 번호(0-5). 번호가 커질수록 더 자세한 정보가 관리 프로그램 로그에 기록됩니다. 기본값은 1입니다. 가능한 값은 다음과 같습니다. 0은 없음, 1은 최소, 2는 기본, 3은 중간, 4는 고급, 5는 자세합니다.

logsize

관리 프로그램 로그의 최대 크기를 설정합니다. 로그 파일에 최대 크기를 설정할 경우, 파일은 랩됩니다. 파일이 지정된 크기에 도달하면 후속 항목은 파일의 맨 위에서부터 이전 로그 항목에 겹쳐 기록됩니다. 로그 크기는 로그의 현재 크기보다 더 작게 설정할 수 없습니다. 로그 항목에는 기록된 순서를 표시하도록 시간 소인이 표시됩니다. 로그 레벨을 높게 설정할수록, 상위 레벨에서 로그될 때 공간이 더 빨리 소모될 수 있으므로 더 주의하여 로그 크기를 선택해야 합니다.

bytes

관리 프로그램 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.

quiesce

더이상 연결을 서버로 전송하지 않도록 지정합니다. 관리 프로그램은 프로그램이 정의된 모든 포트에서 해당 서버의 기중치를 0으로 설정한 후 Cisco CSS Switch로 일시중단 명령을 전송합니다. 신속한 유지보수를 위해 서버 작업을 중지할 경우 이 명령을 사용한 후 다시 서버를 활성화하십시오. 구성에서 일시중단된 서버를 삭제한 후 다시 추가한 경우, 일시중단 이전의 상태를 유지하지 않습니다.

server

기호 이름 또는 점분리 10진수 형식으로 된 서버의 IP 주소.

reach

도달 권고자에 대한 간격, 로그 레벨 및 로그 크기를 설정합니다.

refresh

새 연결과 활성 연결에 대한 정보를 새로 고치기 위해 Cisco CSS Switch를 조회하기 전의 간격 수를 설정하십시오.

refresh cycle

간격 수를 나타내는 양수. 기본값은 1입니다.

report

통계 스냅샷 보고서를 표시합니다.

cluster

보고서에 표시하려는 클러스터의 주소. 이 주소는 기호 이름이나 점분리 10진수 형식일 수 있습니다. 기본값은 모든 클러스터에 대한 관리 프로그램 보고서 화면입니다.

주: 추가 클러스터는 더하기 부호(+)로 구분됩니다.

restart

모든 서버(단절되지 않은)를 재시작하여 가중치를 표준화합니다(최대 가중치의 1/2).

message

관리 프로그램 로그 파일에 기록할 메시지.

sensitivity

가중치가 갱신되는 최소 감도를 설정합니다. 이 설정은 외부 정보를 기반으로 관리 프로그램이 서버에 대한 가중치를 변경해야 할 시기를 정의합니다.

weight

가중치 백분율로 사용될 0-100 사이의 숫자. 기본값 5를 사용하면 최소 감도는 5%가 됩니다.

smoothing

로드 밸런스 시 가중치의 변화를 평탄화하는 지수를 설정합니다. 스무스 색인(smoothing index)을 높게 설정하면 네트워크 조건이 변함에 따라 서버 가중치가 상대적으로 낮게 변경됩니다. 색인이 낮으면 서버 가중치가 상대적으로 높게 변경됩니다.

value

양의 부동 소수점 수. 기본값은 1.5입니다.

start

관리 프로그램을 시작합니다.

logfile

관리 프로그램 데이터가 기록되는 파일 이름. 로그의 각 레코드에는 시간 소인이 표시됩니다.

기본 파일은 **logs** 디렉토리에 설치됩니다. 413 페이지의 『부록F. 예제 구성 파일』을 참조하십시오. 로그 파일이 보존되는 디렉토리 변경에 대한 정보는 232 페이지의 『로그 파일 경로 변경』을 참조하십시오.

metricport

Metric Server가 통신할 포트. 측정 기준 포트를 지정할 경우, 로그 파일 이름을 지정해야 합니다. 기본 측정 기준 포트는 10004입니다.

status

모든 관리 프로그램의 전역 값의 현재 상태 및 기본값을 표시합니다.

stop

관리 프로그램을 정지합니다.

unquiesce

관리 프로그램이 정의된 모든 포트에서 이전에 작업중지된 서버에 0보다 높은 가중치를 부여할 수 있도록 지정하십시오. 관리 프로그램은 Cisco CSS Switch로 활성 명령을 전송합니다.

server

기호 이름이나 점분리 10진수 형식으로 된 서버의 IP 주소.

version

관리 프로그램 현재 버전을 표시합니다.

예제

- 관리 프로그램의 갱신 간격을 매 5초로 설정하려면 다음 명령을 실행하십시오.
`lbcontrol manager interval 5`
- 더 나은 성능을 위해 로그 레벨을 0으로 설정하려면 다음 명령을 실행하십시오.
`lbcontrol manager loglevel 0`
- 관리 프로그램 로그 크기를 1,000,000 바이트를 설정하려면 다음 명령을 실행하십시오.
`lbcontrol manager logsize 1000000`

- 130.40.52.153에서 서버에 더 이상 연결을 전송하지 않도록 지정하려면 다음 명령을 실행하십시오.

```
lbcontrol manager quiesce 130.40.52.153
```

- 가중치가 갱신되기 전의 갱신 간격 수를 3으로 설정하려면 다음 명령을 실행하십시오.

```
lbcontrol manager refresh 3
```

- 관리 프로그램의 통계 스냅샷을 확보하려면 다음 명령을 실행하십시오.

```
lbcontrol manager report
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
lbcontrol>>manager report
```

HOST TABLE LIST	STATUS
server6	ACTIVE
server5	ACTIVE
server4	ACTIVE
server3	ACTIVE
server2	ACTIVE
server1	ACTIVE

9.67.154.35	WEIGHT	ACTIVE %	49	NEW %	50	PORT %	1	SYSTEM %	0		
PORT:	80	NOW	NEW	WT	CONNECT	WT	CONNECT	WT	LOAD	WT	LOAD
server1	4	4	5	0	5	0	3	301	-9999	-1	
server2	5	5	5	0	5	0	6	160	-9999	-1	
PORT TOTALS:	9	9		0		0		461		-2	

9.67.154.35	WEIGHT	ACTIVE %	49	NEW %	50	PORT %	1	SYSTEM %	0		
PORT:	443	NOW	NEW	WT	CONNECT	WT	CONNECT	WT	LOAD	WT	LOAD
server3	4	4	5	0	5	0		0	-9999	-1	
server4	5	5	5	0	5	0	0	0	-9999	-1	
PORT TOTALS:	9	9		0		0		0		-2	

9.67.154.34	WEIGHT	ACTIVE %	49	NEW %	50	PORT %	1	SYSTEM %	0		
PORT:	80	NOW	NEW	WT	CONNECT	WT	CONNECT	WT	LOAD	WT	LOAD
server5	5	5	5	0	5	0	5	160	-9999	-1	
server6	0	0	5	0	5	0	-9999	-1	-9999	-1	
PORT TOTALS:	5	5		0		0		159		-2	

ADVISOR	PORT	TIMEOUT
http	80	unlimited

- 표준화된 가중치에 대해 모든 서버를 재시작하고 메시지를 관리 프로그램 로그 파일에 기록하려면 다음 명령을 실행하십시오.

```
lbcontrol 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.

```
lbcontrol manager sensitivity 10
```

- 스무스 색인을 2.0으로 설정하려면 다음 명령을 실행하십시오.

lbcontrol manager smoothing 2.0

- 관리 프로그램을 시작하고 ndmgr.log(경로는 설정할 수 없음) 로그 파일을 지정하려면 다음 명령을 실행하십시오.

```
lbcontrol manager start ndmgr.log
```

- 관리 프로그램과 관련된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

```
lbcontrol 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 ..... 1
Reach log level ..... 1
Maximum reach log size (bytes) ..... unlimited
Reach update interval (seconds) ..... 7
```

- 관리 프로그램을 중지하려면 다음 명령을 실행하십시오.

```
lbcontrol manager stop
```

- 관리 프로그램의 현재 버전 번호를 표시하려면 다음 명령을 실행하십시오.

```
lbcontrol manager version
```

lbcontrol metric -- 시스템 메트릭 구성

```
lbcontrol metric add cluster+c2+...+cN:metric+metric1+...+metricN
lbcontrol metric remove cluster+c2+...+cN:metric+metric1+...+metricN
lbcontrol metric proportions cluster+c2+...+cN:proportion1 prop2 prop3...propN
lbcontrol metric status cluster+c2+...+cN:metric+metric1+...+metricN
```

add

메트릭을 추가합니다.

cluster

클라이언트가 연결되는 주소. 주소는 시스템의 호스트 이름 또는 점분리 10진 수 IP 주소입니다. 추가 클러스터는 더하기 부호(+)로 구분됩니다.

주: Cisco Consultant의 경우, 클러스터 주소는 Cisco CSS Switch 구성에서 소유자 콘텐츠 규칙의 VIP(가상 IP) 주소에 해당합니다.

metric

시스템 메트릭. 메트릭 선택사항은 다음과 같습니다.

- cpuload
- memload
- port
- system metrics

remove

메트릭을 제거합니다.

proportions

이 오브젝트와 연관된 메트릭의 비율을 설정합니다.

status

이 메트릭의 현재 값을 표시합니다.

예제

- 시스템 메트릭을 추가하려면 다음 명령을 실행하십시오.

```
lbcontrol metric add 10.10.10.20:metric1
```

- 두 개의 시스템 메트릭을 사용하여 클러스터에 대한 비율을 설정하려면 다음 명령을 실행하십시오.

```
lbcontrol metric proportions 10.10.10.20 48 52
```

- 지정된 메트릭과 연관된 값의 현재 상태를 표시하려면 다음 명령을 실행하십시오.

```
lbcontrol metric status 10.10.10.20:metric1
```

이 명령으로 다음과 같은 출력이 작성됩니다.

```
Metric Status:
```

```
-----
```

```
Cluster ..... 10.10.10.20
```

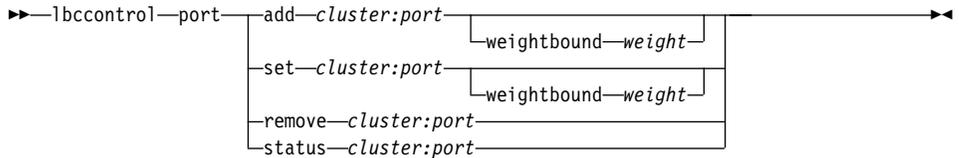
```
Metric name ..... metric1
```

```
Metric proportion ..... 52
```

```
Server ..... 9.37.56.100
```

```
Metric data .... -1
```

lbcontrol port -- 포트 구성



add

클러스터에 포트를 추가합니다. 서버를 해당 포트에 추가하기 전에 클러스터에 포트를 추가해야 합니다. 클러스터 포트가 없는 경우, 모든 클라이언트 요청은 로컬로 처리됩니다. 이 명령을 사용하면 한 번에 둘 이상의 포트를 추가할 수 있습니다.

weightbound

이 포트에서 서버의 최대 가중치를 설정합니다. 이것은 Cisco CSS Switch가 각 서버에 제공할 요청 수 사이의 차이에 영향을 줍니다. 기본값은 10입니다.

weight

최대 가중 바운드를 표시하는 1에서 10 사이의 숫자.

set

포트의 필드를 설정합니다.

remove

해당 포트를 제거합니다.

status

해당 포트의 서버 상태를 보여줍니다. 모든 포트의 상태를 보려면, 이 명령에 *port*를 지정하지 마십시오. 그러나 콜론은 잊지 마십시오.

예제

- 포트 80과 23을 클러스터 주소 130.40.52.153에 추가하려면 다음 명령을 실행하십시오.

