

## **Oracle® Solaris 관리: 네트워크 서비스**

Copyright © 2002, 2011, Oracle and/or its affiliates. All rights reserved.

본 소프트웨어와 관련 문서는 사용 제한 및 기밀 유지 규정을 포함하는 라이선스 계약서에 의거해 제공되며, 지적 재산법에 의해 보호됩니다. 라이선스 계약서 상에 명시적으로 허용되어 있는 경우나 법규에 의해 허용된 경우를 제외하고, 어떠한 부분도 복사, 재생, 번역, 방송, 수정, 라이선스, 전송, 배포, 진열, 실행, 발행, 또는 전시될 수 없습니다. 본 소프트웨어를 리버스 엔지니어링, 디스어셈블리 또는 디컴파일하는 것은 상호 운용에 대한 법규에 의해 명시된 경우를 제외하고는 금지되어 있습니다.

이 안의 내용은 사전 공지 없이 변경될 수 있으며 오류가 존재하지 않음을 보증하지 않습니다. 만일 오류를 발견하면 서면으로 통지해 주시기 바랍니다.

만일 본 소프트웨어나 관련 문서를 미국 정부나 또는 미국 정부를 대신하여 라이선스한 개인이나 법인에게 배송하는 경우, 다음 공지 사항이 적용됩니다.

#### U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

본 소프트웨어 혹은 하드웨어는 다양한 정보 관리 애플리케이션의 일반적인 사용을 목적으로 개발되었습니다. 본 소프트웨어 혹은 하드웨어는 개인적인 상해를 초래할 수 있는 애플리케이션을 포함한 본질적으로 위험한 애플리케이션에서 사용할 목적으로 개발되거나 그 용도로 사용될 수 없습니다. 만일 본 소프트웨어 혹은 하드웨어를 위험한 애플리케이션에서 사용할 경우, 라이선스 사용자는 해당 애플리케이션의 안전한 사용을 위해 모든 적절한 비상-안전, 백업, 대비 및 기타 조치를 반드시 취해야 합니다. Oracle Corporation과 그 자회사는 본 소프트웨어 혹은 하드웨어를 위험한 애플리케이션에서의 사용으로 인해 발생하는 어떠한 손해에 대해서도 책임지지 않습니다.

Oracle과 Java는 Oracle Corporation 및/또는 그 자회사의 등록 상표입니다. 기타의 명칭들은 각 해당 명칭을 소유한 회사의 상표일 수 있습니다.

Intel 및 Intel Xeon은 Intel Corporation의 상표 내지는 등록 상표입니다. SPARC 상표 일체는 라이선스에 의거하여 사용되며 SPARC International, Inc.의 상표 내지는 등록 상표입니다. AMD, Opteron, AMD 로고, 및 AMD Opteron 로고는 Advanced Micro Devices의 상표 내지는 등록 상표입니다. UNIX는 The Open Group의 등록상표입니다.

본 소프트웨어 혹은 하드웨어와 관련문서(설명서)는 제 3자로부터 제공되는 콘텐츠, 제품 및 서비스에 접속할 수 있거나 정보를 제공합니다. Oracle Corporation과 그 자회사는 제 3자의 콘텐츠, 제품 및 서비스와 관련하여 어떠한 책임도 지지 않으며 명시적으로 모든 보증에 대해서도 책임을 지지 않습니다. Oracle Corporation과 그 자회사는 제 3자의 콘텐츠, 제품 및 서비스에 접속하거나 사용으로 인해 초래되는 어떠한 손실, 비용 또는 손해에 대해 어떠한 책임도 지지 않습니다.

# 목차

---

머리말 .....	33
<b>제1부 네트워크 서비스 항목 .....</b>	<b>39</b>
<b>1 네트워크 서비스(개요) .....</b>	<b>41</b>
Oracle Solaris 11 릴리스의 항목 .....	41
Perl 5 .....	42
Perl 문서 액세스 .....	42
Perl 호환성 문제 .....	43
Perl Oracle Solaris 버전의 변경 사항 .....	43
<b>2 웹 캐시 서버 관리 .....</b>	<b>45</b>
네트워크 캐시 및 가속기(개요) .....	45
웹 서버에서 Secure Sockets Layer 프로토콜 사용 .....	46
웹 캐시 서버 관리(작업 맵) .....	47
NCA 계획 .....	47
NCA 시스템 요구 사항 .....	47
NCA 로깅 .....	48
도어 서버의 데몬 지원에 대한 라이브러리 삽입 .....	48
여러 인스턴스 지원 .....	48
웹 페이지의 캐시 관리(작업) .....	48
▼ 웹 페이지의 캐시를 사용으로 설정하는 방법 .....	48
▼ 웹 페이지의 캐시를 사용 안함으로 설정하는 방법 .....	51
▼ NCA 로깅을 사용으로 설정 또는 사용 안함으로 설정하는 방법 .....	52
NCA용 소켓 유틸리티 라이브러리를 로드하는 방법 .....	52
▼ 새 포트를 NCA 서비스에 추가하는 방법 .....	52
▼ SSL 커널 프록시를 사용하여 Apache 2.0 웹 서버를 구성하는 방법 .....	53

▼ SSL 커널 프록시를 사용하여 Sun Java System Web Server를 구성하는 방법 .....	55
영역에서 SSL 커널 프록시 사용 .....	57
웹 페이지 캐싱(참조) .....	57
NCA 파일 .....	57
NCA 구조 .....	58
<b>3 시간 관련 서비스 .....</b>	<b>61</b>
클록 동기화(개요) .....	61
이 릴리스의 NTP 정보 .....	62
NTP(Network Time Protocol) 관리(작업) .....	62
▼ NTP 서버를 설정하는 방법 .....	62
▼ NTP 클라이언트를 설정하는 방법 .....	62
▼ NTP 로깅을 사용으로 설정하는 방법 .....	63
▼ NTP 서비스와 연관된 SMF 등록 정보를 표시하는 방법 .....	63
기타 시간 관련 명령 사용(작업) .....	64
▼ 다른 시스템에서 시간 및 날짜를 동기화하는 방법 .....	64
NTP(Network Time Protocol)(참조) .....	64
<b>제2부 네트워크 파일 시스템 액세스 항목 .....</b>	<b>67</b>
<b>4 네트워크 파일 시스템 관리(개요) .....</b>	<b>69</b>
NFS 서비스의 새로운 기능 .....	69
이번 릴리스의 변경 내용 .....	69
이전 릴리스의 중요한 변경 사항 .....	71
NFS 용어 .....	72
NFS 서버 및 클라이언트 .....	72
NFS 파일 시스템 .....	72
NFS 서비스 정보 .....	73
autofs 정보 .....	73
NFS 서비스의 기능 .....	74
NFS 버전 2 프로토콜 .....	74
NFS 버전 3 프로토콜 .....	74
NFS 버전 4 프로토콜 .....	75
NFS 버전 제어 .....	76

NFS ACL 지원 .....	76
TCP를 통한 NFS .....	76
UDP를 통한 NFS .....	77
RDMA를 통한 NFS 개요 .....	77
네트워크 잠금 관리자 및 NFS .....	77
NFS 큰 파일 지원 .....	78
NFS 클라이언트 페일오버 .....	78
NFS 서비스에 대한 Kerberos 지원 .....	78
WebNFS 지원 .....	78
RPCSEC_GSS 보안 종류 .....	79
NFS 마운트를 위한 Solaris 7 확장 .....	79
WebNFS 서비스의 보안 협상 .....	79
NFS 서버 로깅 .....	79
autofs 기능 .....	80
<b>5 네트워크 파일 시스템 관리(작업) .....</b>	<b>81</b>
자동 파일 시스템 공유 .....	82
▼ 자동 파일 시스템 공유를 설정하는 방법 .....	82
▼ WebNFS 액세스를 사용으로 설정하는 방법 .....	83
▼ NFS 서버 로깅을 사용으로 설정하는 방법 .....	84
파일 시스템 마운트 .....	85
▼ 부트 시 파일 시스템을 마운트하는 방법 .....	86
▼ 명령줄에서 파일 시스템을 마운트하는 방법 .....	86
자동 마운트를 사용한 마운트 .....	87
▼ 서버에서 모든 파일 시스템을 마운트하는 방법 .....	87
▼ NFS 서버에서 큰 파일을 사용 안함으로 설정하는 방법 .....	88
▼ 클라이언트측 페일오버를 사용하는 방법 .....	89
▼ 단일 클라이언트에 대한 마운트 액세스를 사용 안함으로 설정하는 방법 .....	89
▼ 방화벽을 통해 NFS 파일 시스템을 마운트하는 방법 .....	90
▼ NFS URL을 사용하여 NFS 파일 시스템을 마운트하는 방법 .....	90
통합 파일 시스템 서버에 대해 DNS 레코드 설정 .....	91
NFS 서비스 설정 .....	91
▼ NFS 서비스를 시작하는 방법 .....	92
▼ NFS 서비스를 중지하는 방법 .....	92
▼ 자동 마운트를 시작하는 방법 .....	92

▼ 자동 마운트를 중지하는 방법 .....	93
▼ 서버에서 다른 NFS 버전을 선택하는 방법 .....	93
▼ 클라이언트에서 다른 NFS 버전을 선택하는 방법 .....	94
▼ mount 명령을 사용하여 클라이언트에서 다른 NFS 버전을 선택하는 방법 .....	95
보안 NFS 시스템 관리 .....	96
▼ DH 인증을 사용하여 보안 NFS 환경을 설정하는 방법 .....	96
WebNFS 관리 작업 .....	98
WebNFS 액세스 계획 .....	98
NFS URL을 사용한 찾아보기 방법 .....	99
방화벽을 통해 WebNFS 액세스를 사용으로 설정하는 방법 .....	100
Autofs 관리 작업 개요 .....	100
Autofs 관리 작업 맵 .....	100
SMF 매개변수를 사용하여 autofs 환경 구성 .....	102
▼ SMF 매개변수를 사용하여 autofs 환경을 구성하는 방법 .....	102
맵 관련 관리 작업 .....	102
맵 수정 .....	103
▼ 마스터 맵을 수정하는 방법 .....	104
▼ 간접 맵을 수정하는 방법 .....	104
▼ 직접 맵을 수정하는 방법 .....	104
마운트 지점 충돌 방지 .....	105
비 NFS 파일 시스템 액세스 .....	105
▼ autofs를 사용하여 CD-ROM 응용 프로그램에 액세스하는 방법 .....	106
▼ autofs를 사용하여 PC-DOS 데이터 디스켓에 액세스하는 방법 .....	106
자동 마운트 사용자 정의 .....	106
/home의 공통 보기 설정 .....	107
▼ 여러 홈 디렉토리 파일 시스템을 사용하여 /home을 설정하는 방법 .....	107
▼ /ws 아래에서 프로젝트 관련 파일을 통합하는 방법 .....	108
▼ 공유 네임스페이스에 액세스하도록 서로 다른 구조를 설정하는 방법 .....	110
▼ 호환되지 않는 클라이언트 운영 체제 버전을 지원하는 방법 .....	111
▼ 여러 서버에서 공유 파일을 복제하는 방법 .....	111
▼ autofs 보안 제한을 적용하는 방법 .....	111
▼ autofs와 함께 공용 파일 핸들을 사용하는 방법 .....	112
▼ autofs와 함께 NFS URL을 사용하는 방법 .....	112
autofs 찾아보기 기능 사용 안함으로 설정 .....	112
▼ 단일 NFS 클라이언트에서 autofs 찾아보기 기능을 완전히 사용 안함으로 설정하는 방법 .....	113

▼ 모든 클라이언트에 대해 autofs 찾아보기 기능을 사용 안함으로 설정하는 방법 .....	113
▼ 선택한 파일 시스템에 대해 autofs 찾아보기 기능을 사용 안함으로 설정하는 방법 .....	114
NFS 참조 관리 .....	115
▼ NFS 참조를 만들고 액세스하는 방법 .....	115
▼ NFS 참조를 제거하는 방법 .....	116
NFS 문제 해결 전략 .....	116
NFS 문제 해결 절차 .....	117
▼ NFS 클라이언트에서 연결을 확인하는 방법 .....	117
▼ 원격으로 NFS 서버를 확인하는 방법 .....	118
▼ 서버에서 NFS 서비스를 확인하는 방법 .....	119
▼ NFS 서비스를 다시 시작하는 방법 .....	120
NFS 파일 서비스를 제공하는 호스트 식별 .....	121
▼ mount 명령에 사용되는 옵션을 확인하는 방법 .....	121
autofs 문제 해결 .....	122
automount -v를 통해 생성되는 오류 메시지 .....	122
기타 오류 메시지 .....	123
기타 autofs 오류 .....	125
NFS 오류 메시지 .....	125
 6 네트워크 파일 시스템 액세스(참조) .....	131
NFS 파일 .....	131
/etc/default/nfslogd 파일 .....	132
/etc/nfs/nfslog.conf 파일 .....	133
NFS 데몬 .....	134
automountd 데몬 .....	135
lockd 데몬 .....	136
mountd 데몬 .....	137
nfs4cbd 데몬 .....	137
nfsd 데몬 .....	137
nfslogd 데몬 .....	138
nfsmapid 데몬 .....	139
reparse 데몬 .....	145
statd 데몬 .....	145
NFS 명령 .....	146
automount 명령 .....	146

clear_locks 명령 .....	147
fsstat 명령 .....	147
mount 명령 .....	148
umount 명령 .....	153
mountall 명령 .....	154
umountall 명령 .....	155
sharectl 명령 .....	155
share 명령 .....	158
unshare 명령 .....	162
shareall 명령 .....	163
unshareall 명령 .....	163
showmount 명령 .....	163
setmnt 명령 .....	164
nfsref 명령 .....	165
NFS 문제 해결용 명령 .....	165
nfsstat 명령 .....	165
pstack 명령 .....	167
rpcinfo 명령 .....	168
snoop 명령 .....	169
truss 명령 .....	170
RDMA를 통한 NFS .....	170
NFS 서비스의 작동 방식 .....	172
NFS의 버전 협상 .....	172
NFS 버전 4의 기능 .....	173
UDP 및 TCP 협상 .....	182
파일 전송 크기 협상 .....	182
파일 시스템 마운트 방법 .....	183
마운트 시 -public 옵션과 NFS URL의 효과 .....	184
클라이언트측 페일오버 .....	184
큰 파일 .....	186
NFS 서버 로깅의 작동 방식 .....	187
WebNFS 서비스의 작동 방식 .....	187
WebNFS 보안 협상의 작동 방식 .....	188
웹 브라우저 사용 시의 WebNFS 제한 .....	189
보안 NFS 시스템 .....	189
보안 RPC .....	190



미러 마운트의 작동 방식 .....	193
미러 마운트를 사용하는 경우 .....	193
미러 마운트를 사용하여 파일 시스템 마운트 .....	193
미러 마운트를 사용하여 파일 시스템 마운트 해제 .....	194
NFS 참조의 작동 방식 .....	194
NFS 참조를 사용하는 경우 .....	194
NFS 참조 만들기 .....	195
NFS 참조 제거 .....	195
autofs 맵 .....	195
마스터 autofs 맵 .....	195
직접 autofs 맵 .....	197
간접 autofs 맵 .....	199
autofs의 작동 방식 .....	201
autofs가 네트워크(맵)를 탐색하는 방법 .....	202
autofs에서 탐색 프로세스를 시작하는 방법(마스터 맵) .....	203
autofs 마운트 프로세스 .....	203
autofs에서 클라이언트에 대해 가장 가까운 읽기 전용 파일을 선택하는 방법(여러 위치) .....	205
autofs 및 가중치 .....	208
autofs 맵 항목의 변수 .....	208
다른 맵을 참조하는 맵 .....	209
실행 가능 autofs 맵 .....	210
autofs가 네트워크를 탐색하는 방법 수정(맵 수정) .....	210
이름 서비스에 대한 기본 autofs 동작 .....	210
autofs 참조 .....	212
autofs 및 메타 문자 .....	212
autofs 및 특수 문자 .....	213
 제3부   SLP 항목 .....	215
 7   SLP(개요) .....	217
SLP 구조 .....	217
SLP 설계 요약 .....	218
SLP 에이전트 및 프로세스 .....	218
SLP 구현 .....	220

기타 SLP 정보 원본 .....	221
<b>8 SLP 계획 및 사용으로 설정(작업) .....</b>	<b>223</b>
SLP 구성 고려 사항 .....	223
다시 구성할 항목 결정 .....	224
snoop를 사용하여 SLP 작업 모니터링 .....	224
▼ snoop를 사용하여 SLP 추적을 실행하는 방법 .....	225
snoop slp 추적 분석 .....	225
<b>9 SLP 관리(작업) .....</b>	<b>227</b>
SLP 등록 정보 구성 .....	227
SLP 구성 파일: 기본 요소 .....	228
▼ SLP 구성 변경 방법 .....	229
DA 알림 및 검색 빈도 수정 .....	230
UA 및 SA를 정적으로 구성된 DA로 제한 .....	230
▼ UA 및 SA를 정적으로 구성된 DA로 제한하는 방법 .....	230
다이얼 업 네트워크에 대한 DA 검색 구성 .....	231
▼ 다이얼 업 네트워크에 대한 DA 검색을 구성하는 방법 .....	232
자주 분할하는 경우를 위한 DA 하트비트 구성 .....	233
▼ 자주 분할하는 경우를 위한 DA 하트비트를 구성하는 방법 .....	233
네트워크 혼잡 줄이기 .....	234
다른 네트워크 매체, 토폴로지 또는 구성 수용 .....	234
SA 재등록 줄이기 .....	235
▼ SA 재등록을 줄이는 방법 .....	235
멀티캐스트 활성 시간 등록 정보 구성 .....	235
▼ 멀티캐스트 활성 시간 등록 정보를 구성하는 방법 .....	236
패킷 크기 구성 .....	237
▼ 패킷 크기를 구성하는 방법 .....	237
브로드캐스트 전용 경로 지정 구성 .....	238
▼ 브로드캐스트 전용 경로 지정을 구성하는 방법 .....	238
SLP 검색 요청에 대한 시간 초과 수정 .....	239
기본 시간 초과 변경 .....	239
▼ 기본 시간 초과를 변경하는 방법 .....	240
임의 대기 한도 구성 .....	241
▼ 임의 대기 한도를 구성하는 방법 .....	241

범위 배포 .....	242
범위 구성 시기 .....	243
범위 구성 시 고려 사항 .....	243
▼ 범위를 구성하는 방법 .....	244
DA 배포 .....	245
SLP DA를 배포하는 이유 .....	245
DA 배포 시기 .....	247
▼ DA를 배포하는 방법 .....	247
DA 배치 위치 .....	248
SLP 및 멀티홈 .....	249
SLP에 대한 멀티홈 구성 .....	249
경로가 지정되지 않은 다중 네트워크 인터페이스 구성 시기 .....	249
경로가 지정되지 않은 다중 네트워크 인터페이스 구성(작업 맵) .....	250
net.slp.interfaces 등록 정보 구성 .....	250
멀티홈 호스트에서 프록시 알림 .....	252
DA 배치 및 범위 이름 지정 .....	252
경로가 지정되지 않은 다중 네트워크 인터페이스 구성 시 고려 사항 .....	253
<b>10 레거시 서비스 통합 .....</b>	<b>255</b>
레거시 서비스를 알릴 시기 .....	255
레거시 서비스 알림 .....	255
서비스 수정 .....	255
SLP가 사용으로 설정되지 않은 서비스 알림 .....	256
SLP 프록시 등록 .....	256
▼ SLP 프록시 등록을 사용으로 설정하는 방법 .....	256
SLP 프록시 등록을 사용하여 알림 .....	257
레거시 서비스 알림 시 고려 사항 .....	259
<b>11 SLP(참조) .....</b>	<b>261</b>
SLP 상태 코드 .....	261
SLP 메시지 유형 .....	262

<b>제4부 메일 서비스 항목</b> .....	265
<b>12 메일 서비스(개요)</b> .....	267
메일 서비스의 새로운 기능 .....	267
이 릴리스의 변경 사항 .....	268
이전 릴리스의 중요한 변경 사항 .....	268
기타 sendmail 정보 소스 .....	269
메일 서비스 구성 요소 소개 .....	269
소프트웨어 구성 요소 개요 .....	269
하드웨어 구성 요소 개요 .....	270
<b>13 메일 서비스(작업)</b> .....	273
메일 서비스용 작업 맵 .....	273
메일 시스템 계획 .....	274
로컬 메일만 .....	275
로컬 메일 및 원격 구성 .....	276
메일 서비스 설정(작업 맵) .....	277
메일 서비스 설정 .....	277
▼ 메일 서버 설정 방법 .....	278
▼ 메일 클라이언트 설정 방법 .....	279
▼ 메일 호스트 설정 방법 .....	281
▼ 메일 게이트웨이 설정 방법 .....	283
▼ sendmail과 함께 DNS를 사용하는 방법 .....	284
sendmail 구성 변경(작업 맵) .....	285
sendmail 구성 변경 .....	285
▼ 새 sendmail.cf 파일 작성 방법 .....	286
가상 호스트 설정 .....	287
▼ 구성 파일을 자동으로 다시 작성하는 방법 .....	287
▼ 열기 모드에서 sendmail 사용 방법 .....	288
▼ TLS를 사용하도록 SMTP를 설정하는 방법 .....	289
▼ sendmail.cf의 대체 구성을 사용하여 메일 배달을 관리하는 방법 .....	293
편지 별칭 파일 관리(작업 맵) .....	294
편지 별칭 파일 관리 .....	295
▼ NIS mail.aliases 맵 설정 방법 .....	295
▼ 로컬 편지 별칭 파일 설정 방법 .....	296

▼ 키 맵 파일을 만드는 방법 .....	298
postmaster 별칭 관리 .....	298
대기열 디렉토리 관리(작업 맵) .....	300
대기열 디렉토리 관리 .....	301
▼ 메일 대기열 /var/spool/mqueue의 콘텐츠 표시 방법 .....	301
▼ 메일 대기열 /var/spool/mqueue에서 메일 대기열 처리 강제 실행 방법 .....	302
▼ 메일 대기열 /var/spool/mqueue의 일부를 실행하는 방법 .....	302
▼ 메일 대기열 /var/spool/mqueue 이동 방법 .....	303
▼ 이전의 메일 대기열 /var/spool/omqueue 실행 방법 .....	303
.forward 파일 관리(작업 맵) .....	304
.forward 파일 관리 .....	304
▼ .forward 파일을 사용 안함으로 설정하는 방법 .....	304
▼ .forward 파일 검색 경로 변경 방법 .....	305
▼ /etc/shells를 만들고 채우는 방법 .....	306
메일 서비스의 문제 해결 절차 및 팁(작업 맵) .....	306
메일 서비스의 문제 해결 절차 및 팁 .....	307
▼ 메일 구성 테스트 방법 .....	307
편지 별칭 확인 방법 .....	308
▼ sendmail 규칙 세트 테스트 방법 .....	308
다른 시스템에 대한 연결 확인 방법 .....	309
오류 메시지 기록 .....	310
기타 메일 진단 정보 소스 .....	311
오류 메시지 해결 .....	311
<b>14 메일 서비스(참조) .....</b>	<b>315</b>
Oracle Solaris 버전의 sendmail .....	316
sendmail 컴파일에 사용되는 플러그 및 사용되지 않는 플러그 .....	316
MILTER, sendmail용 메일 필터 API .....	317
대체 sendmail 명령 .....	318
구성 파일 버전 .....	318
메일 서비스의 소프트웨어 및 하드웨어 구성 요소 .....	319
소프트웨어 구성 요소 .....	319
하드웨어 구성 요소 .....	326
메일 서비스 프로그램 및 파일 .....	328
vacation 유틸리티의 향상된 기능 .....	329

/usr/bin 디렉토리의 내용 .....	329
/etc/mail 디렉토리의 내용 .....	330
/etc/mail/cf 디렉토리의 내용 .....	331
/usr/lib 디렉토리의 내용 .....	333
메일 서비스에 사용되는 기타 파일 .....	334
메일 프로그램의 상호 작용 .....	335
sendmail 프로그램 .....	335
메일 별칭 파일 .....	339
.forward 파일 .....	342
/etc/default/sendmail 파일 .....	343
메일 주소 및 메일 경로 지정 .....	344
sendmail과 이름 서비스의 상호 작용 .....	345
sendmail.cf 및 메일 도메인 .....	345
sendmail 및 이름 서비스 .....	346
NIS 및 sendmail의 상호 작용 .....	347
sendmail과 NIS 및 DNS의 상호 작용 .....	348
sendmail 버전 8.14의 변경 사항 .....	348
sendmail 버전 8.13의 변경 사항 .....	349
sendmail 버전 8.13에서 TLS를 사용하는 SMTP 실행 지원 .....	349
sendmail 버전 8.13의 추가 명령줄 옵션 .....	354
sendmail 버전 8.13의 추가 및 개정된 구성 파일 옵션 .....	354
sendmail 버전 8.13의 추가 및 개정된 FEATURE() 선언 .....	356
sendmail 버전 8.12에서 변경된 사항 .....	357
sendmail 버전 8.12의 TCP 래퍼에 대한 지원 .....	357
sendmail 버전 8.12의 submit.cf 구성 파일 .....	358
sendmail 버전 8.12의 추가 또는 제거된 명령줄 옵션 .....	359
sendmail 버전 8.12의 PidFile 및 ProcessTitlePrefix 옵션을 위한 추가 인수 .....	360
sendmail 버전 8.12의 추가 정의된 매크로 .....	361
sendmail 버전 8.12의 추가 매크로 .....	362
sendmail 버전 8.12의 추가 MAX 매크로 .....	362
sendmail 버전 8.12의 추가 및 개정된 m4 구성 매크로 .....	363
sendmail 버전 8.12의 FEATURE() 선언 변경 사항 .....	363
sendmail 버전 8.12에서 MAILER() 선언의 변경 사항 .....	366
sendmail 버전 8.12의 추가 배달 에이전트 플래그 .....	367
sendmail 버전 8.12에서 배달 에이전트에 대한 등식 .....	367
sendmail 버전 8.12의 추가 대기열 기능 .....	368

sendmail 버전 8.12의 LDAP에 대한 변경 사항 .....	369
sendmail 버전 8.12의 내장 메일러 변경 사항 .....	370
sendmail 버전 8.12의 추가 규칙 세트 .....	370
sendmail 버전 8.12의 파일 변경 사항 .....	371
sendmail 버전 8.12 및 구성의 IPv6 주소 .....	372
<b>제5부 직렬 네트워킹 항목 .....</b>	<b>373</b>
<b>15 Solaris PPP 4.0(개요) .....</b>	<b>375</b>
Solaris PPP 4.0 기본 사항 .....	375
Solaris PPP 4.0 호환성 .....	376
사용할 Solaris PPP 버전 .....	376
PPP에 대한 추가 정보 .....	377
PPP 구성 및 용어 .....	378
다이얼 업 PPP 개요 .....	379
전용 회선 PPP 개요 .....	382
PPP 인증 .....	384
인증자 및 피인증자 .....	385
PPP 인증 프로토콜 .....	385
PPP 인증을 사용하는 이유 .....	385
PPPoE를 통한 DSL 사용자 지원 .....	386
PPPoE 개요 .....	386
PPPoE 구성의 각 부분 .....	387
PPPoE 터널의 보안 .....	388
<b>16 PPP 링크 계획(작업) .....</b>	<b>389</b>
전반적인 PPP 계획(작업 맵) .....	389
다이얼 업 PPP 링크 계획 .....	390
다이얼 아웃 시스템을 설정하기 전에 .....	390
다이얼 인 서버를 설정하기 전에 .....	390
다이얼 업 PPP 구성의 예 .....	391
다이얼 업 PPP에 대한 추가 정보 .....	393
전용 회선 링크 계획 .....	393
전용 회선 링크를 설정하기 전에 .....	393

전용 회선 링크 구성의 예 .....	394
전용 회선에 대한 추가 정보 .....	395
링크에서 인증 계획 .....	395
PPP 인증을 설정하기 전에 .....	396
PPP 인증 구성의 예 .....	396
인증에 대한 추가 정보 .....	399
PPPoE 터널을 통한 DSL 지원 계획 .....	400
PPPoE 터널을 설정하기 전에 .....	400
PPPoE 터널 구성의 예 .....	401
PPPoE에 대한 추가 정보 .....	403
<b>17 다이얼 업 PPP 링크 설정(작업) .....</b>	<b>405</b>
다이얼 업 PPP 링크를 설정하는 주요 작업(작업 맵) .....	405
다이얼 아웃 시스템 구성 .....	406
다이얼 아웃 시스템 구성 작업(작업 맵) .....	406
다이얼 업 PPP 템플릿 파일 .....	406
다이얼 아웃 시스템에서 장치 구성 .....	407
▼ 모뎀 및 직렬 포트를 구성하는 방법(다이얼 아웃 시스템) .....	407
다이얼 아웃 시스템에서 통신 구성 .....	408
▼ 직렬 회선을 통해 통신을 정의하는 방법 .....	408
▼ 피어 호출 명령을 만드는 방법 .....	409
▼ 개별 피어를 사용하여 연결을 정의하는 방법 .....	410
다이얼 인 서버 구성 .....	412
다이얼 인 서버 구성 작업(작업 맵) .....	412
다이얼 인 서버에서 장치 구성 .....	412
▼ 모뎀 및 직렬 포트를 구성하는 방법(다이얼 인 서버) .....	413
▼ 모뎀 속도를 설정하는 방법 .....	413
다이얼 인 서버의 사용자 설정 .....	414
▼ 다이얼 인 서버의 사용자를 구성하는 방법 .....	414
다이얼 인 서버를 통한 통신 구성 .....	414
▼ 직렬 회선을 통해 통신을 정의하는 방법(다이얼 인 서버) .....	415
다이얼 인 서버 호출 .....	416
▼ 다이얼 인 서버를 호출하는 방법 .....	416



<b>18</b>	<b>전용 회선 PPP 링크 설정(작업)</b> .....	419
	전용 회선 설정(작업 맵) .....	419
	전용 회선에서 동기 장치 구성 .....	420
	동기 장치 설정을 위한 필수 조건 .....	420
	▼ 동기 장치를 구성하는 방법 .....	420
	전용 회선에서 시스템 구성 .....	421
	전용 회선에서의 로컬 시스템 구성을 위한 필수 조건 .....	421
	▼ 전용 회선에서 시스템을 구성하는 방법 .....	421
<b>19</b>	<b>PPP 인증 설정(작업)</b> .....	425
	PPP 인증 구성(작업 맵) .....	425
	PAP 인증 구성 .....	426
	PAP 인증 설정(작업 맵) .....	426
	다이얼 인 서버에서 PAP 인증 구성 .....	427
	▼ PAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버) .....	427
	PAP를 위해 PPP 구성 파일 수정(다이얼 인 서버) .....	428
	▼ PPP 구성 파일에 PAP 지원을 추가하는 방법(다이얼 인 서버) .....	429
	신뢰할 수 있는 호출자에 대해 PAP 인증 구성(다이얼 아웃 시스템) .....	430
	▼ 신뢰할 수 있는 호출자에 대해 PAP 인증 자격 증명을 구성하는 방법 .....	430
	PAP를 위해 PPP 구성 파일 수정(다이얼 아웃 시스템) .....	431
	▼ PPP 구성 파일에 PAP 지원을 추가하는 방법(다이얼 아웃 시스템) .....	432
	CHAP 인증 구성 .....	433
	CHAP 인증 설정(작업 맵) .....	433
	다이얼 인 서버에서 CHAP 인증 구성 .....	434
	▼ CHAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버) .....	435
	CHAP를 위해 PPP 구성 파일 수정(다이얼 인 서버) .....	435
	▼ PPP 구성 파일에 CHAP 지원을 추가하는 방법(다이얼 인 서버) .....	436
	신뢰할 수 있는 호출자에 대해 CHAP 인증 구성(다이얼 아웃 시스템) .....	436
	▼ 신뢰할 수 있는 호출자에 대해 CHAP 인증 자격 증명을 구성하는 방법 .....	437
	구성 파일에 CHAP 추가(다이얼 아웃 시스템) .....	438
	▼ PPP 구성 파일에 CHAP 지원을 추가하는 방법(다이얼 아웃 시스템) .....	438
<b>20</b>	<b>PPPoE 터널 설정(작업)</b> .....	439
	PPPoE 터널을 설정하는 주요 작업(작업 맵) .....	439
	PPPoE 클라이언트 설정 .....	440

PPPoE 클라이언트 설정을 위한 필수 조건 .....	440
▼ PPPoE 클라이언트용 인터페이스를 구성하는 방법 .....	440
▼ PPPoE 액세스 서버 피어를 정의하는 방법 .....	441
PPPoE 액세스 서버 설정 .....	443
▼ PPPoE 액세스 서버를 설정하는 방법 .....	443
▼ 기존 /etc/ppp/pppoe 파일을 수정하는 방법 .....	444
▼ 특정 클라이언트만 인터페이스를 사용할 수 있도록 제한하는 방법 .....	444
<b>21 일반적 인 PPP 문제 해결(작업) .....</b>	<b>447</b>
PPP 문제 해결(작업 맵) .....	447
PPP 문제 해결 도구 .....	448
▼ pppd에서 진단 정보를 가져오는 방법 .....	449
▼ PPP 디버깅을 켜는 방법 .....	450
PPP 관련 문제 및 PPPoE 관련 문제 해결 .....	451
▼ 네트워크 문제를 진단하는 방법 .....	451
PPP에 영향을 주는 일반적인 네트워크 문제 .....	453
▼ 통신 문제를 진단하고 해결하는 방법 .....	454
PPP에 영향을 주는 일반적인 통신 문제 .....	454
▼ PPP 구성을 사용하여 문제를 진단하는 방법 .....	455
일반적인 PPP 구성 문제 .....	456
▼ 모뎀 문제를 진단하는 방법 .....	456
▼ 채트 스크립트에 대한 디버깅 정보를 가져오는 방법 .....	457
일반적인 채트 스크립트 문제 .....	457
▼ 직렬 회선 속도 문제를 진단하고 해결하는 방법 .....	459
▼ PPPoE에 대한 진단 정보를 가져오는 방법 .....	460
전용 회선 문제 해결 .....	463
인증 문제 진단 및 해결 .....	463
<b>22 Solaris PPP 4.0(참조) .....</b>	<b>465</b>
파일 및 명령줄에서 PPP 옵션 사용 .....	465
PPP 옵션을 정의하는 위치 .....	465
PPP 옵션이 처리되는 방법 .....	466
PPP 구성 파일 권한의 작동 방식 .....	467
/etc/ppp/options 구성 파일 .....	469
/etc/ppp/options.ttyname 구성 파일 .....	471

사용자별 옵션 구성 .....	473
다이얼 인 서버에서 \$HOME/.ppprc 구성 .....	473
다이얼 아웃 시스템에서 \$HOME/.ppprc 구성 .....	473
다이얼 인 서버와의 통신을 위한 정보 지정 .....	473
/etc/ppp/peers/peer-name 파일 .....	474
/etc/ppp/peers/myisp.tmpl 템플릿 파일 .....	475
/etc/ppp/peers/peer-name 파일의 예를 찾을 수 있는 위치 .....	476
다이얼 업 링크를 위한 모뎀 속도 구성 .....	476
다이얼 업 링크에서 대화 정의 .....	476
채트 스크립트의 내용 .....	477
채트 스크립트 예 .....	477
채트 스크립트 호출 .....	484
▼ 채트 스크립트를 호출하는 방법(작업) .....	484
실행 가능한 채트 파일 만들기 .....	485
▼ 실행 가능한 chat 프로그램을 만드는 방법 .....	485
링크에서 호출자 인증 .....	486
PAP(암호 인증 프로토콜) .....	486
CHAP(Challenge-Handshake 인증 프로토콜) .....	489
호출자를 위한 IP 주소 지정 체계 만들기 .....	492
호출자에게 동적 IP 주소 지정 .....	492
호출자에게 정적 IP 주소 지정 .....	493
sppp 장치 번호별로 IP 주소 지정 .....	494
DSL 지원을 위해 PPPoE 터널 만들기 .....	494
PPPoE용 인터페이스를 구성하기 위한 파일 .....	495
PPPoE 액세스 서버 명령 및 파일 .....	496
PPPoE 클라이언트 명령 및 파일 .....	502
<b>23 비동기 Solaris PPP에서 Solaris PPP 4.0으로 마이그레이션(작업) .....</b>	<b>505</b>
asppp 파일을 변환하기 전에 .....	505
/etc/asppp.cf 구성 파일의 예 .....	505
/etc/uucp/Systems 파일의 예 .....	506
/etc/uucp/Devices 파일의 예 .....	507
/etc/uucp/Dialers 파일의 예 .....	507
asppp2pppd 변환 스크립트 실행(작업) .....	508
작업 필수 조건 .....	508

▼ asppp에서 Solaris PPP 4.0으로 변환하는 방법 .....	508
▼ 변환 결과를 보는 방법 .....	509
<b>24 UUCP(개요) .....</b>	<b>511</b>
UUCP 하드웨어 구성 .....	511
UUCP 소프트웨어 .....	512
UUCP 데몬 .....	512
UUCP 관리 프로그램 .....	513
UUCP 사용자 프로그램 .....	513
UUCP 데이터베이스 파일 .....	514
UUCP 데이터베이스 파일 구성 .....	515
<b>25 UUCP 관리(작업) .....</b>	<b>517</b>
UUCP 관리(작업 맵) .....	517
UUCP 로그인 추가 .....	518
▼ UUCP 로그인을 추가하는 방법 .....	518
UUCP 시작 .....	519
▼ UUCP를 시작하는 방법 .....	519
uudemon.poll 셸 스크립트 .....	520
uudemon.hour 셸 스크립트 .....	520
uudemon.admin 셸 스크립트 .....	520
uudemon.cleanup 셸 스크립트 .....	520
TCP/IP를 통해 UUCP 실행 .....	521
▼ TCP/IP에 대해 UUCP를 활성화하는 방법 .....	521
UUCP 보안 및 유지 관리 .....	522
UUCP 보안 설정 .....	522
정기 UUCP 유지 관리 .....	522
UUCP 문제 해결 .....	523
▼ 고장난 모뎀이나 ACU를 확인하는 방법 .....	523
▼ 전송을 디버그하는 방법 .....	524
UUCP/etc/uucp/Systems 파일 확인 .....	525
UUCP 오류 메시지 확인 .....	525
기본 정보 확인 .....	525

<b>26 UUCP(참조)</b>	527
UUCP /etc/uucp/Systems 파일	527
/etc/uucp/Systems 파일의 System-Name 필드	528
/etc/uucp/Systems 파일의 Time 필드	528
/etc/uucp/Systems 파일의 Type 필드	529
/etc/uucp/Systems 파일의 Speed 필드	530
/etc/uucp/Systems 파일의 Phone 필드	530
/etc/uucp/Systems 파일의 Chat-Script 필드	531
채트 스크립트를 통해 다이얼 백을 사용으로 설정	532
/etc/uucp/Systems 파일의 하드웨어 흐름 제어	533
/etc/uucp/Systems 파일에서 패리티 설정	533
UUCP /etc/uucp/Devices 파일	534
/etc/uucp/Devices 파일의 Type 필드	534
/etc/uucp/Devices 파일의 Line 필드	536
/etc/uucp/Devices 파일의 Line2 필드	536
/etc/uucp/Devices 파일의 Class 필드	536
/etc/uucp/Devices 파일의 Dialer-Token-Pairs 필드	537
/etc/uucp/Devices 파일의 Dialer-Token-Pairs 필드 구조	537
/etc/uucp/Devices 파일의 프로토콜 정의	539
UUCP /etc/uucp/Dialers 파일	540
/etc/uucp/Dialers 파일에서 하드웨어 흐름 제어를 사용으로 설정	543
/etc/uucp/Dialers 파일에서 패리티 설정	543
다른 기본 UUCP 구성 파일	544
UUCP /etc/uucp/Dialcodes 파일	544
UUCP /etc/uucp/Sysfiles 파일	545
UUCP /etc/uucp/Sysname 파일	546
UUCP /etc/uucp/Permissions 파일	546
UUCP 구성 항목	547
UUCP 고려 사항	547
UUCP REQUEST 옵션	548
UUCP SENDFILES 옵션	548
UUCP MYNAME 옵션	548
UUCP READ 및 WRITE 옵션	549
UUCP NOREAD 및 NOWRITE 옵션	550
UUCP CALLBACK 옵션	550
UUCP COMMANDS 옵션	550

UUCP VALIDATE 옵션 .....	552
OTHER에 대한 UUCP MACHINE 항목 .....	553
UUCP의 MACHINE 및 LOGNAME 항목 결합 .....	553
UUCP 전달 .....	554
UUCP /etc/uucp/Poll 파일 .....	554
UUCP /etc/uucp/Config 파일 .....	554
UUCP/etc/uucp/Grades 파일 .....	555
UUCP User-job-grade 필드 .....	555
UUCP System-job-grade 필드 .....	555
UUCP Job-size 필드 .....	556
UUCP Permit-type 필드 .....	556
UUCP ID-list 필드 .....	557
기타 UUCP 구성 파일 .....	557
UUCP /etc/uucp/Devconfig 파일 .....	557
UUCP /etc/uucp/Limits 파일 .....	558
UUCP remote.unknown 파일 .....	558
UUCP 관리 파일 .....	558
UUCP 오류 메시지 .....	560
UUCP ASSERT 오류 메시지 .....	560
UUCP STATUS 오류 메시지 .....	562
UUCP 숫자 오류 메시지 .....	563
<b>제6부 원격 시스템 작업 항목 .....</b>	<b>565</b>
<b>27 원격 시스템 작업(개요) .....</b>	<b>567</b>
FTP 서버란? .....	567
원격 시스템이란? .....	567
이 릴리스의 FTP 서버 정보 .....	568
표준 ProFTPD와의 차이점 .....	568
ProFTPD 구성 요소 .....	568
ProFTPD 명령 .....	568
ProFTPD 파일 .....	569
ProFTPD 사용자 .....	569

<b>28 FTP 서버 관리(작업)</b> .....	571
FTP 서버 관리(작업 맵) .....	571
FTP 서버 관리(작업) .....	572
▼ SMF를 사용하여 FTP 서버를 시작하는 방법 .....	572
▼ SMF를 사용하여 FTP 서버를 종료하는 방법 .....	572
▼ FTP 연결을 종료하는 방법 .....	572
▼ ProFTPD 구성을 변경하는 방법 .....	573
<b>29 원격 시스템 액세스(작업)</b> .....	575
원격 시스템 액세스(작업 맵) .....	575
원격 시스템에 로그인(rlogin) .....	576
원격 로그인을 위한 인증(rlogin) .....	576
원격 로그인 링크 만들기 .....	578
직접 또는 간접 원격 로그인 .....	578
원격으로 로그인한 후 수행되는 작업 .....	579
▼ .rhosts 파일을 검색하여 제거하는 방법 .....	580
원격 시스템이 작동 중인지 알아보는 방법 .....	580
원격 시스템에 로그인한 사용자를 알아보는 방법 .....	581
원격 시스템에 로그인하는 방법(rlogin) .....	582
원격 시스템에서 로그아웃하는 방법(exit) .....	582
원격 시스템에 로그인(ftp) .....	583
원격 로그인에 대한 인증(ftp) .....	583
필수 ftp 명령 .....	583
▼ 원격 시스템에 대한 ftp 연결을 여는 방법 .....	584
원격 시스템에 대한 ftp 연결을 닫는 방법 .....	585
▼ 원격 시스템에서 파일을 복사하는 방법(ftp) .....	585
▼ 원격 시스템으로 파일을 복사하는 방법(ftp) .....	587
rcp를 사용한 원격 복사 .....	589
복사 작업에 대한 보안 고려 사항 .....	589
소스 및 대상 지정 .....	590
▼ 로컬 시스템과 원격 시스템 간에 파일을 복사하는 방법(rcp) .....	591

<b>제7부 네트워크 서비스 모니터링 항목 .....</b>	<b>595</b>
<b>30 네트워크 성능 모니터링(작업) .....</b>	<b>597</b>
네트워크 성능 모니터링 .....	597
네트워크에서 호스트 응답을 확인하는 방법 .....	598
네트워크에서 호스트로 패킷을 보내는 방법 .....	598
네트워크에서 패킷을 캡처하는 방법 .....	599
네트워크 상태를 확인하는 방법 .....	599
NFS 서버 및 클라이언트 통계를 표시하는 방법 .....	602
 용어집 .....	 605
 색인 .....	 609



# 그림

---

그림 2-1	NCA 서비스를 포함하는 데이터 흐름 .....	59
그림 6-1	RDMA와 다른 프로토콜 간의 관계 .....	171
그림 6-2	서버 파일 시스템 및 클라이언트 파일 시스템 보기 .....	174
그림 6-3	svc:/system/filesystem/autofs 서비스 시작 automount .....	202
그림 6-4	마스터 맵을 통한 탐색 .....	203
그림 6-5	서버 인접도 .....	206
그림 6-6	autofs에서 이름 서비스를 사용하는 방법 .....	211
그림 7-1	SLP 기본 에이전트 및 프로세스 .....	219
그림 7-2	DA로 구현된 SLP 구조적 에이전트 및 프로세스 .....	219
그림 7-3	SLP 구현 .....	221
그림 12-1	일반적인 전자 메일 구성 .....	271
그림 13-1	로컬 메일 구성 .....	275
그림 13-2	UUCP 연결을 사용하는 로컬 메일 구성 .....	276
그림 14-1	여러 통신 프로토콜 사이의 게이트웨이 .....	328
그림 14-2	메일 프로그램의 상호 작용 .....	335
그림 15-1	PPP 링크의 각 부분 .....	379
그림 15-2	기본 아날로그 다이얼 업 PPP 링크 .....	380
그림 15-3	기본 전용 회선 구성 .....	383
그림 15-4	PPPoE 터널의 참가자 .....	387
그림 16-1	샘플 다이얼 업 링크 .....	392
그림 16-2	전용 회선 구성의 예 .....	395
그림 16-3	PAP 인증 시나리오의 예(재택 근무) .....	397
그림 16-4	CHAP 인증 시나리오의 예(개인 네트워크 호출) .....	399
그림 16-5	PPPoE 터널의 예 .....	402
그림 22-1	PAP 인증 프로세스 .....	488
그림 22-2	CHAP 인증 순서 .....	491



## 표

---

표 2-1	NCA 파일 .....	57
표 3-1	NTP 파일 .....	64
표 5-1	파일 시스템 공유 작업 맵 .....	82
표 5-2	파일 시스템 마운트 작업 맵 .....	85
표 5-3	NFS 서비스의 작업 맵 .....	91
표 5-4	WebNFS 관리 작업 맵 .....	98
표 5-5	Autofs 관리 작업 맵 .....	100
표 5-6	autofs 맵의 유형 및 해당 용도 .....	103
표 5-7	맵 유지 관리 .....	103
표 5-8	automount 명령을 실행하는 경우 .....	103
표 6-1	NFS 파일 .....	131
표 6-2	sharectl 유틸리티의 하위 명령 .....	155
표 6-3	미리 정의된 맵 변수 .....	208
표 7-1	SLP 에이전트 .....	218
표 9-1	SLP 구성 작업 .....	227
표 9-2	DA 알림 타이밍 및 검색 요청 등록 정보 .....	230
표 9-3	SLP 성능 등록 정보 .....	234
표 9-4	시간 초과 등록 정보 .....	239
표 9-5	경로가 지정되지 않은 다중 네트워크 인터페이스 구성 .....	250
표 10-1	SLP 프록시 등록 파일 설명 .....	258
표 11-1	SLP 상태 코드 .....	261
표 11-2	SLP 메시지 유형 .....	262
표 14-1	일반 sendmail 플래그 .....	316
표 14-2	맵 및 데이터베이스 유형 .....	316
표 14-3	OS 플래그 .....	317
표 14-4	이 버전의 sendmail에 사용되지 않는 일반 플래그 .....	317
표 14-5	대체 sendmail 명령 .....	318
표 14-6	구성 파일의 버전 값 .....	318

표 14-7	최상위 도메인 .....	322
표 14-8	우편함 이름 형식을 위한 규약 .....	324
표 14-9	메일 서비스에 사용되는 /etc/mail/cf 디렉토리의 내용 .....	331
표 14-10	/usr/lib 디렉토리의 내용 .....	333
표 14-11	메일 서비스에 사용되는 기타 파일 .....	334
표 14-12	TLS를 사용하여 SMTP를 실행하기 위한 구성 파일 옵션 .....	351
표 14-13	TLS를 사용하여 SMTP를 실행하기 위한 매크로 .....	353
표 14-14	TLS를 사용하여 SMTP를 실행하기 위한 규칙 세트 .....	353
표 14-15	버전 8.13의 sendmail에서 사용 가능한 명령줄 옵션 .....	354
표 14-16	버전 8.13의 sendmail에서 사용 가능한 구성 파일 옵션 .....	355
표 14-17	sendmail 버전 8.13에서 사용 가능한 FEATURE() 선언 .....	356
표 14-18	sendmail 버전 8.12의 추가 또는 제거된 명령줄 옵션 .....	359
표 14-19	PidFile 및 ProcessTitlePrefix 옵션을 위한 인수 .....	360
표 14-20	sendmail의 추가 정의된 매크로 .....	361
표 14-21	sendmail 구성 파일 작성에 사용되는 추가 매크로 .....	362
표 14-22	추가 MAX 매크로 .....	362
표 14-23	sendmail의 추가 및 개정된 m4 구성 매크로 .....	363
표 14-24	추가 및 개정된 FEATURE() 선언 .....	364
표 14-25	지원되지 않는 FEATURE() 선언 .....	366
표 14-26	추가 메일러 플러그 .....	367
표 14-27	배달 에이전트의 추가 등식 .....	368
표 14-28	토큰 비교 .....	369
표 14-29	추가 LDAP 맵 플러그 .....	370
표 14-30	첫번째 메일러 인수의 가능한 값 .....	370
표 14-31	새 규칙 세트 .....	370
표 16-1	PPP 계획 작업 맵 .....	389
표 16-2	다이얼 아웃 시스템에 대한 정보 .....	390
표 16-3	다이얼 인 서버에 대한 정보 .....	391
표 16-4	전용 회선 링크 계획 .....	394
표 16-5	인증 구성 전의 필수 조건 .....	396
표 16-6	PPPoE 클라이언트 계획 .....	400
표 16-7	PPPoE 액세스 서버 계획 .....	401
표 17-1	다이얼 업 PPP 링크 설정 작업 맵 .....	405
표 17-2	다이얼 아웃 시스템 설정 작업 맵 .....	406
표 17-3	다이얼 인 서버 설정 작업 맵 .....	412
표 18-1	전용 회선 링크 설정 작업 맵 .....	419

표 19-1	일반 PPP 인증 작업 맵 .....	425
표 19-2	PAP 인증 작업 맵(다이얼 인 서버) .....	426
표 19-3	PAP 인증 작업 맵(다이얼 아웃 시스템) .....	426
표 19-4	CHAP 인증 작업 맵(다이얼 인 서버) .....	433
표 19-5	CHAP 인증 작업 맵(다이얼 아웃 시스템) .....	434
표 20-1	PPPoE 클라이언트 설정 작업 맵 .....	439
표 20-2	PPPoE 액세스 서버 설정 작업 맵 .....	440
표 21-1	PPP 문제 해결 작업 맵 .....	447
표 21-2	PPP에 영향을 주는 일반적인 네트워크 문제 .....	453
표 21-3	PPP에 영향을 주는 일반적인 통신 문제 .....	455
표 21-4	일반적인 PPP 구성 문제 .....	456
표 21-5	일반적인 채트 스크립트 문제 .....	458
표 21-6	일반적인 전용 회선 문제 .....	463
표 21-7	일반적인 인증 문제 .....	463
표 22-1	PPP 구성 파일 및 명령 요약 .....	466
표 22-2	PPPoE 명령 및 구성 파일 .....	494
표 25-1	UUCP 관리용 작업 맵 .....	517
표 26-1	Systems 파일의 Chat-Script 필드에서 사용되는 제어 문자 .....	532
표 26-2	/etc/uucp/Devices에서 사용되는 프로토콜 .....	539
표 26-3	/etc/uucp/Dialers의 백슬래시 문자 .....	542
표 26-4	Dialcodes 파일의 항목 .....	544
표 26-5	Permit-type 필드 .....	557
표 26-6	UUCP 잠금 파일 .....	559
표 26-7	ASSERT 오류 메시지 .....	560
표 26-8	UUCP STATUS 메시지 .....	562
표 26-9	번호별 UUCP 오류 메시지 .....	563
표 27-1	ProFTPD 명령 .....	568
표 27-2	ProFTPD 파일 .....	569
표 28-1	작업 맵: FTP 서버 관리 .....	571
표 29-1	작업 맵: 원격 시스템 액세스 .....	575
표 29-2	로그인 방법과 인증 방법 간의 종속성(rlogin) .....	579
표 29-3	필수 ftp 명령 .....	583
표 29-4	디렉토리 및 파일 이름에 허용되는 구문 .....	590
표 30-1	네트워크 모니터링 명령 .....	597
표 30-2	netstat -r 명령의 출력 .....	602
표 30-3	클라이언트/서버 통계 표시 명령 .....	602

---

표 30-4	nfsstat -c 명령의 출력 .....	603
표 30-5	nfsstat -m 명령의 출력 .....	604

## 코드 예

---

예 2-1	NCA 로그 파일로 원시 장치 사용 .....	50
예 2-2	NCA 로깅에 대해 다중 파일 사용 .....	51
예 2-3	SSL 커널 프록시를 사용하여 Apache 2.0 웹 서버 구성 .....	55
예 2-4	SSL 커널 프록시를 사용하여 Sun Java System Web Server 구성 .....	56
예 2-5	SSL 커널 프록시를 사용하도록 로컬 영역에서 Apache 웹 서버 구성 .....	57
예 3-1	다른 시스템에서 시간 및 날짜 동기화 .....	64
예 5-1	클라이언트 <code>vfstab</code> 파일의 항목 .....	86
예 5-2	파일 시스템을 마운트한 후 미리 마운트 사용 .....	87
예 5-3	기존 참조 수정 .....	115
예 6-1	파일 시스템 마운트 해제 .....	154
예 6-2	<code>umount</code> 에서 옵션 사용 .....	154
예 6-3	샘플 <code>/etc/auto_master</code> 파일 .....	195
예 9-1	DA 서버로 작동하도록 <code>slpd</code> 설정 .....	229
예 13-1	<code>submit.cf</code> 의 자동 재작성 설정 .....	288
예 13-2	Received: 메일 헤더 .....	293
예 13-3	주소 테스트 모드 출력 .....	309
예 21-1	제대로 작동하는 다이얼 업 링크의 출력 .....	449
예 21-2	제대로 작동하는 전용 회선 링크의 출력 .....	449
예 22-1	인라인 채트스크립트 .....	484
예 22-2	기본적인 <code>/etc/ppp/pppoe</code> 파일 .....	498
예 22-3	액세스 서버를 위한 <code>/etc/ppp/pppoe</code> 파일 .....	500
예 22-4	액세스 서버를 위한 <code>/etc/ppp/options</code> 파일 .....	501
예 22-5	액세스 서버를 위한 <code>/etc/hosts</code> 파일 .....	501
예 22-6	액세스 서버를 위한 <code>/etc/ppp/pap-secrets</code> 파일 .....	501
예 22-7	액세스 서버를 위한 <code>/etc/ppp/chap-secrets</code> 파일 .....	501
예 22-8	원격 액세스 서버를 정의하기 위한 <code>/etc/ppp/peers/peer-name</code> .....	503
예 26-1	<code>/etc/uucp/Systems</code> 의 항목 .....	528
예 26-2	Type 필드의 키워드 .....	530

예 26-3	Speed 필드의 항목 .....	530
예 26-4	Phone 필드의 항목 .....	530
예 26-5	Devices 파일 및 Systems 파일의 Type 필드 비교 .....	535
예 26-6	Devices 파일의 Class 필드 .....	536
예 26-7	직접 연결 모뎀의 Dialers 필드 .....	537
예 26-8	동일한 포트 선택기에 있는 컴퓨터의 UUCP Dialers 필드 .....	538
예 26-9	포트 선택기에 연결된 모뎀의 UUCP Dialers 필드 .....	538
예 26-10	/etc/uucp/Dialers 파일의 항목 .....	540
예 26-11	/etc/uucp/Dialers의 인용구 .....	541
예 28-1	가상 호스트를 위한 ProFTPD 구성 파일 변경 사항 .....	573
예 28-2	익명 액세스를 위한 ProFTPD 구성 파일 변경 사항 .....	574
예 29-1	.rhosts 파일을 검색하여 제거 .....	580
예 29-2	원격 시스템에 로그인한 사용자 알아보기 .....	581
예 29-3	원격 시스템에 로그인(rlogin) .....	582
예 29-4	원격 시스템에서 로그아웃(exit) .....	583
예 29-5	원격 시스템에 대한 ftp 연결 열기 .....	585
예 29-6	원격 시스템에서 파일 복사(ftp) .....	586
예 29-7	원격 시스템으로 파일 복사(ftp) .....	588
예 29-8	rcp를 사용하여 원격 파일을 로컬 시스템으로 복사 .....	592
예 29-9	rlogin 및 rcp를 사용하여 원격 파일을 로컬 시스템으로 복사 .....	592
예 29-10	rcp를 사용하여 로컬 파일을 원격 시스템으로 복사 .....	592
예 29-11	rlogin 및 rcp를 사용하여 로컬 파일을 원격 시스템으로 복사 .....	593
예 30-1	네트워크에서 호스트 응답 확인 .....	598
예 30-2	네트워크에서 호스트로 패킷 보내기 .....	599



# 머리말

---

**System Administration Guide: Network Services**는 Oracle Solaris 시스템 관리 정보의 상당 부분을 다루는 여러 권으로 구성된 설명서의 일부입니다. 이 설명서에서는 사용자가 이미 Oracle Solaris 운영 체제를 설치했으며 사용하려는 모든 네트워킹 소프트웨어를 설정했다고 가정합니다. 이 운영 체제는 Oracle Solaris 제품군에 속하며 많은 기능을 포함하고 있습니다.

---

주 - 본 Oracle Solaris 릴리스는 SPARC 및 x86 제품군의 프로세서 구조를 사용하는 시스템을 지원합니다. 지원되는 시스템은 **Oracle Solaris OS: Hardware Compatibility Lists**를 참조하십시오. 이 설명서에서는 플랫폼 유형에 따른 구현 차이가 있는 경우 이에 대하여 설명합니다.

지원되는 시스템은 **Oracle Solaris OS: Hardware Compatibility Lists**를 참조하십시오.

---

## 본 설명서의 대상

이 설명서는 Oracle Solaris 릴리스를 실행하는 하나 이상의 시스템을 관리하는 모든 사용자를 대상으로 합니다. 이 설명서를 사용하려면 UNIX 시스템 관리 경험이 1~2년 정도 있어야 합니다. UNIX 시스템 관리 교육 과정에 참석하는 것도 도움이 될 수 있습니다.

## 시스템 관리 설명서의 구성

시스템 관리 설명서에서 설명하는 항목 목록은 다음과 같습니다.

설명서 제목	내용
<b>SPARC 플랫폼에서 Oracle Solaris 부트 및 종료</b>	시스템 부트 및 종료, 부트 서비스 관리, 부트 동작 수정, ZFS에서 부트, 부트 아카이브 관리, SPARC 플랫폼에서의 부트 문제 해결
<b>x86 플랫폼에서 Oracle Solaris 부트 및 종료</b>	시스템 부트 및 종료, 부트 서비스 관리, 부트 동작 수정, ZFS에서 부트, 부트 아카이브 관리, x86 플랫폼에서의 부트 문제 해결

설명서 제목	내용
<b>Oracle Solaris 관리: 일반 작업</b>	Oracle Solaris 명령 사용, 시스템 부트 및 종료, 사용자 계정 및 그룹 관리, 서비스, 하드웨어 오류, 시스템 정보, 시스템 리소스 및 시스템 성능 관리, 소프트웨어, 인쇄, 콘솔 및 터미널 관리, 시스템 및 소프트웨어 문제 해결
<b>Oracle Solaris 관리: 장치 및 파일 시스템</b>	이동식 매체, 디스크 및 장치, 파일 시스템, 데이터 백업 및 복원
<b>Oracle Solaris 관리: IP 서비스</b>	TCP/IP 네트워크 관리, IPv4 및 IPv6 주소 관리, DHCP, IPsec, IKE, IP 필터, 이동 IP, IPQoS
<b>Oracle Solaris Administration: Naming and Directory Services</b>	DNS, NIS 및 LDAP 이름 지정 및 디렉토리 서비스(NIS에서 LDAP로의 전환 포함)
<b>Oracle Solaris 관리: 네트워크 인터페이스 및 네트워크 가상화</b>	네트워킹 스택, NIC 드라이버 등록 정보 구성, NWAM 구성, 수동 네트워크 인터페이스 구성, VLAN 및 링크 집계 관리, IPMP(IP 네트워킹 다중 경로), WiFi 무선 네트워킹 구성, VNIC(가상 NIC), 네트워크 자원 관리
<b>Oracle Solaris 관리: 네트워크 서비스</b>	웹 캐시 서버, 시간 관련 서비스, 네트워크 파일 시스템(NFS 및 Autofs), 메일, SLP, PPP
<b>Oracle Solaris 관리: Oracle Solaris Zones, Oracle Solaris 10 Zones 및 리소스 관리</b>	응용 프로그램이 사용 가능한 시스템 리소스를 사용하는 방법을 제어할 수 있는 자원 관리 기능, 운영 체제 서비스를 가상화하여 실행 중인 응용 프로그램을 위한 격리된 환경을 만드는 Oracle Solaris Zones 소프트웨어 영역 분할 기술, Oracle Solaris 11 Express 커널에서 실행 중인 Oracle Solaris 10 환경을 호스트하는 Oracle Solaris 10 Zones
<b>Oracle Solaris 관리: 보안 서비스</b>	감사, 장치 관리, 파일 보안, BART, Kerberos 서비스, PAM, 암호화 프레임워크, 키 관리, 권한, RBAC, SASL, 보안 셸, 바이러스 스캐닝
<b>Oracle Solaris Administration: SMB and Windows Interoperability</b>	Oracle Solaris 시스템을 구성하여 SMB 클라이언트에 사용 가능한 SMB 공유를 만들 수 있도록 해주는 SMB 서비스, SMB 공유에 액세스할 수 있도록 해주는 SMB 클라이언트, Oracle Solaris 시스템과 Windows 시스템 사이에 사용자 및 그룹 ID를 매핑할 수 있도록 해주는 고유 ID 매핑 서비스
<b>Oracle Solaris 관리: ZFS 파일 시스템</b>	ZFS 저장소 풀 및 파일 시스템 생성/관리, 스냅샷, 복제, 백업, 액세스 제어 목록(ACL)을 통한 ZFS 파일 보호, 영역이 설치된 Solaris 시스템에서 ZFS 사용, 에뮬레이트된 볼륨, 문제 해결 및 데이터 복구
<b>Trusted Extensions 구성 및 관리</b>	Trusted Extensions에만 적용되는 시스템 설치, 구성 및 관리
<b>Oracle Solaris 11 보안 지침</b>	Oracle Solaris 시스템 보안 및 영역, ZFS, Trusted Extensions 등과 같은 보안 기능에 대한 사용 시나리오

설명서 제목	내용
<b>Oracle Solaris 10에서 Oracle Solaris 11로 전환</b>	설치, 장치, 디스크, 파일 시스템 관리, 소프트웨어 관리, 네트워킹, 시스템 관리, 보안, 가상화, 데스크탑 기능, 사용자 계정 관리, 사용자 환경, 애플리케이션 볼륨, 문제 해결 및 데이터 복구 영역에 걸쳐 Oracle Solaris 10에서 Oracle Solaris 11로의 전환에 대한 시스템 관리 정보 및 예 제공

## 관련 문서

본 설명서에서 참조되는 관련 설명서 목록은 다음과 같습니다.

- **System Administration Guide: Advanced Administration**
- **Oracle Solaris 관리: 일반 작업**
- **Oracle Solaris 관리: IP 서비스**
- **Oracle Solaris Administration: Naming and Directory Services**
- **Oracle Solaris 관리: Oracle Solaris Zones, Oracle Solaris 10 Zones 및 리소스 관리**
- **Oracle Solaris 관리: 보안 서비스**
- Anderson, Bart, Bryan Costales 및 Harry Henderson. **UNIX Communications**. Howard W. Sams & Company, 1987.
- Costales, Bryan. **sendmail, Third Edition**. O'Reilly & Associates, Inc., 2002.
- Frey, Donnalyn 및 Rick Adams. **!%@:: A Directory of Electronic Mail Addressing and Networks**. O'Reilly & Associates, Inc., 1993.
- Krol, Ed. **The Whole Internet User's Guide and Catalog**. O'Reilly & Associates, Inc., 1993.
- O'Reilly, Tim 및 Grace Todino. **Managing UUCP and Usenet**. O'Reilly & Associates, Inc., 1992.

## Oracle Support에 액세스

Oracle 고객은 My Oracle Support를 통해 온라인 지원에 액세스할 수 있습니다. 자세한 내용은 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>를 참조하거나, 청각 장애가 있는 경우 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>를 방문하십시오.

# 활자체 규약

다음 표는 이 책에서 사용되는 활자체 규약에 대해 설명합니다.

표 P-1 활자체 규약

활자체	의미	예
AaBbCc123	명령 및 파일, 디렉토리 이름; 컴퓨터 화면에 출력되는 내용입니다.	.login 파일을 편집하십시오.  모든 파일 목록을 보려면 ls -a 명령을 사용하십시오.  machine_name% you have mail.
AaBbCc123	사용자가 입력하는 내용으로 컴퓨터 화면의 출력 내용과 대조됩니다.	machine_name% su  Password:
aabbcc123	새로 나오는 용어, 강조 표시할 용어입니다. 명령줄 변수를 실제 이름이나 값으로 바꾸십시오.	파일을 제거하는 명령은 rm filename입니다.
AaBbCc123	책 제목, 장, 절	사용자 설명서의 6장을 읽으십시오.  캐시는 로컬로 저장된 복사본입니다.  파일을 저장하면 안 됩니다.  주: 일부 강조된 항목은 온라인에서 굵은체로 나타납니다.

# 명령 예의 셀 프롬프트

다음 표에는 Oracle Solaris OS에 포함된 셀의 기본 UNIX 시스템 프롬프트 및 슈퍼유저 프롬프트가 나와 있습니다. 명령 예제에 표시된 기본 시스템 프롬프트는 Oracle Solaris 릴리스에 따라 다릅니다.

표 P-2 셀 프롬프트

셀	프롬프트
Bash 셸, Korn 셸 및 Bourne 셸	\$
슈퍼유저용 Bash 셸, Korn 셸 및 Bourne 셸	#
C 셸	machine_name%
슈퍼유저용 C 셸	machine_name#





## 제 1 부

# 네트워크 서비스 항목

이 절에서는 설명서에 대한 개요뿐만 아니라 NCA(네트워크 캐시 및 가속기) 및 NTP(네트워크 시간 프로토콜) 서비스에 대한 개요, 작업 및 참조 정보를 제공합니다.





## 네트워크 서비스(개요)

---

이 장에서는 본 설명서에서 다루는 주요 항목 목록을 제공합니다. 여기에는 이 릴리스에 포함된 PERL 서비스의 설명도 포함됩니다.

- 41 페이지 “Oracle Solaris 11 릴리스의 항목”
- 42 페이지 “Perl 5”

## Oracle Solaris 11 릴리스의 항목

이 설명서에서 다음 서비스 또는 유틸리티를 다룹니다.

### 42 페이지 “Perl 5”

Perl(Practical Extraction and Report Language)은 시스템 관리 작업에 도움이 되는 스크립트를 생성하는 데 사용할 수 있는 도구입니다.

### 2 장, “웹 캐시 서버 관리”

NCA에서는 웹 페이지를 캐싱하여 향상된 웹 서버 성능을 제공합니다.

### 3 장, “시간 관련 서비스”

NTP 및 시간 관련 유틸리티는 여러 시스템의 시간을 동기화하는 데 사용할 수 있습니다.

### 4 장, “네트워크 파일 시스템 관리(개요)”

NFS는 원격 호스트에서 파일 시스템에 액세스하는 기능을 제공하는 프로토콜입니다.

### 7 장, “SLP(개요)”

SLP는 동적 서비스 검색 프로토콜입니다.

### 12 장, “메일 서비스(개요)”

메일 서비스는 필요한 모든 네트워크로 메시지의 경로를 지정하는 동안 한 명 이상의 사람에게 메시지를 보낼 수 있습니다.

15 장, “Solaris PPP 4.0(개요)”

PPP는 원격 호스트 사이에 지점 간 링크를 제공하는 프로토콜입니다.

24 장, “UUCP(개요)”

UUCP는 호스트가 파일을 교환할 수 있도록 합니다.

27 장, “원격 시스템 작업(개요)”

이러한 명령은 원격 시스템에 있는 파일에 액세스하는 데 사용됩니다. 이러한 명령에는 ftp, rlogin 및 rcp가 포함됩니다.

## Perl 5

이 Oracle Solaris 릴리스는 일반적으로 무료로 사용할 수 있는 소프트웨어인 Perl(Practical Extraction and Report Language) 5.8.4 버전과 5.12 버전(강력한 일반용 프로그래밍 언어)을 포함하고 있습니다. Perl은 탁월한 프로세스, 파일 및 텍스트 조작 기능을 갖추고 있어 복잡한 시스템 관리 작업을 위한 표준 개발 도구로 자리매김했습니다.

Perl 5는 특정 작업을 위한 새 기능을 추가할 수 있는 동적 로드 가능 모듈 프레임워크를 포함하고 있습니다. <http://www.cpan.org>의 CPAN(Comprehensive Perl Archive Network)에서 많은 모듈을 자유롭게 사용할 수 있습니다. gcc 명령을 사용하여 CPAN을 통해 추가 기능 모듈을 구축 및 설치하려면 /usr/perl5/5.8.4/bin/perlgcc 또는 /usr/perl5/5.12/bin/perlgcc 스크립트를 사용합니다. 자세한 내용은 5.8.4 배포의 perlgcc(1) 매뉴얼 페이지를 참조하십시오.

## Perl 문서 액세스

Perl에 대한 여러 소스 정보가 이 Oracle Solaris 릴리스에 포함되어 있습니다. 이러한 두 가지 방식을 사용하여 동일한 정보를 사용할 수 있습니다.

/usr/perl5/man을 사용자의 MANPATH 환경 변수에 추가하여 매뉴얼 페이지에 액세스할 수 있습니다. 다음 예에서는 Perl 개요를 표시합니다.

```
% setenv MANPATH ${MANPATH}:/usr/perl5/man
% man perl
```

perldoc 유틸리티를 사용하여 추가 문서에 액세스할 수 있습니다. 다음 예에서는 동일한 개요 정보를 표시합니다.

```
% /usr/perl5/bin/perldoc perl
```

perl 개요 페이지에 이 릴리스에 포함된 모든 설명서가 나와 있습니다.

## Perl 호환성 문제

일반적으로 Perl의 5.12 버전은 이전 버전과 호환됩니다. 작동하기 위해 스크립트를 재구성하거나 재컴파일할 필요가 없습니다. 그러나 모든 XSUB-기반(.xs) 모듈은 재컴파일 및 재설치가 필요합니다.

## Perl Oracle Solaris 버전의 변경 사항

Perl의 Oracle Solaris 버전은 시스템 메모리 할당자, 64비트 정수 및 대용량 파일 지원을 포함하도록 컴파일되었습니다. 또한 적절한 패치가 적용되었습니다. 모든 구성 정보의 전체 목록을 보려면 다음 명령의 결과를 검토하십시오.

```
% /usr/perl5/bin/perlbug -dv
---
Flags:
    category=
    severity=
---
Site configuration information for perl v5.12.4:
.
.
```

perl -V를 사용하면 보다 간단한 목록을 볼 수 있습니다.



## 웹 캐시 서버 관리

---

이 장에서는 Oracle Solaris 11 릴리스에서의 NCA(네트워크 캐시 및 가속기)에 대한 개요를 제공합니다. NCA 사용 절차 및 NCA에 대한 참조 자료가 포함됩니다. 또한 SSL(Secure Sockets Layer) 사용 방법에 대한 소개 및 SSL 패킷 프로세싱 성능 향상을 위해 SSL 커널 프록시를 사용하는 절차도 추가됩니다.

- 45 페이지 “네트워크 캐시 및 가속기(개요)”
- 47 페이지 “웹 캐시 서버 관리(작업 맵)”
- 48 페이지 “웹 페이지의 캐시 관리(작업)”
- 57 페이지 “웹 페이지 캐싱(참조)”

### 네트워크 캐시 및 가속기(개요)

NCA(네트워크 캐시 및 가속기)는 HTTP 요청 중에 액세스된 웹 페이지의 커널 내 캐시를 유지 관리하여 웹 서버 성능을 향상시킵니다. 이 커널 내 캐시는 웹 서버에서 일반적으로 처리하는 HTTP 요청에 대한 성능을 크게 향상시키기 위해 시스템 메모리를 사용합니다. HTTP 요청에 대해 웹 페이지를 보유하도록 시스템 메모리를 사용하면 커널과 웹 서버 간의 오버헤드를 줄여 웹 서버 성능이 향상됩니다. NCA에서는 수정을 최소화한 상태로 웹 서버와 통신할 수 있도록 소켓 인터페이스를 제공합니다.

요청된 페이지가 커널 내 캐시에서 검색되는 경우(캐시 적중) 성능은 비약적으로 향상됩니다. 요청된 페이지가 캐시에 없고(캐시 비적중) 웹 서버에서 검색되어야 하는 경우에도 성능은 크게 향상됩니다.

이 제품은 전용 웹 서버에서 실행해야 합니다. NCA를 실행하는 서버에서 다른 큰 프로세스를 실행하는 경우 문제가 발생할 수 있습니다.

NCA에서는 해당 NCA에서 기록하는 모든 캐시 적중의 로깅 지원을 제공합니다. 이 로고는 성능을 향상시키기 위해 이진 형식으로 저장됩니다. `ncab2clf` 명령은 이진 형식의 로그를 CLF(일반 로그 형식)로 변환하는 데 사용할 수 있습니다.

Oracle Solaris 릴리스에는 다음과 같은 향상된 기능이 있습니다.

- 소켓 인터페이스
- AF\_NCA 지원을 제공하는 벡터식 `sendfile` 지원. 자세한 내용은 [sendfilev\(3EXT\)](#) 매뉴얼 페이지를 참조하십시오.
- 선택한 날짜(-s) 앞의 레코드를 건너뛰는 기능과 지정된 레코드 수(-n)를 처리하는 기능을 지원하는 `ncab2clf` 명령에 대한 새로운 옵션
- `ncalogd.conf`의 `logd_path_name`에서 원시 장치, 파일 또는 이 둘의 조합 중 하나를 지정할 수 있음
- 웹 서버에서 다중 AF\_NCA 소켓을 열 수 있도록 지원. 다중 소켓을 사용하여 하나의 서버에서 여러 웹 서버를 실행할 수 있습니다.
- `/etc/nca/ncaport.conf`라는 새 구성 파일. 이 파일은 IP 주소 및 NCA를 사용하는 포트를 관리하는 데 사용할 수 있습니다. 웹 서버에서 AF\_NCA 소켓의 고유 지원을 제공하지 않을 수 있습니다. 서버에서 이를 지원하지 않는 경우 이 파일과 NCA 소켓 유틸리티 라이브러리를 사용하여 AF\_INET 소켓을 AF\_NCA 소켓으로 변환합니다.

## 웹 서버에서 Secure Sockets Layer 프로토콜 사용

Apache 2.0 및 Sun Java System Web Server는 SSL(Secure Sockets Layer) 프로토콜을 사용하도록 구성될 수 있습니다. 프로토콜에서는 기밀성, 메시지 무결성 및 두 응용 프로그램 간의 끝점 인증을 제공합니다. 커널이 SSL 트래픽 속도를 높이기 위해 변경되었습니다.

SSL 커널 프록시는 SSL 프로토콜의 서버측을 구현합니다. 프록시에서는 사용자 레벨 SSL 라이브러리를 사용하는 응용 프로그램을 통해 웹 서버와 같은 향상된 SSL 성능을 제공합니다. 성능 향상은 응용 프로그램의 작업 부하에 따라 35% 이상이 될 수 있습니다.

SSL 커널 프록시에서는 SSL 3.0 및 TLS 1.0 프로토콜뿐 아니라 가장 일반적인 암호 스위트도 지원합니다. [ksslcfg\(1M\)](#) 매뉴얼 페이지에서 전체 목록을 참조하십시오. 지원되지 않는 모든 암호 스위트에 대한 사용자 레벨 SSL 서버를 폴백하도록 프록시를 구성할 수 있습니다.

다음 절차에서는 SSL 커널 프록시를 사용하여 서버를 구성하는 방법을 보여줍니다.

- 53 페이지 “SSL 커널 프록시를 사용하여 Apache 2.0 웹 서버를 구성하는 방법”
- 55 페이지 “SSL 커널 프록시를 사용하여 Sun Java System Web Server를 구성하는 방법”
- 57 페이지 “영역에서 SSL 커널 프록시 사용”

## 웹 캐시 서버 관리(작업 맵)

다음 표에서는 NCA 또는 SSL 사용에 필요한 절차에 대해 설명합니다.

작업	설명	수행 방법
NCA 계획	NCA를 사용으로 설정하기 전에 해결해야 할 문제 목록입니다.	47 페이지 “NCA 계획”
NCA를 사용으로 설정	웹 서버의 웹 페이지 커널 내 캐시를 사용으로 설정하는 단계입니다.	48 페이지 “웹 페이지의 캐시를 사용으로 설정하는 방법”
NCA 사용 안함으로 설정	웹 서버의 웹 페이지 커널 내 캐시를 사용 안함으로 설정하는 단계입니다.	51 페이지 “웹 페이지의 캐시를 사용 안함으로 설정하는 방법”
NCA 로깅 관리	NCA 로깅 프로세스를 사용으로 설정 또는 사용 안함으로 설정하는 단계입니다.	52 페이지 “NCA 로깅을 사용으로 설정 또는 사용 안함으로 설정하는 방법”
NCA 소켓 라이브러리 로드	AF_NCA 소켓이 지원되지 않는 경우 NCA를 사용하는 단계입니다.	52 페이지 “NCA용 소켓 유틸리티 라이브러리를 로드하는 방법”
Apache 2.0 웹 서버에서 SSL 커널 프록시 사용	SSL 패킷 프로세싱을 향상시키기 위해 웹 서버에서 SSL 커널 프록시를 사용하는 단계입니다.	53 페이지 “SSL 커널 프록시를 사용하여 Apache 2.0 웹 서버를 구성하는 방법”
Sun Java System Web Server에서 SSL 커널 프록시 사용	SSL 패킷 프로세싱을 향상시키기 위해 웹 서버에서 SSL 커널 프록시를 사용하는 단계입니다.	55 페이지 “SSL 커널 프록시를 사용하여 Sun Java System Web Server를 구성하는 방법”
로컬 영역에 있는 웹 서버에서 SSL 커널 프록시 사용	로컬 영역에 있는 웹 서버에서 SSL 커널 프록시를 사용하는 단계입니다.	57 페이지 “영역에서 SSL 커널 프록시 사용”

## NCA 계획

다음 절에서는 NCA 서비스를 시작하기 전에 해결해야 하는 문제를 다룹니다.

### NCA 시스템 요구 사항

NCA를 지원하려면 다음 요구 사항을 충족해야 합니다.

- 256MB RAM이 설치되어 있어야 합니다.
- Oracle Solaris 릴리스가 설치되어 있어야 합니다.
- NCA에 대해 고유 지원을 제공하는 웹 서버 또는 NCA에 대해 소켓 유틸리티 라이브러리를 사용하도록 수정된 시작 스크립트가 있는 웹 서버를 지원해야 합니다.
  - Apache 웹 서버(Oracle Solaris 릴리스와 함께 제공됨)

- Sun Java System Web Server
- Zeus Technology(<http://www.zeus.com>)에서 사용 가능한 Zeus 웹 서버

이 제품은 전용 웹 서버에서 실행해야 합니다. NCA를 실행하는 서버에서 다른 큰 프로세스를 실행하면 문제가 발생할 수 있습니다.

## NCA 로깅

웹 작업을 기록하도록 NCA 서비스를 구성할 수 있습니다. 일반적으로 웹 서버 로깅이 사용으로 설정된 경우 NCA 로깅을 사용으로 설정할 수 있습니다.

## 도어 서버의 데몬 지원에 대한 라이브러리 삽입

많은 웹 서버에서 AF\_INET 소켓을 사용합니다. 기본적으로 NCA는 AF\_NCA 소켓을 사용합니다. 이 문제를 수정하기 위해 삽입 라이브러리가 제공됩니다. 표준 소켓 라이브러리(libsocket.so) 앞에 새 라이브러리가 로드됩니다. 새 라이브러리(ncad\_addr.so)가 라이브러리 호출 bind()에 삽입됩니다. /etc/nca/ncakmod.conf에서 상태가 사용으로 설정되었다고 가정합니다. Solaris 9 및 Solaris 10 릴리스에 포함된 Apache 버전은 이미 이 라이브러리를 호출하도록 설정되었습니다. IWS 또는 Netscape 서버를 사용하는 경우 새 라이브러리를 사용하려면 52 페이지 “NCA용 소켓 유틸리티 라이브러리를 로드하는 방법”을 참조하십시오.

## 여러 인스턴스 지원

NCA가 설치된 시스템은 종종 웹 서버의 여러 인스턴스를 실행해야 합니다. 예를 들어 단일 서버가 외부 액세스를 위한 웹 서버뿐 아니라 웹 관리 서버도 지원해야 할 수 있습니다. 이러한 서버를 분리하려면 별도의 포트를 사용하도록 각 서버를 구성합니다.

## 웹 페이지의 캐시 관리(작업)

다음 절에서는 서비스의 일부를 사용으로 설정 또는 사용 안함으로 설정하기 위한 절차를 다룹니다.

### ▼ 웹 페이지의 캐시를 사용으로 설정하는 방법

#### 1 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스**의 “관리 권한을 얻는 방법”을 참조하십시오.



## 2 인터페이스를 등록합니다.

/etc/nca/nca.if 파일에서 각 물리적 인터페이스의 이름을 입력합니다. 자세한 내용은 [nca.if\(4\)](#) 매뉴얼 페이지를 참조하십시오.

```
# cat /etc/nca/nca.if
hme0
hme1
```

각 인터페이스에는 `hostname.interface-name` 파일 및 `hostname.interface-name`의 내용에 대한 /etc/hosts 파일의 항목이 있어야 합니다. 모든 인터페이스에서 NCA 기능을 시작하려면 nca.if 파일에 별표(\*)를 사용합니다.

## 3 ncakmod 커널 모듈을 사용으로 설정합니다.

/etc/nca/ncakmod.conf에서 status 항목을 enabled로 변경합니다.

```
# cat /etc/nca/ncakmod.conf
#
# NCA Kernel Module Configuration File
#
status=enabled
httpd_door_path=/system/volatile/nca_httpd_1.door
nca_active=disabled
```

자세한 내용은 [ncakmod.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오.

## 4 (옵션) NCA 로깅을 사용으로 설정합니다.

/etc/nca/ncalogd.conf에서 status 항목을 enabled로 변경합니다.

```
# cat /etc/nca/ncalogd.conf
#
# NCA Logging Configuration File
#
status=enabled
logd_path_name="/var/nca/log"
logd_file_size=1000000
```

logd\_path\_name 항목으로 표시된 경로를 변경하여 로그 파일의 위치를 변경할 수 있습니다. 로그 파일은 원시 장치 또는 파일이 될 수 있습니다. NCA 로그 파일 경로의 샘플에 대한 다음 예를 참조하십시오. 구성 파일에 대한 자세한 내용은 [ncalogd.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오.

## 5 (옵션) 여러 인스턴스 지원에 대한 포트를 정의합니다.

/etc/nca/ncaport.conf 파일에서 포트 번호를 추가합니다. 이 항목은 NCA가 구성된 모든 IP 주소에서 포트 80을 모니터링하도록 합니다.

```
# cat /etc/nca/ncaport.conf
#
# NCA Kernel Module Port Configuration File
#
.
.
ncaport=*/80
```

## 6 x86만 해당: 가상 메모리 크기를 증가시킵니다.

eeeprom 명령을 사용하여 시스템의 kernelbase를 설정합니다.

```
# eeeprom kernelbase=0x90000000
# eeeprom kernelbase
kernelbase=0x90000000
```

두번째 명령은 매개변수가 설정되었는지 확인합니다.

---

주 - kernelbase를 설정하여 사용자 프로세스에서 사용할 수 있는 가상 메모리의 양을 3GB 미만으로 줄입니다. 이 제한 사항은 시스템이 ABI와 호환되지 않는다는 것을 의미합니다. 시스템을 부트할 때 콘솔에서 비호환에 대한 경고 메시지가 표시됩니다. 대부분의 프로그램에서는 실제로 가상 주소 공간을 위해 3GB 전체가 필요하지 않습니다. 3GB 이상이 필요한 프로그램의 경우 NCA가 사용으로 설정되지 않은 시스템에서 프로그램을 실행해야 합니다.

---

## 7 서버를 재부트합니다.

### 예 2-1 NCA 로그 파일로 원시 장치 사용

ncalogd.conf의 logd\_path\_name 문자열은 NCA 로그 파일을 저장하는 위치로 원시 장치를 정의할 수 있습니다. 원시 장치 사용의 이점은 원시 장치에 액세스하는 데 오버헤드가 적게 발생하기 때문에 서비스를 더 빨리 실행할 수 있다는 것입니다.

NCA 서비스는 파일에 나열된 모든 원시 장치를 테스트하여 해당 위치에 파일 시스템이 있는지 확인합니다. 이 테스트는 실수로 활성 파일 시스템을 덮어 쓰지 않도록 합니다.

테스트에서 파일 시스템을 찾지 않도록 하려면 다음 명령을 실행하십시오. 이 명령은 파일 시스템으로 구성된 적이 있는 모든 디스크 분할에서 파일 시스템의 일부를 삭제합니다. 이 예에서 /dev/rdsk/c0t0d0s7은 이전 파일 시스템을 가지고 있는 원시 장치입니다.

```
# dd if=/dev/zero of=/dev/rdsk/c0t0d0s7 bs=1024 count=1
```

dd 명령을 실행한 후에 원시 장치를 ncalogd.conf 파일에 추가할 수 있습니다.

```
# cat /etc/nca/ncalogd.conf
#
# NCA Logging Configuration File
#
status=enabled
logd_path_name="/dev/rdsk/c0t0d0s7"
logd_file_size=1000000
```

## 예 2-2 NCA 로깅에 대해 다중 파일 사용

ncalogd.conf의 logd\_path\_name 문자열은 NCA 로그 파일을 저장하는 위치로 다중 대상을 정의할 수 있습니다. 첫번째 파일이 꽉차면 두번째 파일을 사용합니다. 다음 예에서는 먼저 /var/nca/log 파일에 쓴 다음 원시 분할을 사용하도록 선택하는 방법을 보여줍니다.

```
# cat /etc/nca/ncalogd.conf
#
# NCA Logging Configuration File
#
status=enabled
logd_path_name="/var/nca/log /dev/rdsdsk/c0t0d0s7"
logd_file_size=1000000
```

## ▼ 웹 페이지의 캐시를 사용 안함으로 설정하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 ncakmod 커널 모듈을 사용 안함으로 설정합니다.

/etc/nca/ncakmod.conf에서 status 항목을 disabled로 변경합니다.

```
# cat /etc/nca/ncakmod.conf
# NCA Kernel Module Configuration File
#
status=disabled
httpd_door_path=/system/volatile/nca_httpd_1.door
nca_active=disabled
```

자세한 내용은 [ncakmod.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오.

### 3 NCA 로깅을 사용 안함으로 설정합니다.

/etc/nca/ncalogd.conf에서 status 항목을 disabled로 변경합니다.

```
# cat /etc/nca/ncalogd.conf
#
# NCA Logging Configuration File
#
status=disabled
logd_path_name="/var/nca/log"
logd_file_size=1000000
```

자세한 내용은 [ncalogd.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오.

### 4 서버를 재부트합니다.

## ▼ NCA 로깅을 사용으로 설정 또는 사용 안함으로 설정하는 방법

NCA를 사용으로 설정한 후에 필요에 따라 NCA 로깅을 켜거나 끌 수 있습니다. 자세한 내용은 48 페이지 “웹 페이지의 캐시를 사용으로 설정하는 방법”을 참조하십시오.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 NCA 로깅을 변경합니다.

영구적으로 로깅을 사용 안함으로 설정하려면 `/etc/nca/ncalogd.conf`에서 상태를 `disabled`로 변경하고 시스템을 재부트해야 합니다. 자세한 내용은 [ncalogd.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오.

#### a. 로깅 중지

```
# /etc/init.d/ncalogd stop
```

#### b. 로깅 시작

```
# /etc/init.d/ncalogd start
```

## NCA용 소켓 유틸리티 라이브러리를 로드하는 방법

웹 서버에서 AF\_NCA 소켓의 고유 지원을 제공하지 않는 경우에만 다음 프로세스를 따르십시오.

웹 서버의 시작스크립트에서 라이브러리가 미리 로드되도록 하는 행을 추가합니다. 해당 행은 다음과 같습니다.

```
LD_PRELOAD=/usr/lib/ncad_addr.so /usr/bin/httpd
```

## ▼ 새 포트를 NCA 서비스에 추가하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

## 2 새 포트를 추가합니다.

새 포트 항목을 `/etc/nca/ncaport.conf`에 추가합니다. 이 예에서는 IP 주소 `192.168.84.71`에서 포트 `8888`을 추가합니다. 자세한 내용은 `ncaport.conf(4)`를 참조하십시오.

```
# cat /etc/nca/ncaport.conf
#
# NCA Kernel Module Port Configuration File
#
.
.
ncaport=*/80
ncaport=192.168.84.71/8888
```

## 3 새 웹 인스턴스를 시작합니다.

NCA 포트 구성을 포함하는 파일에 주소가 있어야 NCA에 대한 주소를 사용할 수 있습니다. 웹 서버가 실행 중인 경우 새 주소를 정의한 후 웹 서버를 다시 시작해야 합니다.

# ▼ SSL 커널 프록시를 사용하여 Apache 2.0 웹 서버를 구성하는 방법

이 절차는 Apache 2.0 웹 서버에서 SSL 패킷 프로세스의 성능 향상을 위해 사용합니다.

**시작하기 전에** 다음 절차에서는 설치 및 구성된 Apache 2.0 웹 서버가 필요합니다. 이 릴리스에는 Apache 2.0 웹 서버가 포함되어 있습니다.

SSL 커널 프록시를 사용하려면 단일 파일에 서버 개인 키와 서버 인증서가 있어야 합니다. `ssl.conf` 파일에서 `SSLCertificateFile` 매개변수만 지정한 경우 커널 SSL에 대해 지정한 파일을 직접 사용할 수 있습니다. `SSLCertificateKeyFile` 매개변수도 지정한 경우 인증서 파일과 개인 키 파일을 결합해야 합니다. 인증서와 키 파일을 결합하는 방법 중 하나는 다음 명령을 실행하는 것입니다.

```
# cat cert.pem key.pem >cert-and-key.pem
```

## 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오. `ksslcfg` 명령이 Network Security 프로파일에 포함되어 있습니다.

## 2 웹 서버를 중지합니다.

이 명령은 서버가 SMF를 사용하여 실행하도록 구성된 시스템에서 웹 서버를 중지합니다.

```
# svcadm disable svc:/network/http:apache2
```

서비스가 아직 변환되지 않은 경우 다음 명령 구문으로 서비스를 중지하십시오.

```
/usr/apache2/bin/apachectl stop
```

### 3 ksslcfg 명령과 함께 사용할 매개변수를 결정합니다.

모든 옵션은 **ksslcfg(1M)** 매뉴얼 페이지에 나와 있습니다. 정보를 가지고 있어야 하는 매개변수는 다음과 같습니다.

- **key-format** – 인증서 및 키 형식을 정의하기 위해 **-f** 옵션과 함께 사용합니다. SSL 커널 프록시의 경우 값은 **pem** 또는 **pkcs12** 중 하나입니다.
- **key-and-certificate-file** – 서버 키와 인증서를 저장하는 파일의 위치를 설정하기 위해 **-i** 옵션과 함께 사용합니다.
- **password-file** – 개인 키를 암호화하는 데 사용한 암호를 포함하는 파일의 위치를 선택하기 위해 **-p** 옵션과 함께 사용합니다. 암호는 무인 재부트하는 데 사용합니다. 파일에 대한 사용 권한은 **0400**이어야 합니다.
- **proxy-port** – SSL 프록시 포트를 설정하기 위해 **-x** 옵션과 함께 사용합니다. 표준 포트(**80**)가 아닌 다른 포트를 선택합니다. 웹 서버는 SSL 프록시 포트에서 수신 대기합니다.
- **ssl-port** – SSL 커널 프록시에 대한 포트를 수신 대기로 선택합니다. 일반적으로 이 포트는 **443**으로 설정합니다.

---

주 – 이러한 포트는 SSL 커널 프록시에서만 사용되므로 NCA에 대해 **ssl-port** 및 **proxy-port** 값을 구성할 수 없습니다. 일반적으로 NCA에는 포트 **80**, **proxy-port**에는 **8443** 그리고 **ssl-port**에는 **443**이 사용됩니다.

---

### 4 서비스 인스턴스를 만듭니다.

SSL 프록시 포트 및 관련 매개변수를 지정하기 위한 **ksslcfg** 명령입니다.

```
ksslcfg create -f key-format -i key-and-certificate-file -p password-file -x proxy-port ssl-port
```

### 5 인스턴스가 제대로 만들어졌는지 확인합니다.

다음 명령에서 보고하는 서비스 상태는 “online”이어야 합니다.

```
# svcs svc:/network/ssl/proxy
```

### 6 웹 서버를 SSL 프록시 포트에서 수신 대기하도록 구성합니다.

**/etc/apache2/http.conf** 파일을 편집하고 SSL 프록시 포트를 정의하도록 행을 추가합니다. 서버 IP 주소를 사용하는 경우 웹 서버는 해당 인터페이스에 대해서만 수신 대기합니다. 해당 행은 다음과 같습니다.

```
Listen 0.0.0.0:proxy-port
```

### 7 웹 서버에 대한 SMF 종속성을 설정합니다.

웹 서버는 SSL 커널 프록시 인스턴스 후에만 시작해야 합니다. 다음 명령은 이러한 종속성을 설정합니다.

```
# svccfg -s svc:/network/http:apache2
svc:/network/http:apache2> addpg kssl dependency
svc:/network/http:apache2> setprop kssl/entities = fmri:svc:/network/ssl/proxy:kssl-INADDR_ANY-443
svc:/network/http:apache2> setprop kssl/grouping = astring: require_all
```

```
svc:/network/http:apache2> setprop kssl/restart_on = astring: refresh
svc:/network/http:apache2> setprop kssl/type = astring: service
svc:/network/http:apache2> end
```

## 8 웹 서버를 사용으로 설정합니다.

```
# svcadm enable svc:/network/http:apache2
```

서비스가 SMF를 사용하여 시작하지 않은 경우 다음 명령을 사용하십시오.  
/usr/apache2/bin/apachectl startssl

## 예 2-3 SSL 커널 프록시를 사용하여 Apache 2.0 웹 서버 구성

다음 명령은 pem 키 형식을 사용하여 인스턴스를 만듭니다.

```
# ksslcfg create -f pem -i cert-and-key.pem -p file -x 8443 443
```

## ▼ SSL 커널 프록시를 사용하여 Sun Java System Web Server를 구성하는 방법

이 절차는 Sun Java System Web Server에서 SSL 패킷 프로세스의 성능 향상을 위해 사용됩니다. 이 웹 서버에 대한 정보는 [Sun Java System Web Server 7.0 Update 1 Administrator's Guide](#)를 참조하십시오.

**시작하기 전에** 다음 절차에서는 설치 및 구성된 Sun Java System Web Server가 필요합니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오. ksslcfg 명령이 Network Security 프로파일에 포함되어 있습니다.

### 2 웹 서버를 중지합니다.

관리자 웹 인터페이스를 사용하여 서버를 중지합니다. 자세한 내용은 [Sun Java System Web Server 7.0 Update 1 Administrator's Guide](#)에서 **Starting and Stopping the Server**를 참조하십시오.

### 3 ksslcfg 명령과 함께 사용할 매개변수를 결정합니다.

모든 옵션은 [ksslcfg\(1M\)](#) 매뉴얼 페이지에 나와 있습니다. 정보를 가지고 있어야 하는 매개변수는 다음과 같습니다.

- key-format – 인증서 및 키 형식을 정의하기 위해 -f 옵션과 함께 사용합니다.
- token-label – PKCS#11 토큰을 지정하기 위해 -T 옵션과 함께 사용합니다.
- certificate-label – PKCS#11 토큰의 인증서 개체에서 레이블을 선택하기 위해 -c 옵션과 함께 사용합니다.

- **password-file** – 웹 서버에 사용되는 PKCS#11 토큰에 사용자를 로그인하기 위해 사용되는 암호가 포함된 파일의 위치를 선택하기 위해 **-p** 옵션과 함께 사용합니다. 암호는 무인 재부트하는 데 사용합니다. 파일에 대한 사용 권한은 **0400**이어야 합니다.
- **proxy-port** – SSL 프록시 포트를 설정하기 위해 **-x** 옵션과 함께 사용합니다. 표준 포트(**80**)가 아닌 다른 포트를 선택합니다. 웹 서버는 SSL 프록시 포트에서 수신 대기합니다.
- **ssl-port** – SSL 커널 프록시에 대한 포트를 수신 대기로 정의합니다. 일반적으로 이 값은 **443**으로 설정합니다.

---

주 – 이러한 포트는 SSL 커널 프록시에서만 사용되므로 NCA에 대해 **ssl-port** 및 **proxy-port** 값을 구성할 수 없습니다. 일반적으로 NCA에는 포트 **80**, **proxy-port**에는 **8443** 그리고 **ssl-port**에는 **443**이 사용됩니다.

---

#### 4 서비스 인스턴스를 만듭니다.

SSL 프록시 포트 및 관련 매개변수를 지정하기 위한 **ksslcfg** 명령입니다.

```
ksslcfg create -f key-format -T PKCS#11-token -C certificate-label -p password-file -x proxy-port ssl-port
```

#### 5 인스턴스가 제대로 만들어졌는지 확인합니다.

다음 명령에서 보고하는 서비스 상태는 “online”이어야 합니다.

```
# svcs svc:/network/ssl/proxy
```

#### 6 웹 서버를 SSL 프록시 포트에서 수신 대기하도록 구성합니다.

자세한 내용은 [Sun Java System Web Server 7.0 Update 1 Administrator's Guide](#)에서 **Adding and Editing Listen Sockets**를 참조하십시오.

#### 7 웹 서버를 시작합니다.

### 예 2-4 SSL 커널 프록시를 사용하여 Sun Java System Web Server 구성

다음 명령은 **pkcs11** 키 형식을 사용하여 인스턴스를 만듭니다.

```
# ksslcfg create -f pkcs11 -T "Sun Software PKCS#11 softtoken" -C "Server-Cert" -p file -x 8443 443
```



## 영역에서 SSL 커널 프록시 사용

SSL 커널 프록시는 다음 제한 사항과 함께 영역에서 작동합니다.

- 모든 커널 SSL 관리는 전역 영역에서 이루어집니다. 전역 영역 관리자는 로컬 영역 인증서 및 키 파일에 액세스해야 합니다. 로컬 영역 웹 서버는 전역 영역에서 `ksslcfg` 명령을 사용하여 서비스 인스턴스를 한 번 구성하면 시작할 수 있습니다.
- 인스턴스를 구성하기 위해 `ksslcfg` 명령을 실행할 경우 특정 호스트 이름이나 IP 주소를 지정해야 합니다. 특히 인스턴스에 `INADDR_ANY`를 사용할 수 없습니다.

예 2-5 SSL 커널 프록시를 사용하도록 로컬 영역에서 Apache 웹 서버 구성

로컬 영역에서 먼저 웹 서버를 중지합니다. 전역 영역에서 서비스 구성에 대한 모든 단계를 수행합니다. `apache-zone`이라는 로컬 영역에 대한 인스턴스를 만들려면 다음 명령을 사용합니다.

```
# ksslcfg create -f pem -i /zone/apache-zone/root/keypair.pem -p /zone/apache-zone/root/pass \
-x 8443 apache-zone 443
```

로컬 영역에서 다음 명령을 실행하여 서비스 인스턴스를 사용으로 설정합니다.

```
# svcadm enable svc:/network/http:apache2
```

## 웹 페이지 캐싱(참조)

다음 절에서는 NCA를 사용하는 데 필요한 파일 및 구성 요소를 다룹니다. 또한 NCA를 웹 서버와 상호 작용하는 방법에 대한 세부 사항도 포함되어 있습니다.

## NCA 파일

NCA 기능을 지원하려면 여러 파일이 필요합니다. 이러한 파일은 대부분 ASCII이지만 일부는 이진 파일일 수 있습니다. 다음 표에서는 모든 파일을 나열합니다.

표 2-1 NCA 파일

파일 이름	기능
<code>/dev/nca</code>	NCA 장치에 대한 경로 이름입니다.
<code>/etc/hostname.*</code>	서버에서 구성된 모든 물리적 인터페이스를 나열하는 파일입니다.
<code>/etc/hosts</code>	서버와 관련된 모든 호스트 이름을 나열하는 파일입니다. 이 파일의 항목이 작동하려면 NCA에 대한 <code>/etc/hostname.*</code> 파일의 항목과 일치해야 합니다.

표 2-1 NCA 파일 (계속)

파일 이름	기능
/etc/init.d/ncakmod	NCA 서버를 시작하는 스크립트입니다. 이 스크립트는 서버가 부트할 때 실행됩니다.
/etc/init.d/ncalogd	NCA 로깅을 시작하는 스크립트입니다. 이 스크립트는 서버가 부트할 때 실행됩니다.
/etc/nca/nca.if	NCA를 실행하는 인터페이스를 나열하는 파일입니다. 자세한 내용은 <a href="#">nca.if(4)</a> 매뉴얼 페이지를 참조하십시오.
/etc/nca/ncakmod.conf	NCA에 대한 구성 매개변수를 나열하는 파일입니다. 자세한 내용은 <a href="#">ncakmod.conf(4)</a> 매뉴얼 페이지를 참조하십시오.
/etc/nca/ncalogd.conf	NCA 로깅에 대한 구성 매개변수를 나열하는 파일입니다. 자세한 내용은 <a href="#">ncalogd.conf(4)</a> 매뉴얼 페이지를 참조하십시오.
/etc/nca/ncaport.conf	NCA에 대한 IP 주소 및 포트를 나열하는 파일입니다. 자세한 내용은 <a href="#">ncaport.conf(4)</a> 매뉴얼 페이지를 참조하십시오.
/system/volatile/nca_httpd_1.door	도어 경로 이름입니다.
/usr/bin/ncab2clf	로그 파일에서 데이터를 일반 로그 형식으로 변환하는 데 사용하는 명령입니다. 자세한 내용은 <a href="#">ncab2clf(1)</a> 매뉴얼 페이지를 참조하십시오.
/usr/lib/net/ncaconfd	부트 동안 다중 인터페이스에서 NCA가 실행되도록 구성하는 데 사용하는 명령입니다. 자세한 내용은 <a href="#">ncaconfd(1M)</a> 매뉴얼 페이지를 참조하십시오.
/usr/lib/nca_addr.so	AF_INET 소켓 대신 AF_NCA 소켓을 사용하는 라이브러리입니다. 이 라이브러리는 AF_INET 소켓을 사용하는 웹 서버에서 사용해야 합니다. 자세한 내용은 <a href="#">ncad_addr(4)</a> 매뉴얼 페이지를 참조하십시오.
/var/nca/log	로그 파일 데이터를 보유하는 파일입니다. 이진 형식의 파일이므로 편집하지 마십시오.

## NCA 구조

NCA 기능은 다음 구성 요소를 포함합니다.

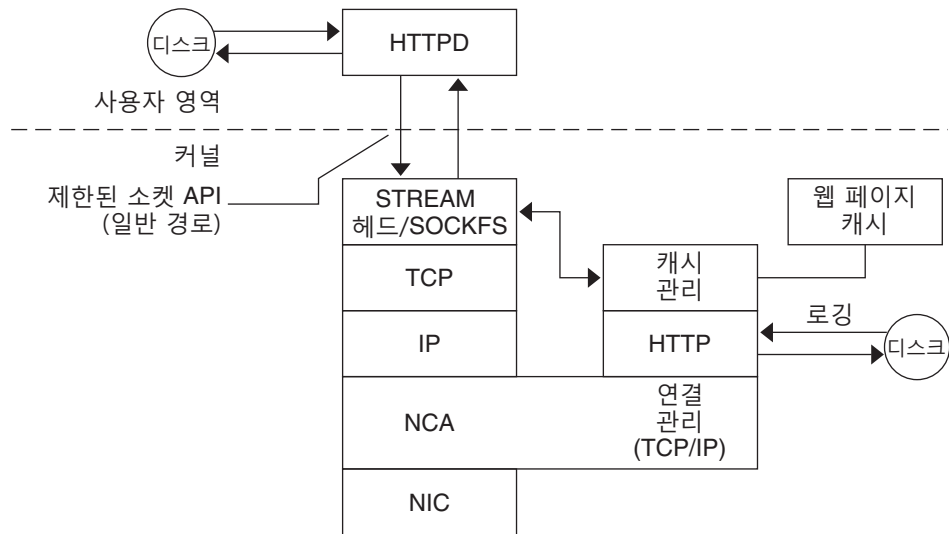
- ncakmod 커널 모듈
- httpd 웹 서버

ncakmod 커널 모듈은 시스템 메모리에서 웹 페이지의 캐시를 유지 관리합니다. 모듈은 소켓 인터페이스를 통해 httpd 웹 서버와 통신합니다. 패밀리 형식은 PF\_NCA입니다.

커널 모듈은 모든 HTTP 캐시 적중을 기록하는 로깅 기능도 제공합니다. NCA 로깅은 이진 형식의 디스크에 HTTP 데이터를 씁니다. NCA는 이진 로그 파일을 CLF(일반 로그 형식)로 변환하는 변환 유틸리티를 제공합니다.

다음 그림은 일반 경로 및 NCA를 사용으로 설정할 때 사용하는 경로에 대한 데이터 흐름을 보여줍니다.

그림 2-1 NCA 서비스를 포함하는 데이터 흐름



## NCA에서 httpd 요청까지의 흐름

다음 목록은 클라이언트와 웹 서버 간의 요청 흐름을 보여줍니다.

1. HTTP 요청은 클라이언트에서 웹 서버로 이루어집니다.
2. 페이지가 캐시에 있는 경우 커널 내 캐시 웹 페이지가 반환됩니다.
3. 페이지가 캐시에 없는 경우 요청은 페이지를 검색 또는 업데이트하기 위해 웹 서버로 이동합니다.
4. 응답에서 사용되는 HTTP 프로토콜 의미에 따라 페이지가 캐시되거나 캐시되지 않습니다. 그런 다음 페이지가 클라이언트로 반환됩니다. Pragma: No-cache 헤더가 HTTP 요청에 포함된 경우 페이지가 캐시되지 않습니다.



## 시간 관련 서비스

---

네트워크 내에서 시스템 클록을 동기화된 상태로 유지하는 것은 여러 데이터베이스 및 인증 서비스에 필요합니다. 이 장에서는 다음 항목을 다룹니다.

- 61 페이지 “클록 동기화(개요)”
- 62 페이지 “NTP(Network Time Protocol) 관리(작업)”
- 64 페이지 “기타 시간 관련 명령 사용(작업)”
- 64 페이지 “NTP(Network Time Protocol)(참조)”

### 클록 동기화(개요)

University of Delaware의 NTP(Network Time Protocol) 공용 도메인 소프트웨어는 Oracle Solaris 소프트웨어에 포함되어 있습니다. `ntpd` 데몬은 시스템 시간을 설정하고 유지 관리합니다. `ntpd` 데몬은 RFC 5905에 정의된 버전 4 표준의 완전한 구현입니다.

`ntpd` 데몬은 시스템 시작 시 `/etc/inet/ntp.conf` 파일을 읽습니다. 구성 옵션에 대한 자세한 내용은 `ntp.conf(4)` 매뉴얼 페이지를 참조하십시오.

네트워크에서 NTP를 사용할 때는 다음 사항을 기억하십시오.

- `ntpd` 데몬은 최소 시스템 리소스를 사용합니다.
- NTP 클라이언트는 부트될 때 NTP 서버와 자동으로 동기화됩니다. 클라이언트가 동기화되지 않은 경우 클라이언트가 시간 서버에 연결할 때 다시 동기화됩니다.

클록을 동기화하는 다른 방법은 `cron`을 사용하는 중에 `rdate`를 실행하는 것입니다.

## 이 릴리스의 NTP 정보

이 Oracle Solaris 릴리스에서 제공되는 변경 사항은 다음과 같습니다.

- 버전 3 표준을 기반으로 하던 `xntpd` 데몬이 버전 4 표준을 기반으로 하는 `ntpd` 데몬으로 대체되었습니다.
- NTP 서비스에 대한 추가 설명서는 Oracle Solaris 11 릴리스를 실행하는 시스템의 `/usr/share/doc/ntp/index.html`에서 찾을 수 있습니다.

## NTP(Network Time Protocol) 관리(작업)

다음 절차에서는 NTP 서비스를 설정 및 사용하는 방법을 보여줍니다.

### ▼ NTP 서버를 설정하는 방법

**1 관리자가 됩니다.**

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

**2 `ntp.conf` 파일을 만듭니다.**

`ntpd` 데몬이 제대로 실행되게 하려면 먼저 `ntp.conf` 파일을 만들어야 합니다. `ntp.client` 파일을 템플릿으로 사용할 수 있습니다.

```
# cd /etc/inet
# cp ntp.client ntp.conf
```

**3 `ntp.server` 파일을 읽습니다.**

필요한 경우 `ntp.conf` 파일에 정보를 추가합니다.

**4 `ntp.conf` 파일을 편집합니다.**

필요한 경우 이 파일에서 사이트 관련 내용을 변경합니다.

**5 `ntpd` 데몬을 시작합니다.**

```
# svcadm enable ntp
```

### ▼ NTP 클라이언트를 설정하는 방법

**1 관리자가 됩니다.**

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

**2 ntp.conf 파일을 만듭니다.**

ntpd 데몬을 활성화하려면 먼저 ntp.conf 파일을 만들어야 합니다.

```
# cd /etc/inet
# cp ntp.client ntp.conf
```

**3 ntp.conf 파일을 편집합니다.**

필요한 경우 이 파일에서 사이트 관련 내용을 변경합니다.

**4 ntpd 데몬을 시작합니다.**

```
# svcadm enable ntp
```

## ▼ NTP 로깅을 사용으로 설정하는 방법

**1 관리자가 됩니다.**

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

**2 로깅을 사용으로 설정합니다.**

```
# svccfg -s svc:/network/ntp:default setprop config/verbose_logging = true
```

자세한 내용은 [svccfg\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

**3 SMF 저장소를 업데이트하고 서비스를 다시 시작합니다.**

```
# svcadm refresh svc:/network/ntp:default
# svcadm restart svc:/network/ntp:default
```

**4 로깅이 사용으로 설정되었는지 확인합니다.**

```
# svcprop -p config/verbose_logging svc:/network/ntp:default
true
```

## ▼ NTP 서비스와 연관된 SMF 등록 정보를 표시하는 방법

**● SMF 등록 정보를 나열합니다.**

- NTP 서비스와 연관된 SMF 등록 정보를 모두 나열하려면 다음을 입력합니다.

```
# svcprop svc:/network/ntp:default
```

- config 등록 정보 그룹의 등록 정보를 모두 나열하려면 다음을 입력합니다.

```
# svcprop -p config svc:/network/ntp:default
```

# 기타 시간 관련 명령 사용(작업)

다음 절차를 사용하면 NTP를 설정하지 않고도 필요할 때 언제든지 현재 시간을 업데이트할 수 있습니다.

## ▼ 다른 시스템에서 시간 및 날짜를 동기화하는 방법

- 1 관리자가 됩니다.  
자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.
- 2 **rdate** 명령을 사용하여 날짜와 시간을 재설정하여 다른 시스템과 동기화합니다.  
`# rdate another-system`  
`another-system`     다른 시스템의 이름
- 3 **date** 명령을 사용하여 시스템 날짜를 올바르게 재설정했는지 확인합니다.  
출력에는 다른 시스템과 일치하는 날짜 및 시간이 표시되어야 합니다.

### 예 3-1 다른 시스템에서 시간 및 날짜 동기화

다음 예에서는 **rdate**를 사용하여 한 시스템의 날짜 및 시간을 다른 시스템과 동기화하는 방법을 보여줍니다. 이 예에서는 여러 시간 전에 실행된 **earth** 시스템이 **starbug** 서버의 날짜 및 시간과 일치하도록 재설정됩니다.

```
earth# date
Tue Jun  5 11:08:27 MDT 2001
earth# rdate starbug
Tue Jun  5 14:06:37 2001
earth# date
Tue Jun  5 14:06:40 MDT 2001
```

# NTP(Network Time Protocol)(참조)

NTP 서비스를 실행하려면 다음 파일이 필요합니다.

표 3-1 NTP 파일

파일 이름	기능
/etc/inet/ntp.conf	NTP의 구성 옵션을 나열합니다.
/etc/inet/ntp.client	NTP 클라이언트 및 서버의 샘플 구성 파일입니다.



표 3-1 NTP 파일 (계속)

파일 이름	기능
/etc/inet/ntp.leap	윤초 구성 파일입니다.
/etc/inet/ntp.keys	NTP 인증 키를 포함합니다.
/etc/inet/ntp.server	일부 NTP 서버에 대한 추가 구성 명령을 포함합니다.
/usr/lib/inet/ntpd	NTP 데몬입니다. 자세한 내용은 <code>ntpd(1M)</code> 매뉴얼 페이지를 참조하십시오.
/usr/sbin/ntp-keygen	NTP의 공용 및 개인 키를 생성하는 데 사용되는 프로그램입니다. 자세한 내용은 <code>ntp-keygen(1M)</code> 매뉴얼 페이지를 참조하십시오.
/usr/sbin/ntpdcc	<code>ntpd</code> 데몬의 NTP 질의 프로그램입니다. 자세한 내용은 <code>ntpdcc(1M)</code> 매뉴얼 페이지를 참조하십시오.
/usr/sbin/ntpdate	NTP를 기반으로 로컬 날짜 및 시간을 설정하는 유틸리티입니다. 자세한 내용은 <code>ntpdate(1M)</code> 매뉴얼 페이지를 참조하십시오.
/usr/sbin/ntpqq	NTP 질의 프로그램입니다. 자세한 내용은 <code>ntpqq(1M)</code> 매뉴얼 페이지를 참조하십시오.
/var/ntp/ntpstats	NTP 통계를 저장하기 위한 디렉토리입니다.
/usr/sbin/ntpptime	커널 시간 변수를 표시하거나 설정하는 프로그램입니다. 자세한 내용은 <code>ntpptime(1M)</code> 매뉴얼 페이지를 참조하십시오.
/usr/sbin/ntptrace	마스터 NTP 서버까지 NTP 호스트를 추적하는 프로그램입니다. 자세한 내용은 <code>ntptrace(1M)</code> 매뉴얼 페이지를 참조하십시오.
/var/ntp/ntp.drift	NTP 서버에서 초기 빈도 오프셋을 설정합니다.



## 제 2 부

# 네트워크 파일 시스템 액세스 항목

이 절에서는 NFS 서비스에 대한 개요, 작업 및 참조 정보를 제공합니다.



## 네트워크 파일 시스템 관리(개요)

---

이 장에서는 네트워크를 통해 파일 시스템에 액세스하는 데 사용할 수 있는 NFS 서비스에 대해 간략하게 소개합니다. 이 장에는 NFS 서비스를 이해하는 데 필요한 개념에 대한 설명과, NFS 및 autofs의 최신 기능에 대한 설명이 포함되어 있습니다.

- 69 페이지 “NFS 서비스의 새로운 기능”
- 72 페이지 “NFS 용어”
- 73 페이지 “NFS 서비스 정보”
- 73 페이지 “autofs 정보”
- 74 페이지 “NFS 서비스의 기능”

---

주 - 시스템에서 영역이 사용으로 설정된 경우 비전역 영역에서 이 기능을 사용하려면 **Oracle Solaris 관리: Oracle Solaris Zones, Oracle Solaris 10 Zones 및 리소스 관리**에서 자세한 내용을 참조하십시오.

---

## NFS 서비스의 새로운 기능

이 절에서는 Oracle Solaris OS 릴리스의 새로운 기능에 대해 소개합니다.

### 이번 릴리스의 변경 내용

Oracle Solaris 11 릴리스에는 다음과 같은 향상된 기능이 포함되어 있습니다.

- 이전에는 /etc/default/autofs 및 /etc/default/nfs를 편집하여 설정했던 구성 매개변수를 이제는 SMF 저장소에서 설정할 수 있습니다. 새 SMF 매개변수에 대한 설명은 해당 매개변수를 사용하는 절차 및 해당 매개변수를 사용하는 데몬의 설명을 참조하십시오.
  - 146 페이지 “automount 명령”
  - 135 페이지 “automountd 데몬”

- [136 페이지 “lockd 데몬”](#)
- [137 페이지 “mountd 데몬”](#)
- [137 페이지 “nfsd 데몬”](#)
- [139 페이지 “nfsmapid 데몬”](#)
- NFS 서비스에서는 미러 마운트를 지원합니다. NFSv4 클라이언트에서는 미러 마운트를 통해 서버 네임스페이스에서 공유 파일 시스템 마운트 지점을 순회할 수 있습니다. NFSv4 마운트의 경우 자동 마운트는 서버 네임스페이스 루트 마운트를 수행하며 미러 마운트를 통해 해당 파일 시스템에 액세스합니다. 기존의 자동 마운트와 비교할 때 미러 마운트가 제공하는 가장 큰 이점은, 미러 마운트를 사용하여 파일 시스템을 마운트하는 경우 자동 마운트 맵 관리 시의 오버헤드가 발생하지 않는다는 것입니다. 미러 마운트에서 제공하는 기능은 다음과 같습니다.
  - 네임스페이스 변경 내용을 모든 클라이언트에서 즉시 볼 수 있습니다.
  - 새로 공유된 파일 시스템이 즉시 검색되며 자동으로 마운트됩니다.
  - 지정된 시간 동안 작업을 하지 않으면 파일 시스템이 자동으로 마운트 해제됩니다.

미러 마운트에 대한 자세한 내용은 다음 항목을 참조하십시오.

- [87 페이지 “서버에서 모든 파일 시스템을 마운트하는 방법”](#)
- [193 페이지 “미러 마운트의 작동 방식”](#)
- NFS 참조가 NFS 서비스에 추가되었습니다. 참조는 NFSv4 클라이언트가 파일 시스템을 찾기 위해 이동할 수 있는 서버 기반 리디렉션입니다. NFS 서버에서는 `nfsref(1M)` 명령으로 생성된 참조를 지원하며, NFSv4 클라이언트는 이러한 참조를 따라 실제 위치에서 파일 시스템을 마운트합니다. 대부분의 경우 이 기능을 자동 마운트 대신 사용할 수 있으며, 자동 마운트 맵을 편집하는 대신 참조를 생성하면 됩니다. NFS 참조에서 제공하는 기능은 다음과 같습니다.
  - 위에 나와 있는 모든 미러 마운트 기능
  - 자동 마운트와 유사한 기능(자동 마운트에 종속되지는 않음)
  - 클라이언트나 서버에서 설정을 수행하지 않아도 됨

NFS 참조에 대한 자세한 내용은 다음 항목을 참조하십시오.

- [115 페이지 “NFS 참조 관리”](#)
- [194 페이지 “NFS 참조의 작동 방식”](#)
- 통합 파일 시스템 네임스페이스의 DNS 도메인 루트별로 마운트하는 기능이 추가되었습니다. 이 마운트 지점을 NFS 참조와 함께 사용하면 파일 서버 간을 연결하여 대형 네임스페이스를 원하는 대로 작성할 수 있습니다. 자세한 내용은 다음 항목을 참조하십시오.
  - [91 페이지 “통합 파일 시스템 서버에 대해 DNS 레코드 설정”](#)
  - [197 페이지 “마운트 지점 /nfs4”](#)
- `sharectl` 유틸리티가 포함되었습니다. 이 유틸리티를 사용하면 NFS 등의 파일 공유 프로토콜을 구성 및 관리할 수 있습니다. 예를 들어 이 유틸리티에서는 클라이언트 및 서버 작동 등록 정보를 설정하고, 특정 프로토콜에 대한 등록 정보 값을 표시하고,

프로토콜 상태를 가져올 수 있습니다. 자세한 내용은 [sharectl\(1M\) man page and 155 페이지 “sharectl 명령”](#)을 참조하십시오.

- NFS 버전 4 도메인을 정의할 수 있습니다. 자세한 내용은 [143 페이지 “Oracle Solaris 11 릴리스에서 NFS 버전 4 기본 도메인 구성”](#)을 참조하십시오.

## 이전 릴리스의 중요한 변경 사항

Solaris 10 11/06 릴리스는 파일 시스템 모니터링 도구를 지원합니다. 다음 항목을 참조하십시오.

- [147 페이지 “fsstat 명령”](#)에 대한 설명 및 예
- 자세한 내용은 [fsstat\(1M\)](#) 매뉴얼 페이지 참조

또한 이 가이드에서는 [nfsmapid](#) 데몬에 대해서도 자세하게 설명합니다. [nfsmapid](#)에 대한 자세한 내용은 다음 항목을 참조하십시오.

- [139 페이지 “nfsmapid 데몬”](#)
- [nfsmapid\(1M\)](#) 매뉴얼 페이지

Solaris 10 릴리스부터는 NFS 버전 4가 기본값입니다. NFS 버전 4의 기능 및 기타 변경 내용에 대한 자세한 내용은 다음 항목을 참조하십시오.

- [75 페이지 “NFS 버전 4 프로토콜”](#)
- [136 페이지 “lockd 데몬”](#)
- [137 페이지 “nfs4cbd 데몬”](#)
- [139 페이지 “nfsmapid 데몬”](#)
- [148 페이지 “NFS 파일 시스템용 mount 옵션”](#)
- [170 페이지 “RDMA를 통한 NFS”](#)
- [172 페이지 “NFS의 버전 협상”](#)
- [173 페이지 “NFS 버전 4의 기능”](#)
- [205 페이지 “autofs에서 클라이언트에 대해 가장 가까운 읽기 전용 파일을 선택하는 방법\(여러 위치\)”](#)

작업 정보는 [91 페이지 “NFS 서비스 설정”](#)을 참조하십시오.

또한 NFS 서비스는 서비스 관리 기능을 통해 관리됩니다. [svcadm](#) 명령을 사용하여 사용으로 설정, 사용 안함으로 설정, 다시 시작 등 이 서비스에 대한 관리 작업을 수행할 수 있습니다. [svcs](#) 명령을 사용하여 서비스의 상태를 질의할 수 있습니다. 서비스 관리 기능에 대한 자세한 내용은 [smf\(5\)](#) 매뉴얼 페이지 및 [Oracle Solaris 관리: 일반 작업의 6 장, “서비스 관리\(개요\)”](#)를 참조하십시오.

## NFS 용어

이 절에서는 NFS 서비스를 사용하려면 이해해야 하는 몇 가지 기본적인 용어를 소개합니다. NFS 서비스에 대한 보다 자세한 설명은 6 장, “네트워크 파일 시스템 액세스(참조)”에 나와 있습니다.

### NFS 서버 및 클라이언트

**클라이언트**와 **서버**라는 용어는 파일 시스템을 공유할 때의 컴퓨터 역할을 설명하는 데 사용됩니다. 네트워크를 통해 해당 파일 시스템을 공유하는 컴퓨터는 서버로 작동합니다. 클라이언트는 파일 시스템에 액세스하는 컴퓨터입니다. NFS 서비스에서는 모든 컴퓨터가 다른 컴퓨터의 파일 시스템에 액세스할 수 있도록 합니다. 그와 동시에 NFS 서비스는 해당 파일 시스템 액세스도 제공합니다. 한 컴퓨터가 네트워크에서 특정 시간에 클라이언트나 서버 역할을 할 수도 있고 클라이언트인 동시에 서버가 될 수도 있습니다.

클라이언트는 서버의 공유 파일 시스템을 마운트하여 서버의 파일에 액세스합니다. 클라이언트는 원격 파일 시스템을 마운트할 때 파일 시스템 복사본을 만들지 않습니다. 대신 마운트 프로세스에서는 클라이언트가 서버 디스크에서 파일 시스템에 투명하게 액세스할 수 있도록 하는 일련의 원격 프로시저 호출을 사용합니다. 이러한 마운트는 로컬 마운트와 비슷합니다. 즉, 파일 시스템이 로컬인 것처럼 사용자가 명령을 입력합니다. 파일 시스템을 마운트하는 작업에 대한 자세한 내용은 85 페이지 “파일 시스템 마운트”를 참조하십시오.

NFS 작업을 통해 서버에서 공유된 파일 시스템은 클라이언트에서 액세스할 수 있습니다. `autofs`를 사용하여 NFS 파일 시스템을 자동으로 마운트할 수 있습니다. `share` 명령 및 `autofs`와 관련된 작업에 대한 내용은 82 페이지 “자동 파일 시스템 공유” 및 100 페이지 “Autofs 관리 작업 개요”를 참조하십시오.

### NFS 파일 시스템

NFS 서비스와 공유할 수 있는 개체에는 전체 또는 부분 디렉토리 트리나 파일 계층이 포함되며, 단일 파일도 포함됩니다. 컴퓨터는 이미 공유된 파일 계층과 겹치는 파일 계층을 공유할 수 없습니다. 모뎀, 프린터 등의 주변 기기는 공유할 수 없습니다.

대부분의 UNIX 시스템 환경에서 공유할 수 있는 파일 계층은 파일 시스템이나 파일 시스템의 일부분에 해당합니다. 그러나 NFS는 운영 체제 간 작업을 지원하며, UNIX 이외의 다른 환경에서는 파일 시스템 개념이 무의미할 수 있습니다. 따라서 **파일 시스템**이라는 용어는 NFS를 통해 공유 및 마운트할 수 있는 파일이나 파일 계층을 지칭합니다.



## NFS 서비스 정보

NFS 서비스는 구조가 서로 다르며 각각 다른 운영 체제를 실행하는 여러 컴퓨터가 네트워크를 통해 파일 시스템을 공유할 수 있도록 합니다. NFS 지원은 MS-DOS에서 VMS 운영 체제에 이르기까지 다양한 플랫폼에서 구현되었습니다.

NFS는 구조 사양이 아닌 파일 시스템의 추상 모델을 정의하므로 NFS 환경을 여러 운영 체제에서 구현할 수 있습니다. 각 운영 체제는 해당 파일 시스템 의미에 NFS 모델을 적용합니다. 이 모델은 읽기 및 쓰기과 같은 파일 시스템 작업이 로컬 파일에 액세스하는 것처럼 작동함을 의미합니다.

NFS 서비스에는 다음과 같은 이점이 있습니다.

- 여러 컴퓨터가 같은 파일을 사용할 수 있도록 하여 네트워크의 모든 사용자가 같은 데이터에 액세스하도록 합니다.
- 각 사용자 응용 프로그램에 대해 로컬 디스크 공간을 필요로 하는 대신 여러 컴퓨터가 응용 프로그램을 공유하도록 하여 저장소 비용을 줄여 줍니다.
- 모든 사용자가 같은 파일 세트를 읽을 수 있으므로 데이터 일관성 및 안정성이 제공됩니다.
- 파일 시스템 마운트가 사용자에게 투명하게 수행됩니다.
- 원격 파일 액세스가 사용자에게 투명하게 수행됩니다.
- 이기종 환경이 지원됩니다.
- 시스템 관리 오버헤드가 줄어듭니다.

NFS 서비스에서는 사용자가 파일 시스템의 실제 위치를 몰라도 됩니다. NFS 구현을 통해 사용자가 위치에 관계없이 관련 파일을 모두 보도록 할 수 있습니다. NFS 서비스는 일반적으로 사용되는 파일의 복사본을 모든 파일에 저장하는 대신 한 컴퓨터 디스크에 복사본 하나를 저장하도록 합니다. 다른 모든 시스템은 네트워크를 통해 파일에 액세스합니다. NFS 작업 시에는 원격 파일 시스템과 로컬 파일 시스템이 거의 동일합니다.

## autofs 정보

NFS 서비스를 통해 공유되는 파일 시스템은 자동 마운트를 사용하여 마운트할 수 있습니다. 클라이언트측 서비스인 **autofs**는 자동 마운트 기능을 제공하는 파일 시스템 구조입니다. **autofs** 파일 시스템은 시스템 부트 시 자동으로 실행되는 **automount**를 통해 초기화됩니다. 자동 마운트 데몬인 **automountd**는 지속적으로 실행되어 필요에 따라 원격 디렉토리를 마운트하고 마운트 해제합니다.

**automountd**를 실행 중인 클라이언트 컴퓨터에서 원격 파일이나 원격 디렉토리에 액세스할 때마다 데몬이 원격 파일 시스템을 마운트합니다. 이 원격 파일 시스템은 필요한 시간 동안 마운트된 상태로 유지됩니다. 특정 시간 동안 원격 파일 시스템에 액세스하지 않으면 해당 파일 시스템이 자동으로 마운트 해제됩니다.

마운트는 부트 시 수행해야 하는 것은 아니며, 사용자는 이제 슈퍼 유저 암호를 몰라도 디렉토리를 마운트할 수 있습니다. 사용자는 `mount` 및 `umount` 명령을 사용할 필요가 없습니다. 사용자가 작업을 수행하지 않아도 `autofs` 서비스에서 필요에 따라 파일 시스템을 마운트 및 마운트 해제합니다.

`automountd`를 사용하여 일부 파일 계층을 마운트하는 경우에도 `mount`를 사용하여 다른 계층을 마운트할 수 있습니다. 디스크가 없는 컴퓨터는 `mount` 명령 및 `/etc/vfstab` 파일을 통해 `/`(루트), `/usr` 및 `/usr/kvm`을 마운트해야 합니다.

100 페이지 “Autofs 관리 작업 개요” 및 201 페이지 “autofs의 작동 방식”에 autofs 서비스에 대한 보다 구체적인 정보가 나와 있습니다.

## NFS 서비스의 기능

이 절에서는 NFS 서비스에 포함된 중요한 기능에 대해 설명합니다.

### NFS 버전 2 프로토콜

버전 2는 널리 사용된 최초의 NFS 프로토콜 버전이었습니다. 현재도 다양한 플랫폼에서 버전 2를 계속 사용할 수 있습니다. 모든 Oracle Solaris 릴리스에서는 버전 2 NFS 프로토콜이 지원됩니다.

### NFS 버전 3 프로토콜

NFS 버전 2 프로토콜과 달리, NFS 버전 3 프로토콜은 2GB보다 큰 파일을 처리할 수 있습니다. 즉, 이전 버전의 제한이 제거되었습니다. 78 페이지 “NFS 큰 파일 지원”을 참조하십시오.

NFS 버전 3 프로토콜을 사용하면 서버에서 비동기 쓰기를 안전하게 수행할 수 있으므로, 서버가 메모리에서 클라이언트 쓰기 요청을 캐시할 수 있어 성능이 향상됩니다. 클라이언트는 서버가 디스크에 변경 내용을 적용할 때까지 기다리지 않아도 되므로 응답 시간이 빨라집니다. 또한 서버는 요청을 일괄 처리할 수 있어 서버의 응답 시간도 빨라집니다.

대부분의 Solaris NFS 버전 3 작업에서는 파일 속성이 저장되며, 이러한 속성은 로컬 캐시에 저장됩니다. 캐시가 보다 자주 업데이트되므로 이 데이터를 업데이트하기 위한 별도의 작업을 수행하는 빈도는 낮아집니다. 따라서 서버에 대한 RPC 호출 횟수가 줄어들어 성능이 향상됩니다.

파일 액세스 권한 확인 프로세스가 개선되었습니다. 버전 2에서는 사용자가 적절한 권한이 없는 원격 파일을 복사하려고 하면 “쓰기 오류” 메시지 또는 “읽기 오류” 메시지가 생성되었습니다. 버전 3에서는 파일을 열기 전에 권한을 확인하므로 오류가 “열기 오류”로 보고됩니다.

NFS 버전 3 프로토콜에서는 전송 크기 제한(8KB)이 제거되었습니다. 클라이언트와 서버는 버전 2에서 적용되었던 것처럼 8KB의 제한을 따르는 대신, 지원하는 전송 크기에 관계없이 협상을 할 수 있습니다. 이전 Solaris 구현에서는 프로토콜의 기본 전송 크기가 32KB였습니다. Solaris 10 릴리스부터는 유선 전송 크기 제한이 완화되었습니다. 전송 크기는 기본 전송 기능을 기반으로 합니다.

## NFS 버전 4 프로토콜

NFS 버전 4에는 이전 버전에서는 사용할 수 없었던 기능이 포함되어 있습니다.

NFS 버전 4 프로토콜에서는 사용자 ID와 그룹 ID를 문자열로 표시합니다. 클라이언트와 서버에서는 `nfsmapid`를 사용하여 다음을 수행합니다.

- 이러한 버전 4 ID 문자열을 로컬 숫자 ID로 매핑
- 로컬 숫자 ID를 버전 4 ID 문자열로 매핑

자세한 내용은 [139 페이지 “nfsmapid 데몬”](#)을 참조하십시오.

NFS 버전 4에서는 ID 매핑 `nfsmapid`를 통해 서버의 ACL 항목에 있는 사용자 또는 그룹 ID를 클라이언트의 ACL 항목에 있는 사용자 또는 그룹 ID로 매핑합니다. 그 반대의 경우도 마찬가지입니다. 자세한 내용은 [181 페이지 “NFS 버전 4의 ACL 및 nfsmapid”](#)를 참조하십시오.

NFS 버전 4를 사용하는 경우 파일 시스템 공유를 해제할 때 해당 파일 시스템의 모든 열린 파일 또는 파일 잠금에 대한 상태가 모두 삭제됩니다. NFS 버전 3의 경우에는 서버에서 파일 시스템 공유를 해제하기 전에 클라이언트가 획득한 잠금을 유지합니다. 자세한 내용은 [173 페이지 “NFS 버전 4에서 파일 시스템 공유 해제 및 다시 공유”](#)를 참조하십시오.

NFS 버전 4 서버에서는 의사 파일 시스템을 사용하여 서버에서 내보낸 객체에 대한 액세스 권한을 클라이언트에 제공합니다. NFS 버전 4 이전에는 의사 파일 시스템이 없었습니다. 자세한 내용은 [174 페이지 “NFS 버전 4의 파일 시스템 네임스페이스”](#)를 참조하십시오.

NFS 버전 2 및 버전 3에서는 서버에서 영구 파일 핸들을 반환했습니다. NFS 버전 4에서는 휘발성 파일 핸들이 지원됩니다. 자세한 내용은 [175 페이지 “NFS 버전 4의 휘발성 파일 핸들”](#)을 참조하십시오.

서버에서 파일 관리 권한을 클라이언트에 위임하는 위임 기능이 클라이언트와 서버에서 모두 지원됩니다. 예를 들어 서버에서 읽기 위임 또는 쓰기 위임을 클라이언트에 부여할 수 있습니다. 자세한 내용은 [179 페이지 “NFS 버전 4의 위임”](#)을 참조하십시오.

Solaris 10 릴리스부터는 NFS 버전 4가 LIPKEY/SPKM 보안 종류를 지원하지 않습니다.

또한 NFS 버전 4에서는 다음 데몬이 사용되지 않습니다.

- mountd
- nfslogd
- statd

NFS 버전 4 기능의 전체 목록은 [173 페이지 “NFS 버전 4의 기능”](#)을 참조하십시오.

NFS 버전 4 사용과 관련된 절차 정보는 [91 페이지 “NFS 서비스 설정”](#)을 참조하십시오.

## NFS 버전 제어

SMF 저장소에는 클라이언트와 서버에서 모두 사용되는 NFS 프로토콜을 제어하는 매개변수가 포함되어 있습니다. 예를 들어 버전 협상을 관리하는 매개변수를 사용할 수 있습니다. 자세한 정보는 [137 페이지 “mountd 데몬”](#)(클라이언트 매개변수) [137 페이지 “nfsd 데몬”](#)(서버 매개변수) 또는 [nfs\(4\)](#) 매뉴얼 페이지를 참조하십시오.

## NFS ACL 지원

액세스 제어 목록(ACL) 지원은 Solaris 2.5 릴리스에 추가되었습니다. 액세스 제어 목록(ACL)은 표준 UNIX 파일 권한을 통해 제공되는 것보다 상세하게 파일 액세스 권한을 설정하는 방식을 제공합니다. NFS ACL이 지원되므로 Oracle Solaris NFS 클라이언트에서 Oracle Solaris NFS 서버로 ACL 항목을 변경하고 볼 수 있습니다.

NFS 버전 2 및 버전 3 프로토콜에서는 이전의 POSIX 드래프트 스타일 ACL이 지원됩니다. POSIX 드래프트 ACL은 UFS에서 기본적으로 지원됩니다. UFS ACL에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “액세스 제어 목록을 사용하여 UFS 파일 보호”](#)를 참조하십시오.

NFS 버전 4 프로토콜에서는 새로운 NFSv4 스타일 ACL이 지원됩니다. NFSv4 ACL은 ZFS에서 기본적으로 지원됩니다. NFSv4 ACL의 모든 기능을 사용하려면 NFSv4 서버에서 ZFS를 기본 파일 시스템으로 사용해야 합니다. NFSv4 ACL에는 다양한 상속 등록 정보 세트와 표준 읽기/쓰기/실행 이외의 권한 비트 세트가 포함되어 있습니다. 새로운 ACL에 대한 개요는 [Oracle Solaris 관리: ZFS 파일 시스템의 8 장, “ACL 및 속성을 사용하여 Oracle Solaris ZFS 파일 보호”](#)를 참조하십시오. NFS 버전 4의 ACL 지원에 대한 자세한 내용은 [181 페이지 “NFS 버전 4의 ACL 및 nfsmapid”](#)를 참조하십시오.

## TCP를 통한 NFS

Solaris 2.5에서는 NFS 프로토콜의 기본 전송 프로토콜이 TCP(전송 제어 프로토콜)로 변경되었습니다. TCP를 사용하면 저속 네트워크 및 WAN(Wide Area Network)의 성능을 개선할 수 있습니다. 또한 TCP는 혼잡 제어 및 오류 복구 기능도 제공합니다. TCP를 통한 NFS는 버전 2, 버전 3 및 버전 4에서 작동합니다. Solaris 2.5 릴리스 이전의 기본 NFS 프로토콜은 UDP(사용자 데이터그램 프로토콜)였습니다.

---

주 - InfiniBand용 RDMA를 사용할 수 있는 경우에는 RDMA가 NFS의 기본 전송 프로토콜로 사용됩니다. 자세한 내용은 [170 페이지 “RDMA를 통한 NFS”](#)를 참조하십시오. 그러나 `proto=tcp` 마운트 옵션을 사용하는 경우에는 NFS 마운트에서 TCP만 사용하도록 강제 지정됩니다.

---

## UDP를 통한 NFS

Solaris 10 릴리스부터는 NFS 클라이언트가 더 이상 많은 수의 UDP 포트를 사용하지 않습니다. 이전에 UDP를 통한 NFS 전송은 처리되지 않은 각 요청에 대해 별도의 UDP 포트를 사용했습니다. 이제 기본적으로 NFS 클라이언트는 예약된 단일 UDP 포트만 사용합니다. 하지만 이 지원은 구성이 가능합니다. 동시 포트를 추가로 사용할 경우 향상된 확장성으로 인해 시스템 성능이 높아진다면 추가 포트를 사용하도록 시스템을 구성할 수 있습니다. 또한 이 기능은 처음부터 이런 유형의 구성 기능이 있는 TCP를 통한 NFS 지원을 미러합니다. 자세한 내용은 [Oracle Solaris 조정 가능 매개변수 참조 설명서](#)를 참조하십시오.

---

주 - NFS 버전 4에서는 UDP가 사용되지 않습니다. `proto=udp` 옵션을 통해 파일 시스템을 마운트하는 경우 NFS 버전 4가 아닌 버전 3이 사용됩니다.

---

## RDMA를 통한 NFS 개요

NFS의 기본 전송 방식은 고속 네트워크를 통한 메모리 간 데이터 전송 기술인 RDMA(Remote Direct Memory Access) 프로토콜입니다. 구체적으로, RDMA는 CPU를 사용하지 않고 메모리에서/메모리로 직접 원격 데이터 전송하는 기능을 제공합니다. 이 기능을 제공하기 위해 RDMA는 InfiniBand-on-SPARC 플랫폼의 상호 연결 I/O 기능과 Oracle Solaris 운영 체제를 결합합니다. 자세한 내용은 [170 페이지 “RDMA를 통한 NFS”](#)를 참조하십시오.

## 네트워크 잠금 관리자 및 NFS

네트워크 잠금 관리자는 NFS 파일에 대한 PC 파일 공유 및 UNIX 레코드 잠금 기능을 제공합니다. NFS 파일에 대한 잠금 방식의 안정성이 높아져 잠금을 사용하는 명령이 정지될 가능성이 낮아집니다.

---

주 - 네트워크 잠금 관리자는 NFS 버전 2 및 버전 3 마운트에만 사용됩니다. NFS 버전 4 프로토콜에서는 파일 잠금 기능이 기본 제공됩니다.

---

## NFS 큰 파일 지원

Solaris 2.6의 NFS 버전 3 프로토콜 구현이 2GB보다 큰 파일을 정상적으로 조작할 수 있도록 변경되었습니다. NFS 버전 2 프로토콜 및 Solaris 2.5의 버전 3 프로토콜 구현에서는 2GB보다 큰 파일을 처리할 수 없었습니다.

## NFS 클라이언트 페일오버

Solaris 2.6 릴리스에는 읽기 전용 파일 시스템의 동적 페일오버 기능이 추가되었습니다. 페일오버는 매뉴얼 페이지, 기타 설명서, 공유 바이너리 등 이미 복제된 읽기 전용 자원의 가용성을 높여 줍니다. 파일 시스템을 마운트한 후 언제든지 페일오버가 수행될 수 있습니다. 이제는 이전 릴리스의 자동 마운트와 마찬가지로 수동 마운트에서도 여러 복제본이 나열될 수 있습니다. 파일 시스템을 다시 마운트할 때까지 페일오버가 기다리지 않아도 된다는 점을 제외하면 자동 마운트는 변경되지 않았습니다. 자세한 내용은 [89 페이지 “클라이언트측 페일오버를 사용하는 방법”](#) 및 [184 페이지 “클라이언트측 페일오버”](#)를 참조하십시오.

## NFS 서비스에 대한 Kerberos 지원

NFS 서비스에서는 Kerberos V4 클라이언트가 지원됩니다. Kerberos V5 인증을 사용하는 NFS 버전 3 마운트를 지원하도록 mount 및 share 명령이 변경되었습니다. 또한 서로 다른 클라이언트에 대해 여러 인증 종류를 사용할 수 있도록 share 명령이 변경되었습니다. 보안 종류와 관련된 변경에 대한 자세한 내용은 [79 페이지 “RPCSEC\\_GSS 보안 종류”](#)를 참조하십시오. Kerberos V5 인증에 대한 내용은 [Oracle Solaris 관리: 보안 서비스의 “Kerberos NFS 서버 구성”](#)을 참조하십시오.

## WebNFS 지원

Solaris 2.6 릴리스에는 인터넷의 파일 시스템을 방화벽을 통해 액세스할 수 있도록 지정하는 기능도 포함되어 있습니다. 이 기능은 NFS 프로토콜 확장을 통해 제공되었습니다. 인터넷 액세스용으로 WebNFS 프로토콜을 사용하는 경우의 이점 중 하나는 안정성입니다. 이 서비스는 NFS 버전 3 및 버전 2 프로토콜의 확장으로 작성되었습니다. 또한 WebNFS 구현에서는 익명 ftp 사이트의 관리 오버헤드 없이 이러한 파일을 공유하는 기능도 제공됩니다. WebNFS 서비스와 관련된 다른 변경 내용에 대한 설명은 [79 페이지 “WebNFS 서비스의 보안 협상”](#)을 참조하십시오. 자세한 작업 정보는 [98 페이지 “WebNFS 관리 작업”](#)을 참조하십시오.



주 - WebNFS 서비스보다 NFS 버전 4 프로토콜이 기본적으로 사용됩니다. NFS 버전 4에는 MOUNT 프로토콜 및 WebNFS 서비스에 추가된 모든 보안 협상 기능이 완전하게 통합되어 있습니다.

## RPCSEC\_GSS 보안 종류

Solaris 7 릴리스에서는 RPCSEC\_GSS라는 보안 종류가 지원됩니다. 이 보안 종류는 표준 GSS-API 인터페이스를 사용하여 인증, 무결성 및 프라이버시를 제공할 뿐 아니라 여러 보안 방식을 지원할 수 있도록 합니다. Kerberos V5 인증 지원에 대한 자세한 내용은 [78 페이지 “NFS 서비스에 대한 Kerberos 지원”](#)을 참조하십시오. GSS-API에 대한 자세한 내용은 [Developer’s Guide to Oracle Solaris 11 Security](#)를 참조하십시오.

## NFS 마운트를 위한 Solaris 7 확장

Solaris 7 릴리스에는 mount 명령 및 automountd 명령에 대한 확장이 포함되어 있습니다. 이러한 확장을 통해 마운트 요청이 MOUNT 프로토콜 대신 공용 파일 핸들을 사용할 수 있습니다. MOUNT 프로토콜은 WebNFS 서비스가 사용하는 것과 같은 액세스 방법입니다. MOUNT 프로토콜을 우회함으로써 방화벽을 통해 마운트를 수행할 수 있습니다. 또한 서버와 클라이언트 간에 수행되어야 하는 트랜잭션 수가 더 적으므로 마운트 속도도 빨라집니다.

또한 확장을 사용하는 경우 표준 경로 이름 대신 NFS URL을 사용할 수 있습니다. 뿐만 아니라 mount 명령과 자동 마운트 맵에서 public 옵션을 사용하여 공용 파일 핸들을 사용하도록 강제 지정할 수 있습니다. WebNFS 서비스의 변경 내용에 대한 자세한 내용은 [78 페이지 “WebNFS 지원”](#)을 참조하십시오.

## WebNFS 서비스의 보안 협상

Solaris 8 릴리스에는 WebNFS 클라이언트가 NFS 서버와 보안 방식을 협상할 수 있는 새로운 프로토콜이 추가되었습니다. 이 프로토콜에서는 WebNFS 서비스 사용 시 보안 트랜잭션을 사용할 수 있습니다. 자세한 내용은 [188 페이지 “WebNFS 보안 협상의 작동 방식”](#)을 참조하십시오.

## NFS 서버 로깅

Solaris 8 릴리스에서는 NFS 서버 로깅 기능을 통해 NFS 서버가 해당 파일 시스템에서 수행된 파일 작업 레코드를 제공할 수 있습니다. 이러한 레코드로는 액세스한 파일, 파일 액세스 시간, 파일에 액세스한 사람 등에 대한 정보가 포함됩니다. 구성 옵션 세트를 통해 이 정보가 포함된 로그 위치를 지정할 수 있습니다. 이러한 옵션을 사용하여 로깅해야

하는 작업을 선택할 수도 있습니다. 이 기능은 NFS 및 WebNFS 클라이언트가 익명 FTP 아카이브를 사용할 수 있도록 하는 사이트의 경우 특히 유용합니다. 자세한 내용은 [84 페이지 “NFS 서버 로깅을 사용으로 설정하는 방법”](#)을 참조하십시오.

---

주 - NFS 버전 4에서는 서버 로깅이 지원되지 않습니다.

---

## autofs 기능

autofs는 로컬 네임스페이스에 지정된 파일 시스템에서 작동합니다. 이 정보는 NIS 또는 로컬 파일에서 유지 관리할 수 있습니다.

완전한 다중 스레드 automountd 버전이 포함되어 있습니다. 이처럼 향상된 기능으로 인해 autofs의 안정성이 높아지며 여러 마운트를 동시에 처리할 수 있어 서버를 사용할 수 없어도 서비스가 정지되지 않습니다.

automountd는 보다 효율적인 주문형 마운트 기능을 제공합니다. 이전 릴리스에서는 파일 시스템이 계층적으로 관련되어 있으면 전체 파일 시스템 세트를 마운트했습니다. 이제는 최상위 파일 시스템만 마운트됩니다. 이 마운트 지점과 관련된 다른 파일 시스템은 필요한 경우 마운트됩니다.

autofs 서비스에서는 간접 맵의 찾아보기 기능을 지원합니다. 이 기능이 지원되므로 사용자는 각 시스템을 실제로 마운트하지 않고도 마운트 가능한 디렉토리를 확인할 수 있습니다. -nobrowse 옵션이 autofs 맵에 추가되어 /net 및 /home 등의 대규모 파일 시스템이 자동으로 찾아보기 가능하도록 지정되지 않습니다. 또한 automount에서 -n 옵션을 사용하여 각 클라이언트에서 autofs 찾아보기 기능을 해제할 수도 있습니다. 자세한 내용은 [112 페이지 “autofs 찾아보기 기능 사용 안함으로 설정”](#)을 참조하십시오.



## 네트워크 파일 시스템 관리(작업)

---

이 장에서는 NFS 서비스 설정, 공유할 새 파일 시스템 추가, 파일 시스템 마운트 등의 NFS 관리 작업을 수행하는 방법에 대한 정보를 제공합니다. 또한 보안 NFS 시스템 및 WebNFS 기능 사용에 대해서도 다룹니다. 이 장의 끝부분에는 문제 해결 절차와 몇 가지 NFS 오류 메시지 및 해당 의미 목록이 포함되어 있습니다.

- 82 페이지 “자동 파일 시스템 공유”
- 85 페이지 “파일 시스템 마운트”
- 91 페이지 “NFS 서비스 설정”
- 96 페이지 “보안 NFS 시스템 관리”
- 98 페이지 “WebNFS 관리 작업”
- 100 페이지 “Autofs 관리 작업 개요”
- 116 페이지 “NFS 문제 해결 전략”
- 117 페이지 “NFS 문제 해결 절차”
- 125 페이지 “NFS 오류 메시지”

NFS 관리자의 작업은 사이트 요구 사항 및 네트워크의 컴퓨터 역할에 따라 달라집니다. 관리자가 로컬 네트워크의 모든 컴퓨터를 관리하는 경우도 있는데, 이 경우 다음과 같은 구성 항목을 결정해야 할 수 있습니다.

- 전용 서버로 지정할 컴퓨터
- 서버인 동시에 클라이언트로 작동해야 하는 컴퓨터
- 클라이언트로만 지정해야 하는 컴퓨터

서버를 설정한 후에 유지 관리할 때는 다음과 같은 작업을 수행합니다.

- 필요에 따라 파일 시스템 공유 및 공유 해제
- 관리 파일을 수정하여 컴퓨터가 공유하거나 자동으로 탑재되는 파일 시스템 목록 업데이트
- 네트워크 상태 확인
- NFS 관련 문제 발생 시 진단 및 해결
- autof 맵 설정

한 컴퓨터가 서버인 동시에 클라이언트가 될 수 있습니다. 따라서 컴퓨터 한 대를 사용하여 원격 컴퓨터와 로컬 파일 시스템을 공유하고 원격 파일 시스템을 마운트할 수 있습니다.

주 - 시스템에서 영역이 사용으로 설정된 경우 비전역 영역에서 이 기능을 사용하려면 **Oracle Solaris 관리: Oracle Solaris Zones, Oracle Solaris 10 Zones 및 리소스 관리**에서 자세한 내용을 참조하십시오.

## 자동 파일 시스템 공유

Oracle Solaris 11 릴리스에서 `share` 명령은 시스템 시작 중에 자동으로 공유되는 영구 공유를 만듭니다. 이전 릴리스와는 달리 이제는 `/etc/dfs/dfstab` 파일을 편집하여 후속 재부트를 위해 공유에 대한 정보를 기록할 필요가 없습니다. `/etc/dfs/dfstab`는 더 이상 사용되지 않습니다.

표 5-1 파일 시스템 공유 작업 맵

작업	설명	수행 방법
자동 파일 시스템 공유 설정	서버를 재부트할 때 파일 시스템이 자동으로 공유되도록 서버를 구성하는 단계	82 페이지 “자동 파일 시스템 공유를 설정하는 방법”
WebNFS 사용으로 설정	사용자가 WebNFS를 통해 파일에 액세스할 수 있도록 서버를 구성하는 단계	83 페이지 “WebNFS 액세스를 사용으로 설정하는 방법”
NFS 서버 로그인 사용으로 설정	선택한 파일 시스템에서 NFS 로그인이 실행되도록 서버를 구성하는 단계	84 페이지 “NFS 서버 로그인을 사용으로 설정하는 방법”

## ▼ 자동 파일 시스템 공유를 설정하는 방법

### 1 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

### 2 공유할 파일 시스템을 정의합니다.

`share` 명령을 사용하여 공유할 각 경로를 정의합니다. 시스템을 재부트할 때 이 정보가 유지됩니다.

```
# share -F nfs -o specific-options pathname
```

`specific-options`의 전체 목록은 [share\\_nfs\(1M\)](#) 매뉴얼 페이지를 참조합니다.

### 3 정보가 정확한지 확인합니다.

share 명령을 실행하여 올바른 옵션이 나열되어 있는지 확인합니다.

```
# share -F nfs
-      /export/share/man    sec=sys,ro    ""
-      /usr/src             sec=sys,rw=eng ""
-      /export/ftp          sec=sys,ro,public ""
```

**참조** 다음 단계에서는 클라이언트가 서버에서 공유하는 파일 시스템에 액세스할 수 있도록 autofs 맵을 설정합니다. 자세한 내용은 [100 페이지 “Autofs 관리 작업 개요”](#)를 참조하십시오.

## ▼ WebNFS 액세스를 사용으로 설정하는 방법

다음 사항에 유의하십시오.

- 기본적으로 NFS를 마운트할 수 있는 모든 파일 시스템은 자동으로 WebNFS 액세스용으로 제공됩니다. 이 절차는 다음과 같은 상황 중 하나에서만 사용해야 합니다.
  - 현재 NFS 마운트가 허용되지 않는 서버에서 NFS 마운트를 허용하려는 경우
  - share 명령에서 public 옵션을 사용하여 NFS URL을 줄이도록 공용 파일 핸들을 재설정하려는 경우
  - share 명령에서 index 옵션을 사용하여 특정 HTML 파일을 강제로 로드하려는 경우
- sharectl 유틸리티를 사용하여 NFS 등의 파일 공유 프로토콜을 구성할 수도 있습니다. [sharectl\(1M\)](#) 매뉴얼 페이지 및 [155 페이지 “sharectl 명령”](#)을 참조하십시오.

WebNFS 서비스를 시작하기 전에 고려해야 하는 문제 목록은 [98 페이지 “WebNFS 액세스 계획”](#)을 참조하십시오.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 WebNFS 서비스에서 공유할 파일 시스템을 정의합니다.

share 명령을 사용하여 각 파일 시스템을 정의합니다. 다음 예제에 표시되어 있는 public 및 index 태그는 선택 사항입니다.

```
# share -F nfs -o ro,public,index=index.html /export/ftp
```

전체 옵션 목록은 [share\\_nfs\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

### 3 정보가 정확한지 확인합니다.

share 명령을 실행하여 올바른 옵션이 나열되어 있는지 확인합니다.

```
# share -F nfs
-      /export/share/man    sec=sys,ro    ""
-      /usr/src             sec=sys,rw=eng ""
-      /export/ftp          sec=sys,ro,public,index=index.html ""
```

## ▼ NFS 서버 로깅을 사용으로 설정하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 (옵션) 파일 시스템 구성 설정을 변경합니다.

/etc/nfs/nfslog.conf에서는 두 가지 방법 중 하나로 설정을 변경할 수 있습니다.global 태그에 연결된 데이터를 변경하여 모든 파일 시스템의 기본 설정을 편집할 수 있습니다. 해당 파일 시스템에 대해 새 태그를 추가할 수도 있습니다. 이와 같이 변경할 필요가 없는 경우에는 이 파일을 변경하지 않아도 됩니다./etc/nfs/nfslog.conf의 형식에 대한 설명은 [nfslog.conf\(4\)](#) 매뉴얼 페이지에 나와 있습니다.

### 3 NFS 서버 로깅을 사용할 파일 시스템을 정의합니다.

share 명령을 사용하여 각 파일 시스템을 정의합니다.log=tag 옵션과 함께 사용되는 태그를 /etc/nfs/nfslog.conf에 입력해야 합니다. 이 예제에서는 global 태그의 기본 설정을 사용합니다.

```
# share -F nfs -ro,log=global /export/ftp
```

### 4 정보가 정확한지 확인합니다.

share 명령을 실행하여 올바른 옵션이 나열되어 있는지 확인합니다.

```
# share -F nfs
-      /export/share/man    sec=sys,ro    ""
-      /usr/src             sec=sys,rw=eng ""
-      /export/ftp          sec=sys,ro,public,log=global ""
```

### 5 nfslogd(NFS 로그 데몬)가 실행 중인지 확인합니다.

```
# ps -ef | grep nfslogd
```

### 6 (옵션) nfslogd가 실행되고 있지 않으면 시작합니다.

```
# svcadm restart network/nfs/server:default
```

# 파일 시스템 마운트

다양한 방법으로 파일 시스템을 마운트할 수 있습니다. 파일 시스템은 시스템 부트 시 자동으로, 명령줄에서 필요 시에 또는 자동 마운트를 통해 마운트할 수 있습니다. 자동 마운트를 사용하는 경우 부트 시에 마운트하거나 명령줄에서 마운트하는 경우에 비해 여러 가지 이점이 있습니다. 그러나 대부분의 경우에는 세 가지 방법을 함께 사용해야 합니다. 또한 파일 시스템 마운트 시 사용하는 옵션에 따라 여러 가지 방법으로 프로세스를 사용으로 설정하거나 사용 안함으로 설정할 수 있습니다. 파일 시스템 마운트와 연관된 작업의 전체 목록은 다음 표를 참조하십시오.

표 5-2 파일 시스템 마운트 작업 맵

작업	설명	수행 방법
부트 시 파일 시스템 마운트	시스템을 재부트할 때마다 파일 시스템이 마운트되도록 하는 단계입니다.	86 페이지 “부트 시 파일 시스템을 마운트하는 방법”.
명령을 사용하여 파일 시스템 마운트	시스템이 실행 중일 때 파일 시스템을 마운트하는 단계입니다. 이 절차는 테스트 시에 유용합니다.	86 페이지 “명령줄에서 파일 시스템을 마운트하는 방법”.
자동 마운트를 통한 마운트	명령줄을 사용하지 않고 필요에 따라 파일 시스템에 액세스하는 단계입니다.	87 페이지 “자동 마운트를 사용한 마운트”.
미러 마운트를 통해 파일 시스템 마운트	미러 마운트를 통해 하나 이상의 파일 시스템을 마운트하는 단계입니다.	Using Mirrormounts After Mounting a File System
미러 마운트를 통해 모든 파일 시스템 마운트	단일 서버에서 모든 파일 시스템을 마운트하는 단계입니다.	87 페이지 “서버에서 모든 파일 시스템을 마운트하는 방법”
큰 파일 방지	파일 시스템에서 큰 파일이 만들어지지 않도록 하는 단계입니다.	88 페이지 “NFS 서버에서 큰 파일을 사용 안함으로 설정하는 방법”.
클라이언트측 페일오버 시작	서버에 오류가 발생하는 경우 작동 중인 파일 시스템으로 자동 스위치오버를 사용으로 설정하는 단계입니다.	89 페이지 “클라이언트측 페일오버를 사용하는 방법”.
클라이언트에 대해 마운트 액세스 사용 안함으로 설정	단일 클라이언트가 원격 파일 시스템에 액세스를 사용 안함으로 설정하는 단계입니다.	89 페이지 “단일 클라이언트에 대한 마운트 액세스를 사용 안함으로 설정하는 방법”.
방화벽을 통해 파일 시스템 액세스 제공	WebNFS 프로토콜을 사용하여 방화벽을 통해 파일 시스템에 액세스하도록 허용하는 단계입니다.	90 페이지 “방화벽을 통해 NFS 파일 시스템을 마운트하는 방법”.
NFS URL을 사용하여 파일 시스템 마운트	NFS URL을 사용하여 파일 시스템 액세스를 허용하는 단계입니다. 이 프로세스를 통해 MOUNT 프로토콜을 사용하지 않고 파일 시스템 액세스를 허용할 수 있습니다.	90 페이지 “NFS URL을 사용하여 NFS 파일 시스템을 마운트하는 방법”.
FedFS 파일 시스템 마운트	/nfs4 마운트 지점을 통해 FedFS 파일 시스템에 액세스할 수 있도록 DNS 레코드를 설정하는 프로세스입니다.	91 페이지 “통합 파일 시스템 서버에 대해 DNS 레코드 설정”.

## ▼ 부트 시 파일 시스템을 마운트하는 방법

autofs 맵을 사용하는 대신 부트 시에 파일 시스템을 마운트하려면 다음 절차를 따릅니다. 원격 파일 시스템에 액세스해야 하는 모든 클라이언트에서 이 절차를 완료해야 합니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 파일 시스템에 대한 항목을 `/etc/vfstab`에 추가합니다.

`/etc/vfstab` 파일에 포함된 항목의 구문은 다음과 같습니다.

```
special fsckdev mountp fstype fsckpass mount-at-boot mntopts
```

자세한 내용은 [vfstab\(4\)](#) 매뉴얼 페이지를 참조하십시오.



주의 - NFS 클라이언트 `vfstab` 항목도 포함하는 NFS 서버의 경우 재부트 중에 시스템이 중단되지 않도록 항상 `bg` 옵션을 지정해야 합니다. 자세한 내용은 [148 페이지 “NFS 파일 시스템용 mount 옵션”](#)을 참조하십시오.

### 예 5-1 클라이언트 `vfstab` 파일의 항목

서버 `wasp`의 `/var/mail` 디렉토리를 클라이언트 컴퓨터에 마운트하려고 합니다. 클라이언트에서 파일 시스템을 `/var/mail`로 마운트하고 클라이언트에 읽기/쓰기 권한을 부여합니다. 다음 항목을 클라이언트의 `vfstab` 파일에 추가합니다.

```
wasp:/var/mail - /var/mail nfs - yes rw
```

## ▼ 명령줄에서 파일 시스템을 마운트하는 방법

새 마운트 지점을 테스트할 때 명령줄에서 파일 시스템을 마운트하는 경우가 많습니다. 이러한 마운트 유형을 사용하는 경우 자동 마운트를 통해 사용할 수 없는 파일 시스템에 일시적으로 액세스할 수 있습니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 파일 시스템을 마운트합니다.

다음 명령을 입력합니다.

```
# mount -F nfs -o ro bee:/export/share/local /mnt
```

이 경우 서버 `bee`의 `/export/share/local` 파일 시스템은 로컬 시스템의 읽기 전용 `/mnt`에 마운트됩니다. 명령줄에서 마운트하면 파일 시스템을 임시로 볼 수 있습니다. `umount`를 사용하여 또는 로컬 호스트를 재부트하여 파일 시스템의 마운트를 해제할 수 있습니다.



**주의** `-mount` 명령의 모든 버전은 잘못된 옵션에 대한 경고를 표시하지 않습니다. 명령은 해석할 수 없는 옵션을 자동으로 무시합니다. 예기치 않은 동작을 방지하려면 사용된 옵션을 모두 확인하십시오.

## 예 5-2 파일 시스템을 마운트한 후 미리 마운트 사용

이 릴리스에는 미리 마운트 기능이 포함되어 있습니다. 이 새 마운트 기술은 NFSv4 서버에서 두번째 파일 시스템에 액세스하는 모든 NFSv4 클라이언트에서 사용할 수 있습니다. `mount` 명령 또는 자동 마운트를 사용하여 서버에서 첫번째 파일 시스템을 마운트하고 나면 해당 마운트 지점에 추가한 파일 시스템에 액세스할 수 있습니다. 파일 시스템 액세스를 시도해 보면 됩니다. 미리 마운트는 자동으로 수행됩니다. 자세한 내용은 193 페이지 “[미리 마운트의 작동 방식](#)”을 참조하십시오.

## 자동 마운트를 사용한 마운트

100 페이지 “[Autofs 관리 작업 개요](#)”에는 자동 마운트를 사용한 마운트 설정 및 지원과 관련한 구체적인 지침이 나와 있습니다. 일반 시스템을 변경하지 않고도 클라이언트는 `/net` 마운트 지점을 통해 원격 파일 시스템에 액세스할 수 있어야 합니다. 이전 예제의 `/export/share/local` 파일 시스템을 마운트하려면 다음을 입력합니다.

```
% cd /net/bee/export/share/local
```

자동 마운트를 사용하면 모든 사용자가 파일 시스템을 마운트할 수 있으므로 `root` 액세스 권한이 필요하지 않습니다. 또한 자동 마운트에서는 파일 시스템 자동 마운트 해제 기능도 제공하므로 작업을 완료한 후에 파일 시스템을 마운트 해제하지 않아도 됩니다.

클라이언트에서 추가 파일 시스템을 마운트하는 방법에 대한 자세한 내용은 [Using Mirrormounts After Mounting a File System](#)을 참조하십시오.

## ▼ 서버에서 모든 파일 시스템을 마운트하는 방법

이 릴리스에는 미리 마운트 기능이 포함되어 있습니다. 이 기능을 사용하면 특정 서버에서 1회의 마운트가 성공한 이후 클라이언트가 해당 서버에서 NFS를 사용하여 공유되는 모든 사용 가능 파일 시스템에 액세스할 수 있습니다. 자세한 내용은 193 페이지 “[미리 마운트의 작동 방식](#)”을 참조하십시오.

**1 관리자가 됩니다.**

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

**2 서버의 내보낸 네임스페이스 루트를 마운트합니다.**

이 명령은 클라이언트에서 서버의 파일 시스템 계층을 미러링합니다. 이 경우 /mnt/export/share/local 디렉토리 구조가 만들어집니다.

```
# mount bee:/ /mnt
```

**3 파일 시스템에 액세스합니다.**

이 명령 또는 파일 시스템에 액세스하는 다른 명령을 통해 파일 시스템이 마운트됩니다.

```
# cd /mnt/export/share/local
```

## ▼ NFS 서버에서 큰 파일을 사용 안함으로 설정하는 방법

2GB보다 큰 파일을 처리할 수 없는 클라이언트를 지원하는 서버의 경우 큰 파일을 만드는 기능을 사용 안함으로 설정해야 할 수 있습니다.

**1 관리자가 됩니다.**

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

**2 파일 시스템에 큰 파일이 없는지 확인합니다.**

예를 들면 다음과 같습니다.

```
# cd /export/home1
# find . -xdev -size +2000000 -exec ls -l {} \;
```

파일 시스템에 큰 파일이 있는 경우 제거하거나 다른 파일 시스템으로 이동해야 합니다.

**3 파일 시스템을 마운트 해제합니다.**

```
# umount /export/home1
```

**4 파일 시스템이 마운트된 경우 largefiles를 사용하여 파일 시스템 상태를 재설정합니다.**

fsck는 파일 시스템에 큰 파일이 없으면 파일 시스템 상태를 재설정합니다.

```
# fsck /export/home1
```

**5 nolargefiles를 사용하여 파일 시스템을 마운트합니다.**

```
# mount -F ufs -o nolargefiles /export/home1
```



명령줄에서 마운트할 수도 있지만, 옵션을 영구적으로 적용하려면 다음과 같은 항목을 `/etc/vfstab`에 추가합니다.

```
/dev/dsk/c0t3d0s1 /dev/rdsk/c0t3d0s1 /export/home1 ufs 2 yes nolargefiles
```

## ▼ 클라이언트측 페일오버를 사용하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 NFS 클라이언트에서 `ro` 옵션을 사용하여 파일 시스템을 마운트합니다.

명령줄에서, 자동 마운트를 통해 또는 다음과 같은 항목을 `/etc/vfstab`에 추가하여 마운트할 수 있습니다.

```
bee,wasp:/export/share/local - /usr/local nfs - no ro
```

이 구문은 자동 마운트에서 허용되었습니다. 그러나 파일 시스템을 마운트하는 동안에는 페일오버를 사용할 수 없었으며 서버를 선택할 때만 사용할 수 있었습니다.

---

주-vfstab 항목에서 또는 명령줄을 사용하여 서로 다른 NFS 프로토콜 버전을 실행하는 서버를 혼합할 수는 없습니다. NFS 버전 2, 버전 3 또는 버전 4 프로토콜을 지원하는 서버를 혼합하려면 autofs를 사용해야 합니다. autofs에서 버전 2, 버전 3, 버전 4 서버에서 가장 좋은 하위 세트가 사용됩니다.

---

## ▼ 단일 클라이언트에 대한 마운트 액세스를 사용 안함으로 설정하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 단일 클라이언트에 대한 마운트 액세스를 사용 안함으로 설정합니다.

```
# share -F nfs ro=-rose:eng /export/share/man
```

`or=-rose:eng`      `rose`라는 호스트를 제외하고는 `eng` 넷 그룹에 있는 모든 클라이언트에 대한 읽기 전용 마운트 액세스를 허용하는 액세스 목록입니다.

`/export/share/man`      공유할 파일 시스템입니다.

## ▼ 방화벽을 통해 NFS 파일 시스템을 마운트하는 방법

방화벽을 통해 파일 시스템에 액세스하려면 다음 절차를 수행합니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 다음과 같은 명령을 사용하여 파일 시스템을 수동으로 마운트합니다.

```
# mount -F nfs bee:/export/share/local /mnt
```

이 예제에서는 공용 파일 핸들을 사용하여 /export/share/local 파일 시스템을 로컬 클라이언트에 마운트합니다. 표준 경로 이름 대신 NFS URL을 사용할 수 있습니다. bee 서버에서 공용 파일 핸들이 지원되지 않는 경우에는 마운트 작업이 실패합니다.

---

주 - 이 절차를 수행하려면 public 옵션을 사용하여 NFS 서버의 파일 시스템을 공유해야 합니다. 또한 클라이언트와 서버 간의 모든 방화벽이 포트 2049에서 TCP 연결을 허용해야 합니다. 공유되는 모든 파일 시스템은 공용 파일 핸들 액세스를 허용하므로 public 옵션이 기본적으로 적용됩니다.

---

## ▼ NFS URL을 사용하여 NFS 파일 시스템을 마운트하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 (옵션) NFS 버전 2 또는 버전 3을 사용하는 경우 다음과 같은 명령을 사용하여 파일 시스템을 수동으로 마운트합니다.

```
# mount -F nfs nfs://bee:3000/export/share/local /mnt
```

이 예제에서는 NFS 포트 번호 3000을 사용하여 bee 서버에서 /export/share/local 파일 시스템을 마운트합니다. 포트 번호는 필요하지 않으며 기본적으로 표준 NFS 포트 번호 2049가 사용됩니다. NFS URL과 함께 public 옵션을 포함할 수 있습니다. public 옵션이 없는 경우 서버에서 공용 파일 핸들이 지원되지 않으면 MOUNT 프로토콜이 사용됩니다. public 옵션은 공용 파일 핸들을 강제로 사용하도록 하며, 공용 파일 핸들이 지원되지 않는 경우에는 마운트가 실패합니다.

### 3 (옵션) NFS 버전 4를 사용하는 경우 다음과 같은 명령을 사용하여 파일 시스템을 수동으로 마운트합니다.

```
# mount -F nfs -o vers=4 nfs://bee:3000/export/share/local /mnt
```

## 통합 파일 시스템 서버에 대해 DNS 레코드 설정

적절한 DNS 레코드를 만든 후 마운트 지점에 액세스하면 자동 마운트에서 통합 파일 시스템 마운트를 완료합니다. 서버의 DNS 레코드는 다음과 같습니다.

```
% nslookup -q=svr_nfs4._domainroot._tcp.example.com bee.example.com
Server:      bee.example.com
Address:     192.168.1.1

_svr_nfs4._domainroot._tcp.example.com      service = 1 0 2049 bee.example.com.
```

## NFS 서비스 설정

이 절에서는 다음을 수행하는 데 필요한 몇 가지 작업에 대해 설명합니다.

- NFS 서버 시작 및 중지
- 자동 마운트 시작 및 중지
- 다른 NFS 버전 선택

주 - Solaris 10 릴리스부터는 NFS 버전 4가 기본값입니다.

표 5-3 NFS 서비스의 작업 맵

작업	설명	수행 방법
NFS 서버 시작	NFS 서비스가 자동으로 시작되지 않은 경우 시작하는 단계입니다.	92 페이지 “NFS 서비스를 시작하는 방법”
NFS 서버 중지	NFS 서비스를 중지하는 단계입니다. 일반적으로는 서비스를 중지할 필요가 없습니다.	92 페이지 “NFS 서비스를 중지하는 방법”
자동 마운트 시작	자동 마운트를 시작하는 단계입니다. 자동 마운트 맵 중 일부가 변경된 경우 이 절차를 수행해야 합니다.	92 페이지 “자동 마운트를 시작하는 방법”
자동 마운트 중지	자동 마운트를 중지하는 단계입니다. 자동 마운트 맵 중 일부가 변경된 경우 이 절차를 수행해야 합니다.	93 페이지 “자동 마운트를 중지하는 방법”
서버에서 다른 NFS 버전 선택	서버에서 다른 NFS 버전을 선택하는 단계입니다. NFS 버전 4를 사용하지 않도록 선택하는 경우 이 절차를 수행합니다.	93 페이지 “서버에서 다른 NFS 버전을 선택하는 방법”
클라이언트에서 다른 NFS 버전 선택	SMF 매개변수를 수정하여 클라이언트에서 다른 NFS 버전을 선택하는 단계입니다. NFS 버전 4를 사용하지 않도록 선택하는 경우 이 절차를 수행합니다.	94 페이지 “클라이언트에서 다른 NFS 버전을 선택하는 방법”

표 5-3 NFS 서비스의 작업 맵 (계속)

작업	설명	수행 방법
	명령줄을 사용하여 클라이언트에서 다른 NFS 버전을 선택하는 대체 단계입니다. NFS 버전 4를 사용하지 않도록 선택하는 경우 이 대체 절차를 수행합니다.	95 페이지 “mount 명령을 사용하여 클라이언트에서 다른 NFS 버전을 선택하는 방법”

## ▼ NFS 서비스를 시작하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 서버에서 NFS 서비스를 사용으로 설정합니다.

다음 명령을 입력합니다.

```
# svcadm enable network/nfs/server
```

이 명령은 NFS 서비스를 사용으로 설정합니다.

주 - 시스템을 부트하면 NFS 서버가 자동으로 시작됩니다. 또한 시스템을 부트한 후에는 언제든지 NFS 파일 시스템을 공유하여 NFS 서비스 데몬을 자동으로 사용으로 설정할 수 있습니다. [82 페이지 “자동 파일 시스템 공유를 설정하는 방법”](#)을 참조하십시오.

## ▼ NFS 서비스를 중지하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 서버에서 NFS 서비스를 사용 안함으로 설정합니다.

다음 명령을 입력합니다.

```
# svcadm disable network/nfs/server
```

## ▼ 자동 마운트를 시작하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 autofs 데몬을 사용으로 설정합니다.

다음 명령을 입력합니다.

```
# svcadm enable system/filesystem/autofs
```

## ▼ 자동 마운트를 중지하는 방법

- 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 autofs 데몬을 사용 안함으로 설정합니다.

다음 명령을 입력합니다.

```
# svcadm disable system/filesystem/autofs
```

## ▼ 서버에서 다른 NFS 버전을 선택하는 방법

NFS 버전 4를 사용하지 않도록 선택하는 경우 이 절차를 수행합니다.

- 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 SMF 매개변수를 변경하여 NFS 버전 번호를 설정합니다.

예를 들어 서버에서 NFS 버전 3만 제공하도록 하려면 아래와 같이 `server_versmax` 및 `server_versmin`의 값을 모두 3으로 설정합니다.

```
# sharectl set -p server_versmax=3 nfs
# sharectl set -p server_versmin=3 nfs
```

---

주 - 기본적으로는 NFS 버전 4가 기본 버전으로 설정됩니다.

---

- 3 (옵션) 서버 위임을 사용 안함으로 설정합니다.

서버 위임을 사용 안함으로 설정하려면 `server_delegation` 등록 정보를 변경합니다.

```
# sharectl set -p server_delegation=on nfs
```

---

주 - NFS 버전 4에서는 서버 위임은 기본적으로 사용으로 설정됩니다. 자세한 내용은 [179 페이지 “NFS 버전 4의 위임”](#)을 참조하십시오.

---

**4 (옵션) 공통 도메인을 설정합니다.**

클라이언트와 서버에 대해 공통 도메인을 설정하려면 `nfsmapid_domain` 등록 정보를 변경합니다.

```
# sharectl set -p server_nfsmapid_domain=my.comany.com
```

`my.comany.com`      공통 도메인 이름 지정

자세한 내용은 [139 페이지 “nfsmapid 데몬”](#)을 참조하십시오.

**5 NFS 서비스가 서버에서 실행 중인지 확인합니다.**

다음 명령을 입력합니다.

```
# svcs network/nfs/server
```

이 명령은 NFS 서버 서비스가 온라인인지 아니면 사용 안함으로 설정되어 있는지를 보고합니다.

**6 (옵션) 필요한 경우 NFS 서비스를 사용 안함으로 설정합니다.**

이전 단계를 통해 NFS 서비스가 온라인 상태임이 확인되면 다음 명령을 입력하여 서비스를 사용 안함으로 설정합니다.

```
# svcadm disable network/nfs/server
```

---

주 - NFS 서비스를 구성해야 하는 경우 [82 페이지 “자동 파일 시스템 공유를 설정하는 방법”](#)을 참조하십시오.

---

**7 NFS 서비스를 사용으로 설정합니다.**

서비스를 사용으로 설정하려면 다음 명령을 입력합니다.

```
# svcadm enable network/nfs/server
```

참조 [172 페이지 “NFS의 버전 협상”](#)

## ▼ 클라이언트에서 다른 NFS 버전을 선택하는 방법

다음 절차에서는 클라이언트에서 사용되는 NFS 버전을 제어하는 방법을 보여줍니다.

**1 관리자가 됩니다.**

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

## 2 SMF 매개변수를 변경하여 NFS 버전 번호를 설정합니다.

예를 들어 서버에서 NFS 버전 3만 제공하도록 하려면 아래와 같이 `client_versmax` 및 `client_versmin`의 값을 모두 3으로 설정합니다.

```
# sharectl set -p client_versmax=3 nfs
# sharectl set -p client_versmin=3 nfs
```

---

주 - 기본적으로는 NFS 버전 4가 기본 버전으로 설정됩니다.

---

## 3 클라이언트에서 NFS를 마운트합니다.

다음 명령을 입력합니다.

```
# mount server-name:/share-point /local-dir
```

`server-name` 서버의 이름을 입력합니다.

`/share-point` 공유할 원격 디렉토리의 경로를 입력합니다.

`/local-dir` 로컬 마운트 지점의 경로를 입력합니다.

참조 172 페이지 “NFS의 버전 협상”

## ▼ mount 명령을 사용하여 클라이언트에서 다른 NFS 버전을 선택하는 방법

다음 절차에서는 `mount` 명령을 사용하여 클라이언트에서 특정 마운트에 대해 사용되는 NFS 버전을 제어하는 방법을 보여줍니다. 클라이언트에서 마운트하는 모든 파일 시스템에 대해 NFS 버전을 수정하려면 94 페이지 “클라이언트에서 다른 NFS 버전을 선택하는 방법”을 참조하십시오.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 클라이언트에서 원하는 NFS 버전을 마운트합니다.

다음 명령을 입력합니다.

```
# mount -o vers=value server-name:/share-point /local-dir
```

`value` 버전 번호를 입력합니다.

`server-name` 서버의 이름을 입력합니다.

`/share-point` 공유할 원격 디렉토리의 경로를 입력합니다.

`/local-dir` 로컬 마운트 지점의 경로를 입력합니다.

---

주 - 이 명령은 NFS 프로토콜을 사용하여 원격 디렉토리를 마운트하고 SMF 저장소에서 클라이언트 설정을 대체합니다.

---

참조 172 페이지 “NFS의 버전 협상”

## 보안 NFS 시스템 관리

보안 NFS 시스템을 사용하려면 모든 관리 대상 컴퓨터에 도메인 이름이 있어야 합니다. 일반적으로 도메인은 대규모 네트워크에 속한 여러 컴퓨터의 관리 엔티티입니다. 이름 서비스를 실행 중인 경우에는 도메인에 대해서도 이름 서비스를 설정해야 합니다.

**Oracle Solaris Administration: Naming and Directory Services** 를 참조하십시오.

NFS 서비스에서는 Kerberos V5 인증을 지원합니다. Kerberos 서비스에 대한 설명은 **Oracle Solaris 관리: 보안 서비스의 19 장, “Kerberos 서비스 소개”**에 나와 있습니다.

Diffie-Hellman 인증을 사용하도록 보안 NFS 환경을 구성할 수도 있습니다. 이 인증 서비스에 대한 설명은 **Oracle Solaris 관리: 보안 서비스의 14 장, “네트워크 서비스 인증(작업)”**에 나와 있습니다.

### ▼ DH 인증을 사용하여 보안 NFS 환경을 설정하는 방법

- 1 도메인에 도메인 이름을 지정하고 도메인의 각 컴퓨터가 도메인 이름을 확인할 수 있도록 합니다.
- 2 클라이언트 사용자용으로 공개 키와 암호 키를 설정합니다.

`newkey` 또는 `nisaddcred` 명령을 사용합니다. 각 사용자가 `chkey` 명령을 사용하여 자신의 보안 RPC 암호를 직접 설정하도록 합니다.

---

주 - 이러한 명령에 대한 자세한 내용은 `newkey(1M)`, `nisaddcred(1M)` 및 `chkey(1)` 매뉴얼 페이지를 참조하십시오.

---

공개 키 및 암호 키를 생성하면 해당 공개 키 및 암호화된 암호 키는 `publickey` 데이터베이스에 저장됩니다.

- 3 이름 서비스가 응답하는지 확인합니다.



예를 들면 다음과 같습니다.

- NIS를 실행 중인 경우 ypbind 데몬이 실행 중인지 확인합니다.

#### 4 키 서버의 keyerv 데몬이 실행 중인지 확인합니다.

다음 명령을 입력합니다.

```
# ps -ef | grep keyerv
root    100      1  16   Apr 11 ?          0:00 /usr/sbin/keyerv
root    2215    2211   5 09:57:28 pts/0    0:00 grep keyerv
```

데몬이 실행 중이지 않으면 다음을 입력하여 키 서버를 시작합니다.

```
# /usr/sbin/keyerv
```

#### 5 암호 키의 암호를 해독하고 키를 저장합니다.

일반적으로 로그인 암호는 네트워크 암호와 같습니다. 이 경우 keylogin은 필요하지 않습니다. 암호가 다른 경우에는 사용자가 로그인하여 keylogin을 실행해야 합니다. /etc/.rootkey에 암호가 해독된 암호 키를 저장하려면 keylogin -r 명령을 root로 사용해야 합니다.

---

주 - 루트 암호 키가 변경되거나 /etc/.rootkey가 손실된 경우에는 keylogin -r을 실행해야 합니다.

---

#### 6 공유할 파일 시스템에 대해 보안 모드를 설정합니다.

Diffie-Hellman 인증의 경우에는 명령줄에 sec=dh 옵션을 추가합니다.

```
# share -F nfs -o sec=dh /export/home
```

보안 모드에 대한 자세한 내용은 [nfssec\(5\)](#) 매뉴얼 페이지를 참조하십시오.

#### 7 파일 시스템의 자동 마운트 맵을 업데이트합니다.

auto\_master 데이터를 편집하여 sec=dh를 Diffie-Hellman 인증의 해당 항목에 마운트 옵션으로 포함합니다.

```
/home    auto_home    -nosuid,sec=dh
```

컴퓨터를 다시 설치하거나 이동하거나 업그레이드할 때 root에 대해 키를 변경하거나 새 키를 설정하지 않는 경우에는 /etc/.rootkey를 저장해야 합니다. /etc/.rootkey를 삭제하는 경우 언제든지 다음을 입력하면 됩니다.

```
# keylogin -r
```

## WebNFS 관리 작업

이 절에서는 WebNFS 시스템 관리 지침을 제공합니다. 관련 작업은 다음과 같습니다.

표 5-4 WebNFS 관리 작업 맵

작업	설명	수행 방법
WebNFS 계획	WebNFS 서비스를 사용으로 설정하기 전에 고려해야 하는 문제점입니다.	98 페이지 “WebNFS 액세스 계획”
WebNFS 사용으로 설정	WebNFS 프로토콜을 사용하여 NFS 파일 시스템을 마운트할 수 있도록 설정하는 단계입니다.	83 페이지 “WebNFS 액세스를 사용으로 설정하는 방법”
방화벽을 통해 WebNFS 사용	WebNFS 프로토콜을 사용하여 방화벽을 통해 파일에 액세스하도록 허용하는 단계입니다.	100 페이지 “방화벽을 통해 WebNFS 액세스를 사용으로 설정하는 방법”
NFS URL을 사용하여 찾아보기	웹 브라우저 내에서 NFS URL을 사용하는 지침입니다.	99 페이지 “NFS URL을 사용한 찾아보기 방법”
autofs와 함께 공용 파일 핸들 사용	자동 마운트를 사용하여 파일 시스템을 마운트할 때 공용 파일 핸들을 강제로 사용하는 단계입니다.	112 페이지 “autofs와 함께 공용 파일 핸들을 사용하는 방법”
autofs와 함께 NFS URL 사용	자동 마운트 맵에 NFS URL을 추가하는 단계입니다.	112 페이지 “autofs와 함께 NFS URL을 사용하는 방법”
방화벽을 통해 파일 시스템 액세스 제공	WebNFS 프로토콜을 사용하여 방화벽을 통해 파일 시스템에 액세스하도록 허용하는 단계입니다.	90 페이지 “방화벽을 통해 NFS 파일 시스템을 마운트하는 방법”
NFS URL을 사용하여 파일 시스템 마운트	NFS URL을 사용하여 파일 시스템 액세스를 허용하는 단계입니다. 이 프로세스를 통해 MOUNT 프로토콜을 사용하지 않고 파일 시스템 액세스를 허용할 수 있습니다.	90 페이지 “NFS URL을 사용하여 NFS 파일 시스템을 마운트하는 방법”

## WebNFS 액세스 계획

WebNFS를 사용하려면 `nfs://server/path`와 같은 NFS URL을 로드 및 실행할 수 있는 응용 프로그램이 필요합니다. 다음 단계에서는 WebNFS 액세스를 위해 내보낼 수 있는 파일 시스템을 선택합니다. 응용 프로그램에서 웹을 검색하는 경우에는 웹 서버의 문서 루트가 사용되는 경우가 많습니다. WebNFS 액세스를 위해 내보낼 파일 시스템을 선택할 때는 다양한 요인을 고려해야 합니다.

1. 각 서버에는 기본적으로 서버의 루트 파일 시스템과 연결되는 공용 파일 핸들이 하나 있습니다. 공용 파일 핸들이 연결된 디렉토리에 상대적으로 NFS URL의 경로를 평가합니다. 경로가 내보낸 파일 시스템 내의 디렉토리나 파일로 연결되는 경우에는 서버에서 액세스를 제공합니다. `share` 명령의 `public` 옵션을 사용하여 공용 파일 핸들을 내보낸 특정 디렉토리나 연결할 수 있습니다. 이 옵션을 사용하면 서버의 루트 파일 시스템이 아닌 공유 파일 시스템에 상대적으로 URL을 지정할 수 있습니다. 루트 파일 시스템은 공유하는 경우가 아니면 웹 액세스를 허용하지 않습니다.

2. WebNFS 환경에서는 이미 마운트 권한을 보유한 사용자가 브라우저를 통해 파일에 액세스할 수 있습니다. `public` 옵션을 사용하여 파일 시스템을 내보내는지 여부에 관계없이 이 기능은 사용으로 설정됩니다. 사용자는 NFS 설정을 통해 이미 이러한 파일에 대한 액세스 권한을 가지고 있으므로, 이 액세스로 인해 보안 위험이 추가로 발생하지는 않습니다. 파일 시스템을 마운트할 수 없는 사용자가 WebNFS 액세스를 사용해야 하는 경우에만 `public` 옵션을 사용하여 파일 시스템을 공유하면 됩니다.
3. 이미 공개적으로 사용할 수 있도록 설정되어 있는 파일 시스템에서 `public` 옵션을 사용할 수 있습니다. `ftp` 아카이브의 최상위 디렉토리, 웹 사이트의 기본 URL 디렉토리 등을 예로 들 수 있습니다.
4. `share` 명령에서 `index` 옵션을 사용하여 HTML 파일을 강제로 로드할 수 있습니다. 그렇지 않으면 NFS URL에 액세스할 때 디렉토리를 나열할 수 있습니다.

파일 시스템을 선택한 후에 파일을 검토하고 액세스 권한을 설정하여 필요에 따라 파일 또는 디렉토리 보기를 제한합니다. 공유하는 NFS 파일 시스템에 대해 권한을 적절하게 설정합니다. 대부분의 사이트에서는 디렉토리의 경우 755 권한, 그리고 파일의 경우 644 권한이 적절한 액세스 레벨을 제공합니다.

단일 웹 사이트에 액세스하는 데 NFS 및 HTTP URL을 모두 사용하려는 경우에는 추가적인 요인을 고려해야 합니다. 이러한 요인에 대한 설명은 [189 페이지 “웹 브라우저 사용 시의 WebNFS 제한”](#)에 나와 있습니다.

## NFS URL을 사용한 찾아보기 방법

WebNFS 서비스를 지원할 수 있는 브라우저에서는 다음과 같은 NFS URL에 대한 액세스를 제공합니다.

`nfs://server<:port>/path`

*server*      파일 서버의 이름

*port*        사용할 포트 번호(2049, 기본값)

*path*        파일 경로(공용 파일 핸들 또는 루트 파일 시스템에 상대적일 수 있음)

주 - 대부분의 브라우저에서는 `nfs` 또는 `http`와 같은 URL 서비스 유형이 트랜잭션 간에 저장됩니다. 다른 서비스 유형이 포함된 URL을 로드하면 예외가 발생합니다. NFS URL을 사용하고 나면 HTTP URL에 대한 참조가 로드될 수 있습니다. 이러한 참조가 로드되면 후속 페이지는 NFS 프로토콜이 아닌 HTTP 프로토콜을 사용하여 로드됩니다.

# 방화벽을 통해 WebNFS 액세스를 사용으로 설정하는 방법

포트 2049에서 TCP 연결을 허용하도록 방화벽을 구성하여 로컬 서브넷에 속하지 않은 클라이언트에 대해 WebNFS 액세스를 사용으로 설정할 수 있습니다. httpd에 대한 액세스를 허용한다고 해서 NFS URL 사용이 허용되지는 않습니다.

## Autofs 관리 작업 개요

이 절에서는 환경에서 수행할 수 있는 몇 가지 일반적인 작업에 대해 설명합니다. 클라이언트의 요구를 가장 효율적으로 충족할 수 있도록 autofs를 구성하는 데 도움이 되는 권장 절차가 각 시나리오에 포함되어 있습니다.

주 - SMF 저장소의 매개변수를 사용하여 autofs 환경을 구성할 수도 있습니다. 작업 정보는 [102 페이지 “SMF 매개변수를 사용하여 autofs 환경 구성”](#)을 참조하십시오.

## Autofs 관리 작업 맵

아래 표에는 autofs와 관련된 대부분의 작업에 대한 설명 및 포인터가 나와 있습니다.

표 5-5 Autofs 관리 작업 맵

작업	설명	수행 방법
autofs 시작	시스템을 재부트하지 않고 자동 마운트 서비스 시작	<a href="#">92 페이지 “자동 마운트를 시작하는 방법”</a>
autofs 중지	다른 네트워크 서비스를 사용 안함으로 설정하지 않고 자동 마운트 서비스 중지	<a href="#">93 페이지 “자동 마운트를 중지하는 방법”</a>
autofs SMF 매개변수를 사용하여 autofs 환경 구성	SMF 저장소에서 매개변수에 값 지정	<a href="#">102 페이지 “SMF 매개변수를 사용하여 autofs 환경 구성”</a>
autofs를 사용하여 파일 시스템 액세스	자동 마운트 서비스를 사용하여 파일 시스템 액세스	<a href="#">87 페이지 “자동 마운트를 사용한 마운트”</a>
autofs 맵 수정	다른 맵을 나열하는 데 사용되는 마스터 맵을 수정하는 단계입니다.  대부분의 맵에 사용되는 간접 맵을 수정하는 단계입니다.	<a href="#">104 페이지 “마스터 맵을 수정하는 방법”</a>  <a href="#">104 페이지 “간접 맵을 수정하는 방법”</a>

표 5-5 Autofs 관리 작업 맵 (계속)

작업	설명	수행 방법
	서버와 클라이언트의 마운트 지점을 직접 연결해야 하는 경우 사용되는 직접 맵을 수정하는 단계입니다.	104 페이지 “직접 맵을 수정하는 방법”
비 NFS 파일 시스템 액세스를 위해 autofs 맵 수정	CD-ROM 응용 프로그램의 항목으로 autofs 맵을 설정하는 단계입니다.  PC-DOS 응용 프로그램의 항목으로 autofs 맵을 설정하는 단계입니다.	106 페이지 “autofs를 사용하여 CD-ROM 응용 프로그램에 액세스하는 방법”  106 페이지 “autofs를 사용하여 PC-DOS 데이터 디스켓에 액세스하는 방법”
/home 사용	공통 /home 맵을 설정하는 방법의 예제입니다.  여러 파일 시스템을 참조하는 /home 맵을 설정하는 단계입니다.	107 페이지 “/home의 공통 보기 설정”  107 페이지 “여러 홈 디렉토리 파일 시스템을 사용하여 /home을 설정하는 방법”
새 autofs 마운트 지점 사용	프로젝트 관련 autofs 맵을 설정하는 단계입니다.  다른 클라이언트 구조를 지원하는 autofs 맵을 설정하는 단계입니다.  다른 운영 체제를 지원하는 autofs 맵을 설정하는 단계입니다.	108 페이지 “/ws 아래에서 프로젝트 관련 파일을 통합하는 방법”  110 페이지 “공유 네임스페이스에 액세스하도록 서로 다른 구조를 설정하는 방법”  111 페이지 “호환되지 않는 클라이언트 운영 체제 버전을 지원하는 방법”
autofs를 사용하여 파일 시스템 복제	폐일오버되는 파일 시스템 액세스 제공	111 페이지 “여러 서버에서 공유 파일을 복제하는 방법”
autofs를 통해 보안 제한 사용	파일에 대한 원격 root 액세스는 제한하면서 파일 시스템 액세스를 제공합니다.	111 페이지 “autofs 보안 제한을 적용하는 방법”
autofs와 함께 공용 파일 핸들 사용	파일 시스템을 마운트할 때 공용 파일 핸들을 강제로 사용합니다.	112 페이지 “autofs와 함께 공용 파일 핸들을 사용하는 방법”
autofs와 함께 NFS URL 사용	자동 마운트에서 사용할 수 있도록 NFS URL을 추가합니다.	112 페이지 “autofs와 함께 NFS URL을 사용하는 방법”
autofs 찾아보기 기능 사용 안함으로 설정	단일 클라이언트에서 autofs 마운트 지점이 자동으로 채워지지 않도록 찾아보기 기능을 사용 안함으로 설정하는 단계입니다.  모든 클라이언트에서 autofs 마운트 지점이 자동으로 채워지지 않도록 찾아보기 기능을 사용 안함으로 설정하는 단계입니다.	113 페이지 “단일 NFS 클라이언트에서 autofs 찾아보기 기능을 완전히 사용 안함으로 설정하는 방법”  113 페이지 “모든 클라이언트에 대해 autofs 찾아보기 기능을 사용 안함으로 설정하는 방법”

표 5-5 Autofs 관리 작업 맵 (계속)

작업	설명	수행 방법
	단일 클라이언트에서 특정 autofs 마운트 지점이 자동으로 채워지지 않도록 찾아보기 기능을 사용 안함으로 설정하는 단계입니다.	114 페이지 “선택한 파일 시스템에 대해 autofs 찾아보기 기능을 사용 안함으로 설정하는 방법”

## SMF 매개변수를 사용하여 autofs 환경 구성

SMF 매개변수를 사용하여 autofs 환경을 구성할 수 있습니다. 구체적으로 이 기능은 autofs 명령 및 autofs 데몬을 구성하는 방법을 추가로 제공합니다. 명령줄에서 지정할 수 있는 항목을 sharectl 명령을 통해서도 지정할 수 있습니다. 명령을 통해 지정하려면 키워드에 값을 제공하면 됩니다.

다음 절차에서는 sharectl 명령을 사용하여 autofs 매개변수를 관리하는 방법을 보여줍니다.

### ▼ SMF 매개변수를 사용하여 autofs 환경을 구성하는 방법

#### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

#### 2 autofs SMF 매개변수를 추가 또는 수정합니다.

예를 들어 모든 autofs 마운트 지점에 대해 찾아보기를 해제하려면 다음 명령을 사용합니다.

```
# sharectl set -p nobrowse=on autofs
```

nobrowse 키워드는 automountd의 -n 옵션에 해당합니다.

#### 3 autofs 데몬을 다시 시작합니다.

다음 명령을 입력합니다.

```
# svcadm restart system/filesystem/autofs
```

## 맵 관련 관리 작업

아래 표에는 autofs 맵을 관리할 때 주의해야 하는 몇 가지 요인에 대한 설명이 나와 있습니다. 선택한 맵 및 이름 서비스에 따라 autofs 맵을 변경하는 데 사용해야 하는 방식이 달라집니다.

아래 표에는 맵의 유형과 해당 용도에 대한 설명이 나와 있습니다.

표 5-6 autofs 맵의 유형 및 해당 용도

맵 유형	용도
마스터	디렉토리를 맵과 연결합니다.
직접	autofs를 특정 파일 시스템에 연결합니다.
간접	autofs를 참조 방식 파일 시스템에 연결합니다.

아래 표에서는 이름 서비스를 기반으로 autofs 환경을 변경하는 방법에 대해 설명합니다.

표 5-7 맵 유지 관리

이름 서비스	방법
로컬 파일	텍스트 편집기
NIS	make 파일

다음 표에서는 맵 유형 수정에 따라 automount 명령을 실행하는 경우에 대해 설명합니다. 예를 들어 직접 맵에 내용을 추가하거나 맵에서 내용을 삭제한 경우에는 로컬 시스템에서 automount 명령을 실행해야 합니다. 명령을 실행하면 변경 내용이 적용됩니다. 그러나 기존 항목을 수정한 경우에는 automount 명령을 실행하지 않아도 변경 내용이 적용됩니다.

표 5-8 automount 명령을 실행하는 경우

맵 유형	automount 다시 시작 여부	
	추가/삭제	수정
auto_master	Y	Y
direct	Y	N
indirect	N	N

## 맵 수정

다음 절차에서는 여러 자동 마운트 맵 유형을 업데이트하는 방법을 보여줍니다.

## ▼ 마스터 맵을 수정하는 방법

- 1 맵 변경 권한이 있는 사용자로 로그인합니다.
- 2 마스터 맵을 변경합니다.  
맵을 변경하기 위해 수행해야 하는 특정 단계는 사용 중인 이름 서비스에 따라 다릅니다.
- 3 각 클라이언트에 대해 관리자 권한을 얻습니다.  
자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.
- 4 각 클라이언트에 대해 `automount` 명령을 실행하여 변경 내용이 적용되었는지 확인합니다.
- 5 변경 내용을 사용자에게 알립니다.  
또한 사용자가 자신의 컴퓨터에서 슈퍼 유저로 `automount` 명령을 실행할 수 있도록 알림을 제공해야 합니다. `automount` 명령은 실행될 때마다 마스터 맵에서 정보를 수집합니다.

## ▼ 간접 맵을 수정하는 방법

- 1 맵 변경 권한이 있는 사용자로 로그인합니다.
- 2 간접 맵을 변경합니다.  
맵을 변경하기 위해 수행해야 하는 특정 단계는 사용 중인 이름 서비스에 따라 다릅니다.

## ▼ 직접 맵을 수정하는 방법

- 1 맵 변경 권한이 있는 사용자로 로그인합니다.
- 2 직접 맵을 변경합니다.  
맵을 변경하기 위해 수행해야 하는 특정 단계는 사용 중인 이름 서비스에 따라 다릅니다.
- 3 변경 내용을 사용자에게 알립니다.  
사용자가 필요한 경우 자신의 컴퓨터에서 슈퍼 유저로 `automount` 명령을 실행할 수 있도록 알림을 제공해야 합니다.



주 - 기존 직접 맵 항목의 내용을 수정하거나 변경하는 경우에는 `automount` 명령을 실행하지 않아도 됩니다.

예를 들어 `/usr/src` 디렉토리가 다른 서버에서 마운트되도록 `auto_direct` 맵을 수정한다고 가정해 보겠습니다. 이때 `/usr/src`가 마운트되어 있지 않으면 `/usr/src`에 액세스할 때 새 항목이 즉시 적용됩니다. `/usr/src`가 마운트되어 있는 경우에는 자동 마운트 해제가 수행될 때까지 기다렸다가 파일에 액세스하면 됩니다.

주 - 가능한 경우에는 항상 간접 맵을 사용하십시오. 간접 맵은 보다 쉽게 생성할 수 있으며 컴퓨터 파일 시스템에 주는 부담이 적습니다. 또한 간접 맵은 마운트 테이블에서 직접 맵만큼 많은 공간을 차지하지 않습니다.

## 마운트 지점 충돌 방지

로컬 디스크 파티션이 `/src`에 마운트되어 있는 상태에서 `autofs` 서비스를 사용하여 다른 원본 디렉토리를 마운트하려는 경우에는 문제가 발생할 수 있습니다. 마운트 지점 `/src`를 지정하는 경우 로컬 파티션에 연결할 때마다 NFS 서비스에서 해당 파티션을 숨깁니다.

따라서 파티션을 `/export/src` 등의 다른 위치에 마운트해야 합니다. 그런 후에 다음과 같은 항목을 `/etc/vfstab`에 추가해야 합니다.

```
/dev/dsk/d0t3d0s5 /dev/rdisk/c0t3d0s5 /export/src ufs 3 yes -
```

`auto_src`에도 이 항목이 필요합니다.

```
terra          terra:/export/src
```

`terra`는 컴퓨터의 이름입니다.

## 비 NFS 파일 시스템 액세스

`Autofs`는 NFS 파일 외의 파일도 마운트할 수 있습니다. `Autofs`는 디스켓이나 CD-ROM 등의 이동식 매체에 파일을 마운트합니다. 일반적으로는 `Volume Manager`를 사용하여 이동식 매체에 파일을 마운트합니다. 다음 예제에서는 `autofs`를 통해 이 마운트를 수행하는 방법을 보여줍니다. `Volume Manager` 및 `autofs`는 함께 작동하지 않으므로 먼저 `Volume Manager`를 비활성화해야 이러한 항목을 사용할 수 있습니다.

서버에서 파일 시스템을 마운트하는 대신 매체를 드라이브에 배치하고 맵에서 파일 시스템을 참조합니다. 비 NFS 파일 시스템에 액세스하려는 경우 `autofs`를 사용 중이라면 다음 절차를 참조하십시오.

## ▼ autofs를 사용하여 CD-ROM 응용 프로그램에 액세스하는 방법

---

주 - Volume Manager를 사용하고 있지 않은 경우 이 절차를 수행합니다.

---

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 autofs 맵을 업데이트합니다.

다음과 같이 CD-ROM 파일 시스템에 대한 항목을 추가합니다.

```
hsfs      -fstype=hsfs,ro      :/dev/sr0
```

마운트하려는 CD-ROM 장치는 콜론 다음의 이름으로 표시되어야 합니다.

## ▼ autofs를 사용하여 PC-DOS 데이터 디스켓에 액세스하는 방법

---

주 - Volume Manager를 사용하고 있지 않은 경우 이 절차를 수행합니다.

---

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 autofs 맵을 업데이트합니다.

다음과 같이 디스켓 파일 시스템에 대한 항목을 추가합니다.

```
pcfs      -fstype=pcfs      :/dev/diskette
```

## 자동 마운트 사용자 정의

다양한 방식으로 자동 마운트 맵을 설정할 수 있습니다. 다음 작업에서는 쉽게 사용할 수 있는 디렉토리 구조를 제공하기 위해 자동 마운트 맵을 사용자 정의하는 방법을 상세하게 설명합니다.

## /home의 공통 보기 설정

모든 네트워크 사용자가 /home 아래에서 자기 자신이나 다른 사용자의 홈 디렉토리를 찾을 수 있어야 합니다. 이 보기는 모든 컴퓨터(클라이언트/서버)에서 공통으로 사용할 수 있어야 합니다.