```
lbcontrol port add 130.40.52.153:80+23
```
- 클러스터 주소 130.40.52.153에 있는 포트 80에 최대 가중치 10을 설정하려면 다음 명령을 실행하십시오.

```
lbcontrol port set 130.40.52.153:80 weightbound 10
```

- 클러스터 주소 130.40.52.153에서 포트 23을 제거하려면 다음 명령을 실행하십시오.

```
lbcontrol port remove 130.40.52.153:23
```

- 클러스터 주소 9.67.131.153에서 포트 80의 상태를 확보하려면 다음 명령을 실행하십시오.

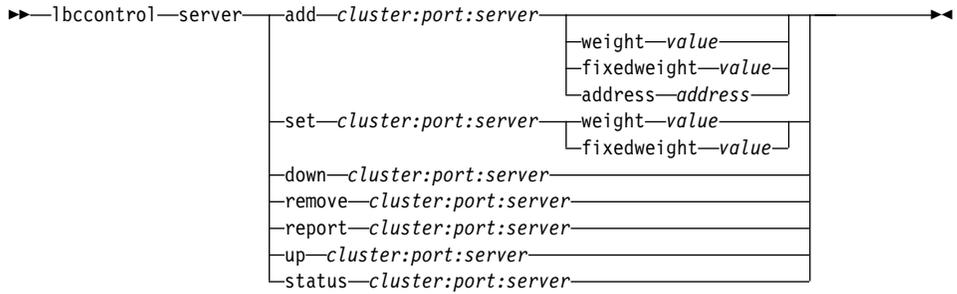
```
lbcontrol port status 9.67.131.153:80
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Port Status:

```
Port number ..... 80
Cluster address ..... 9.67.131.153
Number of servers ..... 2
Weight bound ..... 10
```

lbcontrol server -- 서버 구성



add

이 서버를 추가합니다.

cluster

기호 이름 또는 점분리 10진수 형식의 클러스터 주소.

주: 추가 클러스터는 더하기 부호(+)로 구분됩니다.

port

포트의 번호.

주: 추가 포트는 더하기 부호(+)로 구분됩니다.

server

기호 이름 또는 점분리 10진수 형식의 TCP 서버 시스템의 고유 IP 주소. IP 주소로 분석되지 않는 고유 이름을 사용하는 경우, **lbcontrol server add** 명령에 주소 속성을 제공해야 합니다.

weight

이 서버의 가중치를 표시하는 0에서 10 사이의 숫자. 가중치를 0으로 설정하면 새로운 요청이 서버로 전송되지 않지만, 해당 서버에 대해 현재 활성 연결은 종료되지 않습니다. 기본값은 지정된 포트의 최대 가중치를 2로 나눈 값입니다. 관리 프로그램이 실행 중이고 `fixedweight`가 `no`로 설정되어 있으면 이 설정을 빨리 겹쳐줍니다.

value

가중치 값.

fixedweight

fixedweight 옵션을 사용하면 관리 프로그램이 서버 가중치를 수정할 수 있는지 여부를 지정할 수 있습니다. fixedweight 값을 yes로 설정하면 관리 프로그램이 실행될 때 서버 가중치를 수정할 수 없습니다. 자세한 내용은 155 페이지의 『관리 프로그램 고정 가중치』를 참조하십시오.

value

고정 가중치 값. 기본값은 no입니다.

address

기호 이름 또는 점분리 10진수 형식의 TCP 서버 시스템의 고유 IP 주소. 서버 이름 값을 해석할 수 없는 경우(예: 논리 서버 이름), 물리적 서버 시스템의 주소를 제공해야 합니다.

value

서버 시스템의 고유 ID. 서버를 분석할 수 없는 경우, 주소 속성을 제공해야 합니다.

down

이 서버가 단절되었음을 표시합니다. Cisco CSS Switch는 이 서버에 대한 연결 전송을 정지시킵니다.

remove

해당 서버를 제거합니다.

report

해당 서버에 대해 보고합니다.

set

해당 서버의 값을 설정합니다.

up 이 서버를 연결할 것을 표시합니다. Cisco CSS Switch는 이제 이 서버로 새 연결을 전송합니다.

status

서버의 상태를 표시합니다.

예제

- 클러스터 주소 130.40.52.153에 있는 포트 80에 27.65.89.42에 있는 서버를 추가하려면 다음 명령을 실행하십시오.

```
lbcontrol server add 130.40.52.153:80:27.65.89.42
```

- 모든 클러스터에 있는 모든 포트에서 27.65.89.42에 있는 서버를 제거하려면 다음 명령을 실행하십시오.

```
lbcontrol server remove ::27.65.89.42
```

- 클러스터 주소 130.40.52.153에 있는 포트 80에서 서버 27.65.89.42의 가중치를 10으로 설정하려면 다음 명령을 실행하십시오.

```
lbcontrol server set 130.40.52.153:80:27.65.89.42 weight 10
```

lbccontrol status -- 관리 프로그램 및 권고자의 실행 여부 표시

▶—lbccontrol—status—▶

예제

- 수행되고 있는 내용을 보려면 다음 명령을 실행하십시오.

```
lbccontrol status
```

이 명령으로 다음과 같은 출력이 작성됩니다.

Manager has been started.

ADVISOR	PORT	TIMEOUT
http	80	unlimited
ftp	21	unlimited

부록F. 예제 구성 파일

이 부록에는 Network Dispatcher의 Dispatcher 구성요소에 대한 예제 구성 파일이 들어 있습니다.

예제 Network Dispatcher 구성 파일

예제 파일은 .../nd/servers/samples/ 디렉토리에 있습니다.

Dispatcher 구성 파일--AIX, Red Hat Linux, 및 Solaris

```
#!/bin/ksh
#
# 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
#
# ndserver start
# sleep 5

#
# Then start the executor
#
# ndcontrol executor start

#
# The Dispatcher can be removed at any time using the
# "ndcontrol executor stop" and "ndserver 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"
# ndcontrol 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 "
# ndcontrol 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"

# ndcontrol 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"
# ndcontrol 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
# ndcontrol 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..."
# ndcontrol manager start

# echo "Starting the FTP advisor on port 21 ..."
# ndcontrol advisor start ftp 21
# echo "Starting the HTTP advisor on port 80 ..."
# ndcontrol advisor start http 80
# echo "Starting the Telnet advisor on port 23 ..."
#ndcontrol advisor start telnet 23
# echo "Starting the SMTP advisor on port 25 ..."
# ndcontrol advisor start smtp 25
# echo "Starting the POP3 advisor on port 110 ..."
# ndcontrol advisor start pop3 110
# echo "Starting the NNTP advisor on port 119 ..."
# ndcontrol advisor start nntp 119
# echo "Starting the SSL advisor on port 443 ..."
# ndcontrol advisor start ssl 443
#

# echo "Setting the manager proportions..."
# ndcontrol 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.
# ndcontrol cluster 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.
# ndcontrol cluster 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.
# ndcontrol manager loglevel 1
# ndcontrol manager logsize 1048576
# ndcontrol manager sensitivity 5.000000
# ndcontrol manager interval 2
# ndcontrol manager refresh 2

```

```

#
# ndcontrol advisor interval ftp 21 5
# ndcontrol advisor loglevel ftp 21 1
# ndcontrol advisor logsize ftp 21 1048576
# ndcontrol advisor timeout ftp 21 unlimited
# ndcontrol advisor interval telnet 23 5
# ndcontrol advisor loglevel telnet 23 1
# ndcontrol advisor logsize telnet 23 1048576
# ndcontrol advisor timeout telnet 23 unlimited
# ndcontrol advisor interval smtp 25 5
# ndcontrol advisor loglevel smtp 25 1
# ndcontrol advisor logsize smtp 25 1048576
# ndcontrol advisor timeout smtp 25 unlimited
# ndcontrol advisor interval http 80 5
# ndcontrol advisor loglevel http 80 1
# ndcontrol advisor logsize http 80 1048576
# ndcontrol advisor timeout http 80 unlimited
# ndcontrol advisor interval pop3 110 5
# ndcontrol advisor loglevel pop3 110 1
# ndcontrol advisor logsize pop3 110 1048576
# ndcontrol advisor timeout pop3 110 unlimited
# ndcontrol advisor interval nntp 119 5
# ndcontrol advisor loglevel nntp 119 1
# ndcontrol advisor logsize nntp 119 1048576
# ndcontrol advisor timeout nntp 119 unlimited
# ndcontrol advisor interval ssl 443 5
# ndcontrol advisor loglevel ssl 443 1
# ndcontrol advisor logsize ssl 443 1048576
# ndcontrol advisor timeout ssl 443 unlimited
#

```

Dispatcher 구성 파일--Windows

다음은 Window에서 사용할 **configuration.cmd.sample**이라는 예제 Network Dispatcher 구성 파일입니다.

```

@echo off
rem configuration.cmd.sample - Sample configuration file for the
rem Dispatcher component.
rem

rem ndserver must be started via Services

rem

rem
rem Then start the executor
rem