모든 Oracle Solaris 제품 설치 시에는 /etc/auto\_master 마스터 맵이 기본적으로 설치됩니다.

```
# Master map for autofs
#
+auto_master
/net      -hosts      -nosuid,nobrowse
/home     auto_home   -nobrowse
```

auto\_home의 맵도 /etc 아래에 설치됩니다.

```
# Home directory map for autofs
#
+auto_home
```

외부 auto\_home 맵에 대한 참조를 제외하면 이 맵은 비어 있습니다. /home 아래의 디렉토리를 모든 컴퓨터에서 공통으로 사용하려면 이 /etc/auto\_home 맵을 수정하지 마십시오. 모든 홈 디렉토리는 이름 서비스 파일에 표시되어야 합니다.

---

주 - 사용자가 자신의 홈 디렉토리에서 setuid 실행 파일을 실행하도록 허용해서는 안 됩니다. 이 제한을 적용하지 않으면 모든 사용자가 모든 컴퓨터에 대한 슈퍼 유저 권한을 가질 수 있습니다.

---

## ▼ 여러 홈 디렉토리 파일 시스템을 사용하여 /home을 설정하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 /export/home 아래에 홈 디렉토리 파티션을 설치합니다.

시스템에 파티션이 여러 개 있는 경우에는 /export/home1, /export/home2 등과 같은 별도의 디렉토리 아래에 파티션을 설치합니다.

### 3 auto\_home 맵을 업데이트합니다.

새 사용자 계정을 만들 때마다 `auto_home` 맵에 사용자 홈 디렉토리의 위치를 입력합니다. 맵 항목은 다음과 같이 간단한 형태일 수 있습니다.

```
rusty      dragon:/export/home1/&
gwenda     dragon:/export/home1/&
charles    sundog:/export/home2/&
rich       dragon:/export/home3/&
```

여기서는 `&`(앰퍼센드)를 맵 키 대신 사용했습니다. 앰퍼센드는 다음 예제에서 두번째 `rusty` 항목의 약어입니다.

```
rusty      dragon:/export/home1/rusty
```

`auto_home` 맵이 배치되면 사용자가 `/home/user` 경로를 사용하여 자신의 홈 디렉토리를 비롯한 모든 홈 디렉토리를 참조할 수 있습니다. 여기서 `user`는 사용자의 로그인 이름 및 맵의 키입니다. 이와 같은 모든 홈 디렉토리의 공통 보기는 다른 사용자의 컴퓨터에 로그인할 때 유용하게 사용할 수 있습니다. `autofs`는 홈 디렉토리를 자동으로 마운트합니다. 마찬가지로 다른 컴퓨터에서 원격 윈도우화 시스템 클라이언트를 실행하는 경우 클라이언트 프로그램에도 동일한 `/home` 디렉토리 보기가 있습니다.

이 공통 보기는 서버로도 확장됩니다. 이전 예제에서 `rusty`가 `dragon` 서버에 로그인하는 경우 해당 서버의 `autofs`가 `/export/home1/rusty`를 `/home/rusty`에 루프백 마운트하여 로컬 디스크에 대한 직접 액세스 권한을 제공합니다.

따라서 사용자는 실제 홈 디렉토리 위치를 몰라도 됩니다. `rusty`가 디스크 공간이 더 필요하여 홈 디렉토리를 다른 서버로 옮겨야 하는 경우에는 간단한 변경만 수행하면 됩니다. 즉, 새 위치를 반영하도록 `auto_home` 맵에서 `rusty`의 항목만 변경하면 됩니다. 다른 사용자는 `/home/rusty` 경로를 계속 사용할 수 있습니다.

## ▼ /ws 아래에서 프로젝트 관련 파일을 통합하는 방법

사용자가 대규모 소프트웨어 개발 프로젝트의 관리자라고 가정해 보겠습니다. 이 관리자는 모든 프로젝트 관련 파일을 `/ws`라는 디렉토리 아래에서 제공하려고 합니다. 사이트의 모든 워크스테이션에서 이 디렉토리를 공통으로 사용할 것입니다.

### 1 /ws 디렉토리에 대한 항목을 사이트 `auto_master` 맵에 추가합니다.

```
/ws      auto_ws      -nosuid
```

`auto_ws` 맵에서 `/ws` 디렉토리의 내용이 결정됩니다.

### 2 만약의 경우를 위해 `-nosuid` 옵션을 추가합니다.

이 옵션은 작업 영역에 있을 수 있는 `setuid` 프로그램을 사용자가 실행할 수 없도록 합니다.

### 3 auto\_ws 맵에 항목을 추가합니다.

auto\_ws 맵은 각 항목이 하위 프로젝트를 설명하는 방식으로 구성됩니다. 첫번째 시도에서는 다음과 같은 맵이 생성됩니다.

```
compiler    alpha:/export/ws/&
windows    alpha:/export/ws/&
files      bravo:/export/ws/&
drivers    alpha:/export/ws/&
man        bravo:/export/ws/&
tools      delta:/export/ws/&
```

각 항목 끝의 앰퍼센드(&)는 항목 키의 약어입니다. 예를 들어 첫번째 항목은 다음에 해당합니다.

```
compiler      alpha:/export/ws/compiler
```

이 첫번째 시도에서는 단순히 보이는 맵이 제공되지만 이 맵은 적절하지 않습니다. 프로젝트 구성기는 man 항목의 설명서를 각 하위 프로젝트 아래 하위 디렉토리로 제공해야 함을 결정합니다. 또한 각 하위 프로젝트에서는 하위 디렉토리가 여러 소프트웨어 버전을 설명해야 합니다. 이러한 각 하위 디렉토리는 서버의 전체 디스크 파티션에 지정해야 합니다.

맵의 항목을 다음과 같이 수정합니다.

```
compiler \
  /vers1.0    alpha:/export/ws/&/vers1.0 \
  /vers2.0    bravo:/export/ws/&/vers2.0 \
  /man        bravo:/export/ws/&/man
windows \
  /vers1.0    alpha:/export/ws/&/vers1.0 \
  /man        bravo:/export/ws/&/man
files \
  /vers1.0    alpha:/export/ws/&/vers1.0 \
  /vers2.0    bravo:/export/ws/&/vers2.0 \
  /vers3.0    bravo:/export/ws/&/vers3.0 \
  /man        bravo:/export/ws/&/man
drivers \
  /vers1.0    alpha:/export/ws/&/vers1.0 \
  /man        bravo:/export/ws/&/man
tools \
  /           delta:/export/ws/&
```

이제 맵은 훨씬 더 크게 표시되지만 아직 포함된 항목은 5개뿐입니다. 각 항목은 여러 마운트를 포함하므로 이전보다 크기가 커졌습니다. 예를 들어 /ws/compiler를 참조하려면 vers1.0, vers2.0 및 man 디렉토리에 대한 3개의 마운트가 필요합니다. 각 행의 끝에 오는 백슬래시는 입력 내용이 다음 행으로 이어진다는 것을 autofs에 알립니다. 효율성을 위해 한 행에 모두 입력했지만 가독성을 위해 줄 바꿈과 들여쓰기가 부분적으로 사용되었습니다. tools 디렉토리에는 모든 하위 프로젝트에 대한 소프트웨어 개발 도구가 포함되어 있으므로 이 디렉토리는 같은 하위 디렉토리 구조를 따르지 않습니다. tools 디렉토리는 계속 단일 마운트로 유지됩니다.

디렉토리가 이와 같이 배열되므로 관리자는 작업을 훨씬 유동적으로 수행할 수 있습니다. 일반적으로 소프트웨어 프로젝트는 디스크 공간을 많이 사용합니다.

프로젝트 수명 동안 여러 디스크 파티션을 재배치 및 확장해야 할 수 있습니다. 이러한 변경 내용이 `auto_ws` 맵에 반영되어도 사용자는 알림을 받을 필요가 없습니다. `/ws` 아래의 디렉토리 계층은 변경되지 않기 때문입니다.

`alpha` 및 `bravo` 서버는 동일한 `autofs` 맵을 확인하므로, 이러한 컴퓨터에 로그인하는 사용자는 `/ws` 네임스페이스를 정상적으로 찾을 수 있습니다. 이러한 사용자에게는 NFS 마운트가 아닌 루프백 마운트를 통해 로컬 파일에 대한 직접 액세스 권한이 제공됩니다.

## ▼ 공유 네임스페이스에 액세스하도록 서로 다른 구조를 설정하는 방법

스프레드시트 응용 프로그램, 워드 프로세싱 패키지 등의 응용 프로그램과 로컬 실행 파일에 대해 공유 네임스페이스를 어셈블해야 합니다. 이 네임스페이스의 클라이언트는 서로 다른 실행 파일 형식을 필요로 하는 여러 워크스테이션 구조를 사용합니다. 또한 일부 워크스테이션에서는 다른 운영 체제 릴리스를 실행합니다.

### 1 `auto_local` 맵을 만듭니다.

**Oracle Solaris Administration: Naming and Directory Services**를 참조하십시오.

### 2 공유 네임스페이스에 대해 사이트 특정 이름을 하나 선택합니다.

이 이름을 통해 해당 네임스페이스에 속하는 파일과 디렉토리를 쉽게 식별할 수 있습니다. 예를 들어 이름으로 `/usr/local`을 사용하는 경우 `/usr/local/bin` 경로는 이 네임스페이스에 속합니다.

### 3 사용자 커뮤니티를 쉽게 파악할 수 있도록 `autofs` 간접 맵을 만듭니다.

`/usr/local`에서 이 맵을 마운트합니다. `NISauto_master` 맵에서 다음 항목을 설정합니다.

```
/usr/local      auto_local      -ro
```

`-ro` 마운트 옵션은 클라이언트가 파일 또는 디렉토리에 쓸 수 없음을 의미합니다.

### 4 서버에서 해당하는 디렉토리를 내보냅니다.

### 5 `auto_local` 맵에 `bin` 항목을 포함합니다.

디렉토리 구조는 다음과 같습니다.

```
bin      aa:/export/local/bin
```

### 6 (옵션) 구조가 서로 다른 클라이언트의 작업을 처리하려면 `autofs` CPU 변수를 추가하여 항목을 변경합니다.

```
bin      aa:/export/local/bin/$CPU
```

- SPARC 클라이언트의 경우에는 `/export/local/bin/sparc`에 실행 파일을 저장합니다.
- x86 클라이언트의 경우에는 `/export/local/bin/i386`에 실행 파일을 저장합니다.

## ▼ 호환되지 않는 클라이언트 운영 체제 버전을 지원하는 방법

- 1 클라이언트의 운영 체제 유형을 결정하는 변수와 구조 유형을 결합합니다.

autofs OSREL 변수를 CPU 변수와 결합하여 CPU 유형과 OS 릴리스를 모두 결정하는 이름을 만들 수 있습니다.

- 2 다음 맵 항목을 만듭니다.

```
bin      aa:/export/local/bin/$CPU$OSREL
```

운영 체제 버전 5.6을 실행하는 클라이언트의 경우 다음 파일 시스템을 내보냅니다.

- SPARC 클라이언트의 경우 /export/local/bin/sparc5.6을 내보냅니다.
- x86 클라이언트의 경우 /export/local/bin/i3865.6에 실행 파일을 저장합니다.

## ▼ 여러 서버에서 공유 파일을 복제하는 방법

복제된 읽기 전용 파일 시스템을 공유하는 가장 효율적인 방법은 페일오버를 사용하는 것입니다. 페일오버에 대한 설명은 [184 페이지](#) “클라이언트측 페일오버”를 참조하십시오.

- 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 autofs 맵에서 항목을 수정합니다.

다음과 같이 모든 복제본 서버 목록을 쉼표로 구분된 목록으로 만듭니다.

```
bin      aa,bb,cc,dd:/export/local/bin/$CPU
```

autofs에서 가장 가까운 서버를 선택합니다. 서버에 네트워크 인터페이스가 여러 개 있는 경우 각 인터페이스를 목록에 포함합니다. autofs는 클라이언트에 가장 가까운 인터페이스를 선택하여 불필요한 NFS 트래픽 경로 지정을 방지합니다.

## ▼ autofs 보안 제한을 적용하는 방법

- 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 이름 서비스 auto\_master 파일에서 다음 항목을 만듭니다.

```
/home      auto_home      -nosuid
```

nosuid 옵션은 사용자가 setuid 또는 setgid 비트를 설정하여 파일을 만들지 못하도록 합니다.

이 항목은 일반 로컬 /etc/auto\_master 파일에서 /home에 대한 항목을 대체합니다. 이전 예제를 참조하십시오. 이처럼 항목이 대체되는 이유는 파일의 /home 항목 이전에 외부 이름 서비스 맵에 대한 +auto\_master를 참조하기 때문입니다. auto\_home 맵의 항목에 마운트 옵션이 포함되는 경우에는 nosuid 옵션을 덮어씁니다. 따라서 auto\_home 맵에서 옵션을 사용하지 않거나, nosuid 옵션을 각 항목에 포함해야 합니다.

---

주 - 홈 디렉토리 디스크 파티션을 서버의 /home 또는 그 아래에 마운트하지 마십시오.

---

## ▼ autofs와 함께 공용 파일 핸들을 사용하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 autofs 맵에서 다음과 같은 항목을 만듭니다.

```
/usr/local -ro,public bee:/export/share/local
```

public 옵션은 공용 핸들이 강제로 사용되도록 합니다. NFS 서버가 공용 파일 핸들을 지원하지 않는 경우에는 마운트가 실패합니다.

## ▼ autofs와 함께 NFS URL을 사용하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 다음과 같은 autofs 항목을 만듭니다.

```
/usr/local -ro nfs://bee/export/share/local
```

서비스는 NFS 서버에서 공용 파일 핸들 사용을 시도합니다. 그러나 서버에서 공용 파일 핸들을 지원하지 않는 경우에는 MOUNT 프로토콜이 사용됩니다.

## autofs 찾아보기 기능 사용 안함으로 설정

설치된 /etc/auto\_master의 기본 버전에는 -nobrowse 옵션이 /home 및 /net에 대한 항목에 추가되어 있습니다. 또한 업그레이드 절차를 수행하면 -nobrowse 옵션이 /etc/auto\_master의 /home 및 /net 항목에 추가됩니다(이러한 항목이 수정되지 않은



경우), 그러나 이러한 변경을 수동으로 수행하거나 설치 후에 사이트 특정 autofs 마운트 지점의 찾아보기 기능을 해제해야 할 수 있습니다.

여러 가지 방법으로 찾아보기 기능을 해제할 수 있습니다. automountd 데몬에 대해 명령줄 옵션을 사용하여 기능을 사용 안함으로 설정합니다. 이렇게 하면 클라이언트에 대한 autofs 찾아보기 기능이 완전히 해제됩니다. 또한 autofs 맵을 사용하여 모든 클라이언트에서 각 맵 항목에 대해 찾아보기 기능을 사용 안함으로 설정합니다. 네트워크 차원 네임스페이스를 사용하고 있지 않은 경우에는 로컬 autofs 맵을 사용하여 각 클라이언트에서 각 맵 항목에 대해 기능을 사용 안함으로 설정할 수도 있습니다.

## ▼ 단일 NFS 클라이언트에서 autofs 찾아보기 기능을 완전히 사용 안함으로 설정하는 방법

- 1 NFS 클라이언트에서 관리자 권한을 얻습니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 autofs SMF 구성 매개변수를 변경합니다.

```
# sharectl set -p nobrowse=TRUE autofs
```

- 3 autofs 서비스를 다시 시작합니다.

```
# svcadm restart system/filesystem/autofs
```

## ▼ 모든 클라이언트에 대해 autofs 찾아보기 기능을 사용 안함으로 설정하는 방법

모든 클라이언트에 대해 찾아보기 기능을 사용 안함으로 설정하려면 NIS와 같은 이름 서비스를 적용해야 합니다. 그렇지 않으면 각 클라이언트에서 자동 마운트 맵을 수동으로 편집해야 합니다. 이 예제에서는 /home 디렉토리의 찾아보기 기능을 사용 안함으로 설정합니다. 사용 안함으로 설정해야 하는 각 간접 autofs 노드에 대해 다음 절차를 따라야 합니다.

- 1 이름 서비스 auto\_master 파일에서 -nobrowse 옵션을 /home 항목에 추가합니다.

```
/home      auto_home      -nobrowse
```

- 2 모든 클라이언트에서 automount 명령을 실행합니다.

클라이언트 시스템에서 automount 명령을 실행한 후에 또는 재부트 후에 새 동작이 적용됩니다.

```
# /usr/sbin/automount
```

## ▼ 선택한 파일 시스템에 대해 autofs 찾아보기 기능을 사용 안함으로 설정하는 방법

이 예제에서는 /net 디렉토리의 찾아보기 기능을 사용 안함으로 설정합니다. /home 또는 다른 autofs 마운트 지점에도 같은 절차를 사용할 수 있습니다.

### 1 자동 마운트 이름 지정 서비스의 검색 순서를 확인합니다.

name-service/switch 서비스의 config/automount 등록 정보에 자동 마운트 정보 검색 순서가 표시됩니다.

```
# svcprop -p config svc:/system/name-service/switch
config/value_authorization astring solaris.smf.value.name-service.switch
config/printer astring user\ files
config/default astring files\ nis
config/automount astring files\ nis
```

마지막 항목에는 로컬 자동 마운트 파일이 먼저 검색된 다음 NIS 서비스를 확인함이 표시됩니다. config/default 항목은 구체적으로 나열되어 있지 않은 모든 이름 지정 정보의 검색 순서를 지정합니다.

### 2 /etc/auto\_master에서 +auto\_master 항목의 위치를 확인합니다.

로컬 파일에 추가하는 내용이 네임스페이스의 항목보다 우선적으로 검색되도록 하려면 +auto\_master 항목을 /net 뒤로 이동해야 합니다.

```
# Master map for automounter
#
/net      -hosts      -nosuid
/home     auto_home
/xfn      -xfn
+auto_master
```

표준 구성에서는 +auto\_master 항목이 파일 맨 위에 있습니다. 이러한 배치로 인해 로컬 변경 내용을 사용할 수 없습니다.

### 3 nobrowse 옵션을 /etc/auto\_master 파일의 /net 항목에 추가합니다.

```
/net      -hosts      -nosuid,nobrowse
```

### 4 모든 클라이언트에서 automount 명령을 실행합니다.

클라이언트 시스템에서 automount 명령을 실행한 후에 또는 재부트 후에 새 동작이 적용됩니다.

```
# /usr/sbin/automount
```

## NFS 참조 관리

NFSv4 서버에서는 NFS 참조를 사용하여 다른 NFSv4 서버에 있는 파일 시스템을 가리킵니다. 이러한 방법으로 여러 NFSv4 서버를 통합 네임스페이스로 연결할 수 있습니다.

### ▼ NFS 참조를 만들고 액세스하는 방법

#### 1 NFS 서버에서 참조를 만듭니다.

NFS 공유 파일 시스템에서 하나 이상의 기존 NFS 공유 파일 시스템을 가리키는 참조를 추가합니다.

```
server1% nfsref add /share/docs server2:/usr/local/docs server3:/tank/docs
Created reparse point /share/docs
```

#### 2 참조가 만들어졌는지 확인합니다.

```
server1% nfsref lookup /share/docs
/share/docs points to:
server2:/usr/local/docs
server3:/tank/docs
```

#### 3 클라이언트에서 참조를 마운트합니다.

```
client1% pfexec mount server1:/share/docs /mnt
```

#### 4 마운트가 작동하는지 확인합니다.

```
client1% cd /mnt/docs
client1% df -k .
/mnt/docs      (server2:/usr/local/docs):10372284465 blocks 10372284465 files
```

### 예 5-3 기존 참조 수정

server4:/tank/docs 등의 다른 파일 시스템을 기존 참조에 추가하려면 위의 2단계에서 새 파일 시스템과 함께 명령을 입력합니다.

```
server1% nfsref add /share/docs server2:/usr/local/docs server3:/tank/docs server4:/tank/docs
```

add 하위 명령은 현재 참조의 정보를 명령의 새 정보로 바꿉니다. add 하위 명령을 통해 기존 참조와 연결된 파일 시스템을 수정합니다.

## ▼ NFS 참조를 제거하는 방법

NFS 참조를 제거하려면 다음 절차를 따릅니다.

- 참조를 제거합니다.

```
server1% nfsref remove /share/docs
Removed svc_type 'nfs-basic' from /share/docs
```

## NFS 문제 해결 전략

NFS 문제를 추적할 때는 가능한 주요 오류 지점(클라이언트/네트워크)을 기억하십시오. 이 절에서 설명하는 전략은 개별 구성 요소를 격리시켜 작동하지 않는 구성 요소를 찾는 것입니다. 모든 상황에서 `mountd` 및 `nfsd` 데몬이 서버에서 실행 중이어야 원격 마운트가 성공합니다.

`-intr` 옵션은 모든 마운트에 대해 기본적으로 설정됩니다. 프로그램이 정지되고 `server not responding` 메시지가 표시되는 경우에는 키보드 중단 키 `Ctrl-C`를 눌러 프로그램을 종료할 수 있습니다.

네트워크 또는 서버에 문제가 있는 경우 하드 마운트 원격 파일에 액세스하는 프로그램에서 소프트 마운트된 원격 파일에 액세스하는 프로그램과는 다른 오류가 발생합니다. 하드 마운트된 원격 파일 시스템의 경우 서버가 다시 응답할 때까지 클라이언트 커널에서 요청을 다시 시도합니다. 소프트 마운트된 원격 파일 시스템의 경우에는 일정 시간 액세스를 시도한 후에 클라이언트 시스템 호출에서 오류가 반환됩니다. 이러한 오류로 인해 예기치 않은 응용 프로그램 오류 및 데이터 손상이 발생할 수 있으므로 소프트 마운트는 가능하면 사용하지 마십시오.

파일 시스템을 하드 마운트하는 경우 서버가 응답하지 않으면 파일 시스템 액세스를 시도하는 프로그램이 정지됩니다. 이 경우 NFS 시스템의 콘솔에 다음 메시지가 표시됩니다.

```
NFS server hostname not responding still trying
```

서버가 응답하면 콘솔에는 다음 메시지가 표시됩니다.

```
NFS server hostname ok
```

프로그램이 소프트 마운트된 파일 시스템에 액세스하는 경우 서버가 응답하지 않으면 다음 메시지가 생성됩니다.

```
NFS operation failed for server hostname: error # (error-message)
```

주- 오류가 발생할 가능성이 있으므로 실행 파일이 실행되는 파일 시스템 또는 읽기/쓰기 데이터가 포함된 파일 시스템은 소프트 마운트하지 마십시오. 응용 프로그램에서 오류를 무시하는 경우 쓰기 가능 데이터가 손상될 수 있습니다. 마운트된 실행 파일이 정상적으로 로드되지 않고 오류가 발생할 수 있습니다.

## NFS 문제 해결 절차

NFS 서비스에 오류가 발생했는지 확인하려면 몇 가지 절차를 수행하여 오류를 파악해야 합니다. 다음 항목을 확인하십시오.

- 클라이언트가 서버에 연결할 수 있는지 여부
- 클라이언트가 서버의 NFS 서비스에 연결할 수 있는지 여부
- 서버에서 NFS 서비스가 실행되고 있는지 여부

이러한 항목을 확인하는 과정에서 네트워크의 다른 부분이 작동하지 않음을 확인할 수도 있습니다. 예를 들어 이름 서비스 또는 물리적 네트워크 하드웨어가 작동하지 않을 수 있습니다. **Oracle Solaris Administration: Naming and Directory Services**에 여러 이름 서비스에 대한 디버깅 절차가 나와 있습니다. 또한 프로세스 중에 문제가 클라이언트 쪽에서 발생한 것이 아님을 확인할 수도 있습니다. 예를 들어 작업 영역의 모든 서브넷에서 문제 호출이 하나 이상 발생할 수 있습니다. 이 경우에는 문제가 서버에서 발생했는지 아니면 서버 근처의 네트워크 하드웨어에서 발생했는지를 파악해야 합니다. 따라서 클라이언트가 아닌 서버에서 디버깅 프로세스를 시작해야 합니다.

### ▼ NFS 클라이언트에서 연결을 확인하는 방법

- 1 클라이언트에서 NFS 서버에 연결할 수 있는지 확인합니다. 클라이언트에서 다음 명령을 입력합니다.

```
% /usr/sbin/ping bee
bee is alive
```

명령에서 서버가 활성 상태임이 보고되면 NFS 서버를 원격으로 확인합니다. [118 페이지 “원격으로 NFS 서버를 확인하는 방법”](#)을 참조하십시오.

- 2 클라이언트에서 서버에 연결할 수 없는 경우에는 로컬 이름 서비스가 실행 중인지 확인합니다.
- 3 이름 서비스가 실행 중인 경우 다음을 입력하여 클라이언트가 올바른 호스트 정보를 받았는지 확인합니다.

```
% /usr/bin/getent hosts bee
129.144.83.117 bee.eng.acme.com
```

- 4 호스트 정보가 정확한데 서버가 클라이언트에 연결할 수 없는 경우에는 다른 클라이언트에서 ping 명령을 실행합니다.  
두번째 클라이언트의 명령 실행이 실패하면 119 페이지 “서버에서 NFS 서비스를 확인하는 방법”을 참조하십시오.
- 5 서버가 두번째 클라이언트에서 연결할 수 있는 경우에는 ping을 사용하여 첫번째 클라이언트가 로컬 네트워크의 다른 시스템에 연결할 수 있는지 확인합니다.  
이 명령이 실패하면 클라이언트에서 네트워킹 소프트웨어 구성을 확인합니다(예: /etc/netmasks 및 svc:/system/name-service/switch 서비스와 연결된 등록 정보).
- 6 (옵션) rpcinfo 명령의 출력을 확인합니다.  
rpcinfo 명령을 실행해도 program 100003 version 4 ready and waiting이 표시되지 않으면 NFS 버전 4가 서버에서 사용으로 설정되어 있지 않은 것입니다. NFS 버전 4를 사용으로 설정하는 방법에 대한 자세한 내용은 표 5-3을 참조하십시오.
- 7 소프트웨어가 올바른 경우에는 네트워킹 하드웨어를 확인합니다.  
클라이언트를 두번째 네트워크 놓기 지점으로 이동해 봅니다.

## ▼ 원격으로 NFS 서버를 확인하는 방법

NFS 버전 4 서버를 사용하는 경우에는 UDP 및 MOUNT 프로토콜을 둘 다 지원하지 않아도 됩니다.

- 1 다음 명령을 입력하여 NFS 서버에서 NFS 서비스가 시작되었는지 확인합니다.

```
% rpcinfo -s bee | egrep 'nfs|mountd'
100003 3,2 tcp,udp,tcp6,udp6 nfs superuser
100005 3,2,1 ticots,ticotsord,tcp,tcp6,ticlts,udp,udp6 mountd superuser
```

데몬이 시작되지 않은 경우 120 페이지 “NFS 서비스를 다시 시작하는 방법”을 참조하십시오.

- 2 서버의 nfsd 프로세스가 응답하는지 확인합니다.

클라이언트에서 다음 명령을 입력하여 서버로부터의 UDP NFS 연결을 테스트합니다.

```
% /usr/bin/rpcinfo -u bee nfs
program 100003 version 2 ready and waiting
program 100003 version 3 ready and waiting
```

---

주 - NFS 버전 4에서는 UDP를 지원하지 않습니다.

---

서버가 실행 중인 경우 프로그램 및 버전 번호 목록이 출력됩니다. -t 옵션을 사용하여 TCP 연결을 테스트합니다. 이 명령이 실패하면 119 페이지 “서버에서 NFS 서비스를 확인하는 방법”으로 진행합니다.

### 3 다음 명령을 입력하여 서버의 mountd가 응답하는지 확인합니다.

```
% /usr/bin/rpcinfo -u bee mountd
program 100005 version 1 ready and waiting
program 100005 version 2 ready and waiting
program 100005 version 3 ready and waiting
```

서버가 실행 중인 경우에는 UDP 프로토콜과 연결된 버전 번호 및 프로그램 목록이 출력됩니다. -t 옵션을 사용하여 TCP 연결을 테스트합니다. 이 시도가 실패하면 119 페이지 “서버에서 NFS 서비스를 확인하는 방법”으로 진행합니다.

### 4 로컬 autofs 서비스가 사용되고 있는지 확인합니다.

```
% cd /net/wasp
```

정상적으로 작동하는 /net 또는 /home 마운트 지점을 선택합니다. 이 명령이 실패하면 클라이언트에서 root로 다음을 입력하여 autofs 서비스를 다시 시작합니다.

```
# svcadm restart system/filesystem/autofs
```

### 5 파일 시스템이 서버에서 정상적으로 공유되는지 확인합니다.

```
% /usr/sbin/showmount -e bee
/usr/src                               eng
/export/share/man                     (everyone)
```

서버의 항목과 로컬 마운트 항목에 오류가 있는지 확인합니다. 또한 네임스페이스도 확인합니다. 이 경우 첫번째 클라이언트가 eng 넷 그룹에 있지 않으면 해당 클라이언트는 /usr/src 파일 시스템을 마운트할 수 없습니다.

모든 로컬 파일에서 마운트 정보를 포함하는 모든 항목을 확인합니다. 목록에는 /etc/vfstab 및 모든 /etc/auto\_\* 파일이 포함됩니다.

## ▼ 서버에서 NFS 서비스를 확인하는 방법

#### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

#### 2 서버가 클라이언트에 연결할 수 있는지 확인합니다.

```
# ping lilac
lilac is alive
```

#### 3 서버에서 클라이언트에 연결할 수 없는 경우에는 로컬 이름 서비스가 실행 중인지 확인합니다.

#### 4 이름 서비스가 실행 중이면 서버에서 네트워킹 소프트웨어 구성을 확인합니다(예: /etc/netmasks 및 svc:/system/name-service/switch 서비스와 연결된 등록 정보).

**5 다음 명령을 입력하여 rpcbind 데몬이 실행 중인지 확인합니다.**

```
# /usr/bin/rpcinfo -u localhost rpcbind
program 100000 version 1 ready and waiting
program 100000 version 2 ready and waiting
program 100000 version 3 ready and waiting
```

서버가 실행 중인 경우 UDP 프로토콜에 연결된 버전 번호 및 프로그램 목록이 출력됩니다.

**6 다음 명령을 입력하여 nfsd 데몬이 실행 중인지 확인합니다.**

```
# rpcinfo -u localhost nfs
program 100003 version 2 ready and waiting
program 100003 version 3 ready and waiting
# ps -ef | grep nfsd
root    232      1  0 Apr 07    ?        0:01 /usr/lib/nfs/nfsd -a 16
root    3127    2462  1 09:32:57 pts/3    0:00 grep nfsd
```

주 - NFS 버전 4에서는 UDP를 지원하지 않습니다.

서버가 실행 중인 경우에는 UDP 프로토콜과 연결된 버전 번호 및 프로그램 목록이 출력됩니다. 또한 rpcinfo에서 -t 옵션을 사용하여 TCP 연결을 확인합니다. 이러한 명령이 실패하면 NFS 서비스를 다시 시작합니다. [120 페이지 “NFS 서비스를 다시 시작하는 방법”](#)을 참조하십시오.

**7 다음 명령을 입력하여 mountd 데몬이 실행 중인지 확인합니다.**

```
# /usr/bin/rpcinfo -u localhost mountd
program 100005 version 1 ready and waiting
program 100005 version 2 ready and waiting
program 100005 version 3 ready and waiting
# ps -ef | grep mountd
root    145      1  0 Apr 07    ?        21:57 /usr/lib/autofs/automountd
root    234      1  0 Apr 07    ?        0:04 /usr/lib/nfs/mountd
root    3084    2462  1 09:30:20 pts/3    0:00 grep mountd
```

서버가 실행 중인 경우에는 UDP 프로토콜과 연결된 버전 번호 및 프로그램 목록이 출력됩니다. 또한 rpcinfo에서 -t 옵션을 사용하여 TCP 연결을 확인합니다. 이러한 명령이 실패하면 NFS 서비스를 다시 시작합니다. [120 페이지 “NFS 서비스를 다시 시작하는 방법”](#)을 참조하십시오.

**▼ NFS 서비스를 다시 시작하는 방법****1 관리자가 됩니다.**

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.



## 2 서버에서 NFS 서비스를 다시 시작합니다.

다음 명령을 입력합니다.

```
# svcadm restart network/nfs/server
```

## NFS 파일 서비스를 제공하는 호스트 식별

-m 옵션을 포함해 `nfsstat` 명령을 실행하여 현재 NFS 정보를 수집합니다. 현재 서버의 이름은 “currserver=” 다음에 출력됩니다.

```
% nfsstat -m
/usr/local from bee:wasp:/export/share/local
Flags: vers=3,proto=tcp,sec=sys,hard,intr,llock,link,synlink,
      acl,rsize=32768,wsiz=32678,retrans=5
Failover: noresponse=0, failover=0, remap=0, currserver=bee
```

## ▼ mount 명령에 사용되는 옵션을 확인하는 방법

잘못된 옵션에 대해 경고가 표시되지는 않습니다. 다음 절차를 수행하면 명령줄에서 또는 `/etc/vfstab`를 통해 제공된 옵션이 유효한지를 확인할 수 있습니다.

이 예제에서는 다음 명령을 실행했다고 가정합니다.

```
# mount -F nfs -o ro,vers=2 bee:/export/share/local /mnt
```

### 1 다음 명령을 실행하여 옵션을 확인합니다.

```
% nfsstat -m
/mnt from bee:/export/share/local
Flags: vers=2,proto=tcp,sec=sys,hard,intr,dynamic,acl,rsize=8192,wsiz=8192,
      retrans=5
```

bee의 파일 시스템은 프로토콜 버전이 2로 설정된 상태로 마운트되었습니다. 그러나 `nfsstat` 명령은 모든 옵션에 대한 정보를 표시하지는 않습니다. 그렇기는 하지만 `nfsstat` 명령을 사용하면 옵션을 가장 정확하게 확인할 수 있습니다.

### 2 /etc/mnttab의 항목을 확인합니다.

`mount` 명령을 사용하는 경우 잘못된 옵션을 마운트 테이블에 추가할 수 없습니다. 따라서 파일에 나열되어 있는 옵션이 명령줄에 나열되어 있는 옵션과 일치하는지 확인하십시오. 이러한 방식으로 `nfsstat` 명령에서 보고되지 않는 옵션을 확인할 수 있습니다.

```
# grep bee /etc/mnttab
bee:/export/share/local /mnt nfs      ro,vers=2,dev=2b0005e 859934818
```

## autofs 문제 해결

autofs에서 문제가 발생하는 경우가 있습니다. 이 절의 내용을 참고하여 문제 해결 프로세스를 개선할 수 있습니다. 이 절은 두 개의 하위 절로 구분됩니다.

이 절에서는 autofs에서 생성되는 오류 메시지 목록을 제공합니다. 이 목록은 두 부분으로 나뉩니다.

- automount의 verbose(-v) 옵션을 통해 생성되는 오류 메시지
- 언제든지 표시될 수 있는 오류 메시지

각 오류 메시지 뒤에는 설명과 해당 메시지의 가능한 원인이 표시됩니다.

문제를 해결할 때는 verbose(-v) 옵션을 사용하여 autofs 프로그램을 시작하십시오. 그렇지 않으면 원인을 알 수 없는 문제가 발생할 수 있습니다.

다음 단락에는 autofs에 오류가 발생하는 경우 표시될 수 있는 오류 메시지와 가능한 문제 설명이 표시되어 있습니다.

### automount-v를 통해 생성되는 오류 메시지

bad key *key* in direct map *mapname*

**설명:** 직접 맵을 검색할 때 autofs에서 접두어 /가 없는 항목 키를 발견했습니다.

**해결책:** 직접 맵의 키는 전체 경로 이름이어야 합니다.

bad key *key* in indirect map *mapname*

**설명:** 간접 맵을 검색하는 중 autofs에서 /가 포함된 항목 키를 발견했습니다.

**해결책:** 간접 맵 키는 경로 이름이 아닌 간단한 이름이어야 합니다.

can't mount *server:pathname: reason*

**설명:** 서버의 마운트 데몬이 *server:pathname*에 대한 파일 핸들 제공을 거부했습니다.

**해결책:** 서버의 내보내기 테이블을 확인합니다.

couldn't create mount point *mountpoint: reason*

**설명:** autofs에서 마운트에 필요한 마운트 지점을 만들 수 없습니다. 이 문제는 서버의 모든 내보낸 파일 시스템을 계층적으로 마운트하려고 할 때 발생하는 경우가 가장 많습니다.

**해결책:** 필요한 마운트 지점이 마운트할 수 없는 파일 시스템에만 존재할 수 있습니다(파일 시스템을 내보낼 수 없음). 내보낸 상위 파일 시스템이 읽기 전용으로 내보내기되어 마운트 지점을 만들 수 없습니다.

leading space in map entry *entry* text in *mapname*

**설명:** autofs가 자동 마운트 맵에서 선행 공백이 있는 항목을 발견했습니다. 이 문제는 보통 잘못 계속된 맵 항목을 나타냅니다. 예를 들면 다음과 같습니다.

```
fake
/blank      frobz:/usr/frotz
```

**해결책:** 이 예제에서는 첫번째 행이 백슬래시(\)로 끝나야 하므로 autofs가 두번째 행을 발견하면 경고가 생성됩니다.

*mapname*: Not found

**설명:** 필요한 맵을 찾을 수 없습니다. 이 메시지는 -v 옵션을 사용하는 경우에만 생성됩니다.

**해결책:** 맵 이름의 맞춤법과 경로 이름을 확인합니다.

remount *server:pathname* on *mountpoint*: server not responding

**설명:** autofs에서 이전에 마운트 해제한 파일 시스템을 다시 마운트하지 못했습니다.

**해결책:** Sun에 지원을 요청하십시오. 이 오류 메시지는 거의 표시되지 않으며 직접 해결할 수 있는 방법도 없습니다.

WARNING: *mountpoint* already mounted on

**설명:** autofs에서 기존 마운트 지점에 마운트하려고 합니다. 이 메시지는 autofs에서 내부 오류(비정상적인 상황)가 발생했음을 의미합니다.

**해결책:** Sun에 지원을 요청하십시오. 이 오류 메시지는 거의 표시되지 않으며 직접 해결할 수 있는 방법도 없습니다.

## 기타 오류 메시지

dir *mountpoint* must start with '/'

**해결책:** 자동 마운트 마운트 지점에는 전체 경로 이름을 지정해야 합니다. 마운트 지점의 맞춤법과 경로 이름을 확인합니다.

hierarchical mountpoint: *pathname1* and *pathname2*

**해결책:** autofs에서는 계층 관계가 포함된 마운트 지점을 허용하지 않습니다. autofs 마운트 지점은 자동 마운트된 다른 파일 시스템 내에 포함되어서는 안 됩니다.

host *server* not responding

**설명:** autofs가 *server*에 연결하려고 했으나 응답을 받지 못했습니다.

**해결책:** NFS 서버 상태를 확인합니다.

*hostname: exports: rpc-err*

**설명:** *hostname*에서 내보내기 목록을 가져오는 중에 오류가 발생했습니다. 이 메시지는 서버 또는 네트워크 문제를 나타냅니다.

**해결책:** NFS 서버 상태를 확인합니다.

*map mapname, key key: bad*

**설명:** 맵 항목의 형식이 잘못되었으며 autofs에서 항목을 해석할 수 없습니다.

**해결책:** 항목을 다시 확인합니다. 항목에 이스케이프해야 하는 문자가 있을 수 있습니다.

*mapname: nis-err*

**설명:** NIS 맵에서 항목을 조회할 때 오류가 발생했습니다. 이 메시지는 NIS 문제를 나타낼 수 있습니다.

**해결책:** NIS 서버 상태를 확인합니다.

*mount of server: pathname on mountpoint: reason*

**설명:** autofs에서 마운트를 수행하지 못했습니다. 이러한 현상은 서버 또는 네트워크 문제를 나타낼 수 있습니다. *reason* 문자열에 문제가 정의되어 있습니다.

**해결책:** Sun에 지원을 요청하십시오. 이 오류 메시지는 거의 표시되지 않으며 직접 해결할 수 있는 방법도 없습니다.

*mountpoint: Not a directory*

**설명:** *mountpoint*는 디렉터리가 아니므로 autofs가 마운트될 수 없습니다.

**해결책:** 마운트 지점의 맞춤법과 경로 이름을 확인합니다.

*nfscast: cannot send packet: reason*

**설명:** autofs에서 복제된 파일 시스템 위치 목록의 서버로 질의 패킷을 보낼 수 없습니다. *reason* 문자열에 문제가 정의되어 있습니다.

**해결책:** Sun에 지원을 요청하십시오. 이 오류 메시지는 거의 표시되지 않으며 직접 해결할 수 있는 방법도 없습니다.

*nfscast: cannot receive reply: reason*

**설명:** autofs에서 복제된 파일 시스템 위치 목록의 서버로부터 회신을 받을 수 없습니다. *reason* 문자열에 문제가 정의되어 있습니다.

**해결책:** Sun에 지원을 요청하십시오. 이 오류 메시지는 거의 표시되지 않으며 직접 해결할 수 있는 방법도 없습니다.

**nfscast: select: reason**

**설명:** 이러한 모든 오류 메시지는 복제된 파일 시스템에 대해 서버를 확인하는 중에 문제가 발생했음을 나타냅니다. 이 메시지는 네트워크 문제를 나타낼 수 있습니다. **reason** 문자열에 문제가 정의되어 있습니다.

**해결책:** Sun에 지원을 요청하십시오. 이 오류 메시지는 거의 표시되지 않으며 직접 해결할 수 있는 방법도 없습니다.

**pathconf: no info for server :pathname**

**설명:** autofs에서 경로 이름에 대해 **pathconf** 정보를 가져오지 못했습니다.

**해결책:** **fpathconf(2)** 매뉴얼 페이지를 참조하십시오.

**pathconf: server: server not responding**

**설명:** autofs가 **pathconf()**에 정보를 제공하는 **server**에서 마운트 데몬에 연결할 수 없습니다.

**해결책:** 이 서버에서는 POSIX 마운트 옵션을 사용하지 마십시오.

## 기타 autofs 오류

/etc/auto\* 파일에 실행 비트 세트가 있는 경우 자동 마운트는 맵 실행을 시도하며, 그러면 다음과 같은 메시지가 표시됩니다.

**/etc/auto\_home: +auto\_home: not found**

이 경우 **auto\_home** 파일의 권한이 잘못된 것입니다. 파일의 각 항목은 이 메시지와 같은 오류 메시지를 생성합니다. 다음 명령을 입력하여 파일에 대한 권한을 재설정해야 합니다.

```
# chmod 644 /etc/auto_home
```

## NFS 오류 메시지

이 절에서는 오류 메시지와 오류를 발생시키는 상황에 대한 설명, 그리고 하나 이상의 해결 방법을 소개합니다.

**index 옵션에 대해 잘못된 인수가 지정됨 - 파일이어야 함**

**해결책:** **index** 옵션에 파일 이름을 포함해야 합니다. 디렉토리 이름은 사용할 수 없습니다.

**Cannot establish NFS service over /dev/tcp: transport setup problem**

**설명:** 이 메시지는 보통 네임스페이스의 서비스 정보가 업데이트되지 않은 경우 표시됩니다. 또한 UDP에 대해서도 이 메시지가 보고될 수 있습니다.

**해결책:** 이 문제를 해결하려면 네임스페이스에서 서비스 데이터를 업데이트해야 합니다.

NIS 및 /etc/services의 경우 항목은 다음과 같습니다.

```
nfsd    2049/tcp    nfs      # NFS server daemon
nfsd    2049/udp    nfs      # NFS server daemon
```

Could not start *daemon*: *error*

**설명:** 데몬이 비정상적으로 종료되거나 시스템 호출 오류가 발생하면 이 메시지가 표시됩니다. *error* 문자열에 문제가 정의되어 있습니다.

**해결책:** Sun에 지원을 요청하십시오. 이 오류 메시지는 거의 표시되지 않으며 직접 해결할 수 있는 방법도 없습니다.

Could not use public filehandle in request to *server*

**설명:** 이 메시지는 *public* 옵션이 지정되어 있는데 NFS 서버가 공용 파일 핸들을 지원하지 않는 경우에 표시됩니다. 이 경우 마운트가 실패합니다.

**해결책:** 이 상황을 해결하려면 공용 파일 핸들을 사용하지 않고 마운트 요청을 시도하거나, 공용 파일 핸들을 지원하도록 NFS 서버를 재구성하십시오.

*daemon* running already with pid *pid*

**설명:** 데몬이 이미 실행 중입니다.

**해결책:** 새 복사본을 실행하려면 현재 버전을 종료하고 새 버전을 시작합니다.

error locking *lock file*

**설명:** 이 메시지는 데몬과 연결된 *lock file*을 정상적으로 잠글 수 없을 때 표시됩니다.

**해결책:** Sun에 지원을 요청하십시오. 이 오류 메시지는 거의 표시되지 않으며 직접 해결할 수 있는 방법도 없습니다.

error checking *lock file*: *error*

**설명:** 이 메시지는 데몬과 연결된 *lock file*을 정상적으로 열 수 없을 때 표시됩니다.

**해결책:** Sun에 지원을 요청하십시오. 이 오류 메시지는 거의 표시되지 않으며 직접 해결할 수 있는 방법도 없습니다.

NOTICE: NFS3: failing over from *host1* to *host2*

**설명:** 이 메시지는 페일오버가 수행될 때 콘솔에 표시됩니다. 이 메시지는 정보용으로만 표시됩니다.

**해결책:** 작업을 수행할 필요는 없습니다.

*filename*: File too large

**설명:** NFS 버전 2 클라이언트가 2GB보다 큰 파일에 액세스하려고 합니다.

**해결책:** NFS 버전 2를 사용하지 않습니다. 버전 3 또는 버전 4를 사용하여 파일 시스템을 마운트합니다. 또한 `nolargefiles` option in 148 페이지 “NFS 파일 시스템용 mount 옵션”의 설명을 참조하십시오.

mount: ... server not responding:RPC\_PMAP\_FAILURE - RPC\_TIMED\_OUT

**설명:** 마운트하려는 파일 시스템을 공유하는 서버가 다운된 상태 또는 연결할 수 없거나, 실행 레벨이 잘못되었거나, 해당 `rpcbind`가 사용 불능 상태 또는 정지되었습니다.

**해결책:** 서버가 재부트될 때까지 기다립니다. 서버가 정지된 경우 서버를 재부트합니다.

mount: ... server not responding: RPC\_PROG\_NOT\_REGISTERED

**설명:** 마운트 요청은 `rpcbind`로 등록되었는데 NFS 마운트 데몬 `mountd`는 등록되지 않았습니다.

**해결책:** 서버가 재부트될 때까지 기다립니다. 서버가 정지된 경우 서버를 재부트합니다.

mount: ... No such file or directory

**설명:** 원격 디렉토리 또는 로컬 디렉토리가 없습니다.

**해결책:** 디렉토리 이름의 맞춤법을 확인합니다. 두 디렉토리에서 `ls`를 실행합니다.

mount: .... Permission denied

**설명:** 마운트를 시도한 파일 시스템에 액세스하도록 허용되는 클라이언트 또는 네트워크 그룹 목록에 컴퓨터 이름이 없을 수 있습니다.

**해결책:** `showmount -e`를 사용하여 액세스 목록을 확인합니다.

NFS file temporarily unavailable on the server, retrying ...

**설명:** NFS 버전 4 서버는 파일 관리 권한을 클라이언트에 위임할 수 있습니다. 이 메시지는 서버가 사용자의 클라이언트에서 보낸 요청과 충돌하는 다른 클라이언트에 대한 위임을 회수 중임을 나타냅니다.

**해결책:** 회수가 수행되어야 서버가 사용자 클라이언트의 요청을 처리할 수 있습니다. 위임에 대한 자세한 내용은 179 페이지 “NFS 버전 4의 위임”를 참조하십시오.

NFS fsstat failed for server *hostname*: RPC: Authentication error

**설명:** 이 오류는 다양한 상황에서 발생할 수 있습니다. 디버깅하기 가장 까다로운 상황 중 하나는 사용자가 너무 많은 그룹에 속해 있어 이 문제가 발생하는 경우입니다. 현재는 한 사용자가 17개 이상의 그룹에 속할 수 없습니다(NFS 마운트를 통해 파일에 액세스하는 경우).

**해결책:** 17개 이상의 그룹에 속해 있어야 하는 사용자의 경우에는 다른 방법이 있습니다. 즉, 액세스 제어 목록을 사용하여 필요한 액세스 권한을 제공할 수 있습니다.

**nfs mount: NFS can't support "nolargefiles"**

**설명:** NFS 클라이언트가 - nolargefiles 옵션을 사용하여 NFS 서버에서 파일 시스템을 마운트하려고 시도했습니다.

**해결책:** NFS 파일 시스템 유형의 경우에는 이 옵션이 지원되지 않습니다.

**nfs mount: NFS V2 can't support "largefiles"**

**설명:** NFS 버전 2 프로토콜은 큰 파일을 처리할 수 없습니다.

**해결책:** 큰 파일에 액세스해야 하는 경우에는 버전 3 또는 버전 4를 사용해야 합니다.

**NFS server hostname not responding still trying**

**설명:** 파일 관련 작업을 수행하는 중에 프로그램이 정지되는 경우 NFS 서버에 오류가 발생했을 수 있습니다. 이 메시지는 *hostname* NFS 서버가 다운되었거나, 서버 또는 네트워크에 문제가 발생했음을 나타냅니다.

**해결책:** 페일오버를 사용 중인 경우 *hostname*은 서버 목록입니다. [117 페이지 “NFS 클라이언트에서 연결을 확인하는 방법”](#)의 정보를 참조하여 문제 해결을 시작합니다.

**NFS server recovering**

**설명:** NFS 버전 4 서버 재부트 중에 일부 작업이 허용되지 않았습니다. 이 메시지는 클라이언트가 서버에서 해당 작업 진행을 허용하기를 기다리고 있음을 의미합니다.

**해결책:** 작업을 수행할 필요는 없습니다. 서버에서 작업을 허용할 때까지 기다립니다.

**Permission denied**

**설명:** 이 메시지는 다음과 같은 이유로 인해 `ls -l`, `getfacl` 및 `setfacl` 명령에서 표시합니다.

- NFS 버전 4 서버의 액세스 제어 목록(ACL) 항목에 있는 사용자 또는 그룹을 NFS 버전 4 클라이언트의 유효한 사용자 또는 그룹에 매핑할 수 없는 경우 해당 사용자는 클라이언트에서 ACL을 읽을 수 없습니다.
- NFS 버전 4 클라이언트에서 설정 중인 ACL 항목에 있는 사용자 또는 그룹을 NFS 버전 4 서버의 유효한 사용자 또는 그룹에 매핑할 수 없는 경우 해당 사용자는 클라이언트에서 ACL을 쓰거나 수정할 수 없습니다.
- NFS 버전 4 클라이언트와 서버의 NFSMAPID\_DOMAIN 값이 일치하지 않으면 ID 매핑이 실패합니다.

자세한 내용은 [181 페이지 “NFS 버전 4의 ACL 및 nfsmapid”](#)를 참조하십시오.

**해결책:** 다음을 수행합니다.



- ACL 항목의 모든 사용자 및 그룹 ID가 클라이언트와 서버에 모두 있는지 확인합니다.
- SMF 저장소에서 `nfsmapid_domain`의 값이 올바르게 설정되어 있는지 확인합니다.

서버 또는 클라이언트에서 매핑할 수 없는 사용자나 그룹이 있는지 확인하려면 [181 페이지 “매핑되지 않은 사용자 또는 그룹 ID 확인”](#)에서 제공되는 스크립트를 사용합니다.

`port number` in nfs URL not the same as `port number` in port option

**설명:** NFS URL에 포함된 포트 번호는 마운트할 `-port` 옵션에 포함된 포트 번호와 일치해야 합니다. 포트 번호가 일치하지 않으면 마운트가 실패합니다.

**해결책:** 포트 번호가 같도록 명령을 변경하거나 잘못된 포트 번호를 지정하지 마십시오. 일반적으로는 NFS URL과 `-port` 옵션 둘 다에서 포트 번호를 지정할 필요가 없습니다.

`replicas must have the same version`

**설명:** NFS 페일오버가 정상적으로 작동하려면 복제본인 NFS 서버에서 NFS 프로토콜의 동일 버전을 지원해야 합니다.

**해결책:** 여러 버전을 실행할 수는 없습니다.

`replicated mounts must be read-only`

**설명:** 읽기/쓰기로 마운트된 파일 시스템에서는 NFS 페일오버가 작동하지 않습니다. 파일 시스템을 읽기/쓰기로 마운트하면 파일 변경 가능성이 높아집니다.

**해결책:** NFS 페일오버는 파일 시스템이 동일해야 작동합니다.

`replicated mounts must not be soft`

**설명:** 복제된 마운트의 경우 페일오버가 수행되기 전에 시간이 초과될 때까지 기다려야 합니다.

**해결책:** `soft` 옵션을 사용하려면 시간 초과가 시작될 때 마운트가 즉시 실패해야 합니다. 따라서 복제된 마운트에는 `-soft` 옵션을 포함할 수 없습니다.

`share nfs: Cannot share more than one filesystem with 'public' option`

**해결책:** `/etc/dfs/dfstab` 파일에서 `-public` 옵션을 사용하여 공유할 파일 시스템이 하나만 선택되어 있는지 확인합니다. 공용 파일 핸들은 서버당 하나만 설정할 수 있으므로 이 옵션을 사용하여 서버당 하나의 파일 시스템만 공유할 수 있습니다.

**WARNING:** No network locking on `hostname:path`: contact admin to install server change

**설명:** NFS 클라이언트가 NFS 서버의 네트워크 잠금 관리자에 연결하려고 시도했으나 연결하지 못했습니다. 이 경고가 생성되는 경우 마운트가 실패하지는 않으며 잠금이 작동하지 않는다는 경고만 표시됩니다.

**해결책:** 잠금 관리자를 완전하게 지원하는 새 OS 버전으로 서버를 업그레이드하십시오.

## 네트워크 파일 시스템 액세스(참조)

이 장에서는 NFS 명령과 NFS 환경의 각 부분 및 이러한 부분이 함께 작동하는 방식에 대해 설명합니다.

- 131 페이지 “NFS 파일”
- 134 페이지 “NFS 데몬”
- 146 페이지 “NFS 명령”
- 165 페이지 “NFS 문제 해결용 명령”
- 170 페이지 “RDMA를 통한 NFS”
- 172 페이지 “NFS 서비스의 작동 방식”
- 193 페이지 “미러 마운트의 작동 방식”
- 194 페이지 “NFS 참조의 작동 방식”
- 195 페이지 “autofs 맵”
- 201 페이지 “autofs의 작동 방식”
- 212 페이지 “autofs 참조”

주 - 시스템에서 영역이 사용으로 설정된 경우 비전역 영역에서 이 기능을 사용하려면 **Oracle Solaris 관리: Oracle Solaris Zones, Oracle Solaris 10 Zones 및 리소스 관리**에서 자세한 내용을 참조하십시오.

## NFS 파일

컴퓨터에서 NFS 작업을 지원하려면 여러 파일이 필요합니다. 이러한 파일은 대부분 ASCII이지만 일부는 데이터 파일입니다. 표 6-1에 이러한 파일과 해당 기능의 목록이 나와 있습니다.

표 6-1 NFS 파일

파일 이름	기능
/etc/default/fs	로컬 파일 시스템의 기본 파일 시스템 유형이 나열되어 있습니다.

표 6-1 NFS 파일 (계속)

파일 이름	기능
/etc/default/nfslogd	NFS 서버 로깅 데몬 <code>nfslogd</code> 에 대한 구성 정보가 나열되어 있습니다.
/etc/dfs/dfstab	공유할 로컬 리소스가 나열되어 있습니다.
/etc/dfs/fstypes	원격 파일 시스템의 기본 파일 시스템 유형이 나열되어 있습니다.
/etc/dfs/sharetab	공유할 로컬 및 원격 리소스가 나열되어 있습니다. <a href="#">sharetab(4)</a> 매뉴얼 페이지를 참조하십시오. 이 파일은 편집하지 마십시오.
/etc/mnttab	현재 마운트되어 있는 파일 시스템(자동 마운트된 디렉토리 포함)이 나열되어 있습니다. <a href="#">mnttab(4)</a> 매뉴얼 페이지를 참조하십시오. 이 파일은 편집하지 마십시오.
/etc/netconfig	전송 프로토콜이 나열되어 있습니다. 이 파일은 편집하지 마십시오.
/etc/nfs/nfslog.conf	NFS 서버 로깅에 대한 일반 구성 정보가 나열되어 있습니다.
/etc/nfs/nfslogtab	<code>nfslogd</code> 의 로그 사후 처리에 대한 정보가 나열되어 있습니다. 이 파일은 편집하지 마십시오.
/etc/nfssec.conf	NFS 보안 서비스가 나열되어 있습니다.
/etc/rmtab	NFS 클라이언트에 의해 원격으로 마운트된 파일 시스템이 나열되어 있습니다. <a href="#">rmtab(4)</a> 매뉴얼 페이지를 참조하십시오. 이 파일은 편집하지 마십시오.
/etc/vfstab	로컬로 마운트할 파일 시스템을 정의합니다. <a href="#">vfstab(4)</a> 매뉴얼 페이지를 참조하십시오.

`/etc/dfs/fstypes`의 첫번째 항목은 보통 원격 파일 시스템에 대한 기본 파일 시스템 유형으로 사용됩니다. 이 항목은 NFS 파일 시스템 유형을 기본값으로 정의합니다.

`/etc/default/fs` 파일에는 항목이 하나만 포함되어 있으며, 해당 항목은 로컬 디스크의 기본 파일 시스템 유형입니다. `/kernel/fs`의 파일을 확인하여 클라이언트나 서버에서 지원되는 파일 시스템 유형을 파악할 수 있습니다.

## /etc/default/nfslogd 파일

이 파일은 NFS 서버 로깅 사용 시 사용되는 일부 매개변수를 정의합니다. 다음 매개변수를 정의할 수 있습니다.

### CYCLE\_FREQUENCY

로그 파일을 순환하기 전에 경과해야 하는 시간을 결정합니다. 기본값은 24시간입니다. 이 옵션은 로그 파일이 너무 커지지 않도록 방지하는 데 사용됩니다.

**IDLE\_TIME**

버퍼 파일에서 추가 정보를 확인하기 전에 `nfslogd`가 일시 정지 상태로 유지되는 시간(초)을 설정합니다. 이 매개변수는 구성 파일 확인 빈도도 결정합니다. 이 매개변수는 `MIN_PROCESSING_SIZE`와 함께 버퍼 파일 처리 빈도를 결정합니다. 기본값은 300초입니다. 이 값을 늘리면 확인 횟수를 줄여 성능을 개선할 수 있습니다.

**MAPPING\_UPDATE\_INTERVAL**

파일 핸들-경로 매핑 테이블의 레코드 업데이트 간격(초)을 지정합니다. 기본값은 86400초(1일)입니다. 이 매개변수는 파일 핸들-경로 매핑 테이블을 지속적으로 업데이트하지 않고도 최신 상태로 유지할 수 있도록 합니다.

**MAX\_LOGS\_PRESERVE**

저장할 로그 파일 수를 결정합니다. 기본값은 10입니다.

**MIN\_PROCESSING\_SIZE**

로그 파일을 처리하고 로그 파일에 쓰기 전에 버퍼 파일이 도달해야 하는 최소 바이트 수를 설정합니다. 이 매개변수는 `IDLE_TIME`과 함께 버퍼 파일 처리 빈도를 결정합니다. 기본값은 524288바이트입니다. 이 값을 늘리면 버퍼 파일 처리 횟수를 줄여 성능을 개선할 수 있습니다.

**PRUNE\_TIMEOUT**

파일 핸들-경로 매핑 레코드의 시간이 초과되어 레코드를 줄일 수 있을 때까지 경과해야 하는 시간을 선택합니다. 기본값은 168시간(7일)입니다.

**UMASK**

`nfslogd`에서 만드는 로그 파일의 파일 모드 생성 마스크를 지정합니다. 기본값은 0137입니다.

## /etc/nfs/nfslog.conf 파일

이 파일은 `nfslogd`에서 사용할 경로, 파일 이름 및 로깅 유형을 정의합니다. 각 정의는 *tag*와 연관됩니다. NFS 서버 로깅을 시작하려면 각 파일 시스템에 대해 *tag*를 식별해야 합니다. 전역 태그는 기본값을 정의합니다. 필요에 따라 각 태그와 함께 다음 매개변수를 사용할 수 있습니다.

**defaultdir=path**

로깅 파일의 기본 디렉토리 경로를 지정합니다. 별도로 지정하지 않는 경우 기본 디렉토리는 `/var/nfs`입니다.

**log=path/filename**

로그 파일의 경로 및 파일 이름을 설정합니다. 기본값은 `/var/nfs/nfslog`입니다.

**fhtable=path/filename**

파일 핸들-경로 데이터베이스 파일의 경로 및 파일 이름을 선택합니다. 기본값은 `/var/nfs/fhtable`입니다.

`buffer=path/filename`

버퍼 파일의 경로 및 파일 이름을 결정합니다. 기본 값은 `/var/nfs/nfslog_workbuffer`입니다.

`logformat=basic|extended`

사용자가 읽을 수 있는 로그 파일을 만들 때 사용할 형식을 선택합니다. 기본 형식을 사용하는 경우 일부 `ftpd` 데몬과 비슷한 로그 파일이 생성됩니다. 확장된 형식에서는 보다 상세한 보기가 제공됩니다.

경로를 지정하지 않으면 `defaultdir`에 의해 정의되는 경로가 사용됩니다. 절대 경로를 사용하여 `defaultdir`을 대체할 수도 있습니다.

파일을 보다 쉽게 식별하려면 별도의 디렉토리에 파일을 저장하십시오. 다음은 필요한 변경의 예입니다.

```
% cat /etc/nfs/nfslog.conf
#ident "@(#)nfslog.conf 1.5 99/02/21 SMI"
#
.
.
# NFS server log configuration file.
#

global defaultdir=/var/nfs \
    log=nfslog fhtable=fhtable buffer=nfslog_workbuffer

publicftp log=logs/nfslog fhtable=fh/fhtables buffer=buffers/workbuffer
```

이 예제에서는 `log=publicftp`와 공유하는 모든 파일 시스템에서 다음 값을 사용합니다.

- 기본 디렉토리는 `/var/nfs`입니다.
- 로그 파일은 `/var/nfs/logs/nfslog*`에 저장됩니다.
- 파일 핸들-경로 데이터베이스 테이블은 `/var/nfs/fh/fhtables`에 저장됩니다.
- 버퍼 파일은 `/var/nfs/buffers/workbuffer`에 저장됩니다.

절차 정보는 84 페이지 “NFS 서버 로깅을 사용으로 설정하는 방법”을 참조하십시오.

## NFS 데몬

시스템이 실행 레벨 3 또는 다중 사용자 모드로 진입하면 NFS 작업을 지원하기 위해 여러 데몬이 시작됩니다. `mountd` 및 `nfsd` 데몬은 서버 시스템에서 실행됩니다. 서버 데몬의 자동 시작은 `/etc/dfs/sharetab`에서 NFS 파일 시스템 유형으로 레이블이 지정된 항목이 있는지에 따라 달라집니다. NFS 파일 잠금을 지원하기 위해 `lockd` 및 `statd` 데몬이 NFS 클라이언트 및 서버에서 실행됩니다. 그러나 이전 NFS 버전과는 달리 NFS 버전 4에서는 `lockd`, `statd`, `mountd` 및 `nfslogd` 데몬이 사용되지 않습니다.

이 절에서는 다음 데몬에 대해 설명합니다.

- 135 페이지 “`automountd` 데몬”

- 136 페이지 “lockd 데몬”
- 137 페이지 “mountd 데몬”
- 137 페이지 “nfs4cbd 데몬”
- 137 페이지 “nfsd 데몬”
- 138 페이지 “nfslogd 데몬”
- 139 페이지 “nfsmapid 데몬”
- 145 페이지 “reparse 데몬”
- 145 페이지 “statd 데몬”

## automountd 데몬

이 데몬은 autofs 서비스의 마운트 및 마운트 해제 요청을 처리합니다. 명령의 구문은 다음과 같습니다.

```
automountd [ -Tnv ] [ -D name=value ]
```

명령은 다음과 같은 방식으로 동작합니다.

- -T는 추적을 사용으로 설정합니다.
- -n은 모든 autofs 노드에서 찾아보기를 사용 안함으로 설정합니다.
- -v는 모든 상태 메시지를 콘솔에 기록하도록 선택합니다.
- -D name=value는 name으로 지정된 자동 마운트 맵 변수의 value를 대체합니다.

자동 마운트 맵의 기본값은 /etc/auto\_master입니다. 문제 해결 시에는 -T 옵션을 사용합니다.

sharectl 명령을 통해서도 명령줄에서 지정하는 것과 같은 항목을 지정할 수 있습니다. 그러나 명령줄 옵션과는 달리 SMF 저장소에서는 서비스 다시 시작, 시스템 재부트 및 시스템 업그레이드 시에도 지정 내용이 보존됩니다. 다음과 같은 매개변수를 automountd 데몬에 대해 설정할 수 있습니다.

### automountd\_verbose

상태 메시지를 콘솔에 기록하며, automountd 데몬용 -v 인수와 동등합니다. 기본값은 FALSE입니다.

### nobrowse

모든 autofs 마운트 지점에 대해 찾아보기를 켜거나 끄며, automountd 용 -n 인수와 동등합니다. 기본값은 FALSE입니다.

### trace

각 RPC(원격 프로시저 호출)를 확장하고 표준 출력에 확장된 RPC를 표시합니다. 이 키워드는 automountd용 -T 인수와 동등합니다. 기본값은 0입니다. 0~5 사이의 값을 사용할 수 있습니다.

### environment

각 환경에 서로 다른 값을 지정할 수 있습니다. 이 키워드는 automountd용 -D 인수와 동등합니다. environment 매개변수는 여러 번 사용할 수 있습니다. 그러나 각 환경 지정에 대해 별도의 항목을 사용해야 합니다.

## lockd 데몬

이 데몬은 NFS 파일에 대한 레코드 잠금 작업을 지원합니다. lockd 데몬은 NLM(네트워크 잠금 관리자) 프로토콜에 대해 서버와 클라이언트 간 RPC 연결을 관리합니다. 일반적으로 이 데몬은 옵션을 사용하지 않고 시작됩니다. 이 명령에는 세 가지 옵션을 사용할 수 있습니다. **lockd(1M)** 매뉴얼 페이지를 참조하십시오. 이러한 옵션은 명령줄에서 사용할 수도 있고 **sharectl** 명령을 통해 매개변수를 설정할 수도 있습니다. 아래에는 설정 가능한 매개변수에 대한 설명이 나와 있습니다.

---

주 - LOCKD\_GRACE\_PERIOD 키워드 및 -g 옵션은 사라졌습니다. 사라진 키워드는 새 **grace\_period** 매개변수로 교체되었습니다. 두 키워드가 모두 설정된 경우 **grace\_period**의 값이 LOCKD\_GRACE\_PERIOD의 값을 대체합니다. 아래의 **grace\_period** 설명을 참조하십시오.

---

LOCKD\_GRACE\_PERIOD와 마찬가지로, **grace\_period=graceperiod** 매개변수는 서버가 재부트된 후 클라이언트가 NLM에서 제공하는 NFS 버전 3 잠금과 버전 4 잠금을 모두 재생 이용해야 하는 시간(초)을 설정합니다. 따라서 **grace\_period**의 값은 NFS 버전 3 및 NFS 버전 4 둘 다에 대해 잠금 복구의 유예 기간 길이를 제어합니다.

**lockd\_retransmit\_timeout=timeout** 매개변수는 잠금 요청을 원격 서버로 다시 전송할 때까지 기다릴 시간(초)을 선택합니다. 이 옵션은 NFS 클라이언트측 서비스에 적용됩니다. **timeout**의 기본값은 15초입니다. **timeout** 값을 줄이면 “잡음이 많은” 네트워크에서 NFS 클라이언트의 응답 시간을 단축할 수 있습니다. 그러나 이렇게 값을 변경하면 잠금 요청 빈도가 높아져 서버 로드가 추가될 수 있습니다. -t **timeout** 옵션을 포함해 데몬을 시작하여 명령줄에서도 같은 매개변수를 사용할 수 있습니다.

**lockd\_servers=nthreads** 매개변수는 서버가 연결당 처리하는 최대 동시 스레드 수를 지정합니다. **nthreads** 값은 NFS 서버에서 예상되는 로드를 기준으로 합니다. 기본값은 20입니다. TCP를 사용하는 각 NFS 클라이언트는 NFS 서버와의 단일 연결을 사용합니다. 따라서 각 클라이언트는 서버에서 최대 20개의 동시 스레드를 사용할 수 있습니다.

UDP를 사용하는 모든 NFS 클라이언트는 NFS 서버와의 단일 연결을 공유합니다. 이러한 조건 하에서 UDP 연결에 사용 가능한 스레드 수를 늘릴 수 있습니다. 최소 계산에서는 각 UDP 클라이언트에 대해 스레드를 2개 허용합니다. 그러나 이 숫자는 클라이언트의 작업 부하와 관련되므로 클라이언트당 스레드 2개는 충분하지 않을 수 있습니다. 스레드를 더 사용하는 경우에는 스레드 사용 시 NFS 서버에서 메모리가 더 많이 사용되는 단점이 있습니다. 그러나 스레드가 사용되지 않는 경우에는 **nthreads**를 늘려도 아무런 효과가 없습니다. **nthreads** 옵션을 포함해 데몬을 시작하여 명령줄에서도 같은 매개변수를 사용할 수 있습니다.



## mountd 데몬

이 데몬은 원격 시스템으로부터의 파일 시스템 마운트 요청을 처리하며 액세스 제어 기능을 제공합니다. mountd 데몬은 /etc/dfs/sharetab를 확인하여 원격 마운트에 사용 가능한 파일 시스템 및 원격 마운트 수행이 허용되는 시스템을 결정합니다. 이 명령에는 -v 옵션과 -r 옵션을 사용할 수 있습니다. [mountd\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

-v 옵션은 명령을 Verbose 모드로 실행합니다. NFS 서버가 클라이언트에게 부여해야 하는 액세스 권한을 결정할 때마다 콘솔에 메시지가 인쇄됩니다. 생성되는 정보는 클라이언트가 파일 시스템에 액세스할 수 없는 이유를 확인할 때 유용할 수 있습니다.

-r 옵션은 클라이언트로부터의 모든 이후 마운트 요청을 거부합니다. 이 옵션은 이미 파일 시스템이 마운트된 클라이언트에는 영향을 주지 않습니다.

---

주 - NFS 버전 4에서는 이 데몬이 사용되지 않습니다.

---

명령줄 옵션 외에 여러 SMF 매개변수를 사용하여 mountd 데몬을 구성할 수 있습니다.

### client\_versmin

NFS 클라이언트에서 사용할 최소 NFS 프로토콜 버전을 설정합니다. 기본값은 2입니다. 3과 4도 값으로 사용할 수 있습니다. [91 페이지 “NFS 서비스 설정”](#)을 참조하십시오.

### client\_versmax

NFS 클라이언트에서 사용할 최대 NFS 프로토콜 버전을 설정합니다. 기본값은 4입니다. 2와 3도 값으로 사용할 수 있습니다. [91 페이지 “NFS 서비스 설정”](#)을 참조하십시오.

## nfs4cbd 데몬

NFS 버전 4 클라이언트 전용으로 사용되는 nfs4cbd는 NFS 버전 4 콜백 프로그램의 통신 끝점을 관리합니다. 이 데몬에는 사용자가 액세스할 수 있는 인터페이스가 없습니다. 자세한 내용은 [nfs4cbd\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## nfsd 데몬

이 데몬은 다른 클라이언트 파일 시스템 요청을 처리합니다. 이 명령에는 여러 옵션을 사용할 수 있습니다. 전체 목록은 [nfsd\(1M\)](#) 매뉴얼 페이지를 참조하십시오. 이러한 옵션은 명령줄에서 사용할 수도 있고 sharectl 명령을 통해 적절한 SMF 매개변수를 설정하여 사용할 수도 있습니다.

listen\_backlog=length 매개변수는 NFS 및 TCP에 대한 연결 지향 전송의 연결 대기열 길이를 설정합니다. 기본값은 32개 항목입니다. -l 옵션을 포함해 nfsd를 시작하여 명령줄에서도 같은 항목을 선택할 수 있습니다.

`max_connections=#-conn` 매개변수는 연결 지향 전송당 최대 연결 수를 선택합니다. `#-conn`의 기본값은 무제한입니다. `-c #-conn` 옵션을 포함해 데몬을 시작하여 명령줄에서도 같은 매개변수를 사용할 수 있습니다.

`servers=nservers` 매개변수는 서버에서 처리할 수 있는 최대 동시 요청 수를 선택합니다. `nserver`의 기본값은 16입니다. `nserver` 옵션을 포함해 `nfsd`를 시작하여 명령줄에서도 같은 항목을 선택할 수 있습니다.

이 데몬의 이전 버전과는 달리 `nfsd`는 동시 요청을 처리하기 위해 여러 복사본을 만들지 않습니다. `ps`를 사용하여 프로세스 테이블을 확인하면 실행 중인 데몬 복사본만 표시됩니다.

이러한 SMF 매개변수를 사용하여 `mountd` 데몬을 구성할 수도 있습니다. 이러한 매개변수에 해당하는 명령줄 항목은 없습니다.

#### server\_versmin

서버에서 등록 및 제공할 최소 NFS 프로토콜 버전을 설정합니다. 기본값은 2입니다. 3과 4도 값으로 사용할 수 있습니다. 91 페이지 “NFS 서비스 설정”을 참조하십시오.

#### server\_versmax

서버에서 등록 및 제공할 최대 NFS 프로토콜 버전을 설정합니다. 기본값은 4입니다. 2와 3도 값으로 사용할 수 있습니다. 91 페이지 “NFS 서비스 설정”을 참조하십시오.

#### server\_delegation

CNFS 버전 4 위임 기능이 서버에 대해 사용으로 설정되는지 여부를 제어합니다. 이 기능이 사용으로 설정된 경우 서버는 NFS 버전 4 클라이언트에 위임 제공을 시도합니다. 기본적으로 서버 위임은 사용으로 설정됩니다. 서버 위임을 사용 안함으로 설정하려면 93 페이지 “서버에서 다른 NFS 버전을 선택하는 방법”을 참조하십시오. 자세한 내용은 179 페이지 “NFS 버전 4의 위임”을 참조하십시오.

## nfslogd 데몬

이 데몬은 작동 로깅을 제공합니다. 서버에 대해 기록되는 NFS 작업은 `/etc/default/nfslogd`에 정의된 구성 옵션을 기반으로 합니다. NFS 서버 로깅이 사용으로 설정된 경우 선택한 파일 시스템에서 모든 RPC 작업의 레코드가 커널에 의해 버퍼 파일에 기록됩니다. 그런 후에 `nfslogd`가 이러한 요청을 사후 처리합니다. 이름 서비스 스위치는 UID를 로그인에, IP 주소를 호스트 이름에 매핑하는 데 사용됩니다. 식별된 이름 서비스를 통해 일치하는 항목을 찾을 수 없으면 번호가 기록됩니다.

경로 이름에 대한 파일 핸들 매핑도 `nfslogd`에 의해 처리됩니다. 데몬은 이러한 매핑을 파일 핸들-경로 매핑 테이블에서 추적합니다. `/etc/nfs/nfslogd`에서 식별되는 각 태그에 대해 매핑 테이블이 하나씩 있습니다. 사후 처리 후 레코드는 ASCII 로그 파일에 기록됩니다.

---

주 - NFS 버전 4에서는 이 데몬이 사용되지 않습니다.

---

## nfsmapid 데몬

NFS 프로토콜 버전 4(RFC3530)에서는 클라이언트와 서버 간에 사용자 또는 그룹 식별자(UID 또는 GID)를 교환하는 방식이 변경되었습니다. 이 프로토콜에서는 파일 소유자 및 그룹 속성을 NFS 버전 4 클라이언트와 NFS 버전 4 서버 간에 각각 `user@nfsv4_domain` 또는 `group@nfsv4_domain` 형식의 문자열로 교환해야 합니다.

예를 들어 정규화된 호스트 이름이 `system.example.com` 인 NFS 버전 4 클라이언트에서 `known_user` 사용자의 UID가 123456이라고 가정해 보겠습니다. 이 클라이언트가 NFS 버전 4 서버에 요청을 하려면 UID 123456을 `known_user@example.com`에 매핑한 다음 해당 속성을 NFS 버전 4 서버로 보내야 합니다. NFS 버전 4 서버에서는 `user_or_group@nfsv4_domain` 형식의 사용자 및 그룹 파일 속성을 받습니다. 서버에서는 클라이언트로부터 `known_user@example.com`을 받은 후 해당 문자열을 로컬 UID 123456에 매핑하며, 그러면 기본 파일 시스템에서 이를 인식할 수 있습니다. 이 기능은 네트워크의 모든 UID 및 GID가 고유하며 클라이언트의 NFS 버전 4 도메인이 서버의 NFS 버전 4 도메인과 일치한다고 가정합니다.

---

주 - 서버에서 지정된 사용자 또는 그룹 이름을 인식하지 못하면 NFS 버전 4 도메인이 일치해도 서버가 해당 사용자 또는 그룹 이름을 고유 ID(정수 값)에 매핑할 수 없습니다. 이러한 경우 서버는 인바운드 사용자 또는 그룹 이름을 `nobody` 사용자에게 매핑합니다. 이러한 상황을 방지하려면 관리자가 NFS 버전 4 클라이언트에만 있는 특수 계정을 만들지 않아야 합니다.

---

NFS 버전 4 클라이언트와 서버는 모두 정수에서 문자열 및 문자열에서 정수로의 변환을 수행할 수 있습니다. 예를 들어 GETATTR 작업에 대한 응답으로 NFS 서버 4 서버는 기본 파일 시스템에서 가져온 UID 및 GID를 해당하는 문자열 표현에 매핑하고 이 정보를 클라이언트로 보냅니다. 클라이언트 역시 UID와 GID를 문자열 표현으로 매핑해야 합니다. 예를 들어 `chown` 명령에 대한 응답으로 클라이언트는 새 UID :또는 GID를 문자열 표현에 매핑한 후에 SETATTR 작업을 서버로 보냅니다.

그러나 클라이언트와 서버는 인식되지 않은 문자열에 대해서는 다른 방식으로 응답합니다.

- 사용자가 서버에 없으면 NFS 버전 4 도메인 구성이 같아도 서버에서 RPC(원격 프로시저 호출)을 거부하고 클라이언트에 오류 메시지를 반환합니다. 이러한 상황에서는 원격 사용자가 수행할 수 있는 작업이 제한됩니다.
- 클라이언트와 서버에 모두 사용자가 있지만 사용자의 도메인이 불일치하는 경우에는 서버가 기본 파일 시스템에서 인식할 수 있는 정수 값에 인바운드 사용자 문자열을 매핑해야 하는 SETATTR 등의 속성 수정 작업을 거부합니다. NFS 버전 4 클라이언트와 서버가 정상적으로 작동하려면 해당 NFS 버전 4 도메인(문자열에서 @ 기호 뒷부분)이 일치해야 합니다.
- NFS 버전 4 클라이언트는 서버의 사용자 또는 그룹 이름을 인식하지 못하는 경우 문자열을 고유 ID(정수 값)에 매핑할 수 없습니다. 이러한 경우 클라이언트는 인바운드 사용자 또는 그룹 문자열을 `nobody` 사용자에게 매핑합니다. 이와 같은 `nobody`에 대한 매핑에서 각 응용 프로그램별로 여러 가지 문제가 발생합니다. NFS 버전 4 기능의 경우 파일 속성을 수정하는 작업이 실패합니다.

다음 옵션이 포함된 `sharectl` 명령을 사용하여 클라이언트 및 서버의 도메인 이름을 변경할 수 있습니다.

#### `nfsmapid_domain`

클라이언트 및 서버에 대해 공통 도메인을 설정합니다. 로컬 DNS 도메인 이름을 사용하는 기본 동작을 대체합니다. 자세한 내용은 [91 페이지 “NFS 서비스 설정”](#)을 참조하십시오.

## 구성 파일 및 `nfsmapid`

아래에서는 `nfsmapid` 데몬이 `svc:system/name-service/switch` 및 `svc:/network/dns/client`에서 찾은 SMF 구성 정보를 사용하는 방법에 대해 설명합니다.

- `nfsmapid`는 표준 C 라이브러리 함수를 사용하여 백엔드 이름 서비스에서 암호 및 그룹 정보를 요청합니다. 이러한 이름 서비스는 `svc:system/name-service/switch` SMF 서비스의 설정을 통해 제어됩니다. 서비스 등록 정보의 변경 내용은 `nfsmapid` 작업에 영향을 줍니다. `svc:system/name-service/switch` SMF 서비스에 대한 자세한 내용은 [nsswitch.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오.
- NFS 버전 4 클라이언트가 다른 도메인에서 파일 시스템을 마운트할 수 있도록 하기 위해 `nfsmapid`는 DNS TXT 리소스 레코드(RR)인 `_nfsv4idmapdomain`의 구성을 사용합니다. `_nfsv4idmapdomain` 리소스 레코드 구성에 대한 자세한 내용은 [141 페이지 “nfsmapid 및 DNS TXT 레코드”](#)를 참조하십시오. 또한 다음 사항을 확인하십시오.
  - 원하는 도메인 정보를 사용하여 DNS 서버에서 DNS TXT RR을 명시적으로 구성해야 합니다.
  - 원하는 매개변수를 사용하여 `svc:system/name-service/switch` SMF 서비스를 구성해야 `resolver`가 클라이언트 및 서버 NFS 버전 4 도메인에 대해 DNS 서버를 찾고 TXT 레코드를 검색할 수 있습니다.

자세한 내용은 다음을 참조하십시오.

- 141 페이지 “우선 순위 규칙”
- 143 페이지 “NFS 버전 4 기본 도메인 구성”
- `resolv.conf(4)` 매뉴얼 페이지

## 우선 순위 규칙

`nfsmapid`가 정상적으로 작동하려면 NFS 버전 4 클라이언트 및 서버의 도메인이 같아야 합니다. NFS 버전 4 도메인이 일치하도록 하기 위해 `nfsmapid`는 다음과 같은 엄격한 우선 순위 규칙을 따릅니다.

1. 데몬이 먼저 SMF 저장소에서 `nfsmapid_domain` 매개변수에 지정된 값을 확인합니다. 값을 찾으면 지정된 값이 다른 설정보다 우선합니다. 지정된 값은 송신 속성 문자열에 추가되며 수신 속성 문자열과 비교됩니다. 절차 정보는 91 페이지 “NFS 서비스 설정”을 참조하십시오.

---

주-NFSMAPID\_DOMAIN 설정 사용 시에는 확장이 불가능하므로 대규모 배치에서는 사용하지 않는 것이 좋습니다.

---

2. `nfsmapid_domain`에 값이 지정되지 않은 경우 데몬은 DNS TXT RR에서 도메인 이름을 확인합니다. `nfsmapid`는 resolver의 루틴 세트에 사용되는 `/etc/resolv.conf` 파일의 지시어를 사용합니다. resolver는 구성된 DNS 서버에서 `_nfsv4idmapdomain` TXT RR을 검색합니다. DNS TXT 레코드를 사용하는 경우 보다 확장이 용이합니다. 따라서 SMF 저장소에서 매개변수를 설정하는 것보다는 TXT 레코드를 계속 사용하는 경우가 많습니다.

3. 도메인 이름을 제공하는 DNS TXT 레코드가 구성되어 있지 않으면 `nfsmapid` 데몬은 `/etc/resolv.conf` 파일의 `domain` 또는 `search` 지시어를 사용하며, 이때 마지막으로 지정된 지시어가 우선적으로 사용됩니다.

`domain` 및 `search` 지시어가 모두 사용되는 다음 예에서는 `nfsmapid` 데몬이 `search` 지시어 다음에 나열된 첫 번째 도메인(`company.com`)을 사용합니다.

```
domain example.company.com
search company.com foo.bar.com
```

4. `/etc/resolv.conf` 파일이 없으면 `nfsmapid`는 `domainname` 명령의 동작에 따라 NFS 버전 4 도메인 이름을 가져옵니다. 구체적으로, `/etc/defaultdomain` 파일이 있으면 `nfsmapid`는 NFS 버전 4 도메인에 대해 해당 파일의 콘텐츠를 사용합니다. `/etc/defaultdomain` 파일이 없으면 `nfsmapid`는 네트워크의 구성된 이름 지정 서비스에서 제공하는 도메인 이름을 사용합니다. 자세한 내용은 `domainname(1M)` 매뉴얼 페이지를 참조하십시오.

## `nfsmapid` 및 DNS TXT 레코드

DNS는 다양한 용도로 사용되므로 NFS 버전 4 도메인 이름에 효율적인 저장 및 배포 방식을 제공합니다. 또한, DNS는 기본적으로 확장이 가능하므로 대규모 배치에서는 NFS

버전 4 도메인 이름을 구성할 때 DNS TXT 레코드가 기본적으로 사용됩니다. 엔터프라이즈 레벨 DNS 서버에서 `_nfsv4idmapdomain` TXT 레코드를 구성해야 합니다. 이와 같이 구성하면 모든 NFS 버전 4 클라이언트 또는 서버가 DNS 트리를 순화하면서 NFS 버전 4 도메인을 찾을 수 있습니다.

다음은 DNS 서버가 NFS 버전 4 도메인 이름을 제공할 수 있도록 설정하는 데 기본적으로 사용되는 항목의 예입니다.

```
_nfsv4idmapdomain      IN      TXT      "foo.bar"
```

이 예에서 구성할 도메인 이름은 큰따옴표로 묶인 값입니다. `ttl` 필드는 지정되어 있지 않으며 `_nfsv4idmapdomain` (owner 필드의 값)에는 도메인이 추가되지 않습니다. 이와 같이 구성하면 TXT 레코드가 Start-Of-Authority(SOA) 레코드에서 영역의 `_${ORIGIN}` 항목을 사용할 수 있습니다. 예를 들어 서로 다른 도메인 네임스페이스 레벨에서 레코드를 다음과 같이 읽을 수 있습니다.

```
_nfsv4idmapdomain.subnet.yourcorp.com.    IN      TXT      "foo.bar"
_nfsv4idmapdomain.yourcorp.com.           IN      TXT      "foo.bar"
```

이 구성에서는 DNS 클라이언트가 보다 유동적으로 `resolv.conf` 파일을 사용하여 DNS 트리 계층을 검색할 수 있습니다. [resolv.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오. 이 기능을 사용하면 TXT 레코드를 찾을 확률이 더 높아집니다. 유동성을 높이기 위해 더 낮은 레벨의 DNS 하위 도메인에서는 자체 DNS TXT 리소스 레코드(RR)를 정의할 수 있습니다. 이 기능을 통해 더 낮은 레벨의 DNS 하위 도메인이 최상위 레벨 DNS 도메인에서 정의한 TXT 레코드를 대체할 수 있습니다.

---

주 - TXT 레코드를 통해 지정되는 도메인은 NFS 버전 4를 사용하는 클라이언트와 서버의 DNS 도메인과 일치하지 않는 임의의 문자열일 수 있습니다. NFS 버전 4 데이터를 다른 DNS 도메인과 공유하지 않을 수 있습니다.

---

## NFS 버전 4 도메인 확인

네트워크의 NFS 버전 4 도메인에 대해 값을 지정하기 전에 NFS 버전 4 도메인이 네트워크에 대해 이미 구성되었는지 확인하십시오. 다음 예에서는 네트워크의 NFS 버전 4 도메인을 식별하는 방법을 제공합니다.

- DNS TXT RR에서 NFS 버전 4 도메인을 식별하려면 `nslookup` 또는 `dig` 명령을 사용합니다.

아래에는 `nslookup` 명령의 샘플 출력이 나와 있습니다.

```
# nslookup -q=txt _nfsv4idmapdomain
Server:      10.255.255.255
Address:     10.255.255.255#53

_nfsv4idmapdomain.example.company.com text = "company.com"
```

`dig` 명령의 경우에도 이 샘플 출력을 참조하십시오.

```
# dig +domain=example.company.com -t TXT _nfsv4idmapdomain
...
;; QUESTION SECTION:
;_nfsv4idmapdomain.example.company.com. IN      TXT

;; ANSWER SECTION:
_nfsv4idmapdomain.example.company.com. 21600 IN TXT    "company.com"

;; AUTHORITY SECTION:
...
```

DNS TXT RR을 설정하는 방법에 대한 자세한 내용은 [141 페이지 “nfsmapid 및 DNS TXT 레코드”](#)를 참조하십시오.

- 네트워크가 NFS 버전 4 DNS TXT RR로 설정되어 있지 않은 경우 다음 명령을 실행하여 DNS 도메인 이름에서 NFS 버전 4 도메인을 식별합니다.

```
# egrep domain /etc/resolv.conf
domain example.company.com
```

- 클라이언트에 대해 DNS 도메인 이름을 제공하기 위해 `/etc/resolv.conf` 파일이 구성되어 있지 않은 경우 다음 명령을 사용하여 네트워크의 NFS 버전 4 도메인 구성에서 도메인을 식별합니다.

```
# cat /system/volatile/nfs4_domain
company.com
```

- NIS 등의 다른 이름 지정 서비스를 사용하는 경우 다음 명령을 사용하여 네트워크에 대해 구성된 이름 지정 서비스의 도메인을 식별합니다.

```
# domainname
it.example.company.com
```

자세한 내용은 다음 매뉴얼 페이지를 참조하십시오.

- [nslookup\(1M\)](#)
- [dig\(1M\)](#)
- [resolv.conf\(4\)](#)
- [domainname\(1M\)](#)

## NFS 버전 4 기본 도메인 구성

이 절에서는 네트워크에서 필요한 기본 도메인을 가져오는 방법에 대해 설명합니다.

- 최신 릴리스의 경우 [143 페이지 “Oracle Solaris 11 릴리스에서 NFS 버전 4 기본 도메인 구성”](#)을 참조하십시오.
- 초기 Solaris 10 릴리스의 경우에는 [144 페이지 “Solaris 10 릴리스에서 NFS 버전 4 기본 도메인 구성”](#)을 참조하십시오.

## Oracle Solaris 11 릴리스에서 NFS 버전 4 기본 도메인 구성

Oracle Solaris 11 릴리스에서는 다음 명령을 입력하여 명령줄에서 기본 NFS 도메인 버전을 설정할 수 있습니다.



---

```
# sharectl set -p nfsmapid_domain=example.com nfs
```

---

주 - DNS는 기본적으로 다양한 용도로 사용되며 확장이 가능하므로, 대규모 NFS 버전 4 배치의 도메인을 구성할 때는 DNS TXT 레코드가 계속 사용되고 있으며 사용하는 것이 좋습니다. [141 페이지 “nfsmapid 및 DNS TXT 레코드”](#)를 참조하십시오.

---

## Solaris 10 릴리스에서 NFS 버전 4 기본 도메인 구성

NFS 버전 4의 초기 Solaris 10 릴리스에서는 네트워크에 여러 DNS 도메인이 포함되어 있는데 UID 및 GID 네임스페이스는 하나뿐이면 모든 클라이언트는 `nfsmapid_domain`에 대해 하나의 값을 사용해야 합니다. DNS를 사용하는 사이트의 경우 `nfsmapid`는 `_nfsv4idmapdomain`에 지정된 값에서 도메인 이름을 가져옴으로써 이 문제를 해결합니다. 자세한 내용은 [141 페이지 “nfsmapid 및 DNS TXT 레코드”](#)를 참조하십시오. 네트워크가 DNS를 사용하도록 구성되어 있지 않은 경우 첫번째 시스템 부트 중에 OS에서 `sysidconfig` 유틸리티를 사용하여 NFS 버전 4 도메인 이름에 대해 다음 프롬프트를 제공합니다.

```
This system is configured with NFS version 4, which uses a
domain name that is automatically derived from the system's
name services. The derived domain name is sufficient for most
configurations. In a few cases, mounts that cross different
domains might cause files to be owned by nobody due to the
lack of a common domain name.
```

```
Do you need to override the system's default NFS version 4 domain
name (yes/no)? [no]
```

기본 응답은 `[no]`입니다. `[no]`를 선택하면 다음이 표시됩니다.

```
For more information about how the NFS version 4 default domain name is
derived and its impact, refer to the man pages for nfsmapid(1M) and
nfs(4), and the System Administration Guide: Network Services.
```

`[yes]`를 선택하면 다음 프롬프트가 표시됩니다.

```
Enter the domain to be used as the NFS version 4 domain name.
NFS version 4 domain name []:
```

---

주 - `nfsmapid_domain`의 값이 SMF 저장소에 있는 경우 사용자가 제공하는 `[domain_name]`이 해당 값을 대체합니다.

---

## nfsmapid 관련 추가 정보

`nfsmapid`에 대한 자세한 내용은 다음을 참조하십시오.

- [nfsmapid\(1M\)](#) 매뉴얼 페이지
- [nfs\(4\)](#) 매뉴얼 페이지
- <http://www.ietf.org/rfc/rfc1464.txt>



- 181 페이지 “NFS 버전 4의 ACL 및 `nfsmapid`”

## repared 데몬

`repared` 데몬은 구문 재분석 지점과 연관된 데이터를 해석합니다. 이 데이터는 SMB 및 NFS 파일 서버에서 DFS 및 NFS 참조에 사용됩니다. 이 서비스는 SMF에서 관리하며 수동으로 시작해서는 안 됩니다.

## statd 데몬

이 데몬은 `lockd`와 함께 작동하여 잠금 관리자에 대해 충돌 및 복구 기능을 제공합니다. `statd` 데몬은 NFS 서버에 대한 잠금을 보유한 클라이언트를 추적합니다. 서버가 충돌하면 재부트 시 서버의 `statd`가 클라이언트의 `statd`에 연결합니다. 그러면 클라이언트 `statd`는 서버에서 잠금 재생 이용을 시도할 수 있습니다. 또한 클라이언트가 충돌하면 서버의 클라이언트 잠금을 지울 수 있도록 클라이언트 `statd`가 서버 `statd`에 알림을 보냅니다. 이 데몬에서는 선택할 수 있는 옵션이 없습니다. 자세한 내용은 [statd\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

Solaris 7 릴리스에서는 `statd`가 클라이언트를 추적하는 방식이 개선되었습니다. 모든 이전 Solaris 릴리스에서 `statd`는 클라이언트의 수식되지 않은 호스트 이름을 사용하여 각 클라이언트에 대해 `/var/statmon/sm`에 파일을 만들었습니다. 이러한 파일 이름 지정으로 인해, 서로 다른 도메인의 두 클라이언트가 호스트 이름을 공유하거나 클라이언트가 NFS 서버와 같은 도메인에 있지 않은 경우 문제가 발생했습니다. 수식되지 않은 호스트 이름은 호스트 이름만 나열하고 도메인이나 IP 주소 정보는 포함하지 않으므로, 이전 버전 `statd`는 이러한 유형의 클라이언트를 구분할 수 없었습니다. 이 문제를 해결하기 위해 Solaris 7 `statd`는 클라이언트의 IP 주소를 사용하여 수식되지 않은 호스트 이름에 대한 심볼릭 링크를 `/var/statmon/sm`에 만듭니다. 새로운 링크는 다음과 같습니다.

```
# ls -l /var/statmon/sm
lrwxrwxrwx 1 daemon 11 Apr 29 16:32 ipv4.192.168.255.255 -> myhost
lrwxrwxrwx 1 daemon 11 Apr 29 16:32 ipv6.fec0::56:a00:20ff:feb9:2734 -> v6host
--w----- 1 daemon 11 Apr 29 16:32 myhost
--w----- 1 daemon 11 Apr 29 16:32 v6host
```

이 예제에서 클라이언트 호스트 이름은 `myhost`이고 클라이언트의 IP 주소는 `192.168.255.255`입니다. 이름이 `myhost`인 다른 호스트가 파일 시스템을 마운트하는 경우 두 심볼릭 링크는 호스트 이름으로 연결됩니다.

---

주-NFS 버전 4에서는 이 데몬이 사용되지 않습니다.

---

## NFS 명령

이러한 명령은 root로 실행해야 완전히 적용되지만, 모든 사용자가 정보를 요청할 수는 있습니다.

- 146 페이지 “automount 명령”
- 147 페이지 “clear\_locks 명령”
- 147 페이지 “fsstat 명령”
- 148 페이지 “mount 명령”
- 154 페이지 “mountall 명령”
- 165 페이지 “nfsref 명령”
- 164 페이지 “setmnt 명령”
- 155 페이지 “sharectl 명령”
- 158 페이지 “share 명령”
- 163 페이지 “shareall 명령”
- 163 페이지 “showmount 명령”
- 153 페이지 “umount 명령”
- 155 페이지 “umountall 명령”
- 162 페이지 “unshare 명령”
- 163 페이지 “unshareall 명령”

## automount 명령

이 명령은 autofs 마운트 지점을 설치하고 automaster 파일의 정보를 각 마운트 지점과 연관시킵니다. 명령의 구문은 다음과 같습니다.

```
automount [ -t duration ] [ -v ]
```

-t *duration*은 파일 시스템이 마운트된 상태로 유지되는 시간(초)을 설정하고 -v는 Verbose 모드를 선택합니다. 이 명령을 Verbose 모드로 실행하면 문제를 보다 쉽게 해결할 수 있습니다.

구체적으로 설정하지 않는 경우 기간의 값은 5분으로 설정됩니다. 대부분의 경우에는 기본값을 사용하면 됩니다. 그러나 자동 마운트된 파일 시스템이 많은 시스템에서는 기간 값을 높여야 할 수 있습니다. 특히, 서버의 활성 사용자가 많은 경우에는 5분마다 자동 마운트된 파일 시스템을 확인하는 것은 효율적이지 않을 수 있습니다. autofs 파일 시스템을 1800초(30분)마다 확인하는 것이 보다 적절할 수 있습니다. 파일 시스템의 마운트를 5분마다 해제하지 않으면 /etc/mnttab가 커질 수 있습니다. df가 /etc/mnttab의 각 항목을 확인할 때 출력을 줄이려면 -F 옵션을 사용(df(1M) 매뉴얼 페이지 참조)하거나 egrep를 사용하여 df에서 출력을 필터링하면 됩니다.

기간을 변경하면 자동 마운트 맵에 변경 내용이 반영되는 속도도 변경된다는 점을 고려해야 합니다. 파일 시스템 마운트를 해제해야 변경 내용을 확인할 수 있습니다. 자동 마운트 맵을 수정하는 방법의 지침은 103 페이지 “맵 수정”을 참조하십시오.

`sharectl` 명령을 통해서도 명령줄에서 지정하는 것과 같은 항목을 지정할 수 있습니다. 그러나 명령줄 옵션과는 달리 SMF 저장소에서는 서비스 다시 시작, 시스템 재부트 및 시스템 업그레이드 시에도 지정 내용이 보존됩니다. 다음과 같은 매개변수를 `automount` 명령에 대해 설정할 수 있습니다.

#### `timeout`

파일 시스템 마운트를 해제할 때까지 파일 시스템이 유휴 상태로 유지되는 기간을 설정합니다. 이 키워드는 `automount` 명령용 `-t` 인수와 동등합니다. 기본값은 600입니다.

#### `automount_verbose`

`autofs` 마운트, 마운트 해제 및 기타 필수적이지 않은 이벤트에 대한 알림을 제공합니다. 이 키워드는 `automountd`용 `-v` 인수와 동등합니다. 기본값은 `FALSE`입니다.

## clear\_locks 명령

이 명령을 사용하면 NFS 클라이언트에 대한 모든 파일, 레코드 및 공유 잠금을 제거할 수 있습니다. 이 명령은 `root`에서 실행해야 합니다. NFS 서버에서 특정 클라이언트에 대한 잠금을 지울 수 있습니다. NFS 클라이언트에서 특정 서버의 해당 클라이언트에 대한 잠금을 지울 수 있습니다. 다음 예제에서는 현재 시스템에서 이름이 `tulip`인 NFS 클라이언트에 대한 잠금을 지웁니다.

```
# clear_locks tulip
```

`-s` 옵션을 사용하면 잠금을 지울 NFS 호스트를 지정할 수 있습니다. 이 옵션은 잠금을 만든 NFS 클라이언트에서 실행해야 합니다. 이 경우 클라이언트의 잠금은 `bee`라는 NFS 서버에서 제거됩니다.

```
# clear_locks -s bee
```



주의 - 이 명령은 클라이언트가 충돌하여 잠금을 지울 수 없는 경우에만 실행해야 합니다. 데이터 손상 문제를 방지하려면 활성 클라이언트에 대한 잠금을 지우지 마십시오.

## fsstat 명령

`fsstat` 유틸리티를 사용하면 파일 시스템 유형 및 마운트 지점별로 파일 시스템 작업을 모니터링할 수 있습니다. 다양한 옵션을 통해 출력을 사용자 정의할 수 있습니다. 다음 예를 참조하십시오.

이 예에서는 NFS 버전 3, 버전 4 및 `root` 마운트 지점에 대한 출력을 보여줍니다.

```
% fsstat nfs3 nfs4 /
new      name  name  attr  attr  lookup  rddir  read  read  write  write
file     remov chng  get   set   ops     ops   ops  bytes ops   bytes
```

```

3.81K      90 3.65K 5.89M 11.9K 35.5M 26.6K 109K 118M 35.0K 8.16G nfs3
759       503 457 93.6K 1.44K 454K 8.82K 65.4K 827M 292 223K nfs4
25.2K    18.1K 1.12K 54.7M 1017 259M 1.76M 22.4M 20.1G 1.43M 3.77G /

```

이 예에서는 `-i` 옵션을 사용하여 NFS 버전 3, 버전 4 및 `root` 마운트 지점의 I/O 작업에 대한 통계를 제공합니다.

```

% fsstat -i nfs3 nfs4 /
  read   read   write  write  rddir  rddir  rwlock  rwlock
  ops   bytes   ops   bytes  ops   bytes  ops    ops
109K   118M   35.0K 8.16G 26.6K 4.45M 170K   170K  nfs3
65.4K  827M   292   223K 8.82K 2.62M 74.1K  74.1K  nfs4
22.4M  20.1G   1.43M 3.77G 1.76M 3.29G 25.5M  25.5M  /

```

이 예에서는 `-n` 옵션을 사용하여 NFS 버전 3, 버전 4 및 `root` 마운트 지점의 이름 지정 작업에 대한 통계를 보여줍니다.

```

% fsstat -n nfs3 nfs4 /
lookup  creat  remov  link  renam  mkdir  rmdir  rddir  symlnk  rdlnk
35.5M   3.79K  90     2     3.64K  5      0     26.6K  11     136K  nfs3
454K    403   503    0     101   0      0     8.82K  356    1.20K  nfs4
259M    25.2K 18.1K  114   1017  10     2     1.76M  12     8.23M  /

```

자세한 내용은 [fsstat\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## mount 명령

이 명령을 사용하면 이름이 지정된 파일 시스템(로컬 또는 원격)을 지정된 마운트 지점에 연결할 수 있습니다. 자세한 내용은 [mount\(1M\)](#) 매뉴얼 페이지를 참조하십시오. 인수 없이 사용되는 `mount`는 컴퓨터에 현재 마운트되어 있는 파일 시스템 목록을 표시합니다.

대부분의 파일 시스템 유형은 표준 Oracle Solaris 설치에 포함되어 있습니다. 각 파일 시스템 유형에는 해당 파일 시스템 유형에 적합한 `mount`용 옵션이 나열되는 특정 매뉴얼 페이지가 있습니다. NFS 파일 시스템의 매뉴얼 페이지는 [mount\\_nfs\(1M\)](#)입니다. UFS 파일 시스템의 경우 [mount\\_ufs\(1M\)](#)을 참조하십시오.

Solaris 7 릴리스에는 표준 `server:pathname`구문 대신 NFS URL을 사용하여 NFS 서버에서 마운트할 경로 이름을 선택하는 기능이 포함되어 있습니다. 자세한 내용은 [90 페이지 "NFS URL을 사용하여 NFS 파일 시스템을 마운트하는 방법"](#)을 참조하십시오.



**주의** - `mount` 명령 버전에서는 잘못된 옵션에 대한 경고를 표시하지 않습니다. 명령은 해석할 수 없는 옵션을 자동으로 무시합니다. 예기치 않은 동작을 방지하려면 사용된 모든 옵션을 확인하십시오.

## NFS 파일 시스템용 mount 옵션

이후 내용에서는 NFS 파일 시스템을 마운트할 때 `-o` 플래그 뒤에 올 수 있는 몇 가지 옵션에 대해 소개합니다. 전체 옵션 목록은 [mount\\_nfs\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

**bg|fg**

이 옵션을 사용하여 마운트 실패 시의 재시도 동작을 선택할 수 있습니다. **bg** 옵션을 사용하는 경우 마운트 시도를 백그라운드에서 실행합니다. **fg** 옵션을 사용하는 경우 마운트 시도를 전경에서 실행합니다. 기본값은 **fg**입니다. 사용 가능해야 하는 파일 시스템에서는 이 옵션이 가장 적합합니다. 이 옵션을 선택하면 마운트가 완료될 때까지 추가 처리를 수행할 수 없습니다. 중요하지 않은 파일 시스템의 경우에는 **bg**를 선택하는 것이 좋은데, 마운트 요청이 완료되도록 기다리는 동안 클라이언트가 다른 처리를 수행할 수 있기 때문입니다.

**forcedirectio**

이 옵션은 큰 순차적 데이터 전송의 성능을 개선합니다. 데이터는 사용자 버퍼로 직접 복사됩니다. 클라이언트의 커널에서는 캐싱이 수행되지 않습니다. 이 옵션은 기본적으로 해제됩니다.

이전에는 모든 쓰기 요청이 NFS 클라이언트와 NFS 서버에 의해 일련화되었습니다. NFS 클라이언트는 응용 프로그램이 동시 쓰기뿐만 아니라 동시 읽기와 쓰기를 단일 파일에 수행할 수 있도록 수정되었습니다. **forcedirectio** 마운트 옵션을 사용하여 클라이언트에서 이 기능을 사용으로 설정할 수 있습니다. 이 옵션을 사용하면 마운트된 파일 시스템 내에서 모든 파일에 대해 이 기능을 사용할 수 있게 됩니다. 또한 **directio()** 인터페이스를 사용하여 클라이언트의 단일 파일에서 이 기능을 사용 가능하게 할 수 있었습니다. 이 기능을 사용으로 설정하지 않으면 파일에 대한 쓰기가 일련화됩니다. 또한 동시 쓰거나 동시 읽기 및 쓰기가 발생하는 경우 해당 파일에 대해 더 이상 POSIX 의미가 지원되지 않습니다.

이 옵션을 사용하는 방법의 예제는 [151 페이지 “mount 명령 사용”](#)을 참조하십시오.

**largefiles**

이 옵션을 사용하면 2GB보다 큰 파일에 액세스할 수 있습니다. 큰 파일에 액세스할 수 있는지 여부는 서버에서만 제어할 수 있으므로 NFS 버전 3 마운트에서는 이 옵션이 자동으로 무시됩니다. 기본적으로 모든 UFS 파일 시스템은 **largefiles**를 사용하여 마운트됩니다. NFS 버전 2 프로토콜을 사용하는 마운트의 경우 **largefiles o** 옵션을 사용하면 마운트가 실패하고 오류가 발생합니다.

**nolargefiles**

UFS 마운트에 대해 이 옵션을 사용하면 파일 시스템에 큰 파일이 없도록 할 수 있습니다. **mount\_ufs(1M)** 매뉴얼 페이지를 참조하십시오. 큰 파일의 존재 여부는 NFS 서버에서만 제어할 수 있으므로 NFS 마운트를 사용할 때는 **nolargefiles**에 대한 옵션이 없습니다. 이 옵션을 사용하여 파일 시스템 NFS 마운트를 시도하면 작업이 거부되고 오류가 발생합니다.

**nosuid|suid**

**nosuid** 옵션은 **nosetuid** 옵션과 함께 **nodevices** 옵션을 지정하는 것에 해당합니다. **nodevices** 옵션이 지정된 경우에는 마운트된 파일 시스템에서 장치 특정 파일을 열 수 없습니다. **nosetuid** 옵션이 지정된 경우 파일 시스템에 있는 이진 파일의 **setuid** 비트 및 **setgid** 비트가 무시됩니다. 프로세스는 이진 파일을 실행하는 사용자의 권한으로 실행됩니다.

suid 옵션은 setuid 옵션과 함께 devices 옵션을 지정하는 것에 해당합니다. devices 옵션이 지정된 경우에는 마운트된 파일 시스템에서 장치 특정 파일을 열 수 없습니다. setuid 옵션이 지정된 경우 파일 시스템에 있는 이진 파일의 setuid 비트 및 setgid 비트가 커널에 의해 적용됩니다.

두 옵션이 모두 지정되지 않은 경우의 기본 옵션은 suid입니다. 이 옵션은 setuid 옵션과 함께 devices 옵션을 지정하는 기본 동작을 제공합니다.

아래 표에서는 nosuid 또는 suid를 devices 또는 nodevices와 setuid 또는 nosetuid와 결합하는 경우의 효과에 대해 설명합니다. 각 옵션을 결합할 때는 가장 제한적인 옵션에 따라 동작이 결정됩니다.

결합된 옵션의 동작	옵션	옵션	옵션
nosetuid와 nodevices를 결합한 경우에 해당하는 옵션	nosuid	nosetuid	nodevices
nosetuid와 nodevices를 결합한 경우에 해당하는 옵션	nosuid	nosetuid	devices
nosetuid와 nodevices를 결합한 경우에 해당하는 옵션	nosuid	setuid	nodevices
nosetuid와 nodevices를 결합한 경우에 해당하는 옵션	nosuid	setuid	devices
nosetuid와 nodevices를 결합한 경우에 해당하는 옵션	suid	nosetuid	nodevices
nosetuid와 devices를 결합한 경우에 해당하는 옵션	suid	nosetuid	devices
setuid와 nodevices를 결합한 경우에 해당하는 옵션	suid	setuid	nodevices
setuid와 devices를 결합한 경우에 해당하는 옵션	suid	setuid	devices

nosuid 옵션은 잠재적으로 신뢰되지 않는 서버에 액세스하는 NFS 클라이언트에 대해 추가적인 보안을 제공합니다. 이 옵션을 사용하여 원격 파일 시스템을 마운트하면

신뢰되지 않는 장치 또는 `setuid` 이진 파일 가져오기를 통한 권한 승격 가능성이 줄어듭니다. 모든 Oracle Solaris 파일 시스템에서 이러한 옵션을 모두 사용할 수 있습니다.

#### public

이 옵션은 NFS 서버에 연결할 때 공용 파일 핸들을 사용하도록 강제 지정합니다. 서버에서 공용 파일 핸들을 지원하는 경우 MOUNT 프로토콜이 사용되지 않으므로 마운트 작업 속도가 빨라집니다. 또한 MOUNT 프로토콜이 사용되지 않으므로 공용 옵션을 사용하여 방화벽을 통해 마운트를 수행할 수 있습니다.

#### rw|ro

`-rw` 및 `-ro` 옵션은 파일 시스템을 읽기/쓰기로 마운트할지 읽기 전용으로 마운트할지를 나타냅니다. 기본값은 읽기/쓰기입니다. 이 옵션은 원격 홈 디렉토리, 메일 스프링 디렉토리 또는 사용자가 변경해야 하는 기타 파일 시스템에 적합합니다. 읽기 전용 옵션은 사용자가 변경해서는 안 되는 디렉토리에 적합합니다. 예를 들어 매뉴얼 페이지 공유 복사본은 사용자가 쓸 수 없어야 합니다.

#### sec=mode

이 옵션을 사용하여 마운트 트랜잭션 중에 사용할 인증 방식을 지정할 수 있습니다. `mode`의 값은 다음 중 하나일 수 있습니다.

- Kerberos 버전 5 인증 서비스의 경우 `krb5`를 사용합니다.
- Kerberos 버전 5(무결성 포함)의 경우 `krb5i`를 사용합니다.
- Kerberos 버전 5(프라이버시 포함)의 경우 `krb5p`를 사용합니다.
- 인증을 사용하지 않으려면 `none`을 사용합니다.
- Diffie-Hellman(DH) 인증의 경우 `dh`를 사용합니다.
- 표준 UNIX 인증의 경우 `sys`를 사용합니다.

`/etc/nfssec.conf`에서도 모드가 정의됩니다.

#### soft|hard

`soft` 옵션을 사용하여 NFS 파일 시스템을 마운트한 경우 서버가 응답하지 않으면 오류가 반환됩니다. `hard` 옵션을 사용하면 서버가 응답할 때까지 마운트를 계속 재시도합니다. 기본값은 `hard`로, 대부분의 파일 시스템에서는 이 옵션을 사용해야 합니다. 응용 프로그램에서는 `soft` 옵션을 사용하여 마운트된 파일 시스템의 반환 값을 확인하지 않는 경우가 많아 오류가 발생하거나 파일이 손상될 수 있습니다. 응용 프로그램이 반환 값을 확인하는 경우에도 `soft` 옵션을 사용하면 경로 지정 문제와 기타 상황으로 인해 응용 프로그램 사용 시 혼란을 초래하거나 파일이 손상될 수 있습니다. 대부분의 경우에는 `soft` 옵션을 사용해서는 안 됩니다. `hard` 옵션을 사용하여 마운트한 파일 시스템을 사용할 수 없게 되면 파일 시스템을 다시 사용할 수 있을 때까지 해당 파일 시스템을 사용하는 응용 프로그램이 정지됩니다.

## mount 명령 사용

다음 예제를 참조하십시오.

- NFS 버전 2 또는 버전 3에서는 두 명령이 모두 `bee` 서버에서 NFS 파일 시스템을 읽기 전용으로 마운트합니다.



```
# mount -F nfs -r bee:/export/share/man /usr/man
```

```
# mount -F nfs -o ro bee:/export/share/man /usr/man
```

NFS 버전 4에서는 다음 명령줄이 동일한 마운트를 수행합니다.

```
# mount -F nfs -o vers=4 -r bee:/export/share/man /usr/man
```

- NFS 버전 2 또는 3에서 이 명령은 -o 옵션을 사용하여 /usr/man이 이미 마운트되었어도 bee 서버의 매뉴얼 페이지가 로컬 시스템에 마운트되도록 강제 지정합니다. 다음을 참조하십시오.

```
# mount -F nfs -O bee:/export/share/man /usr/man
```

NFS 버전 4에서는 다음 명령줄이 동일한 마운트를 수행합니다.

```
# mount -F nfs -o vers=4 -O bee:/export/share/man /usr/man
```

- NFS 버전 2 또는 버전 3에서 이 명령은 클라이언트 페일오버를 사용합니다.

```
# mount -F nfs -r bee,waspp:/export/share/man /usr/man
```

NFS 버전 4에서는 다음 명령줄에서 클라이언트 페일오버를 사용합니다.

```
# mount -F nfs -o vers=4 -r bee,waspp:/export/share/man /usr/man
```

---

주 - 명령줄에서 사용하는 경우 나열된 서버는 동일한 NFS 프로토콜 버전을 지원해야 합니다. 명령줄에서 mount를 실행할 때는 버전 2 및 버전 3 서버를 모두 사용해서는 안 됩니다. autofs에서는 두 서버를 모두 사용할 수 있습니다. autofs는 가장 적절한 일부 버전 2 또는 버전 3 서버를 자동으로 선택합니다.

---

- 아래에는 NFS 버전 2 또는 버전 3에서 mount 명령과 함께 NFS URL을 사용하는 예제가 나와 있습니다.

```
# mount -F nfs nfs://bee//export/share/man /usr/man
```

다음은 NFS 버전 4에서 mount 명령과 함께 NFS URL을 사용하는 예제입니다.

```
# mount -F nfs -o vers=4 nfs://bee//export/share/man /usr/man
```

- 클라이언트가 파일에 대한 동시 쓰기 및 동시 읽기 및 쓰기를 모두 허용하도록 하려면 forcedirectio 마운트 옵션을 사용합니다. 아래에 예가 나와 있습니다.

```
# mount -F nfs -o forcedirectio bee:/home/somebody /mnt
```

이 예에서 명령은 bee 서버에서 NFS 파일 시스템을 마운트하고 /mnt 디렉토리의 각 파일에 대해 동시 읽기와 쓰기가 가능하도록 설정합니다. 동시 읽기 및 쓰기 지원을 사용으로 설정하면 다음이 수행됩니다.

- 클라이언트는 응용 프로그램의 파일에 대한 병렬 쓰기를 허용합니다.
- 캐시는 클라이언트에서 사용 안함으로 설정됩니다. 따라서 읽기 및 쓰기의 데이터가 서버에 유지됩니다. 구체적으로는, 읽거나 쓰는 데이터를 클라이언트가 캐시하지 않으므로 응용 프로그램에서 이미 캐시하지 않은 모든 데이터는



서버에서 읽게 됩니다. 클라이언트의 운영 체제에는 이 데이터의 복사본이 없습니다. 일반적으로 NFS 클라이언트는 응용 프로그램에서 사용하도록 커널에서 데이터를 캐시합니다.

클라이언트에서는 캐시가 사용 안함으로 설정되므로 먼저 읽기 및 나중에 쓰기 프로세스도 사용 안함으로 설정됩니다. 응용 프로그램이 다음 번에 요청할 데이터를 커널에서 예상하면 먼저 읽기 프로세스가 수행됩니다. 그런 후에 커널은 해당 데이터를 미리 수집하는 프로세스를 시작합니다. 이를 통해 커널은 응용 프로그램에서 데이터를 요청하기 전에 해당 데이터를 준비하려고 합니다.

클라이언트는 나중에 읽기 프로세스를 사용하여 쓰기 처리 능력을 높입니다. 이 경우 응용 프로그램이 파일에 데이터를 쓸 때마다 I/O 작업을 즉시 시작하는 대신 데이터가 메모리에서 캐시됩니다. 그런 후 나중에 데이터를 디스크에 씁니다.

나중에 쓰기 프로세스를 수행하는 경우 데이터를 큰 청크로 쓰거나 응용 프로그램과 비동기화된 상태로 쓸 가능성이 있습니다. 일반적으로 큰 청크를 사용하는 경우 처리 능력이 높아집니다. 비동기 쓰기에서는 응용 프로그램 처리와 I/O 처리 간의 겹침이 허용됩니다. 또한 비동기 쓰기에서는 저장소 부족 시스템에서 보다 효율적인 I/O 시퀀스를 제공하여 I/O를 최적화할 수 있습니다. 동기 쓰기에서는 저장소 부족 시스템에서 특정 I/O 시퀀스가 강제 적용되므로 시스템이 최적 상태가 아닐 수 있습니다.

- 응용 프로그램이 캐시되지 않은 데이터의 의미를 처리할 준비가 되어 있지 않으면 성능이 심각하게 저하될 수 있습니다. 다중 스레드 응용 프로그램에서는 이 문제가 발생하지 않습니다.

---

주 - 동기 쓰기 지원을 사용으로 설정하지 않으면 모든 쓰기 요청이 일련화됩니다. 요청이 일련화되면 다음과 같은 상황이 발생합니다. 쓰기 요청이 진행 중이면 두번째 쓰기 요청은 첫번째 쓰기 요청이 완료될 때까지 기다린 후에 진행됩니다.

---

- 클라이언트에 마운트된 파일 시스템을 표시하려면 인수를 포함하지 않고 `mount` 명령을 사용합니다. 다음을 참조하십시오.

```
% mount
/ on /dev/dsk/c0t3d0s0 read/write/setuid on Wed Apr 7 13:20:47 2004
/usr on /dev/dsk/c0t3d0s6 read/write/setuid on Wed Apr 7 13:20:47 20041995
/proc on /proc read/write/setuid on Wed Apr 7 13:20:47 2004
/dev/fd on fd read/write/setuid on Wed Apr 7 13:20:47 2004
/tmp on swap read/write on Wed Apr 7 13:20:51 2004
/opt on /dev/dsk/c0t3d0s5 setuid/read/write on Wed Apr 7 13:20:51 20041995
/home/kathys on bee:/export/home/bee7/kathys
intr/noquota/nosuid/remote on Wed Apr 24 13:22:13 2004
```

## umount 명령

이 명령을 사용하면 현재 마운트된 원격 파일 시스템을 제거할 수 있습니다. `umount` 명령은 테스트 허용을 위해 `-v` 옵션을 지원합니다. `-a` 옵션을 사용하여 여러 파일 시스템을 한 번에 마운트 해제할 수도 있습니다. `mount-points`가 `-a` 옵션과 함께 포함되어

있으면 해당 파일 시스템이 마운트 해제됩니다. 마운트 지점이 포함되어 있지 않으면 `/`, `/usr`, `/var`, `/proc`, `/dev/fd`, `/tmp` 등의 “필수” 파일 시스템을 제외하고 `/etc/mnttab`에 나열되어 있는 모든 파일 시스템을 마운트 해제하려고 시도합니다. 파일 시스템은 이미 마운트되어 있으며 `/etc/mnttab`에 해당 항목이 있으므로, 파일 시스템 유형에 대한 플래그를 포함할 필요가 없습니다.

`-f` 옵션은 사용 중인 파일 시스템을 마운트 해제하도록 강제 지정합니다. 이 옵션을 사용하여 마운트할 수 없는 파일 시스템 마운트를 시도하는 동안 정지된 클라이언트의 정지를 해제할 수 있습니다.



**주의** - 파일을 쓰는 중인 경우 파일 시스템을 강제로 마운트 해제하면 데이터가 손실될 수 있습니다.

다음 예를 참조하십시오.

**예 6-1** 파일 시스템 마운트 해제

이 예제에서는 `/usr/man`에 마운트되어 있는 파일 시스템 마운트를 해제합니다.

```
# umount /usr/man
```

**예 6-2** `umount`에서 옵션 사용

이 예에서는 `umount -a -V` 실행 결과를 보여줍니다.

```
# umount -a -V
umount /home/kathys
umount /opt
umount /home
umount /net
```

이 명령은 실제로 파일 시스템 마운트를 해제하지는 않습니다.

## mountall 명령

파일 시스템 표에 나열되어 있는 특정 파일 시스템 그룹이나 모든 파일 시스템을 마운트하려면 이 명령을 사용합니다. 이 명령을 통해 다음 작업을 수행할 수 있습니다.

- `-F FSType` 옵션을 사용하여 액세스할 파일 시스템 유형 선택
- `-r` 옵션을 사용하여 파일 시스템 표에 나열된 모든 원격 파일 시스템 선택
- `-l` 옵션을 사용하여 모든 로컬 파일 시스템 선택

NFS 파일 시스템 유형으로 레이블이 지정된 모든 파일 시스템은 원격 파일 시스템이므로, 이러한 일부 옵션 중 일부는 중복됩니다. 자세한 내용은 [mountall\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

다음의 두 사용자 입력 예는 동일합니다.

```
# mountall -F nfs
```

```
# mountall -F nfs -r
```

## umountall 명령

파일 시스템 그룹 마운트를 해제하려면 이 명령을 사용합니다. **-k** 옵션은 **fuser -k mount-point** 명령을 실행하여 *mount-point*와 연관된 모든 프로세스를 강제 종료합니다. **-s** 옵션은 마운트 해제가 명령으로 수행되지 않음을 나타냅니다. **-l**은 로컬 파일 시스템만 사용하도록 지정하고 **-r**은 원격 파일 시스템만 사용하도록 지정합니다. **-h host** 옵션은 명명된 호스트의 모든 파일 시스템을 마운트 해제해야 함을 나타냅니다. **-h** 옵션을 **-l** 또는 **-r**과 결합하여 사용할 수는 없습니다.

원격 호스트에서 마운트된 모든 파일 시스템을 마운트 해제하는 예제는 다음과 같습니다.

```
# umountall -r
```

bee 서버에서 현재 마운트되어 있는 모든 파일 시스템을 마운트 해제하는 예제는 다음과 같습니다.

```
# umountall -h bee
```

## sharectl 명령

이 릴리스에는 NFS 등의 파일 공유 프로토콜을 구성 및 관리할 수 있는 관리 도구인 **sharectl** 유틸리티가 포함되어 있습니다. 이 명령을 사용하여 다음을 수행할 수 있습니다.

- 클라이언트 및 서버 작동 등록 정보 설정
- 특정 프로토콜의 등록 정보 값 표시
- 프로토콜 상태 가져오기

**sharectl** 유틸리티는 다음 구문을 사용합니다.

```
# sharectl subcommand [option] [protocol]
```

**sharectl** 유틸리티는 다음 하위 명령을 지원합니다.

표 6-2 sharectl 유틸리티의 하위 명령

하위 명령	설명
set	파일 공유 프로토콜의 등록 정보를 정의합니다. 등록 정보 및 등록 정보 값 목록은 <a href="#">nfs(4)</a> 매뉴얼 페이지에 설명되어 있는 매개변수를 참조하십시오.

표 6-2 sharectl 유틸리티의 하위 명령 (계속)

하위 명령	설명
get	지정된 프로토콜의 등록 정보 및 등록 정보 값을 표시합니다.
status	지정된 프로토콜이 사용으로 설정되어 있는지 여부를 표시합니다. 프로토콜이 지정되어 있지 않으면 모든 파일 공유 프로토콜의 상태가 표시됩니다.

sharectl 유틸리티에 대한 자세한 내용은 다음을 참조하십시오.

- sharectl(1M) 매뉴얼 페이지
- 156 페이지 “set 하위 명령”
- 156 페이지 “get 하위 명령”
- 157 페이지 “status 하위 명령”

## set 하위 명령

set 하위 명령은 파일 공유 프로토콜의 등록 정보를 정의하며 다음 옵션을 지원합니다.

- h    온라인 도움말 설명을 제공합니다.
- p    프로토콜의 등록 정보를 정의합니다.

set 하위 명령은 다음 구문을 사용합니다.

```
# sharectl set [-h] [-p property=value] protocol
```

주 - 다음을 참조하십시오.

- set 하위 명령을 사용하려면 root 권한이 있어야 합니다.
- 각 추가 등록 정보 값에 대해 이 명령줄 구문을 반복할 필요는 없습니다. -p 옵션을 여러 번 사용하여 같은 명령줄에서 여러 등록 정보를 정의할 수 있습니다.

다음 예제에서는 클라이언트의 최소 NFS 프로토콜 버전을 3으로 설정합니다.

```
# sharectl set -p nfs_client_versmin=3 nfs
```

## get 하위 명령

get 하위 명령은 지정된 프로토콜의 등록 정보 및 등록 정보 값을 표시하며 다음 옵션을 지원합니다.

- h    온라인 도움말 설명을 제공합니다.
- p    지정된 등록 정보의 등록 정보 값을 식별합니다. -p 옵션을 사용하지 않으면 모든 등록 정보 값이 표시됩니다.

get 하위 명령은 다음 구문을 사용합니다.

```
# sharectl get [-h] [-p property] protocol
```

---

주 -get 하위 명령을 사용하려면 root 권한이 있어야 합니다.

---

다음 예에서는 동시 NFS 요청의 최대 수를 지정할 수 있는 등록 정보인 **servers**를 사용합니다.

```
# sharectl get -p servers nfs
servers=1024
```

다음 예에서는 -p 옵션을 사용하지 않았으므로 모든 등록 정보 값이 표시됩니다.

```
# sharectl get nfs
servers=1024
listen_backlog=32
protocol=ALL
servers=32
lockd_listen_backlog=32
lockd_servers=20
lockd_retransmit_timeout=5
grace_period=90
nfsmapid_domain=company.com
server_versmin=2
server_versmax=4
client_versmin=2
client_versmax=4
server_delegation=on
max_connections=-1
device=
```

## status 하위 명령

status 하위 명령은 지정된 프로토콜이 사용으로 설정되어 있는지 여부를 표시하며 다음 옵션을 지원합니다.

-h     온라인 도움말 설명을 제공합니다.

status 하위 명령은 다음 구문을 사용합니다.

```
# sharectl status [-h] [protocol]
```

다음 예에서는 NFS 프로토콜의 상태를 보여줍니다.

```
# sharectl status nfs
nfs            enabled
```

## share 명령

이 명령을 사용하면 NFS 서버의 로컬 파일 시스템을 마운트 가능하도록 지정할 수 있습니다. `share` 명령을 사용하여 시스템에서 현재 공유 중인 파일 시스템 목록을 표시할 수도 있습니다. NFS 서버가 실행 중이어야 `share` 명령이 작동합니다.

공유할 수 있는 개체에는 모든 디렉토리 트리가 포함됩니다. 그러나 각 파일 시스템 계층은 파일 시스템이 있는 파티션이나 디스크 슬라이스에 의해 제한됩니다.

이미 공유 중인 더 큰 파일 시스템의 일부분인 파일 시스템은 공유할 수 없습니다. 예를 들어 `/usr` 및 `/usr/local`이 한 디스크 슬라이스에 있는 경우 `/usr`를 공유하거나 `/usr/local`을 공유할 수 있습니다. 그러나 서로 다른 공유 옵션을 사용하여 두 디렉토리를 모두 공유해야 하는 경우에는 `/usr/local`을 별도의 디스크 슬라이스로 이동해야 합니다.

읽기/쓰기 공유된 파일 시스템의 파일 핸들을 통해 읽기 전용 공유인 파일 시스템에 액세스할 수 있습니다. 그러나 이 경우 두 파일 시스템이 같은 디스크 슬라이스에 있어야 합니다. 이보다 안전한 방법을 사용할 수 있습니다. 즉, 읽기/쓰기로 지정해야 하는 파일 시스템을 읽기 전용으로 공유해야 하는 파일 시스템과 다른 별도의 파티션이나 디스크 슬라이스에 배치합니다.

---

주 - 파일 시스템 공유를 해제했다가 다시 공유할 때 NFS 버전 4가 작동하는 방식에 대한 자세한 내용은 [173 페이지 “NFS 버전 4에서 파일 시스템 공유 해제 및 다시 공유”](#)를 참조하십시오.

---

## 파일 시스템과 관련이 없는 share 옵션

-o 플래그와 함께 포함할 수 있는 몇 가지 옵션은 다음과 같습니다.

`rw|ro`

*pathname* 파일 시스템은 모든 클라이언트에 대해 읽기/쓰기 또는 읽기 전용으로 공유됩니다.

`rw=accesslist`

파일 시스템이 나열된 클라이언트에 한해 읽기/쓰기로 공유됩니다. 다른 모든 요청은 거부됩니다. Solaris 2.6 릴리스부터는 `accesslist`에서 정의되는 클라이언트 목록이 확장되었습니다. 자세한 내용은 [161 페이지 “share 명령을 사용하여 액세스 목록 설정”](#)을 참조하십시오. 이 옵션을 사용하여 -ro 옵션을 대체할 수 있습니다.

## NFS 관련 share 옵션

NFS 파일 시스템에서 사용 가능한 옵션은 다음과 같습니다.

**aclok**

이 옵션을 사용하면 NFS 버전 2 프로토콜을 지원하는 NFS 서버에서 NFS 버전 2 클라이언트의 액세스를 제어하도록 구성할 수 있습니다. 이 옵션을 사용하지 않으면 모든 클라이언트에 최소한의 액세스 권한이 제공됩니다. 이 옵션을 사용하면 클라이언트에 최대한의 액세스 권한이 제공됩니다. 예를 들어 **-aclok** 옵션을 통해 공유되는 파일 시스템에서는 특정 사용자에게 읽기 권한이 있으면 모든 사용자에게 읽기 권한이 있는 것입니다. 그러나 이 옵션을 사용하지 않는 경우에는 액세스 권한을 가져야 하는 클라이언트에 대한 액세스를 거부할 수 있습니다. 이미 사용되고 있는 보안 시스템에 따라 허용할 액세스 권한 레벨을 결정합니다. 액세스 제어 목록(ACL)에 대한 자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “액세스 제어 목록을 사용하여 UFS 파일 보호”**를 참조하십시오.

---

**주** - ACL을 사용하려면 클라이언트와 서버에서 NFS 버전 3 및 NFS\_ACL 프로토콜을 지원하는 소프트웨어를 실행하는지 확인합니다. 소프트웨어에서 NFS 버전 3 프로토콜만 지원하는 경우에는 클라이언트가 올바른 액세스 권한을 얻지만 ACL을 조작할 수는 없습니다. 소프트웨어에서 NFS\_ACL 프로토콜을 지원하는 경우 클라이언트는 올바른 액세스 권한을 얻으며 ACL을 조작할 수 있습니다.

---

**anon=uid**

**uid**를 사용하여 인증되지 않은 사용자의 사용자 ID를 선택합니다. **uid**를 -1로 설정하면 서버에서 인증되지 않은 사용자의 액세스를 거부합니다. **anon=0**을 설정하여 루트 액세스 권한을 부여할 수는 있지만, 이 옵션을 사용하면 인증되지 않은 사용자도 루트 액세스 권한을 가지게 되므로 대신 **root** 옵션을 사용하십시오.

**index=filename**

사용자가 NFS URL에 액세스하면 **-index=filename** 옵션은 디렉토리 목록을 표시하는 대신 HTML 파일을 강제로 로드합니다. 이 옵션은 HTTP URL이 액세스하는 디렉토리에서 **index.html** 파일을 찾으면 현재 브라우저의 동작을 모방합니다. 이 옵션은 **httpd**에 대해 **DirectoryIndex** 옵션을 설정하는 것과 동등합니다. 예를 들어 **dfstab** 파일 항목이 다음과 같다고 가정하겠습니다.

```
share -F nfs -o ro,public,index=index.html /export/web
```

이러한 URL은 같은 정보를 표시합니다.

```
nfs://<server>/<dir>
nfs://<server>/<dir>/index.html
nfs://<server>/export/web/<dir>
nfs://<server>/export/web/<dir>/index.html
http://<server>/<dir>
http://<server>/<dir>/index.html
```

**log=tag**

이 옵션은 **/etc/nfs/nfslog.conf**에서 파일 시스템에 대한 NFS 서버 로깅 구성 정보가 포함된 태그를 지정합니다. NFS 서버 로깅을 사용으로 설정하려면 이 옵션을 선택해야 합니다.

**nosuid**

이 옵션은 **setuid** 또는 **setgid** 모드를 사용으로 설정하려는 모든 시도를 무시해야 함을 나타냅니다. NFS 클라이언트는 **setuid** 또는 **setgid** 비트가 설정되어 있으면 파일을 만들 수 없습니다.

**public**

WebNFS 찾아보기를 수행할 수 있도록 **-public** 옵션이 **share** 명령에 추가되었습니다. 이 옵션을 사용하는 경우 서버의 파일 시스템 하나만 공유할 수 있습니다.

**root=accesslist**

서버에서 목록의 호스트에 대한 루트 액세스 권한을 부여합니다. 기본적으로 서버에서는 원격 호스트에 대한 루트 액세스 권한을 부여하지 않습니다. 선택한 보안 모드가 **-sec=sys** 이외의 모드인 경우에는 **accesslist**에 클라이언트 호스트 이름만 포함할 수 있습니다. Solaris 2.6 릴리스부터는 **accesslist**에서 정의되는 클라이언트 목록이 확장되었습니다. 자세한 내용은 [161 페이지 “share 명령을 사용하여 액세스 목록 설정”](#)을 참조하십시오.



**주의** - 다른 호스트에 대한 루트 액세스 권한을 부여하는 것은 보안상 다양한 사항을 의미합니다. **-root=** 옵션을 사용할 때는 주의해야 합니다.

**root=client-name**

**client-name** 값은 **exportfs(1B)**에서 제공되는 주소 목록에 대해 클라이언트 IP 주소를 확인하기 위해 **AUTH\_SYS** 인증에서 사용됩니다. 일치하는 항목이 있으면 공유 중인 파일 시스템에 대한 **root** 액세스 권한이 부여됩니다.

**root=host-name**

**AUTH\_SYS** 또는 **RPCSEC\_GSS**와 같은 보안 NFS 모드에서는 서버가 액세스 목록에서 파생되는 호스트 기반 기본 이름 목록에 대해 클라이언트의 기본 이름을 확인합니다. 클라이언트 기본 이름의 일반 구문은 **root@hostname**입니다. Kerberos V의 경우 구문은 **root/hostname.fully.qualified@REALM**입니다. **host-name** 값을 사용하는 경우 액세스 목록의 클라이언트에 기본 이름에 대한 자격 증명이 있어야 합니다. Kerberos V의 경우 클라이언트에는 해당 **root/hostname.fully.qualified@REALM** 기본 이름에 대한 유효한 **keytab** 항목이 있어야 합니다. 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “Kerberos 클라이언트 구성”](#)을 참조하십시오.

**sec=mode[:mode]**

**mode**는 파일 시스템 액세스 권한을 얻는데 필요한 보안 모드를 선택합니다. 기본 보안 모드는 **UNIX** 인증입니다. 여러 모드를 지정할 수는 있지만 명령줄당 각 보안 모드를 한 번씩만 사용해야 합니다. 각 **-mode** 옵션은 다른 **-mode**가 발견될 때까지 모든 후속 **-rw**, **-ro**, **-rw=**, **-ro=**, **-root=** 및 **-window=** 옵션에 적용됩니다. **-sec=none**을 사용하는 경우 모든 사용자가 **nobody** 사용자에게 매핑됩니다.

**window=value**

**value**는 NFS 서버에서 자격 증명의 최대 수명(초)을 선택합니다. 기본 값은 30000초(8.3시간)입니다.



## share 명령을 사용하여 액세스 목록 설정

*accesslist*에는 도메인 이름, 서브넷 번호 또는 액세스를 거부할 항목은 물론 표준 `-ro=`, `-rw=` 또는 `-root=` 옵션도 포함할 수 있습니다. 이러한 확장을 사용하면 긴 클라이언트 목록을 유지 관리하거나 네임스페이스를 변경할 필요 없이 단일 서버에서 파일 액세스를 간편하게 제어할 수 있습니다.

이 명령은 대부분의 시스템에서는 읽기 전용 액세스 권한을 제공하지만 `rose` 및 `lilac`에 대해서는 읽기/쓰기 액세스 권한을 허용합니다.

```
# share -F nfs -o ro,rw=rose:lilac /usr/src
```

다음 예제에서는 `eng` 넷 그룹의 모든 호스트에 대한 읽기 전용 액세스 권한이 지정됩니다. `rose` 클라이언트에만 읽기/쓰기 액세스 권한이 부여됩니다.

```
# share -F nfs -o ro=eng,rw=rose /usr/src
```

---

**주** - 인수를 사용하지 않고는 `rw` 및 `ro`를 모두 지정할 수 없습니다. 읽기/쓰기 옵션이 지정되어 있지 않으면 모든 클라이언트에 대해 기본적으로 읽기/쓰기가 사용됩니다.

---

하나의 파일 시스템을 여러 클라이언트와 공유하려면 같은 행에 모든 옵션을 입력해야 합니다. 같은 개체에 대해 `share` 명령을 여러 번 호출해도 마지막으로 실행한 명령만 “저장”됩니다. 이 명령을 사용하면 3개 클라이언트 시스템에 읽기/쓰기 권한으로 액세스할 수 있지만, `rose` 및 `tulip`에만 루트 파일 시스템 액세스 권한이 부여됩니다.

```
# share -F nfs -o rw=rose:lilac:tulip,root=rose:tulip /usr/src
```

여러 인증 방식을 사용하는 파일 시스템을 공유할 때는 올바른 보안 모드 뒤에 `-ro`, `-rw=`, `-rw`, `-root` 및 `-window` 옵션을 포함하십시오. 이 예에서는 이름이 `eng`인 넷 그룹의 모든 호스트에 대해 UNIX 인증을 선택합니다. 이러한 호스트는 읽기 전용 모드에서만 파일 시스템을 마운트할 수 있습니다. `tulip` 및 `lilac` 호스트는 Diffie-Hellman 인증을 사용하는 경우 파일 시스템을 읽기/쓰기로 마운트할 수 있습니다. 이러한 옵션을 사용하는 경우 `tulip` 및 `lilac`은 DH 인증을 사용하지 않아도 파일 시스템을 읽기 전용으로 마운트할 수 있습니다. 그러나 이 경우에는 `eng` 넷 그룹에 호스트 이름이 포함되어 있어야 합니다.

```
# share -F nfs -o sec=dh,rw=tulip:lilac,sec=sys,ro=eng /usr/src
```

UNIX 인증이 기본 보안 모드이기는 하지만 `-sec` 옵션을 사용하는 경우에는 UNIX 인증이 포함되지 않습니다. 따라서 UNIX 인증을 다른 인증 방식과 함께 사용하려는 경우에는 `-sec=sys` 옵션을 포함해야 합니다.

실제 도메인 이름과 마침표를 앞에 붙여서 액세스 목록에서 DNS 도메인 이름을 사용할 수 있습니다. 마침표 뒤의 문자열은 정규화된 호스트 이름이 아닌 도메인 이름입니다. 다음 항목에서는 `eng.example.com` 도메인의 모든 호스트에 대한 마운트 액세스를 허용합니다.

```
# share -F nfs -o ro=.:eng.example.com /export/share/man
```

이 예제에서는 단일 “.”가 NIS 네임스페이스를 통해 일치되는 모든 호스트와 일치합니다. 이러한 이름 서비스에서 반환되는 결과에는 도메인 이름이 포함되지 않습니다. “.eng.example.com” 항목은 네임스페이스 확인을 위해 DNS를 사용하는 모든 호스트와 일치합니다. DNS는 항상 정규화된 호스트 이름을 반환합니다. 따라서 DNS와 다른 네임스페이스를 결합하여 사용하는 경우에는 더 긴 항목이 필요합니다.

실제 네트워크 번호 또는 네트워크 이름과 “@”를 앞에 붙여 액세스 목록에서 서브넷 번호를 사용할 수 있습니다. 이 문자는 네트워크 이름을 넷 그룹이나 정규화된 호스트 이름과 구분합니다. 서브넷은 /etc/networks 또는 NIS 네임스페이스에서 식별해야 합니다. 다음 항목을 사용하는 경우 192.168 서브넷을 eng 네트워크로 식별한 경우와 같은 효과를 얻을 수 있습니다.

```
# share -F nfs -o ro=@eng /export/share/man
# share -F nfs -o ro=@192.168 /export/share/man
# share -F nfs -o ro=@192.168.0.0 /export/share/man
```

마지막 두 항목은 전체 네트워크 주소를 포함하지 않아도 됨을 보여줍니다.

CIDR(Classless Inter-Domain Routing)에서와 같이 네트워크 접두어가 바이트 맞춤되어 있지 않으면 명령줄에서 마스크 길이를 명시적으로 지정할 수 있습니다. 마스크 길이는 주소 접두어에서 네트워크 이름이나 네트워크 번호(슬래시 포함) 및 중요한 비트 번호를 따라 정의됩니다. 예를 들면 다음과 같습니다.

```
# share -f nfs -o ro=@eng/17 /export/share/man
# share -F nfs -o ro=@192.168.0/17 /export/share/man
```

이 예제에서 “/17”은 주소의 첫 17비트가 마스크로 사용됨을 나타냅니다. CIDR에 대한 추가 정보는 RFC 1519를 참조하십시오.

항목 앞에 “-”를 배치하여 음수 액세스를 선택할 수도 있습니다. 항목은 왼쪽부터 오른쪽으로 읽습니다. 따라서 음수 액세스 항목이 적용되는 항목 앞에 음수 액세스 항목을 배치해야 합니다.

```
# share -F nfs -o ro=-rose:eng.example.com /export/share/man
```

이 예제에서는 이름이 rose인 호스트를 제외하고 eng.example.com 도메인의 모든 호스트에 대한 액세스를 허용합니다.

## unshare 명령

이 명령을 사용하면 이전에 사용 가능했던 파일 시스템을 클라이언트가 마운트할 수 없도록 지정할 수 있습니다. NFS 파일 시스템 공유를 해제하면 기존 마운트를 사용한 클라이언트로부터의 액세스가 금지됩니다. 파일 시스템을 클라이언트에 마운트할 수는 있지만 파일에는 액세스할 수 없습니다. unshare 명령은 -t 옵션을 사용하여 파일 시스템 공유를 일시적으로 해제하는 경우가 아니면 공유를 영구적으로 삭제합니다.

주 - 파일 시스템 공유를 해제했다가 다시 공유할 때 NFS 버전 4가 작동하는 방식에 대한 자세한 내용은 173 페이지 “NFS 버전 4에서 파일 시스템 공유 해제 및 다시 공유”를 참조하십시오.

다음은 특정 파일 시스템의 공유를 해제하는 예입니다.

```
# unshare /usr/src
```

## shareall 명령

이 명령을 사용하면 여러 파일 시스템을 공유할 수 있습니다. 옵션을 포함하지 않고 사용하는 경우 이 명령은 `/etc/dfs/dfstab`의 모든 항목을 공유합니다. 파일 이름을 포함하여 `share` 명령줄이 나열되는 파일 이름을 지정할 수 있습니다. 파일 이름을 포함하지 않으면 `/etc/dfs/dfstab`를 확인합니다. 파일 이름 대신 “.”를 사용하는 경우에는 표준 입력에서 `share` 명령을 입력할 수 있습니다.

다음은 로컬 파일에 나열되는 모든 파일 시스템을 나열하는 예입니다.

```
# shareall /etc/dfs/special_dfstab
```

## unshareall 명령

이 명령은 현재 공유되는 모든 리소스를 사용할 수 없도록 지정합니다. `-F FSType` 옵션은 `/etc/dfs/fstypes`에 정의되어 있는 파일 시스템 유형 목록을 선택합니다. 이 플래그를 사용하면 공유 해제할 특정 파일 시스템 유형을 선택할 수 있습니다. 기본 파일 시스템 유형은 `/etc/dfs/fstypes`에서 정의됩니다. 특정 파일 시스템을 선택하려면 `unshare` 명령을 사용합니다.

다음은 모든 NFS 유형 파일 시스템을 공유 해제하는 예입니다.

```
# unshareall -F nfs
```

## showmount 명령

이 명령은 다음 중 하나를 표시합니다.

- NFS 서버에서 공유되는 원격으로 마운트된 파일 시스템을 포함하는 모든 클라이언트
- 클라이언트가 마운트하는 파일 시스템 전용
- 클라이언트 액세스 정보를 포함하는 공유되는 파일 시스템

주 - `showmount` 명령은 NFS 버전 2 및 버전 3 내보내기만 표시합니다. 이 명령은 NFS 버전 4 내보내기는 표시하지 않습니다.

명령 구문은 다음과 같습니다.

`showmount [ -ade ] [ hostname ]`

-a 모든 원격 마운트 목록을 인쇄합니다. 각 항목에는 클라이언트 이름과 디렉토리가 포함됩니다.

-d 클라이언트가 원격으로 마운트한 디렉토리 목록을 인쇄합니다.

-e 공유되거나 내보낸 파일 목록을 인쇄합니다.

*hostname* 정보를 수집할 NFS 서버를 선택합니다.

*hostname*을 지정하지 않으면 로컬 호스트를 질의합니다.

다음 명령은 클라이언트 및 클라이언트가 마운트한 모든 로컬 디렉토리를 나열합니다.

```
# showmount -a bee
lilac:/export/share/man
lilac:/usr/src
rose:/usr/src
tulip:/export/share/man
```

다음 명령은 마운트된 디렉토리를 나열합니다.

```
# showmount -d bee
/export/share/man
/usr/src
```

다음 명령은 공유된 파일 시스템을 나열합니다.

```
# showmount -e bee
/usr/src                               (everyone)
/export/share/man                     eng
```

## setmnt 명령

이 명령은 `/etc/mnttab` 테이블을 만듭니다. `mount` 및 `umount` 명령이 테이블을 확인합니다. 일반적으로 이 명령은 시스템 부트 시 자동으로 실행되므로 수동으로 실행할 필요가 없습니다.

## nfsref 명령

nfsref 명령은 NFSv4 참조를 추가, 삭제 또는 나열하는 데 사용됩니다. 명령 구문은 다음과 같습니다.

```
nfsref add path location [ location ... ]
```

```
nfsref remove path
```

```
nfsref lookup path
```

*path*           구문 재분석 지점의 이름을 선택합니다.

*location*       구문 재분석 지점과 연관시킬 NFS 또는 SMB 공유 파일 시스템을 하나 이상 식별합니다.

## NFS 문제 해결용 명령

이러한 명령은 NFS 문제를 해결할 때 유용할 수 있습니다.

## nfsstat 명령

이 명령을 사용하여 NFS 및 RPC 연결에 대한 통계 정보를 수집할 수 있습니다. 명령의 구문은 다음과 같습니다.

```
nfsstat [ -cmnrsz ]
```

-c   클라이언트측 정보를 표시합니다.

-m   NFS 마운트된 각 파일 시스템의 통계를 표시합니다.

-n   NFS 정보가 클라이언트측과 서버측에 모두 표시되도록 지정합니다.

-r   RPC 통계를 표시합니다.

-s   서버측 정보를 표시합니다.

-z   통계가 0으로 설정되도록 지정합니다.

명령줄에서 옵션을 제공하지 않으면 -cnrs 옵션이 사용됩니다.

새로운 소프트웨어 또는 하드웨어를 컴퓨팅 환경에 추가하는 경우 문제 디버깅을 위해 서버측 통계 수집은 중요할 수 있습니다. 이 명령을 최소 매주 한 번 실행하고 수치를 저장하면 이전 성능의 내역을 확인할 수 있습니다.

다음 예제를 참조하십시오.

## # nfsstat -s

## Server rpc:

## Connection oriented:

calls	badcalls	nullrecv	badlen	xdrCALL	dupchecks	dupreqs
719949194	0	0	0	0	58478624	33

## Connectionless:

calls	badcalls	nullrecv	badlen	xdrCALL	dupchecks	dupreqs
73753609	0	0	0	0	987278	7254

## Server NFSv2:

calls	badcalls	referrals	referlinks
25733	0	0	0

## Server NFSv3:

calls	badcalls	referrals	referlinks
132880073	0	0	0

## Server NFSv4:

calls	badcalls	referrals	referlinks
488884996	4	0	0

## Version 2: (746607 calls)

null	getattr	setattr	root	lookup	readlink	read
883 0%	60 0%	45 0%	0 0%	177446 23%	1489 0%	537366 71%
wrCache	write	create	remove	rename	link	symlink
0 0%	1105 0%	47 0%	59 0%	28 0%	10 0%	9 0%
mkdir	rmdir	readdir	statfs			
26 0%	0 0%	27926 3%	108 0%			

## Version 3: (728863853 calls)

null	getattr	setattr	lookup	access
1365467 0%	496667075 68%	8864191 1%	66510206 9%	19131659 2%
readlink	read	write	create	mkdir
414705 0%	80123469 10%	18740690 2%	4135195 0%	327059 0%
symlink	mknod	remove	rmdir	rename
101415 0%	9605 0%	6533288 0%	111810 0%	366267 0%
link	readdir	readdirplus	fsstat	fsinfo
2572965 0%	519346 0%	2726631 0%	13320640 1%	60161 0%
pathconf	commit			
13181 0%	6248828 0%			

## Version 4: (54871870 calls)

null	compound
266963 0%	54604907 99%

## Version 4: (167573814 operations)

reserved	access	close	commit
0 0%	2663957 1%	2692328 1%	1166001 0%
create	delegpurge	delegreturn	getattr
167423 0%	0 0%	1802019 1%	26405254 15%
getfh	link	lock	lockt
11534581 6%	113212 0%	207723 0%	265 0%
locku	lookup	lookupp	nverify
230430 0%	11059722 6%	423514 0%	21386866 12%
open	openattr	open_confirm	open_downgrade
2835459 1%	4138 0%	18959 0%	3106 0%
putfh	putpubfh	putrootfh	read
52606920 31%	0 0%	35776 0%	4325432 2%
readdir	readlink	remove	rename
606651 0%	38043 0%	560797 0%	248990 0%
renew	restorefh	savefh	secinfo

```

2330092 1%      8711358 5%      11639329 6%      19384 0%
setattr        setclientid      setclientid_confirm verify
453126 0%      16349 0%      16356 0%      2484 0%
write          release_lockowner illegal
3247770 1%      0 0%      0 0%

```

Server nfs\_acl:

Version 2: (694979 calls)

```

null      getacl      setacl      setattr      access      getxattrdir
0 0%      42358 6%      0 0%      584553 84%  68068 9%      0 0%

```

Version 3: (2465011 calls)

```

null      getacl      setacl      getxattrdir
0 0%      1293312 52% 1131 0%  1170568 47%

```

위의 목록은 NFS 서버 통계의 예제입니다. 처음 5행은 RPC와 관련되며 나머지 행은 NFS 작업을 보고합니다. 두 통계 세트에서 **badcalls** 또는 **calls**의 평균 수치와 주당 통화 수치를 파악하면 문제를 식별하는 데 도움이 될 수 있습니다. **badcalls** 값은 클라이언트로부터의 잘못된 메시지 수를 보고합니다. 이 값은 네트워크 하드웨어 문제를 나타낼 수 있습니다.

일부 연결은 디스크에서 쓰기 작업을 생성합니다. 이러한 통계 수치가 갑자기 증가하면 문제가 발생했을 수 있으므로 조사해야 합니다. NFS 버전 2 통계의 경우 확인해야 하는 연결은 **setattr**, **write**, **create**, **remove**, **rename**, **link**, **symlink**, **mkdir** 및 **rmdir**입니다. NFS 버전 3 및 버전 4 통계의 경우 확인해야 하는 값은 **commit**입니다. 한 NFS 서버에서 **commit** 레벨이 거의 동일한 다른 서버에 비해 높으면 NFS 클라이언트의 메모리가 충분한지 확인하십시오. 클라이언트에 사용 가능한 리소스가 없으면 서버의 **commit** 작업 수가 증가합니다.

## pstack 명령

이 명령은 각 프로세스에 대한 스택 추적을 표시합니다. **pstack** 명령은 프로세스 소유자가 실행하거나 **루트**에서 실행해야 합니다. **pstack**을 사용하여 프로세스가 정지된 위치를 확인할 수 있습니다. 이 명령에 사용할 수 있는 옵션은 확인할 프로세스의 PID뿐입니다. **proc(1)** 매뉴얼 페이지를 참조하십시오.

다음 예제에서는 실행 중인 **nfsd** 프로세스를 확인합니다.

```

# /usr/bin/pgrep nfsd
243
# /usr/bin/pstack 243
243: /usr/lib/nfs/nfsd -a 16
ef675c04 poll (24d50, 2, ffffffff)
000115dc ??????? (24000, 132c4, 276d8, 1329c, 276d8, 0)
00011390 main (3, effffff14, 0, 0, ffffffff, 400) + 3c8
00010fb0 _start (0, 0, 0, 0, 0, 0) + 5c

```

이 예에서는 프로세스에서 새 연결 요청(정상 응답)을 대기하고 있음을 보여줍니다. 요청 후에도 프로세스가 계속 폴링되는 상태로 스택에 표시되면 프로세스가 정지되었을 수 있습니다. 이 문제를 해결하려면 [120 페이지 “NFS 서비스를 다시 시작하는 방법”](#)의

지침을 따르십시오. 문제가 정지된 프로그램 때문에 발생했는지를 완전하게 확인하려면 [117 페이지 “NFS 문제 해결 절차”](#)의 지침을 검토하십시오.

## rpcinfo 명령

이 명령은 시스템에서 실행 중인 RPC 서비스에 대한 정보를 생성합니다. 이 명령을 사용하여 RPC 서비스를 변경할 수도 있습니다. 이 명령에서는 많은 옵션을 사용할 수 있습니다. [rpcinfo\(1M\)](#) 매뉴얼 페이지를 참조하십시오. 다음은 명령에서 사용할 수 있는 몇 가지 옵션의 간략한 개요입니다.

```
rpcinfo [ -m | -s ] [ hostname ]
```

```
rpcinfo -T transport hostname [ progname ]
```

```
rpcinfo [ -t | -u ] [ hostname ] [ progname ]
```

**-m**            rpcbind 작업의 통계 표를 표시합니다.

**-s**            등록된 모든 RPC 프로그램의 간략한 목록을 표시합니다.

**-T**            특정 전송 또는 프로토콜을 사용하는 서비스에 대한 정보를 표시합니다.

**-t**            TCP를 사용하는 RPC 프로그램을 검사합니다.

**-u**            UDP를 사용하는 RPC 프로그램을 검사합니다.

**transport**    서비스용 전송 또는 프로토콜을 선택합니다.

**hostname**    필요한 정보를 가져올 서버의 호스트 이름을 선택합니다.

**progname**    정보를 수집할 RPC 프로그램을 선택합니다.

*hostname*에 대해 값을 지정하지 않으면 로컬 호스트 이름이 사용됩니다. *progname*을 RPC 프로그램 번호로 대체할 수는 있지만 대부분의 사용자는 번호가 아닌 이름을 기억하기가 쉽습니다. NFS 버전 3 소프트웨어를 실행하지 않는 시스템에서는 **-p** 옵션을 **-s** 옵션 대신 사용할 수 있습니다.

이 명령에 의해 생성되는 데이터에는 다음이 포함됩니다.

- RPC 프로그램 번호
- 특정 프로그램의 버전 번호
- 사용 중인 전송 프로토콜
- RPC 서비스의 이름
- RPC 서비스의 소유자

다음 예에서는 서버에서 실행 중인 RPC 서비스에 대한 정보를 수집합니다. 명령에 의해 생성되는 텍스트는 출력을 보다 쉽게 읽을 수 있도록 **sort** 명령을 통해 필터링됩니다. 예에서는 RPC 서비스가 나열되는 여러 행이 삭제되었습니다.



```
% rpcinfo -s bee |sort -n
program version(s) netid(s) service owner
100000 2,3,4 udp6,tcp6,udp,tcp,ticlts,ticotsord,ticots portmapper superuser
100001 4,3,2 udp6,udp,ticlts rstatd superuser
100003 4,3,2 tcp,udp,tcp6,udp6 nfs 1
100005 3,2,1 ticots,ticotsord,tcp,tcp6,ticlts,udp,udp6 mountd superuser
100007 1,2,3 ticots,ticotsord,ticlts,tcp,udp,tcp6,udp6 ypbind 1
100011 1 udp6,udp,ticlts rquotad superuser
100021 4,3,2,1 tcp,udp,tcp6,udp6 nlockmgr 1
100024 1 ticots,ticotsord,ticlts,tcp,udp,tcp6,udp6 status superuser
100068 5,4,3,2 ticlts - superuser
100083 1 ticotsord - superuser
100133 1 ticots,ticotsord,ticlts,tcp,udp,tcp6,udp6 - superuser
100134 1 ticotsord - superuser
100155 1 ticotsord smsrverd superuser
100169 1 ticots,ticotsord,ticlts - superuser
100227 3,2 tcp,udp,tcp6,udp6 nfs_acl 1
100234 1 ticotsord - superuser
390113 1 tcp - superuser
390435 1 tcp - superuser
390436 1 tcp - superuser
1073741824 1 tcp,tcp6 - 1
```

다음의 두 예에서는 서버에서 특정 전송을 선택하여 특정 RPC 서비스에 대한 정보를 수집하는 방법을 보여줍니다. 첫번째 예제에서는 TCP를 통해 실행되는 mountd 서비스를 확인합니다. 두번째 예제에서는 UDP를 통해 실행되는 NFS 서비스를 확인합니다.

```
% rpcinfo -t bee mountd
program 100005 version 1 ready and waiting
program 100005 version 2 ready and waiting
program 100005 version 3 ready and waiting
% rpcinfo -u bee nfs
program 100003 version 2 ready and waiting
program 100003 version 3 ready and waiting
```

## snoop 명령

이 명령은 네트워크의 패킷을 감시하는 데 사용되는 경우가 많습니다. snoop 명령은 루트로 실행해야 합니다. 이 명령을 사용하면 클라이언트와 서버에서 네트워크 하드웨어가 작동하는지를 효율적으로 확인할 수 있습니다. 다양한 옵션을 사용할 수 있습니다. [snoop\(1M\)](#) 매뉴얼 페이지를 참조하십시오. 명령의 간략한 개요는 다음과 같습니다.

```
snoop [ -d device ] [ -o filename ] [ host hostname ]
```

-d device 로컬 네트워크 인터페이스를 지정합니다.

-o filename 캡처된 모든 패킷을 명명된 파일에 저장합니다.

hostname 특정 호스트에서만/호스트로만 이동하는 패킷을 표시합니다.

-d device 옵션은 여러 네트워크 인터페이스가 포함된 서버에서 유용합니다. 호스트를 설정하는 것 외에도 여러 표현식을 사용할 수 있습니다. 명령 표현식을 grep와 결합하여 사용하면 유용하게 활용 가능하도록 구체적인 데이터가 생성되는 경우가 많습니다.

문제를 해결할 때는 패킷이 적절한 호스트로 들어가고 적절한 호스트에서 나오는 지 확인하십시오. 또한 오류 메시지도 확인하십시오. 파일에 패킷을 저장하면 데이터를 간편하게 검토할 수 있습니다.

## truss 명령

이 명령을 사용하여 프로세스가 정지되었는지를 확인할 수 있습니다. truss 명령은 프로세스 소유자가 실행하거나 루트에서 실행해야 합니다. 이 명령에서는 여러 옵션을 사용할 수 있습니다. [truss\(1\)](#) 매뉴얼 페이지를 참조하십시오. 명령의 간략한 구문은 다음과 같습니다.

```
truss [ -t syscall ] -p pid
```

-t syscall      추적할 시스템 호출을 선택합니다.

-p pid          추적할 프로세스의 PID를 나타냅니다.

syscall 옵션은 추적할 시스템의 심프로 구분된 목록일 수 있습니다. 또한 !를 포함하여 syscall을 시작하면 나열된 시스템 호출이 추적에서 제외됩니다.

이 예에서는 프로세스가 새 클라이언트로부터의 다른 연결 요청을 대기하고 있음을 보여줍니다.

```
# /usr/bin/truss -p 243
poll(0x00024D50, 2, -1)          (sleeping...)
```

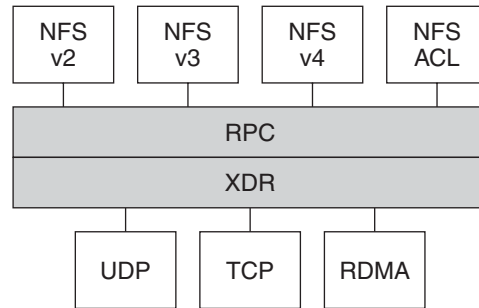
위의 예제에서는 정상 응답을 보여줍니다. 새 연결 요청을 수행한 후에도 응답이 변경되지 않으면 프로세스가 정지될 수 있습니다. 정지된 프로그램 문제를 해결하려면 [120 페이지 “NFS 서비스를 다시 시작하는 방법”](#)의 지침을 따르십시오. 문제가 정지된 프로그램 때문에 발생했는지를 완전하게 확인하려면 [117 페이지 “NFS 문제 해결 절차”](#)의 지침을 검토하십시오.

## RDMA를 통한 NFS

Oracle Solaris 11 릴리스부터는 NFS의 기본 전송이 RDMA(Remote Direct Memory Access) 프로토콜입니다. RDMA는 고속 네트워크를 통한 메모리 간 데이터 전송용 기술입니다. 구체적으로, RDMA는 CPU를 사용하지 않고 메모리에서/메모리로 직접 원격 데이터를 전송하는 기능을 제공합니다. 또한 RDMA는 직접 데이터 배치 기능도 제공하므로 데이터 복사본이 없어져 CPU 작업이 더욱 줄어듭니다. 따라서 RDMA는 호스트 CPU의 부담을 줄일 뿐 아니라 호스트 메모리 및 I/O 버스에 대한 경합도 줄여 줍니다. 이 기능을

제공하기 위해 RDMA는 Oracle Solaris 운영 체제가 설치된 SPARC 플랫폼에서 InfiniBand의 상호 연결 I/O 기술을 결합합니다. 다음 그림에서는 UDP, TCP 등의 다른 프로토콜에 대한 RDMA의 관계를 보여줍니다.

그림 6-1 RDMA와 다른 프로토콜 간의 관계



NFS는 RPC 위에 배치되는 프로토콜 제품군입니다.  
 XDR(eXternal Data Representation) 계층은  
 RPC 인수 및 RPC 결과를  
 UDP, TCP, RDMA 등의  
 여러 RPC 전송 중 하나로 인코딩합니다.

RDMA는 NFS의 기본 전송 프로토콜이므로 클라이언트나 서버에서 RDMA를 사용하기 위해 특수한 `share` 또는 `mount` 옵션은 필요하지 않습니다. 기존 자동 마운트 맵 `vfstab` 및 `dfstab`도 RDMA 전송에서 작동합니다. RDMA 전송을 통한 NFS 마운트는 SPARC 플랫폼에서 클라이언트와 서버 간의 InfiniBand 연결이 있으면 투명하게 수행됩니다. 클라이언트와 서버에서 모두 RDMA 전송을 사용할 수 없는 경우에는 TCP 전송이 초기 폴백으로 사용되고, TCP도 사용할 수 없으면 UDP가 사용됩니다. 그러나 `proto=rdma` 마운트 옵션을 사용하는 경우에는 NFS 마운트가 RDMA만 사용하도록 강제 지정됩니다.

TCP 및 UDP만 사용되도록 지정하려면 `proto=tcp/udp` mount 옵션을 사용하면 됩니다. 이 옵션을 사용하면 NFS 클라이언트에서 RDMA가 사용 안함으로 설정됩니다. NFS 마운트 옵션에 대한 자세한 내용은 `mount_nfs(1M)` man page and 148 페이지 “mount 명령”을 참조하십시오.

주 - InfiniBand용 RDMA는 IP 주소 지정 형식 및 IP 조회 기반구조를 사용하여 피어를 지정합니다. 그러나 RDMA는 별도의 프로토콜 스택이므로 모든 IP 의미를 완전하게 구현하지는 않습니다. 예를 들어 RDMA는 피어와 통신하는 데 IP 주소 지정을 사용하지 않습니다. 따라서 RDMA는 IP 주소를 기반으로 하는 여러 보안 정책의 구성을 우회할 수 있습니다. 그러나 mount 제한 및 보안 RPC와 같은 NFS 및 RPC 관리 정책은 우회되지 않습니다.

# NFS 서비스의 작동 방식

다음 절에서는 NFS 소프트웨어의 몇 가지 복잡한 기능에 대해 설명합니다. 이 절의 일부 기능 설명은 NFS 버전 4에 배타적입니다.

- 172 페이지 “NFS의 버전 협상”
- 173 페이지 “NFS 버전 4의 기능”
- 182 페이지 “UDP 및 TCP 협상”
- 182 페이지 “파일 전송 크기 협상”
- 183 페이지 “파일 시스템 마운트 방법”
- 184 페이지 “마운트 시 -public 옵션과 NFS URL의 효과”
- 184 페이지 “클라이언트측 페일오버”
- 186 페이지 “큰 파일”
- 187 페이지 “NFS 서버 로깅의 작동 방식”
- 187 페이지 “WebNFS 서비스의 작동 방식”
- 189 페이지 “웹 브라우저 사용 시의 WebNFS 제한”
- 189 페이지 “보안 NFS 시스템”
- 190 페이지 “보안 RPC”

---

주- 시스템에서 영역이 사용으로 설정된 경우 비전역 영역에서 이 기능을 사용하려면 **Oracle Solaris 관리: Oracle Solaris Zones, Oracle Solaris 10 Zones 및 리소스 관리**에서 자세한 내용을 참조하십시오.

---

## NFS의 버전 협상

NFS 초기화 프로세스에는 서버 및 클라이언트의 프로토콜 레벨 협상이 포함됩니다. 버전 레벨을 지정하지 않으면 최상의 레벨이 기본적으로 선택됩니다. 예를 들어 클라이언트와 서버가 모두 버전 3을 지원할 수 있으면 버전 3이 사용됩니다. 클라이언트나 서버 중 하나만 버전 3을 지원할 수 있으면 버전 2가 사용됩니다.

sharectl 명령을 사용하여 `client_versmin`, `client_versmax`, `server_versmin` 및 `server_versmax` 매개변수를 설정할 수 있습니다. 서버와 클라이언트에 대해 지정된 최소값 및 최대값은 이 키워드의 기본값을 교체합니다. 클라이언트와 서버 둘 다에 대해 기본 최소값은 2이고 기본 최대값은 4입니다. 서버에서 지원하는 버전을 찾기 위해 NFS 클라이언트는 먼저 `client_versmax`의 설정을 확인한 다음 `client_versmin`의 버전 설정에 도달할 때까지 각 버전을 계속 시도해 봅니다. 지원되는 버전을 찾는 즉시 프로세스가 종료됩니다. 예를 들어 `client_versmax = 4`이고 `client_versmin = 2`이면 클라이언트는 버전 4를 먼저 시도한 후에 버전 3, 버전 2 순으로 시도합니다. `client_versmax` 및 `client_versmin`을 같은 값으로 설정하면 클라이언트는 항상 해당 버전을 사용하며 다른 버전을 시도하지 않습니다. 서버에서 해당 버전을 제공하지 않으면 마운트가 실패합니다.

주 -vers 옵션이 포함된 mount 명령을 사용하여 협상을 통해 결정되는 값을 대체할 수 있습니다. `mount_nfs(1M)` 매뉴얼 페이지를 참조하십시오.

절차 정보는 91 페이지 “NFS 서비스 설정”을 참조하십시오.

## NFS 버전 4의 기능

NFS 버전 4에서는 다양한 항목이 변경되었습니다. 이 절에서는 이러한 새 기능에 대해 설명합니다.

- 173 페이지 “NFS 버전 4에서 파일 시스템 공유 해제 및 다시 공유”
- 174 페이지 “NFS 버전 4의 파일 시스템 네임스페이스”
- 175 페이지 “NFS 버전 4의 회발성 파일 핸들”
- 176 페이지 “NFS 버전 4의 클라이언트 복구”
- 178 페이지 “NFS 버전 4의 OPEN 공유 지원”
- 179 페이지 “NFS 버전 4의 위임”
- 181 페이지 “NFS 버전 4의 ACL 및 nfsmapid”
- 186 페이지 “NFS 버전 4의 클라이언트측 페일오버”

주 - Solaris 10 릴리스부터는 NFS 버전 4가 LIPKEY/SPKM 보안 종류를 지원하지 않습니다. 또한 NFS 버전 4에서는 mountd, nfslogd 및 statd 데몬을 사용하지 않습니다.

NFS 버전 4 사용과 관련된 절차 정보는 91 페이지 “NFS 서비스 설정”을 참조하십시오.

## NFS 버전 4에서 파일 시스템 공유 해제 및 다시 공유

NFS 버전 3 및 버전 4가 모두 있는 경우 클라이언트가 공유 해제된 파일 시스템에 액세스하려고 하면 서버가 오류 코드로 응답을 보냅니다. 그러나 NFS 버전 3에서는 파일 시스템 공유를 해제하기 전에 클라이언트가 얻은 잠금을 서버에서 유지 관리합니다. 따라서 파일 시스템을 다시 공유하면 NFS 버전 3 클라이언트는 파일 시스템 공유를 해제한 적이 없었던 것처럼 파일 시스템에 액세스할 수 있습니다.

NFS 버전 4를 사용하는 경우 파일 시스템 공유를 해제하면 해당 파일 시스템의 모든 파일 잠금 또는 열린 파일 상태가 삭제됩니다. 클라이언트가 이러한 파일 또는 잠금에 액세스하려고 하면 오류가 표시됩니다. 이 오류는 보통 응용 프로그램에 대한 I/O 오류로 보고됩니다. 그러나 옵션을 변경하기 위해 현재 공유된 파일 시스템을 다시 공유해도 서버의 상태가 삭제되지는 않습니다.

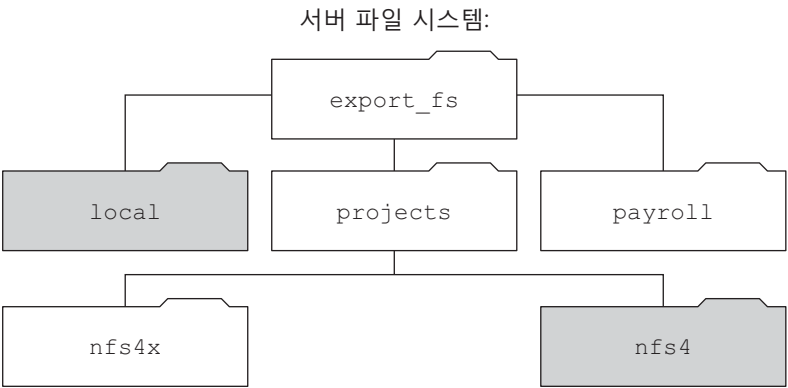
관련 정보는 176 페이지 “NFS 버전 4의 클라이언트 복구” 또는 `unshare_nfs(1M)` 매뉴얼 페이지를 참조하십시오.

## NFS 버전 4의 파일 시스템 네임스페이스

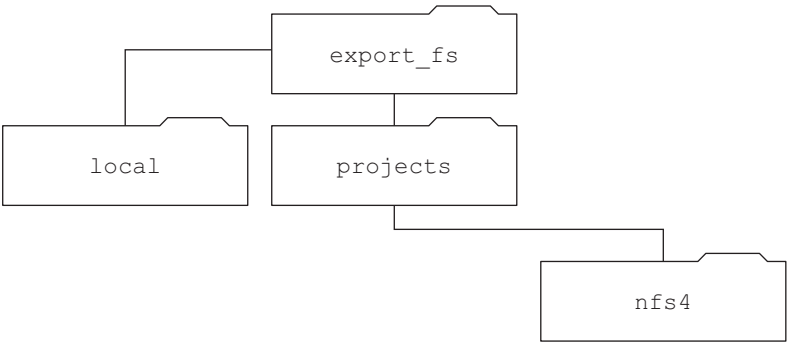
NFS 버전 4 서버는 의사 파일 시스템을 만들고 유지 관리합니다. 따라서 클라이언트가 서버의 모든 내보낸 개체에 원활하게 액세스할 수 있습니다. NFS 버전 4 이전에는 의사 파일 시스템이 없었습니다. 즉, 클라이언트는 액세스할 각 공유 서버 파일 시스템을 강제로 마운트해야 했습니다. 다음 예를 고려하십시오.

그림 6-2 서버 파일 시스템 및 클라이언트 파일 시스템 보기

서버 내보내기:	서버 파일 시스템:
/export_fs/local	/
/export_fs/projects/nfs4	/export_fs



서버의 export\_fs 디렉토리에 대한 클라이언트 보기:



■ 내보낸 디렉토리

클라이언트는 `payroll` 디렉토리 및 `nfs4x` 디렉토리를 볼 수 없습니다. 이러한 디렉토리는 내보내기되지 않으므로 내보낸 디렉토리에 포함되지 않기 때문입니다. 그러나 `local` 디렉토리는 클라이언트에 표시됩니다. `local`은 내보낸 디렉토리이기 때문입니다. `projects` 디렉토리는 클라이언트에 표시됩니다. `projects`는 내보낸 디렉토리(`nfs4`)에 포함되기 때문입니다. 따라서 명시적으로 내보내지 않은 서버 네임스페이스 부분은 내보낸 디렉토리 및 서버 내보내기에 포함되는 디렉토리만 표시되는 의사 파일 시스템에 브릿지됩니다.

의사 파일 시스템은 디렉토리만 포함하는 구조로, 서버에서 만들어집니다. 의사 파일 시스템에서는 클라이언트가 내보낸 파일 시스템의 계층을 찾아볼 수 있도록 합니다. 따라서, 클라이언트의 의사 파일 시스템 보기는 내보낸 파일 시스템에 포함되는 경로로 한정됩니다.

이전 NFS 버전에서는 각 파일 시스템을 마운트하지 않으면 클라이언트가 서버 파일 시스템을 순회할 수 없습니다. 그러나 NFS 버전 4에서는 서버 네임스페이스가 다음을 수행합니다.

- 클라이언트의 파일 시스템 보기를 서버 내보내기에 포함되는 디렉토리로 제한합니다.
- 클라이언트가 각 기본 파일 시스템을 마운트하지 않고도 서버 내보내기에 원활하게 액세스할 수 있도록 합니다. 이전 예제를 참조하십시오. 그러나 각 운영 체제에 따라 클라이언트가 각 서버 파일 시스템을 마운트해야 할 수도 있습니다.

POSIX 관련 이유로 인해, Oracle Solaris NFS 버전 4 클라이언트에서는 서버 파일 시스템 경계를 교차하지 않습니다. 이러한 시도를 하는 경우 클라이언트는 디렉토리가 비어 있는 것으로 표시되도록 합니다. 이러한 상황을 완화하려면 서버의 각 파일 시스템에 대해 마운트를 수행해야 합니다.

## NFS 버전 4의 휘발성 파일 핸들

파일 핸들은 서버에서 만들어지며 파일 및 디렉토리를 고유하게 식별하는 정보를 포함합니다. NFS 버전 2 및 3에서는 서버가 지속 파일 핸들을 반환합니다. 그러므로 클라이언트는 서버가 항상 동일 파일을 참조하는 파일 핸들을 생성하도록 보장할 수 있습니다. 예를 들면 다음과 같습니다.

- 파일을 삭제한 다음 이름이 같은 파일로 교체하는 경우에는 서버에서 새 파일에 대해 새 파일 핸들을 생성합니다. 클라이언트가 이전 파일 핸들을 사용한 경우에는 서버에서 파일 핸들이 사용되지 않는다는 오류를 반환합니다.
- 파일 이름을 바꾼 경우에도 파일 핸들이 동일하게 유지됩니다.
- 서버를 재부트해야 하는 경우에도 파일 핸들은 동일하게 유지됩니다.

그러므로 서버에서 파일 핸들을 포함하는 클라이언트로부터의 요청을 받은 경우 파일이 즉시 확인되며 파일 핸들이 항상 올바른 파일을 참조합니다.

대부분의 UNIX 기반 서버에서는 이와 같은 NFS 작업에 대한 파일 및 디렉토리 식별 방법을 사용할 수 있었습니다. 그러나 파일 경로 이름 등의 다른 식별 방법을 사용했던 서버에서는 이 방법을 구현할 수 없습니다. 이 문제를 해결하려면 NFS 버전 4 프로토콜에서는 서버가 해당 파일 핸들이 휘발성임을 선언하도록 허용합니다. 따라서 파일 핸들이 변경될 수 있습니다. 파일 핸들이 변경되면 클라이언트는 새 파일 핸들을 찾아야 합니다.

NFS 버전 2 및 3에서와 같이, Oracle Solaris NFS 버전 4 서버에서는 항상 지속 파일 핸들을 제공합니다. 그러나 Solaris NFS 버전 4 이외의 서버에 액세스하는 Oracle Solaris NFS 버전 4 클라이언트는 휘발성 파일 핸들(서버에서 사용하는 경우)을 지원해야 합니다. 구체적으로, 서버에서 파일 핸들이 휘발성임을 클라이언트에 알리면 클라이언트는 경로 이름과 파일 핸들 간 매핑을 캐시해야 합니다. 클라이언트는 휘발성 파일 핸들을 만료될 때까지 사용합니다. 핸들 만료 시에는 클라이언트가 다음을 수행합니다.

- 해당 파일 핸들을 참조하는 캐시된 정보를 비웁니다.
- 해당 파일의 새 파일 핸들을 검색합니다.
- 작업을 재시도합니다.

---

**주** - 서버는 지속 파일 핸들과 휘발성 파일 핸들을 항상 클라이언트에 알려 줍니다.

---

휘발성 파일 핸들은 다음과 같은 이유로 만료될 수 있습니다.

- 파일을 닫을 때
- 파일 핸들의 파일 시스템을 마이그레이션할 때
- 클라이언트가 파일 이름을 바꿀 때
- 서버를 재부트할 때

클라이언트가 새 파일 핸들을 찾을 수 없으면 `syslog` 파일에 오류 메시지가 기록됩니다. 이 파일에 액세스하려는 추가 시도는 실패하고 I/O 오류가 발생합니다.

## NFS 버전 4의 클라이언트 복구

NFS 버전 4 프로토콜은 `stateful` 프로토콜입니다. 클라이언트와 서버가 모두 다음에 대한 현재 정보를 유지 관리하는 경우 프로토콜은 `Stateful` 상태입니다.

- 파일 열기
- 파일 잠금

서버 충돌 등의 오류가 발생하면 클라이언트와 서버가 함께 작동하여 오류 전에 존재했던 열기 및 잠금 상태를 재설정합니다.

서버가 충돌하여 재부트되면 서버 상태가 손실됩니다. 클라이언트는 서버가 재부트되었음을 감지하고 서버의 상태 재구성 지원 과정을 시작합니다. 클라이언트가 프로세스를 진행하므로 이 프로세스를 클라이언트 복구라고 합니다.



클라이언트는 서버가 재부트되었음을 확인하면 현재 작업을 즉시 일시 중지하고 클라이언트 복구 프로세스를 시작합니다. 복구 프로세스가 시작되면 다음과 같은 메시지가 시스템 오류 로그 `/var/adm/messages`에 표시됩니다.

NOTICE: Starting recovery server *basil.example.company.com*

복구 프로세스 중에 클라이언트는 이전 클라이언트 상태에 대한 정보를 서버로 보냅니다. 그러나 이 기간 중에 클라이언트는 서버로 새 요청을 보내지 않습니다. 파일을 열거나 파일 잠금을 설정하는 새 요청은 서버의 복구 기간이 완료될 때까지 기다린 후에 진행됩니다.

클라이언트 복구 프로세스가 완료되면 다음 메시지가 시스템 오류 로그 `/var/adm/messages`에 표시됩니다.

NOTICE: Recovery done for server *basil.example.company.com*

그러면 상태 메시지를 서버로 보내는 클라이언트의 작업이 정상적으로 완료된 것입니다. 그러나 클라이언트에서 이 프로세스를 완료했다라도 다른 클라이언트에서 서버로 상태 정보를 보내는 프로세스를 완료하지 않았을 수 있습니다. 따라서 일정 시간 동안 서버는 열기 또는 잠금 요청을 수락하지 않습니다. 이 기간(유예 기간이라고도 함)은 모든 클라이언트가 복구를 완료할 수 있도록 지정됩니다.

유예 기간 동안 클라이언트가 새 파일을 열거나 새 잠금을 설정하려고 시도하면 서버에서 요청을 거부하고 GRACE 오류 코드를 표시합니다. 이 오류를 받으면 클라이언트는 유예 기간이 종료될 때까지 기다렸다가 서버로 요청을 다시 보내야 합니다. 유예 기간 동안에는 다음 메시지가 표시됩니다.

NFS server recovering

유예 기간 동안에도 파일을 열지 않거나 파일 잠금을 설정하지 않는 명령은 계속 실행할 수 있습니다. 예를 들어 `ls` 및 `cd` 명령은 파일을 열거나 파일 잠금을 설정하지 않습니다. 따라서 이러한 명령은 일시 중지되지 않습니다. 그러나 파일을 여는 `cat` 등의 명령은 유예 기간이 종료될 때까지 일시 중지됩니다.

유예 기간이 종료되면 다음 메시지가 표시됩니다.

NFS server recovery ok.

그러면 클라이언트는 새 열기 및 잠금 요청을 서버로 보낼 수 있습니다.

클라이언트 복구는 여러 가지 이유로 실패할 수 있습니다. 예를 들어 서버 재부트 후 네트워크 파티션이 있는 경우 클라이언트가 유예 기간 종료 전에 서버와의 상태를 재설정하지 못할 수 있습니다. 유예 기간이 종료되어도 서버는 클라이언트가 상태를 재설정하도록 허용하지 않습니다. 새 상태 작업으로 인해 충돌이 발생할 수 있기

때문입니다. 예를 들어 새 파일 잠금이 클라이언트에서 복구하려고 하는 이전 파일 잠금과 충돌할 수 있습니다. 이러한 상황이 발생하면 서버에서는 **NO\_GRACE** 오류 코드를 클라이언트로 반환합니다.

특정 파일에 대한 열기 작업 복구가 실패하면 클라이언트는 해당 파일을 사용할 수 없는 것으로 표시하며, 다음 메시지가 표시됩니다.

```
WARNING: The following NFS file could not be recovered and was marked dead
(can't reopen: NFS status 70): file : filename
```

**70**이라는 숫자는 예제로만 사용됩니다.

복구 중에 파일 잠금 재설정이 실패하면 다음 오류 메시지가 게시됩니다.

```
NOTICE: nfs4_send_siglost: pid PROCESS-ID lost
lock on server SERVER-NAME
```

이 경우 **SIGLOST** 신호가 프로세스에 게시됩니다. **SIGLOST** 신호의 기본 작업은 프로세스를 종료하는 것입니다.

이 상태에서 복구하려면 오류 시 파일을 열고 있던 응용 프로그램을 다시 시작해야 합니다. 이때 다음과 같은 현상이 발생할 수 있습니다.

- 파일을 다시 열지 않은 일부 프로세스에서 **I/O** 오류가 발생할 수 있습니다.
- 파일을 다시 열었거나 복구 실패 후 열기 작업을 수행한 다른 프로세스에서는 문제 없이 파일에 액세스할 수 있습니다.

따라서 일부 프로세스는 특정 파일에 액세스할 수 있는 반면 다른 프로세스는 액세스할 수 없습니다.

## NFS 버전 4의 OPEN 공유 지원

NFS 버전 4 프로토콜에서는 클라이언트가 다른 클라이언트의 파일 액세스를 제어하는 데 사용할 수 있는 다양한 파일 공유 모드를 제공합니다. 클라이언트는 다음을 지정할 수 있습니다.

- **DENY\_NONE** 모드는 다른 클라이언트의 파일 읽기 및 쓰기 액세스를 허용합니다.
- **DENY\_READ** 모드는 다른 클라이언트의 파일 읽기 액세스를 거부합니다.
- **DENY\_WRITE** 모드는 다른 클라이언트의 파일 쓰기 액세스를 거부합니다.
- **DENY\_BOTH** 모드는 다른 클라이언트의 파일 읽기 및 쓰기 액세스를 거부합니다.

Oracle Solaris NFS 버전 4 서버에서는 이러한 파일 공유 모드를 완전하게 구현합니다. 따라서 클라이언트가 현재 공유 모드와 충돌하는 방식으로 파일을 열려고 하면 서버에서 작업이 실패하도록 하여 해당 시도를 거부합니다. 열기 또는 만들기 작업을 시작할 때 이러한 시도가 실패하면 NFS 버전 4 클라이언트는 프로토콜 오류를 받게 됩니다. 이 오류는 응용 프로그램 오류 **EACCES**에 매핑됩니다.

프로토콜에서 여러 공유 모드를 제공하기는 하지만, 현재 Oracle Solaris에서 열기 작업을 수행하는 경우 여러 공유 모드가 제공되지 않습니다. 파일을 열 때 Oracle Solaris NFS 버전 4 클라이언트는 `DENY_NONE` 모드만 사용할 수 있습니다.

또한 `fcntl` 시스템 호출에는 파일 공유를 제어하는 `F_SHARE` 명령이 있지만, NFS 버전 4에서는 `fcntl` 명령을 올바르게 구현할 수 없습니다. NFS 버전 4 클라이언트에서 이러한 `fcntl` 명령을 사용하는 경우 클라이언트가 `EAGAIN` 오류를 응용 프로그램으로 반환합니다.

## NFS 버전 4의 위임

NFS 버전 4에서는 위임을 위한 클라이언트 지원과 서버 지원을 모두 제공합니다. 위임은 서버에서 파일 관리를 클라이언트에 위임하는 기술입니다. 예를 들어 서버에서 읽기 위임 또는 쓰기 위임을 클라이언트에 부여할 수 있습니다. 읽기 위임은 서로 충돌하지 않으므로 여러 클라이언트에게 동시에 부여할 수 있습니다. 쓰기 위임은 다른 클라이언트의 파일 액세스와 충돌하므로 한 클라이언트에게만 부여할 수 있습니다. 쓰기 위임을 보유한 클라이언트는 파일에 배타적으로 액세스할 수 있기 때문에 서버로 여러 작업을 보내지 않습니다. 마찬가지로, 읽기 위임을 보유한 클라이언트도 서버로 여러 작업을 보내지 않습니다. 서버에서 어떤 클라이언트도 쓰기 모드로 파일을 열지 못하도록 보장하기 때문입니다. 위임을 사용하는 경우 위임된 파일에 대해 서버와 클라이언트의 상호 작용이 크게 줄어듭니다. 따라서 네트워크 트래픽이 감소하고 클라이언트와 서버의 성능이 개선됩니다. 그러나 성능 개선 정도는 응용 프로그램에서 사용하는 파일 상호 작용의 종류와 네트워크 및 서버 혼잡 정도에 따라 달라집니다.

위임을 부여할지 여부는 전적으로 서버에서 결정합니다. 클라이언트는 위임을 요청하지 않습니다. 서버는 파일에 대한 액세스 패턴을 기반으로 위임을 부여할지 여부를 결정합니다. 서로 다른 여러 클라이언트에서 최근 쓰기 모드로 파일에 액세스한 경우 서버에서 위임을 부여하지 않았을 수 있습니다. 이 액세스 패턴은 이후 충돌 가능성을 나타내기 때문입니다.

클라이언트가 현재 파일에 대해 부여된 위임과 일치하지 않는 방식으로 해당 파일에 액세스하면 충돌이 발생합니다. 예를 들어 클라이언트가 파일에 대한 쓰기 위임을 보유하고 있는데 두번째 클라이언트가 읽기 또는 쓰기 액세스를 위해 해당 파일을 열면 서버에서 첫번째 클라이언트의 쓰기 위임을 회수합니다. 마찬가지로, 클라이언트에서 읽기 위임을 보유하고 있는데 다른 클라이언트가 쓰기를 위해 같은 파일을 열면 서버가 읽기 위임을 회수합니다. 두 상황에서 모두 충돌이 발생하므로 두번째 클라이언트에게는 위임이 부여되지 않습니다. 충돌이 발생하면 서버에서는 콜백 방식을 사용하여 현재 위임을 보유한 클라이언트에 연결합니다. 이 콜백을 받으면 클라이언트는 파일의 업데이트된 상태를 서버로 보내고 위임을 반환합니다. 클라이언트가 회수에 응답하지 못하면 서버에서 위임을 해지합니다. 이 경우 서버는 해당 파일에 대한 클라이언트의 모든 작업을 거부하며 클라이언트는 요청한 작업을 실패로 보고합니다. 일반적으로 이러한 실패는 응용 프로그램에 `I/O` 오류로 보고됩니다.

이 오류에서 복구하려면 파일을 닫았다가 다시 열어야 합니다. 클라이언트가 위임을 보유한 상태에서 클라이언트와 서버 간에 네트워크 파티션이 있으면 해지된 위임에서 오류가 발생할 수 있습니다.

서버는 다른 서버에 저장된 파일에 대한 액세스 충돌을 해결하지는 않습니다. 따라서 NFS 서버는 서버에 저장된 파일에 대해서만 충돌을 해결합니다. 또한, 여러 NFS 버전을 실행하는 클라이언트에서 발생하는 충돌에 대해 NFS 서버는 NFS 버전 4를 실행하는 클라이언트에 대한 회수만 시작합니다. NFS 서버는 이전 NFS 버전을 실행 중인 클라이언트에 대해 회수를 시작할 수 없습니다.

충돌 감지 프로세스는 상황에 따라 다릅니다. 예를 들어, NFS 버전 4와는 달리 버전 2와 버전 3에는 열기 절차가 없기 때문에 충돌은 클라이언트가 파일을 읽거나 쓰거나 잠그려고 시도한 후에만 감지됩니다. 이러한 충돌에 대한 서버의 응답도 다릅니다. 예를 들면 다음과 같습니다.

- NFS 버전 3의 경우에는 서버에서 JUKEBOX 오류를 반환하며, 그러면 클라이언트가 액세스 요청을 중지했다가 나중에 다시 시도합니다. 클라이언트는 File unavailable 메시지를 인쇄합니다.
- NFS 버전 2의 경우에는 JUKEBOX 오류에 해당하는 항목이 없기 때문에 서버에서 응답을 하지 않으며, 클라이언트는 기다렸다가 다시 시도합니다. 클라이언트는 NFS server not responding 메시지를 인쇄합니다.

위임 충돌이 해결되면 이러한 상황도 해결됩니다.

기본적으로 서버 위임은 사용으로 설정됩니다. server\_delegation 매개변수를 none으로 설정하여 위임을 사용 안함으로 설정할 수 있습니다. 절차 정보는 [93 페이지 “서버에서 다른 NFS 버전을 선택하는 방법”](#)을 참조하십시오.

클라이언트 위임에는 키워드가 필요하지 않습니다. NFS 버전 4 콜백 데몬 nfs4cbd는 클라이언트에서 콜백 서비스를 제공합니다. 이 데몬은 NFS 버전 4에 대한 마운트를 사용으로 설정할 때마다 자동으로 시작됩니다. 기본적으로 클라이언트는 /etc/netconfig 시스템 파일에 나와 있는 모든 인터넷 전송에 대해 필요한 콜백 정보를 서버에 제공합니다. IPv6에 대해 클라이언트를 사용으로 설정하는 경우 클라이언트 이름의 IPv6 주소를 확인할 수 있으면 콜백 데몬은 IPv6 연결을 수락합니다.

콜백 데몬은 일시 프로그램 번호와 동적으로 지정된 포트 번호를 사용합니다. 이 정보는 서버에 제공되며, 서버는 위임을 부여하기 전에 콜백 경로를 테스트합니다. 콜백 경로 테스트가 실패하면 서버에서 위임을 부여하지 않습니다. 외부에서 확인 가능한 동작은 이 동작뿐입니다.

콜백 정보는 NFS 버전 4 요청 내에 내장되므로, 서버는 Network Address Translation(NAT)을 사용하는 장치를 통해 클라이언트에 연결할 수 없습니다. 또한 콜백 데몬은 동적 포트 번호를 사용합니다. 따라서 방화벽이 포트 2049에서 일반 NFS 트래픽을 사용으로 설정해도 서버가 방화벽을 순회하지 못할 수 있습니다. 이 경우에는 서버에서 위임을 할당하지 않습니다.

## NFS 버전 4의 ACL 및 nfsmapid

액세스 제어 목록(ACL)은 파일 소유자가 파일 소유자, 그룹 및 기타 특정 사용자/그룹의 파일 권한을 정의할 수 있도록 하여 보다 효율적인 파일 보안을 제공합니다. ZFS 파일 시스템에서 ACL은 `chmod` 명령을 사용하여 서버와 클라이언트에서 설정됩니다. UFS 파일 시스템에서는 `setfacl` 명령을 사용합니다. 자세한 내용은 `chmod(1)` 및 `setfacl(1)` 매뉴얼 페이지를 참조하십시오. NFS 버전 4에서는 ID 매핑 `nfsmapid`를 사용하여 서버의 ACL 항목에 있는 사용자 또는 그룹 ID를 클라이언트의 ACL 항목에 있는 사용자 또는 그룹 ID로 매핑합니다. 그 반대의 경우도 마찬가지입니다. ACL 항목의 사용자 및 그룹 ID는 클라이언트와 서버에 모두 있어야 합니다.

### ID 매핑 실패 이유

다음과 같은 상황에서 ID 매핑이 실패할 수 있습니다.

- 서버의 ACL 항목에 있는 사용자 또는 그룹을 클라이언트의 유효한 사용자 또는 그룹에 매핑할 수 없는 경우 사용자가 ACL을 읽을 수는 있지만 일부 사용자 또는 그룹이 "알 수 없음"으로 표시됩니다.  
예를 들어 `ls -lv` 또는 `ls -lv` 명령을 실행하면 일부 ACL 항목의 그룹 또는 사용자가 "알 수 없음"으로 표시됩니다. 이 명령에 대한 자세한 내용은 `ls(1)` 매뉴얼 페이지를 참조하십시오.
- 클라이언트에 설정되어 있는 ACL 항목의 사용자 또는 그룹 ID를 서버의 유효한 사용자 또는 그룹 ID에 매핑할 수 없는 경우 `setfacl` 또는 `chmod` 명령이 실패할 수 있으며 **권한이 거부됨** 오류 메시지가 반환됩니다.
- 클라이언트와 서버의 `nfsmapid_domain` 값이 일치하지 않으면 ID 매핑은 실패합니다. 자세한 내용은 [139 페이지 "nfsmapid 데몬"](#)을 참조하십시오.

### ACL을 사용한 ID 매핑 문제 방지

ID 매핑 문제를 방지하려면 다음을 수행합니다.

- `nfsmapid_domain`의 값이 올바르게 설정되어 있는지 확인합니다.
- ACL 항목의 모든 사용자 및 그룹 ID가 NFS 버전 4 클라이언트와 서버에 모두 있는지 확인합니다.

### 매핑되지 않은 사용자 또는 그룹 ID 확인

사용자 또는 그룹을 서버나 클라이언트에서 매핑할 수 없는지 확인하려면 다음 스크립트를 사용합니다.

```
#!/usr/sbin/dtrace -Fs
sdt:::nfs4-acl-nobody
{
    printf("validate_idmapping: (%s) in the ACL could not be mapped!",
```

```
stringof(arg0));
}
```

---

주 - 이 스크립트에서 사용되는 검사 이름은 이후 변경될 수 있는 인터페이스입니다. 자세한 내용은 [Solaris Dynamic Tracing Guide](#)의 “Stability Levels”을 참조하십시오.

---

## ACL 또는 nfsmapid에 대한 추가 정보

다음 항목을 참조하십시오.

- [Oracle Solaris 관리: ZFS 파일 시스템의 8 장, “ACL 및 속성을 사용하여 Oracle Solaris ZFS 파일 보호”](#)
- [139 페이지 “nfsmapid 데몬”](#)

## UDP 및 TCP 협상

시작 중에는 전송 프로토콜도 협상합니다. 기본적으로 클라이언트와 서버에서 모두 지원되는 첫번째 연결 지향 전송이 선택됩니다. 이러한 전송을 선택할 수 없으면 사용 가능한 첫번째 비연결 프로토콜이 사용됩니다. 시스템에서 지원되는 전송 프로토콜은 `/etc/netconfig`에 나와 있습니다. 이번 릴리스에서 지원되는 연결 지향 전송 프로토콜은 TCP입니다. 비연결 전송 프로토콜은 UDP입니다.

NFS 프로토콜 버전과 전송 프로토콜이 모두 협상을 통해 결정되면 NFS 프로토콜 버전이 전송 프로토콜보다 우선적으로 사용됩니다. UDP를 사용하는 NFS 버전 3 프로토콜의 우선 순위가 TCP를 사용하는 NFS 버전 2 프로토콜보다 높습니다. `mount` 명령을 사용하면 NFS 프로토콜 버전과 전송 프로토콜을 모두 수동으로 선택할 수 있습니다.

`mount_nfs(1M)` 매뉴얼 페이지를 참조하십시오. 대부분의 경우에는 협상에서 최적의 옵션을 선택하도록 허용합니다.

## 파일 전송 크기 협상

파일 전송 크기는 클라이언트와 서버 간에 데이터를 전송할 때 사용되는 버퍼 크기를 설정합니다. 일반적으로는 전송 크기가 클수록 효율적입니다. NFS 버전 4 프로토콜에서는 전송 크기가 무제한입니다. 클라이언트는 필요한 경우 마운트 시에 더 작은 전송 크기를 사용할 수 있지만, 대부분의 경우에는 이렇게 할 필요가 없습니다.

NFS 버전 2 프로토콜을 사용하는 시스템과는 전송 크기를 협상하지 않습니다. 이 경우의 최대 전송 크기는 8KB로 설정됩니다.



mount 명령에서 `-rsize` 및 `-wsize` 옵션을 사용하여 전송 크기를 수동으로 설정할 수 있습니다. 일부 PC 클라이언트의 경우에는 전송 크기를 줄여야 할 수 있습니다. NFS 서버가 더 큰 전송 크기를 사용하도록 구성되어 있는 경우에도 전송 크기를 늘릴 수 있습니다.

주 - Solaris 10 릴리스부터는 유선 전송 크기 제한이 완화되었습니다. 전송 크기는 기본 전송 기능을 기반으로 합니다. 예를 들어 UDP에 대한 NFS 전송 제한은 여전히 32KB입니다. 그러나 TCP가 UDP의 데이터그램 제한이 없는 스트리밍 프로토콜이기 때문에 TCP를 통한 최대 전송 크기가 1MB로 늘어났습니다.

## 파일 시스템 마운트 방법

다음 설명은 NFS 버전 3 마운트에 적용됩니다. NFS 버전 4 마운트 프로세스에는 포트맵 서비스 및 MOUNT 프로토콜이 포함되지 않습니다.

클라이언트가 서버에서 파일 시스템을 마운트해야 하는 경우에는 서버로부터 파일 핸들을 가져와야 합니다. 파일 핸들은 파일 시스템에 해당해야 합니다. 이 프로세스를 수행하려면 클라이언트와 서버 간에 여러 트랜잭션을 수행해야 합니다. 이 예에서는 클라이언트가 서버에서 `/home/terry`를 마운트하려고 합니다. 그런 후에 이 트랜잭션에 대한 snoop 추적이 진행됩니다.

```
client -> server PORTMAP C GETPORT prog=100005 (MOUNT) vers=3 proto=UDP
server -> client PORTMAP R GETPORT port=33492
client -> server MOUNT3 C Null
server -> client MOUNT3 R Null
client -> server MOUNT3 C Mount /export/home9/terry
server -> client MOUNT3 R Mount OK FH=9000 Auth=unix
client -> server PORTMAP C GETPORT prog=100003 (NFS) vers=3 proto=TCP
server -> client PORTMAP R GETPORT port=2049
client -> server NFS C NULL3
server -> client NFS R NULL3
client -> server NFS C FSINFO3 FH=9000
server -> client NFS R FSINFO3 OK
client -> server NFS C GETATTR3 FH=9000
server -> client NFS R GETATTR3 OK
```

이 추적에서 클라이언트는 먼저 NFS 서버의 포트맵 서비스에서 마운트 포트 번호를 요청합니다. 클라이언트가 마운트 포트 번호(33492)를 받으면 해당 번호를 사용하여 서버의 서비스 사용 가능 여부를 테스트합니다. 클라이언트는 서비스가 해당 포트 번호에서 실행 중임을 확인하고 나면 마운트 요청을 수행합니다. 서버는 이 요청에 응답할 때 마운트하려는 파일 시스템의 파일 핸들(9000)을 포함합니다. 그러면 클라이언트에서 NFS 포트 번호에 대한 요청을 보냅니다. 클라이언트는 서버로부터 번호를 받으면 NFS 서비스(nfsd)의 사용 가능 여부를 테스트합니다. 또한 클라이언트는 파일 핸들을 사용하는 파일 시스템에 대한 NFS 정보를 요청합니다.

다음 추적에서는 클라이언트가 `public` 옵션을 사용하여 파일 시스템을 마운트합니다.

```

client -> server NFS C LOOKUP3 FH=0000 /export/home9/terry
server -> client NFS R LOOKUP3 OK FH=9000
client -> server NFS C FSINFO3 FH=9000
server -> client NFS R FSINFO3 OK
client -> server NFS C GETATTR3 FH=9000
server -> client NFS R GETATTR3 OK

```

기본 공용 파일 핸들(0000)을 사용하면 포트맵 서비스에서 정보를 가져오고 NFS 포트 번호를 확인하는 모든 트랜잭션을 건너뛸니다.

---

주-NFS 버전 4 프로토콜에서는 휘발성 파일 핸들을 지원합니다. 자세한 내용은 175 페이지 “NFS 버전 4의 휘발성 파일 핸들”을 참조하십시오.

---

## 마운트 시 -public 옵션과 NFS URL의 효과

-public 옵션을 사용하면 마운트가 실패하는 상황이 발생할 수 있습니다. NFS URL을 추가하는 경우에도 비슷한 상황이 발생할 수 있습니다. 다음 목록에서는 이러한 옵션을 사용하는 경우 파일 시스템을 마운트하는 방법을 구체적으로 설명합니다.

**NFS URL의 public 옵션** - 공용 파일 핸들을 사용하도록 강제 지정합니다. 공용 파일 핸들이 지원되지 않으면 마운트는 실패합니다.

**정규 경로의 public 옵션** - 공용 파일 핸들을 사용하도록 강제 지정합니다. 공용 파일 핸들이 지원되지 않으면 마운트는 실패합니다.

**NFS URL만** - NFS 서버에서 공용 파일 핸들이 사용으로 설정된 경우에만 해당 파일 핸들을 사용합니다. 공용 파일 핸들 사용 시 마운트가 실패하면 MOUNT 프로토콜을 사용하여 마운트를 시도해 보십시오.

**정규 경로만** - 공용 파일 핸들을 사용하지 않습니다. MOUNT 프로토콜이 사용됩니다.

## 클라이언트측 페일오버

NFS 클라이언트는 클라이언트측 페일오버를 사용하여 동일한 데이터를 제공하는 여러 서버를 파악하고 현재 서버를 사용할 수 없게 되면 대체 서버로 전환할 수 있습니다. 다음 중 하나가 발생하는 경우 파일 시스템을 사용하지 못할 수 있습니다.

- 파일 시스템이 연결된 서버가 충돌하는 경우
- 서버가 과부하되는 경우
- 네트워크 결함이 발생하는 경우

이러한 상황에서 페일오버는 일반적으로 사용자에게 투명하게 수행됩니다. 따라서 페일오버는 클라이언트에서 실행 중인 프로세스를 중단하지 않고 언제든지 수행될 수 있습니다.



페일오버를 수행하려면 파일 시스템이 읽기 전용으로 마운트되어 있어야 합니다. 파일 시스템이 동일해야 페일오버가 정상적으로 수행됩니다. 파일 시스템을 동일하게 만드는 요소에 대한 설명은 [185 페이지 “복제된 파일 시스템이란?”](#)을 참조하십시오. 정적 파일 시스템 또는 자주 변경되지 않는 파일 시스템이 페일오버를 수행하기에 가장 적합합니다.

같은 NFS 마운트에서 CacheFS와 클라이언트측 페일오버를 사용할 수는 없습니다. 각 CacheFS 파일 시스템에 대한 추가 정보가 저장됩니다. 페일오버 중에는 이 정보를 업데이트할 수 없으므로 파일 시스템을 마운트할 때는 이 두 기능 중 하나만 사용할 수 있습니다.