```

```

rem call ndcontrol 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 ndcontrol 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 ndcontrol executor set fintimeout 30
rem call ndcontrol executor set fincount 4000
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 ndcontrol 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 ndcontrol 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 ndcontrol 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 via
rem ndcontrol 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 ndcontrol manager start

rem echo "Starting the FTP advisor on port 21 ..."
rem call ndcontrol advisor start ftp 21
rem echo "Starting the HTTP advisor on port 80 ..."
rem call ndcontrol advisor start http 80
rem echo "Starting the Telnet advisor on port 23 ..."
rem call ndcontrol advisor start telnet 23
rem echo "Starting the SMTP advisor on port 25 ..."
rem call ndcontrol advisor start smtp 25
rem echo "Starting the POP3 advisor on port 110 ..."
rem call ndcontrol advisor start pop3 110
rem echo "Starting the NNTP advisor on port 119 ..."
rem call ndcontrol advisor start nntp 119
rem echo "Starting the SSL advisor on port 443 ..."
rem call ndcontrol advisor start ssl 443
rem

rem echo "Setting the cluster proportions..."
rem call ndcontrol 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 ndcontrol cluster 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 ndcontrol cluster 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 ndcontrol manager loglevel 1
rem call ndcontrol manager logsize 1048576
rem call ndcontrol manager sensitivity 5.000000
rem call ndcontrol manager interval 2
rem call ndcontrol manager refresh 2
rem
rem call ndcontrol advisor interval ftp 21 5
rem call ndcontrol advisor loglevel ftp 21 1
rem call ndcontrol advisor logsize ftp 21 1048576
rem call ndcontrol advisor timeout ftp 21 unlimited
rem call ndcontrol advisor interval telnet 23 5
rem call ndcontrol advisor loglevel telnet 23 1
rem call ndcontrol advisor logsize telnet 23 1048576
rem call ndcontrol advisor timeout telnet 23 unlimited
rem call ndcontrol advisor interval smtp 25 5
rem call ndcontrol advisor loglevel smtp 25 1
rem call ndcontrol advisor logsize smtp 25 1048576
rem call ndcontrol advisor timeout smtp 25 unlimited
rem call ndcontrol advisor interval http 80 5
rem call ndcontrol advisor loglevel http 80 1
rem call ndcontrol advisor logsize http 80 1048576
rem call ndcontrol advisor timeout http 80 unlimited
rem call ndcontrol advisor interval pop3 110 5
rem call ndcontrol advisor loglevel pop3 110 1
rem call ndcontrol advisor logsize pop3 110 1048576
rem call ndcontrol advisor timeout pop3 110 unlimited
rem call ndcontrol advisor interval nntp 119 5
rem call ndcontrol advisor loglevel nntp 119 1
rem call ndcontrol advisor logsize nntp 119 1048576
rem call ndcontrol advisor timeout nntp 119 unlimited
rem call ndcontrol advisor interval ssl 443 5
rem call ndcontrol advisor loglevel ssl 443 1
rem call ndcontrol advisor logsize ssl 443 1048576
rem call ndcontrol advisor timeout ssl 443 unlimited
rem

```

권고자 예제

다음은 **ADV_sample**이라는 권고자 예제 파일입니다.

```

/**
 * ADV_sample: The Network Dispatcher HTTP advisor
 *
 *
 * This class defines a sample custom advisor for Network Dispatcher.
 * 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 Network Dispatcher for use in the Network Dispatcher'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 Network Dispatcher
 *   Directories:
 *
 *     nd/servers/lib/CustomAdvisors/
 *     (nd\servers\lib\CustomAdvisors on Windows 2000)
 *
 * - 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 Network Dispatcher, must be compiled with
 * the prereq version of Java.
 * To ensure access to Network Dispatcher classes, make sure that the
 * ibmd.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):
*
*   - Params
*     - 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 sDefaultLogFileName = 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.
*   - Params
*     - String sDataString - The data to be sent is sent in the form of a
*       string
*   - Return
*     - int RC - Whether the data was successfully sent or not: zero
*       indicates data was sent; a negative integer indicates an
*       error.
*
* - receive - Receive information from the socket connection.
*   - Params
*     - 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 Network Dispatcher 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_INTERVĀL = 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_Network_Dispatcher_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_INTERVĀL,
              "", // 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 Network
*            Dispatcher-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 higher the weight will be within Network Dispatcher
* regarding load balancing.
*
* 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.
* Network Dispatcher will not attempt to load balance to a server that
* is down. Network Dispatcher will resume load balancing to the server
* when a positive value is received.
*
* A value of zero is typically not returned; Network Dispatcher handles
* a load of zero in a special way. Zero is assumed to indicate an
* unknown status, and Network Dispatcher gives the server a high
* weight in response.
*/
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);

// If the receive is successful, a load of zero is returned.
// This is because the "replace" flag is set to false,
// indicating that the load built within the base advisor is
// to be used.
// Since nothing was done with the returned data, additional
// load is not necessary.

// Note: it is known that the advisor base load will not be
// zero, therefore a zero load will

```

```
// not be returned for use in calculating the weight.  
if (iRc >= 0)  
{  
  iLoad = 0;  
}  
}  
return iLoad;  
}  
  
} // End - ADV_sample
```

부록G. Dispatcher, CBR 및 Caching Proxy를 사용하는 2단 고가용성 구성 샘플

이 부록은 두 개의 Network Dispatcher 구성요소(Dispatcher 구성요소 및 CBR 구성요소)를 Caching Proxy와 결합하는 2단 고가용성 구성을 어떻게 설정하는지를 설명합니다.

서버 시스템 설정

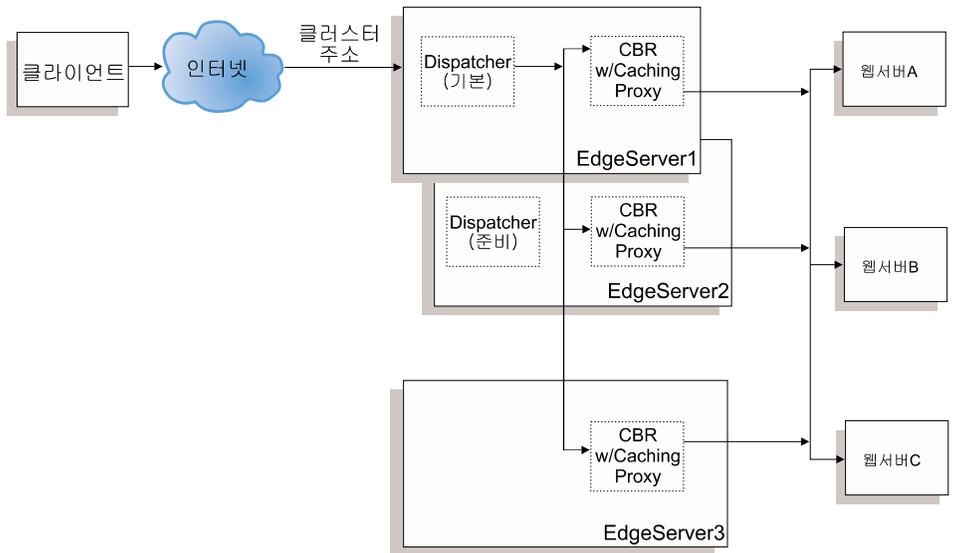


그림 30. Dispatcher, CBR 및 Caching Proxy를 사용하는 2단 고가용성 구성 예제

그림30에 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

427 페이지의 그림30에서는 다중 백엔드 웹 서버를 가로질러 로드 밸런싱하는 다중 서버(EdgeServer1, EdgeServer2, EdgeServer3)의 기본 표현을 보여줍니다. CBR 구성요소는 Caching Proxy를 사용하여 URL 콘텐츠에 근거한 요청을 백엔드 웹 서버에 전달합니다. Dispatcher 구성요소는 Edge Server를 가로질러 CBR 구성요소를 로드 밸런싱하는데 사용합니다. Dispatcher 구성요소의 고가용성 기능을 사용하면 기본 고가용성 시스템(EdgeServer1)이 작동하지 않을 때 백엔드 서버에 계속해서 요청할 수 있습니다.

기본 구성 지침

- Caching Proxy를 모든 Edge Server에서 동일하게 설정하십시오. 백엔드 서버에서 웹 페이지에 접근 가능성을 전반적으로 향상시키려면 Caching Proxy를 설정하여 메모리 캐시를 하십시오. 이렇게 하면 Edge Server는 더 자주 요청되는 웹 페이지를 캐시할 수 있습니다. Caching Proxy 설정에 대한 자세한 정보는 멀티플랫폼용 *IBM WebSphere Edge Server* 관리 안내서를 참조하십시오.
- 클러스터 주소 및 포트를 Network Dispatcher의 Dispatcher 구성요소와 CBR에서 모두 동일하게 정의하십시오.
- CBR 구성요소를 모든 Edge Server에서 동일하게 구성하십시오. 클러스터에 대하여 정의하려는 포트의 서버로 웹 서버 A, B 및 C를 사용하십시오. CBR 구성에 대한 자세한 정보는 95 페이지의 『제7장 Content Based Routing 구성요소 구성』을 참조하십시오.
- Dispatcher 구성요소를 Edge Server 1 및 2에서 동일하게 구성하십시오. Dispatcher가 로드 밸런싱하는 클러스터에서 정의하려는 포트의 서버로 모든 Edge Server를 정의하십시오. Dispatcher 구성 방법에 대한 자세한 정보는 67 페이지의 『제5장 Dispatcher 구성요소 구성』을 참조하십시오.
- Edge Server 1은 기본 고가용성 시스템으로, Edge Server 2는 대기 고가용성 시스템으로 구성하십시오. 자세한 정보는 187 페이지의 『고가용성』을 참조하십시오.

주:

1. 클러스터 주소와 연관된 호스트 이름(예: www.company.com)은 "Hostname" 지정문에 대한 Caching Proxy 구성에서 갱신되어야 합니다.
2. URL에 백엔드 서버 주소가 표시되지 않게 하려면 Caching Proxy 구성 파일에서 "SendRevProxyName" 지정문을 "yes"로 설정해야 합니다.
3. 웹 메모리가 효과적으로 사용되는지 확인하려면, "캐시" 지정문을 "ON"으로 설정하고 "CacheMemory" 지정문을 Caching Proxy 구성 파일이 요구하는 크기로 증가시키십시오.
4. IP 주소 대신에 인바운드 URL 이름으로 캐시하려면, Caching Proxy 구성 파일의 맵핑 규칙 섹션에서 프록시 지정문으로 행을 추가하십시오.

위의 주 1-4에서 언급한 샘플 행

```

Hostname          www.company.com
SendRevProxyName  yes
Caching           ON
CacheMemory       128000 K
Proxy             /* http://www.company.com/* www.company.com

```

5. EdgeServer1의 네트워크 인터페이스 카드의 클러스터 주소에 별명을 지정하고 나머지 Edge Server의 루프백 장치 클러스터 주소에 별명을 지정하십시오.
6. Edge Server에 Linux 플랫폼을 사용할 경우, Linux 커널에 패치를 설치해야 합니다. 자세한 정보는 83 페이지의 『Linux 커널 패치 설치(루프백 인터페이스에서 arp 응답 억제)』를 참조하십시오.
7. CBR에 경우 포트 연관 관계(stickytime)는 콘텐츠 규칙이 사용될 때는 사용하지 않습니다. 포트 연관 관계를 사용하는 경우, 백엔드 웹 서버로 요청을 처리하는 동안 콘텐츠 규칙이 적용되지 않습니다.

샘플 구성 파일:

다음의 샘플 구성 파일은 427 페이지의 그림30에서와 같이 Edge Server 구성 설정 시 작성되는 파일과 유사합니다. 샘플 구성 파일은 Network Dispatcher의 CBR 구성요소 및 Dispatcher용 파일을 표시합니다. 샘플 구성에서 하나의 이더넷 어댑터는 Edge Server 시스템 각각에 대하여 사용되고 모든 주소는 개인용 서브넷 내에서 표시됩니다. 샘플 구성 파일은 지정된 시스템에 대해 다음 IP 주소를 사용합니다.