모든 파일 시스템에 대해 설정해야 하는 복제의 수는 여러 요인에 따라 달라집니다. 이상적으로는 최소 2개 서버가 있어야 합니다. 각 서버는 여러 서브넷을 지원해야 합니다. 각 서브넷에서 고유한 서버를 사용하는 것보다 이 설정이 효율적입니다. 프로세스를 수행하려면 나열된 각 서버를 확인해야 합니다. 따라서 나열된 서버의 수가 많으면 각 마운트 속도는 더 느려집니다.

## 페일오버 용어

프로세스를 완전하게 구현하려면 두 가지 용어를 이해해야 합니다.

- **페일오버** -복제된 파일 시스템을 지원하는 서버 목록에서 서버를 선택하는 프로세스입니다. 일반적으로 정렬된 목록에서 다음 서버가 사용됩니다(응답하지 못하는 경우는 제외).
- **다시 매핑** - 새 서버를 사용합니다. 정상 사용 시 클라이언트는 원격 파일 시스템에 있는 각 활성 파일의 경로 이름을 저장합니다. 다시 매핑 중에는 이러한 경로 이름을 평가하여 새 서버에서 파일을 찾습니다.

## 복제된 파일 시스템이란?

각 파일의 크기가 같고 파일 크기나 파일 형식이 원본 파일 시스템과 같으면 페일오버를 위해 파일 시스템을 **복제**로 사용할 수 있습니다. 이때 권한, 생성일 및 기타 파일 속성은 고려하지 않습니다. 파일 크기나 파일 형식이 다른 경우 다시 매핑이 실패하고 이전 서버를 사용할 수 있을 때까지 프로세스가 정지됩니다. NFS 버전 4에서는 동작이 다릅니다. [186 페이지 “NFS 버전 4의 클라이언트측 페일오버”](#)를 참조하십시오.

rsync, cpio 또는 다른 파일 전송 방식을 사용하여 복제된 파일 시스템을 유지 관리할 수 있습니다. 복제된 파일 시스템을 업데이트하면 불일치가 발생하므로 최선의 결과를 위해서는 다음 사항을 고려하십시오.

- 새 파일 버전을 설치하기 전에 이전 파일 버전의 이름 바꾸기
- 클라이언트 사용량이 낮은 야간에 업데이트 실행
- 업데이트를 소규모로 유지
- 복사본 수 최소화

## 파일오버 및 NFS 잠금

일부 소프트웨어 패키지의 경우 파일에 대한 읽기 잠금이 필요합니다. 이러한 제품이 손상되지 않도록 하기 위해 읽기 전용 파일 시스템에 대한 읽기 잠금은 허용은 되지만 클라이언트측에서만 확인할 수 있습니다. 서버는 잠금을 확인할 수 없으므로 잠금은 다시 매핑 전체에서 지속됩니다. 파일은 변경되지 않아야 하므로 서버측에서는 파일을 잠글 필요가 없습니다.

## NFS 버전 4의 클라이언트측 파일오버

NFS 버전 4에서는 파일 크기나 파일 형식이 달라 복제를 설정할 수 없으면 다음과 같은 현상이 발생합니다.

- 파일이 사용 불가능으로 표시됩니다.
- 경고가 인쇄됩니다.
- 응용 프로그램에서 시스템 호출 오류를 받습니다.

---

주 - 응용 프로그램을 다시 시작한 후에 파일 액세스를 다시 시도하면 정상적으로 액세스할 수 있습니다.

---

NFS 버전 4에서는 크기가 다른 디렉토리에 대해 더 이상 복제 오류가 발생하지 않습니다. 이전 버전 NFS에서는 이러한 상황이 오류로 간주되어 다시 매핑 프로세스가 지연될 수 있습니다.

또한 NFS 버전 4에서는 디렉토리 읽기 작업이 실패하면 목록의 다음 서버에서 해당 작업을 수행합니다. 이전 버전 NFS에서는 읽기 작업이 실패하면 다시 매핑도 실패하고 원래 서버를 사용할 수 있을 때까지 프로세스가 정지되었습니다.

## 큰 파일

OS는 2GB보다 큰 파일을 지원합니다. 기본적으로 UFS 파일 시스템은 새 기능을 지원하기 위해 `-largefiles` 옵션을 사용하여 마운트됩니다. 필요한 경우 [88 페이지 “NFS 서버에서 큰 파일을 사용 안함으로 설정하는 방법”](#)에서 지침을 참조하십시오.

서버의 파일 시스템이 `-largefiles` 옵션을 사용하여 마운트된 경우 클라이언트는 변경을 수행하지 않아도 큰 파일에 액세스할 수 있습니다. 그러나 모든 명령이 이러한 큰 파일을 처리할 수 있는 것은 아닙니다. 큰 파일을 처리할 수 있는 명령 목록은 [largefile\(5\)](#)을 참조하십시오. 큰 파일 확장을 사용하는 NFS 버전 4 프로토콜을 지원할 수 없는 클라이언트는 큰 파일에 액세스할 수 없습니다.

## NFS 서버 로깅의 작동 방식

NFS 서버 로깅에서는 NFS 읽기 및 쓰기 레코드와 파일 시스템을 수정하는 작업을 제공합니다. 이 데이터를 사용하여 정보에 대한 액세스를 추적할 수 있습니다. 또한 레코드를 통해 정보에 대한 관심도를 수량적으로 측정할 수 있습니다.

로깅이 사용으로 설정된 파일 시스템에 액세스하면 커널에서 버퍼 파일에 원시(raw) 데이터를 씁니다. 이 데이터에는 다음이 포함됩니다.

- 시간 기록
- 클라이언트 IP 주소
- 요청자의 UID
- 액세스 중인 파일 또는 디렉토리 객체의 파일 핸들
- 수행한 작업의 유형

`nfslogd` 데몬은 이 원시(raw) 데이터를 로그 파일에 저장되는 ASCII 레코드로 변환합니다. 사용으로 설정된 이름 서비스에서 일치 항목을 찾을 수 있는 경우 변환 중에 IP 주소는 호스트 이름으로 수정되고 UID는 로그인으로 수정됩니다. 파일 핸들도 경로 이름으로 변환됩니다. 변환을 수행하기 위해 데몬은 파일 핸들을 추적하고 별도의 파일 핸들-경로 테이블에 정보를 저장합니다. 이러한 방식이 사용되므로 파일 핸들에 액세스할 때마다 경로를 다시 식별하지 않아도 됩니다. `nfslogd`를 해제한 경우에는 파일 핸들-경로 테이블에서 매핑이 변경되지 않으므로 데몬을 계속 실행해야 합니다.

---

주 - NFS 버전 4에서는 서버 로깅이 지원되지 않습니다.

---

## WebNFS 서비스의 작동 방식

WebNFS 서비스는 공개 파일 핸들을 사용하여 클라이언트가 디렉토리의 파일을 사용할 수 있도록 합니다. 파일 핸들은 NFS 클라이언트에 대해 파일을 식별하는 커널에서 생성하는 주소입니다. **공개 파일 핸들**에는 미리 정의된 값이 있으므로 서버에서 클라이언트용으로 파일 핸들을 생성하지 않아도 됩니다. 이 미리 정의된 파일 핸들을 사용할 수 있으므로 MOUNT 프로토콜을 사용할 필요가 없어 네트워크 트래픽이 감소합니다. 또한 이 기능을 통해 클라이언트의 프로세스 속도도 높일 수 있습니다.

기본적으로 NFS 서버의 공개 파일 핸들은 루트 파일 시스템에서 설정됩니다. 이러한 기본값이 사용되므로 WebNFS에서 이미 서버에 대한 마운트 권한이 있는 모든 클라이언트에 액세스할 수 있습니다. `share` 명령을 사용하면 임의의 파일 시스템을 가리키도록 공개 파일 핸들을 변경할 수 있습니다.

클라이언트에 파일 시스템용 파일 핸들이 있으면 LOOKUP이 실행되어 액세스할 파일의 파일 핸들을 확인합니다. NFS 프로토콜에서는 경로 이름 구성 요소를 한 번에 하나씩만 평가하도록 허용합니다. 각각의 추가 디렉토리 계층 레벨에는 다른 LOOKUP을 사용해야 합니다. LOOKUP이 공개 파일 핸들에 상대적이면 WebNFS 서버는 다중 구성 요소 조회

트랜잭션 하나를 사용하여 전체 경로 이름을 평가할 수 있습니다. WebNFS 서버에서는 다중 구성 요소 조회를 통해 경로 이름의 각 디렉토리 레벨에 대해 파일 핸들을 교환하지 않고도 파일 핸들을 원하는 파일로 배달할 수 있습니다.

또한 NFS 클라이언트는 단일 TCP 연결에 대한 동시 다운로드를 시작할 수 있습니다. 이 연결을 사용하면 여러 연결을 설정하면 발생하는 서버에 대한 추가 로드 없이 빠른 액세스가 가능합니다. 웹 브라우저 응용 프로그램은 여러 파일의 동시 다운로드를 지원하지만 각 파일에는 고유한 연결이 사용됩니다. WebNFS 소프트웨어에서는 연결 하나를 사용하므로 서버에 대한 오버헤드가 줄어듭니다.

경로 이름의 최종 구성 요소가 다른 파일 시스템에 대한 심볼릭 링크인 경우 정상 NFS 작업을 통해 이미 액세스 권한을 가지고 있는 클라이언트는 파일에 액세스할 수 있습니다.

일반적으로 NFS URL은 공개 파일 핸들에 상대적으로 평가됩니다. 경로 첫 부분에 슬래시를 더 추가하면 평가를 서버 루트 파일 시스템에 상대적으로 변경할 수 있습니다. 이 예제에서는 공용 파일 핸들이 `/export/ftp` 파일 시스템에서 설정된 경우 두 NFS URL은 동일합니다.

```
nfs://server/junk
nfs://server//export/ftp/junk
```

---

주 - WebNFS 서비스보다 NFS 버전 4 프로토콜이 기본적으로 사용됩니다. NFS 버전 4에는 MOUNT 프로토콜 및 WebNFS 서비스에 추가된 모든 보안 협상 기능이 완전하게 통합되어 있습니다.

---

## WebNFS 보안 협상의 작동 방식

NFS 서비스에는 WebNFS 클라이언트가 선택한 보안 방식을 WebNFS 서버와 협상할 수 있도록 하는 프로토콜이 포함되어 있습니다. 새로운 프로토콜은 이전 버전 WebNFS 프로토콜에서 사용되었던 다중 구성 요소 조회의 확장인 보안 협상 다중 구성 요소 조회를 사용합니다.

WebNFS 클라이언트는 공개 파일 핸들을 사용하여 정규 다중 구성 요소 조회 요청을 수행함으로써 프로세스를 시작합니다. 클라이언트는 서버에서 경로를 보호하는 방법을 알 수 없으므로 기본 보안 방식이 사용됩니다. 기본 보안 방식이 충분하지 않으면 서버는 AUTH\_TOOWEAK 오류로 회신합니다. 이 회신은 기본 방식이 유효하지 않음을 나타냅니다. 그러면 클라이언트는 보다 강력한 기본 방식을 사용해야 합니다.

클라이언트는 AUTH\_TOOWEAK 오류를 받으면 필요한 보안 방식을 결정하라는 요청을 서버로 보냅니다. 요청이 성공하면 서버는 지정된 경로에 필요한 보안 방식 배열로

응답합니다. 보안 방식 배열의 크기에 따라 클라이언트가 추가 요청을 통해 전체 배열을 얻어야 할 수도 있습니다. 서버에서 WebNFS 보안 협상을 지원하지 않으면 요청은 실패합니다.

요청이 성공하고 나면 WebNFS 클라이언트는 클라이언트가 지원하는 배열에서 첫번째 보안 방식을 선택합니다. 그런 다음 클라이언트는 선택한 보안 방식을 사용해 정규 다중 구성 요소 조회 요청을 실행하여 파일 핸들을 가져옵니다. 모든 후속 NFS 요청은 선택한 보안 방식 및 파일 핸들을 사용하여 수행됩니다.

---

주 - WebNFS 서비스보다 NFS 버전 4 프로토콜이 기본적으로 사용됩니다. NFS 버전 4에는 MOUNT 프로토콜 및 WebNFS 서비스에 추가된 모든 보안 협상 기능이 완전하게 통합되어 있습니다.

---

## 웹 브라우저 사용 시의 WebNFS 제한

WebNFS 소프트웨어는 HTTP를 사용하는 웹 사이트에서 제공할 수 있는 여러 기능을 지원하지 않습니다. 이와 같이 기능에 차이가 있는 것은, NFS 서버는 파일을 보내기만 하므로 특수한 처리는 클라이언트에서 수행해야 하기 때문입니다. WebNFS와 HTTP 액세스에 대해 웹 사이트 하나를 구성해야 하는 경우 다음 문제를 고려하십시오.

- NFS 검색에서는 CGI 스크립트가 실행되지 않습니다. 따라서 CGI 스크립트를 많이 사용하는 활성 사이트가 포함된 파일 시스템은 NFS 검색용으로 적합하지 않을 수 있습니다.
- 브라우저에서 다른 파일 형식의 파일을 처리하기 위해 다른 뷰어가 시작될 수 있습니다. NFS URL을 통해 이러한 파일에 액세스하는 경우 파일 이름으로 파일 유형을 확인할 수 있으면 외부 브라우저가 시작됩니다. NFS URL을 사용하는 경우 브라우저는 표준 MIME 유형의 파일 이름 확장을 인식해야 합니다. WebNFS 소프트웨어는 파일 유형을 확인하기 위해 파일 내부를 확인하지 않습니다. 따라서 파일 이름 확장자를 통해서만 파일 유형을 확인할 수 있습니다.
- NFS 검색에서는 서버측 이미지 맵(클릭 가능한 이미지)을 사용할 수 없습니다. 그러나 URL이 위치와 함께 정의되므로 NFS 검색에서 클라이언트측 이미지 맵(클릭 가능한 이미지)은 사용할 수 있습니다. 문서 서버로부터의 추가 응답은 필요하지 않습니다.

## 보안 NFS 시스템

NFS 환경을 사용하면 서로 다른 여러 컴퓨터 구조와 운영 체제의 네트워크에서 파일 시스템을 효율적이고도 편리하게 공유할 수 있습니다. 그러나 NFS 작업을 통해 파일 시스템을 공유하는 동일 기능은 편리하기는 하지만 몇 가지 보안 문제를 야기합니다. 기존에는 대부분의 NFS 구현에서 UNIX 또는 AUTH\_SYS 인증이 사용되었지만, AUTH\_DH와 같은 보다 강력한 인증 방법도 사용할 수 있었습니다. UNIX 인증을

사용하는 경우 NFS 서버는 사용자가 아닌 파일 요청을 하는 컴퓨터를 인증하여 해당 요청을 인증합니다. 따라서 클라이언트 사용자는 su를 실행하여 파일 소유자를 가장할 수 있습니다. DH 인증을 사용하는 경우에는 NFS 서버에서 사용자를 인증하므로 이러한 종류의 가장이 훨씬 어려워집니다.

루트 액세스 권한과 네트워크 프로그래밍에 대한 지식이 있는 모든 사람이 임의의 데이터를 네트워크로 가져와서 네트워크에서 데이터를 추출할 수 있습니다. 이와 같이 데이터를 가져오는 공격이 가장 위험한 공격입니다. 올바른 패킷을 생성하거나 "대화"를 기록했다가 나중에 재생하여 사용자를 가장하는 경우를 예로 들 수 있습니다. 이러한 공격은 데이터 무결성에 영향을 줍니다. 수동적 도청(사용자를 가장하지는 않고 네트워크 트래픽만 수신함)을 포함하는 공격의 경우에는 데이터 무결성이 손상되지 않으므로 그만큼 위험하지 않습니다. 사용자는 네트워크를 통해 보내는 데이터를 암호화하여 중요한 정보의 프라이버시를 보호할 수 있습니다.

네트워크 보안 문제에 대한 일반적인 해결 방식은 각 응용 프로그램에서 문제를 해결하도록 하는 것입니다. 이보다 효율적인 방식은 모든 응용 프로그램이 포함되는 레벨에서 표준 인증 시스템을 구현하는 것입니다.

Oracle Solaris 운영 체제의 RPC(원격 프로시저 호출) 레벨에는 인증 시스템이 포함되어 있으며, 이 시스템은 NFS 작업의 기반이 되는 방식입니다. 보안 RPC라고도 하는 이 시스템은 네트워크 환경의 보안을 크게 개선하며 NFS 시스템과 같은 서비스에 추가적인 보안 기능을 제공합니다. 보안 RPC에서 제공하는 기능을 사용하는 NFS 시스템을 보안 NFS 시스템이라고 합니다.

## 보안 RPC

보안 RPC는 보안 NFS 시스템의 기반이 되는 요소입니다. 보안 RPC는 최소한 시간 공유 시스템 레벨의 보안이 유지되는 시스템을 작성하는 데 사용됩니다. 시간 공유 시스템에서는 모든 사용자가 컴퓨터 한 대를 공유합니다. 시간 공유 시스템은 로그인 암호를 통해 사용자를 인증합니다. 데이터 암호화 표준(DES) 인증에서는 동일한 인증 프로세스가 완료됩니다. 사용자는 로컬 터미널에 로그인하는 것처럼 원격 컴퓨터에 로그인할 수 있습니다. 사용자의 로그인 암호는 네트워크 보안을 적용하는 수단입니다. 시간 공유 환경에서 시스템 관리자에게는 암호를 변경하여 사용자를 가장하지 않아야 한다는 윤리적 책임이 있습니다. 보안 RPC에서 네트워크 관리자는 **공개 키**가 저장되는 데이터베이스의 항목을 변경하지 않는 것으로 신뢰됩니다.

RPC 인증 시스템을 이해하려면 자격 증명과 검증기의 두 가지 용어를 숙지하고 있어야 합니다. ID 배치를 예로 사용하는 경우 자격 증명은 이름, 주소, 생일 등 사용자를 식별하는 수단입니다. 검증기는 배지에 첨부된 사진입니다. 배지의 사진을 배지 소유자와 비교 확인하여 배지가 도용되는 것이 아닌지를 확인할 수 있습니다. RPC에서 클라이언트 프로세스는 자격 증명과 검증기를 모두 각 RPC 요청과 함께 서버로 보냅니다. 클라이언트가 서버의 자격 증명을 이미 알고 있으므로 서버는 검증기만 다시 보냅니다.



RPC의 인증은 개방형이므로 UNIX, DH, KERB 등의 다양한 인증 시스템을 연결할 수 있습니다.

네트워크 서비스에서 UNIX 인증을 사용하는 경우 자격 증명에는 클라이언트의 호스트 이름, UID, GUID 및 그룹 액세스 목록이 포함됩니다. 그러나 검증기는 아무런 작업을 수행하지 않습니다. 이처럼 검증기가 없으므로 슈퍼 유저는 su와 같은 명령을 사용하여 적절한 자격 증명을 위조할 수 있습니다. UNIX 인증의 또 다른 문제는 네트워크의 모든 컴퓨터가 UNIX 컴퓨터라고 가정하는 것입니다. 이기종 네트워크의 다른 운영 체제에 적용하는 경우 UNIX 인증은 손상됩니다.

UNIX 인증의 문제를 해결하기 위해 보안 RPC는 DH 인증을 사용합니다.

## DH 인증

DH 인증은 데이터 암호화 표준(DES) 및 Diffie-Hellman 공개 키 암호화를 사용하여 네트워크의 사용자 및 컴퓨터를 모두 인증합니다. DES는 표준 암호화 방식입니다. Diffie-Hellman 공개 키 암호화는 두 개의 키(공개 키 하나, 보안 키 하나)가 포함된 암호화 시스템입니다. 공개 키와 보안 키는 이름 공간에 저장됩니다. NIS는 공개 키 맵에 키를 저장합니다. 이러한 맵에는 모든 잠재적 사용자에 대한 공개 키 및 보안 키가 포함됩니다. 맵을 설정하는 방법에 대한 자세한 내용은 **Oracle Solaris Administration: Naming and Directory Services** 를 참조하십시오.

DH 인증의 보안은 보낸 사람이 현재 시간을 암호화하는 기능을 기반으로 합니다. 그러면 받는 사람이 시간을 암호 해독하여 자신의 시계와 비교 확인할 수 있습니다. 시간 기록은 DES를 통해 암호화됩니다. 이 체계가 작동하는 데 필요한 요구 사항은 다음과 같습니다.

- 두 에이전트가 현재 시간에 동의해야 합니다.
- 보낸 사람과 받는 사람이 같은 암호화 키를 사용 중이어야 합니다.

네트워크에서 시간 동기화 프로그램을 실행하는 경우 클라이언트와 서버의 시간이 자동으로 동기화됩니다. 시간 동기화 프로그램을 사용할 수 없는 경우에는 네트워크 시간이 아닌 서버의 시간을 사용하여 시간 기록을 계산할 수 있습니다. 클라이언트는 RPC 세션을 시작하기 전에 서버에 시간을 물은 다음 자체 시계와 서버 시계 간의 시간 차이를 계산합니다. 이 차이를 사용하여 시간 기록 계산 시 클라이언트 시계를 오프셋합니다. 클라이언트 및 서버의 시계가 동기화되지 않은 상태이면 서버는 클라이언트 요청을 거부합니다. 클라이언트의 DH 인증 시스템은 서버와 재동기화됩니다.

클라이언트와 서버는 임의의 **대화 키**(**세션 키**라고도 함)를 생성하고 공개 키 암호화를 통해 **공통 키**를 추론함으로써 동일한 암호화 키를 사용하게 됩니다. 공통 키는 클라이언트와 서버만 추론할 수 있는 키입니다. 대화 키는 클라이언트 시간 기록을 암호화하고 암호 해독하는 데 사용됩니다. 공통 키는 대화 키를 암호화하고 암호 해독하는 데 사용됩니다.

## KERB 인증

Kerberos는 MIT에서 개발된 인증 시스템입니다. Kerberos는 DES를 비롯한 여러 암호화 유형을 제공합니다. Kerberos 지원은 더 이상 보안 RPC의 일부분으로 제공하지 않지만, 서버측 및 클라이언트측 구현은 릴리스에 포함되어 있습니다. Kerberos 인증 구현에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 19 장, “Kerberos 서비스 소개”](#)를 참조하십시오.

## NFS에서 보안 RPC 사용

보안 RPC를 사용하려는 경우 다음 사항에 유의하십시오.

- 정전 이후 등과 같이 사용자가 없는데 서버가 충돌하는 경우 시스템에 저장된 모든 보안 키가 삭제됩니다. 그러면 프로세스에서 보안 네트워크 서비스에 액세스하거나 NFS 파일 시스템을 마운트할 수 없습니다. 재부트 중의 중요한 프로세스는 보통 root로 실행됩니다. 따라서 루트의 보안 키를 외부에 저장한 경우에는 이러한 프로세스가 작동하지만 해당 키의 암호를 해독하는 암호를 입력할 사용자가 없기 때문에 문제가 됩니다. 이 경우 `keylogin -r`은 root를 통해 일반 보안 키를 `/etc/.rootkey`(`keyserv`에서 읽을 수 있음)에 저장하도록 허용합니다.
- 일부 시스템은 단일 사용자 모드로 부트됩니다. 이때 콘솔의 루트 로그인 셸이 사용되며 암호 프롬프트는 표시되지 않습니다. 이러한 경우에는 물리적 보안을 반드시 적용해야 합니다.
- 디스크가 없는 컴퓨터 부트는 완벽하게 안전하지 않습니다. 즉, 누군가가 부트 서버를 가장해 우회 커널을 부트하여 원격 컴퓨터에서 보안 키 레코드를 만드는 등의 작업을 수행할 수 있습니다. 보안 NFS 시스템에서는 커널 및 키 서버를 실행한 후에만 보호 기능을 제공합니다. 그 외에는 부트 서버에서 제공하는 회신을 인증할 방법이 없습니다. 이러한 제한은 심각한 문제가 될 수 있지만 이 제한으로 인해 커널 소스 코드를 사용한 복잡한 공격을 수행해야 합니다. 또한 범죄의 증거가 남게 됩니다. 부트 서버에 대해 네트워크를 폴링한 경우 우회 부트 서버 위치를 확인할 수 있습니다.
- 대부분의 `setuid` 프로그램은 root에서 소유합니다. root의 보안 키가 `/etc/.rootkey`에 저장되는 경우 이러한 프로그램은 이전과 동일하게 동작합니다. 그러나 사용자 소유의 `setuid` 프로그램은 작동하지 않을 수도 있습니다. 예를 들어 `setuid` 프로그램을 dave가 소유하고 있으며 dave가 컴퓨터 부트 후 컴퓨터에 로그인하지 않았다고 가정하겠습니다. 그러면 프로그램에서 보안 네트워크 서비스에 액세스할 수 없습니다.
- `login`, `rlogin`, 또는 `telnet`을 사용하여 원격 컴퓨터에 로그인하고 `keylogin`을 사용하여 액세스 권한을 얻는 경우에는 계정 액세스 권한이 부여됩니다. 보안 키가 해당 컴퓨터의 키 서버로 전달되며, 이 서버에 보안 키가 저장되기 때문입니다. 이 프로세스는 원격 컴퓨터를 신뢰하지 않는 경우에만 문제가 됩니다. 의심스러운 부분이 있는 경우 원격 컴퓨터에 암호가 필요하다면 원격 컴퓨터에 로그인하지



마십시오. 대신 NFS 환경을 사용하여 원격 컴퓨터에서 공유하는 파일 시스템을 마운트합니다. 또한 `keylogout`을 사용하여 키 서버에서 보안 키를 삭제할 수도 있습니다.

- `-o sec=dh` 옵션을 사용하여 홈 디렉토리를 공유하는 경우 원격 로그인 시 문제가 발생할 수 있습니다. 암호 프롬프트를 표시하도록 `/etc/hosts.equiv` 또는 `~/.rhosts` 파일이 설정되어 있지 않으면 로그인은 성공합니다. 그러나 로컬에서 인증이 수행되지 않았으므로 사용자는 홈 디렉토리에 액세스할 수 없습니다. 사용자에게 암호 프롬프트가 표시되는 경우 암호가 네트워크 암호와 일치하면 사용자는 홈 디렉토리 액세스 권한을 얻습니다.

## 미러 마운트의 작동 방식

Oracle Solaris 릴리스에는 미러 마운트라는 새로운 마운트 기능이 포함되어 있습니다. 미러 마운트를 사용하면 NFSv4 서버에서 파일 시스템을 공유하는 즉시 NFSv4 클라이언트가 해당 파일 시스템의 파일에 액세스할 수 있습니다. 즉, 마운트 명령을 사용하거나 `autofs` 맵을 업데이트하는 오버헤드 없이도 파일에 액세스할 수 있습니다. 따라서 특정 NFSv4 파일 시스템을 클라이언트에 마운트하고 나면 해당 서버의 다른 파일 시스템도 마운트할 수 있습니다.

## 미러 마운트를 사용하는 경우

일반적으로 미러 마운트 기능은 NFSv4 클라이언트에서 최적 상태로 사용할 수 있습니다. 단, 다음 경우는 예외입니다.

- 클라이언트에서 서버에 없는 다른 계층을 사용해야 하는 경우
- 상위 파일 시스템과 다른 마운트 옵션을 사용해야 하는 경우

## 미러 마운트를 사용하여 파일 시스템 마운트

NFSv4 클라이언트에서 파일 시스템이 수동 마운트 또는 `autofs`를 사용하여 마운트된 경우 마운트된 파일 시스템에 더 추가하는 파일 시스템은 미러 마운트 기능을 사용하여 클라이언트에 마운트할 수 있습니다. 클라이언트는 상위 디렉토리에서 사용된 것과 같은 마운트 옵션을 사용하여 새 파일 시스템에 대한 액세스 권한을 요청합니다. 마운트가 실패하면 서버와 클라이언트 간에 일반 NFSv4 보안 협상이 수행되어 마운트 요청이 성공하도록 마운트 옵션을 조정합니다.

특정 서버 파일 시스템에 대해 기존 자동 마운트 트리거 지점이 설정되어 있는 경우 자동 마운트 트리거가 미러 마운트보다 우선적으로 사용되므로 해당 파일 시스템에 대해 미러 마운트가 수행되지 않습니다. 이 경우 미러 마운트를 사용하려면 자동 마운트 항목을 제거해야 합니다.

Oracle Solaris 11 릴리스에서는 `/net` 또는 `/home` 자동 마운트 지점에 액세스하면 `/net` 또는 `/home` 서버 이름 공간이 마운트됩니다. 이러한 디렉토리 아래의 디렉토리나 파일에 대한 액세스 권한도 미리 마운트 기능을 통해 제공됩니다.

미러 마운트가 작동하도록 하는 방법에 대한 특정 지침은 다음 항목을 참조하십시오.

- [Using Mirrormounts After Mounting a File System](#)
- [87 페이지 “서버에서 모든 파일 시스템을 마운트하는 방법”](#)

## 미러 마운트를 사용하여 파일 시스템 마운트 해제

미러 마운트된 파일 시스템은 일정 기간 동안 작업을 수행하지 않는 유휴 상태인 경우 자동으로 마운트 해제됩니다. 이 기간은 `timeout` 매개변수를 사용하여 설정합니다(자동 마운트에서도 동일한 용도로 사용됨).

NFS 파일 시스템을 수동으로 마운트 해제하는 경우 해당 시스템에 포함된 미러 마운트된 파일 시스템도 유휴 상태 시 마운트 해제됩니다. 활성 미러 마운트된 파일 시스템이 포함되어 있으면 수동 마운트 해제는 원본 파일 시스템이 사용 중일 때와 마찬가지로 실패합니다. 그러나 강제 마운트 해제는 포함된 모든 미러 마운트 파일 시스템을 통해 전파됩니다.

자동 마운트된 파일 시스템 내에서 파일 시스템 경계가 발견되면 미러 마운트가 수행됩니다. 자동 마운트에서 상위 파일 시스템을 마운트 해제하면 해당 파일 시스템 내의 미러 마운트된 파일 시스템도 유휴 상태 시 자동으로 마운트 해제됩니다. 활성 미러 마운트된 파일 시스템이 있으면 자동 마운트 해제는 수행되지 않으므로 현재 자동 마운트 동작이 유지됩니다.

## NFS 참조의 작동 방식

Oracle Solaris 11 릴리스에는 NFS 참조라는 새로운 NFS 기능이 포함되어 있습니다. NFSv4 서버에서는 NFS 참조를 사용하여 다른 NFSv4 서버에 있는 파일 시스템을 가리킵니다. 이러한 방법으로 여러 NFSv4 서버를 통합 네임스페이스로 연결할 수 있습니다.

NFSv2, NFSv3 및 기타 클라이언트도 참조를 따를 수 있습니다. 이러한 클라이언트에게는 참조가 심볼릭 링크로 표시되기 때문입니다.

## NFS 참조를 사용하는 경우

NFS 참조는 여러 서버에 대해 단일 파일 이름 세트로 표시되는 항목을 만들되 `autofs`는 사용하지 않으려는 경우에 유용합니다. 참조를 호스트하려면 NFSv4 서버만 사용할 수 있으며 서버에서는 Oracle Solaris 11 릴리스 이상을 실행해야 합니다.

## NFS 참조 만들기

nfsref 명령을 사용하여 NFS 참조를 만듭니다. 참조를 만들 때 마운트 지점이 아직 없으면 객체를 구문 재분석 지점으로 식별하는 특수 플래그가 포함된 심볼릭 링크가 생성됩니다. 구문 재분석 지점이 이미 있으면 경우에 따라 NFS 서비스 데이터가 추가되거나 기존 NFS 서비스 데이터가 바뀝니다.

## NFS 참조 제거

NFS 참조를 제거할 때도 nfsref 명령을 사용합니다. 이 명령을 실행하면 지정된 구문 재분석 지점에서 NFS 서비스 데이터를 제거하고, 다른 유형의 서비스 데이터가 없으면 구문 재분석 지점을 제거합니다.

## autofs 맵

autofs에서는 세 가지 유형의 맵을 사용합니다.

- 마스터 맵
- 직접 맵
- 간접 맵

## 마스터 autofs 맵

auto\_master 맵은 디렉토리를 맵과 연관시킵니다. 이 맵은 autofs에서 확인해야 하는 모든 맵을 지정하는 마스터 목록입니다. 다음 예에서는 auto\_master 파일에 포함될 수 있는 항목을 보여줍니다.

예 6-3 샘플 /etc/auto\_master 파일

```
# Master map for automounter
#
+auto_master
/net          -hosts          -nosuid,nobrowse
/home         auto_home    -nobrowse
/nfs4         -fedfs           -ro,nosuid,nobrowse
/-           auto_direct   -ro
```

이 예에서는 auto\_direct 맵이 하나 추가된 일반 auto\_master 파일을 보여줍니다. 마스터 맵 /etc/auto\_master의 각 행 구문은 다음과 같습니다.

*mount-point map-name [ mount-options ]*

<i>mount-point</i>	<i>mount-point</i> 는 디렉토리의 전체(절대) 경로 이름입니다. 디렉토리가 없으면 autofs에서 가능한 경우 디렉토리를 만듭니다. 디렉토리가 있으며 비어 있지 않은 경우 해당 디렉토리에 마운트하면 콘텐츠가 숨겨집니다. 이 경우에는 autofs에서 경고가 표시됩니다.  /-(마운트 지점)의 표기법은 해당 특정 맵이 직접 맵임을 나타냅니다. 또한 이 표기법은 특정 마운트 지점이 맵에 연관되어 있지 않음을 나타냅니다.
<i>map-name</i>	<i>map-name</i> 은 autofs에서 위치에 대한 방향이나 마운트 정보를 찾기 위해 사용하는 맵입니다. 이름 앞에 슬래시(/)가 붙으면 autofs에서는 해당 이름을 로컬 파일로 해석합니다. 그렇지 않으면 autofs는 이름 서비스 스위치 구성 파일(/etc/nsswitch.conf)에 지정된 검색을 사용하여 마운트 정보를 검색합니다. 특수 맵은 /net에도 사용됩니다. 자세한 내용은 <a href="#">197 페이지 “마운트 지점 /net”</a> 을 참조하십시오.
<i>mount-options</i>	<i>mount-options</i> 는 선택적인 쉘표로 구분된 옵션 목록입니다. 이 목록에 포함된 옵션은 맵 이름의 항목에 다른 옵션이 나열된 경우를 제외하면 맵 이름에 지정된 항목 마운트 시에 적용됩니다. 각각의 특정 파일 시스템 유형에 대한 옵션은 해당 파일 시스템의 마운트 매뉴얼 페이지에 나열됩니다. 예를 들어 NFS 관련 마운트 옵션은 <a href="#">mount_nfs(1M)</a> 매뉴얼 페이지를 참조하십시오. NFS 관련 마운트 지점의 경우 bg(백그라운드) 및 fg(전경) 옵션이 적용되지 않습니다.

#로 시작하는 행은 주석입니다. 해당 행이 끝날 때까지 표시되는 모든 텍스트는 무시됩니다.

긴 행을 짧게 분할하려면 행 끝에 백슬래시(\)를 추가합니다. 항목당 최대 문자 수는 1024자입니다.

---

주 - 두 항목에서 같은 마운트 지점이 사용되는 경우 첫번째 항목은 automount 명령에서 사용됩니다. 두번째 항목은 무시됩니다.

---

## 마운트 지점 /home

/home 마운트 지점은 /etc/auto\_home (간접 맵)에 나열된 항목을 마운트할 디렉토리입니다.

---

주 - autofs는 모든 컴퓨터에서 실행되며 /net 및 /home(자동 마운트된 홈 디렉토리)을 기본적으로 지원합니다. 이러한 기본값은 NIS auto.master 맵의 항목을 사용하거나 /etc/auto\_master 파일을 로컬로 편집하여 대체할 수 있습니다.

---

## 마운트 지점 /net

autofs는 특수 맵 -hosts의 모든 항목을 /net 디렉토리 아래에 마운트합니다. 이 맵은 호스트 데이터베이스만 사용하는 내장 맵입니다. gumbo 컴퓨터가 호스트 데이터베이스에 있고 해당 파일 시스템을 내보낸다고 가정하겠습니다. 다음 명령은 현재 디렉토리를 gumbo 컴퓨터의 루트 디렉토리로 변경합니다.

```
% cd /net/gumbo
```

autofs는 gumbo 호스트의 **내보낸** 파일 시스템만 마운트할 수 있습니다. 즉, 로컬 디스크의 파일 시스템이 아닌 네트워크 사용자에게 제공되는 서버의 파일 시스템만 마운트할 수 있습니다. 따라서 gumbo의 모든 파일 및 디렉토리는 /net/gumbo를 통해 사용할 수 없습니다.

/net 액세스 방법을 사용하는 경우 서버 이름은 경로에 포함되며 위치에 따라 달라집니다. 내보낸 파일 시스템을 서버 간에 이동하는 경우에는 경로가 더 이상 작동하지 않을 수 있습니다. 따라서 /net을 사용하는 대신 원하는 파일 시스템 전용으로 맵에 항목을 설정해야 합니다.

---

**주 -** autofs는 마운트 시에만 서버의 내보내기 목록을 확인합니다. 서버의 파일 시스템이 마운트되고 나면 autofs는 서버의 파일 시스템이 자동으로 마운트 해제될 때까지 서버를 다시 확인하지 않습니다. 따라서 새로 내보낸 파일 시스템은 클라이언트의 파일 시스템을 마운트 해제했다가 다시 마운트할 때까지는 표시되지 않습니다.

---

## 마운트 지점 /nfs4

/nfs4 마운트 지점은 의사 맵을 사용하여 통합 파일 시스템 도메인 루트를 마운트합니다. /nfs4/example.net을 참조하는 경우 DNS 도메인 example.net의 도메인 루트를 찾아 해당 위치에서 마운트하려고 시도합니다. 이렇게 하려면 DNS 서버가 [91 페이지 “통합 파일 시스템 서버에 대해 DNS 레코드 설정”](#)에 설명된 레코드를 반환해야 합니다.

## 직접 autofs 맵

직접 맵은 자동 마운트 지점입니다. 직접 맵을 사용하는 경우 클라이언트의 마운트 지점과 서버의 디렉토리가 직접 연관됩니다. 직접 맵은 전체 경로 이름을 포함하며 관계를 명시적으로 나타냅니다. 다음은 일반적인 /etc/auto\_direct 맵입니다.

```
/usr/local      -ro \
    /bin          ivy:/export/local/sun4 \
    /share        ivy:/export/local/share \
    /src          ivy:/export/local/src
/usr/man        -ro
    oak:/usr/man \
    rose:/usr/man \
    willow:/usr/man
/usr/games      -ro peach:/usr/games
```

```
/usr/spool/news    -ro    pine:/usr/spool/news \
                    willow:/var/spool/news
```

직접 맵의 행 구문은 다음과 같습니다.

*key* [ *mount-options* ] *location*

*key*                    *key*는 직접 맵의 마운트 지점 경로 이름입니다.

*mount-options*        *mount-options*은 해당 특정 마운트에 적용할 옵션입니다. 이러한 옵션은 맵 기본값과 다른 경우에만 필요합니다. 각각의 특정 파일 시스템 유형에 대한 옵션은 해당 파일 시스템의 마운트 매뉴얼 페이지에 나열됩니다. 예를 들어 NFS 관련 마운트 옵션은 [mount\\_nfs\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

*location*              *location*은 파일 시스템의 위치입니다. NFS 파일 시스템에 대해 하나 이상의 파일 시스템이 *server:pathname*으로 지정됩니다.

---

주 - *pathname*에는 자동 마운트된 마운트 지점을 포함할 수 없습니다. *pathname*은 파일 시스템에 대한 실제 절대 경로여야 합니다. 예를 들어 홈 디렉토리의 위치는 *server:/home/username*이 아닌 *server:/export/home/username*으로 나열되어야 합니다.

---

마스터 맵에서와 마찬가지로, #로 시작하는 행은 주석입니다. 해당 행이 끝날 때까지 표시되는 모든 텍스트는 무시됩니다. 긴 행을 짧은 행으로 분할하려면 행 끝에 백슬래시를 추가합니다.

모든 맵 중 직접 맵의 항목이 */etc/vfstab*의 해당 항목과 가장 비슷합니다. 항목은 */etc/vfstab*에서 다음과 같이 표시될 수 있습니다.

```
dancer:/usr/local - /usr/local/tmp nfs - yes ro
```

직접 맵에서는 동일한 항목이 다음과 같이 표시됩니다.

```
/usr/local/tmp    -ro    dancer:/usr/local
```

---

주 - 자동 마운트 맵 간에 옵션 연결은 수행되지 않습니다. 자동 마운트 맵에 추가되는 모든 옵션은 이전에 검색한 맵에 나열된 모든 옵션을 대체합니다. 예를 들어 *auto\_master* 맵에 포함된 옵션은 다른 맵의 해당하는 항목으로 대체됩니다.

---

이 맵 유형과 연관된 기타 주요 기능은 [205 페이지](#) “*autofs*에서 클라이언트에 대해 가장 가까운 읽기 전용 파일을 선택하는 방법(여러 위치)”을 참조하십시오.

## 마운트 지점 /-

예 6-3에서 /-는 `auto_direct`의 항목을 특정 마운트 지점과 연관시키지 않도록 `autofs`에 지시합니다. 간접 맵에서는 `auto_master` 파일에 정의된 마운트 지점을 사용합니다. 직접 맵은 명명된 맵에 지정되어 있는 마운트 지점을 사용합니다. 직접 맵에서는 키(마운트 지점)가 전체 경로 이름입니다.

`NISauto_master` 파일은 직접 맵 항목을 하나만 포함할 수 있습니다. 마운트 지점이 이름 공간에서 고유한 값이어야 하기 때문입니다. `auto_master` 파일(로컬 파일)은 직접 맵 항목을 수에 제한 없이 포함할 수 있습니다(항목이 중복되지 않는 경우).

## 간접 autofs 맵

간접 맵은 키의 대체 값을 사용하여 클라이언트의 마운트 지점과 서버의 디렉토리 간 연관을 설정합니다. 간접 맵은 홈 디렉토리 등 특정 파일 시스템에 액세스하는 데 유용합니다. 간접 맵의 예로는 `auto_home` 맵을 들 수 있습니다.

간접 맵의 행에서는 일반적으로 다음과 같은 구문을 사용합니다.

*key* [ *mount-options* ] *location*

*key*                      *key*는 간접 맵의 간단한 이름(슬래시 없음)입니다.

*mount-options*        *mount-options*은 해당 특정 마운트에 적용할 옵션입니다. 이러한 옵션은 맵 기본값과 다른 경우에만 필요합니다. 각각의 특정 파일 시스템 유형에 대한 옵션은 해당 파일 시스템의 마운트 매뉴얼 페이지에 나열됩니다. 예를 들어 NFS 관련 마운트 옵션은 `mount_nfs(1M)` 매뉴얼 페이지를 참조하십시오.

*location*                *location*은 파일 시스템의 위치입니다. 하나 이상의 파일 시스템이 `server:pathname`으로 지정됩니다.

---

주 - *pathname*에는 자동 마운트된 마운트 지점을 포함할 수 없습니다. *pathname*은 파일 시스템에 대한 실제 절대 경로여야 합니다. 예를 들어 디렉토리의 위치는 `server:/net/server/usr/local`이 아닌 `server/usr/local`로 나열되어야 합니다.

---

마스터 맵에서와 마찬가지로, #로 시작하는 행은 주석입니다. 해당 행이 끝날 때까지 표시되는 모든 텍스트는 무시됩니다. 긴 행을 짧은 행으로 분할하려면 행 끝에 백슬래시(\)를 추가합니다. 예 6-3에는 다음 항목이 포함된 `auto_master` 맵이 나와 있습니다.

```
/home      auto_home      -nobrowse
```

auto\_home은 /home 아래에 마운트할 항목이 포함된 간접 맵의 이름입니다. 일반적인 auto\_home 맵에는 다음 항목이 포함됩니다.

```
david          willow:/export/home/david
rob            cypress:/export/home/rob
gordon         poplar:/export/home/gordon
rajan          pine:/export/home/rajan
tammy          apple:/export/home/tammy
jim            ivy:/export/home/jim
linda         -rw,nosuid peach:/export/home/linda
```

예를 들어 이전 맵이 oak 호스트에 있다고 가정해 보겠습니다. linda 사용자의 암호 데이터베이스에 홈 디렉토리를 /home/linda로 지정하는 항목이 있다고 가정하겠습니다. linda가 oak 컴퓨터에 로그인할 때마다 autofs는 peach 컴퓨터에 있는 /export/home/linda 디렉토리를 마운트합니다. 홈 디렉토리는 읽기/쓰기(nosuid)로 마운트됩니다.

다음으로 사용자 linda의 홈 디렉토리가 암호 데이터베이스에 /home/linda로 포함되어 있다고 가정해 보겠습니다. Linda를 비롯한 모든 사용자는 이전 예제의 맵을 참조하는 마스터 맵을 사용하여 설정된 컴퓨터에서 이 경로에 액세스할 수 있습니다.

이와 같은 조건 하에서 사용자 linda는 이러한 모든 컴퓨터에서 login 또는 rlogin을 실행할 수 있으며 홈 디렉토리를 대신 마운트하도록 할 수 있습니다.

또한 이제 Linda는 다음 명령을 입력할 수도 있습니다.

```
% cd ~david
```

그러면 autofs는 David의 홈을 Linda 대신 마운트합니다(모든 권한이 허용하는 경우).

---

주 - 자동 마운트 맵 간에 옵션 연결은 수행되지 않습니다. 자동 마운트 맵에 추가되는 모든 옵션은 이전에 검색한 맵에 나열된 모든 옵션을 대체합니다. 예를 들어 auto\_master 맵에 포함된 옵션은 다른 맵의 해당하는 항목으로 대체됩니다.

---

이름 서비스가 없는 네트워크에서는 Linda가 파일에 액세스하도록 하려면 네트워크의 모든 시스템에서 /etc/passwd 등의 관련 파일을 모두 변경해야 합니다. NIS를 사용하는 경우 NIS 마스터 서버에서 변경을 수행하고 관련 데이터베이스를 슬레이브 서버로 전파합니다.



## autofs의 작동 방식

autofs는 해당하는 파일 시스템을 자동으로 마운트하는 클라이언트측 서비스입니다. 자동 마운트를 위해 함께 작동하는 구성 요소는 다음과 같습니다.

- automount 명령
- autofs 파일 시스템
- automountd 데몬

자동 마운트 서비스 `svc:/system/filesystem/autofs`(시스템 시작 시 호출됨)는 마스터 맵 파일 `auto_master`를 읽어 초기 autofs 마운트 세트를 만듭니다. 이러한 autofs 마운트는 시작 시 자동으로 마운트되지 않습니다. 이러한 마운트는 이후에 파일 시스템에 마운트되는 지점입니다. 이러한 지점은 트리거 노드라고도 합니다.

autofs 마운트를 설정하고 나면 이러한 마운트가 하위에 마운트할 파일 시스템을 트리거할 수 있습니다. 예를 들어 autofs는 현재 마운트되어 있지 않은 파일 시스템 액세스 요청을 받으면 automountd를 호출하며, 그러면 automountd가 요청된 파일 시스템을 자동으로 마운트합니다.

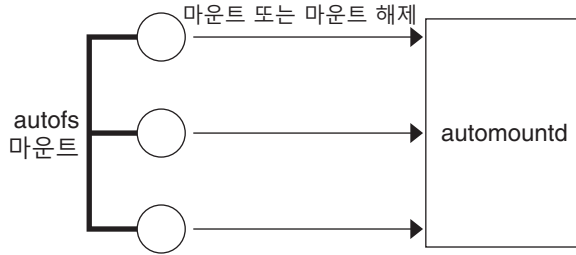
autofs 마운트를 초기 마운트한 후에 automount 명령을 사용하여 필요에 따라 autofs 마운트를 업데이트합니다. 이 명령은 `auto_master` 맵의 마운트 목록을 마운트 테이블 파일 `/etc/mnttab`(이전의 `/etc/mtab`)에 있는 마운트된 파일 시스템 목록과 비교합니다. 그리고 나면 automount가 적절한 변경을 수행합니다. 시스템 관리자는 이 프로세스를 통해 `auto_master` 내의 마운트 정보를 변경하며, autofs 데몬을 중지했다가 다시 시작하지 않고도 autofs 프로세스에서 해당 변경 내용을 사용할 수 있도록 합니다. 파일 시스템을 마운트한 이후에는 해당 파일 시스템이 자동으로 마운트 해제되기 전까지 후속 액세스 시에 automountd가 작업을 수행하지 않아도 됩니다.

mount와는 달리 automount는 `/etc/vfstab` 파일(각 컴퓨터마다 다름)에서 마운트할 파일 시스템 목록을 읽지 않습니다. automount 명령은 도메인 내와 컴퓨터에서 이름 공간 또는 로컬 파일을 통해 제어됩니다.

아래에는 autofs의 간단한 작동 방식 개요가 나와 있습니다.

자동 마운트 데몬 automountd는 `svc:/system/filesystem/autofs` 서비스를 통해 부트 시에 시작됩니다. [그림 6-3](#)을 참조하십시오. 이 서비스는 마스터 맵을 읽고 autofs 마운트 지점을 설치하는 automount 명령도 실행합니다. 자세한 내용은 [203 페이지 “autofs에서 탐색 프로세스를 시작하는 방법\(마스터 맵\)”](#)을 참조하십시오.

그림 6-3 svc:/system/filesystem/autofs 서비스 시작 automount



autofs는 자동 마운트 및 마운트 해제를 지원하는 커널 파일 시스템입니다.

autofs 마운트 지점에서 파일 시스템 액세스 요청을 하면 다음이 수행됩니다.

1. autofs는 해당 요청을 가로칩니다.
2. autofs는 마운트가 요청된 파일 시스템에 대한 메시지를 automountd로 보냅니다.
3. automountd는 맵에서 파일 시스템 정보를 찾고, 트리거 노드를 만들고, 마운트를 수행합니다.
4. autofs는 가로챈 요청의 처리를 허용합니다.
5. autofs는 무작동 기간이 경과되면 파일 시스템을 마운트 해제합니다.

주 - autofs 서비스를 통해 관리되는 마운트는 수동으로 마운트 또는 마운트 해제해서는 안 됩니다. 해당 작업이 성공해도 autofs 서비스에서는 객체가 마운트 해제되었는지를 확인하지 않으므로 불일치 현상이 발생할 수 있습니다. 재부트를 수행하면 autofs 마운트 지점이 모두 지워집니다.

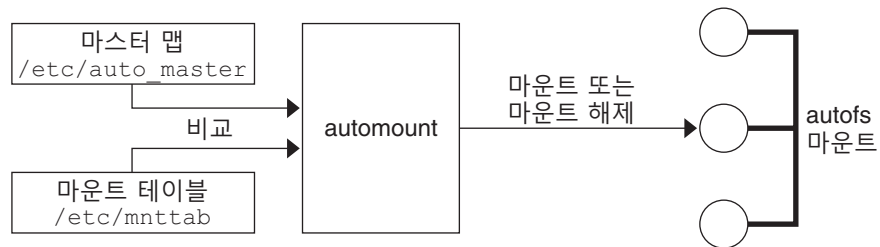
## autofs가 네트워크(맵)를 탐색하는 방법

autofs는 일련의 맵을 검색하여 네트워크를 탐색합니다. 맵은 네트워크에 있는 모든 사용자의 암호 항목이나 네트워크의 모든 호스트 컴퓨터 이름과 같은 정보가 포함된 파일입니다. 즉, 맵은 UNIX 관리 파일에 포함된 항목에 해당하는 네트워크 항목을 포함합니다. 맵은 로컬에서 또는 NIS 등의 네트워크 이름 서비스를 통해 사용할 수 있습니다. 210 페이지 “autofs가 네트워크를 탐색하는 방법 수정(맵 수정)”을 참조하십시오.

## autofs에서 탐색 프로세스를 시작하는 방법(마스터 맵)

automount 명령은 시스템 시작 시 마스터 맵을 읽습니다. [그림 6-4](#)에 나와 있는 것처럼 마스터 맵의 모든 항목은 직접 맵 이름 또는 간접 맵 이름, 해당 경로 및 해당 마운트 옵션입니다. 항목의 특정 순서는 중요하지 않습니다. automount는 마스터 맵의 항목을 마운트 테이블의 항목과 비교하여 현재 목록을 생성합니다.

그림 6-4 마스터 맵을 통한 탐색



## autofs 마운트 프로세스

마운트 요청이 트리거될 때 autofs 서비스에서 수행하는 작업은 자동 마운트 맵이 구성된 방식에 따라 다릅니다. 일반적으로 마운트 프로세스는 모든 마운트에 대해 동일합니다. 그러나 최종 결과는 지정된 마운트 지점과 맵의 복잡도에 따라 달라집니다. 마운트 프로세스에는 트리거 노드 생성이 포함됩니다.

### 단순 autofs 마운트

autofs 마운트 프로세스를 쉽게 설명하기 위해 다음과 같은 파일이 설치되어 있다고 가정하겠습니다.

```

$ cat /etc/auto_master
# Master map for automounter
#
+auto_master
/net      -hosts      -nosuid,nobrowse
/home     auto_home   -nobrowse
/share    auto_share
$ cat /etc/auto_share
# share directory map for automounter
#
ws        gumbo:/export/share/ws
  
```

/share 디렉토리에 액세스하면 autofs 서비스는 /share/ws에 대한 트리거 노드를 만듭니다. /share/ws는 /etc/mnttab의 항목으로, 다음 항목과 비슷합니다.

```
-hosts /share/ws autofs nosuid,nobrowse,ignore,nest,dev=###
```

/share/ws 디렉토리에 액세스하면 autofs 서비스는 다음과 같은 단계를 수행하여 프로세스를 완료합니다.

1. 서버 마운트 서비스를 사용할 수 있는지 확인합니다.
2. 요청된 파일 시스템을 /share 아래에 마운트합니다. 그러면 /etc/mnttab 파일에 다음 항목이 포함됩니다.

```
-hosts /share/ws autofs nosuid,nobrowse,ignore,nest,dev=###
gumbo:/export/share/ws /share/ws nfs nosuid,dev=#### #####
```

## 계층적 마운트

자동 마운트 파일에 여러 계층이 정의되어 있으면 마운트 프로세스가 더 복잡해집니다. 다음 항목을 포함하여 이전 예제에서 사용했던 /etc/auto\_shared 파일을 확장한다고 가정해 보겠습니다.

```
# share directory map for automounter
#
ws      /      gumbo:/export/share/ws
        /usr   gumbo:/export/share/ws/usr
```

/share/ws 마운트 지점에 액세스할 때의 마운트 프로세스는 기본적으로 이전 예제와 동일합니다. 또한 다음 레벨로의 트리거 노드(/usr)가 /share/ws 파일 시스템에 만들어지므로 다음 레벨을 액세스한 적이 있었던 것처럼 마운트할 수 있습니다. 이 예에서는 /export/share/ws/usr가 NFS 서버에 있어야 트리거 노드가 만들어집니다.



**주의** - 계층적 계층을 지정할 때는 -soft 옵션을 사용하지 마십시오. 이 제한에 대한 설명은 204 페이지 “autofs 마운트 해제”를 참조하십시오.

## autofs 마운트 해제

특정 유휴 시간이 경과되고 나면 수행되는 마운트 해제는 상향식(마운트의 역순)입니다. 계층에서 더 높은 레벨의 디렉토리 중 하나를 사용 중이면 해당 디렉토리 아래의 파일 시스템만 마운트 해제됩니다. 마운트 해제 프로세스 중에는 트리거 노드가 제거되고 파일 시스템이 마운트 해제됩니다. 파일 시스템이 사용 중이면 마운트 해제가 실패하고 트리거 노드가 다시 설치됩니다.



**주의** - 계층적 계층을 지정할 때는 -soft 옵션을 사용하지 마십시오. -soft 옵션을 사용하는 경우 트리거 노드 다시 설치 요청의 시간이 초과될 수 있습니다. 트리거 노드 다시 설치가 실패하면 다음 마운트 레벨에 액세스할 수 없습니다. 이 문제를 해결하는 방법은 자동 마운트에서 계층의 모든 구성 요소를 마운트 해제하도록 하는 것뿐입니다. 자동 마운트는 파일 시스템이 자동으로 마운트 해제되도록 기다리거나 시스템을 재부트하여 마운트 해제를 완료할 수 있습니다.

## autofs에서 클라이언트에 대해 가장 가까운 읽기 전용 파일을 선택하는 방법(여러 위치)

예제 직접 맵에는 다음 항목이 포함됩니다.

```
/usr/local      -ro \
  /bin          ivy:/export/local/sun4\
  /share        ivy:/export/local/share\
  /src          ivy:/export/local/src
/usr/man        -ro oak:/usr/man \
                rose:/usr/man \
                willow:/usr/man
/usr/games      -ro peach:/usr/games
/usr/spool/news -ro pine:/usr/spool/news \
                willow:/var/spool/news
```

마운트 지점 `/usr/man` 및 `/usr/spool/news`는 둘 이상의 위치가 나열됩니다(첫번째 마운트 지점의 경우 3개 위치, 두번째 마운트 지점의 경우 2개 위치). 복제된 위치도 사용자에게 대해 같은 서비스를 제공할 수 있습니다. 이 절차는 읽기 전용인 파일 시스템을 마운트할 때만 적용됩니다. 쓰거나 수정하는 파일 위치에 대한 어느 정도의 제어권이 있어야 하기 때문입니다. 한 서버에서 파일을 수정한 후에 다른 서버에서 “같은” 파일을 다시 수정해야 한다면 작업이 번거로워질 것입니다. 사용 가능한 최적의 서버가 자동으로 사용되므로 사용자가 작업을 수행하지 않아도 된다는 이점도 있습니다.

파일 시스템이 복제(185 페이지 “복제된 파일 시스템이란?” 참조)로 구성된 경우 클라이언트에서는 파일오버를 사용할 수 있습니다. 최적의 서버가 자동으로 결정될 뿐만 아니라, 해당 서버를 사용할 수 없으면 클라이언트가 다음 최상의 서버를 자동으로 사용합니다.

복제로 구성하기에 적합한 파일 시스템의 예로는 매뉴얼 페이지가 있습니다. 대규모 네트워크에서는 여러 서버가 현재 매뉴얼 페이지 세트를 내보낼 수 있습니다. 서버가 실행 중이며 해당 파일 시스템을 내보낸다면, 어떤 서버에서 매뉴얼 페이지를 마운트하는지는 중요하지 않습니다. 이전 예제에서는 여러 마운트 위치가 맵 항목에서 마운트 위치 목록으로 표현됩니다.

```
/usr/man -ro oak:/usr/man rose:/usr/man willow:/usr/man
```

이 예에서는 `oak`, `rose` 또는 `willow` 서버에서 매뉴얼 페이지를 마운트할 수 있습니다. 가장 효율적인 서버는 다음과 같은 다양한 요인에 따라 달라집니다.

- 특정 NFS 프로토콜 레벨을 지원하는 서버의 수
- 서버의 인접도
- 가중치 지정

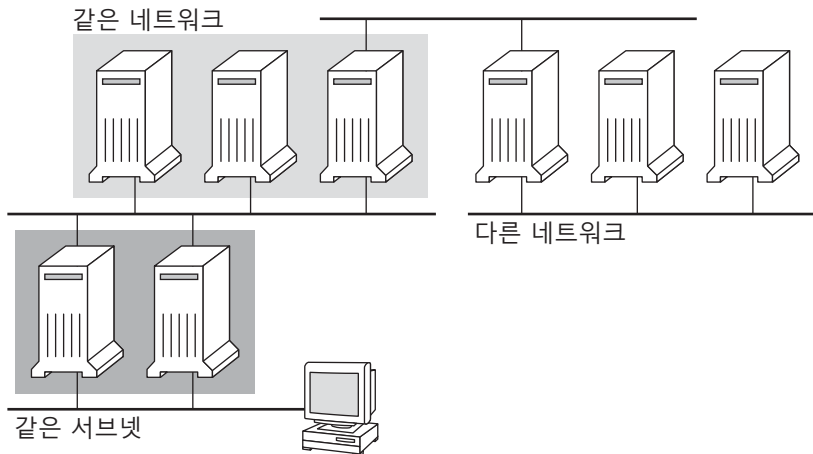
정렬 프로세스 중에는 각 NFS 프로토콜 버전을 지원하는 서버 수를 계산합니다. 가장 많은 서버에서 지원되는 프로토콜 버전이 기본적으로 사용되는 프로토콜이 됩니다. 이러한 선택 방법으로 인해 클라이언트는 가장 많은 수의 서버를 사용할 수 있습니다.

프로토콜 버전이 같은 최대 서버 하위 세트를 찾으면 해당 서버 목록이 인접도에 따라 정렬됩니다. 인접도를 확인하기 위해 IPv4 주소를 검사합니다. IPv4 주소는 각 서버넷 내의 서버를 보여줍니다. 로컬 서버넷의 서버가 원격 서버넷의 서버보다 우선적으로 사용됩니다. 가장 가까운 서버가 우선적으로 사용되므로 대기 시간과 네트워크 트래픽이 줄어듭니다.

주 - IPv6 주소를 사용하는 복제본의 경우에는 인접도를 확인할 수 없습니다.

그림 6-5에서는 서버 인접도를 보여줍니다.

그림 6-5 서버 인접도



로컬 서버넷에 같은 프로토콜을 지원하는 서버가 여러 대 있는 경우 각 서버에 연결하는 시간을 확인하여 연결 시간이 가장 빠른 서버가 사용됩니다. 가중치를 지정하여 정렬을 조정할 수도 있습니다(208 페이지 “autofs 및 가중치” 참조).

예를 들어 버전 4 서버의 수가 가장 많으면 버전 4가 기본적으로 사용되는 프로토콜이 됩니다. 그러나 이제는 정렬 프로세스가 보다 복잡합니다. 아래에 정렬 프로세스의 작동 방식을 보여주는 몇 가지 예제가 나와 있습니다.

- 로컬 서브넷의 서버가 원격 서브넷의 서버보다 우선적으로 사용됩니다. 따라서 버전 3 서버가 로컬 서브넷에 있는데 가장 가까운 버전 4 서버는 원격 서브넷에 있으면 버전 3 서버가 우선적으로 사용됩니다. 마찬가지로 로컬 서브넷이 버전 2 서버로 구성된 경우 버전 3 및 버전 4 서버로 구성된 원격 서브넷보다 우선적으로 사용됩니다.
- 로컬 서브넷이 각각 다른 수의 버전 2, 버전 3, 버전 4 서버로 구성된 경우에는 추가적인 정렬이 필요합니다. 자동 마운트는 로컬 서브넷에서 가장 높은 버전을 우선적으로 사용합니다. 이 경우에는 버전 4가 가장 높은 버전입니다. 그러나 로컬 서브넷의 버전 3 또는 버전 2 서버 수가 버전 4 서버의 수보다 많은 경우 자동 마운트는 로컬 서브넷의 가장 높은 버전에서 한 버전 낮은 서버를 사용합니다. 예를 들어 로컬 서브넷에 버전 4 서버 3대, 버전 3 서버 3대, 버전 2 서버 10대가 있는 경우에는 버전 3 서버가 선택됩니다.
- 마찬가지로 로컬 서브넷이 각각 다른 수의 버전 2 및 버전 3 서버로 구성된 경우 자동 마운트는 먼저 로컬 서브넷에서 가장 높은 버전을 나타내는 버전을 확인합니다. 그런 다음 자동 마운트는 각 버전을 실행하는 서버의 수를 계산합니다. 로컬 서브넷의 가장 높은 버전이 수가 가장 많은 서버와 일치하면 가장 높은 버전이 선택됩니다. 더 낮은 버전의 수가 더 많으면 자동 마운트는 로컬 서브넷에서 가장 높은 버전보다 하나 낮은 버전을 선택합니다. 예를 들어 로컬 서브넷에서 버전 3 서버보다 버전 2 서버의 수가 더 많으면 버전 2 서버가 선택됩니다.

---

주 - 가중치 역시 SMF 저장소에 저장된 매개변수의 영향을 받습니다. 구체적으로 `server_versmin`, `client_versmin`, `server_versmax` 및 `client_versmax` 의 값에 따라 일부 버전은 정렬 프로세스에서 제외될 수도 있습니다. 이러한 매개변수에 대한 자세한 내용은 137 페이지 “[mountd 데몬](#)” 및 137 페이지 “[nfsd 데몬](#)”을 참조하십시오.

---

페일오버를 사용하는 경우 서버를 선택하면 마운트 시에 정렬을 확인합니다. 개별 서버에서 해당 파일 시스템을 일시적으로 내보내지 않을 수 있는 환경에서는 여러 위치를 사용하면 유용합니다.

서브넷이 많은 대규모 네트워크에서는 페일오버가 특히 유용합니다. autofs는 적절한 서버를 선택하며, NFS 네트워크 트래픽을 로컬 네트워크 세그먼트로 제한할 수 있습니다. 서버에 네트워크 인터페이스가 여러 개 있으면 각 네트워크 인터페이스가 별도의 서버인 것처럼 인터페이스에 연관된 호스트 이름을 나열할 수 있습니다. autofs는 클라이언트에 가장 가까운 인터페이스를 선택합니다.

---

주 - 수동 마운트의 경우 가중치 및 인접도 확인을 수행하지 않습니다. mount 명령은 나열된 서버의 우선 순위를 왼쪽부터 오른쪽으로 지정합니다.

---

자세한 내용은 [automount\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## autofs 및 가중치

autofs 맵에 가중치 값을 추가하여 인접도 레벨이 같은 서버 선택을 조정할 수 있습니다. 예를 들면 다음과 같습니다.

```
/usr/man -ro oak,rose(1),willow(2):/usr/man
```

괄호 안의 숫자가 가중치를 나타냅니다. 가중치가 없는 서버는 값이 0이므로 선택될 가능성이 가장 높습니다. 가중치 값이 높을수록 서버가 선택될 가능성은 낮아집니다.

---

주 - 기타 모든 서버 선택 요인은 가중치보다 중요합니다. 가중치는 네트워크 인접도가 동일한 서버 중 선택할 항목을 결정할 때만 고려됩니다.

---

## autofs 맵 항목의 변수

이름 앞에 달러 기호(\$)를 접두어로 붙여 클라이언트별 변수를 만들 수 있습니다. 변수를 사용하면 같은 파일 시스템 위치에 액세스하는 여러 구조 유형을 포함할 수 있습니다. 중괄호를 사용하여 변수 이름과 추가된 문자/숫자를 구분할 수도 있습니다. 표 6-3에는 미리 정의된 맵 변수가 나와 있습니다.

표 6-3 미리 정의된 맵 변수

변수	의미	파생 원본	예제
ARCH	구조 유형	uname -m	sun4
CPU	프로세서 유형	uname -p	sparc
HOST	호스트 이름	uname -n	dinky
OSNAME	운영 체제 이름	uname -s	SunOS
OSREL	운영 체제 릴리스	uname -r	5.8
OSVERS	운영 체제 버전(릴리스 버전)	uname -v	GENERIC

변수는 항목 행의 어디에나 사용할 수 있습니다. 단, 키로는 사용할 수 없습니다. 예를 들어 SPARC 및 x86 구조의 이진을 각각 `/usr/local/bin/sparc` 및 `/usr/local/bin/x86`에서 내보내는 파일 서버가 있다고 가정해 보겠습니다. 클라이언트는 다음과 같은 맵 항목을 통해 마운트할 수 있습니다.

```
/usr/local/bin -ro server:/usr/local/bin/$CPU
```



그러면 모든 클라이언트에 대해 같은 항목이 모든 구조에 적용됩니다.

주 - sun4 구조용으로 작성된 모든 응용 프로그램은 모든 sun4 플랫폼에서 실행할 수 있습니다. -ARCH 변수는 sun4로 하드 코드됩니다.

## 다른 맵을 참조하는 맵

파일 맵에서 맵 항목 *+mapname*을 사용하면 자동 마운트에서 지정된 맵을 현재 파일에 포함된 것처럼 읽습니다. *mapname* 앞에 슬래시가 없으면 autofs는 맵 이름을 문자열로 처리하며 이름 서비스 스위치 정책을 사용하여 맵 이름을 찾습니다. 경로 이름이 절대 경로 이름이면 automount는 해당 이름의 로컬 맵을 확인합니다. 맵 이름이 대시(-)로 시작하는 경우 automount는 hosts와 같은 해당 내장 맵을 찾습니다.

svc:system/name-service/switch 서비스는 이름 지정 서비스의 검색 순서를 포함합니다. config 등록 정보 그룹의 automount 등록 정보는 자동 마운트 항목을 찾을 때 이름 서비스 데이터베이스를 검색하는 순서를 지정합니다. 특정 config/automount 등록 정보가 지정되어 있지 않으면 config/default 등록 정보에 정의된 순서가 사용됩니다. 예를 들면 다음과 같습니다.

```
# svcprop -p config svc:/system/name-service/switch
config/value_authorization astring solaris.smf.value.name-service.switch
config/printer astring user\ files
config/default astring files\ nis
config/automount astring files\ nis
```

이 예제에서는 로컬 파일의 맵을 NIS 맵보다 먼저 검색합니다. config/automount 등록 정보가 지정되지 않은 경우에도 마찬가지입니다. 이 경우에는 config/default 항목이 사용되기 때문입니다. 따라서 가장 자주 액세스하는 홈 디렉토리의 경우 로컬 /etc/auto\_home 맵에 몇 개의 항목을 포함할 수 있습니다. 그런 다음 스위치를 사용하여 해당 항목에 대해 NIS 맵으로 폴백할 수 있습니다.

```
bill          cs.csc.edu:/export/home/bill
bonny         cs.csc.edu:/export/home/bonny
```

포함된 맵을 확인한 후 일치하는 항목이 없으면 automount는 현재 맵을 계속 스캔합니다. 따라서 + 항목 뒤에 항목을 더 추가할 수 있습니다.

```
bill          cs.csc.edu:/export/home/bill
bonny         cs.csc.edu:/export/home/bonny
+auto_home
```

포함된 맵은 로컬 파일 또는 내장 맵일 수 있습니다. 로컬 파일만 + 항목을 포함할 수 있습니다.

```
+/etc/auto_mystuff # local map
+auto_home         # NIS map
+.hosts            # built-in hosts map
```

---

주-NIS 맵에서는 + 항목을 사용할 수 없습니다.

---

## 실행 가능 autofs 맵

일부 명령을 실행하여 autofs 마운트 지점을 생성하는 autofs 맵을 만들 수 있습니다. 데이터베이스 또는 플랫폼 파일에서 autofs 구조를 만들어야 하는 경우 실행 가능 autofs 맵을 사용하면 효율적입니다. 그러나 실행 가능 맵을 사용하는 경우 각 호스트에 해당 맵을 설치해야 한다는 단점이 있습니다. NIS 이름 서비스에는 실행 가능 맵을 포함할 수 없습니다.

실행 가능 맵에는 `auto_master` 파일의 항목이 있어야 합니다.

```
/execute    auto_execute
```

아래에 실행 가능 맵의 예가 나와 있습니다.

```
#!/bin/ksh
#
# executable map for autofs
#

case $1 in
    src) echo '-nosuid,hard bee:/export1' ;;
esac
```

이 예가 작동하려면 파일을 `/etc/auto_execute`로 설치해야 하며 파일에 실행 가능한 비트 세트가 있어야 합니다. 권한을 744로 설정합니다. 이러한 상황에서 다음 명령을 실행하면 `bee`의 `/export1` 파일 시스템이 마운트됩니다.

```
% ls /execute/src
```

## autofs가 네트워크를 탐색하는 방법 수정(맵 수정)

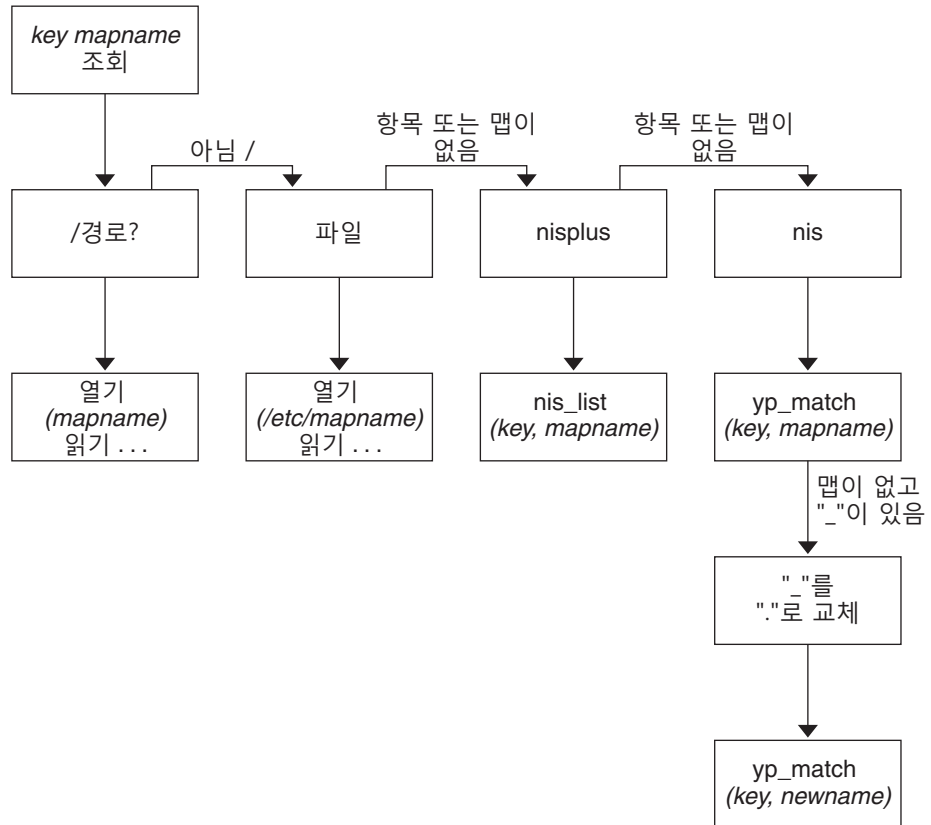
환경의 요구를 충족하기 위해 맵을 수정 또는 삭제하거나 맵에 항목을 추가할 수 있습니다. 사용자가 필요로 하는 응용 프로그램 및 기타 파일 시스템의 위치가 변경되면 맵에 해당 변경 내용을 반영해야 합니다. autofs 맵은 언제든지 수정할 수 있습니다. 다음 번에 automountd가 파일 시스템을 마운트할 때 수정 내용이 적용되는지 여부는 수정하는 맵 및 수정의 종류에 따라 달라집니다.

## 이름 서비스에 대한 기본 autofs 동작

부트 시에는 autofs가 `svc:/system/filesystem/autofs` 서비스에 의해 호출되어 마스터 `auto_master` 맵을 확인합니다. autofs는 규칙에 따라 달라지며, 여기에 대해서는 아래에서 설명합니다.

autofs는 `svc:/system/name-service/switch` 서비스의 `config/automount` 등록 정보에 지정된 이름 서비스 순서를 사용합니다. `config/automount` 등록 정보가 정의되어 있지 않으면 `config/default` 등록 정보가 사용됩니다. NIS를 선택했는데 autofs가 필요한 맵은 찾지 못하고 밑줄이 하나 이상 포함된 맵 이름을 찾으면 밑줄이 점으로 변경됩니다. 이와 같이 이름이 변경되므로 이전 NIS 파일 이름도 계속 작동합니다. 그러면 autofs는 그림 6-6에 나와 있는 것처럼 맵을 다시 확인합니다.

그림 6-6 autofs에서 이름 서비스를 사용하는 방법



이 세션의 화면 작업은 다음 예제와 같습니다.

```
$ grep /home /etc/auto_master
/home          auto_home
```

```
$ ypmatch brent auto_home
Can't match key brent in map auto_home. Reason: no such map in
server's domain.
```

```
$ ypmatch brent auto.home
diskus:/export/home/diskus1/&
```

“파일”을 이름 서비스로 선택하면 모든 맵이 `/etc` 디렉토리의 로컬 파일로 간주됩니다. autofs는 사용하는 이름 서비스에 관계없이 슬래시(/)로 시작하는 맵 이름을 로컬로 해석합니다.

## autofs 참조

이 장의 나머지 절에서는 고급 autofs 기능 및 항목에 대해 설명합니다.

## autofs 및 메타 문자

autofs는 일부 문자를 특별한 의미가 있는 것으로 인식합니다. 대체에 사용되는 문자도 있고, autofs 맵 구문 분석기로부터 다른 문자를 보호하는 데 사용되는 문자도 있습니다.

### 앰퍼센드(&)

다음과 같이 맵의 하위 디렉토리가 많이 지정되어 있는 경우 문자열 대체를 사용할 수 있습니다.

```
john      willow:/home/john
mary      willow:/home/mary
joe       willow:/home/joe
able      pine:/export/able
baker     peach:/export/baker
```

앰퍼센드 문자(&)를 사용하여 키가 표시되는 모든 위치에서 키를 대체할 수 있습니다. 앰퍼센드를 사용하는 경우 이전 맵이 다음과 같이 변경됩니다.

```
john      willow:/home/&
mary      willow:/home/&
joe       willow:/home/&
able      pine:/export/&
baker     peach:/export/&
```

직접 맵에서도 다음과 같은 경우에 키 대체를 사용할 수 있습니다.

```
/usr/man                                willow,cedar,poplar:/usr/man
```

또한 다음과 같이 항목을 더욱 단순화할 수도 있습니다.

```
/usr/man                                willow,cedar,poplar:&
```

앰퍼센드 대체에서는 전체 키 문자열을 사용합니다. 따라서 직접 맵의 키가 원래 규칙대로 /로 시작하는 경우 대체에 슬래시가 포함됩니다. 따라서 다음은 수행할 수 없습니다.

```
/progs                                &1,&2,&3:/export/src/progs
```

그 이유는 autofs에서 이 예를 다음과 같이 해석하기 때문입니다.

```
/progs                                /progs1,/progs2,/progs3:/export/src/progs
```

## 별표(\*)

범용 대체 문자인 별표(\*)를 사용하여 모든 키와 일치하도록 지정할 수 있습니다. 이 맵 항목을 통해 모든 호스트에서 /export 파일 시스템을 마운트할 수 있습니다.

```
*                                    &:/export
```

각 앰퍼센드는 지정된 키의 값으로 대체됩니다. autofs는 별표를 파일 끝 문자로 해석합니다.

## autofs 및 특수 문자

맵 항목에 특수 문자가 포함된 경우에는 디렉토리를 마운트할 때 이름으로 인해 autofs 맵 구문 분석기에 혼란을 줄 수도 있습니다. autofs 구문 분석기는 콜론, 쉼표, 공백 등이 포함된 이름에 민감합니다. 이러한 이름은 다음과 같이 큰따옴표로 묶어야 합니다.

```
/vms      -ro      vmsserver: - - - "rc0:dk1 - "  
/mac      -ro      gator:/ - "Mr Disk - "
```



## 제 3 부

# SLP 항목

이 절에서는 SLP(Service Location Protocol) 서비스에 대한 개요, 계획, 작업 및 참조 정보를 제공합니다.





## SLP(개요)

---

서비스 위치 프로토콜(SLP)은 검색에 대해 이동 가능한 플랫폼 독립 프레임워크 및 SLP 사용 가능 네트워크 서비스의 프로비저닝을 제공합니다. 이 장에서는 SLP 구조 및 IP 인트라넷에 대한 SLP의 Oracle Solaris 구현에 대해서 설명합니다.

- 217 페이지 “SLP 구조”
- 220 페이지 “SLP 구현”

## SLP 구조

이 절에서는 SLP의 기본 작업에 대해 간략하게 설명하고 SLP 관리에 사용되는 에이전트 및 프로세스에 대해 설명합니다.

SLP는 약간의 구성 또는 아무런 구성 없이 다음과 같은 모든 서비스를 자동으로 제공합니다.

- 클라이언트 응용 프로그램이 서비스에 액세스하는 데 필요한 정보 요청
- 네트워크 하드웨어 장치 또는 소프트웨어 서버에서 서비스 알림(예: 프린터, 파일 서버, 비디오 카메라 및 HTTP 서버)
- 기본 서버 오류로부터 복구 관리

또한 필요한 경우 다음을 수행하여 SLP 작업을 관리하고 조정할 수 있습니다.

- 논리적 또는 기능적 그룹으로 구성된 **범위**로 서비스 및 사용자 구성
- SLP 로깅을 사용으로 설정하여 네트워크에서 SLP 작업 모니터링 및 문제 해결
- SLP 타이밍 매개변수를 조정하여 성능 및 확장성 향상
- 멀티캐스트 경로 지정에 대한 지원이 부족한 네트워크에서 SLP를 배포할 때 멀티캐스트 메시지를 처리 및 보내지 않도록 SLP 구성
- SLP 디렉토리 에이전트를 배포하여 확장성 및 성능 향상

## SLP 설계 요약

SLP 라이브러리는 네트워크에서 해당 서비스를 검색할 수 있도록 서비스를 알리는 네트워크 인식 에이전트를 알립니다. SLP 에이전트는 서비스 유형 및 위치에 대한 최신 정보를 유지 관리합니다. 또한 이러한 에이전트는 프록시 등록을 사용하여 SLP가 직접 사용으로 설정되지 않은 서비스를 알릴 수 있습니다. 자세한 내용은 [10 장, “레저시 서비스 통합”](#)을 참조하십시오.

클라이언트 응용 프로그램은 서비스를 알리는 에이전트에 직접 요청을 만드는 SLP 라이브러리에 의존합니다.

## SLP 에이전트 및 프로세스

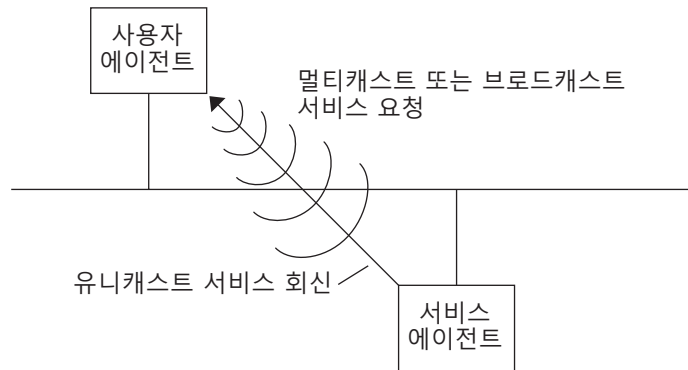
다음 표는 SLP 에이전트에 대해 설명합니다. 이 볼륨에 사용된 용어 및 기타 용어에 대한 확장된 정의는 [용어집](#)을 참조하십시오.

표 7-1 SLP 에이전트

SLP 에이전트	설명
DA(디렉토리 에이전트)	SA(서비스 에이전트)에서 등록한 SLP 알림을 캐시하는 프로세스입니다. DA는 요구 시 서비스 알림을 UA(사용자 에이전트)에 전달합니다.
SA(서비스 에이전트)	서비스 알림을 분산하고 DA(디렉토리 에이전트)를 사용하여 서비스를 등록하기 위해 서비스 대신 사용하는 SLP 에이전트입니다.
UA(사용자 에이전트)	서비스 알림 정보를 가져오기 위해 사용자 또는 응용 프로그램 대신 사용하는 SLP 에이전트입니다.
범위	관리 또는 서비스의 논리적 그룹

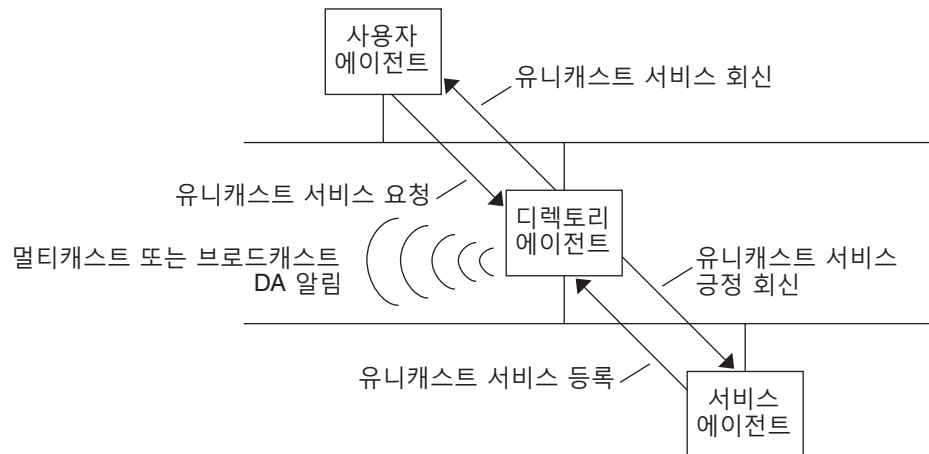
다음 그림은 SLP 구조를 구현하는 기본 에이전트 및 프로세스를 보여줍니다. 그림은 SLP의 기본 배포를 나타냅니다. 수행할 특별한 구성은 없습니다. UA 및 SA의 두 가지 에이전트만 필요합니다. SLP 프레임워크를 사용하면 UA가 SA에 대한 서비스 요청을 멀티캐스트할 수 있습니다. SA는 UA에 회신을 유니캐스트합니다. 예를 들어, UA가 서비스 요청 메시지를 보내면 SA는 서비스 회신 메시지로 응답합니다. 서비스 회신은 클라이언트의 요구 사항과 일치하는 서비스 위치를 포함합니다. 속성 및 서비스 유형에서 기타 요청 및 회신이 가능합니다. 자세한 내용은 [11 장, “SLP\(참조\)”](#)를 참조하십시오.

그림 7-1 SLP 기본 에이전트 및 프로세스



다음 그림은 DA가 프레임워크에서 배포될 때 SLP 구조를 구현하는 기본 에이전트 및 프로세스를 보여줍니다.

그림 7-2 DA로 구현된 SLP 구조적 에이전트 및 프로세스



DA를 배포하면 네트워크에 더 적은 메시지가 보내지고 UA는 더 빨리 정보를 검색할 수 있습니다. DA는 네트워크의 크기가 증가할 때 또는 멀티캐스트 경로 지정이 지원되지 않는 상황에서 필수입니다. DA는 등록된 서비스 알림에 캐시로 제공됩니다. SA는 DA에 알리는 모든 서비스를 나열하는 등록 메시지(SrvReg)를 보냅니다. 그런 다음 SA는 회신으로 긍정 응답(SrvAck)을 받습니다. 서비스 알림은 DA로 새로 고쳐지거나 알림에 대해 설정된 수명에 따라 만료됩니다. UA가 DA를 검색한 후 UA는 SA에 대한 요청을 멀티캐스트하는 대신 DA에 대한 요청을 유니캐스트합니다.

Oracle Solaris SLP 메시지에 대한 자세한 내용은 11 장, “SLP(참조)”를 참조하십시오.

## SLP 구현

Oracle Solaris SLP 구현에서는 [표 7-1](#)의 SLP SA, UA, DA, SA 서버, 범위 및 기타 구조 구성 요소가 `slpd` 및 응용 프로그램 프로세스로 부분 매핑됩니다. SLP 데몬 `slpd`는 특정 해제 호스트 SLP 상호 작용을 구성하여 다음 작업을 수행합니다.

- 네트워크에서 모든 DA를 검색하기 위해 수동 및 활성 디렉토리 에이전트 검색 사용
- 로컬 호스트의 UA 및 SA의 사용에 대해 업데이트된 DA 테이블 유지 관리
- 기존 서비스 알림(프록시 등록)에 대해 프록시 SA 서버로 작동

`net.slpisDA` 등록 정보를 설정하여 DA로 작동하도록 `slpd`를 구성할 수도 있습니다. [9 장](#), “SLP 관리(작업)”를 참조하십시오.

SLP 데몬에 대한 자세한 내용은 [slpd\(1M\)](#)를 참조하십시오.

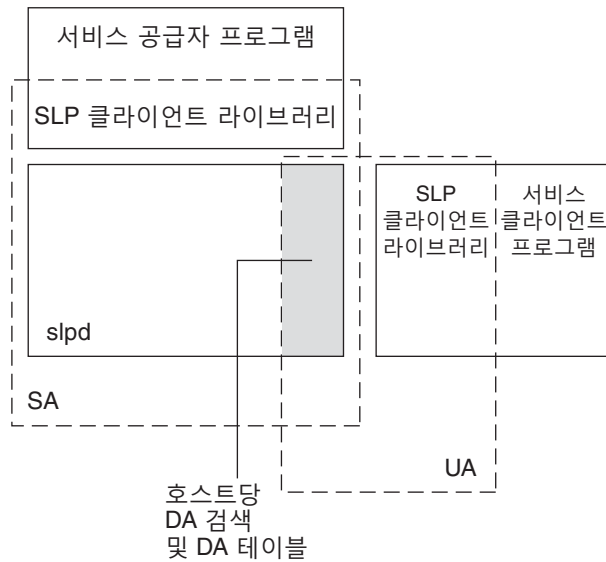
`slpd` 뿐만 아니라 C/C++ 및 Java 클라이언트 라이브러리(`libslp.so` 및 `slp.jar`)에서도 UA 및 SA 클라이언트에 대한 SLP 프레임워크에 액세스할 수 있습니다. 클라이언트 라이브러리는 다음 기능을 제공합니다.

- 서비스 알림을 등록 및 등록 해제할 수 있는 네트워크 서비스를 제공하는 소프트웨어
- 서비스 알림에 대한 질의를 실행하여 서비스를 요청할 수 있는 클라이언트 소프트웨어
- 등록 및 요청에 사용 가능한 SLP 범위 목록

`slpd` 및 이전 서비스를 제공하는 클라이언트 라이브러리 간의 내부 프로세스 통신을 사용으로 설정하는데 특별한 구성은 필요하지 않습니다. 그러나 라이브러리를 작동하려면 클라이언트 라이브러리를 로드하기 전에 먼저 `slpd` 프로세스를 실행해야 합니다.

다음 그림에서는 서비스 공급자 프로그램의 SLP 클라이언트 라이브러리가 SA 기능을 사용합니다. 서비스 공급자 프로그램은 SLP 클라이언트 라이브러리를 사용하여 `slpd`를 통해 서비스를 등록 및 등록 해제합니다. 서비스 클라이언트 프로그램의 SLP 클라이언트 라이브러리는 UA 기능을 사용합니다. 서비스 클라이언트 프로그램은 SLP 클라이언트를 사용하여 요청을 만듭니다. SLP 클라이언트 라이브러리는 SA에 대한 요청을 멀티캐스트하거나 DA에 대한 요청을 유니캐스트합니다. 이 통신은 요청 실행의 유니캐스트 메소드가 더 빠른 경우를 제외하고는 응용 프로그램에 명확히 나타납니다. 다양한 SLP 구성 등록 정보를 설정하여 클라이언트 라이브러리의 동작에 영향을 줄 수 있습니다. 자세한 내용은 [9 장](#), “SLP 관리(작업)”를 참조하십시오. `slpd` 프로세스는 멀티캐스트 요청 응답 및 DA를 통한 등록 등과 같은 모든 SA 기능을 처리합니다.

그림 7-3 SLP 구현



- 프로세스
- SLP 에이전트

## 기타 SLP 정보 원본

SLP에 대한 자세한 내용은 다음 문서를 참조하십시오.

- Kempf, James, 및 Pete St. Pierre. **Service Location Protocol for Enterprise Networks**. John Wiley & Sons, Inc. ISBN 번호: 0-471-31587-7.
- **Authentication Management Infrastructure Administration Guide**. 부품 번호: E25840
- Guttman, Erik, Charles Perkins, John Veizades 및 Michael Day. IETF(Internet Engineering Task Force)의 **Service Location Protocol, Version 2, RFC 2608**(<http://www.ietf.org/rfc/rfc2608.txt>)
- Kempf, James 및 Erik Guttman. IETF(Internet Engineering Task Force)의 **An API for Service Location, RFC 2614**(<http://www.ietf.org/rfc/rfc2614.txt>)



## SLP 계획 및 사용으로 설정(작업)

---

이 장에서는 SLP를 계획하고 사용으로 설정하는 데 대한 정보를 제공합니다. 다음 절에서는 SLP 구성과 SLP를 사용으로 설정하는 프로세스에 대해 설명합니다.

- 223 페이지 “SLP 구성 고려 사항”
- 224 페이지 “snoop를 사용하여 SLP 작업 모니터링”

### SLP 구성 고려 사항

SLP 데몬은 기본 등록 정보로 미리 구성됩니다. 기업이 기본 설정으로 잘 작동하는 경우 SLP 배포에는 관리가 거의 필요하지 않습니다.

그러나 경우에 따라 SLP 등록 정보를 수정하여 네트워크 작업을 조정하거나 특정 기능을 활성화할 수 있습니다. 예를 들어 구성을 약간 변경하여 SLP 로깅을 사용으로 설정할 수 있습니다. 그런 다음 SLP 로그 및 snoop 추적의 정보를 참조하여 추가 구성이 필요한지 여부를 결정할 수 있습니다.

SLP 구성 등록 정보는 `/etc/inet` 디렉토리의 `slp.conf` 파일에 있습니다. 기본 등록 정보 설정을 변경하기로 한 경우 9 장, “SLP 관리(작업)”에서 적절한 절차를 참조하십시오.

SLP 구성 설정을 수정하기 전에 네트워크 관리의 주요 측면과 관련된 다음 질문을 고려하십시오.

- 기업에서 운영 중인 네트워크 기술은 무엇입니까?
- 이 기술이 원활하게 처리할 수 있는 네트워크 트래픽은 얼마나 됩니까?
- 네트워크에서 사용할 수 있는 서비스의 수와 유형은 어떻게 됩니까?
- 네트워크의 사용자 수는 얼마나 됩니까? 필요한 서비스는 무엇입니까? 사용자가 가장 자주 액세스한 서비스와 관련하여 사용자가 있는 위치는 어디입니까?

## 다시 구성할 항목 결정

SLP 사용 가능 snoop 유틸리티와 SLP 로깅 유틸리티를 사용하여 재구성이 필요한지 여부와 수정해야 하는 등록 정보를 결정할 수 있습니다. 예를 들어 특정 등록 정보를 재구성하여 다음을 수행할 수 있습니다.

- 대기 시간 및 대역폭 특성이 다른 네트워크 매체의 조합 수용
- 기업의 네트워크 오류나 계획되지 않은 분할 복구
- DA를 추가하여 SLP 멀티캐스트의 급증 줄이기
- 새로운 범위를 구성하여 가장 자주 액세스한 서비스를 중심으로 사용자 환경 구성

## snoop를 사용하여 SLP 작업 모니터링

snoop 유틸리티는 네트워크 트래픽 정보를 제공하는 수동 관리 도구입니다. 유틸리티 자체는 최소의 트래픽을 생성하지만 이 유틸리티를 통해 네트워크에서 발생하는 모든 작업을 볼 수 있습니다.

snoop 유틸리티는 실제 SLP 메시지 트래픽을 추적합니다. 예를 들어 snoop 유틸리티를 slp 명령줄 인수와 함께 실행하면 SLP 등록 및 등록 취소에 대한 정보가 포함된 추적이 표시됩니다. 이 정보를 통해 어떤 서비스가 등록되어 있고 얼마나 많은 재등록 작업이 일어나고 있는지 확인하여 네트워크 로드를 측정할 수 있습니다.

snoop 유틸리티는 기업에 있는 SLP 간의 트래픽 흐름을 관찰하는 데에도 유용합니다. snoop 유틸리티를 slp 명령줄 인수와 함께 실행하면 다음과 같은 유형의 SLP 작업을 모니터링하여 네트워크 또는 에이전트 구성이 필요한지 여부를 결정할 수 있습니다.

- 특정 DA를 사용하고 있는 호스트 수. 이 정보를 참조하여 로드 균형 조정을 위해 추가 DA를 배포할지 여부를 결정할 수 있습니다.
- 특정 DA를 사용하고 있는 호스트 수. 이 정보를 참조하여 특정 호스트를 새로운 범위로 구성할지 또는 다른 범위로 구성할지를 결정할 수 있습니다.
- UA가 시간 초과를 요청하는지 또는 DA 승인이 느린지 여부. UA 시간 초과 및 재전송을 모니터링하여 DA가 오버로드되었는지 여부를 확인할 수 있습니다. DA가 SA에 등록 승인을 보내기 위해 몇 초 이상이 필요한지 여부도 확인할 수 있습니다. 이 정보를 참조하여 필요한 경우 추가 DA를 배포하거나 범위 구성을 변경하는 방법으로 DA의 네트워크 로드 균형을 다시 조정할 수 있습니다.

snoop를 -V(verbose) 명령줄 인수와 함께 사용하면 SrvReg에서 등록 수명 및 새 태그 값을 가져와 재등록 수를 줄여야 하는지 여부를 결정할 수 있습니다.

snoop를 사용하여 다음과 같은 다른 종류의 SLP 트래픽도 추적할 수 있습니다.

- UA 클라이언트와 DA 사이의 트래픽
- 멀티캐스트 UA 클라이언트와 수신 SA 사이의 트래픽

snoop에 대한 자세한 내용은 [snoop\(1M\)](#)를 참조하십시오.



---

참고 - netstat 명령을 snoop와 함께 사용하여 트래픽 및 정체 통계를 볼 수 있습니다. netstat에 대한 자세한 내용은 [netstat\(1M\)](#)를 참조하십시오.

---

## ▼ snoop를 사용하여 SLP 추적을 실행하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 snoop를 slp 명령줄 인수와 함께 실행합니다.

**Brief Mode:**

# snoop slp

snoop를 기본값인 *brief* 모드로 실행하면 화면에 지속적인 출력이 제공됩니다. SLP 메시지는 SLP 추적당 한 행에 맞게 잘립니다.

**Verbose Mode:**

# snoop -v slp

snoop를 *Verbose* 모드로 실행하면 snoop는 다음과 같은 정보를 포함하는 간략화되지 않은 지속적인 출력을 화면에 제공합니다.

- 서비스 URL의 전체 주소
- 모든 서비스 속성
- 등록수명
- 모든 보안 매개변수 및 플래그(있는 경우)

---

주 - slp 명령줄 인수를 다른 snoop 옵션과 함께 사용할 수 있습니다.

---

## snoop slp 추적 분석

다음 예에서는 slpd가 기본 모드의 *slphost1*에서 SA 서버로 실행됩니다. SLP 데몬은 *slphost2*를 예코 서버로 초기화하고 등록합니다. 그런 다음 *slphost1*에서 snoop slp 프로세스를 호출합니다.

---

주 - 추적 결과에 대한 설명을 간소화하기 위해 다음 snoop 출력의 행에는 행 번호가 플래그로 지정되어 있습니다.

---

```
(1) slphost1 -> 239.255.255.253 SLP V@ SrvRqst [24487] service:directory-agent []
(2) slphost2 -> slphost1 SLP V2 DAAdvert [24487] service:directory-agent://129
(3) slphost1 -> 239.255.255.253 SLP V2 SrvRqst [24487] service:directory-agent []
```

```
(4) slphost1 -> 239.255.255.253 SLP V2 SrvRqst [24487] service:directory-agent []
(5) slphost1 -> slphost2 SLP V2 SrvReg [24488/tcp] service:echo.sun:tcp://slphost1:
(6) slphost2 -> slphost1 SLP V2 SrvAck [24488/tcp] ok
(7) slphost1 -> slphost2 SLP V2 SrvDereg [24489/tcp] service:echo.sun:tcp://slphost1:
(8) slphost2 -> slphost1 SLP V2 SrvAck [24489/tcp] ok
```

1. 디렉토리 에이전트 검색에서 SLP 멀티캐스트 그룹 주소에 멀티캐스트하여 활성 디렉토리 에이전트 검색을 수행하는 *slpd*를 *slphost1*에 표시합니다. 활성 디렉토리의 메시지 번호 24487은 추적 표시에서 대괄호로 표시됩니다.
2. 추적 1의 활성 디렉토리 요청 24487이 *slphost2* 호스트에서 DA로 실행되는 *slpd*에 의해 응답됨을 나타냅니다. *slphost2*의 서비스 URL은 한 행에 맞게 잘렸습니다. DA는 추적 1 및 2의 일치하는 메시지 번호로 표시된 멀티캐스트 디렉토리 에이전트 검색 메시지에 회신하여 DA 알림을 보냈습니다.
3. 추가 DA에 대해 *slphost1*의 UA에서 보내는 멀티캐스트를 표시합니다. *slphost2*가 요청에 이미 응답했으므로 다시 응답하지 못하게 하며 다른 DA가 회신하지 않습니다.
4. 이전 행에 표시된 멀티캐스트 작업을 반복합니다.
5. SA 클라이언트 등록을 *slphost2*의 DA에 전달하는 *slphost1*의 *slpd*를 표시합니다. 에코 서버에 대한 유니캐스트 서비스 등록(SrvReg)은 *slphost1*이 *slphost2*의 DA에 합니다.
6. *slphost1* SrvReg에 등록이 완료되었음을 나타내는 서비스 승인(SrvAck)으로 응답하는 *slphost2*를 표시합니다.  
SA 클라이언트를 실행하는 에코 서버와 *slphost1*의 SLP 데몬 간 트래픽은 *snoop* 추적에 표시되지 않습니다. 이 정보가 없는 것은 *snoop* 작업이 네트워크 루프백을 통해 수행되기 때문입니다.
7. 에코 서비스 알림을 등록 취소하는 *slphost1*의 에코 서버를 표시합니다. *slphost1*의 SLP 데몬은 등록 취소를 *slphost2*의 DA에 전달합니다.
8. *slphost1*에 등록 취소가 완료되었음을 나타내는 서비스 승인(SrvAck)으로 응답하는 *slphost2*를 표시합니다.  
5, 6, 7 및 8행의 메시지 번호에 추가된 /tcp 매개변수는 메시지 교환이 TCP를 통해 발생했음을 나타냅니다.

## 여기에서 이동할 위치

SLP 트래픽을 모니터링한 후 *snoop* 추적에서 수집한 정보를 사용하여 SLP 기본값에 대한 재구성이 필요한지 여부를 결정할 수 있습니다. 9 장, “SLP 관리(작업)”의 관련 정보를 사용하여 SLP 등록 정보 설정을 구성합니다. SLP 메시징 및 서비스 등록에 대한 자세한 내용은 11 장, “SLP(참조)”를 참조하십시오.

## SLP 관리(작업)

다음 절에서는 SLP 에이전트 및 프로세스 구성에 대한 정보 및 작업을 제공합니다.

- 227 페이지 “SLP 등록 정보 구성”
- 230 페이지 “DA 알림 및 검색 빈도 수정”
- 234 페이지 “다른 네트워크 매체, 토폴로지 또는 구성 수용”
- 239 페이지 “SLP 검색 요청에 대한 시간 초과 수정”
- 242 페이지 “범위 배포”
- 245 페이지 “DA 배포”
- 249 페이지 “SLP 및 멀티홈”

### SLP 등록 정보 구성

SLP 구성 등록 정보는 네트워크 상호 작용, SLP 에이전트 특성, 상태 및 로깅을 제어합니다. 대부분의 경우, 이러한 등록 정보의 기본 구성을 수정할 필요는 없습니다. 그러나 네트워크 매체 또는 토폴로지를 변경하는 경우 및 다음과 같은 목적을 달성하려는 경우 이 장에 나와 있는 절차를 수행할 수 있습니다.

- 네트워크 대기 시간 보정
- 네트워크 혼잡 줄이기
- 에이전트 추가 및 IP 주소 재지정
- SLP 로깅 활성화

SLP 구성 파일 `/etc/inet/slp.conf`를 편집하여 다음 표에 표시된 작업을 수행할 수 있습니다.

표 9-1 SLP 구성 작업

작업	설명
slpd가 DA 서버로 작동해야 하는지 여부를 지정합니다. SA 서버가 기본값입니다.	net.slpisDA 등록 정보를 True로 설정합니다.

표 9-1 SLP 구성 작업 (계속)

작업	설명
DA 멀티캐스트 메시지에 대한 타이밍을 설정합니다.	<code>net.slp.DAHeartBeat</code> 등록 정보를 설정하여 DA가 요청하지 않은 DA 알림을 멀티캐스트하는 횟수를 제어합니다.
DA 로깅을 사용으로 설정하여 네트워크 트래픽을 모니터링합니다.	<code>net.slp.traceDATraffic</code> 등록 정보를 <code>True</code> 로 설정합니다.

## SLP 구성 파일: 기본 요소

`/etc/inet/slp.conf` 파일은 SLP 데몬을 다시 시작할 때마다 모든 SLP 작업을 정의하고 활성화합니다. 구성 파일은 다음 요소로 구성되어 있습니다.

- 구성 등록 정보
- 주식 행 및 표기법

### 구성 등록 정보

`net.slp.isDA` 및 `net.slp.DAHeartBeat`와 같은 모든 기본 SLP 등록 정보의 이름 형식은 다음과 같습니다.

```
net.slp.<keyword>
```

SLP 동작은 `slp.conf` 파일의 등록 정보 조합 또는 등록 정보 값으로 정의됩니다. 등록 정보는 SLP 구성 파일의 키 값 쌍으로 구조화됩니다. 다음 예에서처럼 키 값 쌍은 등록 정보 이름 및 관련 설정으로 구성됩니다.

```
<property name>=<value>
```

각 등록 정보의 핵심은 등록 정보 이름입니다. 값은 등록 정보에 대한 숫자(거리 또는 시간), `True/False` 상태 또는 문자열 값 매개변수를 설정합니다. 등록 정보 값은 다음 데이터 유형 중 하나로 구성됩니다.

- `True/False` 설정(부울)
- 정수
- 정수 목록
- 문자열
- 문자열 목록

정의된 값이 허용되지 않는 경우 해당 등록 정보 이름의 기본 값이 사용됩니다. 또한 `syslog`를 통해 오류 메시지가 기록됩니다.

### 주식 행 및 표기법

행의 특성 및 기능을 설명하는 주석을 `slp.conf` 파일에 추가할 수 있습니다. 파일의 주식 행은 선택 사항이지만 관리하는 데 유용할 수 있습니다.

---

주 - 구성 파일의 설정은 대소문자를 구분하지 않습니다. 자세한 내용은 IETF(Internet Engineering Task Force)의 Guttman, Erik, James Kempf 및 Charles Perkins, “Service Templates and service: scheme”, RFC 2609(<http://www.ietf.org/rfc/rfc2609.txt>)를 참조하십시오.

---

## ▼ SLP 구성 변경 방법

다음 절차를 수행하여 SLP 구성 파일의 등록 정보 설정을 변경합니다. SLP 사용 가능 클라이언트 또는 서비스에서도 SLP API를 사용하여 SLP 구성을 변경할 수 있습니다. 이 API는 IETF(Internet Engineering Task Force)의 “An API for Service Location”, RFC 2614(<http://www.ietf.org/rfc/rfc2614.txt>)에 설명되어 있습니다.

### 1 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

### 2 호스트에서 `slpd` 및 모든 SLP 작업을 중지합니다.

```
# svcadm disable network/slp
```

### 3 구성 설정을 변경하기 전에 기본 `/etc/inet/slp.conf` 파일을 백업합니다.

### 4 필요에 따라 `/etc/inet/slp.conf` 파일의 등록 정보 설정을 편집합니다.

SLP 등록 정보 설정에 대한 일반적인 정보는 228 페이지 “구성 등록 정보”를 참조하십시오. `slp.conf` 등록 정보를 변경할 수 있는 다른 시나리오에 대한 예는 이 절차 다음에 나오는 절을 참조하십시오. `slp.conf(4)`를 참조하십시오.

### 5 변경 사항을 저장하고 파일을 닫습니다.

### 6 `slpd`를 다시 시작하여 변경 사항을 활성화합니다.

```
# svcadm enable network/slp
```

---

주 - SLP 데몬은 `slpd`를 중지하거나 시작할 때 구성 파일에서 정보를 가져옵니다.

---

## 예 9-1 DA 서버로 작동하도록 `slpd` 설정

`slpd.conf` 파일의 `net.slp.isDA` 등록 정보를 `True`로 설정하여 SA 서버 기본값을 변경하면 `slpd`가 DA 서버로 작동하도록 할 수 있습니다.

```
net.slp.isDA=True
```

각 영역의 다양한 등록 정보는 구성에 대한 여러 측면을 제어합니다. 다음 절에서는 SLP 구성에 사용되는 기본 등록 정보 설정을 변경할 수 있는 다른 시나리오에 대해 설명합니다.

## DA 알림 및 검색 빈도 수정

다음과 같은 경우 DA 알림 및 검색 요청의 타이밍을 제어하는 등록 정보를 수정할 수 있습니다.

- SA 또는 UA가 `slp.conf` 파일의 `net.slp.DAAddresses` 등록 정보에서 DA 구성 정보를 정적으로 가져오도록 하려는 경우 DA 검색을 사용 안함으로 설정할 수 있습니다.
- 네트워크가 반복 분할되는 경우 수동 알림 및 활성 검색의 빈도를 변경할 수 있습니다.
- UA 및 SA 클라이언트가 다이얼 업 연결의 다른 쪽에서 DA에 액세스하는 경우 DA 하트비트 빈도 및 활성 검색 간격을 줄여 다이얼 업 회선이 활성화되는 횟수를 줄일 수 있습니다.
- 네트워크가 매우 혼잡한 경우 멀티캐스트를 제한할 수 있습니다.

이 절의 절차는 다음 등록 정보를 수정하는 방법에 대해 설명합니다.

표 9-2 DA 알림 타이밍 및 검색 요청 등록 정보

등록 정보	설명
<code>net.slp.passiveDADetection</code>	<code>slpd</code> 가 요청하지 않은 DA 알림에 대해 수신 대기하는지 여부를 지정하는 부울
<code>net.slp.DAActiveDiscoveryInterval</code>	<code>slpd</code> 가 새 DA에 대한 활성 DA 검색을 수행하는 횟수를 지정하는 값
<code>net.slp.DAHeartBeat</code>	DA가 요청하지 않은 DA 알림을 멀티캐스트하는 횟수를 지정하는 값

## UA 및 SA를 정적으로 구성된 DA로 제한

경우에 따라 `slp.conf` 파일의 정적 구성 정보에서 DA 주소를 가져오기 위해 UA 및 SA를 제한해야 할 수도 있습니다. 다음 절차에서는 `slpd`가 `net.slp.DAAddresses` 등록 정보에서 독점적으로 DA 정보를 가져오도록 하는 두 등록 정보를 수정할 수 있습니다.

### ▼ UA 및 SA를 정적으로 구성된 DA로 제한하는 방법

다음 절차를 수행하여 `net.slp.passiveDADetection` 및 `net.slp.DAActiveDiscoveryInterval` 등록 정보를 변경할 수 있습니다.

---

주 - 정적 구성으로 제한된 UA 및 SA를 실행하는 호스트에서만 이 절차를 수행하십시오.

---

**1 관리자가 됩니다.**

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

**2 호스트에서 slpd 및 모든 SLP 작업을 중지합니다.**

```
# svcadm disable network/slp
```

**3 구성 설정을 변경하기 전에 기본 /etc/inet/slp.conf 파일을 백업합니다.**

**4 slp.conf 파일의 net.slp.passiveDADetection 등록 정보를 False로 설정하여 수동 검색을 사용 안함으로 설정합니다. 이렇게 설정하면 slpd가 요청하지 않은 DA 알림을 무시합니다.**

```
net.slp.passiveDADetection=False
```

**5 net.slp.DAActiveDiscoveryInterval을 -1로 설정하여 최초 및 주기적 활성 검색을 사용 안함으로 설정합니다.**

```
net.slp.DAActiveDiscoveryInterval=-1
```

**6 변경 사항을 저장하고 파일을 닫습니다.**

**7 slpd를 다시 시작하여 변경 사항을 활성화합니다.**

```
# svcadm enable network/slp
```

## 다이얼 업 네트워크에 대한 DA 검색 구성

UA 또는 SA가 다이얼 업 네트워크에 의해 DA에서 분리된 경우 DA 검색을 구성하여 검색 요청 및 DA 알림 수를 줄이거나 제거할 수 있습니다. 일반적으로 다이얼 업 네트워크가 활성화되면 비용이 부과됩니다. 필요 없는 통화를 최소화하면 다이얼 업 네트워크 사용 비용을 줄일 수 있습니다.

---

주 - 230 페이지 “UA 및 SA를 정적으로 구성된 DA로 제한”에 설명된 방법을 통해 DA 검색을 완전히 사용 안함으로 설정할 수 있습니다.

---

## ▼ 다이얼 업 네트워크에 대한 DA 검색을 구성하는 방법

다음 절차를 수행하여 DA 하트비트 주기 및 활성 검색 간격을 늘리면 요청하지 않은 DA 알림 및 활성 검색을 줄일 수 있습니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 호스트에서 `slpd` 및 모든 SLP 작업을 중지합니다.

```
# svcadm disable network/slp
```

### 3 구성 설정을 변경하기 전에 기본 `/etc/inet/slp.conf` 파일을 백업합니다.

### 4 `slpd.conf` 파일에서 `net.slp.DAHeartbeat` 등록 정보를 늘립니다.

```
net.slp.DAHeartbeat=value
```

**값** 수동 DA 알림 하트비트에 대한 초 수를 설정하는 32비트 정수

기본값=10800초(3시간)

값 범위=2000-259200000초

예를 들어, 다음과 같이 DA를 실행하는 호스트에서 DA 하트비트를 약 18시간으로 설정할 수 있습니다.

```
net.slp.DAHeartbeat=65535
```

### 5 다음과 같이 `slpd.conf` 파일의 `net.slp.DAActiveDiscoveryInterval` 등록 정보를 늘립니다.

```
net.slp.DAActiveDiscoveryInterval value
```

**value** DA 활성 검색 질의에 대한 초 수를 설정하는 32비트 정수

기본값=900초(15분)

값 범위=300-10800초

예를 들어, 다음과 같이 UA 및 SA를 실행 중인 호스트에서 DA 활성 검색 간격을 18시간으로 설정할 수 있습니다.

```
net.slp.DAActiveDiscoveryInterval=65535
```

### 6 변경 사항을 저장하고 파일을 닫습니다.



## 7 slpd를 다시 시작하여 변경 사항을 활성화합니다.

```
# svcadm enable network/slp
```

## 자주 분할하는 경우를 위한 DA 하트비트 구성

SA는 해당 범위를 지원하는 모든 DA에 등록하는 데 필요합니다. slpd가 활성 검색을 수행한 다음 DA가 표시될 수 있습니다. DA가 slpd 범위를 지원하는 경우 SLP 데몬은 해당 호스트의 모든 알림을 DA에 등록합니다.

slpd가 DA를 검색하는 방법 중 한 가지는 부트할 때 DA가 보내는 요청하지 않은 초기 알림을 사용하는 것입니다. SLP 데몬은 주기적으로 요청하지 않은 알림(하트비트)을 사용하여 DA를 계속 활성화할지 여부를 결정합니다. 하트비트가 표시되지 않는 경우 데몬은 데몬이 사용하는 DA 및 데몬이 UA에 제공하는 DA를 제거합니다.

마지막으로 DA에 제어된 종료가 발생하면 DA는 서비스를 중단할 수신 대기 SA 서비스를 알려주는 특수 DA 알림을 전송합니다. 또한 SLP 데몬은 이 알림을 사용하여 캐시에서 비활성 DA를 제거합니다.

네트워크가 자주 분할되고 SA의 수명이 길면 slpd는 하트비트 알림을 받지 않은 경우 분할 도중 캐시된 DA를 제거할 수 있습니다. 하트비트 시간을 줄이면 분할이 복구된 다음 비활성화된 DA가 캐시에 복원되기 전에 지연을 줄일 수 있습니다.

## ▼ 자주 분할하는 경우를 위한 DA 하트비트를 구성하는 방법

다음 절차를 수행하여 DA 하트비트 주기를 줄이도록 net.slp.DAHeartBeat 등록 정보를 변경할 수 있습니다.

---

주 - DA 검색이 완전히 사용 안함으로 설정된 경우 올바른 DA에 액세스할 수 있도록 UA 및 SA를 실행 중인 호스트의 slp.conf에서 net.slp.DAAddresses 등록 정보를 설정해야 합니다.

---

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 호스트에서 slpd 및 모든 SLP 작업을 중지합니다.

```
# svcadm disable network/slp
```

### 3 구성 설정을 변경하기 전에 기본 /etc/inet/slp.conf 파일을 백업합니다.

- 4 `net.slp.DAHeartBeat` 값을 1시간(3600초)으로 줄입니다. 기본적으로 DA 하트비트 주기는 3시간(10800초)으로 설정됩니다.

```
net.slp.DAHeartBeat=3600
```

- 5 변경 사항을 저장하고 파일을 닫습니다.

- 6 `slpd`를 다시 시작하여 변경 사항을 활성화합니다.

```
# svcadm enable network/slp
```

## 네트워크 혼잡 줄이기

네트워크가 매우 혼잡한 경우 멀티캐스트 작업 양을 제한할 수 있습니다. DA가 아직 네트워크에 배포되어 있지 않은 경우 DA를 배포하여 SLP 관련 멀티캐스트의 양을 크게 줄일 수 있습니다.

그러나 DA를 배포한 후에도 DA 검색에 멀티캐스트가 여전히 필요할 수 있습니다. 232 페이지 “다이얼 업 네트워크에 대한 DA 검색을 구성하는 방법”에 설명된 메소드를 사용하여 DA 검색에 필요한 멀티캐스트의 양을 줄일 수 있습니다. 230 페이지 “UA 및 SA를 정적으로 구성된 DA로 제한”에 설명된 메소드를 사용하여 DA 검색에 대한 멀티캐스트를 완전히 제거할 수 있습니다.

## 다른 네트워크 매체, 토폴로지 또는 구성 수용

이 절에서는 다음 등록 정보를 변경하여 SLP 성능을 조정할 수 있는 가능한 시나리오에 대해 설명합니다.

표 9-3 SLP 성능 등록 정보

등록 정보	설명
<code>net.slp.DAAttributes</code>	DA가 알림을 수락하는 최소 새로 고침 간격입니다.
<code>net.slp.multicastTTL</code>	멀티캐스트 패킷에 대해 지정된 <b>활성 시간</b> 값입니다.
<code>net.slp.MTU</code>	네트워크 패킷의 바이트 크기 세트입니다. 크기는 IP 및 TCP 또는 UDP 헤더를 포함합니다.
<code>net.slp.isBroadcastOnly</code>	브로드캐스트를 DA 및 비DA 기반 서비스 검색에 사용해야 하는지 여부를 나타내도록 설정된 부울입니다.

## SA 재등록 줄이기

수명이 만료되기 전에 SA는 해당 서비스 알림을 주기적으로 새로 고쳐야 합니다. DA가 많은 수의 UA 및 SA에서 상당히 많은 로드를 처리 중인 경우 잦은 새로 고침으로 인해 DA에 과부하가 발생할 수 있습니다. DA에 과부하가 발생하면 UA 요청의 시간 초과가 시작된 다음 연결이 끊깁니다. UA 요청 시간 초과는 여러 가지 원인으로 인해 발생할 수 있습니다. DA 과부하가 문제라고 가정하기 전에 `snoop` 추적을 사용하여 서비스 등록을 통해 등록된 서비스 알림의 수명을 확인하십시오. 수명이 짧고 재등록이 자주 발생하는 경우 잦은 재등록으로 인해 시간 초과가 발생할 수 있습니다.

주 - FRESH 플래그를 설정하지 않은 경우 서비스 등록은 **재등록**입니다. 서비스 등록 메시지에 대한 자세한 내용은 11 장, “SLP(참조)”를 참조하십시오.

### ▼ SA 재등록을 줄이는 방법

다음 절차를 수행하여 재등록을 줄이도록 SA에 대한 최소 새로 고침 간격을 늘립니다.

#### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

#### 2 호스트에서 `slpd` 및 모든 SLP 작업을 중지합니다.

```
# svcadm disable network/slp
```

#### 3 구성 설정을 변경하기 전에 기본 `/etc/inet/slp.conf` 파일을 백업합니다.

#### 4 `net.slp.DAAttributes` 등록 정보의 `min-refresh-interval` 속성 값을 늘립니다.

기본 최소 재등록 주기는 0입니다. 기본값 0을 사용하면 SA가 언제든지 재등록하도록 할 수 있습니다. 다음 예제에서는 간격을 3600초(1시간)로 늘립니다.

```
net.slp.DAAttributes(min-refresh-interval=3600)
```

#### 5 변경 사항을 저장하고 파일을 닫습니다.

#### 6 `slpd`를 다시 시작하여 변경 사항을 활성화합니다.

```
# svcadm enable network/slp
```

## 멀티캐스트 활성 시간 등록 정보 구성

멀티캐스트 활성 시간 등록 정보(`net.slp.multicastTTL`)는 멀티캐스트 패킷이 인터넷에 전파되는 범위를 결정합니다. `net.slp.multicastTTL` 등록 정보를 1부터 255 사이의 정수로 설정하여 멀티캐스트 TTL을 구성합니다. 멀티캐스트 TTL의 기본값은

255입니다. 즉, 이론적으로 패킷 경로 지정은 제한되지 않습니다. 그러나 TTL을 255로 설정하면 멀티캐스트 패킷이 관리 도메인의 가장자리에 있는 경계 라우터에 대한 인터넷에 전파되도록 할 수 있습니다. 경계 라우터의 올바른 멀티캐스트 구성은 멀티캐스트 패킷이 인터넷의 멀티캐스트 백본 또는 ISP로 유출되는 것을 방지하는 데 필요합니다.

멀티캐스트 TTL 범위는 TTL 비교가 만들어진 점을 제외하고는 표준 IP TTL과 유사합니다. 멀티캐스트가 사용으로 설정된 라우터의 각 인터페이스에는 TTL 값이 지정됩니다. 멀티캐스트 패킷이 도착하면 라우터는 패킷 TTL을 인터페이스 TTL과 비교합니다. 패킷 TTL이 인터페이스 TTL보다 크거나 같은 경우 패킷 TTL은 표준 IP TTL과 같이 하나로 줄어듭니다. TTL이 0이 되면 패킷이 무시됩니다. SLP 멀티캐스트에 TTL 범위를 사용하는 경우 라우터를 올바르게 구성하여 패킷을 인터넷의 특정 세부절로 제한해야 합니다.

## ▼ 멀티캐스트 활성 시간 등록 정보를 구성하는 방법

다음 절차를 수행하여 net.slp.multicastTTL 등록 정보를 재설정합니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 호스트에서 slpd 및 모든 SLP 작업을 중지합니다.

```
# svcadm disable network/slp
```

### 3 구성 설정을 변경하기 전에 기본 /etc/inet/slp.conf 파일을 백업합니다.

### 4 다음과 같이 slpd.conf 파일의 net.slp.multicastTTL 등록 정보를 변경합니다.

```
net.slp.multicastTTL=value
```

**값** 멀티캐스트 TTL을 정의하는 255보다 작거나 같은 양의 정수

---

주 - TTL 값을 줄여 멀티캐스트 전파 범위를 줄일 수 있습니다. TTL 값이 1인 경우 패킷은 서브넷으로 제한됩니다. 값이 32인 경우 패킷은 사이트로 제한됩니다. 그러나 **사이트** 용어는 멀티캐스트 TTL이 설명된 RFC 1075에 정의되어 있지 않습니다. 33 이상의 값은 인터넷에서의 이론상 경로 지정을 의미하며 사용하면 안됩니다. 라우터가 TTL을 통해 올바르게 구성된 경우 32 이하의 값을 사용하여 멀티캐스트를 액세스 가능한 서브넷 세트로 제한할 수 있습니다.

---

### 5 변경 사항을 저장하고 파일을 닫습니다.

### 6 slpd를 다시 시작하여 변경 사항을 활성화합니다.

```
# svcadm enable network/slp
```

## 패킷 크기 구성

SLP의 기본 패킷 크기는 1400바이트입니다. 크기는 대부분의 근거리 통신망(LAN)에 충분해야 합니다. 무선 네트워크 또는 WAN(wide area network)의 경우에는 패킷 크기를 줄여 메시지 단편화를 방지하고 네트워크 트래픽을 줄일 수 있습니다. 더 큰 패킷을 가진 근거리 통신망(LAN)의 경우 패킷 크기를 늘리면 성능을 향상시킬 수 있습니다. 네트워크에 대한 최소 패킷 크기를 확인하여 패킷 크기를 줄여야 하는지 여부를 결정할 수 있습니다. 네트워크 매체의 패킷 크기가 더 작은 경우 그에 따라 `net.slp.MTU` 값을 줄일 수 있습니다.

네트워크 매체의 패킷이 더 큰 경우에는 패킷 크기를 늘릴 수 있습니다. 그러나 SA의 서비스 알림 또는 UA의 질의가 기본 패킷 크기를 자주 오버플로우하지 않으면 `net.slp.MTU` 값을 변경하지 않아도 됩니다. `snoop`를 사용하여 UA 요청이 기본 패킷 크기를 오버플로우하는 횟수를 결정하고 UDP가 아닌 TCP를 사용하도록 롤오버할 수 있습니다.

`net.slp.MTU` 등록 정보는 연결 계층 헤더, IP 헤더, UDP 또는 TCP 헤더 및 SLP 메시지를 포함하여 전체 IP 패킷 크기를 측정합니다.

### ▼ 패킷 크기를 구성하는 방법

다음 절차를 통해 `net.slp.MTU` 등록 정보를 조정하여 기본 패킷 크기를 변경합니다.

#### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

#### 2 호스트에서 `slpd` 및 모든 SLP 작업을 중지합니다.

```
# svcadm disable network/slp
```

#### 3 구성 설정을 변경하기 전에 기본 `/etc/inet/slp.conf` 파일을 백업합니다.

#### 4 다음과 같이 `slpd.conf` 파일의 `net.slp.MTU` 등록 정보를 변경합니다.

```
net.slp.MTU=value
```

*value*      네트워크 패킷 크기(바이트)를 지정하는 16비트 정수

기본값=1400

값 범위=128-8192

#### 5 변경 사항을 저장하고 파일을 닫습니다.

6 **slpd**를 다시 시작하여 변경 사항을 활성화합니다.

```
# svcadm enable network/slp
```

## 브로드캐스트 전용 경로 지정 구성

SLP는 DA가 없는 서비스 검색 및 DA 검색에 멀티캐스트를 사용하도록 설계되었습니다. 네트워크에서 멀티캐스트 경로 지정을 배포하지 않는 경우 SLP를 구성하여 `net.slp.isBroadcastOnly` 등록 정보를 True로 설정하면 브로드캐스트를 사용할 수 있습니다.

멀티캐스트와는 달리 브로드캐스트 패킷은 기본적으로 서브넷에 전파되지 않습니다. 이러한 이유로 인해 비멀티캐스트 네트워크의 DA가 없는 서비스 검색은 단일 서브넷에서만 작동합니다. 또한 브로드캐스트를 사용하는 네트워크에서 DA 및 범위를 배포할 경우에는 특별한 고려 사항이 요구됩니다. 멀티홈 호스트의 DA는 멀티캐스트가 사용 안함으로 설정된 다중 서브넷 간에 서비스 검색을 연결할 수 있습니다. 멀티홈 호스트에서의 DA 배포에 대한 자세한 내용은 [252 페이지 “DA 배치 및 범위 이름 지정”](#)을 참조하십시오.

### ▼ 브로드캐스트 전용 경로 지정을 구성하는 방법

다음 절차를 수행하여 `net.slp.isBroadcastOnly` 등록 정보를 True로 변경합니다.

1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

2 호스트에서 **slpd** 및 모든 SLP 작업을 중지합니다.

```
# svcadm disable network/slp
```

3 구성 설정을 변경하기 전에 기본 `/etc/inet/slp.conf` 파일을 백업합니다.

4 다음과 같이 `slpd.conf` 파일의 `net.slp.isBroadcastOnly` 등록 정보를 True로 변경합니다.

```
net.slp.isBroadcastOnly=True
```

5 변경 사항을 저장하고 파일을 닫습니다.

6 **slpd**를 다시 시작하여 변경 사항을 활성화합니다.

```
# svcadm enable network/slp
```

## SLP 검색 요청에 대한 시간 초과 수정

다음과 같은 두 가지 경우 SLP 검색 요청에 대한 시간 초과를 변경해야 할 수 있습니다.

- SLP 에이전트가 다중 서브넷, 다이얼 업 회선 또는 기타 WAN에 의해 분리된 경우 네트워크 대기 시간이 길어서 기본 시간 초과로는 요청 또는 등록을 완료하기에 부족할 수 있습니다. 반대로 네트워크 대기 시간이 짧은 경우 시간 초과를 줄여 성능을 향상시킬 수 있습니다.
- 네트워크의 트래픽이 심하거나 충돌률이 높은 경우 메시지를 보내기 전에 SA 및 UA가 기다려야 하는 최대 기간이 충돌 자유 트랜잭션을 보장하는 데 부족할 수 있습니다.

### 기본 시간 초과 변경

네트워크 대기 시간이 높으면 요청 및 등록에 대한 응답이 반환되기 전에 UA 및 SA가 시간 초과될 수 있습니다. UA가 SA와 분리되거나 UA 및 SA 모두 DA와 분리된 경우 또는 다중 서브넷, 다이얼 업 회선 또는 WAN으로 분리된 경우 대기 시간으로 인해 문제가 발생할 수 있습니다. UA 및 SA 요청/등록의 시간 초과 때문에 SLP 요청이 실패하는지 여부를 확인하여 대기 시간이 문제인지 여부를 알 수 있습니다. 또한 ping 명령을 사용하여 실제 대기 시간을 측정할 수도 있습니다.

다음 표에는 시간 초과를 제어하는 구성 등록 정보가 나열되어 있습니다. 이 절에 나와 있는 절차를 수행하여 이러한 등록 정보를 수정할 수 있습니다.

표 9-4 시간 초과 등록 정보

등록 정보	설명
net.slp.multicastTimeouts net.slp.DADiscoveryTimeouts net.slp.datagramTimeouts	전송이 중단되기 전에 반복되는 멀티캐스트 및 유니캐스트 UDP 메시지 전송에 대한 시간 초과를 제어하는 등록 정보입니다.
net.slp.multicastMaximumWait	중단되기 전에 멀티캐스트 메시지가 전송되는 시간의 최대 양을 제어하는 등록 정보입니다.
net.slp.datagramTimeouts	이 등록 정보에 대해 나열된 값의 합으로 지정된 DA 시간 초과와 상한입니다. UDP 데이터그램은 응답을 받거나 시간 초과 한도에 도달할 때까지 DA에 반복적으로 전송됩니다.

멀티캐스트 서비스 검색 또는 DA 검색 도중 잦은 시간 초과가 발생하는 경우 net.slp.multicastMaximumWait 등록 정보의 기본값 15000밀리초(15초)를 늘립니다. 최대 대기 기간을 늘리면 완료할 높은 대기 시간 네트워크에 대한 요청에 더 많은 시간을 허용할 수 있습니다. net.slp.multicastMaximumWait를 변경한 다음

`net.slp.multicastTimeouts` 및 `net.slp.DADiscoveryTimeouts`도 수정해야 합니다. 이러한 등록 정보에 대한 시간 초과 값의 합은 `net.slp.multicastMaximumWait` 값과 같습니다.

## ▼ 기본 시간 초과를 변경하는 방법

다음 절차를 수행하여 시간 초과를 제어하는 SLP 등록 정보를 변경합니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 호스트에서 `slpd` 및 모든 SLP 작업을 중지합니다.

```
# svcadm disable network/slp
```

### 3 구성 설정을 변경하기 전에 기본 `/etc/inet/slp.conf` 파일을 백업합니다.

### 4 다음과 같이 `slpd.conf` 파일의 `net.slp.multicastMaximumWait` 등록 정보를 변경합니다.

```
net.slp.multicastMaximumWait=value
```

**값** `net.slp.multicastTimeouts` 및 `net.slp.DADiscoveryTimeouts`에 설정된 값의 합을 나열하는 32비트 정수

기본값=15000밀리초(15초)

값 범위=1000-60000밀리초

예를 들어, 멀티캐스트 요청에 20초(20000밀리초)가 필요하다고 결정한 경우

`net.slp.multicastTimeouts` 및 `net.slp.DADiscoveryTimeouts` 등록 정보에 나열된 값을 20000밀리초와 함께 조정합니다.

```
net.slp.multicastMaximumWait=20000
net.slp.multicastTimeouts=2000,5000,6000,7000
net.slp.DADiscoveryTimeouts=3000,3000,6000,8000
```

### 5 필요한 경우 다음과 같이 `slpd.conf` 파일의 `net.slp.datagramTimeouts` 등록 정보를 변경합니다.

```
net.slp.datagramTimeouts=value
```

**값** DA에 대한 유니캐스트 데이터그램 전송을 구현하기 위해 시간 초과(밀리초)를 지정하는 32비트 정수 목록

기본값=3000,3000,3000



예를 들어, 데이터그램 시간 초과를 20000밀리초로 늘려 잦은 시간 초과를 방지할 수 있습니다.

```
net.slp.datagramTimeouts=2000,5000,6000,7000
```

고성능 네트워크에서는 멀티캐스트 및 유니캐스트 UDP 데이터그램 전송에 대한 시간 초과 한도를 줄일 수 있습니다. 시간 초과 한도를 줄이면 SLP 요청을 충족시키는 데 필요한 대기 시간이 줄어듭니다.

- 6 변경 사항을 저장하고 파일을 닫습니다.
- 7 **slpd**를 다시 시작하여 변경 사항을 활성화합니다.  
# svcadm enable network/slp

## 임의 대기 한도 구성

네트워크의 트래픽이 심하거나 충돌률이 높은 경우에는 DA를 통한 통신에 영향을 받을 수 있습니다. 충돌률이 높으면 전송 에이전트가 UDP 데이터그램을 재전송해야 합니다. **snoop**를 사용하여 재전송을 실행할지 여부를 결정하여 **slpd**를 SA 서버로 실행 중인 호스트 및 **slpd**를 DA로 실행 중인 호스트의 네트워크에서 트래픽을 모니터링할 수 있습니다. 같은 서비스에 대한 여러 서비스 등록 메시지가 **slpd**를 SA 서버로 실행 중인 호스트의 **snoop** 추적에 나타나는 경우 충돌 알림을 받을 수 있습니다.

충돌은 부트 시의 특정 문제일 수 있습니다. DA를 처음으로 시작할 때 요청하지 않은 알림 및 등록에 대한 SA 응답을 보냅니다. SLP에서는 SA가 DA 알림을 받은 다음 응답하기 전에 임의의 시간을 대기해야 합니다. 임의 대기 한도는 `net.slp.randomWaitBound`에서 제어하는 최대 값으로 균등 분산됩니다. 기본 임의 대기 한도는 1000밀리초(1초)입니다.

## ▼ 임의 대기 한도를 구성하는 방법

다음 절차를 수행하여 `slp.conf` 파일의 `net.slp.RandomWaitBound` 등록 정보를 변경합니다.

- 1 관리자가 됩니다.  
자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.
- 2 호스트에서 **slpd** 및 모든 SLP 작업을 중지합니다.  
# svcadm disable network/slp
- 3 구성 설정을 변경하기 전에 기본 `/etc/inet/slp.conf` 파일을 백업합니다.

#### 4 다음과 같이 `slpd.conf` 파일의 `net.slp.RandomWaitBound` 등록 정보를 변경합니다.

```
net.slp.RandomWaitBound=value
```

**값** DA에 대한 연결을 시도하기 전의 임의 대기 시간을 계산하기 위한 상한

기본값=1000밀리초(1초)

값 범위=1000-3000밀리초

예를 들어 최대 대기를 2000밀리초(2초)로 연장할 수 있습니다.

```
net.slp.randomWaitBound=2000
```

임의 대기 한도를 연장하면 등록 시 더 긴 지연이 발생합니다. SA는 새로 검색된 DA에 대한 등록을 더 느리게 완료하여 충돌 및 시간 초과를 방지할 수 있습니다.

#### 5 필요한 경우 다음과 같이 `slpd.conf` 파일의 `net.slp.datagramTimeouts` 등록 정보를 변경합니다.

```
net.slp.datagramTimeouts=value
```

**value** DA에 대한 유니캐스트 데이터그램 전송을 구현하기 위해 시간 초과(밀리초)를 지정하는 32비트 정수 목록

기본값=3000,3000,3000

예를 들어, 데이터그램 시간 초과를 20000밀리초로 늘려 잦은 시간 초과를 방지할 수 있습니다.

```
net.slp.datagramTimeouts=2000,5000,6000,7000
```

고성능 네트워크에서는 멀티캐스트 및 유니캐스트 UDP 데이터그램 전송에 대한 시간 초과 한도를 줄일 수 있습니다. 이 설정은 SLP 요청에 충족하는 대기 시간의 양을 줄입니다.

#### 6 변경 사항을 저장하고 파일을 닫습니다.

#### 7 `slpd`를 다시 시작하여 변경 사항을 활성화합니다.

```
# svcadm enable network/slp
```

## 범위 배포

범위를 사용하면 사용자의 논리적, 물리적 및 관리 그룹을 기반으로 하는 서비스를 프로비전할 수 있습니다. 범위를 사용하여 서비스 알림에 대한 액세스를 관리할 수 있습니다.

`net.slp.useScopes` 등록 정보를 사용하여 범위를 만듭니다. 예를 들어, 다음과 같이 호스트의 `/etc/inet/slp.conf` 파일에서 `newscope`라고 하는 새 범위를 추가합니다.

```
net.slp.useScopes=newscope
```

예를 들어, 조직의 6동 건물 2층 남쪽 홀의 끝에 있는 프린터 및 팩스와 같이 네트워크로 연결된 장치를 위한 공간이 있을 수 있습니다. 이러한 장치는 2층의 모든 사람이 사용할 수 있도록 하거나 특정 부서의 구성원만 사용하도록 제한할 수도 있습니다. 범위는 이러한 장치의 서비스 알림에 대한 액세스를 프로비전하는 방법을 제공합니다.

장치가 단일 부서 전용인 경우 부서 이름(예:mktg)으로 범위를 만들 수 있습니다. 다른 부서에 속한 장치는 다른 범위 이름을 사용하여 구성할 수 있습니다.

다른 시나리오에서 부서가 분산될 수 있습니다. 예를 들어, 기계 엔지니어링 및 CAD/CAM 부서가 1층과 2층으로 나뉘어 있을 수 있습니다. 그러나 같은 범위에 장치를 지정하여 두 층 모두의 호스트에 2층 기계를 제공할 수 있습니다. 네트워크 및 사용자에 맞는 방법으로 범위를 배포할 수 있습니다.

---

주 - 특정 범위가 있는 UA는 실제로 다른 범위에 알려진 서비스의 사용으로부터 보호되지 않습니다. 범위 구성은 UA가 감지하는 서비스 알림만 제어합니다. 서비스는 모든 액세스 제어 제한 실행을 위한 것입니다.

---

## 범위 구성 시기

SLP는 어떠한 범위 구성 없이도 적절히 작동할 수 있습니다. Oracle Solaris 동작 환경에서 SLP의 기본 범위는 default입니다. 범위가 구성되지 않은 경우 default가 모든 SLP 메시지의 범위입니다.

다음과 같은 경우에 범위를 구성할 수 있습니다.

- 지원하는 조직에서 고유의 구성원에 대한 서비스 알림 액세스를 제한하려 합니다.
- 지원하는 조직의 물리적 레이아웃에서 특정 사용자가 특정 영역의 서비스에 액세스할 수 있도록 제안합니다.
- 표시할 특정 사용자에게 대해 적절한 서비스 알림이 분할되어야 합니다.

첫번째 경우의 예는 231 페이지 “다이얼 업 네트워크에 대한 DA 검색 구성”에 나와 있습니다. 두번째 예는 조직이 두 건물로 나뉘어 있고 한 건물의 사용자를 해당 건물의 로컬 서비스에 액세스하도록 하려는 경우에 해당합니다. B2 범위를 사용하여 2동 건물의 사용자를 구성하는 동안 B1 범위를 사용하여 1동 건물의 사용자를 구성할 수 있습니다.

## 범위 구성 시 고려 사항

slpd.conf 파일의 net.slp.useScopes 등록 정보를 수정하는 경우 호스트의 모든 에이전트에 대한 범위를 구성합니다. 호스트가 SA를 실행 중이거나 DA로 작동 중인 경우 default가 아닌 범위로 SA 또는 DA를 구성하려면 이 등록 정보를 구성해야 합니다. UA가

시스템에서 실행 중이고 UA에서 **default** 이외의 범위를 지원하는 SA 및 DA를 검색해야 하는 경우 UA가 사용하는 범위를 제한하지 않으려면 등록 정보를 구성할 필요는 없습니다. 등록 정보가 구성되지 않은 경우 UA는 **slpd**를 통해 자동으로 사용 가능한 DA 및 범위를 검색합니다. SLP 데몬은 활성 및 수동 DA 검색을 통해 DA를 검색하거나 DA가 실행 중이지 않은 경우 SA 검색을 사용합니다. 또는 등록 정보가 구성되지 않은 경우 UA는 구성된 범위만 사용하고 해당 범위를 무시하지 않습니다.

범위를 구성하려는 경우 네트워크의 모든 SA에 구성된 범위가 있는지 확인하지 않으면 구성된 범위 목록에서 **default** 범위를 유지하도록 고려해야 합니다. SA가 구성되지 않은 경우 구성된 범위가 있는 UA에서는 범위를 검색할 수 없습니다. 구성되지 않은 SA는 자동으로 범위 **default**를 가지지만 UA에는 구성된 범위가 있기 때문에 이러한 상황이 발생합니다.

**net.slp.DAAddresses** 등록 정보를 설정하여 DA를 구성하도록 선택한 경우에도 구성된 DA에서 지원하는 범위가 **net.slp.useScopes** 등록 정보를 통해 구성한 범위와 같은지 확인해야 합니다. 범위가 다른 경우 **slpd**는 다시 시작할 때 오류 메시지가 표시됩니다.

## ▼ 범위를 구성하는 방법

다음 절차를 수행하여 **slp.conf** 파일의 **net.slp.useScopes** 등록 정보에 범위 이름을 추가합니다.

### 1 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

### 2 호스트에서 **slpd** 및 모든 SLP 작업을 중지합니다.

```
# svcadm disable network/slp
```

### 3 구성 설정을 변경하기 전에 기본 **/etc/inet/slp.conf** 파일을 백업합니다.

### 4 다음과 같이 **slpd.conf** 파일의 **net.slp.useScopes** 등록 정보를 변경합니다.

```
net.slp.useScopes=<scope names>
```

**scope names**      요청을 만들 때 DA 또는 SA에서 사용하도록 허용된 범위나 DA에서 지원해야 하는 범위를 나타내는 문자열 목록

기본값=SA 및 DA에 대한 기본값/UA에 대해 지정되지 않음

---

주-

다음을 사용하여 범위 이름을 구성합니다.

- 영숫자, 대/소문자
- 문장 부호 문자(예외: ", \, !, <, =, > 및 ~)
- 이름 부분을 나타내는 공백
- 비ASCII 문자

백슬래시를 사용하여 비ASCII 문자를 제어합니다. 예를 들어, UTF-8 인코딩에서는 0xc3a9 16진수 코드를 사용하여 프랑스어 *aigue* 악센트가 있는 문자 *e*를 표현합니다. 플랫폼에서 UTF-8을 지원하지 않는 경우 UTF-8 16진수 코드를 이스케이프 시퀀스 \c3\a9로 사용합니다.

예를 들어, bldg6의 eng 및 mktg 그룹에 범위를 지정하려면 net.slp.useScopes 행을 다음과 같이 변경합니다.

```
net.slp.useScopes=eng,mktg,bldg6
```

- 5 변경 사항을 저장하고 파일을 닫습니다.
- 6 slpd를 다시 시작하여 변경 사항을 활성화합니다.  
# svcadm enable network/slp

## DA 배포

이 절에서는 SLP를 실행 중인 네트워크에서의 전략적 DA 배포에 대해 설명합니다.

SLP는 기본 에이전트(DA 및 SA)만 있고 배포된 DA나 구성된 범위가 없어도 올바르게 작동합니다. 특정 구성이 부족한 모든 에이전트는 default 범위를 사용합니다. DA는 서비스 알림에 대한 캐시로 작동합니다. DA를 배포하면 네트워크에서 전송된 메시지 수가 줄어들고 메시지에 대한 응답을 받는 데 필요한 시간이 줄어듭니다. 이러한 기능을 사용하면 SLP가 더 큰 네트워크를 수용할 수 있습니다.

## SLP DA를 배포하는 이유

DA를 배포하는 기본적인 이유는 멀티캐스트 트래픽의 양과 유니캐스트 회신 수집과 관련된 지연을 줄이기 위해서입니다. 여러 UA 및 SA가 있는 큰 규모의 네트워크에서는 네트워크 성능 저하로 인해 서비스 검색에 포함된 멀티캐스트 트래픽의 양이 커질 수 있습니다. 하나 이상의 DA를 배포하면 UA는 서비스에 대한 DA로 유니캐스트해야 하며 SA는 유니캐스트를 사용하여 DA에 등록해야 합니다. DA가 있는 네트워크의 SLP 등록 멀티캐스트만 활성화 및 수동 DA 검색을 위한 것입니다.

SA는 멀티캐스트 서비스 요청을 수락하는 대신 공통 범위 세트 내에서 SA가 검색하는 모든 DA를 자동으로 등록합니다. 그러나 DA에서 지원하지 않는 범위의 멀티캐스트 요청은 SA에 의해 직접 응답됩니다.

UA의 서비스 요청은 UA의 범위 내에서 DA를 배포할 때 네트워크에 대한 멀티캐스트가 아닌 DA로 유니캐스트됩니다. 따라서 UA의 범위 내의 DA는 멀티캐스트를 줄입니다. 일반 UA 요청에 대한 멀티캐스트를 제거하면 질의에 대한 회신을 가져오는 데 필요한 시간이 초 단위에서 밀리초 단위로 크게 줄어듭니다.

DA는 SA 및 UA 작업에 대한 중심점으로 작동합니다. 범위 모음에 대해 하나 또는 여러 DA를 배포하면 SLP 작업을 모니터링하기 위한 중앙화된 지점이 제공됩니다. DA 로깅을 활성화시키면 네트워크 주변에 산재되어 있는 여러 SA에서 로그를 확인하는 것보다 등록 및 요청을 모니터링하는 것이 더 쉬워집니다. 로드 균형 조정 필요 여부에 따라 하나의 특정 범위 또는 여러 범위에 대해 DA를 몇 개든지 배포할 수 있습니다.

사용으로 설정된 멀티캐스트 경로 지정이 없는 네트워크에서는 SLP를 구성하여 브로드캐스트를 사용할 수 있습니다. 그러나 브로드캐스트는 메시지를 처리하기 위해 각 호스트가 필요하므로 매우 비효율적입니다. 또한 브로드캐스트는 일반적으로 라우터를 통해 전파되지 않습니다. 결과적으로 멀티캐스트 경로 지정이 지원되지 않는 네트워크에서는 같은 서브넷에서만 서비스를 검색할 수 있습니다. 멀티캐스트 경로 지정에 대한 부분 지원으로 인해 네트워크에서 서비스를 검색하는 데 일치하지 않는 기능이 발생할 수 있습니다. 멀티캐스트 메시지는 DA를 검색하는 데 사용됩니다. 따라서 멀티캐스트 경로 지정에 대한 부분 지원은 SA의 범위에서 알려진 모든 DA가 있는 UA 및 SA 등록 서비스로 간주됩니다. 예를 들어, UA에서 DA1이라고 하는 DA를 질의하고 SA가 DA2에 서비스를 등록한 경우 UA는 서비스를 검색할 수 없습니다. 사용으로 설정된 멀티캐스트가 없는 네트워크에서 SLP를 배포하는 방법에 대한 자세한 내용은 [238 페이지 “브로드캐스트 전용 경로 지정 구성”](#)을 참조하십시오.

멀티캐스트 경로 지정에 대해 일치하지 않는 사이트 차원 지원이 있는 네트워크에서는 `net.slp.DAAdresses` 등록 정보를 사용하여 DA 위치의 일치 목록이 있는 SLP UA 및 SA를 구성해야 합니다.

마지막으로 SLPv2 DA는 SLPv1과의 상호 운용성을 지원합니다. SLPv1 상호 운용성은 DA에서 기본적으로 사용으로 설정됩니다. 네트워크에 프린터와 같은 SLPv1 장치가 포함되어 있는 경우 또는 서비스 검색을 위해 SLPv1을 사용하는 Novell Netware 5와 상호 운용해야 하는 경우 DA를 배포해야 합니다. DA가 없으면 Oracle Solaris SLP UA는 SLPv1 알림 서비스를 검색할 수 없습니다.

## DA 배포 시기

다음 조건을 충족하는 경우 엔터프라이즈에 DA를 배포합니다.

- 멀티캐스트 SLP 트래픽이 snoop에 의해 측정된 네트워크의 대역폭이 1퍼센트를 초과합니다.
- 멀티캐스트 서비스 요청 도중 UA 클라이언트에 긴 지연 또는 시간 초과가 발생합니다.
- 하나 또는 여러 호스트에서 특정 범위에 대한 SLP 서비스 알림의 모니터링을 집중화하려고 합니다.
- 네트워크에 멀티캐스트가 사용으로 설정되어 있지 않고 네트워크가 서비스를 공유하는 다중 서브넷으로 구성되어 있습니다.
- 네트워크에서 이전 버전의 SLP(SLPv1)를 지원하는 장치를 사용하고 있거나 SLP 서비스 검색이 Novell Netware 5와 상호 운영되게 하려고 합니다.

## ▼ DA를 배포하는 방법

다음 절차를 수행하여 slp.conf 파일의 net.slp.isDA 등록 정보를 True로 설정합니다.

---

주 - 호스트당 하나의 DA만 지정할 수 있습니다.

---

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 호스트에서 slpd 및 모든 SLP 작업을 중지합니다.

```
# svcadm disable network/slp
```

### 3 구성 설정을 변경하기 전에 기본 /etc/inet/slp.conf 파일을 백업합니다.

### 4 다음과 같이 slpd.conf 파일의 net.slp.isDA 등록 정보를 True로 설정합니다.

```
net.slp.isDA=True
```

### 5 변경 사항을 저장하고 파일을 닫습니다.

### 6 slpd를 다시 시작하여 변경 사항을 활성화합니다.

```
# svcadm enable network/slp
```

## DA 배치 위치

이 절에서는 여러 상황에서의 DA 배치 위치에 대해 제안합니다.

- 멀티캐스트 경로 지정이 사용으로 설정되어 있지 않고 서브넷 간에 서비스 검색을 연결하는 데 DA가 필요한 경우

이 경우 인터페이스를 사용하는 호스트 및 서비스를 공유하는 모든 서브넷에 DA를 배치해야 합니다. IP 패킷 경로가 인터페이스 전체에 지정되어 있지 않으면 `net.slp.interfaces` 구성 등록 정보를 설정할 필요가 **없습니다**. `net.slp.interfaces` 등록 정보 구성에 대한 자세한 내용은 [249 페이지](#) “SLP에 대한 멀티홈 구성”을 참조하십시오.

- 확장성을 위해 DA를 배포하는 경우 및 주 고려 사항이 에이전트 액세스 최적화인 경우

일반적으로 UA는 서비스에 대한 여러 요청을 DA에 생성합니다. SA는 DA를 한 번만 사용하여 등록하고 간헐적 간격이 아닌 주기적으로 알림을 새로 고칠 수 있습니다. 결과적으로 DA에 액세스하는 UA는 SA 액세스보다 더 반복됩니다. 또한 일반적으로 서비스 알림의 수가 요청 수보다 더 작습니다. 결국 대부분의 DA 배포는 배포가 UA 액세스에 최적화된 경우 더 효율적입니다.

- UA 액세스를 최적화하기 위해 네트워크에서 DA가 위치적으로 UA에 근접하도록 DA 배치

물론 UA 및 SA 클라이언트 모두에서 공유하는 범위로 DA를 구성해야 합니다.

## 로드 균형 조정을 위한 다중 DA 배치

로드 균형 조정의 수단으로서 같은 범위 모음에 여러 DA를 배포할 수 있습니다. 다음과 같은 경우에 DA를 배포합니다.

- DA에 대한 UA 요청이 시간 초과되거나 `DA_BUSY_NOW` 오류가 반환됩니다.
- DA 로그에 여러 SLP 요청이 중단되었다고 표시됩니다.
- 범위에서 서비스를 공유하는 사용자의 네트워크가 여러 건물 또는 물리적 위치에 해당합니다.

SLP 트래픽의 `snoop` 추적을 실행하여 UA 요청이 반환하는 `DA_BUSY_NOW` 오류 횟수를 확인할 수 있습니다. 반환된 UA 요청 수가 많은 경우 DA로부터 물리적 및 위치상 거리에서 건물 내의 UA는 느린 응답 또는 과도한 시간 초과를 발생시킬 수 있습니다. 이러한 시나리오에서는 각 건물에 DA를 배포하여 건물 내의 UA 클라이언트에 대한 응답을 향상시킬 수 있습니다.

건물을 연결하는 링크는 종종 건물 내의 근거리 통신망(LAN)보다 더 느려집니다. 네트워크가 여러 건물 또는 물리적 위치에 해당되는 경우 UA가 지정한 DA에만 액세스하도록 `/etc/inet/slp.conf` 파일의 `net.slp.DAAddresses` 등록 정보를 특정 호스트 이름 또는 주소 목록으로 설정합니다.



특정 DA가 서비스 등록에 대해 더 큰 호스트 메모리 양을 사용 중인 경우 DA가 지원하는 범위의 수를 줄여 SA 등록의 수를 줄입니다. 여러 등록이 있는 범위를 두 개로 분할할 수 있습니다. 그런 다음 다른 호스트에 다른 DA를 배포하여 새 범위 중 하나를 지원할 수 있습니다.

## SLP 및 멀티홈

멀티홈 서버는 다중 IP 서브넷의 호스트로 동작합니다. 서버는 두 개 이상의 네트워크 인터페이스 카드를 가질 수도 있으며 라우터로 동작할 수 있습니다. 멀티캐스트 패킷을 비롯한 IP 패킷은 인터페이스 간에 경로를 지정합니다. 일부 경우에는 인터페이스 간의 경로 지정이 사용 안함으로 설정됩니다. 다음 절에서는 이러한 경우에 SLP를 구성하는 방법에 대해 설명합니다.

### SLP에 대한 멀티홈 구성

구성이 없는 경우 `slpd`는 기본 네트워크 인터페이스에서 멀티캐스트 및 UDP/TCP를 수신 대기합니다. 유니캐스트 및 멀티캐스트 경로 지정이 멀티홈 시스템의 인터페이스 간에 사용으로 설정되어 있는 경우 추가 구성이 필요하지 않습니다. 이는 다른 인터페이스에 도착하는 멀티캐스트 패킷의 경로가 기본값으로 올바르게 지정되기 때문입니다. 결과적으로 DA 또는 다른 서비스 알림에 대한 멀티캐스트 요청은 `slpd`에 도착합니다. 일부 이유로 인해 경로 지정이 활성화되지 않은 경우 구성이 필요합니다.

### 경로가 지정되지 않은 다중 네트워크 인터페이스 구성 시기

다음 조건 중 하나에 해당되는 경우 멀티홈 시스템을 구성해야 합니다.

- 인터페이스 간에 유니캐스트 경로 지정이 사용으로 설정되어 있고 멀티캐스트 경로 지정이 사용 안함으로 설정되어 있습니다.
- 인터페이스 간에 유니캐스트 경로 지정 및 멀티캐스트 경로 지정이 모두 사용 안함으로 설정되어 있습니다.

인터페이스 간에 멀티캐스트 경로 지정이 사용 안함으로 설정되어 있는 경우 이는 일반적으로 네트워크에 멀티캐스트가 배포되지 않았기 때문입니다. 이러한 상황에서는 일반적으로 브로드캐스트가 DA 기반이 아닌 서비스 검색과 개별 서브넷의 DA 검색에 사용됩니다. `net.slp.isBroadcastOnly` 등록 정보를 `True`로 설정하여 브로드캐스트를 구성합니다.

## 경로가 지정되지 않은 다중 네트워크 인터페이스 구성(작업 맵)

표 9-5 경로가 지정되지 않은 다중 네트워크 인터페이스 구성

작업	설명	수행 방법
net.slp.interfaces 등록 정보 구성	이 등록 정보를 설정하면 slpd가 지정된 인터페이스에서 유니캐스트 및 멀티캐스트/브로드캐스트 SLP 요청을 수신 대기하도록 할 수 있습니다.	250 페이지 “net.slp.interfaces 등록 정보 구성”
서브넷의 UA가 도달 가능한 주소가 있는 서비스 URL을 얻도록 프록시 서비스 알림 정렬	프록시 알림을 멀티홈 호스트가 아닌 단일 서브넷에 연결된 slpd를 실행하는 시스템으로 제한합니다.	252 페이지 “멀티홈 호스트에서 프록시 알림”
UA 및 SA 간의 도달 가능성을 보장하도록 DA 배치 및 범위 구성	단일 인터페이스 호스트 이름 또는 주소가 있는 멀티홈 호스트에서 net.slp.interfaces 등록 정보를 구성합니다.  멀티홈 호스트에서 DA를 실행하지만 각 서브넷의 SA 및 UA가 다른 호스트를 사용하도록 범위를 구성합니다.	252 페이지 “DA 배치 및 범위 이름 지정”

### net.slp.interfaces 등록 정보 구성

net.slp.interfaces 등록 정보를 설정한 경우 slpd는 기본 인터페이스 대신 등록 정보에 나열된 인터페이스에서 유니캐스트 및 멀티캐스트/브로드캐스트 SLP 요청을 수신 대기합니다.

일반적으로 멀티캐스트는 네트워크에 배포되지 않기 때문에 net.slp.isBroadcastOnly 등록 정보를 설정하여 브로드캐스트를 사용으로 설정하는 것과 함께

net.slp.interfaces 등록 정보를 설정합니다. 그러나 멀티캐스트가 배포되었지만 이 특정 멀티홈 호스트에서 경로가 지정되지 않는 경우 멀티캐스트 요청은 두 개 이상의 인터페이스로부터 slpd에 도착할 수 있습니다. 패킷의 경로 지정이 인터페이스에서 제공하는 서브넷을 연결하는 다른 멀티홈 호스트 또는 라우터에서 처리될 때 이러한 상황이 발생할 수 있습니다.

이러한 상황이 발생하면 요청을 보내는 SA 서버 또는 UA는 멀티홈 호스트의 slpd로부터 두 개의 응답을 받습니다. 그런 다음 응답이 클라이언트 라이브러리로 필터링되고 클라이언트에게는 표시되지 않습니다. 그러나 응답은 snoop 추적에는 표시됩니다.

주 -

유니캐스트 경로 지정이 해제된 경우 멀티홈 호스트의 SA 클라이언트에서 알리는 서비스는 모든 서브넷으로부터 도달하지 않을 수 있습니다. 서비스가 도달 가능하지 않은 경우 SA 클라이언트는 다음을 수행할 수 있습니다.

- 각 개별 서브넷에 대해 하나의 서비스 URL을 알립니다.
- 특정 서브넷의 요청이 도달 가능한 URL에 응답되도록 보장합니다.

SA 클라이언트는 도달 가능한 URL을 알리도록 보장하기 위해 어떠한 작업도 하지 않습니다. 그런 다음 경로 지정 없이 멀티홈 호스트를 처리하거나 처리하지 않는 서비스 프로그램은 도달 가능한 URL을 알리도록 보장하는 역할을 합니다.

유니캐스트 경로 지정이 사용 안함으로 설정된 멀티홈 호스트에서 서비스를 배포하기 전에 `snoop`를 사용하여 서비스가 다중 서브넷의 요청을 올바르게 처리하는지 여부를 확인합니다. 또한 멀티홈 호스트에서 DA를 배포하려는 경우 [252 페이지 “DA 배치 및 범위 이름 지정”](#)을 참조하십시오.

## ▼ net.slp.interfaces 등록 정보를 구성하는 방법

다음 절차를 수행하여 `slp.conf` 파일의 `net.slp.interfaces` 등록 정보를 변경합니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 호스트에서 `slpd` 및 모든 SLP 작업을 중지합니다.

```
# svcadm disable network/slp
```

### 3 구성 설정을 변경하기 전에 기본 `/etc/inet/slp.conf` 파일을 백업합니다.

### 4 다음과 같이 `slpd.conf` 파일의 `net.slp.interfaces` 등록 정보를 변경합니다.

```
net.slp.interfaces=value
```

**값** DA 또는 SA가 포트 427에서 멀티캐스트, 유니캐스트 UDP 및 TCP 메시지를 수신 대기해야 하는 네트워크 인터페이스 카드의 IPv4 주소 또는 호스트 이름의 목록

예를 들어, 세 개의 네트워크 카드가 있는 서버 및 해제된 멀티캐스트 경로 지정은 세 개의 서브넷에 연결됩니다. 세 개의 네트워크 인터페이스 IP 주소는 192.147.142.42, 192.147.143.42 및 192.147.144.42입니다. 서브넷 마스크는 255.255.255.0입니다. 다음과 같이 등록 정보를 설정하면 `slpd`는 유니캐스트 및 멀티캐스트/브로드캐스트 메시징에 대한 세 개의 모든 인터페이스를 수신 대기합니다.

```
net.slp.interfaces=192.147.142.42,192.147.143.42,192.147.144.42
```

---

주-net.slp.interfaces 등록 정보에 대한 IP 주소 또는 분석 가능한 호스트 이름을 지정할 수 있습니다.

---

- 5 변경 사항을 저장하고 파일을 닫습니다.
- 6 **slpd**를 다시 시작하여 변경 사항을 활성화합니다.

```
# svcadm enable network/slp
```

## 멀티홈 호스트에서 프록시 알림

다중 인터페이스가 있는 호스트에서 **slpd** 및 프록시 등록을 사용하여 서비스를 알리는 경우 **slpd**에서 알리는 서비스 URL은 도달 가능한 호스트 이름 또는 주소를 포함해야 합니다. 인터페이스 간에 유니캐스트 경로 지정이 사용으로 설정되어 있는 경우 모든 서브넷의 호스트는 다른 서브넷의 호스트에 도달할 수 있습니다. 또한 프록시 등록이 모든 서브넷의 서비스에 만들어질 수 있습니다. 그러나 유니캐스트 경로 지정이 사용 안함으로 설정되어 있는 경우 하나의 서브넷에 있는 서비스 클라이언트는 멀티홈 호스트를 통해 다른 서브넷의 서비스에 도달할 수 없습니다. 하지만 해당 클라이언트는 다른 라우터를 통해 서비스에 도달할 수 있습니다.

예를 들어, 기본 호스트 이름이 **bigguy**인 호스트에 경로가 지정되지 않은 세 개의 다른 서브넷에 대한 세 개의 인터페이스 카드가 있다고 가정합니다. 이러한 서브넷의 호스트 이름은 **bigguy**(IP 주소: 192.147.142.42), **bigguy1**(IP 주소: 192.147.143.42) 및 **bigguy2**(IP 주소: 192.147.144.42)입니다. 지금은 모든 인터페이스에서 수신하도록 기존 프린터인 **oldprinter**가 143 서브넷에 연결되어 있고 URL **service:printing:lpr://oldprinter/queue1**이 **net.slp.interfaces**로 구성되어 있다고 가정합니다. **oldprinter** URL은 모든 인터페이스의 프록시 알림입니다. 142 및 144 서브넷의 시스템은 서비스 요청에 대한 응답으로 URL을 받지만 **oldprinter** 서비스에는 액세스할 수 없습니다.

이 문제를 해결하려면 멀티홈 호스트가 아닌 143 서브넷에 연결된 시스템에서 실행 중인 **slpd**로 프록시 알림을 수행해야 합니다. 143 서브넷의 호스트만 서비스 요청에 대한 응답으로 알림을 가져올 수 있습니다.

## DA 배치 및 범위 이름 지정

멀티홈 호스트가 있는 네트워크의 DA 배치 및 범위 이름 지정은 클라이언트가 액세스 가능한 서비스를 확실히 가져올 수 있도록 주의하여 수행해야 합니다. 경로 지정이 사용 안함으로 설정되어 있고 **net.slp.interfaces** 등록 정보가 구성되어 있는 경우에는 특별히 주의해야 합니다. 또한 멀티홈 시스템의 인터페이스 간에 유니캐스트 경로 지정이 사용으로 설정되어 있는 경우에는 특수 DA 및 범위 구성이 필요하지 않습니다.

알림은 모든 서브넷에서 액세스 가능한 DA ID 서비스로 캐시됩니다. 그러나 유니캐스트 경로 지정이 사용 안함으로 설정되어 있는 경우 DA가 잘못 배치되어 있으면 문제가 발생할 수 있습니다.

이전 예에서 발생할 수 있는 문제를 확인하려면 **bigguy**에서 DA를 실행하고 모든 서브넷의 클라이언트에 동일한 범위가 있는 경우를 참고하십시오. 143 서브넷의 SA는 해당 서비스 알림을 DA에 등록합니다. 144 서브넷의 UA는 143 서브넷의 호스트에 도달할 수 없는 경우에도 해당 서비스 알림을 가져올 수 있습니다.

이 문제를 해결하는 한 가지 방법은 멀티홈 호스트가 아닌 각 서브넷에서 DA를 실행하는 것입니다. 이 경우 멀티홈 호스트의 **net.slp.interfaces** 등록 정보를 단일 인터페이스 호스트 이름 또는 주소로 구성하거나 사용할 기본 인터페이스가 실행되도록 구성하지 않은 채로 두어야 합니다. 이 해결 방법의 단점은 종종 멀티홈 호스트가 DA의 컴퓨터 로드를 더 잘 처리할 수 있는 대형 시스템이 된다는 것입니다.

다른 해결 방법은 멀티홈 호스트에서 DA를 실행하는 것이지만 각 서브넷의 SA 및 UA가 다른 범위를 가지도록 범위를 구성해야 합니다. 예를 들어, 앞의 상황에서 142 서브넷의 UA 및 SA에 **scope142**라고 하는 범위가 있을 수 있습니다. 143 서브넷의 UA 및 SA는 **scope143**이라고 하는 다른 범위가 있을 수 있으며 144 서브넷의 UA 및 SA는 **scope144**라고 하는 세번째 범위를 가질 수 있습니다. DA가 세 개의 서브넷에 세 개의 범위를 제공할 수 있도록 세 개의 인터페이스가 있는 **bigguy**에서 **net.slp.interfaces** 등록 정보를 구성할 수 있습니다.

## 경로가 지정되지 않은 다중 네트워크 인터페이스 구성 시 고려 사항

**net.slp.interfaces** 등록 정보를 구성하면 멀티홈 호스트의 DA가 서브넷 간에 서비스 알림을 연결할 수 있습니다. 이러한 구성은 네트워크에서 멀티캐스트 경로 지정이 해제된 경우에 유용하지만 멀티홈 호스트의 인터페이스 간에 유니캐스트 경로 지정이 사용으로 설정됩니다. 인터페이스 간에 유니캐스트의 경로가 지정되므로 서비스가 위치한 서브넷과 다른 서브넷은 서비스 URL을 받을 때 서비스에 연결할 수 있습니다. DA가 없으면 특정 서브넷의 SA 서버는 서브넷에 대해 해제된 서비스를 찾을 수 없도록 같은 서브넷에 생성된 브로드캐스트만 받습니다.

**net.slp.interfaces** 등록 정보를 구성해야 하는 일반적인 대부분의 경우는 네트워크에 멀티캐스트가 배포되지 않고 대신 브로드캐스트가 사용되는 경우입니다. 다른 상황에서는 주의 깊게 검토해야 하며 불필요한 중복 응답 또는 도달 가능하지 않은 서비스 방지를 계획해야 합니다.



## 레거시 서비스 통합

---

레거시 서비스는 SLP 개발 및 구현보다 날짜가 앞서는 네트워크 서비스입니다. 예를 들어 NFS 서비스 및 NIS 이름 서비스와 같은 서비스에는 SLP용 내부 SA가 포함되어 있지 않습니다. 이 장에서는 레거시 서비스를 알릴 시기와 방법에 대해 설명합니다.

- 255 페이지 “레거시 서비스를 알릴 시기”
- 255 페이지 “레거시 서비스 알림”
- 259 페이지 “레거시 서비스 알림 시 고려 사항”

### 레거시 서비스를 알릴 시기

레거시 서비스 알림을 사용하면 SLP UA가 네트워크에서 다음과 같은 장치와 서비스를 찾을 수 있도록 할 수 있습니다. SLP SA를 포함하지 않는 하드웨어 장치와 소프트웨어 서비스를 찾을 수 있습니다. 예를 들어 SLP UA가 포함된 응용 프로그램에서 SLP SA가 포함되지 않은 프린터나 데이터베이스를 찾아야 하는 경우 레거시 알림이 필요할 수 있습니다.

### 레거시 서비스 알림

다음 방법을 사용하여 레거시 서비스를 알릴 수 있습니다.

- SLP SA를 통합하도록 서비스를 수정합니다.
- SLP가 사용으로 설정되지 않은 서비스를 대신하여 알리는 작은 프로그램을 작성합니다.
- 프록시 알림을 사용하여 slpd가 서비스를 알리도록 합니다.

### 서비스 수정

소프트웨어 서버의 소스 코드를 사용할 수 있는 경우 SLP SA를 통합할 수 있습니다. SLP용 C 및 Java API는 비교적 사용하기 간단합니다. C API에 대한 정보를 제공하는

매뉴얼 페이지와 Java API에 대한 설명서를 참조하십시오. 서비스가 하드웨어 장치인 경우 제조업체에서 SLP를 통합하는 업데이트된 PROM을 제공할 수 있습니다. 자세한 내용은 장치 제조업체에 문의하십시오.

## SLP가 사용으로 설정되지 않은 서비스 알림

소스 코드나 SLP가 포함된 업데이트된 PROM을 사용할 수 없는 경우 SLP 클라이언트 라이브러리를 사용하여 서비스를 알리는 작은 응용 프로그램을 작성할 수 있습니다. 이 응용 프로그램은 서비스를 시작하거나 중지하는 데 사용하는 것과 동일한 셸 스크립트를 시작하거나 중지하는 작은 데몬으로 사용될 수 있습니다.

## SLP 프록시 등록

Oracle Solaris `slpd`에서는 프록시 등록 파일을 사용한 레거시 서비스 알림을 지원합니다. 프록시 등록 파일은 이동식 형식의 서비스 알림 목록입니다.

### ▼ SLP 프록시 등록을 사용으로 설정하는 방법

- 1 호스트 파일 시스템에서 또는 HTTP로 액세스할 수 있는 네트워크 디렉토리에서 프록시 등록 파일을 만듭니다.

- 2 서비스에 대한 서비스 유형 템플릿이 있는지 확인합니다.

템플릿은 서비스 URL에 대한 설명 및 서비스 유형의 속성입니다. 템플릿은 특정 서비스 유형에 대한 알림의 구성 요소를 정의하는 데 사용됩니다.

- 서비스 유형 템플릿이 있는 경우 템플릿을 사용하여 프록시 등록을 구성합니다. 서비스 유형 템플릿에 대한 자세한 내용은 RFC 2609를 참조하십시오.
- 서비스에 대한 서비스 유형 템플릿을 사용할 수 없으면 서비스를 정확히 설명하는 속성의 모음을 선택합니다. 알림에 대한 기본값과 다른 이름 지정 권한을 사용합니다. 기본 이름 지정 권한은 표준화된 서비스 유형에 대해서만 허용됩니다. 이름 지정 권한에 대한 자세한 내용은 RFC 2609를 참조하십시오.

예를 들어 *BizApp*이라는 회사에 소프트웨어 결함을 추적하는 데 사용되는 로컬 데이터베이스가 있다고 가정합니다. 이 데이터베이스를 알리기 위해 회사는 서비스 유형이 `service:bugdb.bizapp`인 URL을 사용할 수 있습니다. 그러면 이름 지정 권한은 `bizapp`가 됩니다.

- 3 다음 단계에 따라 `/etc/inet/slp.conf` 파일에서 `net.slp.serializedRegURL` 등록 정보를 이전 단계에서 만든 등록 파일의 위치로 구성합니다.



**4 관리자가 됩니다.**

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

**5 호스트에서 slpd 및 모든 SLP 작업을 중지합니다.**

```
# svcadm disable network/slp
```

**6 구성 설정을 변경하기 전에 기본 /etc/inet/slp.conf 파일을 백업합니다.****7 /etc/inet/slp.conf 파일의 net.slp.serializedRegURL 등록 정보에 프록시 등록 파일의 위치를 지정합니다.**

```
net.slp.net.slp.serializedRegURL=proxy registration file URL
```

예를 들어 일련화된 등록 파일이 /net/inet/slp.reg이면 등록 정보를 다음과 같이 구성합니다.

```
net.slp.serializedRegURL=file:/etc/inet/slp.reg
```

**8 변경 사항을 저장하고 파일을 닫습니다.****9 slpd를 다시 시작하여 변경 내용을 활성화합니다.**

```
# svcadm enable network/slp
```

## SLP 프록시 등록을 사용하여 알림

서비스 알림은 서비스 URL, 선택적 범위 및 일련의 속성 정의를 식별하는 행으로 구성됩니다. SLP 데몬은 SA 클라이언트와 똑같이 프록시 알림을 읽고, 등록하고, 유지 관리합니다. 다음은 프록시 등록 파일을 통한 알림의 예입니다.

이 예에서는 LPR 프로토콜 및 FTP 서버를 지원하는 레거시 프린터를 알립니다. 행 번호는 설명을 위해 추가되었으며 파일의 일부가 아닙니다.

```
(1)#Advertise legacy printer.
(2)
(3)service:lpr://bizserver/mainspool,en,65535
(4)scope=eng,corp
(5)make-model=Laserwriter II
(6)location-description=B16-2345
(7)color-supported=monochromatic
(8)fonts-supported=Courier,Times,Helvetica 9 10
(9)
(10)#Advertise FTP server
(11)
(12)ftp://archive/usr/src/public,en,65535,src-server
(13)content=Source code for projects
(14)
```

주 - 프록시 등록 파일에서는 ASCII가 아닌 문자를 이스케이프하기 위해 구성 파일과 동일한 규칙을 지원합니다. 프록시 등록 파일의 형식에 대한 자세한 내용은 RFC 2614를 참조하십시오.

표 10-1 SLP 프록시 등록 파일 설명

행 번호	설명
1 및 10	주석 행은 그물 무늬 기호(#)로 시작하며 파일의 작업에 영향을 주지 않습니다. 주석 행 끝까지의 모든 문자는 무시됩니다.
2, 9 및 14	알림을 구분하는 빈 행입니다.
3, 12	<p>서비스 URL로, 각각 3개의 필드와 1개의 선택적 필드가 쉼표로 구분되어 있습니다.</p> <ul style="list-style-type: none"> <li>■ 알려지는 일반 또는 <b>service:</b> URL입니다. <b>service:</b> URL을 만드는 방법에 대한 사양은 RFC 2609를 참조하십시오.</li> <li>■ 알림의 언어입니다. 이전 예에서는 이 필드에 영어, <i>en</i>을 지정했습니다. 언어는 RFC 1766 언어 태그입니다.</li> <li>■ 등록의 수명(초)입니다. 수명은 부호 없는 16비트 정수로 제한됩니다. 수명이 최대값 65535보다 적으면 <b>slpd</b>에서 알림 시간을 초과합니다. 수명이 65535이면 <b>slpd</b>에서는 알림을 정기적으로 갱신하며 <b>slpd</b>가 종료될 때까지는 수명이 영구적인 것으로 간주됩니다.</li> <li>■ (옵션) 서비스 유형 필드 - 사용하는 경우 이 필드에서는 서비스 유형을 정의합니다. 서비스 URL이 정의된 경우 URL이 알려지는 서비스 유형을 변경할 수 있습니다. 이전 예의 프록시 등록 파일에서는 12행에 일반 FTP URL이 포함됩니다. 선택적 유형 필드를 사용하면 URL이 서비스 유형 이름 <i>src-server</i>로 알려집니다. <b>service</b> 접두어는 유형 이름에 기본적으로 추가되지 않습니다.</li> </ul>
4	<p>범위 지정입니다.</p> <p>선택적 행에는 토큰 <b>scope</b>, 등호 및 쉼표로 구분된 범위 이름이 순서대로 포함됩니다. 범위 이름은 <b>net.slp.useScopes</b> 구성 등록 정보로 정의됩니다. 호스트에 대해 구성된 범위만 목록에 포함해야 합니다. 범위 행을 추가하지 않으면 <b>slpd</b>가 구성된 모든 범위에서 등록이 만들어집니다. 범위 행은 URL 행 바로 다음에 표시되어야 합니다. 그렇지 않으면 범위 이름이 속성으로 인식됩니다.</p>
5-8	<p>속성 정의입니다.</p> <p>선택적 범위 행 뒤의 대량의 서비스 알림에는 속성/값 목록 쌍 행이 포함됩니다. 각 쌍은 속성 태그, 등호 및 속성 값 또는 쉼표로 구분된 값 목록으로 구성됩니다. 이전 예의 프록시 등록 파일에서 8행에는 여러 값이 있는 속성 목록이 나와 있습니다. 다른 모든 목록에는 단일 값이 있습니다. 속성 이름 및 값의 형식은 실시간 SLP 메시지와 동일합니다.</p>

## 레거시 서비스 알림 시 고려 사항

일반적으로 다른 서비스 대신 SLP API를 사용하여 알리는 SLP 사용 가능 서비스를 작성하는 것보다 소스 코드를 수정하여 SLP를 추가하는 것이 좋습니다. 또한 소스 코드를 수정하는 것이 프록시 등록을 사용하는 것보다 좋습니다. 소스 코드를 수정할 때 서비스 관련 기능을 추가하고 서비스 가용성을 세심하게 추적할 수 있습니다. 소스 코드를 사용할 수 없는 경우에는 다른 서비스 대신 알리는 SLP 사용 가능 도우미 서비스를 작성하는 것이 프록시 등록을 사용하는 것보다 좋습니다. 이상적으로는 이 도우미 서비스가 활성화 및 비활성화를 제어하는 데 사용되는 서비스 시작/중지 프로시저로 통합되는 것이 좋습니다. 일반적으로 프록시 알림은 소스 코드를 사용할 수 없고 독립형 SA가 실용적이지 않을 때 사용하는 세번째 선택 사항입니다.

프록시 알림은 `slpd`가 실행되어 프록시 등록 파일을 읽고 있는 경우에만 유지됩니다. 프록시 알림 및 서비스 간의 직접적인 연결은 없습니다. 알림 시간이 초과되거나 `slpd`가 중지되는 경우 프록시 알림을 더 이상 사용할 수 없습니다.

서비스가 종료되면 `slpd`를 중지해야 합니다. 일련화된 등록 파일이 편집되어 프록시 알림이 주석 처리되거나 제거되고 `slpd`가 다시 시작됩니다. 서비스를 다시 시작하거나 다시 설치할 때는 동일한 절차를 따라야 합니다. 프록시 알림과 서비스 간의 연결이 없는 것이 프록시 알림의 주된 장점입니다.



## SLP(참조)

이 장에서는 SLP 상태 코드와 메시지 유형에 대해 설명합니다. SLP 메시지 유형이 약어 및 기능 코드와 함께 나열되어 있습니다. SLP 상태 코드는 요청을 받았음(코드 0) 또는 받는 사람이 작업 중임을 나타내는 기능 코드 및 설명과 함께 표시됩니다.

주 - SLP 데몬(slpd)은 유니캐스트 메시지에 대한 상태 코드만 반환합니다.

### SLP 상태 코드

표 11-1 SLP 상태 코드

상태 유형	상태 코드	설명
오류 없음	0	요청이 오류 없이 처리되었습니다.
LANGUAGE_NOT_SUPPORTED	1	AttrRqst 또는 SrvRqst의 경우 범위의 서비스 유형에 대한 데이터가 있지만 표시된 언어에는 데이터가 없습니다.
PARSE_ERROR	2	메시지가 SLP 구문을 따르지 않습니다.
INVALID_REGISTRATION	3	SrvReg에 문제가 있습니다. 예를 들어 수명이 0이거나 언어 태그가 생략되었습니다.
SCOPE_NOT_SUPPORTED	4	SLP 메시지가 요청에 응답한 SA 또는 DA에서 지원하는 범위 목록의 범위를 포함하지 않았습니다.
AUTHENTICATION_UNKNOWN	5	DA 또는 SA가 지원되지 않는 SLP SPI에 대한 요청을 받았습니다.
AUTHENTICATION_ABSENT	6	UA 또는 DA는 SrvReg에서 URL 및 속성 인증이 필요한데 받지 못했습니다.

표 11-1 SLP 상태 코드 (계속)

상태 유형	상태 코드	설명
AUTHENTICATION_FAILED	7	UA 또는 DA가 인증 블록에서 인증 오류를 발견했습니다.
VER_NOT_SUPPORTED	9	메시지의 버전 번호가 지원되지 않습니다.
INTERNAL_ERROR	10	DA 또는 SA에서 알 수 없는 오류가 발생했습니다. 예를 들어 운영 체제에 남은 파일 공간이 없습니다.
DA_BUSY_NOW	11	UA 또는 SA가 지수 백오프를 사용하여 다시 시도해야 합니다. DA가 다른 메시지를 처리하고 있습니다.
OPTION_NOT_UNDERSTOOD	12	DA 또는 SA가 필수 범위에서 알 수 없는 옵션을 받았습니다.
INVALID_UPDATE	13	DA가 등록되지 않은 서비스에 대해 FRESH 설정 없이 또는 일치하지 않는 서비스 유형으로 SrvReg를 받았습니다.
MSG_NOT_SUPPORTED	14	SA가 지원하지 않는 AttrRqst 또는 SrvTypeRqst를 받았습니다.
REFRESH_REJECTED	15	SA가 SrvReg 또는 부분 SrvDereg를 DA에 DA의 최소 갱신 간격보다 자주 보냈습니다.

# SLP 메시지 유형

표 11-2 SLP 메시지 유형

메시지 유형	약어	기능 코드	설명
서비스 요청	SrvRqst	1	UA가 서비스를 찾기 위해 실행하거나 UA 또는 SA 서버가 활성 DA를 검색하는 동안 실행합니다.
서비스 회신	SrvRply	2	서비스 요청에 대한 DA 또는 SA 응답입니다.
서비스 등록	SrvReg	3	SA가 새 알림을 등록하거나, 기존 알림을 새 속성이나 변경된 속성으로 업데이트하거나, URL 수명을 새로 고칠 수 있도록 합니다.
서비스 등록 취소	SrvDereg	4	SA에서 나타내는 서비스를 더 이상 사용할 수 없는 경우 해당 알림의 등록을 취소하기 위해 사용됩니다.
승인	SrvAck	5	SA의 서비스 요청 또는 서비스 등록 취소 메시지에 대한 DA 응답입니다.

표 11-2 SLP 메시지 유형 (계속)

메시지 유형	약어	기능 코드	설명
속성 요청	AttrRqst	6	URL 또는 서비스 유형으로 만들어지며 속성 목록을 요청합니다.
속성 회신	AttrRply	7	속성 목록을 반환하는 데 사용됩니다.
DA 알림	DAAdvert	8	멀티캐스트 서비스 요청에 대한 DA 응답입니다.
서비스 유형 요청	SrvTypeRqst	9	특정 이름 지정 권한을 가지며 특정 범위 세트에 속하는 등록된 서비스 유형을 조회하는 데 사용됩니다.
서비스 유형 회신	SrvTypeRply	10	서비스 유형 요청에 대한 응답으로 반환되는 메시지입니다.
SA 알림	SAAdvert	11	UA는 SAAdvert를 사용하여 DA가 배포되지 않은 네트워크에서 SA 및 해당 범위를 검색합니다.





## 제 4 부

# 메일 서비스 항목

이 절에서는 메일 서비스에 대한 개요, 작업 및 참조 정보를 제공합니다.



## 메일 서비스(개요)

---

전자 메일 서비스 설정 및 유지 관리에는 일상적인 네트워크 업무에 필수적인 복잡한 작업이 포함됩니다. 네트워크 관리자는 기존의 메일 서비스를 확장해야 하거나 이를 대신하여 새로운 네트워크나 서브넷에서 메일 서비스를 설정해야 할 수도 있습니다. 메일 서비스와 관련한 장은 네트워크에 맞게 메일 서비스를 계획하고 설정하는 데 유용합니다. 이 장에는 `sendmail`의 새로운 기능 및 기타 정보 소스 목록에 대한 설명 링크가 있습니다. 또한 메일 서비스를 설정하는 데 필요한 소프트웨어 및 하드웨어 구성 요소의 개요도 제공합니다.

- [267 페이지 “메일 서비스의 새로운 기능”](#)
- [269 페이지 “기타 `sendmail` 정보 소스”](#)
- [269 페이지 “메일 서비스 구성 요소 소개”](#)

메일 서비스를 설정 및 관리하는 방법에 대한 절차 정보는 [13 장, “메일 서비스\(작업\)”](#)를 참조하십시오. 자세한 내용은 [273 페이지 “메일 서비스용 작업 맵”](#)을 참조하십시오.

메일 서비스의 구성 요소에 대한 자세한 내용은 [14 장, “메일 서비스\(참조\)”](#)를 참조하십시오. 이 장에서는 메일 서비스 프로그램 및 파일, 메일 경로 지정 프로세스, 이름 서비스와 `sendmail`의 상호 작용 및 `sendmail` 버전 8.13의 새로운 기능에 대해 설명합니다. [349 페이지 “`sendmail` 버전 8.13의 변경 사항”](#)을 참조하십시오.

## 메일 서비스의 새로운 기능

이 절에서는 여러 Oracle Solaris 릴리스의 새로운 기능에 대한 정보를 제공합니다.

## 이 릴리스의 변경 사항

Oracle Solaris 11 릴리스에서 변경된 사항은 다음과 같습니다.

- sendmail의 기본 버전이 8.14.5로 업데이트되었습니다.
- 일반적인 데몬(svc:/network/smtp:sendmail) 및 클라이언트 대기열 실행기(svc:/network/smtp:sendmail-client)를 보다 효과적으로 관리할 수 있도록 sendmail 인스턴스를 인스턴스 두 개로 분할했습니다.
- sendmail.cf 및 submit.mc 구성 파일을 자동으로 재구성하도록 시스템을 구성할 수 있습니다. 필요한 단계는 [287 페이지 “구성 파일을 자동으로 다시 작성하는 방법”](#)에 설명되어 있습니다.
- 기본적으로 sendmail 데몬은 새 로컬 데몬 모드에서 실행됩니다. 로컬 전용 모드에서는 로컬 호스트나 루프백 SMTP 연결에서 받는 메일만 수락합니다. 예를 들어, cron 작업에서 보낸 메일이나 로컬 사용자 간의 메일이 수락됩니다. 아웃바운드 메일의 경로가 예상대로 지정되고 받는 메일만 변경됩니다. Become Local(로컬 전환) 모드로도 알려진 로컬 전용 모드를 선택하기 위해 -bl 옵션이 사용됩니다. 이 모드에 대한 자세한 내용은 [sendmail\(1M\)](#) 매뉴얼 페이지를 참조하십시오. -bd 또는 Become Daemon(데몬 전환) 모드로 다시 변경하는 방법에 대한 자세한 내용은 [288 페이지 “열기 모드에서 sendmail 사용 방법”](#)을 참조하십시오.
- makemap에 대한 -t 및 -u 옵션이 이제 예상한 대로 작동합니다. -u 옵션을 사용해도 -t 옵션으로 선언된 분리자가 분리자로 사용됩니다. 전에는 -u 옵션이 사용된 경우 -t 옵션으로 정의된 분리자가 있어도 공백이 분리자로 사용되었습니다. 이 옵션에 대한 자세한 내용은 [makemap\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## 이전 릴리스의 중요한 변경 사항

- sendmail은 TLS(전송 계층 보안)를 사용하는 SMTP를 지원합니다. 자세한 내용은 다음을 참조하십시오.
  - [349 페이지 “sendmail 버전 8.13에서 TLS를 사용하는 SMTP 실행 지원”](#)
  - [289 페이지 “TLS를 사용하도록 SMTP를 설정하는 방법”](#)
- sendmail 버전 8.13이 추가되었습니다. 버전 8.13 및 기타 변경 사항에 대한 자세한 내용은 다음을 참조하십시오.
  - [316 페이지 “sendmail 컴파일에 사용되는 플래그 및 사용되지 않는 플래그”](#)
  - [317 페이지 “MILTER, sendmail용 메일 필터 API”](#)
  - [318 페이지 “구성 파일 버전”](#)
  - [329 페이지 “vacation 유틸리티의 향상된 기능”](#)
  - [331 페이지 “/etc/mail/cf 디렉토리의 내용”](#)
  - [349 페이지 “sendmail 버전 8.13의 변경 사항”](#)
  - [357 페이지 “sendmail 버전 8.12의 TCP 래퍼에 대한 지원”](#)

- 서비스 관리 기능을 사용하여 메일 서비스를 관리합니다. `svcadm` 명령을 사용하여 사용으로 설정, 사용 안함으로 설정, 다시 시작 등 이 서비스에 대한 관리 작업을 수행할 수 있습니다. `svcs` 명령을 사용하여 서비스의 상태를 질의할 수 있습니다. 서비스 관리 기능에 대한 자세한 내용은 [smf\(5\)](#) 매뉴얼 페이지 및 [Oracle Solaris 관리: 일반 작업의 6 장, “서비스 관리\(개요\)”](#)를 참조하십시오.

## 기타 sendmail 정보 소스

다음은 sendmail에 대한 추가 정보 소스 목록입니다.

- Costales, Bryan. **sendmail, Third Edition**. O'Reilly & Associates, Inc., 2002.
- sendmail 홈페이지 – <http://www.sendmail.org>
- sendmail FAQ – <http://www.sendmail.org/faq>
- 새 sendmail 구성 파일의 README – <http://www.sendmail.org/m4/readme.html>
- sendmail 최신 버전으로의 마이그레이션 관련 문제에 대한 설명서 – <http://www.sendmail.org/vendor/sun/>

## 메일 서비스 구성 요소 소개

메일 서비스를 설정하려면 여러 소프트웨어 및 하드웨어 구성 요소가 필요합니다. 다음 절에서는 이러한 구성 요소를 간단하게 소개합니다. 또한 구성 요소를 설명하는 데 사용되는 몇 가지 용어를 제공합니다.

첫번째 절인 [269 페이지 “소프트웨어 구성 요소 개요”](#)에서는 메일 배달 시스템의 소프트웨어 부분을 설명하는 데 사용되는 용어를 정의합니다. 다음 절인 [270 페이지 “하드웨어 구성 요소 개요”](#)에서는 메일 구성에서 하드웨어 시스템의 기능을 중점적으로 설명합니다.

## 소프트웨어 구성 요소 개요

다음 표에서는 메일 시스템의 일부 소프트웨어 구성 요소를 소개합니다. 모든 소프트웨어 구성 요소에 대한 자세한 내용은 [319 페이지 “소프트웨어 구성 요소”](#)를 참조하십시오.

구성 요소	설명
.forward 파일	자동으로 메일을 재지정하거나 프로그램에 메일을 보내기 위해 사용자의 홈 디렉토리에서 설정할 수 있는 파일
우편함	메일 서버에 있는 파일이며 전자 메일 메시지의 최종 목적지

구성 요소	설명
메일 주소	메일 메시지가 배달될 수신자 및 시스템 이름이 포함된 주소
메일 별칭	메일 주소에 사용되는 대체 이름
메일 대기열	메일 서버에서 처리할 메일 메시지 모음
포스트마스터	문제를 보고하고 메일 서비스에 대해 질문하는 데 사용되는 특수한 메일 별칭
sendmail 구성 파일	메일 경로 지정에 필요한 모든 정보가 포함된 파일

## 하드웨어 구성 요소 개요

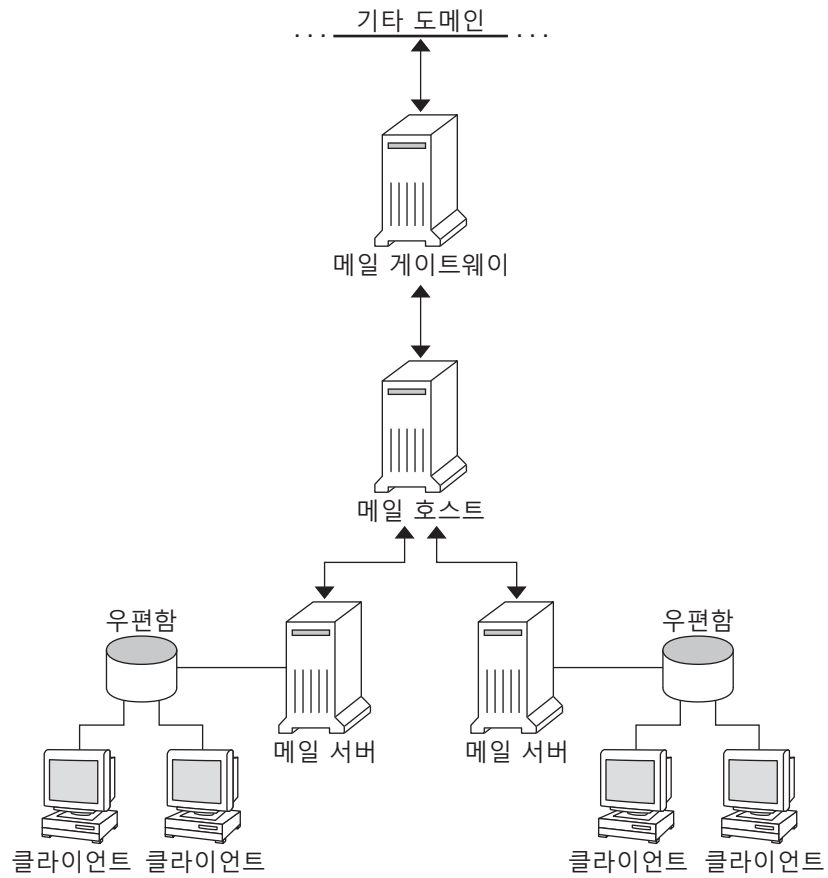
메일 구성에는 세 가지 요소가 필요하며 이 요소를 같은 시스템에서 결합하거나 별도의 시스템에 제공할 수 있습니다.

- 메일 호스트 - 확인하기 어려운 전자 메일 주소를 처리하기 위해 구성된 시스템
- 최소 1개의 메일 서버 - 하나 이상의 우편함을 보유할 수 있도록 구성된 시스템
- 메일 클라이언트 - 메일 서버의 메일에 액세스하는 시스템

사용자가 도메인 외부의 네트워크와 통신해야 할 경우 네번째 요소인 메일 게이트웨이도 추가해야 합니다.

[그림 12-1](#)에서는 세 가지 기본 메일 요소와 메일 게이트웨이를 사용하는 일반적인 전자 메일 구성을 보여줍니다.

그림 12-1 일반적인 전자 메일 구성



326 페이지 “하드웨어 구성 요소”에서 각 요소에 대해 자세히 설명합니다.





## 메일 서비스(작업)

이 장에서는 메일 서비스를 설정하고 관리하는 방법에 대해 설명합니다. 메일 서비스 관리에 익숙하지 않은 경우 12 장, “메일 서비스(개요)”에서 메일 서비스 구성 요소에 대한 소개 내용을 참조하십시오. 이 장에서는 그림 12-1에서와 같은 일반적인 메일 서비스 구성에 대해서도 설명합니다. 다음 목록에서는 이 장에서 다루는 관련 절차 그룹을 찾을 수 있습니다.

- 273 페이지 “메일 서비스용 작업 맵”
- 277 페이지 “메일 서비스 설정(작업 맵)”
- 285 페이지 “sendmail 구성 변경(작업 맵)”
- 294 페이지 “편지 별칭 파일 관리(작업 맵)”
- 300 페이지 “대기열 디렉토리 관리(작업 맵)”
- 304 페이지 “.forward 파일 관리(작업 맵)”
- 306 페이지 “메일 서비스의 문제 해결 절차 및 팁(작업 맵)”

메일 서비스의 구성 요소에 대한 자세한 내용은 14 장, “메일 서비스(참조)”를 참조하십시오. 이 장에서는 메일 서비스 프로그램 및 파일, 메일 경로 지정 프로세스, 이름 서비스와 sendmail의 상호 작용, 그리고 sendmail(1M) 매뉴얼 페이지에서 자세히 설명하지 않은 sendmail 버전 8.13의 기능에 대해 설명합니다.

### 메일 서비스용 작업 맵

다음 표에서는 특정 절차 그룹을 중점적으로 다루는 다른 작업 맵을 참조할 수 있습니다.

작업	설명	수행 방법
메일 서비스 설정	이 절차를 사용하여 메일 서비스의 각 구성 요소를 설정합니다. 메일 서버, 메일 클라이언트, 메일 호스트 및 메일 게이트웨이 설정 방법을 배웁니다. sendmail과 함께 DNS를 사용하는 방법을 배웁니다.	277 페이지 “메일 서비스 설정(작업 맵)”

작업	설명	수행 방법
sendmail 구성 변경	이 절차를 사용하여 구성 파일 또는 서비스 등록 정보를 수정합니다.	285 페이지 “sendmail 구성 변경(작업 맵)”
편지 별칭 파일	이 절차를 사용하여 네트워크에 별칭을 제공합니다. NIS 맵, 로컬 편지 별칭, 키 맵 파일 및 포스트마스터 별칭 설정 방법을 배웁니다.	294 페이지 “편지 별칭 파일 관리(작업 맵)”
메일 대기열 관리	이 절차를 사용하여 원활한 대기열 처리를 제공합니다. 메일 대기열을 표시 및 이동하고 메일 대기열 처리를 강제 수행하며 메일 대기열의 일부를 실행하는 방법을 배웁니다. 또한 이전의 메일 대기열을 실행하는 방법을 배웁니다.	300 페이지 “대기열 디렉토리 관리(작업 맵)”
.forward 파일 관리	이 절차를 사용하여 .forward 파일을 사용 안함으로 설정하거나 .forward 파일의 검색 경로를 변경합니다. 또한 /etc/shells를 만들고 채워 사용자가 .forward 파일을 사용하도록 허용하는 방법을 배웁니다.	304 페이지 “.forward 파일 관리(작업 맵)”
메일 서비스의 문제 해결 절차 및 팁	이 절차와 팁을 사용하여 메일 서비스의 문제를 해결합니다. 메일 구성을 테스트하고, 편지 별칭을 확인하고, sendmail 규칙 세트를 테스트하고, 다른 시스템과의 연결을 확인하고, 메시지를 기록하는 방법을 배웁니다. 또한 다른 메일 진단 정보를 찾을 위치를 알아봅니다.	306 페이지 “메일 서비스의 문제 해결 절차 및 팁(작업 맵)”
오류 메시지 해결	이 절의 정보를 사용하여 일부 메일 관련 오류 메시지를 해결합니다.	311 페이지 “오류 메시지 해결”

## 메일 시스템 계획

다음 목록에서는 계획 프로세스에 포함해야 하는 몇 가지 내용에 대해 설명합니다.

- 요구 사항에 맞는 메일 구성 유형을 확인합니다. 이 절에서는 메일 구성의 기본 유형 두 가지에 대해 설명하고, 각 구성을 설정하는 데 무엇이 필요한지 간단히 보여줍니다. 새 메일 시스템을 설정해야 하거나 기존의 메일 시스템을 확장하는 경우 이 절이 도움이 됩니다. 275 페이지 “로컬 메일만”에서는 첫번째 구성 유형을 설명하고, 276 페이지 “로컬 메일 및 원격 구성”에서는 두번째 유형에 대해 설명합니다.
- 필요에 따라 메일 서버, 메일 호스트 및 메일 게이트웨이 역할을 할 시스템을 선택합니다.

- 서비스를 제공하고 우편함 위치를 포함하는 모든 메일 클라이언트의 목록을 만듭니다. 사용자의 편지 별칭을 만들 수 있을 때 이 목록이 유용합니다.
- 별칭을 업데이트하고 메일 메시지를 전달하는 방법을 결정합니다. 사용자가 메일 전달을 위한 요청을 보낼 위치로 **aliases** 우편함을 설정할 수 있습니다. 사용자가 이 우편함을 사용하여 기본 편지 별칭의 변경을 위한 요청을 보낼 수도 있습니다. 시스템에서 NIS를 사용할 경우 사용자에게 메일 전달을 관리하도록 요청하는 대신 메일 전달을 관리할 수 있습니다. [294 페이지 “편지 별칭 파일 관리\(작업 맵\)”](#)에는 별칭과 관련된 작업 목록이 있습니다. [304 페이지 “.forward 파일 관리\(작업 맵\)”](#)에는 .forward 파일 관리와 관련된 작업 목록이 있습니다.

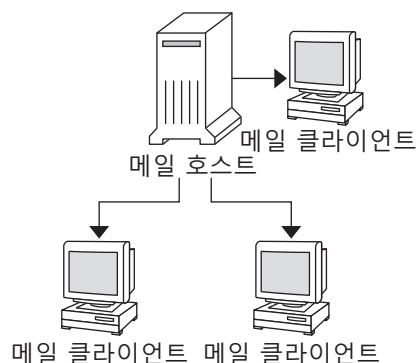
계획 프로세스를 완료한 후 사이트에서 시스템을 설정하여 [277 페이지 “메일 서비스 설정\(작업 맵\)”](#)에 설명된 기능을 수행합니다. 기타 작업 정보는 [273 페이지 “메일 서비스용 작업 맵”](#)을 참조하십시오.

## 로컬 메일만

[그림 13-1](#)에 나온 대로 가장 간단한 메일 구성은 메일 호스트 하나에 두 대 이상의 워크스테이션이 연결된 경우입니다. 메일은 완전히 로컬입니다. 모든 클라이언트는 로컬 디스크에 메일을 저장하고 클라이언트가 메일 서버 역할을 합니다.

/etc/mail/aliases 파일을 사용하면 메일 주소가 구문 분석됩니다.

그림 13-1 로컬 메일 구성



이 종류의 메일 구성을 설정하려면 다음이 필요합니다.

- 각 메일 클라이언트 시스템에 편집할 필요가 없는 기본 `/etc/mail/sendmail.cf` 파일이 있어야 함

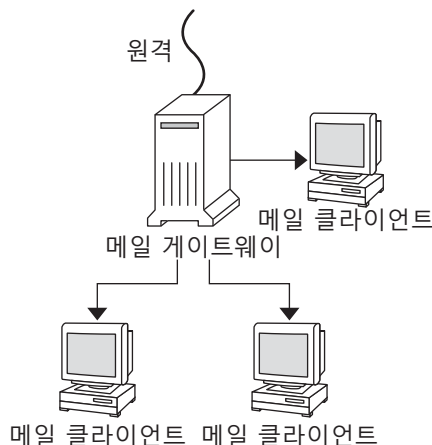
- 메일 호스트로 지정된 서버.NIS를 실행하는 경우 `mailhost.domain-name`을 메일 호스트의 `/etc/hosts` 파일에 추가하여 이와 같이 지정할 수 있습니다.DNS 또는 LDAP와 같은 다른 이름 서비스를 실행하는 경우 `/etc/hosts` 파일에 추가 정보를 제공해야 합니다.281 페이지 “메일 호스트 설정 방법”을 참조하십시오.
- NIS 이외의 이름 서비스를 사용하는 경우 로컬 우편함이 있는 시스템에 일치하는 `/etc/mail/aliases` 파일이 있어야 함
- 각 메일 클라이언트 시스템의 `/var/mail`에 우편함을 위한 충분한 공간이 필요함

메일 서비스 설정에 대한 작업 정보는 277 페이지 “메일 서비스 설정”을 참조하십시오.  
메일 서비스 설정에 대한 특정 절차를 찾는 경우 277 페이지 “메일 서비스 설정(작업 맵)”을 참조하십시오.

## 로컬 메일 및 원격 구성

소규모 네트워크에서 가장 공통된 메일 구성이 그림 13-2에 나와 있습니다.시스템 하나에 메일 서버, 메일 호스트 및 원격 연결을 제공하는 메일 게이트웨이가 포함됩니다. 메일 게이트웨이에서 `/etc/mail/aliases` 파일을 사용하여 메일이 배포됩니다.이름 서비스는 필요 없습니다.

그림 13-2 UUCP 연결을 사용하는 로컬 메일 구성



이 구성에서 메일 클라이언트가 `/var/mail`의 메일 파일을 메일 호스트에 마운트한다고 가정할 수 있습니다.이 종류의 메일 구성을 설정하려면 다음이 필요합니다.

- 각 메일 클라이언트 시스템에 있는 기본 `/etc/mail/sendmail.cf` 파일.이 파일은 편집할 필요가 없습니다.

- 메일 호스트로 지정된 서버.NIS를 실행하는 경우 `mailhost.domain-name`을 메일 호스트의 `/etc/hosts` 파일에 추가하여 이와 같이 지정할 수 있습니다. DNS 또는 LDAP와 같은 다른 이름 서비스를 실행하는 경우 `/etc/hosts` 파일에 추가 정보를 제공해야 합니다. [281 페이지 “메일 호스트 설정 방법”](#)을 참조하십시오.
- NIS 이외의 이름 서비스를 사용하는 경우 로컬 우편함이 있는 시스템에 일치하는 `/etc/mail/aliases` 파일이 있어야 함
- 메일 서버의 `/var/mail`에 클라이언트 우편함을 위한 충분한 공간이 필요함

메일 서비스 설정에 대한 작업 정보는 [277 페이지 “메일 서비스 설정”](#)을 참조하십시오. 메일 서비스 설정에 대한 특정 절차를 찾는 경우 [277 페이지 “메일 서비스 설정\(작업 맵\)”](#)을 참조하십시오.

## 메일 서비스 설정(작업 맵)

다음 표에서는 메일 서비스 설정을 위한 절차에 대해 설명합니다.