- EdgeServer1 (기본 고가용성 Edge Server): 192.168.1.10
- EdgeServer2 (백업 고가용성 Edge Server): 192.168.1.20
- EdgeServer3 (웹 캐싱 Edge Server): 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 Edge Server의 Dispatcher 구성요소에 대한 샘플 구성 파일

```
ndcontrol executor start

ndcontrol cluster add 192.168.1.11 primaryhost 192.168.1.10

ndcontrol port add 192.168.1.11:80

ndcontrol server add 192.168.1.11:80:edgeserver1 address 192.168.1.10
ndcontrol server add 192.168.1.11:80:edgeserver2 address 192.168.1.20
ndcontrol server add 192.168.1.11:80:edgeserver3 address 192.168.1.30

ndcontrol manager start manager.log 10004

ndcontrol highavailability heartbeat add 192.168.1.10 192.168.1.20
ndcontrol highavailability backup add primary auto 4567
```

Edge Server의 CBR 구성요소에 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
```

부록H. 기타 자원

명령행 액세스

여러 상황에서, 마우스 조치로 수행할 수 있는 작업을 키 또는 키 조합을 통해 수행할 수도 있습니다. 대부분의 메뉴 조치는 키보드에서 시작할 수 있습니다.

키보드 사용에 대한 지침은 운영 체제에 대한 문서를 참조하십시오.

온라인 도움말 받기

Network Dispatcher에는 제품을 설치, 계획, 구성 및 작동하는 동안 수행할 작업을 설명하는 온라인 도움말 기능이 들어 있습니다.

현재 창에 대한 도움말을 받으려면 오른쪽 상단 모서리에서 물음표(?)를 누르십시오. 다음 중에서 선택하십시오.

필드 도움말

수행 중인 태스크에 관한 상황에 맞는 도움말입니다.

수행 방법

현재 창과 관련된 태스크를 나열합니다.

목차 모든 도움말 정보의 목차입니다.

색인 도움말 항목의 영문자 색인입니다.

참조서 정보

Network Dispatcher 사용에 관한 자세한 정보는 다음을 참조하십시오.

- WebSphere Edge Server 웹 사이트.

<http://www.ibm.com/software/webservers/edgeserver>

- Network Dispatcher technote 웹 사이트.

<http://www.ibm.com/software/webservers/edgeserver/support.html>

Network Dispatcher 힌트 및 팁 탐색을 누르십시오.

부록. 주의사항

IBM은 다른 국가에서 이 책에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 이 책에서 IBM 제품, 프로그램 또는 서비스를 언급했다고 해서 해당 IBM 제품, 프로그램 또는 서비스만을 사용할 수 있다는 것을 의미하지는 않습니다. IBM의 지적 재산을 침해하지 않는한, 기능상으로 동등한 제품, 프로그램 또는 서비스를 대신 사용할 수도 있습니다. 그러나 비IBM 제품, 프로그램 또는 서비스 운영에 대한 평가 및 검증은 사용자 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 조회는 다음 주소로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩
한국 아이.비.엠 주식회사
고객만족센터
전화번호: 080-023-8080

(i) 독자적으로 작성된 프로그램 및 기타 프로그램(이 프로그램 포함)간의 정보 교환 (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

이 책에 기술된 사용권 프로그램과 이 프로그램에 사용 가능한 모든 사용권 자료는 IBM 고객 협약 조건하에 IBM에서 제공합니다.

이 책은 어떠한 종류의 보증도 없이 현상태대로 제공되므로, 판매 가능성을 보장하거나 특정 목적에 적합한지 여부에 대해서는 책임질 수 없습니다.

이 제품에는 CERN에서 작성되어 사용할 수 있게 된 컴퓨터 소프트웨어가 포함되어 있습니다. 이에 대한 승인은 여기에 포함된 CERN 컴퓨터 소프트웨어나 이에 관한 부품을 포함하는 모든 제품에 그대로 언급됩니다.

상표

다음 용어는 미국 또는 기타 국가에서 사용되는 IBM Corporation의 상표 또는 등록상표입니다.

AIX

IBM

IBMLink

LoadLeveler

OS/2

NetView

WebSphere

Lotus는 미국 또는 기타 국가에서 사용되는 Lotus Development Corporation의 등록상표입니다.

Domino는 미국 또는 기타 국가에서 사용되는 Lotus Development Corporation의 상표입니다.

Tivoli는 미국 또는 기타 국가에서 사용되는 Tivoli Systems, Inc.의 등록상표입니다.

Java, 모든 Java 기반 상표와 로고는 미국 또는 기타 국가에서 사용되는 Sun Microsystems, Inc.의 상표 또는 등록상표입니다.

Solaris는 미국 또는 기타 국가에서 사용되는 Sun Microsystems, Inc.의 상표입니다.

Microsoft와 Windows 2000은 미국 또는 기타 국가에서 사용되는 Microsoft Corporation의 상표 또는 등록상표입니다.

Cisco는 미국 또는 기타 국가에서 사용되는 Cisco Systems, Inc.의 등록상표입니다.

HP는 미국 또는 기타 국가에서 사용되는 Hewlett-Packard Company의 상표입니다.

Linux는 Linus Torvalds의 등록상표입니다.

Red Hat은 Red Hat, Inc.의 등록상표입니다.

UNIX는 미국 또는 기타 국가에서 사용되는 The Open Group의 등록상표입니다.

이중 별표(**)가 표시된 다른 회사, 제품 및 서비스 이름은 해당 회사의 상표 또는 서비스표입니다.

용어집

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 Dispatcher가 넘겨 받을 수 있는 Dispatcher 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.

관리 프로그램. 여러 네트워크 Dispatcher 기능 중 하나. 관리 프로그램은 실행 프로그램의 내부 카운터와 권고자가 제공하는 피드백에 따라 가중치를 설정합니다. 실행 프로그램에서는 가중치를 사용하여 로드 밸런스를 수행합니다.

권고자. 권고자는 Network Dispatcher의 기능입니다. 권고자는 각각의 서버의 피드백을 수집하여 분석한 후 관리 프로그램 기능에 알립니다.

규칙. 규칙 기반 로드 밸런스에서, 목적지 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 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 엔터티의 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. NATP, 포트 매핑이라고도 합니다. 다른 포트 번호에서 인식하도록 여러 서버 디면을 하나의 물리적 서버 내에 구성할 수 있습니다.

넷마스크. 인터넷 서브네트워크의 경우, IP 주소의 호스트 부분에서 서브네트워크 주소 비트를 식별하는 데 사용하는 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 권고자.

도달 주소. Dispatcher의 고가용성에서 권고자가 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 원인지는 디먼이 잠재해 있다는 것을 인식하지 않아도 됩니다(그러나 프로그램은 디먼을 암시적으로 호출할 것을 알기 때문에 조치를 파악함).

라

라우터. 패킷을 네트워크 사이로 전달하는 장치. 전달 결정은 흔히 경로 지정 제품에 의해 작성되는 네트워크 계층 정보와 라우팅 테이블을 기준으로 합니다.

라우트. 기점에서 목적지까지의 네트워크 통신 경로.

루우프백 별명. 루우프백 인터페이스와 연관된 대체 IP 주소. 대체 주소는 실제 인터페이스에서 눈에 띄지 않는 유용한 부수 영향을 미칩니다.

루우프백 인터페이스. 같은 시스템 내에서 정보가 한 엔티티에 대해 주소지정될 때, 불필요한 통신 기능을 생략하는 인터페이스.

루트 사용자. 대개 시스템을 관리하는 사용자와 관련된 AIX, Red Hat Linux 또는 Solaris 운영 체제의 일부를 액세스하고 수정할 수 있는 제한되지 않는 권한.

리턴 주소. 고유한 IP 주소 또는 호스트 이름. Dispatcher 시스템에 구성되며, 서버에 대한 클라이언트의 요청을 로드 밸런싱할 때 Dispatcher가 출발지 주소로 사용합니다.

마

마법사. 특정 작업을 통해 사용자를 안내하는 단계별 지침을 사용하는 응용프로그램 내의 대화.

메트릭. 네트워크의 로드 밸런싱에 사용할 수 있는 수치(예: 현재 로그인한 사용자의 수)를 리턴하는 프로세스 또는 명령.

목적지 주소. 핵심 메시지와 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). 전용 시스템이 없는 경우, Dispatcher는 로드 밸런싱인 동일한 시스템에 설치됩니다.

주: 결합 배치는 AIX, Red Hat Linux 및 Solaris 운영 체제에만 적용됩니다.

백업. Dispatcher의 고가용성에서 기본 시스템의 파트너. 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.

비전달 주소. 관리 및 구성에 사용되는 Network Dispatcher 시스템의 기본 IP 주소.

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. 표준 IP 주소는 32비트 주소 필드입니다. 서버 주소는 점분리 10진수 IP 주소 또는 호스트 이름일 수 있습니다.

서브넷 마스크. 인터넷 서브네트워크의 경우, IP 주소의 호스트 부분에서 서브네트워크 주소 비트를 식별하는 데 사용되는 32비트 마스크.

서비스. 하나 이상의 노드에서 제공되는 기능으로 HTTP, FTP, 텔넷 등이 있습니다.

소스 주소. Dispatcher 고가용성에서 핵심 메시지를 전송하는 고가용성 상대 시스템의 주소.

셸. 사용자의 워크스테이션에서 명령행을 승인하고 처리하는 소프트웨어. Korn 셸은 사용 가능한 몇 가지의 UNIX 셸 중 하나입니다.

시간 종료. 발생한 조작에 할당된 시간 간격.

실행 프로그램. 여러 Dispatcher 기능 중 하나. 실행 프로그램은 요청을 TCP나 UDP 서버에 라우트하고, 새로운 연결, 활성 연결 및 종료된 연결의 수를 모니터링하여 완료되거나 재설정된 연결의 가비지 컬렉션을 수행합니다. 실행 프로그램은 관리 프로그램 기능에 새로운 연결과 활성 연결을 제공합니다. Cisco Consultant에서 실행 프로그램에는 구성 정보가 보관되고 Cisco CSS Switch에 연결하는 데 필요한 정보가 포함됩니다.

아

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 관리 프로그램의 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, Network Dispatcher 가비지 컬렉터는 연결에 예약된 메모리를 지울 수 있습니다.

우선순위. 규칙 기반 로드 밸런스에서 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.

전력어. Dispatcher 고가용성에서 활성 시스템의 장애 후에 수행할 복구 방법을 지정하는 키워드.

점분리 십진수 표기법. 기본 10으로 기록되고 마침표(점)로 구분되는 네 개의 8비트 숫자로 이루어진 32비트 정수의 구분 표시. IP 주소를 나타내는 데 사용됩니다.

주소. 네트워크에 연결되어 있는 각 장치나 워크스테이션에 지정되는 고유한 코드. 표준 IP 주소는 32비트 주소 필드입니다. 이 필드에는 두 부분이 포함되어 있습니다. 첫 번째 부분은 네트워크 주소이고, 두 번째 부분은 호스트 번호입니다.

차

최대 범위. 규칙 기반 로드 밸런스에서 하나의 규칙에 지정된 상위 값. 이 값의 기본값은 규칙 유형에 따라 달라집니다.

최소 범위. 규칙 기반 로드 밸런스에서 하나의 규칙에 지정된 하위 값. 이 값의 기본값은 규칙 유형에 따라 달라집니다.