작업	설명	수행 방법
메일 서버 설정	서버가 메일의 경로를 지정하도록 설정하는 단계	<a href="#">278 페이지 “메일 서버 설정 방법”</a>
메일 클라이언트 설정	사용자가 메일을 수신하도록 설정하는 단계	<a href="#">279 페이지 “메일 클라이언트 설정 방법”</a>
메일 호스트 설정	전자 메일 주소를 확인할 수 있는 메일을 설정하는 단계	<a href="#">281 페이지 “메일 호스트 설정 방법”</a>
메일 게이트웨이 설정	도메인 외부 네트워크와의 통신을 관리하는 단계	<a href="#">283 페이지 “메일 게이트웨이 설정 방법”</a>
sendmail과 함께 DNS 사용	DNS 호스트 조회를 사용으로 설정하는 단계	<a href="#">284 페이지 “sendmail과 함께 DNS를 사용하는 방법”</a>

## 메일 서비스 설정

사이트에서 회사 외부의 전자 메일 서비스에 대한 연결을 제공하지 않을 경우 또는 회사가 단일 도메인에 속한 경우 메일 서비스를 설정할 수 있습니다.

메일에는 두 가지 유형의 로컬 메일용 구성이 필요합니다. 이 구성을 보려면 [275 페이지 “로컬 메일만”](#)의 [그림 13-1](#)을 참조하십시오. 메일에는 도메인 외부 네트워크와의 통신을 위한 추가 구성 두 가지가 필요합니다. 이 구성을 보려면 [270 페이지 “하드웨어 구성 요소 개요”](#)의 [그림 12-1](#) 또는 [276 페이지 “로컬 메일 및 원격 구성”](#)의 [그림 13-2](#)를 참조하십시오. 이 구성을 같은 시스템에 결합하거나 별도의 시스템에 제공할 수

있습니다. 예를 들어 메일 호스트 및 메일 서버 기능이 같은 시스템에 있을 경우 이 절의 지침에 따라 시스템을 메일 호스트로 설정합니다. 그런 다음 이 절의 지침에 따라 같은 시스템을 메일 서버로 설정합니다.

주-우편함에 NFS가 마운트된 경우 메일 서버와 메일 클라이언트를 설정하기 위한 다음 절차가 적용됩니다. 그러나 대개 우편함은 로컬로 마운트된 `/var/mail` 디렉토리에 보관되므로 다음 절차가 필요 없습니다.

## ▼ 메일 서버 설정 방법

로컬 사용자용 메일에만 사용되는 메일 서버를 설정하는 데에는 특별한 단계가 필요 없습니다. 사용자는 암호 파일이나 이름 공간에 항목이 있어야 합니다. 또한 메일이 배달되려면 `~/ .forward` 파일을 확인할 로컬 홈 디렉토리가 있어야 합니다. 따라서 홈 디렉토리 서버가 종종 메일 서버로 설정됩니다. 메일 서버에 대한 자세한 내용은 14 장, “메일 서비스(참조)”의 326 페이지 “하드웨어 구성 요소”를 참조하십시오.

메일 서버는 여러 메일 클라이언트의 메일의 경로를 지정할 수 있습니다. 이 유형의 메일 서버에는 클라이언트 우편함에 적합한 스펙링 공간이 있어야 합니다.

주-`mail.local` 프로그램은 메시지가 처음 배달될 때 `/var/mail` 디렉토리에 자동으로 우편함을 만듭니다. 메일 클라이언트마다 개별 우편함을 만들 필요는 없습니다.

클라이언트가 우편함에 액세스하려면 `/var/mail` 디렉토리를 원격 마운트에 사용할 수 있어야 합니다. 또는 POP(Post Office Protocol)나 IMAP(Internet Message Access Protocol)와 같은 서비스를 서버에서 사용할 수 있어야 합니다. 다음 작업은 `/var/mail` 디렉토리를 사용하여 메일 서버를 설정하는 방법을 보여줍니다. 이 문서에서는 POP나 IMAP를 위한 구성 지침을 제공하지 않습니다.

다음 작업의 경우 `/var/mail` 디렉토리를 내보냈다는 내용이 `/etc/dfs/dfstab` 파일에 표시되어야 합니다.

### 1 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

### 2 sendmail을 중지합니다.

```
# svcadm disable -t network/smtp:sendmail
```

### 3 원격 액세스에 `/var/mail` 디렉토리를 사용할 수 있는지 확인합니다.

```
# share
```

`/var/mail` 디렉토리가 나열되면 5단계로 이동합니다.

/var/mail 디렉토리가 나열되지 않거나 목록이 나타나지 않으면 알맞은 하위 단계를 계속 진행합니다.

- a. (옵션) 목록이 나타나지 않으면 NFS 서비스를 시작합니다.

82 페이지 “자동 파일 시스템 공유를 설정하는 방법” 절차에 따라 /var/mail 디렉토리를 사용하여 NFS 서비스를 시작합니다.

- b. (옵션) /var/mail 디렉토리가 목록에 없으면 디렉토리를 /etc/dfs/dfstab에 추가합니다.

다음 명령줄을 /etc/dfs/dfstab 파일에 추가합니다.

```
share -F nfs -o rw /var/mail
```

- 4 파일 시스템을 마운트에 사용할 수 있도록 합니다.

```
# shareall
```

- 5 이름 서비스가 시작되었는지 확인합니다.

- a. (옵션) NIS를 실행 중이면 다음 명령을 사용합니다.

```
# ypwhich
```

자세한 내용은 [ypwhich\(1\)](#) 매뉴얼 페이지를 참조하십시오.

- b. (옵션) DNS를 실행 중이면 다음 명령을 사용합니다.

```
# nslookup hostname
```

hostname 호스트 이름을 사용합니다.

자세한 내용은 [nslookup\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

- c. (옵션) LDAP를 실행 중이면 다음 명령을 사용합니다.

```
# ldaplist
```

자세한 내용은 [ldaplist\(1\)](#) 매뉴얼 페이지를 참조하십시오.

- 6 sendmail을 다시 시작합니다.

```
# svcadm enable network/smtp:sendmail
```

## ▼ 메일 클라이언트 설정 방법

메일 클라이언트는 메일 서버에 우편함이 있는 메일 서비스의 사용자입니다. 또한 메일 클라이언트는 우편함 위치를 가리키는 /etc/mail/aliases 파일에 별칭이 있습니다.

주 - POP(Post Office Protocol) 또는 IMAP(Internet Message Access Protocol)와 같은 서비스를 사용하여 메일 클라이언트 설정 작업을 수행할 수도 있습니다. 그러나 이 문서에서는 POP나 IMAP를 위한 구성 지침을 제공하지 않습니다.

**1 메일 클라이언트 시스템의 관리자가 됩니다.**

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

**2 sendmail을 중지합니다.**

```
# svcadm disable -t network/smtp:sendmail
```

**3 메일 클라이언트 시스템에 /var/mail 마운트 지점이 있는지 확인합니다.**

설치 프로세스 중에 마운트 지점을 만들어야 합니다. `ls`를 사용하여 파일 시스템이 있는지 확인할 수 있습니다. 다음 예에서는 파일 시스템이 만들어지지 않은 경우 받게 되는 응답을 보여줍니다.

```
# ls -l /var/mail
/var/mail not found
```

**4 /var/mail 디렉토리에 파일이 없는지 확인합니다.**

이 디렉토리에 메일 파일이 있으면 서버에서 /var/mail 디렉토리를 마운트할 때 덮어쓰지 않도록 파일을 이동해야 합니다.

**5 메일 서버에서 /var/mail 디렉토리를 마운트합니다.**

자동으로 또는 부트 시에 메일 디렉토리를 마운트할 수 있습니다.

**a. (옵션) /var/mail을 자동으로 마운트합니다.**

다음과 같은 항목을 /etc/auto\_direct 파일에 추가합니다.

```
/var/mail -rw,hard,actimeo=0 server:/var/mail
server      지정된 서버 이름을 사용합니다.
```

**b. (옵션) 부트 시에 /var/mail을 마운트합니다.**

다음 항목을 /etc/vfstab 파일에 추가합니다. 이 항목은 /var/mail 디렉토리를 마운트하도록 지정된 메일 서버에 /var/mail 디렉토리를 허용합니다.

```
server:/var/mail - /var/mail nfs - no rw,hard,actimeo=0
```

시스템이 재부트될 때마다 클라이언트의 우편함이 자동으로 마운트됩니다. 시스템을 재부트하지 않으면 다음 명령을 입력하여 클라이언트 우편함을 마운트합니다.

```
# mountall
```





**주의** - 우편함 잠금과 우편함 액세스가 제대로 작동하려면 NFS 서버에서 메일을 마운트할 때 `actimeo=0` 옵션을 포함해야 합니다.

## 6 /etc/hosts를 업데이트합니다.

/etc/hosts 파일을 편집하고 메일 서버에 대한 항목을 추가합니다. 이름 서비스를 사용하지 않는 경우 이 단계가 필요 없습니다.

```
# cat /etc/hosts
#
# Internet host table
#
..
IP-address      mailhost mailhost mailhost.example.com
IP-address      지정된 IP 주소를 사용합니다.
example.com     지정된 도메인을 사용합니다.
mailhost        지정된 메일 호스트를 사용합니다.

자세한 내용은 hosts\(4\) 매뉴얼 페이지를 참조하십시오.
```

## 7 별칭 파일 중 하나에 클라이언트에 대한 항목을 추가합니다.

편지 별칭 파일 관리에 대한 작업 맵은 [294 페이지](#) “편지 별칭 파일 관리(작업 맵)”를 참조하십시오. `mail.local` 프로그램은 메시지가 처음 배달될 때 /var/mail 디렉토리에 자동으로 우편함을 만듭니다. 메일 클라이언트마다 개별 우편함을 만들 필요는 없습니다.

## 8 sendmail을 다시 시작합니다.

```
# svcadm enable network/smtp:sendmail
```

# ▼ 메일 호스트 설정 방법

메일 호스트는 전자 메일 주소를 확인하고 도메인 내에서 메일의 경로를 지정합니다. 네트워크에 원격 연결을 제공하거나 네트워크를 부모 도메인에 연결하는 시스템이 메일 호스트로 적합합니다. 다음 절차에서는 메일 호스트 설정 방법을 보여줍니다.

## 1 메일 호스트 시스템의 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스](#)의 “관리 권한을 얻는 방법”을 참조하십시오.

## 2 sendmail을 중지합니다.

```
# svcadm disable -t network/smtp:sendmail
```

### 3 host-name 구성을 확인합니다.

check-hostname 스크립트를 실행하여 sendmail이 이 서버의 정규화된 호스트 이름을 식별할 수 있는지 확인합니다.

```
% /usr/sbin/check-hostname
hostname phoenix OK: fully qualified as phoenix.example.com
```

이 스크립트로 정규화된 호스트 이름을 식별하지 못하면 /etc/hosts에서 정규화된 호스트 이름을 호스트의 첫번째 별칭으로 추가해야 합니다.

### 4 /etc/hosts 파일을 업데이트합니다.

알맞은 단계를 선택합니다.

#### a. (옵션)NIS를 사용하는 경우 새 메일 호스트가 될 시스템에서 /etc/hosts 파일을 편집합니다.

메일 호스트 시스템의 IP 주소와 시스템 이름 뒤에 단어 mailhost 및 mailhost.  
domain을 추가합니다.

*IP-address mailhost mailhost mailhost.domain loghost*

*IP-address* 지정된 IP 주소를 사용합니다.

*mailhost* 메일 호스트 시스템의 시스템 이름을 사용합니다.

*domain* 확장된 도메인 이름을 사용합니다.

이제 시스템이 메일 호스트로 지정됩니다. *domain*은 다음 명령의 출력에 하위 도메인 이름으로 제공된 문자열과 같아야 합니다.

```
% /usr/lib/sendmail -bt -d0 </dev/null
Version 8.13.1+Sun
Compiled with: LDAPMAP MAP_REGEX LOG MATCHGECOS MIME7TO8 MIME8TO7
               NAMED_BIND NDBM NETINET NETINET6 NETUNIX NEWDB NIS
               NISPLUS QUEUE SCANF SMTP USERDB XDEBUG
```

```
===== SYSTEM IDENTITY (after readcf) =====
  (short domain name) $w = phoenix
  (canonical domain name) $j = phoenix.example.com
    (subdomain name) $m = example.com
      (node name) $k = phoenix
=====
```

이 변경 후에 hosts 파일이 표시되는 방식은 다음 예를 참조하십시오.

```
# cat /etc/hosts
#
# Internet host table
#
172.31.255.255 localhost
192.168.255.255 phoenix mailhost mailhost.example.com loghost
```

- b. (옵션) NIS를 사용하지 않는 경우 네트워크의 각 시스템에서 `/etc/hosts` 파일을 편집합니다.

다음 항목을 만듭니다.

```
IP-address mailhost mailhost mailhost.domain loghost
```

- 5 **sendmail**을 다시 시작합니다.

```
# svcadm enable network/smtp:sendmail
```

- 6 메일 구성을 테스트합니다.

자세한 내용은 307 페이지 “메일 구성 테스트 방법”을 참조하십시오.

---

주 - 메일 호스트에 대한 자세한 내용은 14 장, “메일 서비스(참조)”의 326 페이지 “하드웨어 구성 요소”를 참조하십시오.

---

## ▼ 메일 게이트웨이 설정 방법

메일 게이트웨이는 도메인 외부 네트워크와의 통신을 관리합니다. 보내는 메일 게이트웨이의 메일러가 받는 시스템의 메일러와 일치할 수 있습니다.

이더넷과 전화선에 연결된 시스템이 메일 게이트웨이로 적합합니다. 인터넷에 대한 라우터로 구성된 시스템도 좋습니다. 메일 호스트나 다른 시스템을 메일 게이트웨이로 구성할 수 있습니다. 도메인에 둘 이상의 메일 게이트웨이를 구성할 수도 있습니다. UUCP(UNIX-to-UNIX Copy Program) 연결이 있을 경우 UUCP 연결이 있는 시스템을 메일 게이트웨이로 구성해야 합니다.

- 1 메일 게이트웨이 시스템의 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스**의 “관리 권한을 얻는 방법”을 참조하십시오.

- 2 **sendmail**을 중지합니다.

```
# svcadm disable -t network/smtp:sendmail
```

- 3 **host-name** 구성을 확인합니다.

`check-hostname` 스크립트를 실행하여 `sendmail`이 이 서버의 정규화된 호스트 이름을 식별할 수 있는지 확인합니다.

```
# /usr/sbin/check-hostname
hostname phoenix OK: fully qualified as phoenix.example.com
```

이 스크립트로 정규화된 호스트 이름을 식별하지 못하면 `/etc/hosts`에서 정규화된 호스트 이름을 호스트의 첫번째 별칭으로 추가해야 합니다. 이 단계에서 도움이 필요할 경우 281 페이지 “메일 호스트 설정 방법”의 단계 4를 참조하십시오.

4 이름 서비스가 시작되었는지 확인합니다.

a. (옵션)NIS를 실행 중이면 다음 명령을 사용합니다.

```
# ypwhich
```

자세한 내용은 [ypwhich\(1\)](#) 매뉴얼 페이지를 참조하십시오.

b. (옵션)DNS를 실행 중이면 다음 명령을 사용합니다.

```
# nslookup hostname
```

*hostname* 호스트 이름을 사용합니다.

자세한 내용은 [nslookup\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

c. (옵션)LDAP를 실행 중이면 다음 명령을 사용합니다.

```
# ldaplist
```

자세한 내용은 [ldaplist\(1\)](#) 매뉴얼 페이지를 참조하십시오.

5 sendmail을 다시 시작합니다.

```
# svcadm enable network/smtp:sendmail
```

6 메일 구성을 테스트합니다.

자세한 내용은 [307 페이지](#) “메일 구성 테스트 방법”을 참조하십시오.

---

주 - 메일 게이트웨이에 대한 자세한 내용은 [14 장](#), “메일 서비스(참조)”의 [326 페이지](#) “하드웨어 구성 요소”를 참조하십시오.

---

## ▼ sendmail과 함께 DNS를 사용하는 방법

DNS 이름 서비스는 개인의 별칭을 지원하지 않습니다. 이 이름 서비스는 MX(메일 교환기) 레코드 및 CNAME 레코드를 사용하는 호스트나 도메인에 별칭을 지원합니다. DNS 데이터베이스에서 호스트 이름, 도메인 이름 또는 둘 다 지정할 수 있습니다. sendmail 및 DNS에 대한 자세한 내용은 [14 장](#), “메일 서비스(참조)”의 [345 페이지](#) “sendmail과 이름 서비스의 상호 작용”을 참조하거나 [Oracle Solaris Administration: Naming and Directory Services](#) 를 참조하십시오.

1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스](#)의 “관리 권한을 얻는 방법”을 참조하십시오.

## 2 mailhost 및 mailhost.domain 항목이 있는지 확인합니다.

nslookup을 사용하여 DNS 데이터베이스에 mailhost 및 mailhost.domain에 대한 항목이 있는지 확인합니다. 자세한 내용은 [nslookup\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

# sendmail 구성 변경(작업 맵)

작업	설명	수행 방법
sendmail 구성 파일 작성	이 절차를 사용하여 <code>sendmail.cf</code> 파일을 수정합니다. 도메인 가장을 사용하여 설정하는 방법에 대한 예가 포함됩니다.	286 페이지 “새 <code>sendmail.cf</code> 파일 작성 방법”
가상 호스트 설정	둘 이상의 도메인에 대해 메일을 수락하도록 sendmail을 구성하는 단계입니다.	287 페이지 “가상 호스트 설정”
sendmail 구성 파일의 자동 재작성 설정	이 절차를 사용하여 업그레이드 후 <code>sendmail.cf</code> 및 <code>submit.mc</code> 구성 파일이 자동으로 재작성되도록 sendmail 서비스를 수정합니다.	287 페이지 “구성 파일을 자동으로 다시 작성하는 방법”
열기 모드에서 sendmail 실행	이 절차를 사용하여 열기 모드를 사용하여 설정하도록 sendmail 서비스 등록 정보를 수정합니다.	288 페이지 “열기 모드에서 sendmail 사용 방법”
TLS(전송 계층 보안)를 사용하도록 SMTP 설정	이 프로시저를 사용하여 SMTP가 TLS와의 보안 연결을 보유하도록 합니다.	289 페이지 “TLS를 사용하도록 SMTP를 설정하는 방법”
대체 구성을 사용하여 메일 배달 관리	이 절차를 사용하여 마스터 데몬을 사용하여 설정한 경우 발생할 수 있는 메일 배달 문제를 방지합니다.	293 페이지 “ <code>sendmail.cf</code> 의 대체 구성을 사용하여 메일 배달을 관리하는 방법”

# sendmail 구성 변경

286 페이지 “새 `sendmail.cf` 파일 작성 방법”에서는 구성 파일을 작성하는 방법을 보여줍니다. 아직은 이전 버전의 `sendmail.cf` 파일을 사용할 수 있지만 새 형식을 사용하는 것이 가장 좋습니다.

자세한 내용은 다음을 참조하십시오.

- `/etc/mail/cf/README`에는 구성 프로세스에 대한 전체 설명이 있습니다.
- <http://www.sendmail.org>에는 sendmail 구성에 대한 온라인 정보가 있습니다.
- 14 장, “메일 서비스(참조)”의 318 페이지 “구성 파일 버전” 및 338 페이지 “sendmail 구성 파일”에서 일부 지침을 제공합니다.
- 363 페이지 “sendmail 버전 8.12의 추가 및 개정된 m4 구성 매크로”도 유용합니다.

## ▼ 새 sendmail.cf 파일 작성 방법

다음 절차에서는 새 구성 파일 작성 방법을 보여줍니다.

---

주-/usr/lib/mail/cf/main-v7sun.mc는 이제 /etc/mail/cf/cf/sendmail.mc입니다.

---

### 1 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

### 2 sendmail을 중지합니다.

```
# svcadm disable -t network/smtp:sendmail
```

### 3 변경할 구성 파일의 복사본을 만듭니다.

```
# cd /etc/mail/cf/cf
# cp sendmail.mc myhost.mc
```

*myhost* .mc 파일의 새 이름을 선택합니다.

### 4 필요하면 새 구성 파일(예: *myhost.mc*)을 편집합니다.

예를 들어, 다음 명령줄을 추가하여 도메인 가장을 사용으로 설정합니다.

```
# cat myhost.mc
```

```
..
MASQUERADE_AS('host.domain')
```

*host.domain* 원하는 호스트 이름과 도메인 이름을 사용합니다.

이 예에서 MASQUERADE\_AS는 \$j가 아니라 *host.domain*에서와 마찬가지로 보낸 메일에 레이블이 지정되도록 합니다.

### 5 m4를 사용하여 구성 파일을 작성합니다.

```
# make myhost.cf
```

### 6 -C 옵션을 사용하여 새 파일을 지정하여 새 구성 파일을 테스트합니다.

```
# /usr/lib/sendmail -C myhost.cf -v testaddr </dev/null
```

이 명령이 메시지를 표시하는 동안 *testaddr*로 메시지를 보냅니다. 시스템에서 sendmail 서비스를 다시 시작하지 않고 보내는 메일만 테스트할 수 있습니다. 아직 메일을 처리하지 않는 시스템의 경우 [307 페이지 “메일 구성 테스트 방법”](#)의 전체 테스트 절차를 사용하십시오.

### 7 원본의 복사본을 만든 다음 새 구성 파일을 설치합니다.

```
# cp /etc/mail/sendmail.cf /etc/mail/sendmail.cf.save
# cp myhost.cf /etc/mail/sendmail.cf
```

## 8 sendmail 서비스를 다시 시작합니다.

```
# svcadm enable network/smtp:sendmail
```

## 가상 호스트 설정

둘 이상의 IP 주소를 호스트에 지정해야 하는 경우 <http://www.sendmail.org/tips/virtualHosting> 웹 사이트를 참조하십시오. 이 사이트에는 sendmail을 사용하여 가상 호스트를 설정하는 방법에 대한 전체 지침이 있습니다. 그러나 “Sendmail Configuration(Sendmail 구성)” 절에서 다음과 같이 3b 단계를 수행하지 마십시오.

```
# cd sendmail-VERSION/cf/cf
# ./Build mailserver.cf
# cp mailserver.cf /etc/mail/sendmail.cf
```

대신 Oracle Solaris 운영 체제일 경우 다음 단계를 수행합니다.

```
# cd /etc/mail/cf/cf
# make mailserver.cf
# cp mailserver.cf /etc/mail/sendmail.cf
```

*mailserver* .cf 파일의 이름을 사용합니다.

285 페이지 “[sendmail 구성 변경](#)”에서는 구축 프로세스의 일부분과 동일한 세 단계를 간략하게 설명합니다.

/etc/mail/sendmail.cf 파일을 생성한 후 다음 단계로 진행하여 가상 사용자 테이블을 만들 수 있습니다.

## ▼ 구성 파일을 자동으로 다시 작성하는 방법

sendmail.cf나 submit.cf의 복사본을 작성한 경우 업그레이드 프로세스 중에 구성 파일이 대체되지 않습니다. 다음 절차에서는 sendmail.cf 파일이 자동으로 다시 작성되도록 sendmail 서비스 등록 정보를 구성하는 방법을 보여줍니다. 자동으로 submit.cf 구성 파일을 작성하는 방법에 대한 자세한 내용은 [예 13-1](#)을 참조하십시오. 두 파일을 모두 작성해야 하는 경우 이 절차를 조합할 수 있습니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 sendmail 등록 정보를 설정합니다.

```
# svccfg -s sendmail
svc:/network/smtp:sendmail> setprop config/path_to_sendmail_mc=/etc/mail/cf/cf/myhost.mc
svc:/network/smtp:sendmail> quit
```

### 3 sendmail 서비스를 새로 고치고 다시 시작합니다.

첫번째 명령은 실행 중인 스냅샷에 변경 사항을 적용합니다. 두번째 명령은 새 옵션을 사용하여 sendmail을 다시 시작합니다.

```
# svcadm refresh svc:/network/smtp:sendmail
# svcadm restart svc:/network/smtp:sendmail
```

#### 예 13-1 submit.cf의 자동 재작성 설정

이 절차는 submit.mc 구성 파일이 자동으로 재작성되는 것과 같은 sendmail 서비스를 구성합니다.

```
# svccfg -s sendmail-client:default
svc:/network/smtp:sendmail> setprop config/path_to_submit_mc=/etc/mail/cf/cf/submit-myhost.mc
svc:/network/smtp:sendmail> exit
# svcadm refresh svc:/network/sendmail-client
# svcadm restart svc:/network/sendmail-client
```

## ▼ 열기 모드에서 sendmail 사용 방법

기본적으로 로컬 전용 모드에서 실행되도록 sendmail 서비스가 변경되었습니다. 로컬 전용 모드는 로컬 호스트의 메일만 수락한다는 의미입니다. 다른 시스템의 메시지는 거부됩니다. 이전 릴리스는 모든 원격 시스템에서 받는 메일을 수락하도록 구성됩니다(열기 모드). 열기 모드를 사용하려면 다음 절차를 사용하십시오.



**주의** - 로컬 전용 모드에서 sendmail을 실행하는 것이 열기 모드에서 실행하는 것보다 훨씬 안전합니다. 이 절차를 따를 경우 잠재적 보안 위험을 알고 있어야 합니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 sendmail 등록 정보를 설정합니다.

```
# svccfg -s sendmail
svc:/network/smtp:sendmail> setprop config/local_only = false
svc:/network/smtp:sendmail> quit
```

### 3 sendmail 서비스를 새로 고치고 다시 시작합니다.

```
# svcadm refresh svc:/network/smtp:sendmail
# svcadm restart svc:/network/smtp:sendmail
```



## ▼ TLS를 사용하도록 SMTP를 설정하는 방법

SMTP는 버전 8.13의 sendmail에서 TLS(전송 계층 보안)를 사용할 수 있습니다. 이 서비스를 사용하면 SMTP 서버와 클라이언트에 인터넷을 통해 개인 인증 통신을 제공하며, 도청자와 공격자로부터 보호합니다. 이 서비스는 기본적으로 사용으로 설정되지 않습니다.

다음 절차에서는 샘플 데이터를 사용하여 sendmail에 TLS를 사용할 수 있도록 하는 인증서를 설정하는 방법을 보여줍니다. 자세한 내용은 [349 페이지 “sendmail 버전 8.13에서 TLS를 사용하는 SMTP 실행 지원”](#)을 참조하십시오.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 sendmail을 중지합니다.

```
# svcadm disable -t network/smtp:sendmail
```

### 3 sendmail에 TLS를 사용할 수 있게 하는 인증서를 설정합니다.

#### a. 다음을 완료합니다.

```
# cd /etc/mail
# mkdir -p certs/CA
# cd certs/CA
# mkdir certs crt newcerts private
# echo "01" > serial
# cp /dev/null index.txt
# cp /etc/sfw/openssl/openssl.cnf .
```

#### b. 기본 텍스트 편집기를 사용하여 openssl.cnf 파일에서 dir 값을 /etc/sfw/openssl에서 /etc/mail/certs/CA로 변경합니다.

#### c. openssl 명령줄 도구를 사용하여 TLS를 구현합니다.

다음 명령줄에서는 대화형 텍스트를 생성합니다.

```
# openssl req -new -x509 -keyout private/cakey.pem -out cacert.pem -days 365 \
-config openssl.cnf
Generating a 1024 bit RSA private key
.....+++++
.....+++++
writing new private key to 'private/cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
```

```
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) []:US
State or Province Name (full name) []:California
Locality Name (eg, city) []:Menlo Park
Organization Name (eg, company) [Unconfigured OpenSSL Installation]:Oracle
Organizational Unit Name (eg, section) []:Solaris
Common Name (eg, YOUR name) []:somehost.somedomain.example.com
Email Address []:someuser@example.com
```

req	이 명령은 인증서 요청을 만들고 처리합니다.
-new	이 req 옵션은 새 인증서 요청을 생성합니다.
-x509	이 req 옵션은 자체 서명 인증서를 만듭니다.
-keyout private/cakey.pem	이 req 옵션을 사용하면 새로 만든 개인 키의 파일 이름으로 private/cakey.pem을 지정할 수 있습니다.
-out cacert.pem	이 req 옵션을 사용하면 cacert.pem을 출력 파일로 지정할 수 있습니다.
-days 365	이 req 옵션을 사용하면 인증서를 365일 동안 인증할 수 있습니다. 기본값은 30입니다.
-config openssl.cnf	req 옵션을 사용하면 openssl.cnf를 구성 파일로 지정할 수 있습니다.

이 명령을 사용하려면 다음을 제공해야 합니다.

- Country Name(예: US)
- State or Province Name(예: California)
- Locality Name(예: Menlo Park)
- Organization Name(예: Oracle)
- Organizational Unit Name(예: Solaris)
- Common Name(시스템의 정규화된 호스트 이름) 자세한 내용은 [check-hostname\(1M\)](#) 매뉴얼 페이지를 참조하십시오.
- Email Address(예: someuser@example.com)

#### 4 (옵션) 새 보안 연결이 필요할 경우 새 인증서를 만들고 인증 기관을 통해 새 인증서에 서명합니다.

##### a. 새 인증서를 만듭니다.

```
# cd /etc/mail/certs/CA
# openssl req -nodes -new -x509 -keyout newreq.pem -out newreq.pem -days 365 \
-config openssl.cnf
Generating a 1024 bit RSA private key
.....+++++
.....+++++
```

writing new private key to 'newreq.pem'

-----

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

-----

Country Name (2 letter code) []:US

State or Province Name (full name) []:California

Locality Name (eg, city) []:Menlo Park

Organization Name (eg, company) [Unconfigured OpenSSL Installation]:Oracle

Organizational Unit Name (eg, section) []:Solaris

Common Name (eg, YOUR name) []:somehost.somedomain.example.com

Email Address []:someuser@example.com

이 명령을 사용하려면 3c단계에서 제공한 것과 동일한 정보를 제공해야 합니다.

이 예에서 인증서와 개인 키는 newreq.pem 파일에 있습니다.

## b. 인증 기관을 통해 새 인증서에 서명합니다.

```
# cd /etc/mail/certs/CA
```

```
# openssl x509 -x509toreq -in newreq.pem -signkey newreq.pem -out tmp.pem
```

```
Getting request Private Key
```

```
Generating certificate request
```

```
# openssl ca -config openssl.cnf -policy policy_anything -out newcert.pem -infiles tmp.pem
```

```
Using configuration from openssl.cnf
```

```
Enter pass phrase for /etc/mail/certs/CA/private/cakey.pem:
```

```
Check that the request matches the signature
```

```
Signature ok
```

```
Certificate Details:
```

```
Serial Number: 1 (0x1)
```

```
Validity
```

```
Not Before: Jun 23 18:44:38 2005 GMT
```

```
Not After : Jun 23 18:44:38 2006 GMT
```

```
Subject:
```

```
countryName = US
```

```
stateOrProvinceName = California
```

```
localityName = Menlo Park
```

```
organizationName = Oracle
```

```
organizationalUnitName = Solaris
```

```
commonName = somehost.somedomain.example.com
```

```
emailAddress = someuser@example.com
```

```
X509v3 extensions:
```

```
X509v3 Basic Constraints:
```

```
CA:FALSE
```

```
Netscape Comment:
```

```
OpenSSL Generated Certificate
```

```
X509v3 Subject Key Identifier:
```

```
93:D4:1F:C3:36:50:C5:97:D7:5E:01:E4:E3:4B:5D:0B:1F:96:9C:E2
```

```
X509v3 Authority Key Identifier:
```

```
keyid:99:47:F7:17:CF:52:2A:74:A2:C0:13:38:20:6B:F1:B3:89:84:CC:68
```

```
DirName:/C=US/ST=California/L=Menlo Park/O=Oracle/OU=Solaris/
```

```
CN=someuser@example.com/emailAddress=someuser@example.com
```

```
serial:00
```

```
Certificate is to be certified until Jun 23 18:44:38 2006 GMT (365 days)
Sign the certificate? [y/n]:y
```

```
1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
# rm -f tmp.pem
```

이 예에서 파일 newreq.pem에는 서명되지 않은 인증서와 개인 키가 있습니다. 파일 newcert.pem에 서명된 인증서가 있습니다.

x509 유틸리티      인증서 정보를 표시하고 인증서를 다양한 형태로 변환하며 인증서 요청에 서명합니다.

ca 응용 프로그램      다양한 형태의 인증서 요청에 서명하고 CRL(인증서 해지 목록)을 생성하는 데 사용됩니다.

- 5 .mc 파일에 다음 행을 추가하여 sendmail이 인증서를 사용할 수 있게 합니다.

```
define('confCACERT_PATH', '/etc/mail/certs')dnl
define('confCACERT', '/etc/mail/certs/CAcert.pem')dnl
define('confSERVER_CERT', '/etc/mail/certs/MYcert.pem')dnl
define('confSERVER_KEY', '/etc/mail/certs/MYkey.pem')dnl
define('confCLIENT_CERT', '/etc/mail/certs/MYcert.pem')dnl
define('confCLIENT_KEY', '/etc/mail/certs/MYkey.pem')dnl
```

자세한 내용은 350 페이지 “TLS를 사용하여 SMTP를 실행하기 위한 구성 파일 옵션”을 참조하십시오.

- 6 sendmail.cf 파일을 /etc/mail 디렉토리에 재작성하고 설치합니다.

자세한 내용은 285 페이지 “sendmail 구성 변경”을 참조하십시오.

- 7 openssl을 사용하여 만든 파일에서 .mc 파일에 정의한 파일로 심볼릭 링크를 만듭니다.

```
# cd /etc/mail/certs
# ln -s CA/cacert.pem CAcert.pem
# ln -s CA/newcert.pem MYcert.pem
# ln -s CA/newreq.pem MYkey.pem
```

- 8 보안 강화를 위해 그룹에 대한 읽기 권한과 MYkey.pem에 대한 기타 권한을 거부합니다.

```
# chmod go-r MYkey.pem
```

- 9 심볼릭 링크를 사용하여 confCACERT\_PATH에 지정된 디렉토리에 CA 인증서를 설치합니다.

```
# C=CAcert.pem
# ln -s $C 'openssl x509 -noout -hash < $C'.0
```

## 10 다른 호스트와 주고받는 메일의 보안을 위해 해당 호스트 인증서를 설치합니다.

- a. 다른 호스트의 `confCACERT` 옵션으로 정의한 파일을 `/etc/mail/certs/host.domain.cert.pem`에 복사합니다.  
`host.domain`을 다른 호스트의 정규화된 호스트 이름으로 대체합니다.
- b. 심볼릭 링크를 사용하여 `confCACERT_PATH`에 지정된 디렉토리에 CA 인증서를 설치합니다.

```
# C=host.domain.cert.pem
# ln -s $C 'openssl x509 -noout -hash < $C'.0
```

`host.domain`을 다른 호스트의 정규화된 호스트 이름으로 대체합니다.

## 11 sendmail을 다시 시작합니다.

```
# svcadm enable network/smtp:sendmail
```

### 예 13-2 Received: 메일 헤더

다음은 TLS를 사용하는 보안 메일의 Received: 헤더 예입니다.

```
Received: from his.example.com ([IPv6:2001:db8:3c4d:15::1a2f:1a2b])
  by her.example.com (8.13.4+Sun/8.13.4) with ESMTP id j2TNUB8i242496
  (version=TLSv1/SSLv3 cipher=DHE-RSA-AES256-SHA bits=256 verify=OK)
  for <janepc@her.example.com>; Tue, 29 Mar 2005 15:30:11 -0800 (PST)
Received: from her.example.com (her.city.example.com [192.168.0.0])
  by his.example.com (8.13.4+Sun/8.13.4) with ESMTP id j2TNU7cl571102
  version=TLSv1/SSLv3 cipher=DHE-RSA-AES256-SHA bits=256 verify=OK)
  for <janepc@her.example.com>; Tue, 29 Mar 2005 15:30:07 -0800 (PST)
```

verify의 값은 OK이며 인증이 성공적이라는 의미입니다. 자세한 내용은 [352 페이지](#) “TLS를 사용하여 SMTP를 실행하기 위한 매크로”를 참조하십시오.

참조 다음 OpenSSL 매뉴얼 페이지를 참조하십시오.

- `openssl(1)` (<http://www.openssl.org/docs/apps/openssl.html>)
- `req(1)` (<http://www.openssl.org/docs/apps/req.html>)
- `x509(1)` (<http://www.openssl.org/docs/apps/x509.html>)
- `ca(1)` (<http://www.openssl.org/docs/apps/ca.html>)

## ▼ sendmail.cf의 대체 구성을 사용하여 메일 배달을 관리하는 방법

인바운드 메일과 아웃바운드 메일을 쉽게 전송하기 위해 sendmail의 새 기본 구성에 데몬 및 클라이언트 대기열 실행자가 사용됩니다. 클라이언트 대기열 실행자는 로컬 SMTP 포트의 데몬에 메일을 제출할 수 있어야 합니다. 데몬이 SMTP 포트에서 수신

대기하지 않는 경우 메일이 대기열에 남아 있습니다. 이 문제를 방지하려면 다음 작업을 수행하십시오. 데몬 및 대기열 실행자에 대한 자세한 내용을 확인하고, 이 대체 구성을 사용하는 이유를 이해하려면 [358 페이지 “sendmail 버전 8.12의 submit.cf 구성 파일”](#)을 참조하십시오.

이 절차에서는 로컬 호스트의 연결만 수락하기 위해 데몬이 실행되는지 확인합니다.

#### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

#### 2 sendmail 클라이언트 서비스를 중지합니다.

```
# svcadm disable -t sendmail-client
```

#### 3 변경할 구성 파일의 복사본을 만듭니다.

```
# cd /etc/mail/cf/cf
# cp submit.mc submit-myhost.mc
myhost .mc 파일의 새 이름을 선택합니다.
```

#### 4 새 구성 파일(예: submit-myhost.mc)을 편집합니다.

수신 호스트 IP 주소를 msp 정의로 변경합니다.

```
# grep msp submit-myhost.mc
FEATURE('msp', '[#.#.#]')dn1
```

#### 5 m4를 사용하여 구성 파일을 작성합니다.

```
# make submit-myhost.cf
```

#### 6 원본의 복사본을 만든 다음 새 구성 파일을 설치합니다.

```
# cp /etc/mail/submit.cf /etc/mail/submit.cf.save
# cp submit-myhost.cf /etc/mail/submit.cf
```

#### 7 sendmail 클라이언트 서비스를 다시 시작합니다.

```
# svcadm enable sendmail-client
```

## 편지 별칭 파일 관리(작업 맵)

다음 표에서는 편지 별칭 파일 관리를 위한 절차에 대해 설명합니다. 이 항목에 대한 자세한 내용은 [14 장, “메일 서비스\(참조\)”](#)의 [339 페이지 “메일 별칭 파일”](#)을 참조하십시오.

작업	설명	수행 방법
NISmail.aliases 맵 설정	이름 서비스가 NIS인 경우 다음 지침에 따라 mail.aliases 맵으로 간편하게 별칭을 지정합니다.	295 페이지 “NISmail.aliases 맵 설정 방법”
로컬 편지 별칭 파일 설정	NIS 등의 이름 서비스를 사용하지 않는 경우 다음 지침에 따라 /etc/mail/aliases 파일로 간편하게 별칭을 지정합니다.	296 페이지 “로컬 편지 별칭 파일 설정 방법”
키 맵 파일 만들기	이 단계를 사용하여 키 맵 파일로 간편하게 별칭을 지정합니다.	298 페이지 “키 맵 파일을 만드는 방법”
postmaster 별칭 설정	이 절의 절차를 사용하여 postmaster 별칭을 관리합니다. 이 별칭을 갖고 있어야 합니다.	298 페이지 “postmaster 별칭 관리”

## 편지 별칭 파일 관리

편지 별칭은 도메인에서 고유해야 합니다. 이 절에는 편지 별칭 파일을 관리하기 위한 절차가 있습니다.

또한 makemap을 사용하여 로컬 메일 호스트의 데이터베이스 파일을 만들 수 있습니다. [makemap\(1M\)](#) 매뉴얼 페이지를 참조하십시오. 이 데이터베이스 파일을 사용해도 NIS 등의 이름 서비스를 사용하여 얻을 수 있는 모든 이점을 얻을 수는 없습니다. 그러나 네트워크 조화가 포함되지 않으므로 이 로컬 데이터베이스 파일에서 더 빠르게 데이터를 검색할 수 있습니다. 자세한 내용은 14 장, “메일 서비스(참조)”의 345 페이지 “sendmail과 이름 서비스의 상호 작용” 및 339 페이지 “메일 별칭 파일”을 참조하십시오.

### ▼ NIS mail.aliases 맵 설정 방법

다음 절차를 사용하여 NISmail.aliases 맵으로 간편하게 별칭을 지정합니다.

- 1 각 메일 클라이언트 목록, 우편함 위치 및 메일 서버 시스템 이름을 컴파일합니다.
- 2 NIS 마스터 서버의 관리자가 됩니다.  
자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.
- 3 /etc/mail/aliases 파일을 편집하고 다음 항목을 만듭니다.
  - a. 각 메일 클라이언트에 대한 항목을 추가합니다.

```
# cat /etc/mail/aliases
...
alias:expanded-alias
```

*alias* 짧은 별칭을 사용합니다.

*expanded-alias* 확장된 별칭(*user@host.domain.com*)을 사용합니다.

**b. Postmaster: root 항목이 있는지 확인합니다.**

```
# cat /etc/mail/aliases
..
Postmaster: root
```

**c. root의 별칭을 추가합니다. 포스트마스터로 지정된 사용자의 메일 주소를 사용합니다.**

```
# cat /etc/mail/aliases
..
root: user@host.domain.com
```

*user@host.domain.com* 지정된 포스트마스터의 지정된 주소를 사용합니다.

**4 NIS 마스터 서버에서 이름 서비스를 실행하여 각 메일 서버의 호스트 이름을 확인해야 합니다.**

**5 /var/yp 디렉토리로 변경합니다.**

```
# cd /var/yp
```

**6 make 명령을 적용합니다.**

```
# make
```

/etc/hosts 및 /etc/mail/aliases 파일의 변경 사항이 NIS 슬레이브 시스템으로 전파됩니다. 몇 분 내에 변경 내용이 반영됩니다.

## ▼ 로컬 편지 별칭 파일 설정 방법

다음 절차를 사용하여 로컬 편지 별칭 파일로 별칭을 확인합니다.

**1 각 사용자 목록과 해당 우편함 위치를 컴파일합니다.**

**2 메일 서버에서 관리자 root가 됩니다.**

자세한 내용은 [Oracle Solaris 관리: 보안 서비스](#)의 “관리 권한을 얻는 방법”을 참조하십시오.

**3 /etc/mail/aliases 파일을 편집하고 다음 항목을 만듭니다.**

**a. 각 사용자에게 대한 항목을 추가합니다.**

```
user1: user2@host.domain
```

*user1* 새 별칭을 사용합니다.



`user2@host.domain` 새 별칭에 실제 주소를 사용합니다.

**b. Postmaster: root 항목이 있는지 확인합니다.**

```
# cat /etc/mail/aliases
..
Postmaster: root
```

**c. root의 별칭을 추가합니다. 포스트마스터로 지정된 사용자의 메일 주소를 사용합니다.**

```
# cat /etc/mail/aliases
..
root: user@host.domain.com
```

`user@host.domain.com` 지정된 포스트마스터의 지정된 주소를 사용합니다.

**4 별칭 데이터베이스를 재구성합니다.**

```
# newaliases
```

`/etc/mail/sendmail.cf`에 있는 `AliasFile` 옵션의 구성은 이 명령이 단일 파일 `/etc/mail/aliases.db`와 파일 쌍 `/etc/mail/aliases.dir` 및 `/etc/mail/aliases.pag` 중 어느 것을 이진 형식으로 생성하는지 결정합니다.

**5 다음 단계를 수행하여 생성된 파일을 복사합니다.**

**a. (옵션) `/etc/mail/aliases`, `/etc/mail/aliases.dir` 및 `/etc/mail/aliases.pag` 파일을 각각의 다른 시스템으로 복사합니다.**

`rcp` 또는 `rsync` 명령을 사용하여 파일 3개를 복사할 수 있습니다. 자세한 내용은 [rcp\(1\)](#) 매뉴얼 페이지 또는 `rsync(1)` 매뉴얼 페이지를 참조하십시오. 또는 이 용도로 스크립트를 만들 수 있습니다.

이 파일을 복사할 때 각각의 다른 시스템에서 `newaliases` 명령을 실행할 필요는 없습니다. 그러나 메일 클라이언트를 추가하거나 제거할 때마다 `/etc/mail/aliases` 파일을 모두 업데이트해야 합니다.

**b. (옵션) `/etc/mail/aliases` 및 `/etc/mail/aliases.db` 파일을 각각의 다른 시스템으로 복사합니다.**

`rcp` 또는 `rsync` 명령을 사용하여 이 파일을 복사할 수 있습니다. 자세한 내용은 [rcp\(1\)](#) 매뉴얼 페이지 또는 `rsync(1)` 매뉴얼 페이지를 참조하십시오. 또는 이 용도로 스크립트를 만들 수 있습니다.

이 파일을 복사할 때 각각의 다른 시스템에서 `newaliases` 명령을 실행할 필요는 없습니다. 그러나 메일 클라이언트를 추가하거나 제거할 때마다 `/etc/mail/aliases` 파일을 모두 업데이트해야 합니다.

## ▼ 키 맵 파일을 만드는 방법

키 맵 파일을 만들려면 다음 지침을 따릅니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 입력 파일을 만듭니다.

항목에 다음 구문을 포함할 수 있습니다.

```
old-name@newdomain.com    new-name@newdomain.com
old-name@olddomain.com    error:nouser No such user here
@olddomain.com            %1@newdomain.com
```

*old\_name@newdomain.com*      새로 지정된 도메인과 함께 이전에 지정된 사용자 이름을 사용합니다.

*new\_name@newdomain.com*      새로 지정된 주소를 사용합니다.

*old\_name@olddomain.com*      이전에 지정된 도메인과 함께 이전에 지정된 사용자 이름을 사용합니다.

*olddomain.com*              이전에 지정된 도메인을 사용합니다.

*newdomain.com*              새로 지정된 도메인을 사용합니다.

첫번째 항목이 메일을 새 별칭에 재지정합니다. 잘못된 별칭이 사용되면 다음 항목이 메시지를 만듭니다. 마지막 항목은 받는 메일을 모두 *olddomain*에서 *newdomain*으로 재지정합니다.

### 3 데이터베이스 파일을 만듭니다.

```
# /usr/sbin/makemap maptype newmap < newmap
```

*maptype*      dbm, btree 또는 hash 등의 데이터베이스 유형을 선택합니다.

*newmap*      입력 파일 이름과 데이터베이스 파일 이름의 첫번째 부분을 사용합니다. dbm 데이터베이스 유형이 선택된 경우 .pag 및 .dir 접미어를 사용하여 데이터베이스 파일이 생성됩니다. 나머지 데이터베이스 유형 두 개의 경우 파일 이름 뒤에 .db가 옵니다.

## postmaster 별칭 관리

모든 시스템이 postmaster 우편함에 메일을 보낼 수 있어야 합니다. postmaster의 NIS 별칭을 만들거나 각 로컬 /etc/mail/aliases 파일에서 별칭을 만들 수 있습니다. 다음 절차를 참조하십시오.

- [299 페이지 “각 로컬 /etc/mail/aliases 파일에서 postmaster 별칭을 만드는 방법”](#)

- 299 페이지 “postmaster에 대해 별도의 우편함을 만드는 방법”
- 300 페이지 “/etc/mail/aliases 파일에서 postmaster 우편함을 별칭에 추가하는 방법”

## ▼ 각 로컬 /etc/mail/aliases 파일에서 postmaster 별칭을 만드는 방법

각 로컬 /etc/mail/aliases 파일에서 postmaster 별칭을 만드는 경우 다음 지침을 따릅니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 /etc/mail/aliases 항목을 봅니다.

```
# cat /etc/mail/aliases
# Following alias is required by the mail protocol, RFC 2821
# Set it to the address of a HUMAN who deals with this system's
# mail problems.
Postmaster: root
```

### 3 각 시스템의 /etc/mail/aliases 파일을 편집합니다.

포스트마스터로 지정된 사용자의 메일 주소로 root를 변경합니다.

```
Postmaster: mail-address
```

*mail-address*     포스트마스터로 지정된 사용자에 대해 지정된 주소를 사용합니다.

### 4 (옵션) 포스트마스터에 대해 별도의 우편함을 만듭니다.

포스트마스터에 대해 별도의 우편함을 만들어 포스트마스터 메일을 개인 메일과 분리할 수 있습니다. 별도의 우편함을 만들 경우 /etc/mail/aliases 파일을 편집할 때 포스트마스터의 개인 메일 주소 대신 우편함 주소를 사용합니다. 자세한 내용은 299 페이지 “postmaster에 대해 별도의 우편함을 만드는 방법”을 참조하십시오.

## ▼ postmaster에 대해 별도의 우편함을 만드는 방법

postmaster에 대해 별도의 우편함을 만드는 경우 다음 지침을 따릅니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 postmaster로 지정된 사용자의 사용자 계정을 만듭니다. 암호 필드에 별표(\*)를 입력합니다.

사용자 계정 추가에 대한 자세한 내용은 [Oracle Solaris 관리: 일반 작업의 “사용자 계정 설정 및 관리\(작업 맵\)”](#)를 참조하십시오.

- 3 메일이 배달된 후 mail 프로그램이 우편함 이름에 쓰고 읽을 수 있도록 설정합니다.

```
# mail -f postmaster
```

postmaster 지정된 주소를 사용합니다.

▼ **/etc/mail/aliases 파일에서 postmaster 우편함을 별칭에 추가하는 방법**

/etc/mail/aliases 파일에서 postmaster 우편함을 별칭에 추가하는 경우 다음 지침을 따릅니다.

- 1 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

- 2 root의 별칭을 추가합니다. 포스트마스터로 지정된 사용자의 메일 주소를 사용합니다.

```
# cat /etc/mail/aliases
```

```
..
```

```
root: user@host.domain.com
```

user@host.domain.com 포스트마스터로 지정된 사용자의 지정된 주소를 사용합니다.

- 3 포스트마스터의 로컬 시스템에서 별칭의 이름을 정의하는 /etc/mail/aliases 파일에 항목을 만듭니다. sysadmin은 예입니다. 또한 로컬 우편함의 경로를 포함합니다.

```
# cat /etc/mail/aliases
```

```
..
```

```
sysadmin: /usr/somewhere/somefile
```

sysadmin 새 별칭의 이름을 만듭니다.

/usr/somewhere/somefile 로컬 우편함의 경로를 사용합니다.

- 4 별칭 데이터베이스를 재구성합니다.

```
# newaliases
```

## 대기열 디렉토리 관리(작업 맵)

다음 표에서는 메일 대기열을 관리하기 위한 절차에 대해 설명합니다.

작업	설명	수행 방법
메일 대기열 /var/spool/mqueue의 콘텐츠 표시	이 절차를 사용하여 대기열에 있을 수 있는 메시지의 수 및 메시지가 대기열에서 얼마나 빨리 지워지는지를 확인합니다.	301 페이지 “메일 대기열 /var/spool/mqueue의 콘텐츠 표시 방법”

작업	설명	수행 방법
메일 대기열 <code>/var/spool/mqueue</code> 에 대해 메일 대기열 처리 강제 실행	이 절차를 사용하여 이전에 메시지를 수신할 수 없던 시스템에 대해 메시지를 처리합니다.	302 페이지 “메일 대기열 <code>/var/spool/mqueue</code> 에서 메일 대기열 처리 강제 실행 방법”
메일 대기열 <code>/var/spool/mqueue</code> 의 일부 실행	이 절차를 사용하여 호스트 이름과 같은 주소의 하위 문자열을 강제로 처리합니다. 또한 특정 메시지를 대기열에서 강제로 내보냅니다.	302 페이지 “메일 대기열 <code>/var/spool/mqueue</code> 의 일부를 실행하는 방법”
메일 대기열 <code>/var/spool/mqueue</code> 이동	이 절차를 사용하여 메일 대기열을 이동합니다.	303 페이지 “메일 대기열 <code>/var/spool/mqueue</code> 이동 방법”
이전의 메일 대기열 <code>/var/spool/omqueue</code> 실행	이 절차를 사용하여 이전의 메일 대기열을 실행합니다.	303 페이지 “이전의 메일 대기열 <code>/var/spool/omqueue</code> 실행 방법”

## 대기열 디렉토리 관리

이 절에서는 대기열 관리에 도움이 되는 몇 가지 작업에 대해 설명합니다. 클라이언트 전용 대기열에 대한 자세한 내용은 358 페이지 “`sendmail` 버전 8.12의 `submit.cf` 구성 파일”을 참조하십시오. 기타 관련 정보는 368 페이지 “`sendmail` 버전 8.12의 추가 대기열 기능”을 참조하십시오.

다음을 참조하십시오.

- 301 페이지 “메일 대기열 `/var/spool/mqueue`의 콘텐츠 표시 방법”
- 302 페이지 “메일 대기열 `/var/spool/mqueue`에서 메일 대기열 처리 강제 실행 방법”
- 302 페이지 “메일 대기열 `/var/spool/mqueue`의 일부를 실행하는 방법”
- 303 페이지 “메일 대기열 `/var/spool/mqueue` 이동 방법”
- 303 페이지 “이전의 메일 대기열 `/var/spool/omqueue` 실행 방법”

### ▼ 메일 대기열 `/var/spool/mqueue`의 콘텐츠 표시 방법

- 대기열에 있을 수 있는 메시지의 수 및 메시지가 대기열에서 얼마나 빨리 지워지는지를 표시합니다.

다음과 같이 입력하십시오.

```
# /usr/bin/mailq | more
```

이 명령은 다음 정보를 제공합니다.

- 대기열 ID
- 메시지 크기
- 메시지가 대기열에 들어간 날짜
- 메시지 상태
- 보낸 사람 및 받는 사람

또한 이 명령은 이제 인증 속성 `solaris.admin.mail.mailq`가 있는지 확인합니다. 검사가 성공하면 `sendmail`에 `-bp` 플래그를 지정하는 작업이 실행됩니다. 검사가 실패하면 오류 메시지가 인쇄됩니다. 기본적으로 이 인증 속성은 모든 사용자에게 대해 사용으로 설정됩니다. `prof_attr`의 사용자 항목을 수정하여 인증 속성을 사용 안함으로 설정할 수 있습니다. 자세한 내용은 `prof_attr(4)` 및 `mailq(1)`의 매뉴얼 페이지를 참조하십시오.

## ▼ 메일 대기열 `/var/spool/mqueue`에서 메일 대기열 처리 강제 실행 방법

예를 들어, 이 절차를 사용하여 이전에 메시지를 받을 수 없던 시스템에 대해 메시지를 처리합니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 대기열 처리를 강제 실행하고 대기열을 만들 때 작업 진행률을 표시합니다.

```
# /usr/lib/sendmail -q -v
```

## ▼ 메일 대기열 `/var/spool/mqueue`의 일부를 실행하는 방법

예를 들어, 이 절차를 사용하여 호스트 이름 등 주소의 하위 문자열을 강제로 처리합니다. 또한 특정 메시지를 대기열에서 강제로 내보냅니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 `-qRstring`을 사용하여 언제든지 메일 대기열의 일부를 실행합니다.

```
# /usr/lib/sendmail -qRstring
```

`string`    받는 사람 별칭이나 `user@host.domain`의 하위 문자열(예: 호스트 이름)을 사용합니다.

또는 `-qI nnnnn`을 사용하여 메일 대기열의 일부를 실행할 수 있습니다.

```
# /usr/lib/sendmail -qInnnnn
```

`nnnnn`    대기열 ID를 사용합니다.

## ▼ 메일 대기열 /var/spool/mqueue 이동 방법

메일 대기열을 이동하는 경우 다음 지침을 따릅니다.

### 1 메일 호스트의 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 sendmail 데몬을 강제 종료합니다.

```
# svcadm disable network/smtp:sendmail
```

이제 sendmail은 더 이상 대기열 디렉토리를 처리하지 않습니다.

### 3 /var/spool 디렉토리로 변경합니다.

```
# cd /var/spool
```

### 4 디렉토리 mqueue 및 해당 콘텐츠를 모두 omqueue 디렉토리로 이동합니다. 그런 다음 이름이 mqueue인 빈 디렉토리를 새로 만듭니다.

```
# mv mqueue omqueue; mkdir mqueue
```

### 5 디렉토리 권한을 읽기/쓰기/소유자에 의해 실행 및 읽기/그룹에 의해 실행으로 설정합니다. 또한 소유자 및 그룹을 daemon으로 설정합니다.

```
# chmod 750 mqueue; chown root:bin mqueue
```

### 6 sendmail을 시작합니다.

```
# svcadm enable network/smtp:sendmail
```

## ▼ 이전의 메일 대기열 /var/spool/omqueue 실행 방법

이전의 메일 대기열을 실행하려면 다음 지침을 따릅니다.

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 이전의 메일 대기열을 실행합니다.

```
# /usr/lib/sendmail -oQ/var/spool/omqueue -q
```

-oQ 플래그는 대체 대기열 디렉토리를 지정합니다. -q 플래그는 대기열의 모든 작업을 실행하도록 지시합니다. 화면에 상세 정보 출력을 표시하는 경우 -v 플래그를 사용합니다.

### 3 빈 디렉토리를 제거합니다.

```
# rmdir /var/spool/omqueue
```

## .forward 파일 관리(작업 맵)

다음 표에서는 .forward 파일 관리를 위한 절차에 대해 설명합니다. 자세한 내용은 14 장, “메일 서비스(참조)”의 342 페이지 “.forward 파일”을 참조하십시오.

작업	설명	수행 방법
.forward 파일 사용 안함	예를 들어, 이 절차를 사용하여 자동 전달을 방지할 수 있습니다.	304 페이지 “.forward 파일을 사용 안함으로 설정하는 방법”
.forward 파일 검색 경로 변경	예를 들어, 이 절차를 사용하여 모든 .forward 파일을 공통 디렉토리로 이동할 수 있습니다.	305 페이지 “.forward 파일 검색 경로 변경 방법”
/etc/shells 만들기 및 채우기	이 절차를 사용하여 사용자가 .forward 파일을 사용하여 프로그램이나 파일에 메일을 전달할 수 있게 합니다.	306 페이지 “/etc/shells를 만들고 채우는 방법”

## .forward 파일 관리

이 절에는 .forward 파일 관리에 관련된 몇 가지 절차가 있습니다. 사용자가 이러한 파일을 편집할 수 있으므로 파일로 인해 문제가 발생할 수 있습니다. 자세한 내용은 14 장, “메일 서비스(참조)”의 342 페이지 “.forward 파일”을 참조하십시오.

다음은 참조하십시오.

- 304 페이지 “.forward 파일을 사용 안함으로 설정하는 방법”
- 305 페이지 “.forward 파일 검색 경로 변경 방법”
- 306 페이지 “/etc/shells를 만들고 채우는 방법”

### ▼ .forward 파일을 사용 안함으로 설정하는 방법

자동 전달을 방지하는 이 절차에서는 특정 호스트에 대해 .forward 파일을 사용 안함으로 설정합니다.

#### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스](#)의 “관리 권한을 얻는 방법”을 참조하십시오.



- 2 /etc/mail/cf/domain/solaris-generic.m4 또는 사이트 관련 도메인 m4 파일의 복사본을 만듭니다.

```
# cd /etc/mail/cf/domain
# cp solaris-generic.m4 mydomain.m4
```

*mydomain*     사용자가 선택한 파일 이름을 사용합니다.

- 3 방금 만든 파일에 다음 행을 추가합니다.

```
define('confFORWARD_PATH','')dnl
```

confFORWARD\_PATH의 값이 이미 m4 파일에 있으면 이 null 값으로 해당 값을 대체합니다.

- 4 새 구성 파일을 작성 및 설치합니다.

이 단계에서 도움이 필요할 경우 [286 페이지](#) “새 sendmail.cf 파일 작성 방법”을 참조하십시오.

---

주 - .mc 파일을 편집할 때는 DOMAIN('solaris-generic')을 DOMAIN('mydomain')으로 변경해야 합니다.

---

## ▼ .forward 파일 검색 경로 변경 방법

예를 들어 모든 .forward 파일을 공통 디렉토리에 배치하려면 다음 지침을 따르십시오.

- 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 /etc/mail/cf/domain/solaris-generic.m4 또는 사이트 관련 도메인 m4 파일의 복사본을 만듭니다.

```
# cd /etc/mail/cf/domain
# cp solaris-generic.m4 mydomain.m4
```

*mydomain*     사용자가 선택한 파일 이름을 사용합니다.

- 3 방금 만든 파일에 다음 행을 추가합니다.

```
define('confFORWARD_PATH','$z/.forward:/var/forward/$u')dnl
```

confFORWARD\_PATH의 값이 이미 m4 파일에 있으면 이 새로운 값으로 해당 값을 대체합니다.

- 4 새 구성 파일을 작성 및 설치합니다.

이 단계에서 도움이 필요할 경우 [286 페이지](#) “새 sendmail.cf 파일 작성 방법”을 참조하십시오.

주 - .mc 파일을 편집할 때는 DOMAIN('solaris-generic')을 DOMAIN('mydomain')으로 변경해야 합니다.

## ▼ /etc/shells를 만들고 채우는 방법

이 파일은 표준 릴리스에 포함되지 않습니다. 사용자가 .forward 파일을 사용하여 프로그램이나 파일에 메일을 전달하도록 허용된 경우 파일을 추가해야 합니다. grep을 사용하여 암호 파일에 나열된 모든 셸을 식별하는 경우 수동으로 파일을 만들 수 있습니다. 그런 다음 파일에 셸을 입력할 수 있습니다. 그러나 다음 절차에서는 다운로드할 수 있는 스크립트를 사용하므로 더 간편합니다.

### 1 스크립트를 다운로드합니다.

<http://www.sendmail.org/vendor/sun/gen-etc-shells.html>

### 2 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

### 3 셸 목록을 생성하려면 gen-etc-shells 스크립트를 실행합니다.

```
# ./gen-etc-shells.sh > /tmp/shells
```

이 스크립트는 getent 명령을 사용하여 svc:/system/name-service/switch 서비스에 나열되는 암호 파일 소스에 포함된 셸 이름을 수집합니다.

### 4 /tmp/shells에서 셸 목록을 보고 편집합니다.

선택 항목의 편집기를 사용하여 포함하지 않을 셸을 제거합니다.

### 5 파일을 /etc/shells로 이동합니다.

```
# mv /tmp/shells /etc/shells
```

## 메일 서비스의 문제 해결 절차 및 팁(작업 맵)

다음 표에서는 메일 서비스를 위한 문제 해결 절차 및 팁에 대해 설명합니다.

작업	설명	수행 방법
메일 구성 테스트	sendmail 구성 파일 변경 사항 테스트 단계	<a href="#">307 페이지 “메일 구성 테스트 방법”</a>

작업	설명	수행 방법
편지 별칭 확인	지정된 받는 사람에게 메일이 배달될 수 있는지 여부를 확인하는 단계	308 페이지 “편지 별칭 확인 방법”
규칙 세트 테스트	sendmail 규칙 세트의 입력 및 반환 확인 단계	308 페이지 “sendmail 규칙 세트 테스트 방법”
다른 시스템에 대한 연결 확인	다른 시스템에 대한 연결 확인 팁	309 페이지 “다른 시스템에 대한 연결 확인 방법”
syslogd 프로그램을 사용하여 메시지 기록	오류 메시지 정보 수집 팁	310 페이지 “오류 메시지 기록”
기타 진단 정보 소스 확인	다른 소스에서 진단 정보를 가져오기 위한 팁	311 페이지 “기타 메일 진단 정보 소스”

## 메일 서비스의 문제 해결 절차 및 팁

이 절에는 메일 서비스의 문제 해결에 사용할 수 있는 몇 가지 절차와 팁이 있습니다.

### ▼ 메일 구성 테스트 방법

구성 파일에 대한 변경 사항을 테스트하려면 다음 절차를 따릅니다.

- 1 개정된 구성 파일이 있는 시스템에서 **sendmail**을 다시 시작합니다.

```
# svcadm refresh network/smtp:sendmail
```

- 2 각 시스템에서 테스트 메시지를 보냅니다.

```
# /usr/lib/sendmail -v names </dev/null
```

**names** 받는 사람 전자 메일 주소를 지정합니다.

이 명령은 지정된 받는 사람에게 **null** 메시지를 보내고 모니터에 메시지 작업을 표시합니다.

- 3 메시지 주소를 일반 사용자 이름으로 지정하여 자신이나 로컬 시스템의 다른 사용자에게 메일을 보냅니다.

- 4 (옵션) 네트워크에 연결된 경우 다른 시스템의 사용자에게 세 방향으로 메시지를 보냅니다.

- 주 시스템에서 클라이언트 시스템으로
- 클라이언트 시스템에서 주 시스템으로
- 클라이언트 시스템에서 다른 클라이언트 시스템으로

- 5 (옵션) 메일 게이트웨이가 있을 경우 메일 호스트에서 다른 도메인으로 메일을 보내 중계 메일러와 호스트가 제대로 구성되었는지 확인합니다.
- 6 (옵션) 전화선에서 다른 호스트로 UUCP 연결을 설정한 경우 해당 호스트에 있는 사용자에게 메일을 보냅니다. 메시지를 받으면 해당 사용자가 다시 메일을 보내거나 전화하도록 합니다.
- 7 다른 사용자에게 UUCP 연결을 통해 메일을 보내달라고 요청합니다.  
sendmail 프로그램은 배달을 위해 UUCP를 통과하므로 메시지가 배달되는지 여부를 알 수 없습니다.
- 8 다른 시스템에서 **postmaster**에게 메시지를 보내고 포스트마스터의 우편함으로 메시지가 배달되는지 확인합니다.

## 편지 별칭 확인 방법

다음 예에서는 별칭 확인 방법을 보여줍니다.

```
% mconnect
connecting to host localhost (127.0.0.1), port 25
connection open
220 your.domain.com ESMTP Sendmail 8.13.6+Sun/8.13.6; Tue, 12 Sep 2004 13:34:13 -0800 (PST)
expn sandy
250 2.1.5 <sandy@phoenix.example.com>
quit
221 2.0.0 your.domain.com closing connection
%
```

이 예에서 **mconnect** 프로그램은 로컬 호스트에서 메일 서버와의 연결을 열고 해당 연결을 테스트할 수 있게 해줍니다. 프로그램은 대화형으로 실행되므로 다양한 진단 명령을 실행할 수 있습니다. 자세한 내용은 **mconnect(1)** 매뉴얼 페이지를 참조하십시오. 항목 **expn sandy**는 확장된 주소 **sandy@phoenix.example.com**을 제공합니다. 따라서 별칭 **sandy**를 사용하여 메일을 배달할 수 있음을 확인했습니다.

로컬 및 도메인 차원의 별칭이 둘 다 사용되는 경우 루프와 일관성 없는 데이터베이스를 피해야 합니다. 특히 한 시스템에서 다른 시스템으로 사용자를 이동할 때 별칭 루프가 생성되지 않도록 주의하십시오.

## ▼ sendmail 규칙 세트 테스트 방법

sendmail 규칙 세트의 입력과 반환을 확인하려면 다음 지침을 따릅니다.

- 1 주소 테스트 모드로 변경합니다.

```
# /usr/lib/sendmail -bt
```

## 2 메일 주소를 테스트합니다.

마지막 프롬프트(>)에 다음 번호와 숫자를 입력합니다.

```
> 3,0 mail-sraddress
```

*mail-address* 테스트하려는 메일 주소를 사용합니다.

## 3 세션을 끝냅니다.

Ctrl-d를 누릅니다.

### 예 13-3 주소 테스트 모드 출력

다음은 주소 테스트 모드의 출력 예입니다.

```
% /usr/lib/sendmail -bt
ADDRESS TEST MODE (ruleset 3 NOT automatically invoked)
Enter <ruleset> <address>
> 3,0 sandy@phoenix
canonify          input: sandy @ phoenix
Canonify2         input: sandy < @ phoenix >
Canonify2         returns: sandy < @ phoenix . example . com . >
canonify          returns: sandy < @ phoenix . example . com . >
parse            input: sandy < @ phoenix . example . com . >
Parse0           input: sandy < @ phoenix . example . com . >
Parse0           returns: sandy < @ phoenix . example . com . >
ParseLocal       input: sandy < @ phoenix . example . com . >
ParseLocal       returns: sandy < @ phoenix . example . com . >
Parse1           input: sandy < @ phoenix . example . com . >
MailerToTriple   input: < mailhost . phoenix . example . com >
                 sandy < @ phoenix . example . com . >
MailerToTriple   returns: $# relay $# mailhost . phoenix . example . com
                 $: sandy < @ phoenix . example . com . >
Parse1          returns: $# relay $# mailhost . phoenix . example . com
                 $: sandy < @ phoenix . example . com . >
parse           returns: $# relay $# mailhost . phoenix . example . com
                 $: sandy < @ phoenix . example . com . >
```

## 다른 시스템에 대한 연결 확인 방법

mconnect 프로그램은 지정된 메일 서버와의 연결을 열고 해당 연결을 테스트할 수 있게 해줍니다. 프로그램은 대화형으로 실행되므로 다양한 진단 명령을 실행할 수 있습니다. 자세한 내용은 [mconnect\(1\)](#) 매뉴얼 페이지를 참조하십시오. 다음 예에서는 사용자 이름 sandy에 대한 메일을 배달할 수 있는지 확인합니다.

```
% mconnect phoenix

connecting to host phoenix (172.31.255.255), port 25
connection open
220 phoenix.example.com ESMTP Sendmail 8.13.1+Sun/8.13.1; Sat, 4 Sep 2004 3:52:56 -0700
expn sandy
```

```
250 2.1.5 <sandy@phoenix.example.com>
quit
```

mconnect를 사용하여 SMTP 포트에 연결할 수 없으면 다음 조건을 확인하십시오.

- 시스템 로드가 너무 높습니까?
- sendmail 데몬이 실행 중입니까?
- 시스템에 적절한 /etc/mail/sendmail.cf 파일이 있습니까?
- sendmail에 사용되는 포트 25가 활성화 상태입니까?

## 오류 메시지 기록

메일 서비스는 syslogd 프로그램을 사용하여 대부분의 오류 메시지를 기록합니다. 기본적으로 syslogd 프로그램은 loghost라는 시스템에 이 메시지를 보냅니다. 이 시스템은 /etc/hosts 파일에 지정되어 있습니다. loghost를 정의하여 전체 NIS 도메인의 로그를 모두 보관할 수 있습니다. loghost가 지정되지 않으면 syslogd의 오류 메시지가 보고되지 않습니다.

/etc/syslog.conf 파일은 syslogd 프로그램이 메시지를 전달하는 위치를 제어합니다. /etc/syslog.conf 파일을 편집하여 기본 구성을 변경할 수 있습니다. syslog 데몬을 다시 시작해야 변경 사항이 활성화됩니다. 메일에 대한 정보를 수집하려면 다음 선택 항목을 파일에 추가합니다.

- mail.alert – 지금 해결할 조건에 대한 메시지
- mail.crit – 중요 메시지
- mail.warning – 경고 메시지
- mail.notice – 오류는 아니지만 주의해야 할 메시지
- mail.info – 정보 메시지
- mail.debug – 디버깅 메시지

/etc/syslog.conf 파일에서 다음 항목이 중요, 정보, 디버그 메시지를 모두 /var/log/syslog로 보냅니다.

```
mail.crit;mail.info;mail.debug          /var/log/syslog
```

시스템 로그의 각 행에는 시간 기록, 행을 생성한 시스템의 이름 및 메시지가 있습니다. syslog 파일은 대량의 정보를 기록할 수 있습니다.

로그는 연속된 레벨로 배열됩니다. 가장 낮은 레벨에서는 비정상적인 사항만 기록됩니다. 가장 높은 레벨에서는 가장 일상적인 이벤트까지도 기록됩니다. 규약에 따라 10 이하의 로그 레벨이 “유용”합니다. 10보다 높은 로그 레벨은 주로 디버깅에 사용됩니다. loghost 및 syslogd 프로그램에 대한 자세한 내용은 [Oracle Solaris 관리: 일반 작업의 “시스템 메시지 로깅 사용자 정의”](#)를 참조하십시오.

## 기타 메일 진단 정보 소스

기타 진단 정보는 다음 소스를 확인하십시오.

- 메시지 헤더에서 **Received** 행을 확인합니다. 이 행은 메시지가 중계될 때 메시지가 이동한 경로를 추적합니다. 표준 시간대가 다르다는 점에 유의합니다.
- **MAILER-DAEMON**의 메시지를 확인합니다. 이 메시지는 대개 배달 문제를 보고합니다.
- 시스템 그룹의 배달 문제를 기록하는 시스템 로그를 확인합니다. **sendmail** 프로그램은 항상 시스템 로그에 작업을 기록합니다. **crontab** 파일을 수정하여 셀 스크립트를 야간에 실행할 수 있습니다. 스크립트는 로그에서 **SYSERR** 메시지를 검색하고 발견된 메시지를 포스트마스터에게 메일로 보냅니다.
- **mailstats** 프로그램을 사용하여 메일 유형을 테스트하고 받는 메시지와 보내는 메시지 수를 결정합니다.

## 오류 메시지 해결

이 절에서는 일부 **sendmail** 관련 오류 메시지를 해결하는 방법에 대해 설명합니다. <http://www.sendmail.org/faq>를 참조할 수도 있습니다.

다음 오류 메시지는 다음과 같은 유형의 정보 중 둘 이상이 있습니다.

- **원인:** 메시지를 발생하게 한 원인
- **설명:** 오류 메시지가 발생할 때 사용자가 수행하고 있던 작업
- **해결 방법:** 문제를 해결하거나 작업을 계속하기 위해 할 수 있는 일

### 451 timeout waiting for input during source

**원인:** **sendmail**이 SMTP 연결과 같이 시간이 초과될 수 있는 소스에서 읽을 경우 프로그램은 타이머를 다양한 **Timeout** 옵션의 값으로 설정한 후 읽기 시작합니다. 타이머가 만료되기 전에 읽기가 완료되지 않으면 이 메시지가 나타나고 읽기가 중지됩니다. 대개 **RCPT** 도중 이런 상황이 발생합니다. 그런 다음 나중에 배달하기 위해 메일 메시지가 대기열에 추가됩니다.

**해결책:** 이 메시지가 자주 나타나면 **/etc/mail/sendmail.cf** 파일에서 여러 **Timeout** 옵션의 값을 늘립니다. 타이머가 이미 더 큰 수로 설정되어 있으면 잘못된 네트워크 케이블 연결과 같은 하드웨어 문제를 찾습니다.

### 550 hostname... Host unknown

**원인:** 이 **sendmail** 메시지는 DNS(도메인 이름 시스템) 조회 도중 기호(@) 뒤의 주소 부분에 지정된 대상 호스트 시스템을 찾을 수 없음을 나타냅니다.

**해결책:** **nslookup** 명령을 사용하여 해당 도메인이나 다른 도메인에 대상 호스트가 있는지 확인합니다. 철자가 약간 달라서 발생하는 문제일 수 있습니다. 그렇지 않으면 예정된 받는 사람에게 연락하여 올바른 주소를 확인하십시오.

**550 *username...* User unknown**

**원인:** 이 `sendmail` 메시지는 기호(`@`) 앞의 주소 부분에 지정된 예정된 받는 사람이 대상 호스트 시스템에 없음을 나타냅니다.

**해결책:** 전자 메일 주소를 확인하고 다시 시도합니다. 철자가 약간 달라서 발생하는 문제일 수 있습니다. 그래도 해결되지 않으면 예정된 받는 사람에게 연락하여 올바른 주소를 확인하십시오.

**554 *hostname...* Local configuration error**

**원인:** 이 `sendmail` 메시지는 대개 로컬 호스트가 자신에게 메일을 보내려고 시도함을 나타냅니다.

**해결책:** `/etc/mail/sendmail.cf` 파일에서 `$j` 매크로 값을 검사하여 이 값이 정규화된 도메인 이름인지 확인합니다.

**설명:** SMTP HELO 명령에서 보내는 시스템이 받는 시스템에 호스트 이름을 제공할 때 받는 시스템이 이 이름과 받는 사람 이름을 비교합니다. 두 이름이 같으면 받는 시스템이 이 오류 메시지를 표시하고 연결을 닫습니다. HELO 명령에 제공된 이름은 `$j` 매크로 값입니다.

자세한 내용은 <http://www.sendmail.org/faq/section4#4.5>를 참조하십시오.

**config error: mail loops back to myself.**

**원인:** MX 레코드를 설정하고 호스트 *bar*를 도메인 *foo*의 메일 교환기로 설정하면 이 오류 메시지가 발생합니다. 그러나 도메인 *foo*의 메일 교환기임을 인식하도록 호스트 *bar*를 구성하는 데 실패합니다.

또한 보내는 시스템과 받는 시스템 둘 다 같은 도메인으로 식별될 수도 있습니다.

**해결책:** 자세한 지침은 <http://www.sendmail.org/faq/section4#4.5>를 참조하십시오.

**host name configuration error**

**설명:** `I refuse to talk to myself`를 대체한 이전의 `sendmail` 메시지이며 지금은 Local configuration error 메시지로 대체되었습니다.

**해결책:** 이 오류 메시지 554 *hostname...* Local configuration error를 해결하기 위해 제공된 지침을 따르십시오.

**user unknown**

**원인:** 사용자에게 메일을 보낼 때 오류 Username... user unknown이 표시됩니다. 사용자가 같은 시스템에 있습니다.

**해결책:** 입력한 전자 메일 주소에 인쇄상 오류가 있는지 확인하십시오. 그렇지 않으면 `/etc/mail/aliases` 또는 사용자의 `.mailrc` 파일에서 존재하지 않는 전자 메일 주소로



사용자의 별칭이 지정되었을 수 있습니다. 또한 사용자 이름에 대문자가 있는지 확인하십시오. 전자 메일 주소는 대소문자를 구분하지 않습니다.

자세한 내용은 <http://www.sendmail.org/faq/section4#4.17>을 참조하십시오.



## 메일 서비스(참조)

---

sendmail 프로그램은 메일 전송 에이전트입니다. 이 프로그램은 구성 파일을 사용하여 별칭과 전달, 네트워크 게이트에 대한 자동 경로 지정 및 유연한 구성을 제공합니다. Oracle Solaris OS에서는 대부분의 사이트에서 사용할 수 있는 표준 구성 파일을 제공합니다. 12 장, “메일 서비스(개요)”에서는 메일 서비스의 구성 요소를 소개하고 일반적인 메일 서비스 구성에 대해 설명합니다. 13 장, “메일 서비스(작업)”에서는 전자 메일 시스템 설정 및 관리 방법에 대해 설명합니다. 이 장에서는 다음과 같은 내용을 다룹니다.

- 316 페이지 “Oracle Solaris 버전의 sendmail”
- 319 페이지 “메일 서비스의 소프트웨어 및 하드웨어 구성 요소”
- 328 페이지 “메일 서비스 프로그램 및 파일”
- 344 페이지 “메일 주소 및 메일 경로 지정”
- 345 페이지 “sendmail과 이름 서비스의 상호 작용”
- 348 페이지 “sendmail 버전 8.14의 변경 사항”
- 349 페이지 “sendmail 버전 8.13의 변경 사항”
- 357 페이지 “sendmail 버전 8.12에서 변경된 사항”

이 장에서 다루지 않는 내용은 다음 매뉴얼 페이지를 참조하십시오.

- `sendmail(1M)`
- `mail.local(1M)`
- `mailstats(1)`
- `makemap(1M)`
- `editmap(1M)`

# Oracle Solaris 버전의 sendmail

이 절은 다음 항목으로 이루어져 있으며, 일반 Berkeley 버전과 Oracle Solaris 버전 sendmail의 몇 가지 차이점에 대해서 설명합니다.

- 316 페이지 “sendmail 컴파일에 사용되는 플래그 및 사용되지 않는 플래그”
- 317 페이지 “MILTER, sendmail용 메일 필터 API”
- 318 페이지 “대체 sendmail 명령”
- 318 페이지 “구성 파일 버전”

## sendmail 컴파일에 사용되는 플래그 및 사용되지 않는 플래그

다음과 같은 플래그가 sendmail 컴파일에 사용됩니다. 구성에 다른 플래그가 필요할 경우 소스를 다운로드하고 이진을 다시 컴파일해야 합니다. <http://www.sendmail.org>에서 이 프로세스에 대한 정보를 찾을 수 있습니다.

표 14-1 일반 sendmail 플래그

플래그	설명
SOLARIS=21000	Solaris 10 릴리스에 대한 지원
MILTER	메일 필터 API에 대한 지원. 버전 8.13의 sendmail에서는 기본적으로 이 플래그가 사용으로 설정됩니다. 317 페이지 “MILTER, sendmail용 메일 필터 API”를 참조하십시오.
NETINET6	IPv6에 대한 지원. 이 플래그는 conf.h에서 Makefile로 이동했습니다.

표 14-2 맵 및 데이터베이스 유형

플래그	설명
NDBM	ndbm 데이터베이스에 대한 지원
NEWDB	Berkeley DB 데이터베이스에 대한 지원
USERDB	사용자 데이터베이스에 대한 지원
NIS	nis 데이터베이스에 대한 지원
NISPLUS	nisplus 데이터베이스에 대한 지원
LDAPMAP	LDAP 맵에 대한 지원
MAP_REGEX	정규 표현식 맵에 대한 지원

표 14-3 OS 플래그

플래그	설명
SUN_EXTENSIONS	sun_compat.o에 포함된 확장에 대한 지원
SUN_INIT_DOMAIN	역방향 호환성을 위해 NIS 도메인 이름 사용을 지원하여 로컬 호스트 이름 정규화. 자세한 내용은 <a href="http://www.sendmail.org">http://www.sendmail.org</a> 에서 공급업체 관련 정보를 참조하십시오.
SUN_SIMPLIFIED_LDAP	간소화된 Sun 관련 LDAP API에 대한 지원. 자세한 내용은 <a href="http://www.sendmail.org">http://www.sendmail.org</a> 에서 공급업체 관련 정보를 참조하십시오.
VENDOR_DEFAULT=VENDOR_SUN	Sun을 기본 공급업체로 선택합니다.

다음 표에서는 sendmail 버전을 컴파일하는 데 사용되지 않는 일반 플래그를 나열합니다.

표 14-4 이 버전의 sendmail에 사용되지 않는 일반 플래그

플래그	설명
SASL	단순 인증 및 보안 계층(RFC 2554)
STARTTLS	트랜잭션 레벨 보안(RFC 2487)

sendmail을 컴파일하는 데 사용되는 플래그 목록을 보려면 다음 명령을 사용하십시오.

```
% /usr/lib/sendmail -bt -d0.10 < /dev/null
```

주 - preceding 명령은 Sun 관련 플래그를 나열하지 않습니다.

## MILTER, sendmail용 메일 필터 API

sendmail의 메일 필터 API인 MILTER를 사용하면 타사 프로그램이 메타 정보와 콘텐츠를 필터링하기 위해 처리될 때 메일 메시지에 액세스할 수 있습니다. 이를 사용하기 위해 필터를 구축하고 sendmail을 구성할 필요가 없습니다. sendmail 버전 8.13에서는 기본적으로 이 API가 사용으로 설정됩니다.

자세한 내용은 다음을 참조하십시오.

- <http://www.sendmail.org>
- <https://www.milter.org/>

## 대체 sendmail 명령

Oracle Solaris 릴리스에는 `sendmail.org`의 일반 릴리스에 제공되는 명령 동의어 중 일부가 포함되지 않습니다. 이 표에는 전체 명령 별칭 목록이 있습니다. 또한 명령이 Oracle Solaris 릴리스에 포함되어 있는지 여부와 `sendmail`을 사용하여 같은 동작을 생성하는 방법이 있습니다.

표 14-5 대체 sendmail 명령

대체 이름	이 릴리스에 포함 여부	sendmail과 함께 사용되는 옵션
hoststat	아니오	sendmail -bh
mailq	예	sendmail -bp
newaliases	예	sendmail -bi
purgestat	아니오	sendmail -bH
smtpd	아니오	sendmail -bd

## 구성 파일 버전

`sendmail`은 `sendmail.cf` 파일의 버전을 정의하는 데 사용할 수 있는 구성 옵션을 포함합니다. 이 옵션을 사용하면 이전 구성 파일을 현재 버전의 `sendmail`에 사용할 수 있습니다. 버전 레벨을 0과 10 사이의 값으로 설정할 수 있습니다. 공급업체도 정의할 수 있습니다. 유효한 공급업체 옵션은 Berkeley 또는 Sun입니다. 버전 레벨이 지정되고 공급업체는 정의되지 않은 경우 Sun이 기본 공급업체 설정으로 사용됩니다. 다음 표에서는 유효한 옵션 몇 가지를 나열합니다.

표 14-6 구성 파일의 버전 값

필드	설명
V7/Sun	버전 8.8의 <code>sendmail</code> 에 사용되는 설정입니다.
V8/Sun	버전 8.9의 <code>sendmail</code> 에 사용되는 설정입니다. 이 설정은 Solaris 8 릴리스에 포함되어 있습니다.
V9/Sun	버전 8.10 및 8.11의 <code>sendmail</code> 에 사용되는 설정입니다.
V10/Sun	버전 8.12, 8.13 및 버전 8.14의 <code>sendmail</code> 에 사용되는 설정입니다. 버전 8.12는 Solaris 9 릴리스의 기본값입니다. Solaris 10 릴리스부터는 버전 8.13이 기본값입니다. 버전 8.14는 Oracle Solaris 11 릴리스의 기본값입니다.

주-V1/Sun은 사용하지 마십시오. 자세한 내용은 <http://www.sendmail.org/vendor/sun/differences.html#4>를 참조하십시오.

작업 정보는 13 장, “메일 서비스(작업)”의 285 페이지 “sendmail 구성 변경”을 참조하십시오.

## 메일 서비스의 소프트웨어 및 하드웨어 구성 요소

이 절에서는 메일 시스템의 소프트웨어 및 하드웨어 구성 요소에 대해 설명합니다.

- 319 페이지 “소프트웨어 구성 요소”
- 326 페이지 “하드웨어 구성 요소”

### 소프트웨어 구성 요소

메일 서비스마다 다음 소프트웨어 구성 요소 중 하나 이상이 각각 포함됩니다.

- 319 페이지 “메일 사용자 에이전트”
- 319 페이지 “메일 전송 에이전트”
- 320 페이지 “로컬 배달 에이전트”

이 절에서는 다음 소프트웨어 구성 요소에 대해서도 설명합니다.

- 320 페이지 “메일러 및 sendmail”
- 321 페이지 “메일 주소”
- 323 페이지 “우편함 파일”
- 325 페이지 “메일 별칭”

### 메일 사용자 에이전트

메일 사용자 에이전트는 사용자와 메일 전송 에이전트 사이의 인터페이스 역할을 하는 프로그램입니다. sendmail 프로그램은 메일 전송 에이전트입니다. Oracle Solaris 운영 체제에서는 다음 메일 사용자 에이전트를 제공합니다.

- /usr/bin/mail
- /usr/bin/mailx

### 메일 전송 에이전트

메일 전송 에이전트는 메일 메시지 경로 지정과 메일 주소 확인을 담당합니다. 이 에이전트는 메일 전송(transport) 에이전트라고도 합니다. Oracle Solaris 운영 체제용 전송 에이전트는 sendmail입니다. 전송 에이전트는 다음과 같은 기능을 합니다.

- 메일 사용자 에이전트에서 보낸 메시지 수락

- 대상 주소 확인
- 메일을 배달하기 위해 적절한 배달 에이전트 선택
- 다른 메일 전송 에이전트로부터 받는 메일 수신

## 로컬 배달 에이전트

로컬 배달 에이전트는 메일 배달 프로토콜을 구현하는 프로그램입니다. Oracle Solaris 운영 체제에서는 다음과 같은 로컬 배달 에이전트가 제공됩니다.

- UUCP 로컬 배달 에이전트 - uux를 사용하여 메일 배달
- 로컬 배달 에이전트 - 표준 Oracle Solaris 릴리스의 mail.local

357 페이지 “sendmail 버전 8.12에서 변경된 사항”에는 다음과 같은 관련 항목이 있습니다.

- 367 페이지 “sendmail 버전 8.12의 추가 배달 에이전트 플래그”
- 367 페이지 “sendmail 버전 8.12에서 배달 에이전트에 대한 등식”

## 메일러 및 sendmail

메일러는 sendmail 관련 용어입니다. 메일러는 사용자 정의된 로컬 배달 에이전트나 사용자 정의된 메일 전송 에이전트의 특정 인스턴스를 식별하기 위해 sendmail에 사용됩니다. sendmail.cf 파일에서 적어도 하나의 메일러를 지정해야 합니다. 작업 정보는 13 장, “메일 서비스(작업)”의 285 페이지 “sendmail 구성 변경”을 참조하십시오. 이 절에서는 두 가지 유형의 메일러를 간단히 설명합니다.

- 320 페이지 “SMTP(Simple Mail Transfer Protocol) 메일러”
- 321 페이지 “UUCP(UNIX-to-UNIX Copy Program) 메일러”

메일러에 대한 자세한 내용은 <http://www.sendmail.org/m4/readme.html> 또는 [/etc/mail/cf/README](#)를 참조하십시오.

## SMTP(Simple Mail Transfer Protocol) 메일러

SMTP는 인터넷에서 사용되는 표준 메일 프로토콜입니다. 이 프로토콜은 다음 메일러를 정의합니다.

- smtp는 정규 SMTP 전송을 다른 서버에 제공합니다.
- esmtp는 확장된 SMTP 전송을 다른 서버에 제공합니다.
- smtp8은 8비트 데이터를 MIME으로 변환하지 않고 다른 서버에 SMTP 전송을 제공합니다.
- dsmtplib는 F= 메일러 플래그를 사용하여 주문형 배달을 제공합니다. 366 페이지 “sendmail 버전 8.12에서 MAILER() 선언의 변경 사항” 및 367 페이지 “sendmail 버전 8.12의 추가 배달 에이전트 플래그”를 참조하십시오.



## UUCP(UNIX-to-UNIX Copy Program) 메일러

가능하면 UUCP를 사용하지 마십시오. 설명은 [http://www.sendmail.org/m4/uucp\\_mailers.html](http://www.sendmail.org/m4/uucp_mailers.html)을 참조하거나 /etc/mail/cf/README에서 문자열 USING UUCP MAILERS를 검색하십시오.

UUCP는 다음 메일러를 정의합니다.

**uucp-old**     \$=U 클래스의 이름이 **uucp-old**로 보내집니다. **uucp**는 이 메일러에 대한 오래된 이름입니다. **uucp-old** 메일러는 헤더에 느낌표 주소를 사용합니다.

**uucp-new**     \$=Y 클래스의 이름이 **uucp-new**로 보내집니다. 이 메일러를 사용하려면 수신 UUCP 메일러가 하나의 전송에서 받는 사람을 여러 명 관리할 수 있다는 점을 이해해야 합니다. **suucp**는 이 메일러에 대한 오래된 이름입니다. **uucp-new** 메일러도 헤더에 느낌표 주소를 사용합니다.

구성에 MAILER(smtp)도 지정되어 있을 경우 메일러 2개가 더 정의됩니다.

**uucp-dom**     이 메일러는 도메인 스타일 주소를 사용하며 기본적으로 SMTP 다시 쓰기 규칙을 적용합니다.

**uucp-uudom**    \$=Z 클래스의 이름이 **uucp-uudom**으로 보내집니다. **uucp-uudom** 및 **uucp-dom**은 같은 헤더 주소 형식인 도메인 스타일 주소를 사용합니다.

---

주 - smtp 메일러가 UUCP 메일러를 수정하므로 .mc 파일에서 항상 MAILER(smtp)를 MAILER(uucp) 앞에 놓으십시오.

---

## 메일 주소

**메일 주소**에는 메일 메시지가 배달되는 받는 사람 및 시스템 이름이 포함됩니다. 이름 서비스를 사용하지 않는 소규모 메일 시스템을 관리할 경우 쉽게 메일 주소를 지정할 수 있습니다. 로그인 이름은 사용자를 고유하게 식별합니다. 우편함이 있는 시스템이 둘 이상 있거나 도메인이 하나 이상 있는 메일 시스템을 관리할 경우 주소 지정이 복잡해지고, 네트워크 밖에 있는 서버와의 UUCP 또는 다른 메일 연결이 있을 경우에는 더욱 복잡해집니다. 다음 절의 내용은 메일 주소의 부분과 복잡성을 이해하는 데 도움이 됩니다.

- 321 페이지 “도메인 및 하위 도메인”
- 322 페이지 “이름 서비스 도메인 이름 및 메일 도메인 이름”
- 322 페이지 “메일 주소의 일반 형식”
- 323 페이지 “경로 독립적 메일 주소”

## 도메인 및 하위 도메인

전자 메일 주소에는 도메인이 사용됩니다. **도메인**은 네트워크 주소 이름 지정을 위한 디렉토리 구조입니다. 도메인 하나에 하나 이상의 **하위 도메인**을 포함할 수 있습니다.

주소의 도메인과 하위 도메인을 파일 시스템의 계층에 비교할 수 있습니다. 하위 디렉토리가 상위 디렉토리 안에 포함된다고 간주되는 것처럼 메일 주소에 포함된 각 하위 도메인도 오른쪽에 있는 상위 도메인 안에 포함된다고 간주됩니다.

다음 표에서는 일부 최상위 도메인을 보여줍니다.

표 14-7 최상위 도메인

도메인	설명
com	상업적 사이트
edu	교육 사이트
gov	미국 정부 기관
mil	미국 군사 기관
net	네트워크 조직
org	기타 비영리 조직

도메인은 대소문자를 구분합니다. 오류 없이 대문자, 소문자 또는 대소문자를 함께 주소의 도메인 부분에 사용할 수 있습니다.

## 이름 서비스도메인 이름 및 메일도메인 이름

이름 서비스도메인 이름과 메일도메인 이름을 사용하여 작업할 때는 다음 사항을 기억하십시오.

- 기본적으로 **sendmail** 프로그램은 NIS도메인 이름에서 첫번째 구성 요소를 제거하여 메일도메인 이름을 구성합니다. 예를 들어, NIS도메인 이름이 **bldg5.example.com**일 경우 메일도메인 이름은 **example.com**입니다.
- 메일도메인 주소는 대소문자를 구분하지 않지만 NIS도메인 이름은 대소문자를 구분합니다. 메일 및 NIS도메인 이름을 설정할 때 소문자를 사용하는 것이 가장 좋습니다.
- DNS도메인 이름과 메일도메인 이름은 같아야 합니다.

자세한 내용은 [345 페이지 “sendmail과 이름 서비스의 상호 작용”](#)을 참조하십시오.

## 메일 주소의 일반 형식

일반적으로 메일 주소 형식은 다음과 같습니다. 자세한 내용은 [323 페이지 “경로 독립적 메일 주소”](#)를 참조하십시오.

*user@subdomain. ... .subdomain2.subdomain1.top-level-domain*

@기호 왼쪽의 주소 부분은 로컬 주소입니다. 로컬 주소는 다음을 포함할 수 있습니다.

- 다른 메일 전송과의 경로 지정에 대한 정보(예: bob::vmsvax@gateway 또는 smallberries%mill.uucp@gateway)
- 별칭(예: iggy.ignatz)

---

주-수신 메일러는 주소의 로컬 부분의 의미를 확인합니다. 메일러에 대한 자세한 내용은 320 페이지 “메일러 및 sendmail”을 참조하십시오.

---

@ 기호 오른쪽의 주소 부분은 로컬 주소가 위치한 도메인 레벨을 보여줍니다. 점은 각 하위 도메인을 분리합니다. 주소의 도메인 부분은 조직, 실제 영역 또는 지역일 수 있습니다. 또한 도메인 정보의 순서는 계층적이므로 하위 도메인이 로컬일수록 @ 기호와 가깝습니다.

## 경로 독립적 메일 주소

메일 주소는 경로 독립적일 수 있습니다. 경로 독립적인 주소를 지정하려면 전자 메일 메시지를 보낸 사람이 받는 사람 이름과 최종 대상을 지정해야 합니다. 인터넷과 같은 고속 네트워크에서는 경로 독립적 주소를 사용합니다. 경로 독립적 주소는 다음과 같은 형식일 수 있습니다.

*user@host.domain*

UUCP 연결을 위한 경로 독립적 주소는 다음과 같은 형식일 수 있습니다.

*host.domain!user*

도메인 계층적 이름 지정 체계가 컴퓨터에 널리 쓰이면서 경로 독립적 주소가 일반화되고 있습니다. 실제로 가장 일반적인 경로 독립적 주소는 호스트 이름을 생략하고 도메인 이름 서비스를 사용하여 전자 메일 메시지의 최종 대상을 적절하게 식별합니다.

*user@domain*

처음에는 @ 기호를 검색하여 경로 독립적 주소를 읽습니다. 그런 다음 오른쪽(최상위 레벨)에서 왼쪽(@ 기호 오른쪽에 있는 가장 구체적인 주소 부분)으로 도메인 계층을 읽습니다.

## 우편함 파일

**우편함**은 전자 메일 메시지 최종 대상인 파일입니다. 우편함 이름은 사용자 이름이거나 포스트마스터와 같은 특정 기능의 ID일 수 있습니다. 우편함은 */var/mail/username* 파일에 있으며 이 파일은 사용자의 로컬 시스템이나 원격 메일 서버에 있을 수 있습니다. 두 경우 모두 메일이 배달되는 시스템에 우편함이 있습니다.

사용자 에이전트가 메일 스푼에서 메일을 가져와 로컬 우편함에 저장할 수 있도록 메일은 항상 로컬 파일 시스템으로 배달되어야 합니다. NFS 마운트된 파일 시스템을 사용자 우편함의 대상으로 사용하지 마십시오. 특히 원격 서버에서 */var/mail* 파일

시스템을 마운트하는 메일 클라이언트로 메일을 전송하지 마십시오. 이 경우 사용자의 메일에 대해 클라이언트 호스트 이름이 아닌 메일 서버로 주소를 지정해야 합니다. NFS 마운트된 파일 시스템을 사용하면 메일 배달 및 처리에 문제가 발생할 수 있습니다.

/etc/mail/aliases 파일과 NIS 등의 이름 서비스는 전자 메일 주소의 별칭을 만들기 위한 방식을 제공합니다. 그러므로 사용자 우편함의 정확한 로컬 이름을 알 필요가 없습니다.

다음 표에서는 특별한 용도의 우편함에 대한 공통된 이름 지정 규약 몇 가지를 보여줍니다.

표 14-8 우편함 이름 형식을 위한 규약

형식	설명
<i>username</i>	사용자 이름이 우편함 이름과 같은 경우가 많습니다.
<i>Firstname.Lastname</i> <i>Firstname_Lastname</i> <i>Firstinitial.Lastname</i> <i>Firstinitial_Lastname</i>	점이나 밑줄로 첫째 이름과 마지막 이름을 분리하는 전체 이름으로 사용자 이름을 식별할 수 있습니다. 또는 점이나 밑줄로 이니셜과 마지막 이름을 분리하는 첫째 이니셜로 사용자 이름을 식별할 수 있습니다.
<i>postmaster</i>	사용자는 질문을 해결하고 메일 시스템의 문제를 <b>postmaster</b> 우편함에 보고할 수 있습니다. 사이트와 도메인마다 <b>postmaster</b> 우편함이 있어야 합니다.
<b>MAILER-DAEMON</b>	<b>sendmail</b> 은 주소가 <b>MAILER-DAEMON</b> 으로 지정된 메일의 경로를 포스트마스터로 자동으로 지정합니다.
<i>aliasname-request</i>	-request로 끝나는 이름은 배포 목록을 위한 관리 주소입니다. 이 주소는 배포 목록을 유지 관리하는 담당자에게 메일을 재지정합니다.
<i>owner-aliasname</i>	owner-로 시작하는 이름은 배포 목록을 위한 관리 주소입니다. 이 주소는 메일 오류를 처리하는 담당자에게 메일을 재지정합니다.
<i>owner-owner</i>	<i>owner-aliasname</i> 별칭이 없을 경우 오류가 반환되지 않도록 이 별칭이 사용됩니다. 이 주소는 메일 오류를 처리하는 담당자에게 메일을 재지정합니다. 여러 별칭을 유지 관리하는 시스템에서도 이 주소를 정의해야 합니다.
<b>로컬 %domain</b>	퍼센트 기호(%)는 메시지가 대상에 도착하면 확장되는 로컬 주소를 표시합니다. 대부분의 메일 시스템은 % 문자가 있는 우편함 이름을 전체 메일 주소로 해석합니다. %는 @으로 대체되고 메일이 알맞게 재지정됩니다. % 규약이 널리 사용되기는 하지만 공식 표준은 아닙니다. 이 규약을 “percent hack”이라고 합니다. 종종 이 기능을 사용하여 메일 문제를 디버깅합니다.

**sendmail** 버전 8부터, 소유자 별칭이 있을 경우 그룹 별칭으로 전송되는 메일의 Envelope 보낸 사람이 소유자 별칭에서 확장된 주소로 변경되었습니다. 이 변경으로 인해 메일 오류를 보낸 사람에게 돌려보내지 않고 별칭 소유자에게 보낼 수 있습니다. 이 변경으로 인해 메일이 배달되면 사용자는 별칭으로 전송된 메일이 별칭 소유자가 보낸 메일이라고 판단합니다. 다음 별칭 형식은 이 변경과 관련된 몇 가지 문제를 해결하는 데 도움이 됩니다.

```
mygroup: :include:/pathname/mygroup.list
owner-mygroup: mygroup-request
mygroup-request: sandys, ignatz
```

이 예에서는 mygroup 별칭이 그룹의 실제 메일 별칭입니다. owner-mygroup 별칭이 오류 메시지를 받습니다. mygroup-request 별칭을 관리 요청에 사용해야 합니다. 이 구조는 mygroup 별칭에게 보낸 메일에서 Envelope 보낸 사람이 mygroup-request로 변경됨을 의미합니다.

## 메일 별칭

별칭은 대체 이름입니다. 전자 메일에 별칭을 사용하여 우편함 위치를 지정하거나 메일링 목록을 정의할 수 있습니다. 작업 맵은 [13 장](#), “메일 서비스(작업)”의 [294 페이지](#) “편지 별칭 파일 관리(작업 맵)”를 참조하십시오. 이 장에서 [339 페이지](#) “메일 별칭 파일”을 참조할 수도 있습니다.

대규모 사이트의 경우 메일 별칭은 일반적으로 우편함 위치를 정의합니다. 메일 별칭을 제공하는 것은 방이 여러 개 있는 큰 회사에서 방 번호를 개인의 주소로 제공하는 것과 같습니다. 방 번호를 제공하지 않으면 우편물이 중앙 주소로 배달됩니다. 방 번호가 없으면 건물 안에서 우편물을 배달할 위치를 확인할 수 없으므로 잘못 배달될 가능성이 있습니다. 예를 들어, 이름이 Kevin Smith인 두 사람이 같은 건물에 있을 경우 둘 중 한 명만 우편물을 받을 수 있습니다. 문제를 해결하려면 각 Kevin Smith의 주소에 방 번호를 추가해야 합니다.