카

클라이언트. 다른 컴퓨터 시스템이나 프로세스의 서비스를 요청하는 컴퓨터 시스템이나 프로세스. 예를 들면, 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. Network Dispatcher는 이 클러스터된 서버 사이에서 TCP 또는 UDP 통신량의 밸런스를 조정합니다.

타

텔넷. 터미널 에뮬레이션 프로토콜로 원격 연결 서비스를 위한 TCP/IP 응용프로그램 프로토콜. 텔넷을 사용하여 사용자의 워크스테이션이 해당 원격 호스트에 직접 연결된 것처럼 한 사이트에서 원격 호스트에 액세스할 수 있습니다.

과

패킷. 인터넷이나 기타 패킷 교환 네트워크에서 기점 및 목적지 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

핵심 메시지. 활성 Dispatcher의 상태를 모니터링하기 위해 대기 Dispatcher가 사용하는 고가용성 모드의 두 Dispatcher 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 태스크를 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. Network Dispatcher의 구성요소. CBR은 지정된 규칙 유형을 사용하는 웹 페이지 콘텐츠에 따라 HTTP 또는 HTTP 서버에 대한 수신 요청을 로드 밸런싱하기 위해 Caching Proxy와 함께 작동합니다.

cbrcontrol. Network Dispatcher의 콘텐츠 기반 라우터 구성요소에 인터페이스를 제공합니다.

cbrserver. 콘텐츠 기반 라우터에서 실행 프로그램, 관리 프로그램 및 권고자의 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 태스크를 수행합니다.

CGI 스크립트(CGI script). 양식 처리와 같이 대개 서버에서 수행되지 않는 태스크를 수행하기 위해 공통 게이트웨이 인터페이스를 사용하는 Perl이나 REXX와 같은 스크립트 언어로 작성되는 CGI 프로그램.

Cisco Consultant. IBM Network Dispatcher의 구성요소. Cisco Consultant는 Network Dispatcher 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.

D

Dispatcher. 링크된 각각의 서버 그룹 사이에서 TCP 또는 UDP 통신량 밸런싱을 효율적으로 조정하는 Network Dispatcher의 구성요소. Dispatcher 시스템은 Dispatcher 코드를 실행 중인 서버입니다.

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. 하이퍼 텍스트 마크업 언어. 하이퍼 텍스트 문서를 작성하는 데 사용되는 언어. 하이퍼 텍스트 문서에는 강조 표시된 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.

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 32비트 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.

lbc. 로드 밸런스 컨설턴트

lbcontrol. Cisco Consultant에서 Cisco CSS Switch에 인터페이스를 제공합니다.

lbserver. Cisco Consultant에서 구성 정보를 포함하며 명령을 수행합니다.

M

MAC 주소. LAN 또는 LAN 에뮬레이션 개념(LAN emulation concept).

Mailbox Locator. Network Dispatcher의 구성요소. IMAP 및 POP3 프로토콜의 경우, Mailbox Locator는 사용자 ID와 암호에 근거하여 올바른 서버를 선택하는 프록시입니다.

Metric Server. 전에는 SMA(Server Monitor Agent)라고 했습니다. Metric server는 Network Dispatcher 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.

mlcontrol. Network Dispatcher의 Mailbox Locator 구성요소에 인터페이스를 제공합니다.

mlserver. Mailbox Locator에서 구성 정보를 포함하며 명령을 수행합니다.

N

ndcontrol. Network Dispatcher의 Dispatcher 구성요소에 인터페이스를 제공합니다.

ndserver. Dispatcher에서 실행 프로그램, 관리 프로그램 및 권고자에 대한 명령행의 요청을 처리합니다.

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.

P

PICS. 인터넷 내용 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 집합. 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 오브젝트에 액세스할 수 있는 Java 프로그래밍 언어 라이브러리의 부분.

RPM. Red Hat 패키지 관리 프로그램.

S

Site Selector. Network Dispatcher의 DNS 기반 로드 밸런스 구성요소. Site Selector는 해당 서버에서 실행되는 Mertric Server 구성요소에서 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. Network Dispatcher의 Site Selector 구성요소에 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.

ssserver. Site Selector에서 사이트 이름, 관리 프로그램 및 권고자에 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. Network Dispatcher가 다른 서버에 대해 하나의 가상 서버로 링크하는 서버. Network Dispatcher는 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.

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 관리 프로그램. Dispatcher와 함께 제공되는 권고자. MVS WLM(작업로드 관리 프로그램) 구성요소를 실행하는 OS/390 메인프레임의 서버하고만 작동되도록 설계되었습니다.

색인

[가]

가비지 컬렉션 234

가중치

관리 프로그램 설정 방법 154, 228
설정

서버용 341, 374, 409, 410
포트상의 모든 서버 경계 154,
325, 406

가중치 갱신에 대한 감도, 설정 156,
314, 362, 365, 399, 402

간격, 빈도 설정

관리 프로그램에서 실행 프로그램 조
회 155, 313, 398, 401
관리 프로그램에서 실행 프로그램에
대한 가중치 갱신 155, 312, 361,
363, 397, 400
권고자가 서버 조회 284, 355, 382,
385

개요

CBR 구성 95
Cisco Consultant 구성 141
Dispatcher 구성요소 구성 67
Mailbox Locator 구성 113
Site Selector 구성 127

개인용 키

원격 인증용 230

검사

구성 147

결합 배치(키워드) 176, 341

결합 배치, Network Dispatcher 및 서버
70, 77, 175, 179, 336, 341

결합(연관 관계)

결합(규칙 연관 관계 무시) 336
수동 쿠키 219, 331
지금 작업중지 310, 315

결합(연관 관계) (계속)

포트간 연관 관계 320

활성 쿠키 218, 331

stickymask 321

stickytime 64, 65, 321, 331

URI 331

결합(친근성)

결합 시간 212, 214

결합(규칙 친근성 무시) 216

규칙 친근성 무시 216

사용법 212

수동 쿠키 217

지금 작업중지 216

친근성 주소 마스크 215

포트간 친근성 214, 215

활성 쿠키 217

SDA(Server Directed
Affinity) 213

stickymask 214, 215

URI 217

계획

CBR 89

Cisco Consultant 135

Dispatcher 구성요소 55

Mailbox Locator 109

Site Selector 121

고가용성 34, 53, 58, 187

구성 188

기본 호스트 289

상호 59, 189, 289, 291, 305

스크립트 193

goActive 193

goIdle 194

goInOp 194

goStandby 194

고가용성 (계속)

스크립트 (계속)

highavailChange 195

ndcontrol 302

primaryhost 291

공용 키

원격 인증용 230

관리 프로그램

고정 가중치 155

버전 316, 363, 365, 400, 403

비율 152

시작 77, 146, 315, 363, 365, 399,
403

정지 315, 363, 365, 400, 403

광역 지원 177

구성 예제 182

원격 권고자 사용 179

원격 Dispatcher 사용 178

GRE 사용 185

구문 다이어그램

구두점 273

기호 273

매개변수 273

예제 274

읽기 273

구성

검사 147

관리 프로그램 시작 146

로드 밸런스 서버 정의 146

마법사 5

방법

마법사(CBR) 100

마법사(Dispatcher) 70

마법사(Mailbox Locator) 117

마법사(Site Selector) 130

구성 (계속)

방법 (계속)

명령행(CBR) 96
 명령행(Cisco Consultant) 142
 명령행(Dispatcher) 68
 명령행(Mailbox Locator) 114
 명령행(Site Selector) 128
 스크립트(CBR) 98
 스크립트(Cisco Consultant) 143
 스크립트(Dispatcher) 68
 스크립트(Mailbox Locator) 115
 스크립트(Site Selector) 129
 GUI(CBR) 98
 GUI(Cisco Consultant) 143
 GUI(Dispatcher) 69
 GUI(Mailbox Locator) 115
 GUI(Site Selector) 129

예제 파일 413
 클러스터 비율 설정 147
 클러스터 정의 145
 태스크, 고급 149
 포트 145
 확인 82
 Cisco Consultant 141
 Consultant와 CSS 간의 맵핑 137
 Content Based Routing 95
 Dispatcher 구성요소 67
 Mailbox Locator 113
 Metric Server 147
 Site Selector 127

권고자

목록 282, 384
 시작 146
 예제 구성 파일 421
 cbrcontrol 280
 Cisco Consultant
 간격 382, 385
 목록 383, 384
 버전 385, 387
 보고서 시간 종료 385, 387

권고자 (계속)

Cisco Consultant (계속)

상태 보고서 386
 상태 표시 385, 386
 서버 수신 시간 종료 383, 386
 서버 연결 시간 종료 382, 385
 시작 384, 386
 이름 382
 정지 385, 387
 포트 382
 Dispatcher component
 ssl2http 권고자 93
 Dispatcher 구성요소 157
 간격 159, 284
 목록 161, 284
 버전 286
 보고서 시간 종료 160, 283
 빠른 실패 발견 161
 사용자 정의 163
 상태 보고서 285
 서버 수신 시간 종료 160, 282, 285
 서버 연결 시간 종료 160, 280, 284
 시작 77, 284
 시작/정지 158
 이름 280
 자가 권고자 163, 186
 정지 284
 포트 288
 Caching Proxy권고자
 (advisor) 162
 report 286
 ssl2http 권고자 162
 HTTP 권고자 요청/응답 174
 lbcontrol 382
 Linux 제한사항 158
 mlcontrol 280
 ndcontrol 280

권고자 (계속)

Site Selector

간격 355
 목록 354, 356
 버전 355, 357
 보고서 시간 종료 355, 357
 상태 보고서 354, 356
 서버 수신 시간 종료 354, 356
 서버 연결 시간 종료 352, 355
 시작 354, 356
 이름 352
 정지 355, 357
 포트 280, 352
 interval 352
 list 353
 loglevel 353
 sscontrol 352, 360
 URL 옵션, HTTP 권고자 174
 권고자, Network Dispatcher 구성요소
 목록 385
 상태 보고서 384
 시작 77
 규칙 기반 로드 밸런스 196
 공유 대역폭 201, 202, 329, 334
 규칙 선택, 구성요소로 196
 서버 평가 옵션 206
 서비스 유형(TOS) 200, 328, 334
 시간 199, 328, 334, 370, 372
 예약된 대역폭 201, 202, 328, 334
 요청의 내용 63, 205, 329
 초당 연결 199, 328
 클라이언트 포트 200, 328
 클라이언트 IP 주소 198, 328, 334, 369, 372
 평가 옵션 206
 포트에 대한 활성 연결 199, 328
 항상 참 204, 329, 334, 370, 372
 Metric All 203
 metric average 204
 metricall 369

규칙 기반 로드 밸런스 (계속)

metricavg 370

규칙 연관 관계 무시

server 336, 341

규칙 친근성 무시

서버 216

근접 옵션 125

기본 호스트 189

[나]

내게 필요한 옵션 433

내용 규칙 205

네트워크 근접 125

[다]

다중 주소 결합 배치 77

단절, 서버 표시 341, 374

도움말, 온라인 433

등록상표 436

[라]

라우트, 여분 삭제 82

라우트, 여분의 81

로그

레벨, 설정

관리 프로그램용 231, 361, 397

권고자용 231, 285, 356, 383, 386

서버 231

서브에이전트 231

서버 정보는 2진 파일에 223

크기, 설정

관리 프로그램용 231, 312, 313, 361, 363, 364, 398, 400

권고자용 231, 285, 353, 356, 383, 386

서버 231

로그 (계속)

크기, 설정 (계속)

서브에이전트 231

파일, 이름 설정

관리 프로그램용 363, 400

권고자용 355, 384

CBR 로그 사용 243

Cisco Consultant 로그 사용 245

Mailbox Locator 로그 사용 243

Metric Server 로그 사용 245

Network Dispatcher 로그 사용 231

Site Selector 로그 사용 244

로그온/로그오프 13

로드 밸런스 설정(최적화) 152

로드 밸런스의 중요성 비율, 설정 153, 291

[마]

마법사, 구성

Dispatcher 5

명령

라우트

여분의 라우트 삭제 81, 82

cbrcontrol

advisor 280

cluster 287

executor 293

file 298

help 300

host 307

log 308

manager 309

metric 317

port 319

rule 327

server 335

set 342

status 343

Cisco Consultant 381

명령 (계속)

ifconfig 75, 180

루프백 장치에 별명 지정 79

lbcontrol

권고자 382

서버, 구성 408

cluster 388

executor 390

file 392

help 394

host 395

log 396

manager 397

metric 404

port 406

set 411

status 412

mlcontrol

advisor 280

cluster 287

executor 293

file 298

help 300

host 307

log 308

manager 309

metric 317

port 319

server 335

set 342

status 343

ndconfig 75, 180

ndcontrol

고가용성, 제어 302

관리 프로그램 제어 77

권고자 제어 77

비전달 주소 정의 73, 297, 390, 391

서버 정의 76

서브에이전트, SNMP 구성 344

명령 (계속)

- ndcontrol (계속)
 - 포트 정의 76
 - 프롬프트 278
 - advisor 280
 - cluster 287
 - executor 293
 - file 298
 - help 300
 - host 307
 - log 308
 - manager 309
 - metric 317
 - port 319
 - rule 327
 - server 335
 - set 342
 - status 343
- netstat
 - IP 주소와 별명을 확인 81
- Site Selector 351
- sscontrol
 - advisor 352
 - file 358
 - help 360
 - manager 361
 - metric 366
 - nameserver 368
 - rule 369
 - server 373
 - set 375
 - sitename 376
 - status 380
- 명령어 참조서
 - 읽는 방법 273
- 명령행
 - 구성 예제 4
 - 액세스 433
- 명시적 링크 207
- 모니터 메뉴 옵션 235

문제점 진단

- 공통 문제점 및 해결책 256, 258, 265, 267, 268, 270, 271
- 구문 또는 구성 오류 266
- 권고자가 작동하지 않음 258
- 권고자는 모든 서버가 중단되었음을 표시함 263
- 도움말 패널이 사라짐 261
- 온라인 도움말을 보려고 할 때 오류 메시지 발생 260
- 요청이 로드 밸런스되지 않음 266
- 추가 라우트 258
- 큰 구성 파일을 로드할 때의 예기치 못한 작동 264
- 포트 추가를 시도할 때 Mailbox Locator 오류 발생 268
- 포트 14099에서 레지스트리를 작성할 수 없음 270
- 포트를 추가할 수 없음 268
- 핵심 메시지를 추가할 수 없음 257
- Caching Proxy가 설치된 Dispatcher 실행 중 오류 260
- cbrcontrol 또는 ndadmin 명령 실패 265
- cbrserver 명령이 정지됨 267
- CBR에서 사용되는 포트 번호 253
- CBR이 실행되지 않음 265
- Cisco Consultant에서 사용하는 포트 번호 255
- Discovery 경로로 인해 Network Dispatcher와의 리턴 통신이 발생하지 못함 262
- Dispatcher 고가용성이 작동되지 않음 257
- Dispatcher 및 서버가 응답하지 않음 256
- Dispatcher 요청이 라우트되지 않음 256
- Dispatcher가 수행되지 않음 256

문제점 진단 (계속)

- Dispatcher에서 사용되는 포트 번호 252
- Dispatcher, Microsoft IIS 및 SSL이 작동하지 않음 258
- GUI가 올바르게 시작되지 않음 260
- GUI가 올바르게 표시되지 않음 261
- lbcontrol 또는 ndadmin 명령 실패 270
- lbserver가 시작되지 않음 270
- Mailbox Locator가 실행되지 않음 267
- Mailbox Locator에서 사용하는 포트 번호 254
- Metric Server 로그가 "에이전트에 액세스하려면 서명이 필요합니다"라고 보고합니다. 272
- Metric Server 로드를 보고하지 않음 271
- mlcontrol 또는 ndadmin 명령 실패 267
- ndcontrol 또는 ndadmin 명령 실패 259
- Network Dispatcher 실행 프로그램을 시작할 때 피란색 화면이 표시됨 262
- Network Dispatcher가 프레임을 처리하고 전달할 수 없음 261
- Network Dispatcher의 광역 모드에서 고가용성이 작동되지 않음 263
- Site Selector가 리운드 로빈하지 않음 (Solaris) 269
- Site Selector가 실행되지 않음 268
- Site Selector가 올바르게 로드 밸런스하지 않음 270
- Site Selector에서 사용하는 포트 번호 255
- SNMPD가 작동하지 않음 258
- Solaris 2.7에서 ndserver를 시작할 때 유사 오류 메시지 발생 260

문제점 진단 (계속)

Solaris에서 cbrcontrol 실패 266
 sscntrl 또는 ndadmin 명령 실패 269
 ssserver가 Windows 2000에서 시작에 실패함 269
 Windows 2000의 Metric Server IOException 271

문제점 해결 247

공통 문제점 및 해결책 256, 258, 265, 267, 268, 270, 271

구문 또는 구성 오류 266

권고자가 작동하지 않음 258

권고자는 모든 서버가 중단되었음을 표시함 263

도움말 패널이 사라짐 261

온라인 도움말을 보려고 할 때 오류 메시지 발생 260

요청이 로드 밸런스되지 않음 266

추가 라우트 258

큰 구성 파일을 로드할 때의 예기치 못한 작동 264

포트 추가를 시도할 때 Mailbox Locator 오류 발생 268

포트 14099에서 레지스트리를 작성할 수 없음 270

포트를 추가할 수 없음 268

핵심 메시지를 추가할 수 없음 257

Caching Proxy가 설치된 Dispatcher 실행 중 오류 260

cbrcontrol 또는 ndadmin 명령 실패 265

cbrserver 명령이 정지됨 267

CBR에서 사용되는 포트 번호 253

CBR이 실행되지 않음 265

Cisco Consultant에서 사용하는 포트 번호 255

Discovery 경로로 인해 Network Dispatcher와의 리턴 통신이 발생하지 못함 262

문제점 해결 (계속)

Dispatcher 고가용성이 작동되지 않음 257

Dispatcher 및 서버가 응답하지 않음 256

Dispatcher 요청이 라우트되지 않음 256

Dispatcher가 수행되지 않음 256

Dispatcher에서 사용되는 포트 번호 252

Dispatcher, Microsoft IIS 및 SSL이 작동하지 않음 258

GUI가 올바르게 시작되지 않음 260

GUI가 올바르게 표시되지 않음 261

lbccntrl 또는 ndadmin 명령 실패 270

lbserver가 시작되지 않음 270

Mailbox Locator가 실행되지 않음 267

Mailbox Locator에서 사용하는 포트 번호 254

Metric Server로고가 "에이전트에 액세스하려면 서명이 필요합니다"라고 보고합니다. 272

Metric Server로드를 보고하지 않음 271

mlcontrol 또는 ndadmin 명령 실패 267

ndcontrol 또는 ndadmin 명령 실패 259

Network Dispatcher 실행 프로그램을 시작할 때 파란색 화면이 표시됨 262

Network Dispatcher가 프레임을 처리하고 전달할 수 없음 261

Network Dispatcher의 광역 모드에서 고가용성이 작동되지 않음 263

Site Selector가 라운드 로빈하지 않음 (Solaris) 269

Site Selector가 실행되지 않음 268

문제점 해결 (계속)

Site Selector가 올바르게 로드 밸런스하지 않음 270

Site Selector에서 사용하는 포트 번호 255

SNMPD가 작동하지 않음 258

Solaris 2.7에서 ndserver를 시작할 때 유사 오류 메시지 발생 260

Solaris에서 cbrcontrol 실패 266

ssccontrol 또는 ndadmin 명령 실패 269

sssriver가 Windows 2000에서 시작에 실패함 269

Windows 2000의 Metric Server IOException 271

문제점 해결 테이블

CBR 249

Cisco Consultant 251

Dispatcher 구성요소 247

Mailbox Locator 250

Metric Server 251

Site Selector 251

[바]

바인드 고유 서버 76, 77

백업, 고가용성 58, 302

구성 188

버전, 표시

관리 프로그램 316, 363, 365, 400, 403

권고자 286, 355, 357

변경

활동해제 타이머 234

FIN 계수 234

FIN 시간 종료 234

별명

루프백 장치 78

Linux 커널 패치 79, 83

NIC 74, 105

- 보안 소켓 계층 76
 - 비율 147
 - 비전달 주소
 - 설정 297, 390, 391
 - 정의 73
 - 빠른 시작 예제 1
- [사]**
- 사설 네트워크, Dispatcher에 사용 208
 - 사용자 종료 스크립트 157
 - 서비스 거부 감지 222
 - managerAlert 157
 - managerClear 157
 - serverDown 157
 - serverUp 157
 - 삭제
 - 여분의 라우트 82
 - 클러스터 291, 378, 388, 389
 - 클러스터에서 포트 325, 406, 407
 - 포트에서 서버 341, 374, 409, 410
 - 상태, 표시
 - 모든 클러스터 389
 - 특정 포트상의 서버 325, 406, 407
 - 하나의 클러스터 389
 - 상표 436
 - 상호 고가용성 59, 188, 189
 - 스크립트 193
 - primaryhost 289, 291
 - takeover 192
 - 새 연결 226
 - 새로운 기능, V2.0
 - 사이트(클러스터) 특정 권고자 39
 - 새로운 중국어 NLS 표준 지원 35
 - 서버 파티션 38
 - 서비스 거부(DoS) 39
 - 수동 쿠키 연관 관계 38
 - 클러스터 고유 비율 38
 - 확장 사용자 엑시트 39
 - AIX v5.1 지원 34
 - 새로운 기능, V2.0 (계속)
 - CBR 가용성 향상 36
 - Cisco Consultant 35
 - DB2 권고자 40
 - Dispatcher의 Content Based Routing 37
 - HTTP 권고자 요청/응답 39
 - Linux 및 Solaris NLS 35
 - Mailbox Locator 36
 - Metric Server 36
 - NAT 및 NAPT 37
 - Red Hat Linux v7.1 지원 35
 - Site Selector 35
 - SuSE Linux v7.1 지원 35
 - URI 연관 관계 38
 - 새로운 연결, 중요성 비율 설정 153, 288, 388
 - 서버
 - 가중치 설정 341, 374, 409, 410
 - 논리 172
 - 단절 표시 341, 374
 - 리턴 주소 338
 - 물리적 172
 - 연결 표시 341, 374
 - 작업중지 216, 313, 401
 - 작업중지 해제 316, 400
 - 제거 341, 374, 409, 410
 - 추가 340, 374, 410
 - 파티션 172
 - 포트에 대해 정의 76, 340, 374, 410
 - 표준화된 가중치에 대해 모두 재시작 314, 362, 364, 399, 402
 - mapport 92
 - 서버 작업중지 216, 310, 313, 315, 401
 - 서버 통계의 2진 로그 223, 233
 - 서버 표시
 - 단절 341, 374
 - 연결 341, 374
 - 서브에이전트 231, 235
 - ndcontrol 344
 - 서비스 거부 중지 감지 222
 - halfopenaddressreport 324
 - maxhalfopen 323
 - 설정
 - 가중치 갱신에 대한 감도 156, 314, 362, 365, 399, 402
 - 간격 시간
 - 서버를 조회할 권고자 284, 355, 382, 385
 - 실행 프로그램을 갱신할 관리 프로그램 155, 312, 361, 363, 397, 400
 - 관리 프로그램에서 실행 프로그램을 조회하는 간격 155, 313, 398, 401
 - 로그 레벨
 - 관리 프로그램용 361, 397
 - 권고자용 231, 285, 356, 383, 386
 - 로그 파일의 이름 355, 384
 - 관리 프로그램용 363, 400
 - 로그의 최대 크기
 - 관리 프로그램용 312, 313, 361, 363, 364, 398, 400
 - 권고자용 231, 285, 353, 356, 383, 386
 - 로드 밸런스에서 중요성 비율 291
 - 비전달 주소 71
 - 서버의 가중치 313, 316, 341, 374, 400, 401, 409, 410
 - 최대 가중치
 - 특정 포트상의 서버 154, 325, 406
 - 클러스터 주소 76
 - smoothing index 156, 315, 363, 365, 399, 402

설정, 모든 전역값 표시
 관리 프로그램용 315, 363, 365, 400, 403
 권고자용 285, 355, 357
 설치
 AIX 16
 Linux 21
 Network Dispatcher 13
 Solaris 24
 Windows 2000 27, 28
 설치 계획 31, 55, 121
 설치 제거
 AIX 17
 Linux 21
 Solaris 24
 Windows 2000 28
 소프트웨어 요구사항
 CBR 89
 Cisco Consultant 135
 Dispatcher 구성요소 55
 Mailbox Locator 109
 Site Selector 121
 수동 쿠키 연관 관계 219, 331
 수동 쿠키 친근성 217
 수행 방법 433
 스무스 색인, 설정 156, 315, 363, 365, 399, 402
 스크립트 193
 사용자 종료 157
 goActive 193
 goIdle 194
 goInOp 194
 goStandby 194
 highavailChange 195
 시스템 메트릭
 구성 317, 366
 중요성 비율 설정 153, 287, 288
 시작
 관리 프로그램 77, 315, 363, 365, 399, 403

시작 (계속)
 권고자 77, 284, 354, 356
 서버 73
 실행 프로그램 73, 297, 391
 Cisco Consultant 244
 Dispatcher 4
 Metric Server 245
 Site Selector 244
 시작 및 정지
 CBR 242
 Dispatcher 233
 Mailbox Locator 243
 신체 장애 433
 실행 프로그램
 시작 297, 391
 정지 297

[아]

여분의 라우트 81, 82
 연결, 서버 표시 341, 374
 연결, 중요성 비율 설정 153, 291
 연관 관계 주소 마스크 321
 연관 관계(결합)
 결합(규칙 연관 관계 무시) 336
 수동 쿠키 219, 331
 지금 작업중지 310, 315
 포트간 연관 관계 320
 활성 쿠키 218, 331
 mailbox locator 112
 SSL ID (cbr 전달) 64, 65
 stickymask 321
 stickytime 64, 65, 321, 331
 URI 220, 331
 예제
 로컬 서버 관리 42, 43, 46, 47, 49, 51
 빠른 시작 1
 예제 구성 파일 413
 권고자 421

예제 구성 파일 (계속)
 Dispatcher 구성요소(AIX) 413
 Dispatcher 구성요소(Windows) 417
 온라인 도움말 433
 와일드 카드 클러스터 73, 291
 방화벽 로드 밸런스 유지 210
 서버 구성 조합 209
 투명 프록시의 경우 Caching Proxy 포함 211
 와일드 카드 포트 76, 324
 구성되지 않은 포트 통신량 경로 지정 212
 ping 권고자 162
 요구사항
 AIX 15
 Linux 20
 Solaris 23
 Windows 2000 26
 원격 관리 25, 229
 인터넷 NIC
 ibmnd.conf
 Solaris용으로 구성 71
 이주 13

[자]

자원 433
 작업로드 관리 프로그램 권고자 (WLM) 168
 전달 방법
 cbr 63
 mac 60, 62
 MAC, NAT 또는 cbr 64
 mac, nat 또는 cbr 322
 NAT 61
 정의
 비전달 주소 73, 297, 390, 391
 클러스터 291, 389
 클러스터에 포트 76, 324, 406
 포트에 서버 76, 340, 374, 410

정지
 관리 프로그램 315, 363, 365, 400, 403
 권고자 284, 355, 357
 실행 프로그램 297
 Cisco Consultant 244

제거
 여분의 라우트 82
 클러스터 291, 378, 388, 389
 클러스터에서 포트 325, 406, 407
 포트에서 서버 341, 374, 409, 410

제품 구성요소 56

조정(사용자 정의 기능) 권고자 작성
 163

주소 맵핑 파일
 예제 209

주의사항 435

[차]

최대 가중치, 설정
 특정 포트상의 서버 154, 325, 406

추가
 클러스터 291, 389
 클러스터에 포트 76, 324, 406
 포트에 서버 76, 340, 374, 410

친근성 주소 마스크 215

친근성(결합)
 결합 시간 212, 214
 결합(규칙 친근성 무시) 216
 규칙 옵션 217
 규칙 친근성 무시 216
 사용법 212
 수동 쿠키 217
 지금 작업중지 216
 친근성 주소 마스크 215
 포트간 친근성 214, 215
 활성 쿠키 217
 SDA(Server Directed Affinity) 213

친근성(결합) (계속)
 stickymask 214, 215
 URI 217

[카]

컨텐츠 규칙 63

클러스터
 비율 설정 78, 147
 와일드 카드 73
 정의 73, 145, 291, 389
 제거 291, 378, 388, 389
 주소 구성 74
 추가 291, 389
 표시
 이 클러스터의 상태 291, 389
 FIN 계수 변경 234
 FIN 시간 종료 변경 234

클러스터 고유의 proportions 377

키
 ndkeys 170, 230

키보드 433

[타]

통계 스냅샷 보고서, 표시 313, 362, 364, 399, 401
 특정 바인드 서버 158, 179

[파]

포트
 구성 145
 권고자용 280, 352
 와일드 카드 76
 제거 325, 406, 407
 최대 가중치 설정 154, 325, 406
 추가 324, 406
 클러스터에 대해 정의 76, 324, 406

포트 (계속)
 표시
 이 포트상의 서버 상태 325, 406, 407

포트간 연관 관계 320

포트간 친근성 214

표시

권고자의 상태에 대한 보고서 285, 354, 356, 384
 내부 카운터 297, 391

목록
 현재 측정 기준을 제공하는 권고자 284, 356, 385

버전 번호
 관리 프로그램의 316, 363, 365, 400, 403
 권고자의 286, 355, 357

상태

포트상의 서버 325, 406, 407
 하나의 클러스터 또는 모든 클러스터 291, 389

전역 값과 해당되는 기본 설정값

관리 프로그램용 315, 363, 365, 400, 403

권고자용 285, 355, 357

통계 보고서 313, 362, 364, 399, 401

표준화된 가중치에 대해 모든 서버 재시작 314, 362, 364, 399, 402

필드 도움말 433

[하]

하드웨어 요구사항

CBR 89

Cisco Consultant 135

Dispatcher 구성요소 55

Mailbox Locator 109

Site Selector 121

해상도, GUI 261

확인

- 여분의 라우트 81
- 활동해제 제한 시간 233, 290, 294, 322
- 활성 연결 225
- 활성 쿠키 연관 관계 218, 331
- 활성 쿠키 친근성 217

A

AIX

- 설치 16
- 요구사항 15
- apCnsvHits 226
- apSvcConnections 225

C

Caching Proxy 91

- CBR용으로 구성 101

Caching Proxy권고자(advisor) 162

CBR

- 계획 89
- 구문 또는 구성 오류 266
- 구성
 - 태스크 개요 95
 - CBR 시스템 설정 101
- 로드 밸런스 설정 152
- 문제점 해결 테이블 249
- 시작 및 정지 242
- 실행되지 않음 265
- 요청이 로드 밸런스가 되지 않음 266
- 하드웨어 및 소프트웨어 요구사항 89

Caching Proxy와 함께

- 개요 90
- 구성 108
- mapport 키워드 92
- SSL 연결 92
- ssl2http 권고자 93
- cbrcontrol 실패 265

CBR (계속)

- Dispatcher 구성요소 사용 63
- ifconfig 명령 105
- ndadmin 실패 265
- NIC 별명 지정 105
- Solaris에서 cbrcontrol 실패 266
- cbr 전달 방법 63
 - stickytime 64, 65
- cbrcontrol 명령
 - advisor 280
 - cluster 287
 - executor 293
 - file 298
 - help 300
 - host 307
 - log 308
 - manager 309
 - metric 317
 - port 319
 - rule 327
 - server 335
 - set 342
 - status 343

Cisco Consultant

- 계획 135
- 관리 프로그램 136
- 구성
 - 예제 51
 - 태스크 개요 141
 - CSS 시스템 설정 145
- 로드 밸런스 설정
 - 권고자 보고서 시간 종료 385, 387
 - 권고자 서버 시간 종료 382, 383, 385, 386
- 명령 381
- 문제점 해결 테이블 251
- 사용 244
- 시작 244
- 시작 및 정지 244

Cisco Consultant (계속)

- 시작되지 않음 270
- 실행 프로그램 136
- 포트 14099에서 레지스트리를 작성할 수 없음 270
- 하드웨어 및 소프트웨어 요구사항 135
- lbcontrol 136
- lbcontrol 실패 270
- lbserver 136
- ndadmin 137
- ndadmin 실패 270
- cluster
 - cbrcontrol 287
 - lbcontrol 388
 - mlcontrol 287
 - ndcontrol 287
 - proportions 287
- connecttimeout
 - Cisco Consultant 382
 - Site Selector 352
- Consultant와 CSS 간의 맵핑 137
- Content Based Routing 34
 - 계획 89
 - 구성
 - 태스크 개요 95
 - CBR 시스템 설정 101
 - 로드 밸런스 설정 152
 - 문제점 해결 테이블 249
 - 사용 242
 - 하드웨어 및 소프트웨어 요구사항 89
 - Dispatcher 구성요소 사용 63

D

- DB2 권고자 163
- default.cfg 73, 104, 118, 131
- Dispatcher
 - 구성
 - TCP 서버 시스템 설정 78

- Dispatcher 구성요소
 - 계획 55
 - 고가용성이 작동하지 않음 257
 - 구성
 - 사설 네트워크 설정 208
 - 태스크 개요 67
 - Network Dispatcher 시스템 설정 70
 - 권고자가 작동하지 않음 258
 - 권고자는 모든 서버가 중단되었음을 표시함 263
 - 도움말 창을 열 수 없음 260
 - 도움말 창이 사라짐 261
 - 로드 밸런스 설정 152
 - 가중치 154
 - 감도 임계치 156
 - 관리 프로그램 간격 155
 - 권고자 간격 159
 - 권고자 보고서 시간 종료 160
 - 권고자 서버 시간 종료 160
 - 상태 정보에 제공되는 중요성의 비율 152
 - 스무스 색인 156
 - 문제점 해결 테이블 247
 - 사용 233
 - 서버가 응답하지 않음 256
 - 시작 233
 - 실행 프로그램을 시작할 때 파란색 화면이 표시됨 262
 - 실행되지 않음 256
 - 요청이 밸런스되지 않음 256
 - 원격 시스템에 연결 258
 - 추가 라우트(Windows 2000) 258
 - 큰 구성 파일을 로드할 때의 예기치 못한 작동 264
 - 프레임을 전달할 수 없음 261
 - 하드웨어 및 소프트웨어 요구사항 55
 - 핵심 메시지를 추가할 수 없음 257
 - Caching Proxy 설치 시 오류 260
 - content-based routing 63

- Dispatcher 구성요소 (계속)
 - Discovery 경로로 인해 Network Dispatcher와의 리턴 통신이 발생하지 못함 262
 - GUI가 올바르게 시작되지 않음 260
 - GUI가 올바르게 표시되지 않음 261
 - MAC 전달 60
 - MS IIS와 SSL이 작동하지 않음 258
 - NAT/ NATP 61
 - ndadmin 실패 259
 - ndcontrol 실패 259
 - Network Dispatcher의 광역 모드에서 고가용성이 작동되지 않음 263
 - SNMPd가 작동하지 않음 258
 - Solaris 2.7에서 ndserver 시작 오류 260
- DPID2 237

E

- executor
 - cbrcontrol 293
 - lbcontrol 390
 - mlcontrol 293
 - ndcontrol 293

F

- file
 - cbrcontrol 298
 - lbcontrol 392
 - mlcontrol 298
 - ndcontrol 298
 - sscontrol 358
- FIN 계수 234
- FIN 계수 한계 변경 234
- FIN 시간 종료 변경 234

- Firewall 28
- ftp 권고자 280, 352

G

- goActive 193
- goIdle 194
- goInOp 194
- goStandby 194
- GRE(일반 경로 지정 캡슐화)
 - 광역 지원 185
 - OS/390 185
- GUI 6
 - 해상도 261
- GUI(Graphical User Interface) 6

H

- help
 - cbrcontrol 300
 - lbcontrol 394
 - mlcontrol 300
 - ndcontrol 300
- highavailChange 195
- host
 - cbrcontrol 307
 - lbcontrol 395
 - mlcontrol 307
 - ndcontrol 307
- http 권고자 280, 352

I

- ibmnd.conf
 - Solaris용으로 구성 71
- ibmproxy 92, 101
 - 권고자 162
- ifconfig 명령 75, 79, 105, 180
- imap
 - 달여쓰기 112

J

Java 런타임 환경(JRE) 15, 20, 23

L

lbccontrol 명령

권고자 382
cluster 388
executor 390
file 392
help 394
host 395
log 396
manager 397
metric 404
port 406
server 408
set 411
status 412

lbcserver

시작되지 않음 255, 270

Linux

설치 21
요구사항 20
캐널 패치
버전 2.2.12, 2.2.13 86
버전 2.4.x 84

log

서버 정보는 2진 파일에 308, 396
cbrcontrol 308
lbccontrol 396
mlcontrol 308
ndcontrol 308

M

mac 전달 방법 60

Mailbox Locator

개요 111

Mailbox Locator (계속)

계획 109
구성
시스템 설정 117
태스크 개요 113
로드 밸런스 설정 152
문제점 해결 테이블 250
비활동 시간 종료 290, 295, 322
사용 243
시작 및 정지 243
실행되지 않음 267
포트 추가 시도할 때 프록시 오류 268
포트를 추가할 수 없음 268
프록시 프로토콜 323, 324
하드웨어 및 소프트웨어 요구사항 109
mlcontrol 실패 267
mlserver 111
mlserver 명령이 정지됨 267
ndadmin 실패 267
staletimeout 290, 295, 322

manager

cbrcontrol 309
lbccontrol 397
mlcontrol 309
ndcontrol 309
proportions 388
sscontrol 361

metric

cbrcontrol 317
lbccontrol 404
mlcontrol 317
ndcontrol 317
sscontrol 366

Metric Server

개요 169
문제점 해결 테이블 251
사용 245
시작 147

Metric Server (계속)

시작 및 정지 245
Metric Server 로그가 "서명이 에이전트 액세스에 필요합니다"라고 보고합니다. 272
Metric Server로드를 보고하지 않음 271
Windows 2000의 Metric Server IOException 271

mlcontrol 명령

advisor 280
cluster 287
executor 293
file 298
help 300
host 307
log 308
manager 309
metric 317
port 319
server 335
set 342
status 343

N

nameserver

sscontrol 368

NAPT(네트워크 주소 포트 변환) 61

NAT 전달 방법 61

NAT(네트워크 주소 변환) 60, 61

ndconfig 180

명령 75

ndcontrol 명령

관리 프로그램 77

권고자 77

명령 매개변수 최소화 278

명령 프롬프트 278

서버 76

실행 프로그램 73

ndcontrol 명령 (계속)

- 포트 76
- advisor 280
- cluster 287
- executor 293
- file 298
- help 300
- highavailability 302
- host 307
- log 308
- manager 309
- metric 317
- port 319
- rule 327
- server 335
- set 342
- status 343
- subagent 344

ndkeys 171, 230

ndserver

- 시작 4

netstat 명령 81

Network Dispatcher

- 개요 31, 40
- 계획 고려사항 55, 121
- 구성
 - CBR 95
 - Cisco Consultant 141
 - Dispatcher 구성요소 70, 101, 117, 131
 - Site Selector 127
- 구성 태스크, 고급 149
- 기능 31, 40
- 문제점 해결 247
- 빠른 시작 예제 1
- 설치 13
- 소프트웨어 요구사항 55, 89, 109, 121, 135
- 이점 33
- 작동 및 관리 229, 244

Network Dispatcher (계속)

- 하드웨어 요구사항 55, 89, 109, 121, 135
- cnfiguring
 - Mailbox Locator 113
- Network Dispatcher 관리 229
- Network Dispatcher 작동 229
- NIC
 - 매펕(Windows 2000의 경우) 75
 - 별명 74
 - 이더넷(Solaris용) 71

O

OS/390

- GRE 지원 185

P

pop3

- 덮어쓰기 112

port

- cbrcontrol 319
- lbcontrol 406
- mlcontrol 319
- ndcontrol 319

primaryhost 291

R

RMI(원격 메소드 호출) 229

route 명령 81, 82

rule

- cbrcontrol 327
- ndcontrol 327
- sscontrol 369

S

SDA(Server Directed Affinity) 174, 213

server

- 비결합(규칙 연관 관계 무시) 336, 341
- 작업중지 310, 315
- address 336, 409
- advisorrequest 339
- advisorresponse 339
- cbrcontrol 335
- collocated 336, 341
- cookievalue 338
- fixedweight 337
- lbcontrol 408
- mapport 337
- mlcontrol 335
- ndcontrol 335
- router 338
- sscontrol 373
- weight 337

set

- cbrcontrol 342
- lbcontrol 411
- mlcontrol 342
- ndcontrol 342
- sscontrol 375

Site Selector

- 개요 48
- 계획 121
- 구성
 - 시스템 설정 131
 - 태스크 개요 127
- 구성 예제 49
- 로드 밸런스 설정 152
- 명령 351
- 문제점 해결 테이블 251
- 사용 244
- 시작 및 정지 244

Site Selector (계속)

- 실행되지 않음 268
- 중복 라우트를 통해 올바른 로드 밸런싱하지 않음 270
- 하드웨어 및 소프트웨어 요구사항 121
- HA Dispatcher 로드 밸런싱 195
- ndadmin 실패 269
- Solaris 클라이언트로부터 통신을 라운드 로빈하지 않음 269
- sscontrol 실패 269
- ssserver가 Windows 2000에서 시작에 실패함 269

sitename

- sscontrol 376

SNMP 231, 235

SNMP(Simple Network Management

- Protocol) 235

Solaris

- 설치 24
- 요구사항 23
- apr publish 명령 76
- Dispatcher 시스템 설정 71

sscontrol 명령

- advisor 352
- file 358
- help 360
- manager 361
- metric 366
- nameserver 368
- rule 369
- server 373
- set 375
- sitename 376
- status 380

SSL 76

SSL 연결

- 권고자 162
- 사용 기능에 대한 문제점 258
- CBR 92

SSL 연결 (계속)

- ibmproxy 구성 92
- ssl2http 권고자 93, 162
- status
 - cbrcontrol 343
 - lbcontrol 412
 - mlcontrol 343
 - ndcontrol 343
- system metrics
 - 구성 404
 - 중요성 비율 설정 389

U

URI 연관 관계 220, 331

URI 친근성 217

W

WAS(WebSphere Application Server)

- 권고자 165

weight

- xml 예제 227

Windows 2000

- 설치 27
- 요구사항 26
- 클러스터 구성 명령 75
- Dispatcher 시스템 설정 72
- ndconfig 명령 75



Printed in U.S.A

GA30-0345-06



Spine information:



멀티플랫폼용 WebSphere
Edge Server

Network Dispatcher 관리 안내서

버전 2.0