메일링 목록을 만들 때 가급적 많은 도메인과 위치 독립적 주소를 사용하십시오. 별칭 파일의 이식성과 유연성을 개선하려면 메일링 목록의 별칭 항목을 가급적 일반적이고 시스템 독립적으로 만드십시오. 예를 들어, 이름이 ignatz인 사용자가 도메인 example.com의 시스템 mars에 있으면 ignatz@mars 대신 별칭 ignatz@example.com을 만드십시오. 사용자 ignatz가 시스템 이름을 변경하고 example 도메인은 유지할 경우 시스템 이름의 변경 사항을 반영하도록 별칭 파일을 업데이트할 필요가 없습니다.

별칭 항목을 만들 때 한 행에 하나씩 별칭을 입력하십시오. 사용자의 시스템 이름이 포함된 항목은 하나만 있어야 합니다. 예를 들어, 사용자 ignatz에 대해 다음 항목을 만들 수 있습니다.

```
ignatz: iggy.ignatz
iggyi: iggy.ignatz
iggy.ignatz: ignatz@mars
```

로컬 이름이나 도메인에 대해 별칭을 만들 수 있습니다. 예를 들어, 시스템 mars와 도메인 planets에 우편함이 있는 사용자 fred의 별칭 항목은 NIS 별칭 맵에 다음 항목을 포함할 수 있습니다.

```
fred: fred@planets
```

도메인 외부의 사용자를 포함하는 메일 목록을 만들 때는 사용자 이름과 도메인 이름을 사용하여 별칭을 만드십시오. 예를 들어, 이름이 smallberries인 사용자가 도메인 example.com의 시스템 privet에 있으면 별칭을 smallberries@example.com으로

만드십시오. 이제 메일이 사용자의 도메인을 벗어나면 보낸 사람의 전자 메일 주소가 정규화된 도메인 이름으로 자동 변환됩니다.

다음 목록에서는 메일 별칭을 만들고 관리하는 방법에 대해 설명합니다.

- NISaliases 맵이나 로컬 `/etc/mail/aliases` 파일에서 전역적으로 사용할 메일 별칭을 만들 수 있습니다. 같은 별칭 파일을 사용하는 메일링 목록을 만들어 관리할 수도 있습니다.
- 메일 서비스 구성에 따라 NIS 이름 서비스를 사용해 전역 `aliases` 데이터베이스를 유지 관리하여 별칭을 관리할 수 있습니다. 또는 모든 로컬 `/etc/mail/aliases` 파일을 업데이트하여 별칭을 동기화할 수 있습니다.
- 사용자도 별칭을 만들고 사용할 수 있습니다. 해당 사용자만 사용할 수 있는 `~/.mailrc` 파일이나 아무나 사용할 수 있는 로컬 `/etc/mail/aliases` 파일에서 별칭을 만들 수 있습니다. 사용자는 보통 NIS 별칭 파일을 만들거나 관리할 수 없습니다.

## 하드웨어 구성 요소

같은 시스템이나 별도의 시스템에 메일 구성의 세 가지 필수 요소를 제공할 수 있습니다.

- [326 페이지 “메일 호스트”](#)
- [327 페이지 “메일 서버”](#)
- [327 페이지 “메일 클라이언트”](#)

사용자가 도메인 외부의 네트워크와 통신해야 할 경우 네번째 요소인 메일 게이트웨이도 추가해야 합니다. 자세한 내용은 [327 페이지 “메일 게이트웨이”](#)를 참조하십시오. 다음 절에서는 각각의 하드웨어 구성 요소에 대해 설명합니다.

### 메일 호스트

**메일 호스트**는 네트워크의 기본 메일 시스템으로 지정한 시스템입니다. 메일 호스트는 사이트의 다른 시스템이 배달할 수 없는 메일을 전달하는 대상 시스템입니다. 로컬 `/etc/hosts` 파일에서 IP 주소 오른쪽에 단어 `mailhost`를 추가하여 `hosts` 데이터베이스에서 시스템을 메일 호스트로 지정합니다. 또는 이름 서비스의 호스트 파일과 비슷하게 단어 `mailhost`를 추가할 수 있습니다. 자세한 작업 정보는 [13 장, “메일 서비스\(작업\)”](#)의 [281 페이지 “메일 호스트 설정 방법”](#)을 참조하십시오.

네트워크에서 인터넷 전역 네트워크로 향하는 라우터로 구성된 시스템을 메일 호스트로 사용하는 것이 좋습니다. 자세한 내용은 [15 장, “Solaris PPP 4.0\(개요\)”](#), [24 장, “UUCP\(개요\)”](#) 및 **Oracle Solaris 관리: IP 서비스의 “IPv4 라우터 구성”**을 참조하십시오. 로컬 네트워크의 시스템에 모뎀이 없을 경우 시스템을 메일 호스트로 지정하십시오.

일부 사이트에서는 시간 공유 구성에서 네트워크로 연결되지 않은 독립형 시스템을 사용합니다. 특히 독립형 시스템은 직렬 포트에 연결된 단말기에 적합합니다. 독립형 시스템을 단일 시스템 네트워크의 메일 호스트로 지정하여 이 구성에 대한 전자 메일을 설정할 수 있습니다. [12 장, “메일 서비스\(개요\)”](#)의 [270 페이지 “하드웨어 구성 요소 개요”](#)에는 일반적인 전자 메일 구성을 보여주는 그림이 있습니다.

## 메일 서버

**우편함**은 특정 사용자의 전자 메일을 포함하는 단일 파일입니다. 사용자의 우편함이 있는 시스템으로 메일이 배달됩니다. 이 우편함은 로컬 시스템이나 원격 서버에 있을 수 있습니다. **메일 서버**는 `/var/mail` 디렉토리에 사용자 우편함을 유지 관리하는 시스템입니다. 작업 정보는 13 장, “메일 서비스(작업)”의 278 페이지 “메일 서버 설정 방법”을 참조하십시오.

메일 서버는 클라이언트에서 오는 모든 메일의 경로를 지정합니다. 클라이언트가 메일을 보내면 메일 서버가 배달을 위해 대기열에 메일을 넣습니다. 메일이 대기열에 들어간 후 사용자는 해당 메일 메시지를 유지하면서 클라이언트를 재부트하거나 전원을 끌 수 있습니다. 받는 사람이 클라이언트에서 메일을 받을 때 메시지의 **From** 행의 경로에 메일 서버 이름이 포함되어 있습니다. 받는 사람이 응답할 경우 사용자의 우편함으로 응답이 전송됩니다. 정기적으로 백업되는 사용자나 시스템에 대해 홈 디렉토리를 제공하는 시스템을 메일 서버로 사용하는 것이 좋습니다.

메일 서버가 사용자의 로컬 시스템이 아닐 경우 NFS 소프트웨어를 사용하는 구성의 사용자는 **root** 액세스 권한이 있을 경우 `/etc/vfstab` 파일을 사용하여 `/var/mail` 디렉토리를 마운트할 수 있습니다. 기타 방법으로, 사용자는 자동 마운트를 사용할 수도 있습니다. NFS 지원을 사용할 수 없는 경우 사용자는 서버에 로그인하여 메일을 읽을 수 있습니다.

네트워크의 사용자가 오디오 파일이나 데스크탑 게시 시스템의 파일과 같이 다른 유형의 메일을 보낼 경우 우편함을 위해 메일 서버에 공간을 더 할당해야 합니다.

모든 우편함에 대한 메일 서버를 설정하여 백업 프로세스를 단순화할 수 있습니다. 메일이 여러 시스템에 퍼져 있으면 백업이 어려울 수 있습니다. 여러 우편함을 서버 하나에 저장할 경우 여러 사용자의 오류가 단일 지점에서 발생할 수 있다는 단점이 있습니다. 그러나 제대로 백업을 수행하면 위험에 비해 큰 장점을 얻을 수 있습니다.

## 메일 클라이언트

메일 클라이언트는 메일 서버에 우편함이 있는 메일 서비스 사용자입니다. 또한 메일 클라이언트는 `/etc/mail/aliases` 파일에 우편함 위치를 가리키는 메일 별칭이 있습니다. 작업 정보는 13 장, “메일 서비스(작업)”의 279 페이지 “메일 클라이언트 설정 방법”을 참조하십시오.

## 메일 게이트웨이

**메일 게이트웨이**는 여러 통신 프로토콜을 실행하는 네트워크 사이의 연결 또는 같은 프로토콜을 사용하는 여러 네트워크 사이의 통신을 처리하는 시스템입니다. 예를 들어, 메일 게이트웨이가 SNA(Systems Network Architecture) 프로토콜 제품군을 실행하는 네트워크에 TCP/IP 네트워크를 연결할 수 있습니다.

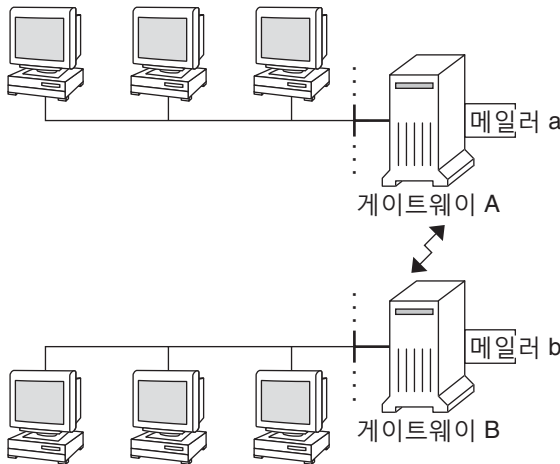
가장 단순하게 설정할 수 있는 메일 게이트웨이는 같은 프로토콜이나 메일러를 사용하는 네트워크 두 개를 연결하는 메일 게이트웨이입니다. 이 시스템은 **sendmail**이



해당 주소로 도메인에서 받는 사람을 찾을 수 없는 메일을 처리합니다. 메일 게이트웨이가 있을 경우 `sendmail`은 게이트웨이를 사용하여 도메인 외부로 메일을 보내고 받습니다.

다음 그림에 표시된 대로 서로 다른 메일러를 사용하는 두 네트워크 간에 메일 게이트웨이를 설정할 수 있습니다. 이 구성을 지원하려면 메일 게이트웨이 시스템에서 `sendmail.cf` 파일을 사용자 정의해야 합니다. 이 프로세스는 어렵고 시간이 오래 걸릴 수 있습니다.

그림 14-1 여러 통신 프로토콜 사이의 게이트웨이



인터넷 연결을 제공하는 시스템이 있을 경우 이 시스템을 메일 게이트웨이로 구성할 수 있습니다. 메일 게이트웨이를 구성하기 전에 사이트의 보안 요구 사항을 신중히 고려하십시오. 회사 네트워크와 다른 네트워크 사이에 방화벽 게이트웨이를 만들고 해당 게이트웨이를 메일 게이트웨이로 설정해야 합니다. 작업 정보는 [13 장, “메일 서비스\(작업\)”](#)의 [283 페이지 “메일 게이트웨이 설정 방법”](#)을 참조하십시오.

## 메일 서비스 프로그램 및 파일

메일 서비스에는 서로 상호 작용하는 여러 프로그램과 데몬이 포함됩니다. 이 절에서는 전자 메일 관리에 관련된 파일, 프로그램, 용어 및 개념을 소개합니다.

- 329 페이지 “vacation 유틸리티의 향상된 기능”
- 329 페이지 “/usr/bin 디렉토리의 내용”
- 330 페이지 “/etc/mail 디렉토리의 내용”
- 333 페이지 “/usr/lib 디렉토리의 내용”
- 334 페이지 “메일 서비스에 사용되는 기타 파일”
- 335 페이지 “메일 프로그램의 상호 작용”



- 335 페이지 “sendmail 프로그램”
- 339 페이지 “메일 별칭 파일”
- 342 페이지 “.forward 파일”
- 343 페이지 “/etc/default/sendmail 파일”

## vacation 유틸리티의 향상된 기능

vacation 유틸리티의 기능이 향상되어 사용자가 자동 생성된 회신을 수신하는 받는 메시지를 지정할 수 있습니다. 이 향상된 기능으로 인해 사용자는 기밀 또는 연락처 정보를 알 수 없는 사용자와 공유하는 문제를 방지할 수 있습니다. 스팸머나 알 수 없는 사용자가 보낸 메시지에는 회신이 전송되지 않습니다.

이러한 향상된 기능은 수신되는 송신자의 전자우편 주소를 .vacation.filter 파일의 도메인 또는 전자우편 주소 목록과 비교하여 작동합니다. 이 파일은 사용자가 만들며 사용자의 홈 디렉토리에 있습니다. 도메인이나 전자 메일 주소 일치 항목이 있으면 회신이 전송됩니다. 일치가 검색되지 않는 경우 응답이 송신되지 않습니다.

.vacation.filter는 다음과 같은 항목을 포함할 수 있습니다.

```
company.com
mydomain.com
onefriend@hisisp.com
anotherfriend@herisp.com
```

행마다 도메인이나 전자 메일 주소가 하나씩 있습니다. 각 항목은 별도의 행에 있어야 합니다. 보낸 사람의 전자 메일 주소가 전자 메일 주소 항목과 일치하기 위해서는 대소문자를 제외하고 정확히 일치해야 합니다. 보낸 사람 주소의 문자가 소문자인지 대문자인지는 무시됩니다. 보낸 사람의 전자 메일 주소가 도메인 항목과 일치하기 위해서는 나열된 도메인이 보낸 사람 주소에 포함되어야 합니다. 예를 들어, somebody@dept.company.com과 someone@company.com 모두 company.com의 도메인 항목에 대한 일치 항목이 됩니다.

자세한 내용은 [vacation\(1\)](#) 매뉴얼 페이지를 참조하십시오.

## /usr/bin 디렉토리의 내용

다음 표에서는 메일 서비스에 사용되는 /usr/bin 디렉토리의 내용을 보여줍니다.

이름	유형	설명
mail	파일	사용자 에이전트
mailcompat	파일	SunOS 4.1 우편함 형식으로 메일을 저장하기 위한 필터
mailq	파일	메일 대기열 콘텐츠를 나열하는 프로그램

이름	유형	설명
mailstats	파일	/etc/mail/statistics 파일(있을 경우)에 저장되는 메일 통계를 읽기 위해 사용되는 프로그램
mailx	파일	사용자 에이전트
mconnect	파일	주소 확인 및 디버깅을 위해 메일러에 연결하는 프로그램
praliases	파일	별칭 데이터베이스를 “컴파일 취소”하기 위한 명령 <a href="#">praliases(1)</a> 의 매뉴얼 페이지에 제공되는 컴파일 취소 정보를 참조하십시오.
rmail	심볼릭 링크	/usr/bin/mail에 대한 심볼릭 링크. 메일 보내기만 허용하기 위해 종종 사용되는 명령.
vacation	파일	메일에 대한 자동 회신을 설정하기 위한 명령

## /etc/mail 디렉토리의 내용

다음 표에서는 /etc/mail 디렉토리의 내용을 보여줍니다.

이름	유형	설명
Mail.rc	파일	mailx 사용자 에이전트의 기본 설정
aliases	파일	메일 전달 정보
aliases.db	파일	newaliases를 실행하여 만든 메일 전달 정보의 기본 이진 양식
aliases.dir	파일	newaliases를 실행하여 만든 메일 전달 정보의 이진 양식. 여전히 사용할 수 있지만 Solaris 9 릴리스부터 더 이상 기본적으로 사용되지 않습니다.
aliases.pag	파일	newaliases를 실행하여 만든 메일 전달 정보의 이진 양식. 여전히 사용할 수 있지만 Solaris 9 릴리스부터 더 이상 기본적으로 사용되지 않습니다.
mailx.rc	파일	mailx 사용자 에이전트의 기본 설정
main.cf	심볼릭 링크	역방향 호환성을 위해 주 시스템에 대한 이 샘플 구성 파일에서 sendmail.cf로 연결되는 심볼릭 링크가 제공됩니다. 버전 8.13의 sendmail에는 이 파일이 필요 없습니다.
relay-domains	파일	중계가 허용된 모든 도메인 목록. 기본적으로 로컬 도메인만 허용됩니다.
sendmail.cf	파일	메일 경로 지정을 위한 구성 파일

이름	유형	설명
submit.cf	파일	MSP(Mail Submission Program)를 위한 새 구성 파일 자세한 내용은 358 페이지 “ <a href="#">sendmail 버전 8.12의 submit.cf 구성 파일</a> ”을 참조하십시오.
local-host-names	파일	메일 호스트에 대한 별칭 개수가 너무 많을 경우 만들 수 있는 선택적 파일
helpfile	파일	SMTP HELP 명령에 사용되는 도움말 파일
sendmail.pid	파일	수신 데몬의 PID를 나열하고 현재 /system/volatile에 있는 파일
statistics	파일	sendmail 통계 파일. 이 파일이 있으면 sendmail이 각 메일러를 통과하는 트래픽의 양을 기록합니다. 전에는 이 파일을 sendmail.st라고 했습니다.
subsidiary.cf	심볼릭 링크	역방향 호환성을 위해 보조 시스템에 대한 이 샘플 구성 파일에서 sendmail.cf로 연결되는 심볼릭 링크가 제공됩니다. 버전 8.13의 sendmail에는 이 파일이 필요 없습니다.
trusted-users	파일	특정 메일 작업을 수행할 수 있도록 인증된 사용자를 나열(한 행에 한 명)하는 파일. 기본적으로 이 파일에는 root만 있습니다. 신뢰할 수 없는 사용자가 특정 메일 작업을 수행할 경우 X-Authentication-Warning: header being added to a message라는 경고가 표시됩니다.

## /etc/mail/cf 디렉토리의 내용

/etc/mail 디렉토리에는 하위 디렉토리인 cf가 있습니다. sendmail.cf 파일을 작성하는데 필요한 모든 파일이 여기에 포함됩니다. cf의 내용은 표 14-9에 나와 있습니다.

읽기 전용 /usr 파일 시스템을 지원하기 위해 /usr/lib/mail 디렉토리의 내용이 /etc/mail/cf 디렉토리로 이동했습니다. 그러나 다음의 예외에 유의해야 합니다. /usr/lib/mail/sh/check-hostname 및 /usr/lib/mail/sh/check-permissions 쉘 스크립트는 이제 /usr/sbin 디렉토리에 있습니다. 334 페이지 “[메일 서비스에 사용되는 기타 파일](#)”을 참조하십시오. 역방향 호환성을 위하여 심볼 링크가 각 파일의 새 위치를 가리킵니다.

표 14-9 메일 서비스에 사용되는 /etc/mail/cf 디렉토리의 내용

이름	유형	설명
README	파일	구성 파일에 대해 설명합니다.
cf/main.cf	심볼릭 링크	이 파일 이름은 cf/sendmail.cf에 연결됩니다. 이 파일은 주 구성 파일이었습니다.

표 14-9 메일 서비스에 사용되는 /etc/mail/cf 디렉토리의 내용 (계속)

이름	유형	설명
cf/main.mc	심볼릭 링크	이 파일 이름은 cf/sendmail.mc에 연결됩니다. 이 파일은 주 구성 파일을 만들기 위해 사용된 파일입니다.
cf/Makefile	파일	새 구성 파일을 작성하기 위한 규칙을 제공합니다.
cf/submit.cf	파일	메시지를 제출하는 데 사용되는 MSP(Mail Submission Program)의 구성 파일입니다.
cf/submit.mc	파일	submit.cf 파일을 작성하는 데 사용되는 파일입니다. 이 파일은 MSP(Mail Submission Program)의 m4 매크로를 정의합니다.
cf/sendmail.cf	파일	sendmail의 주 구성 파일입니다.
cf/sendmail.mc	파일	sendmail.cf 파일을 생성하는 데 사용되는 m4 매크로를 포함합니다.
cf/subsidiary.cf	심볼릭 링크	이 파일 이름은 cf/sendmail.cf에 연결됩니다. 이 파일은 다른 호스트에서 /var/mail을 NFS 마운트하는 호스트의 구성 파일이었습니다.
cf/subsidiary.mc	심볼릭 링크	이 파일 이름은 cf/sendmail.mc에 연결됩니다. 이 파일에는 subsidiary.cf 파일을 생성하는 데 사용된 m4 매크로가 포함되어 있었습니다.
domain	디렉토리	사이트에 종속되는 하위 도메인 설명을 제공합니다.
domain/generic.m4	파일	Berkeley 소프트웨어 배포의 일반 도메인 파일입니다.
domain/solaris-antispam.m4	파일	sendmail을 이전 버전의 sendmail처럼 작동하게 하는 변경 사항이 있는 도메인 파일입니다. 그러나 중계는 완전히 사용 안함으로 설정되고, 호스트 이름이 없는 보낸 사람 주소는 거부되며, 확인할 수 없는 도메인은 거부됩니다.
domain/solaris-generic.m4	파일	sendmail을 이전 버전의 sendmail처럼 작동하게 하는 변경 사항이 있는 기본 도메인 파일입니다.
feature	디렉토리	특정 호스트에 대한 특정 기능 정의를 포함합니다. 기능에 대한 자세한 설명은 README를 참조하십시오.

표 14-9 메일 서비스에 사용되는 /etc/mail/cf 디렉토리의 내용 (계속)

이름	유형	설명
m4	디렉토리	사이트 독립적 포함 파일을 포함합니다.
mailer	디렉토리	local, smtp 및 uucp를 비롯한 메일러의 정의를 포함합니다.
main-v7sun.mc	파일	오래된 파일 이름이며 cf/sendmail.mc로 이름이 바뀌었습니다.
ostype	디렉토리	다양한 운영 체제 환경에 대해 설명합니다.
ostype/solaris2.m4	파일	기본 로컬 메일러를 mail.local로 정의합니다.
ostype/solaris2.ml.m4	파일	기본 로컬 메일러를 mail.local로 정의합니다.
ostype/solaris2.pre5.m4	파일	로컬 메일러를 mail로 정의합니다.
ostype/solaris8.m4	파일	로컬 메일러를 mail.local로 정의하고(LMTP 모드), IPv6을 사용으로 설정하며, /system/volatile을 sendmail.pid 파일의 디렉토리로 지정합니다.
subsidiary-v7sun.mc	파일	오래된 파일 이름이며 cf/sendmail.mc로 이름이 바뀌었습니다.

## /usr/lib 디렉토리의 내용

다음 표에서는 메일 서비스에 사용되는 /usr/lib 디렉토리의 내용을 보여줍니다.

표 14-10 /usr/lib 디렉토리의 내용

이름	유형	설명
mail.local	파일	우편함에 메일을 배달하는 메일러
sendmail	파일	메일 전송 에이전트라고도 하는 경로 지정 프로그램
smrsh	파일	sendmail의 “[program]” 구문을 사용하여 sendmail이 실행할 수 있는 프로그램을 /var/adm/sm.bin 디렉토리에 있는 프로그램으로 제한하는 쉘 프로그램(sendmail 제한 쉘)입니다. /var/adm/sm.bin에 포함할 수 있는 내용은 smrsh(1M) 매뉴얼 페이지를 참조하십시오. 사용으로 설정하려면 m4 명령인 FEATURE(‘smrsh’)를 mc 파일에 포함하십시오.
mail	심볼릭 링크	심볼릭 링크는 /etc/mail/cf 디렉토리를 가리킵니다. 자세한 내용은 331 페이지 “/etc/mail/cf 디렉토리의 내용”을 참조하십시오.

## 메일 서비스에 사용되는 기타 파일

표 14-11에 표시된 대로 기타 여러 파일과 디렉토리가 메일 서비스에 사용됩니다.

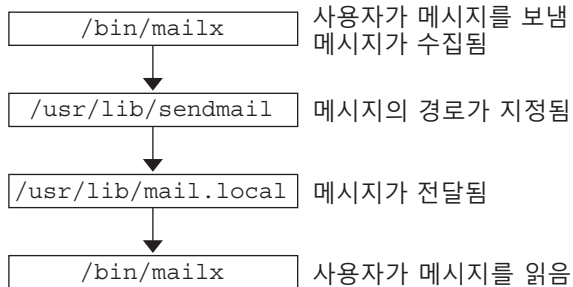
표 14-11 메일 서비스에 사용되는 기타 파일

이름	유형	설명
/etc/default/sendmail	파일	sendmail의 시작 스크립트용 환경 변수를 나열합니다.
/etc/shells	파일	유효한 로그인 셸을 나열합니다.
/etc/mail/cf/sh	디렉토리	m4 구축 프로세스와 마이그레이션 도움말에 사용되는 셸 스크립트를 포함합니다.
/system/volatile/sendmail.pid	파일	수신 데몬의 PID를 나열하는 파일
/usr/sbin/check-permissions	파일	:include: 별칭 및 .forward 파일의 권한 설정을 확인하고 부모 디렉토리 경로의 권한 설정이 올바른지 확인합니다.
/usr/sbin/check-hostname	파일	sendmail이 정규화된 호스트 이름을 확인할 수 있는지 검증합니다.
/usr/sbin/editmap	파일	sendmail용 데이터베이스 맵에서 단일 레코드를 질의하고 편집합니다.
/usr/sbin/in.comsat	파일	메일 통지 데몬
/usr/sbin/makemap	파일	키 맵의 이진 형식을 구축합니다.
/usr/sbin/newaliases	심볼릭 링크	/usr/lib/sendmail에 대한 심볼릭 링크. 별칭 데이터베이스의 이진 형식을 만드는 데 사용됩니다. 이전 위치는 /usr/bin입니다.
/usr/sbin/syslogd	파일	sendmail에 사용되는 오류 메시지 로거
/usr/sbin/etrn	파일	클라이언트측 원격 메일 대기열을 시작하기 위한 Perl 스크립트
/var/mail/mailbox1, /var/mail/mailbox2	파일	배달된 메일을 위한 우편함.
/var/spool/clientmqueue	디렉토리	클라이언트 데몬이 배달한 메일의 저장소
/var/spool/mqueue	디렉토리	마스터 데몬이 배달한 메일의 저장소

## 메일 프로그램의 상호 작용

다음 프로그램의 조합으로 메일 서비스가 제공됩니다. 이들 프로그램은 [그림 14-2](#)와 같이 상호 작용합니다.

그림 14-2 메일 프로그램의 상호 작용



다음은 메일 프로그램의 상호 작용에 대한 설명입니다.

1. 사용자가 `mailx`와 같은 프로그램을 사용하여 메시지를 보냅니다. 자세한 내용은 [mailx\(1\)](#)의 매뉴얼 페이지를 참조하십시오.
2. 메시지를 생성한 프로그램이 메시지를 수집하고 `sendmail` 데몬에 메시지가 전달됩니다.
3. `sendmail` 데몬이 메시지의 주소를 **구문 분석**하며 식별할 수 있는 세그먼트로 나눕니다. 데몬은 구성 파일 `/etc/mail/sendmail.cf`의 정보를 사용하여 네트워크 이름 구문, 별칭, 전달 정보 및 네트워크 토폴로지를 확인합니다. `sendmail`은 이 정보를 사용하여 받는 사람에게 도달하기 위해 메시지가 전송될 경로를 결정합니다.
4. `sendmail` 데몬은 적절한 시스템에 메시지를 전달합니다.
5. 로컬 시스템의 `/usr/lib/mail.local` 프로그램이 메시지 받는 사람의 `/var/mail/username` 디렉토리에 메일을 전달합니다.
6. 받는 사람은 `mail`, `mailx` 또는 유사 프로그램을 사용하여 메일이 도착했다는 통지를 받고 메일을 검색합니다.

## sendmail 프로그램

다음 목록에서는 `sendmail` 프로그램의 몇 가지 기능에 대해 설명합니다.

- `sendmail`은 TCP/IP 및 UUCP와 같은 여러 유형의 통신 프로토콜을 사용할 수 있습니다.
- `sendmail`은 SMTP 서버, 메시지 대기열 및 메일링 목록을 구현합니다.

- **sendmail**은 다음 이름 지정 규약과 함께 사용할 수 있는 패턴 일치 시스템을 사용하여 이름 해석을 제어합니다.
  - 도메인 기반 이름 지정 규약. 도메인 기술은 물리적 이름 지정을 논리적 이름 지정과 구분합니다. 도메인에 대한 자세한 내용은 [321 페이지 “메일 주소”](#)를 참조하십시오.
  - 다른 네트워크에 있는 호스트에 로컬로 표시되는 네트워크 이름을 제공하는 등의 임시 기술
  - 임의의(이전) 이름 지정 구문
  - 서로 다른 이름 지정 체계

Oracle Solaris 운영 체제에서는 **sendmail** 프로그램을 메일 라우터로 사용합니다. 다음 목록에서는 그 기능 중 몇 가지에 대해 설명합니다.

- **sendmail**은 전자 메일 메시지를 받아 **mail.local** 또는 **procmail**과 같은 로컬 배달 에이전트로 배달합니다.
- **sendmail**은 **mailx** 및 **Mozilla Mail**과 같은 사용자 에이전트에서 메시지를 받고 인터넷을 통해 대상으로 메시지의 경로를 지정하는 메일 전송 에이전트입니다.
- **sendmail**은 다음과 같은 방법으로 사용자가 보내는 전자 메일 메시지를 제어합니다.
  - 받는 사람의 주소 평가
  - 알맞은 배달 프로그램 선택
  - 배달 에이전트가 처리할 수 있는 형식으로 주소 다시 쓰기
  - 필요에 따라 메일 헤더 형식 다시 지정
  - 배달을 위해 전송된 메시지를 메일 프로그램에 최종 전달

**sendmail** 프로그램에 대한 자세한 내용은 다음 항목을 참조하십시오.

- [336 페이지 “sendmail 및 경로 재지정 방식”](#)
- [338 페이지 “sendmail의 기능”](#)
- [338 페이지 “sendmail 구성 파일”](#)

## sendmail 및 경로 재지정 방식

**sendmail** 프로그램은 메일 경로 재지정을 위한 세 가지 방식을 지원합니다. 관련된 변경 사항 유형에 따라 적합한 방식을 선택해야 합니다.

- 서버 변경 사항
- 도메인 차원의 변경 사항
- 사용자 한 명의 변경 사항

또한 선택한 경로 재지정 방식은 필요한 관리 레벨에 영향을 줄 수 있습니다. 다음 옵션을 고려해 보십시오.

### 1. 경로 지정 방식 중 한 가지는 **별칭**입니다.

사용된 파일 유형에 따라 별칭은 서버 또는 이름 서비스 차원에서 주소에 이름을 매핑할 수 있습니다.



이름 서비스 별칭에는 다음과 같은 장단점이 있습니다.

- 이름 서비스 별칭 파일을 사용하면 단일 소스에서 메일 경로 지정 변경을 관리할 수 있습니다. 그러나 다시 경로 지정 변경이 전파되면 이름 서비스 별칭으로 인해 지체 시간이 생길 수 있습니다.
- 이름 서비스 관리는 대개 선택된 시스템 관리자 그룹으로 제한됩니다. 일반 사용자는 이 파일을 관리할 수 없습니다.

서버 별칭 파일을 사용하면 다음과 같은 장단점이 있습니다.

- 서버 별칭 파일을 사용하면 지정된 서버에서 root가 될 수 있는 사용자가 경로 재지정을 관리할 수 있습니다.
- 경로 재지정 변경이 전파될 경우 서버 별칭으로 인해 지체 시간이 거의 또는 전혀 생기지 않아야 합니다.
- 변경 사항은 로컬 서버에만 영향을 미치며, 대부분의 메일이 한 서버로 전송되는 경우 수락할 수 있습니다. 그러나 이 변경 사항을 여러 메일 서버로 전파해야 할 경우 이름 서비스를 사용하십시오.
- 일반 사용자는 이 변경 사항을 관리하지 않습니다.

자세한 내용은 이 장에서 [339 페이지](#) “[메일 별칭 파일](#)”을 참조하십시오. 작업 맵은 [13 장](#), “[메일 서비스\(작업\)](#)”의 [294 페이지](#) “[편지 별칭 파일 관리\(작업 맵\)](#)”를 참조하십시오.

## 2. 다음 방식은 전달입니다.

이 방식을 사용하면 사용자가 메일 경로 재지정을 관리할 수 있습니다. 로컬 사용자가 받는 메일을 다음 대상에게 경로를 다시 지정할 수 있습니다.

- 다른 우편함
- 다른 메일러
- 다른 메일 호스트

이 방식은 `.forward` 파일을 사용할 때 지원됩니다. 이 파일에 대한 자세한 내용은 이 장에서 [342 페이지](#) “[.forward 파일](#)”을 참조하십시오. 작업 맵은 [13 장](#), “[메일 서비스\(작업\)](#)”의 [304 페이지](#) “[.forward 파일 관리\(작업 맵\)](#)”를 참조하십시오.

## 3. 마지막 경로 재지정 방식은 포함입니다.

이 방식을 사용하면 사용자가 root 액세스 권한 없이 별칭 목록을 유지 관리할 수 있습니다. 이 기능을 제공하려면 root 사용자가 서버에서 별칭 파일에 해당 항목을 만들어야 합니다. 이 항목이 생성된 후 사용자는 필요할 경우 메일의 경로를 다시 지정할 수 있습니다. 포함에 대한 자세한 내용은 이 장에서 [340 페이지](#) “[/etc/mail/aliases 파일](#)”을 참조하십시오. 작업 맵은 [13 장](#), “[메일 서비스\(작업\)](#)”의 [294 페이지](#) “[편지 별칭 파일 관리\(작업 맵\)](#)”를 참조하십시오.

주 - /usr/bin/mailx와 같이 메일을 읽는 프로그램은 메시지가 sendmail에 도착하기 전에 확장되는 고유한 별칭을 가질 수 있습니다. sendmail에 대한 별칭은 로컬 파일이나 NIS와 같은 다양한 이름 서비스 소스에서 생겨날 수 있습니다. 조회 순서는 svc:/system/name-service/switch 서비스로 결정됩니다. [nsswitch.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오.

## sendmail의 기능

sendmail 프로그램은 다음과 같은 기능을 제공합니다.

- sendmail은 안정적입니다. 모든 메시지를 정확하게 배달할 수 있도록 설계된 프로그램입니다. 메시지가 완전히 사라지는 일은 일어나지 않습니다.
- sendmail은 가능할 경우 항상 기존의 소프트웨어를 배달에 사용합니다. 예를 들어, 사용자가 메일 생성 및 메일 전송 프로그램과 상호 작용합니다. 메일이 제출되면 메일 생성 프로그램이 sendmail을 호출하여 메시지의 경로를 올바른 메일러로 지정합니다. 보낸 사람이 네트워크 서버이고 메일러가 네트워크 클라이언트인 경우가 있으므로 sendmail을 인터넷 메일 게이트웨이로 사용할 수 있습니다. 프로세스에 대한 자세한 내용은 [335 페이지 “메일 프로그램의 상호 작용”](#)을 참조하십시오.
- 여러 네트워크를 비롯하여 복잡한 환경을 처리하도록 sendmail을 구성할 수 있습니다. sendmail은 주소 구분과 주소 내용을 확인하여 사용할 메일러를 결정합니다.
- sendmail은 구성 정보를 코드로 컴파일할 필요 없이 구성 파일을 사용하여 메일 구성을 제어합니다.
- 사용자는 자신의 메일링 목록을 유지 관리할 수 있습니다. 또한 개별 사용자는 대개 NIS가 유지 관리하는 도메인 차원 별칭에 있는 도메인 차원 별칭 파일을 수정하지 않고 고유한 전달 방식을 지정할 수 있습니다.
- 각 사용자는 사용자 정의 메일러가 받는 메일을 처리하도록 지정할 수 있습니다. 사용자 정의 메일러는 “I am on vacation.”이라는 메시지를 반환하는 등의 기능을 제공할 수 있습니다. 자세한 내용은 [vacation\(1\)](#) 매뉴얼 페이지를 참조하십시오.
- sendmail은 주소를 단일 호스트로 일괄 처리하여 네트워크 트래픽을 줄입니다.

## sendmail 구성 파일

구성 파일은 sendmail이 기능을 수행하는 방식을 제어합니다. 구성 파일은 배달 에이전트, 주소 다시 쓰기 규칙 및 메일 헤더 형식의 선택 사항을 결정합니다. sendmail 프로그램은 /etc/mail/sendmail.cf 파일의 정보를 사용하여 기능을 수행합니다.

Oracle Solaris 운영 체제는 /etc/mail 디렉토리에 기본 구성 파일을 제공합니다.

1. sendmail.cf - 데몬 모드에서 sendmail을 실행하는 데 사용되는 구성 파일입니다.

2. `submit.cf` - 데몬 모드가 아니라 Mail Submission Program 모드에서 `sendmail`을 실행하는 데 사용되는 구성 파일입니다. 자세한 내용은 358 페이지 “`sendmail` 버전 8.12의 `submit.cf` 구성 파일”을 참조하십시오.

메일 클라이언트, 메일 서버, 메일 호스트 또는 메일 게이트웨이를 설정할 때 다음 사항을 고려하십시오.

- 메일 클라이언트나 메일 서버의 경우 기본 구성 파일을 설정하거나 편집하기 위해 아무 것도 수행할 필요가 없습니다.
- 메일 호스트나 메일 게이트웨이를 설정하려면 메일 구성에 필요한 중계 메일러 및 중계 호스트 매개변수를 설정해야 합니다. 작업 정보는 13 장, “메일 서비스(작업)”의 277 페이지 “메일 서비스 설정(작업 맵)” 또는 285 페이지 “`sendmail` 구성 변경”을 참조하십시오. `sendmail` 버전 8.13을 사용할 경우 더 이상 `main.cf` 파일이 필요 없습니다.

다음 목록에서는 사이트 요구 사항에 따라 변경할 수 있는 몇 가지 구성 매개변수에 대해 설명합니다.

- 시간 값 - 다음 정보를 지정합니다.
  - 읽기 시간 초과
    - 메시지가 보낸 사람에게 돌아가기 전에 대기열에서 배달되지 않은 채 남아있는 시간. 368 페이지 “`sendmail` 버전 8.12의 추가 대기열 기능”을 참조하십시오. 작업 맵은 300 페이지 “대기열 디렉토리 관리(작업 맵)”를 참조하십시오.
- 배달 모드 - 메일이 배달되기까지의 시간을 지정합니다.
- 로드 한계 - 사용량이 많을 때 효율성을 높입니다. 이러한 매개변수를 사용하면 `sendmail`이 대용량 메시지, 받는 사람이 여러 명인 메시지 및 오랫동안 작동이 중지된 사이트로 보내는 메시지를 배달하지 않습니다.
- 로그 레벨 - 기록되는 문제의 종류를 지정합니다.

## 메일 별칭 파일

다음과 같은 파일, 맵 또는 테이블 중 하나를 사용하여 별칭을 유지 관리합니다.

- 340 페이지 “.mailrc 별칭”
- 340 페이지 “/etc/mail/aliases 파일”
- 341 페이지 “NIS aliases 맵”

별칭 사용자 및 별칭을 변경할 수 있는 사용자에게 따라 별칭을 유지 관리하는 방법이 결정됩니다. 별칭 유형마다 고유한 형식 요구 사항이 있습니다.

작업 정보는 13 장, “메일 서비스(작업)”의 294 페이지 “편지 별칭 파일 관리(작업 맵)”를 참조하십시오.

## .mailrc 별칭

.mailrc 파일에 나열된 별칭에는 파일 소유자만 액세스할 수 있습니다. 이 제한으로 인해 사용자는 자신이 제어하고 해당 소유자만 사용할 수 있는 별칭 파일을 만들 수 있습니다. .mailrc 파일에 있는 별칭은 다음 형식을 따릅니다.

```
alias aliasname value value ...
```

*aliasname*은 메일을 보낼 때 사용자가 사용하는 이름이며 *value*는 유효한 전자 메일 주소입니다.

사용자가 이름 서비스의 **scott**에 대한 전자 메일 주소와 일치하지 않는 전자 메일 주소를 **scott**에 대해 설정할 경우 오류가 발생합니다. 이 사용자가 생성한 메일에 회신할 경우 잘못된 상대에게 메일의 경로가 지정됩니다. 유일한 해결 방법은 다른 별칭 설정 방식을 사용하는 것입니다.

## /etc/mail/aliases 파일

/etc/mail/aliases 파일에 설정된 별칭은 별칭의 이름과 파일이 포함된 시스템의 호스트 이름을 아는 사용자만 사용할 수 있습니다. 로컬 /etc/mail/aliases 파일의 배포 목록 형식은 다음과 같습니다.

```
aliasname: value,value,value ...
```

*aliasname*은 이 별칭으로 메일을 보낼 때 사용자가 사용하는 이름이며 *value*는 유효한 전자 메일 주소입니다.

네트워크에서 이름 서비스를 실행하지 않는 경우 각 시스템의 /etc/mail/aliases 파일에 모든 메일 클라이언트에 대한 항목이 있어야 합니다. 각 시스템에서 파일을 편집하거나 시스템 하나에서 파일을 편집하고 다른 시스템에 각각 파일을 복사할 수 있습니다.

/etc/mail/aliases 파일의 별칭은 텍스트 형식으로 저장됩니다. /etc/mail/aliases 파일을 편집할 때 **newaliases** 프로그램을 실행해야 합니다. 이 프로그램은 데이터베이스를 다시 컴파일하고 **sendmail** 프로그램에 이진 형식으로 별칭을 사용할 수 있도록 합니다. 작업 정보는 [13 장](#), “[메일 서비스\(작업\)](#)”의 [296 페이지](#) “[로컬 편지 별칭 파일 설정 방법](#)”을 참조하십시오.

현재 호스트 이름이나 호스트 이름이 없는 경우처럼 로컬 이름에 대해서만 별칭을 만들 수 있습니다. 예를 들어, 시스템 **saturn**에 우편함이 있는 사용자 **ignatz**에 대한 별칭 항목은 /etc/mail/aliases 파일에 다음과 같은 항목이 있습니다.

```
ignatz: ignatz@saturn
```

각 메일 서버에 대해 관리 계정을 만들어야 합니다. 메일 서버의 우편함을 **root**에 지정하고 **root**에 대한 항목을 /etc/mail/aliases 파일에 추가하여 해당 계정을 만듭니다. 예를 들어, 시스템 **saturn**이 우편함 서버일 경우 항목 **root: sysadmin@saturn**을 /etc/mail/aliases 파일에 추가합니다.

보통 root 사용자만 이 파일을 편집할 수 있습니다. 또는 다음과 같은 항목을 만들 수도 있습니다.

```
aliasname: :include:/path/aliasfile
```

*aliasname*은 메일을 보낼 때 사용자가 사용하는 이름이며 */path/aliasfile*은 별칭 목록이 포함된 파일의 전체 경로입니다. 별칭 파일은 행마다 항목 한 개씩, 다른 표기 없이 전자 메일 항목을 포함해야 합니다.

```
user1@host1
user2@host2
```

*/etc/mail/aliases*에서 추가 메일 파일을 정의하여 로그나 백업 복사본을 보관할 수 있습니다. 다음 항목은 *filename*의 *aliasname*에게 보내는 모든 메일을 저장합니다.

```
aliasname: /home/backup/filename
```

다른 프로세스로 메일의 경로를 지정할 수도 있습니다. 다음 예에서는 메일 메시지 복사본을 *filename*에 저장하고 복사본을 인쇄합니다.

```
aliasname: "|tee -a /home/backup/filename |lp"
```

작업 맵은 13 장, “메일 서비스(작업)”의 294 페이지 “편지 별칭 파일 관리(작업 맵)”를 참조하십시오.

## NIS aliases 맵

로컬 도메인의 모든 사용자가 NIS aliases 맵에 있는 항목을 사용할 수 있습니다. sendmail 프로그램은 NIS aliases 맵을 로컬 */etc/mail/aliases* 파일 대신 사용하여 메일링 주소를 확인하기 때문입니다. 자세한 내용은 [nsswitch.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오.

NIS aliases 맵의 별칭은 다음 형식을 따릅니다.

```
aliasname: value,value,value ...
```

*aliasname*은 메일을 보낼 때 사용자가 사용하는 이름이며 *value*는 유효한 전자 메일 주소입니다.

NIS aliases 맵은 모든 메일 클라이언트에 대한 항목을 포함해야 합니다. 일반적으로 NIS 마스터의 루트 사용자만 이 항목을 변경할 수 있습니다. 계속 변경되는 별칭에는 이 별칭 유형이 적합하지 않습니다. 그러나 다음 구문 예와 같이 별칭이 다른 별칭 파일을 가리키는 경우에는 해당 별칭이 유용합니다.

```
aliasname: aliasname@host
```

*aliasname*은 메일을 보낼 때 사용자가 사용하는 이름이며 *host*는 */etc/mail/alias* 파일이 포함된 서버의 호스트 이름입니다.

작업 정보는 13 장, “메일 서비스(작업)”의 295 페이지 “NISmail.aliases 맵 설정 방법”을 참조하십시오.

## .forward 파일

사용자는 sendmail이 다른 프로그램과 함께 메일을 재지정하거나 보내기 위해 사용할 수 있는 .forward 파일을 홈 디렉토리에 만들 수 있습니다. 다음 항목을 참조하십시오.

- 342 페이지 “피해야 할 상황”
- 342 페이지 “.forward 파일 제어”
- 343 페이지 “.forward.hostname 파일”
- 343 페이지 “.forward+detail 파일”

작업 맵은 13 장, “메일 서비스(작업)”의 304 페이지 “.forward 파일 관리(작업 맵)”를 참조하십시오.

### 피해야 할 상황

다음 목록에서는 피하거나 쉽게 해결할 수 있는 몇 가지 상황에 대해 설명합니다.

- 올바른 주소로 메일이 배달되지 않는 경우 사용자의 .forward 파일을 확인합니다. 사용자가 .forward 파일을 host1의 홈 디렉토리에 넣어 메일이 user@host2로 전달됩니다. 메일이 host2에 도착하면 sendmail은 NIS 별칭에 user가 있는지 확인하고 메시지를 다시 user@host1로 보냅니다. 이 경로 지정으로 인해 루프가 발생하고 반송 메일이 증가합니다.
- 보안 문제를 방지하려면 .forward 파일을 root 및 bin 계정에 포함시키지 마십시오. 필요할 경우 aliases 파일을 대신 사용하여 메일을 전달합니다.

### .forward 파일 제어

.forward 파일을 효과적으로 메일 배달에 포함하려면 다음과 같은 제어(대부분 권한 설정)가 올바르게 적용되어 있는지 확인하십시오.

- 파일 소유자만 .forward 파일을 쓸 수 있어야 합니다. 이 제한이 있으면 다른 사용자가 보안을 손상시킬 수 없습니다.
- root만 홈 디렉토리에 대한 경로를 소유하고 쓸 수 있어야 합니다. 예를 들어, .forward 파일이 /export/home/terry에 있으면 root만 /export 및 /export/home을 소유하고 쓸 수 있어야 합니다.
- 사용자만 실제 홈 디렉토리에 쓸 수 있어야 합니다.
- .forward 파일은 심볼릭 링크일 수 없으며 이 파일은 둘 이상의 하드 링크를 가질 수 없습니다.

## .forward.hostname 파일

.forward.hostname 파일을 만들어 특정 호스트로 보내는 메일을 재지정할 수 있습니다. 예를 들어, 사용자의 별칭이 sandy@phoenix.example.com에서 sandy@example.com으로 변경된 경우 .forward.phoenix 파일을 sandy의 홈 디렉토리에 놓으십시오.

```
% cat .forward.phoenix
sandy@example.com
"/usr/bin/vacation sandy"
% cat .vacation.msg
From: sandy@example.com (via the vacation program)
Subject: my alias has changed
```

```
My alias has changed to sandy@example.com.
Please use this alias in the future.
The mail that I just received from you
has been forwarded to my new address.
```

Sandy

이 예에서 보낸 사람은 별칭이 변경된다는 통지를 받고 메일은 올바른 대상에게 전달됩니다. vacation 프로그램은 메시지 파일을 한 개만 허용하므로 한 번에 하나씩만 메시지를 전달할 수 있습니다. 그러나 메시지가 호스트에 국한되지 않은 경우 .forward 파일이 vacation 메시지 파일 한 개를 여러 호스트에 사용할 수 있습니다.

## .forward+detail 파일

전달 방식을 다르게 확장한 것이 .forward+detail 파일입니다. detail 문자열은 연산자 문자를 제외한 모든 문자 시퀀스가 될 수 있습니다. 연산자 문자는 .:~!^[]+입니다. 이 유형의 파일을 사용하면 다른 사람이 전자 메일 주소를 몰래 사용하는지 여부를 확인할 수 있습니다. 예를 들어, 사용자가 다른 사람에게 전자 메일 주소

sandy+test1@example.com을 사용하도록 하면 사용자는 이 별칭으로 배달되는 이후의 메시지를 식별할 수 있습니다. 기본적으로 sandy+test1@example.com 별칭에 보내는 모든 메일을 별칭 및 .forward+detail 파일에 대하여 확인합니다. 일치하는 항목이 생성되지 않을 경우 메일은 sandy@example.com에 대한 배달로 폴백하지만 사용자는 받는 사람: 메일 헤더의 변경 사항을 알 수 있습니다.

## /etc/default/sendmail 파일

이 파일은 호스트가 업그레이드되면 sendmail에 대한 시작 옵션이 제거되지 않도록 옵션을 저장하는 데 사용됩니다. 다음과 같은 변수를 사용할 수 있습니다.

CLIENTOPTIONS="string"

클라이언트 데몬에 사용할 추가 옵션을 선택합니다. 이 데몬은 클라이언트 전용 대기열(/var/spool/clientmqueue)에 있으며 클라이언트 대기열 실행자 역할을 합니다. 구문 검사가 수행되지 않으므로 이 변수를 변경할 때 주의하십시오.



**CLIENTQUEUEINTERVAL=#**

QUEUEINTERVAL 옵션과 유사하게 CLIENTQUEUEINTERVAL은 메일 대기열 실행의 시간 간격을 설정합니다. 그러나 CLIENTQUEUEINTERVAL 옵션은 마스터 데몬의 기능이 아닌 클라이언트 데몬의 기능을 제어합니다. 대개 마스터 데몬은 모든 메시지를 SMTP 포트로 배달할 수 있습니다. 그러나 메시지 로드가 너무 높거나 마스터 데몬이 실행되고 있지 않은 경우 메시지가 클라이언트 전용 대기열 /var/spool/clientmqueue로 들어갑니다. 클라이언트 전용 대기열을 확인하는 클라이언트 데몬이 클라이언트 대기열 프로세서 역할을 합니다.

**ETRN\_HOSTS="string"**

SMTP 클라이언트와 서버가 주기적인 대기열 실행 간격을 기다리지 않고 즉시 상호 작용할 수 있도록 합니다. 서버는 지정된 호스트로 이동하는 대기열 부분을 즉시 배달합니다. 자세한 내용은 [etrn\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

**MODE=-bd**

sendmail을 시작할 모드를 선택합니다. -bd 옵션을 사용하거나 정의되지 않은 상태로 둡니다.

**OPTIONS=string**

마스터 데몬에 사용할 추가 옵션을 선택합니다. 구문 검사가 수행되지 않으므로 이 변수를 변경할 때 주의하십시오.

**QUEUEINTERVAL=#**

마스터 데몬의 메일 대기열 실행 간격을 설정합니다. #은 s(초), m(분), h(시간), d(일) 또는 w(주)가 뒤에 오는 양수일 수 있습니다. sendmail을 시작하기 전에 구문을 검사합니다. 간격이 음수이거나 항목이 적절한 문자로 끝나지 않을 경우 간격이 무시되고 sendmail이 대기열 간격 15분으로 시작됩니다.

**QUEUEOPTIONS=p**

대기열 실행 간격마다 새 대기열 실행자를 지정하는 대신 대기열 실행 간격 사이에 일시 정지되는 지속 대기열 실행자 하나를 사용으로 설정합니다. 이 옵션을 p로 설정할 수 있으며 사용할 수 있는 유일한 설정입니다. 그렇지 않으면 이 옵션이 설정되지 않습니다.

## 메일 주소 및 메일 경로 지정

배달하는 동안 메일 메시지가 전송되는 경로는 클라이언트 시스템의 설정과 메일 도메인의 토폴로지로 결정됩니다. 메일 호스트와 메일 도메인의 추가 레벨마다 별칭 확인이 이루어지지만 대부분의 호스트에서 경로 지정 프로세스는 기본적으로 같습니다.

로컬로 메일을 받도록 클라이언트 시스템을 설정할 수 있습니다. 로컬 메일 수신은 로컬 모드에서 sendmail 실행이라고 합니다. 로컬 모드는 모든 메일 서버와 일부 클라이언트의 기본값입니다. 로컬 모드의 메일 서버나 메일 클라이언트에서는 메일 메시지의 경로가 다음과 같이 지정됩니다.



주 - 다음 예에서는 `sendmail.cf` 파일에 설정된 기본 규칙을 사용한다고 가정합니다.

1. 가능할 경우 메일 별칭을 확장하고 로컬 경로 지정 프로세스를 다시 시작합니다.  
이름 서비스에 메일 별칭이 있는지 확인하고 새 값이 있을 경우 이름 값을 대체하여 메일 주소를 확장합니다. 이 새로운 별칭을 다시 확인합니다.
2. 메일이 로컬일 경우 `/usr/lib/mail.local`로 배달합니다.  
메일이 로컬 우편함으로 배달됩니다.
3. 메일 주소의 메일 도메인에 호스트가 있으면 메일을 해당 호스트로 배달합니다.
4. 주소의 이 도메인에 호스트가 없으면 메일을 메일 호스트로 전달합니다.  
메일 호스트는 메일 서버와 같은 경로 지정 프로세스를 사용합니다. 그러나 메일 호스트는 주소가 호스트 이름뿐 아니라 도메인 이름으로 지정된 메일을 받을 수 있습니다.

## sendmail과 이름 서비스의 상호 작용

이 절에서는 `sendmail` 및 이름 서비스에 적용되는 도메인 이름에 대해 설명합니다. 또한 이름 서비스의 효과적인 사용을 위한 규칙 및 `sendmail`과 이름 서비스의 특정 상호 작용에 대해 설명합니다. 자세한 내용은 다음 항목을 참조하십시오.

- 345 페이지 “`sendmail.cf` 및 메일 도메인”
- 346 페이지 “`sendmail` 및 이름 서비스”
- 347 페이지 “NIS 및 `sendmail`의 상호 작용”
- 348 페이지 “`sendmail`과 NIS 및 DNS의 상호 작용”

관련 작업 정보는 13 장, “메일 서비스(작업)”의 284 페이지 “`sendmail`과 함께 DNS를 사용하는 방법” 또는 294 페이지 “편지 별칭 파일 관리(작업 맵)”를 참조하십시오.

## sendmail.cf 및 메일 도메인

표준 `sendmail.cf` 파일은 메일 도메인을 사용하여 메일이 직접 배달되는지 또는 메일 호스트를 통해 배달되는지 결정합니다. 도메인 내 메일은 직접 SMTP 연결을 통해 배달되며 도메인 간 메일은 메일 호스트로 전달됩니다.

보안 네트워크에서는 선택된 호스트만 외부 대상을 목표로 하는 패킷을 생성하도록 권한이 부여됩니다. 메일 도메인의 외부에 있는 원격 호스트의 IP 주소를 호스트가 보유하고 있어도 SMTP 연결을 설정할 수 있습니다. 표준 `sendmail.cf`에서는 다음과 같이 가정합니다.

- 현재 호스트는 메일 도메인 외부의 호스트에 직접 패킷을 전송할 권한이 부여되지 않았습니다.

- 메일 호스트는 외부 호스트에 직접 패킷을 전송할 수 있는 권한이 부여된 호스트로 메일을 전달할 수 있습니다. 실제로 메일 호스트는 권한이 부여된 호스트일 수 있습니다.

이러한 가정 하에 메일 호스트는 도메인 간 메일을 배달하거나 전달합니다.

## sendmail 및 이름 서비스

sendmail은 이름 서비스에 여러 요구 사항을 부과합니다. 이 요구 사항에 대한 이해를 돕기 위해 이 절에서는 먼저 이름 서비스 도메인과 메일 도메인의 관계를 설명한 다음 여러 가지 요구 사항에 대해 설명합니다. 다음을 참조하십시오.

- 346 페이지 “메일 도메인 및 이름 서비스 도메인”
- 346 페이지 “이름 서비스의 요구 사항”
- [nsswitch.conf\(4\)](#)의 매뉴얼 페이지

### 메일 도메인 및 이름 서비스 도메인

메일 도메인 이름은 이름 서비스 도메인의 접미어여야 합니다. 예를 들어, 이름 서비스의 도메인 이름이 A.B.C.D이면 메일 도메인 이름은 다음 중 하나입니다.

- A.B.C.D
- B.C.D
- C.D
- D

처음 설정했을 때 메일 도메인 이름이 종종 이름 서비스 도메인과 동일합니다. 네트워크가 증가할수록 관리하기 쉽도록 이름 서비스 도메인이 세분화될 수 있습니다. 그러나 일관된 별칭을 제공하기 위해 이름 서비스 도메인은 나뉘지 않고 그대로 유지되기도 합니다.

### 이름 서비스의 요구 사항

이 절에서는 sendmail에서 이름 서비스에 부과하는 요구 사항에 대해 설명합니다.

세 가지 유형의 `gethostbyname()` 질의를 지원하도록 이름 서비스의 호스트 테이블이나 맵을 설정해야 합니다.

- `mailhost` – 일부 이름 서비스 구성은 자동으로 이 요구 사항을 충족합니다.
- 전체 호스트 이름(예: `smith.admin.acme.com`) – 많은 이름 서비스 구성이 이 요구 사항을 충족합니다.
- 짧은 호스트 이름(예: `smith`) – sendmail은 외부 메일을 전달하기 위해 메일 호스트에 연결해야 합니다. 메일 주소가 현재 메일 도메인에 속하는지 여부를 확인하기 위해 `gethostbyname()`이 전체 호스트 이름으로 호출됩니다. 해당 항목이 있으면 내부 주소로 간주됩니다.

NIS 및 DNS는 짧은 호스트 이름의 `gethostbyname()`을 인수로 지원하므로 이 요구 사항은 자동으로 충족됩니다.

이름 서비스 내에 효과적인 `sendmail` 서비스를 설정하려면 호스트 이름 서비스에 대한 추가 규칙 두 개를 따라야 합니다.

- 전체 호스트 이름 인수와 짧은 호스트 이름 인수를 가진 `gethostbyname()`은 일관된 결과를 만들어냅니다. 예를 들어, 두 함수가 `admin.acme.com`에서 호출될 경우 `gethostbyname(smith.admin.acme.com)`은 `gethostbyname(smith)`과 같은 결과를 반환해야 합니다.
- 공통된 메일 도메인에 속한 모든 이름 서비스 도메인에 대해 짧은 호스트 이름을 가진 `gethostbyname()`이 같은 결과를 만들어내야 합니다. 예를 들어, 메일 도메인 `smith.admin.acme.com`이 제공되면 `gethostbyname(smith)`은 `ebb.admin.acme.com` 도메인이나 `esg.admin.acme.com` 도메인에서 호출이 시작되는 경우와 같은 결과를 반환해야 합니다. 메일 도메인 이름은 대개 이름 서비스도메인보다 짧아 이 요구 사항에 여러 이름 서비스에 대한 특별한 의미를 부여합니다.

`gethostbyname()` 함수에 대한 자세한 내용은 [gethostbyname\(3NSL\)](#) 매뉴얼 페이지를 참조하십시오.

## NIS 및 sendmail의 상호 작용

다음 목록에서는 `sendmail`과 NIS의 상호 작용을 설명하고 몇 가지 지침을 제공합니다.

- **메일 도메인 이름** - NIS를 기본 이름 서비스로 설정하면 `sendmail`이 자동으로 NIS 도메인 이름의 첫번째 구성 요소를 제거하고 결과를 메일 도메인 이름으로 사용합니다. 예를 들어, `ebs.admin.acme.com`이 `admin.acme.com`이 됩니다.
- **메일 호스트 이름** - NIS 호스트 맵에 `mailhost` 항목이 있어야 합니다.
- **전체 호스트 이름** - 일반 NIS 설정은 전체 호스트 이름을 “이해”하지 못합니다. NIS에 전체 호스트 이름을 이해시키는 대신 `sendmail.cf` 파일을 편집하고 모든 `%l`을 `%y`로 대체하여 `sendmail`측에서 이 요구 사항을 해제하십시오. 이렇게 변경하면 `sendmail`의 도메인 간 메일 감지가 해제됩니다. 대상 호스트를 IP 주소로 확인할 수 있으면 직접 SMTP 배달을 시도합니다. NIS 호스트 맵이 현재 메일 도메인 외부에 있는 호스트 항목을 포함하지 않는지 확인하십시오. 그렇지 않으면 `sendmail.cf` 파일을 추가로 사용자 정의해야 합니다.
- **일치하는 전체 호스트 이름 및 짧은 호스트 이름** - 전체 호스트 이름의 `gethostbyname()`을 해제하는 방법에 대한 이전 지침을 따르십시오.
- **메일 도메인 하나의 여러 NIS 도메인** - 공통 메일 도메인에 속한 모든 NIS 호스트 맵에는 동일한 호스트 항목 세트가 있어야 합니다. 예를 들어, `ebs.admin.acme.com` 도메인의 호스트 맵은 `esg.admin.acme.com`의 호스트 맵과 같아야 합니다. 그렇지 않으면 주소 한 개가 NIS 도메인 한 개에서 작동할 수 있지만 다른 NIS 도메인에 속하게 됩니다.

작업 정보는 13 장, “메일 서비스(작업)”의 294 페이지 “편지 별칭 파일 관리(작업 맵)”를 참조하십시오.

## sendmail과 NIS 및 DNS의 상호 작용

다음 목록에서는 sendmail과 NIS 및 DNS의 상호 작용을 설명하고 몇 가지 지침을 제공합니다.

- **메일 도메인 이름** - NIS를 기본 이름 서비스로 설정하면 sendmail이 자동으로 NIS 도메인 이름의 첫 번째 구성 요소를 제거하고 결과를 메일 도메인 이름으로 사용합니다. 예를 들어, `ebs.admin.acme.com`이 `admin.acme.com`이 됩니다.
- **메일 호스트 이름** - DNS 전달 기능이 설정된 경우 NIS가 확인할 수 없는 질의가 DNS로 전달되므로 NIS 호스트 맵에 `mailhost` 항목이 필요 없습니다.
- **전체 호스트 이름** - NIS가 전체 호스트 이름을 “이해”하지 못해도 DNS는 이해합니다. NIS와 DNS를 설정하는 일반적인 절차를 따르면 이 요구 사항이 충족됩니다.
- **일치하는 전체 호스트 이름 및 짧은 호스트 이름** - NIS 호스트 테이블의 모든 호스트 항목의 경우 DNS에 해당 호스트 항목이 있어야 합니다.
- **메일 도메인 하나의 여러 NIS 도메인** - 공통 메일 도메인에 속한 모든 NIS 호스트 맵에는 동일한 호스트 항목 세트가 있어야 합니다. 예를 들어, `ebs.admin.acme.com` 도메인의 호스트 맵은 `esg.admin.acme.com` 도메인의 호스트 맵과 같아야 합니다. 그렇지 않으면 주소 한 개가 NIS 도메인 한 개에서 작동할 수 있지만 다른 NIS 도메인에 속하게 됩니다.

작업 정보는 13 장, “메일 서비스(작업)”의 284 페이지 “sendmail과 함께 DNS를 사용하는 방법” 및 294 페이지 “편지 별칭 파일 관리(작업 맵)”를 참조하십시오.

## sendmail 버전 8.14의 변경 사항

sendmail 서비스가 버전 8.14로 업데이트되었습니다. 또한 sendmail에는 다음과 같은 몇 가지 중요한 변경 사항이 있습니다.

- `sendmail.cf` 및 `submit.mc` 구성 파일을 자동으로 재구성하도록 시스템을 구성할 수 있습니다. 필요한 단계는 287 페이지 “구성 파일을 자동으로 다시 작성하는 방법”에 설명되어 있습니다.
- 기본적으로 sendmail 데몬은 새 로컬 데몬 모드에서 실행됩니다. 로컬 전용 모드에서는 cron 작업에서 보낸 메일이나 로컬 사용자 간의 메일처럼 로컬 호스트에서 받는 메일만 수락합니다. 아웃바운드 메일의 경로가 예상대로 지정되고 받는 메일만 변경됩니다. Become Local(로컬 전환) 모드로도 알려진 로컬 전용 모드를 선택하기 위해 `-bl` 옵션이 사용됩니다. 이 모드에 대한 자세한 내용은 [sendmail\(1M\)](#) 매뉴얼 페이지를 참조하십시오. `-bd` 또는 Become Daemon(데몬 전환) 모드로 다시 변경하는 방법에 대한 자세한 내용은 288 페이지 “열기 모드에서 sendmail 사용 방법”을 참조하십시오.

- `makemap`에 대한 `-t` 및 `-u` 옵션이 이제 예상한 대로 작동합니다. `-u` 옵션을 사용해도 `-t` 옵션으로 선언된 분리자가 분리자로 사용됩니다. 전에는 `-u` 옵션이 사용된 경우 `-t` 옵션으로 정의된 분리자가 있어도 공백이 분리자로 사용되었습니다. 이 옵션에 대한 자세한 내용은 [makemap\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## sendmail 버전 8.13의 변경 사항

이 버전의 `sendmail`에서는 여러 가지 새로운 기능을 제공하지만 `FallBackSmartHost` 옵션이 가장 중요한 추가 기능입니다. 이 옵션 때문에 더 이상 `main.cf`와 `subsidiary.cf`를 사용할 필요가 없습니다. `main.cf`은 MX 레코드를 지원하는 환경에서 사용되었습니다. `subsidiary.cf`은 완전히 작동하는 DNS가 없는 환경에서 사용되었습니다. 이러한 환경에서는 MX 레코드 대신 스마트 호스트가 사용되었습니다. `FallBackSmartHost` 옵션은 통합 구성을 제공합니다. 모든 환경에 대해 가능한 마지막 기본 설정의 MX 레코드처럼 작동합니다. 메일이 클라이언트에게 배달되도록 이 옵션은(사용으로 설정한 경우) 실패하는 MX 레코드의 백업 또는 페일오버 역할을 하는 스마트 호스트 또는 올바르게 연결된 호스트를 제공합니다.

버전 8.13에 대한 자세한 내용은 다음 절을 참조하십시오.

- [354 페이지](#) “`sendmail` 버전 8.13의 추가 명령줄 옵션”
- [354 페이지](#) “`sendmail` 버전 8.13의 추가 및 개정된 구성 파일 옵션”
- [356 페이지](#) “`sendmail` 버전 8.13의 추가 및 개정된 `FEATURE()` 선언”

또한 TLS(전송 계층 보안)를 사용하여 SMTP를 실행할 수 있습니다. 다음 설명을 참조하십시오.

## sendmail 버전 8.13에서 TLS를 사용하는 SMTP 실행 지원

SMTP 서버와 클라이언트 사이의 통신은 대개 한쪽에서 제어하거나 신뢰하지 않습니다. 이 보안 취약점으로 인하여 제3자가 서버와 클라이언트 사이의 통신을 모니터링하고 변경할 수 있습니다. SMTP는 버전 8.13의 `sendmail`에서 TLS(전송 계층 보안)를 사용하여 이 문제를 해결할 수 있습니다. SMTP 서버와 클라이언트에 대한 이 확장 서비스는 다음을 제공합니다.

- 인터넷을 통한 개인 인증 통신
- 도청 및 공격으로부터 보호

---

주 - TLS 구현은 SSL(Secure Sockets Layer) 프로토콜을 기반으로 합니다.

---

STARTTLS는 TLS를 사용하여 보안 SMTP 연결을 시작하는 SMTP 키워드입니다. 두 서버 사이 또는 서버와 클라이언트 사이에 이 보안 연결이 이루어질 수 있습니다. 보안 연결은 다음과 같이 정의됩니다.

- 보안 전자 메일 주소와 대상 주소가 암호화됩니다.
- 전자 메일 메시지 내용이 암호화됩니다.

클라이언트가 STARTTLS 명령을 실행하면 서버가 다음 중 하나로 응답합니다.

- 220 Ready to start TLS
- 501 Syntax error (no parameters allowed)
- 454 TLS not available due to temporary reason

220 응답은 TLS 협상을 시작하도록 클라이언트에 요구합니다. 501 응답은 클라이언트가 STARTTLS 명령을 잘못 실행했음을 나타냅니다. STARTTLS가 매개변수 없이 실행됩니다.

454 응답은 클라이언트가 규칙 세트 값을 적용하여 연결을 수락할지 또는 유지할지 결정하도록 요구합니다.

인터넷의 SMTP 기반구조를 유지 관리하려면 공용으로 사용되는 서버가 TLS 협상을 요구하면 안 됩니다. 그러나 개인적으로 사용되는 서버는 TLS 협상을 수행하도록 클라이언트에 요구할 수 있습니다. 이 경우 서버는 다음 응답을 반환합니다.

530 Must issue a STARTTLS command first

530 응답은 STARTTLS 명령을 실행하여 연결을 설정하도록 클라이언트에 지시합니다.

인증 및 프라이버시 레벨이 충족되지 않으면 서버나 클라이언트가 연결을 거부할 수 있습니다. 또는 대부분의 SMTP 연결이 안전하지 않으므로 서버와 클라이언트가 비보안 연결을 유지할 수 있습니다. 연결을 유지할지 거부할지 여부는 서버와 클라이언트 구성에 의해 결정됩니다.

TLS를 사용하는 SMTP 실행 지원은 기본적으로 사용으로 설정되지 않습니다. SMTP 클라이언트가 STARTTLS 명령을 실행하면 TLS가 사용으로 설정됩니다. SMTP 클라이언트가 이 명령을 실행하려면 먼저 sendmail이 TLS를 사용할 수 있게 하는 인증서를 설정해야 합니다. [289 페이지](#) “TLS를 사용하도록 SMTP를 설정하는 방법”을 참조하십시오. 이 절차에는 새 구성 파일 옵션 정의와 sendmail.cf 파일 재작성이 포함됩니다.

## TLS를 사용하여 SMTP를 실행하기 위한 구성 파일 옵션

다음 표에서는 TLS를 사용하여 SMTP를 실행하는 데 사용되는 구성 파일 옵션에 대해 설명합니다. 이러한 옵션 중 하나를 선언할 경우 다음 구문 중 하나를 사용하십시오.

- 0 *OptionName*= *argument* # for the configuration file
- -0 *OptionName*= *argument* # for the command line
- define('m4Name', *argument*) # for m4 configuration

표 14-12 TLS를 사용하여 SMTP를 실행하기 위한 구성 파일 옵션

옵션	설명
CACertFile	m4 이름: confCACERT 인수: <i>filename</i> 기본값: 정의되지 않음 CA 인증서 하나를 포함하는 파일을 식별합니다.
CACertPath	m4 이름: confCACERT_PATH 인수: <i>path</i> 기본값: 정의되지 않음 CA의 인증서가 포함된 디렉토리 경로를 식별합니다.
ClientCertFile	m4 이름: confCLIENT_CERT 인수: <i>filename</i> 기본값: 정의되지 않음 클라이언트의 인증서가 포함된 파일을 식별합니다. sendmail이 클라이언트 역할을 할 때 이 인증서가 사용됩니다.
ClientKeyFile	m4 이름: confCLIENT_KEY 인수: <i>filename</i> 기본값: 정의되지 않음 클라이언트 인증서에 속한 개인 키가 포함된 파일을 식별합니다.
CRLFile	m4 이름: confCRL 인수: <i>filename</i> 기본값: 정의되지 않음 X.509v3 인증에 사용되는 인증서 해지 상태가 포함된 파일을 식별합니다.
DHParameters	m4 이름: confDH_PARAMETERS 인수: <i>filename</i> 기본값: 정의되지 않음 DH(Diffie-Hellman) 매개변수가 포함된 파일을 식별합니다.

표 14-12 TLS를 사용하여 SMTP를 실행하기 위한 구성 파일 옵션 (계속)

옵션	설명
RandFile	m4 이름: confRAND_FILE 인수: file:filename 또는 egd:UNIX socket 기본값: 정의되지 않음 file: 접두어를 사용하여 임의 데이터가 포함된 파일을 식별하거나 egd: 접두어를 사용하여 UNIX 소켓을 식별합니다. Oracle Solaris OS는 난수 생성기 장치를 지원하므로 이 옵션을 지정할 필요가 없습니다. random(7D) 매뉴얼 페이지를 참조하십시오.
ServerCertFile	m4 이름: confSERVER_CERT 인수: filename 기본값: 정의되지 않음 서버의 인증서가 포함된 파일을 식별합니다. sendmail이 서버 역할을 할 때 이 인증서가 사용됩니다.
Timeout.starttls	m4 이름: confTO_STARTTLS 인수: amount of time 기본값: 1h SMTP 클라이언트가 STARTTLS 명령에 대한 응답을 기다리는 시간을 설정합니다.
TLSsrvOptions	m4 이름: confTLS_SRV_OPTIONS 인수: v 기본값: 정의되지 않음 서버가 클라이언트의 인증서를 요구하는지 여부를 확인합니다. 이 옵션이 v로 설정된 경우 클라이언트 검증이 수행됩니다.

sendmail이 SMTP의 TLS 사용을 지원하려면 다음 옵션을 정의해야 합니다.

- CACertPath
- CACertFile
- ServerCertFile
- ClientKeyFile

다른 옵션은 필요 없습니다.

### TLS를 사용하여 SMTP를 실행하기 위한 매크로

다음 표에서는 STARTTLS 명령에 사용되는 매크로에 대해 설명합니다.



표 14-13 TLS를 사용하여 SMTP를 실행하기 위한 매크로

매크로	설명
<code>\${cert_issuer}</code>	인증서 발급자인 CA(인증 기관)의 DN(고유 이름)을 보유합니다.
<code>\${cert_subject}</code>	인증서 주체라는 인증서의 DN을 보유합니다.
<code>\${cn_issuer}</code>	인증서 발급자인 CA의 CN(공통 이름)을 보유합니다.
<code>\${cn_subject}</code>	인증서 주체라는 인증서의 CN을 보유합니다.
<code>\${tls_version}</code>	연결에 사용되는 TLS의 버전을 보유합니다.
<code>\${cipher}</code>	연결에 사용되는 암호화 알고리즘 세트(암호 슈트라고 함)를 보유합니다.
<code>\${cipher_bits}</code>	연결에 사용되는 대칭 암호화 알고리즘의 키 길이를 비트 단위로 보유합니다.
<code>\${verify}</code>	제공된 인증서의 검증 결과를 보유합니다. 가능한 값은 다음과 같습니다. <ul style="list-style-type: none"> <li>■ OK - 검증에 성공했습니다.</li> <li>■ NO - 인증서가 제공되지 않았습니다.</li> <li>■ NOT - 인증서가 요청되지 않았습니다.</li> <li>■ FAIL - 제공된 인증서를 검증할 수 없습니다.</li> <li>■ NONE - STARTTLS가 수행되지 않았습니다.</li> <li>■ TEMP - 일시적인 오류가 발생했습니다.</li> <li>■ PROTOCOL - SMTP 오류가 발생했습니다.</li> <li>■ SOFTWARE - STARTTLS 핸드셰이크에 실패했습니다.</li> </ul>
<code>\${server_name}</code>	현재 나가는 SMTP 연결이 있는 서버 이름을 보유합니다.
<code>\${server_addr}</code>	현재 나가는 SMTP 연결이 있는 서버 주소를 보유합니다.

## TLS를 사용하여 SMTP를 실행하기 위한 규칙 세트

다음 표에서는 TLS를 사용하는 SMTP 연결이 수락될지 계속될지 거부될지를 결정하는 규칙 세트에 대해 설명합니다.

표 14-14 TLS를 사용하여 SMTP를 실행하기 위한 규칙 세트

규칙 세트	설명
<code>tls_server</code>	클라이언트 역할을 하는 sendmail은 이 규칙 세트를 사용하여 TLS에서 현재 서버를 지원하는지 확인합니다.
<code>tls_client</code>	서버 역할을 하는 sendmail은 이 규칙 세트를 사용하여 TLS에서 현재 클라이언트를 지원하는지 확인합니다.
<code>tls_rcpt</code>	이 규칙 세트는 받는 사람의 MTA를 검증하도록 요구합니다. 이 받는 사람 제한 사항으로 인해 DNS 스푸핑과 같은 공격이 불가능해집니다.
<code>TLS_connection</code>	이 규칙 세트는 현재 TLS 연결의 실제 매개변수에 대해 액세스 맵의 RHS에서 지정한 요구 사항을 검사합니다.

표 14-14 TLS를 사용하여 SMTP를 실행하기 위한 규칙 세트 (계속)

규칙 세트	설명
try_tls	sendmail은 이 규칙 세트를 사용하여 다른 MTA에 연결할 때 STARTTLS를 사용할 수 있는지 결정합니다. MTA가 STARTTLS를 제대로 구현할 수 없으면 STARTTLS가 사용되지 않습니다.

자세한 내용은 <http://www.sendmail.org/m4/starttls.html>을 참조하십시오.

## TLS를 사용하는 SMTP 실행 관련 보안 고려 사항

인터넷에서 실행되는 메일러를 정의하는 표준 메일 프로토콜인 SMTP는 종단간 방식이 아닙니다. 이 프로토콜 제한으로 인해 SMTP를 통한 TLS 보안에 메일 사용자에게이전트가 포함되지 않습니다. 메일 사용자에게이전트는 사용자와 메일 전송 에이전트(예: sendmail) 사이의 인터페이스 역할을 합니다.

또한 여러 서버를 통해 메일의 경로가 지정될 수도 있습니다. 철저한 SMTP 보안을 위해 SMTP 연결의 전체 체인에 TLS 지원이 필요합니다.

마지막으로, 각 서버 쌍이나 클라이언트와 서버 쌍의 협상된 인증 및 프라이버시 레벨을 고려해야 합니다. 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “인증 서비스”](#)를 참조하십시오.

## sendmail 버전 8.13의 추가 명령줄 옵션

다음 표에서는 버전 8.13의 sendmail에서 사용할 수 있는 추가 명령줄 옵션에 대해 설명합니다. 기타 명령줄 옵션은 [sendmail\(1M\)](#) 매뉴얼 페이지에서 설명합니다.

표 14-15 버전 8.13의 sendmail에서 사용 가능한 명령줄 옵션

옵션	설명
-D logfile	표준 출력에 이 정보를 포함하지 않고 표시된 logfile에 디버깅 출력을 보냅니다.
-q[!]Qsubstr	격리 reason의 하위 문자열인 이 substr이 포함된 격리된 작업의 처리를 지정합니다. -Qreason 옵션에 대한 설명을 참조하십시오. !가 추가되면 이 옵션은 이 substr이 없는 격리된 작업을 처리합니다.
-Qreason	이 reason으로 일반 대기열 항목을 격리합니다. reason이 제공되지 않은 경우 격리된 대기열 항목의 격리가 해제됩니다. 이 옵션은 -q[!]Qsubstr 옵션과 함께 사용됩니다. substr은 reason의 일부분 또는 하위 문자열입니다.

## sendmail 버전 8.13의 추가 및 개정된 구성 파일 옵션

다음 표에서는 추가 및 개정된 구성 파일 옵션에 대해 설명합니다. 이러한 옵션 중 하나를 선언할 경우 다음 구문 중 하나를 사용하십시오.

```

0 OptionName=argument      # for the configuration file
-0 OptionName=argument      # for the command line
define('m4Name',argument)  # for m4 configuration

```

표 14-16 버전 8.13의 sendmail에서 사용 가능한 구성 파일 옵션

옵션	설명
ConnectionRateWindowSize	<p>m4 이름: confCONNECTION_RATE_WINDOW_SIZE</p> <p>인수: <i>number</i></p> <p>기본 값: 60</p> <p>받는 연결을 유지할 초 수를 설정합니다.</p>
FallBackSmarHost	<p>m4 이름: confFALLBACK_SMARTHOST</p> <p>인수: <i>hostname</i></p> <p>메일이 클라이언트에게 배달되도록 이 옵션은 실패하는 MX 레코드의 백업 또는 페일오버 역할을 하는 올바르게 연결된 호스트를 제공합니다.</p>
InputMailFilters	<p>m4 이름: confINPUT_MAIL_FILTERS</p> <p>인수: <i>filename</i></p> <p>sendmail 데몬에 대한 입력 메일 필터를 나열합니다.</p>
PidFile	<p>m4 이름: confPID_FILE</p> <p>인수: <i>filename</i></p> <p>기본 값: /system/volatile/sendmail.pid</p> <p>이전 릴리스에서와 같이 파일을 열기 전에 파일 이름이 매크로 확장됩니다. 또한 버전 8.13에서는 sendmail이 종료되면 파일이 링크 해제됩니다.</p>
QueueSortOrder	<p>m4 이름: confQUEUE_SORT_ORDER</p> <p>추가된 인수: none</p> <p>버전 8.13에서는 none을 사용하여 정렬 순서가 없도록 지정합니다.</p>
RejectLogInterval	<p>m4 이름: confREJECT_LOG_INTERVAL</p> <p>인수: <i>period-of-time</i></p> <p>기본 값: 3시간을 나타내는 3h</p> <p>지정된 <i>period-of-time</i> 동안 데몬 연결이 거부되면 정보가 기록됩니다.</p>

표 14-16 버전 8.13의 sendmail에서 사용 가능한 구성 파일 옵션 (계속)

옵션	설명
SuperSafe	m4 이름: confSAFE_QUEUE  짧은 이름: s  추가된 인수: postmilter  기본 값: true  postmilter가 설정된 경우 sendmail은 모든 milters가 메시지 수락을 알릴 때까지 대기열 파일 동기화를 연기합니다. 이 인수를 유용하게 사용하려면 sendmail이 SMTP 서버로 실행되어야 합니다. 그렇지 않으면 postmilter는 true 인수를 사용하는 것처럼 작동합니다.

## sendmail 버전 8.13의 추가 및 개정된 FEATURE() 선언

다음 표에서는 추가 및 개정된 FEATURE() 선언에 대해 설명합니다. 이 m4 매크로에는 다음 구문이 사용됩니다.

FEATURE('name', 'argument')

표 14-17 sendmail 버전 8.13에서 사용 가능한 FEATURE() 선언

FEATURE() 이름	설명
conncontrol	access_db 규칙 세트와 함께 사용되어 받는 SMTP 연결 수를 확인합니다. 자세한 내용은 /etc/mail/cf/README를 참조하십시오.
greet_pause	개방형 프록시 및 smtp 슬래밍 보호를 사용으로 설정하는 greet_pause 규칙 세트를 추가합니다. 자세한 내용은 /etc/mail/cf/README를 참조하십시오.
local_lmtp	기본 인수는 계속 이 Oracle Solaris 릴리스에서 LMTP 가능 메일러인 mail.local입니다. 그러나 버전 8.13에서는 다른 LMTP 가능 메일러가 사용될 경우 경로 이름을 두번째 매개변수로 지정할 수 있으며 두번째 매개변수로 전달되는 인수를 세번째 매개변수에 지정할 수 있습니다. 예를 들면 다음과 같습니다.  FEATURE('local_lmtp', '/usr/local/bin/lmtp', 'lmtp')
mtamark	“TTXRR을 사용하여 메일 전송 에이전트 반전”(MTAMark)을 위해 실험적 지원을 제공합니다. 자세한 내용은 /etc/mail/cf/README를 참조하십시오.
ratecontrol	access_db 규칙 세트와 함께 사용되어 호스트의 연결 속도를 제어합니다. 자세한 내용은 /etc/mail/cf/README를 참조하십시오.
use_client_ptr	이 FEATURE()가 사용으로 설정되면 규칙 세트 check_relay가 이 인수 \${client_ptr}로 첫번째 인수를 대체합니다.

## sendmail 버전 8.12에서 변경된 사항

이 절에는 다음 항목에 대한 정보가 있습니다.

- 357 페이지 “sendmail 버전 8.12의 TCP 래퍼에 대한 지원”
- 358 페이지 “sendmail 버전 8.12의 submit.cf 구성 파일”
- 359 페이지 “sendmail 버전 8.12의 추가 또는 제거된 명령줄 옵션”
- 360 페이지 “sendmail 버전 8.12의 PidFile 및 ProcessTitlePrefix 옵션을 위한 추가 인수”
- 361 페이지 “sendmail 버전 8.12의 추가 정의된 매크로”
- 362 페이지 “sendmail 버전 8.12의 추가 매크로”
- 362 페이지 “sendmail 버전 8.12의 추가 MAX 매크로”
- 363 페이지 “sendmail 버전 8.12의 추가 및 개정된 m4 구성 매크로”
- 363 페이지 “sendmail 버전 8.12의 FEATURE() 선언 변경 사항”
- 366 페이지 “sendmail 버전 8.12에서 MAILER() 선언의 변경 사항”
- 367 페이지 “sendmail 버전 8.12의 추가 배달 에이전트 플러그인”
- 367 페이지 “sendmail 버전 8.12에서 배달 에이전트에 대한 등식”
- 368 페이지 “sendmail 버전 8.12의 추가 대기열 기능”
- 369 페이지 “sendmail 버전 8.12의 LDAP에 대한 변경 사항”
- 370 페이지 “sendmail 버전 8.12의 내장 메일러 변경 사항”
- 370 페이지 “sendmail 버전 8.12의 추가 규칙 세트”
- 371 페이지 “sendmail 버전 8.12의 파일 변경 사항”
- 372 페이지 “sendmail 버전 8.12 및 구성의 IPv6 주소”

## sendmail 버전 8.12의 TCP 래퍼에 대한 지원

TCP 래퍼는 특정 네트워크 서비스를 요청하는 호스트의 주소를 액세스 제어 목록(ACL)에 대해 검사하여 액세스 제어를 구현하는 방법을 제공합니다. 요청은 이에 따라 허용 또는 거부됩니다. 이 액세스 제어 메커니즘 외에도 TCP 래퍼는 또한 네트워크 서비스용 호스트 요청을 기록하며, 이는 유용한 모니터 기능입니다. 액세스 제어 아래에 있는 네트워크 서비스의 예는 rlogind, telnetd, ftpd 등입니다.

버전 8.12부터 sendmail에 TCP 래퍼를 사용할 수 있습니다. 이 검사로 다른 보안 수단이 생략되지는 않습니다. sendmail에서 TCP 래퍼를 사용하도록 설정하면 네트워크 요청을 허용하기 전에 요청의 소스를 검증하는 검사가 추가됩니다. hosts\_access(4) 매뉴얼 페이지를 참조하십시오.

---

주 - Solaris 9 릴리스부터 inetd(1M) 및 sshd(1M)에서 TCP 래퍼가 지원됩니다.

---

ACL에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스](#)의 “액세스 제어 목록을 사용하여 UFS 파일 보호”를 참조하십시오.

## sendmail 버전 8.12의 submit.cf 구성 파일

버전 8.12부터는 sendmail에 추가 구성 파일 /etc/mail/submit.cf가 포함됩니다. 이 파일 submit.cf는 데몬 모드가 아니라 Mail Submission Program 모드에서 sendmail을 실행하기 위해 사용됩니다. Mail Submission Program 모드는 데몬 모드와 달리 root 권한을 필요로 하지 않으므로 이 새로운 패러다임은 더욱 뛰어난 보안을 제공합니다.

다음의 submit.cf용 함수 목록을 참조하십시오.

- sendmail은 submit.cf를 사용하여 MSP(Mail Submission Program) 모드에서 실행됩니다. 이 모드에서는 전자 메일 메시지를 제출하며 사용자는 물론 mailx 등의 프로그램으로 MSP 모드를 시작할 수 있습니다. [sendmail\(1M\)](#) 매뉴얼 페이지에서 -Ac 옵션 및 -Am 옵션에 대한 설명을 참조하십시오.
- 다음 운영 모드에서 submit.cf가 사용됩니다.
  - -bm - 기본 운영 모드
  - -bs - 표준 입력을 사용하여 SMTP 실행
  - -bt - 주소 확인에 사용되는 테스트 모드
- submit.cf를 사용할 때 sendmail은 SMTP 데몬으로 실행되지 않습니다.
- submit.cf를 사용할 때 sendmail은 sendmail 데몬으로 배달되지 않은 메시지를 보관하는 클라이언트 전용 메일 대기열 /var/spool/clientmqueue를 사용합니다. 클라이언트 전용 대기열의 메시지는 실제로 클라이언트 대기열 실행자 역할을 하는 클라이언트 “데몬”에서 배달합니다.
- 기본적으로 sendmail은 submit.cf를 사용하여 주기적으로 MSP 대기열(클라이언트 전용 대기열이라고도 함)인 /var/spool/clientmqueue를 실행합니다.

```
/usr/lib/sendmail -Ac -q15m
```

다음 사항에 유의하십시오.

- Solaris 9 릴리스부터는 submit.cf가 자동으로 제공됩니다.
- submit.cf는 Solaris 9 릴리스 이상의 최신 릴리스를 설치하기 전에 계획이나 예비 절차를 필요로 하지 않습니다.
- 구성 파일을 지정하지 않으면 sendmail이 필요할 경우 자동으로 submit.cf를 사용합니다. 기본적으로 sendmail은 submit.cf에 적합한 작업과 sendmail.cf에 적합한 작업을 파악합니다.

## sendmail.cf와 submit.cf의 기능 차이

sendmail.cf 구성 파일은 데몬 모드용입니다. 이 파일을 사용할 때 sendmail은 root에서 시작하는 MTA(메일 전송 에이전트) 역할을 합니다.

```
/usr/lib/sendmail -L sm-mta -bd -q1h
```

sendmail.cf를 차별화하는 기타 기능은 다음 목록을 참조하십시오.

- 기본적으로 sendmail.cf는 포트 25와 587에서 SMTP 연결을 수신합니다.
- 기본적으로 sendmail.cf는 기본 대기열인 /var/spool/mqueue를 실행합니다.

## sendmail 버전 8.12 기능에서 변경된 사항

submit.cf가 추가되어 다음과 같이 기능이 변경되었습니다.

- sendmail 버전 8.12부터는 root만 메일 대기열을 실행할 수 있습니다. 자세한 내용은 [mailq\(1\)](#) 매뉴얼 페이지에서 설명하는 변경 사항을 참조하십시오. 새로운 작업 정보는 [300 페이지](#) “대기열 디렉토리 관리(작업 맵)”를 참조하십시오.
- Mail Submission Program 모드는 root 권한 없이 실행됩니다. 이 권한을 사용하면 sendmail이 .forward 파일과 같은 특정 파일에 액세스하지 못할 수도 있습니다. 따라서 sendmail용 -bv 옵션은 사용자에게 잘못된 출력을 제공할 수 있습니다. 해결책은 없습니다.
- sendmail 버전 8.12 이전에는 데몬 모드에서 sendmail을 실행하는 경우 인바운드 메일 배달만 방지합니다. sendmail 버전 8.12부터는 기본 구성으로 sendmail 데몬을 실행하지 않는 경우 아웃바운드 메일의 배달도 방지합니다. Mail Submission Program이라고도 알려진 클라이언트 대기열 실행자는 로컬 SMTP 포트에서 데몬에 메일을 제출할 수 있어야 합니다. 클라이언트 대기열 실행자가 로컬 호스트가 있는 SMTP 세션을 열 경우 데몬이 SMTP 포트에서 수신하지 않으면 메일이 대기열에 남아 있습니다. 기본 구성이 데몬을 실행하므로 기본 구성을 사용하면 이 문제가 발생하지 않습니다. 그러나 데몬을 사용 안함으로 설정한 경우 이 문제를 해결하는 방법은 [293 페이지](#) “sendmail.cf의 대체 구성을 사용하여 메일 배달을 관리하는 방법”을 참조하십시오.

## sendmail 버전 8.12의 추가 또는 제거된 명령줄 옵션

다음 표에서는 sendmail의 추가 또는 제거된 명령줄 옵션에 대해 설명합니다. 기타 명령줄 옵션은 [sendmail\(1M\)](#) 매뉴얼 페이지에서 설명합니다.

표 14-18 sendmail 버전 8.12의 추가 또는 제거된 명령줄 옵션

옵션	설명
- Ac	운영 모드가 초기 메일 제출을 지시하지 않아도 구성 파일 submit.cf를 사용할 것임을 나타냅니다. submit.cf에 대한 자세한 내용은 <a href="#">358 페이지</a> “sendmail 버전 8.12의 submit.cf 구성 파일”을 참조하십시오.
- Am	운영 모드가 초기 메일 제출을 지시해도 구성 파일 sendmail.cf를 사용할 것임을 나타냅니다. 자세한 내용은 <a href="#">358 페이지</a> “sendmail 버전 8.12의 submit.cf 구성 파일”을 참조하십시오.
- bP	각 대기열의 항목 수를 인쇄할 것임을 나타냅니다.

표 14-18 sendmail 버전 8.12의 추가 또는 제거된 명령줄 옵션 (계속)

옵션	설명
-G	명령줄에서 제출할 메시지가 최초 제출용이 아니라 중계용임을 나타냅니다. 주소가 정규화되지 않으면 메시지가 거부됩니다. 정규화가 수행되지 않습니다. <a href="http://ftp.sendmail.org">ftp://ftp.sendmail.org</a> 의 sendmail 배포에 포함된 릴리스 정보에서 설명한 대로 향후 릴리스에서는 잘못된 형식의 메시지가 거부될 수 있습니다.
-L tag	syslog 메시지에 사용되는 식별자를 제공된 tag로 설정합니다.
-q[!]I substring	받는 사람 중 한 명의 이 substring이 포함된 작업만 처리합니다. !가 추가되면 옵션은 받는 사람 중 한 명의 이 substring이 없는 작업만 처리합니다.
-q[!]R substring	대기열 ID의 이 substring이 포함된 작업만 처리합니다. !가 추가되면 옵션은 대기열 ID의 이 substring이 없는 작업만 처리합니다.
-q[!]S substring	보낸 사람의 이 substring이 포함된 작업만 처리합니다. !가 추가되면 옵션은 보낸 사람의 이 substring이 없는 작업만 처리합니다.
-qf	fork 시스템 호출을 사용하지 않고 대기열에 한 번 저장된 메시지를 처리하며 전경에서 프로세스를 실행합니다. <a href="#">fork(2)</a> 매뉴얼 페이지를 참조하십시오.
-qGname	name 대기열 그룹의 메시지만 처리합니다.
-qptime	각 대기열에 대해 포크되는 단일 자식을 사용하여 특정 시간 간격으로 대기열에 저장된 메시지를 처리합니다. 대기열 실행 사이에 자식이 일시 정지됩니다. 이 새로운 옵션은 자식을 주기적으로 포크하여 대기열을 처리하는 -q time과 비슷합니다.
-U	<a href="http://ftp.sendmail.org">ftp://ftp.sendmail.org</a> 의 sendmail 배포에 포함된 릴리스 정보에서 설명한 대로 버전 8.12에서는 이 옵션을 사용할 수 없습니다. 메일 사용자 에이전트는 -G 인수를 사용해야 합니다.

## sendmail 버전 8.12의 PidFile 및 ProcessTitlePrefix 옵션을 위한 추가 인수

다음 표에서는 PidFile 및 ProcessTitlePrefix 옵션의 추가 매크로 처리 인수에 대해 설명합니다. 이 옵션에 대한 자세한 내용은 [sendmail\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

표 14-19 PidFile 및 ProcessTitlePrefix 옵션을 위한 인수

매크로	설명
\${daemon_addr}	데몬 주소(예:0.0.0.0) 제공
\${daemon_family}	데몬 그룹(예:inet 및 inet6) 제공
\${daemon_info}	데몬 정보(예:SMTP+queueing@00:30:00) 제공
\${daemon_name}	데몬 이름(예:MSA) 제공
\${daemon_port}	데몬 포트(예:25)제공



표 14-19 PidFile 및 ProcessTitlePrefix 옵션을 위한 인수 (계속)

매크로	설명
<code>#{queue_interval}</code>	대기열 실행 간격(예: 00:30:00) 제공

## sendmail 버전 8.12의 추가 정의된 매크로

다음 표에서는 `sendmail` 프로그램에 사용하기 위해 예약된 추가 매크로에 대해 설명합니다. 매크로의 값은 내부적으로 지정됩니다. 자세한 내용은 [sendmail\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

표 14-20 sendmail의 추가 정의된 매크로

매크로	설명
<code>#{addr_type}</code>	현재 주소를 <code>Envelope</code> 보낸 사람 또는 받는 사람 주소로 식별합니다.
<code>#{client_resolve}</code>	<code>#{client_name}</code> : OK, FAIL, FORGED 또는 TEMP에 대한 <code>resolve</code> 호출 결과를 보유합니다.
<code>#{deliveryMode}</code>	<code>sendmail</code> 이 <code>DeliveryMode</code> 옵션의 값 대신 사용하는 현재 배달 모드를 지정합니다.
<code>#{dsn_notify}</code> , <code>#{dsn_envid}</code> , <code>#{dsn_ret}</code>	해당 DSN 매개변수 값을 보유합니다.
<code>#{if_addr}</code>	인터페이스가 루프백 네트에 속하지 않은 경우 받는 연결에 대한 인터페이스의 주소를 제공합니다. 이 매크로는 특히 가상 호스팅에 유용합니다.
<code>#{if_addr_out}</code> , <code>#{if_name_out}</code> , <code>#{if_family_out}</code>	<code>#{if_addr}</code> 의 재사용을 피합니다. 각각 다음 값을 보유합니다.  나가는 연결에 대한 인터페이스의 주소  나가는 연결에 대한 인터페이스의 호스트 이름  나가는 연결에 대한 인터페이스의 그룹
<code>#{if_name}</code>	받는 연결에 대해 인터페이스의 호스트 이름을 제공하며 특히 가상 호스팅에 유용합니다.
<code>#{load_avg}</code>	실행 대기열에 있는 현재 평균 작업 수를 확인하고 보고합니다.
<code>#{msg_size}</code>	메시지가 수집되기 전 ESMTP 대화상자의 메시지 크기 값( <code>SIZE=parameter</code> )을 유지합니다. 그런 후에 매크로는 <code>sendmail</code> 에서 계산한 대로 메시지 크기를 유지하며 <code>check_compat</code> 에 사용됩니다. <code>check_compat</code> 에 대한 자세한 내용은 표 14-24를 참조하십시오.

표 14-20 sendmail의 추가 정의된 매크로 (계속)

매크로	설명
<code>\${nrcpts}</code>	검증된 받는 사람 수를 보유합니다.
<code>\${ntries}</code>	배달 시도 수를 보유합니다.
<code>\${rcpt_mailer}</code> , <code>\${rcpt_host}</code> , <code>\${rcpt_addr}</code> , <code>\${mail_mailer}</code> , <code>\${mail_host}</code> , <code>\${mail_addr}</code>	메일 배달 에이전트( <code>##mailer</code> ), 호스트( <code>@host</code> ) 및 사용자( <code>\$: addr</code> )로부터 RHS(Right-Hand Side) 트리플릿으로 확인되는 RCPT 및 MAIL 인수의 구문 분석 결과를 보유합니다.

## sendmail 버전 8.12의 추가 매크로

이 절에는 sendmail 구성 파일을 작성하는 데 사용되는 추가 매크로를 설명하는 표가 있습니다.

표 14-21 sendmail 구성 파일 작성에 사용되는 추가 매크로

매크로	설명
<code>LOCAL_MAILER_EOL</code>	로컬 메일러의 기본 행 끝 문자열을 대체합니다.
<code>LOCAL_MAILER_FLAGS</code>	기본적으로 <code>Return-Path:</code> 헤더를 추가합니다.
<code>MAIL_SETTINGS_DIR</code>	메일 설정 디렉토리의 경로(후행 슬래시 포함)를 포함합니다.
<code>MODIFY_MAILER_FLAGS</code>	<code>*_MAILER_FLAGS</code> 를 향상시킵니다. 이 매크로를 플래그를 설정, 추가 또는 삭제합니다.
<code>RELAY_MAILER_FLAGS</code>	중계 메일러에 대해 추가 플래그를 정의합니다.

## sendmail 버전 8.12의 추가 MAX 매크로

다음 매크로를 사용하여 sendmail이 배달을 지연시키기 전에 받을 수 있는 최대 명령 수를 구성합니다. 컴파일 시 이 MAX 매크로를 설정할 수 있습니다. 또한 다음 표의 최대값은 현재 기본값을 나타냅니다.

표 14-22 추가 MAX 매크로

매크로	최대값	각 매크로로 확인하는 명령
<code>MAXBADCOMMANDS</code>	25	알 수 없는 명령
<code>MAXNOOPCOMMANDS</code>	20	NOOP, VERB, ONEX, XUSR
<code>MAXHELOCOMMANDS</code>	3	HELO, EHLO

표 14-22 추가 MAX 매크로 (계속)

매크로	최대값	각 매크로로 확인하는 명령
MAXVRFYCOMMANDS	6	VRFY, EXPN
MAXETRNCOMMANDS	8	ETRN

주 - 매크로의 값을 0으로 설정하여 매크로의 확인을 사용 안함으로 설정할 수 있습니다.

## sendmail 버전 8.12의 추가 및 개정된 m4 구성 매크로

이 절에는 sendmail의 추가 및 개정된 m4 구성 매크로가 있습니다. 다음 구문을 사용하여 이러한 매크로를 선언합니다.

*symbolic-name*(*'value'*)

새 `sendmail.cf` 파일을 작성하려면 13 장, “메일 서비스(작업)”의 285 페이지 “[sendmail 구성 변경](#)”을 참조하십시오.

표 14-23 sendmail의 추가 및 개정된 m4 구성 매크로

m4 매크로	설명
FEATURE()	자세한 내용은 363 페이지 “ <a href="#">sendmail 버전 8.12의 FEATURE() 선언 변경 사항</a> ”을 참조하십시오.
LOCAL_DOMAIN()	이 매크로는 클래스 <i>w</i> 에 항목을 추가합니다( $\$=w$ ).
MASQUERADE_EXCEPTION()	가장할 수 없는 호스트 또는 하위 도메인을 정의하는 새 매크로입니다.
SMART_HOST()	이제 이 매크로를 대괄호 안의 주소에 사용할 수 있습니다(예: <code>user@[host]</code> ).
VIRTUSER_DOMAIN() 또는 VIRTUSER_DOMAIN_FILE()	이러한 매크로를 사용하는 경우 $\$=\{VirtHost\}$ 를 $\$=R$ 에 포함하십시오. 미리 알림을 나타내는 $\$=R$ 은 중계가 허용된 호스트 이름 세트입니다.

## sendmail 버전 8.12의 FEATURE() 선언 변경 사항

FEATURE() 선언의 특정 변경 사항에 대한 자세한 내용은 다음 표를 참조하십시오.

신규 및 개정된 FEATURE 이름을 사용하려면 다음 구문을 사용하십시오.

FEATURE(*'name'*, *'argument'*)

새 `sendmail.cf` 파일을 작성하려면 13 장, “메일 서비스(작업)”의 285 페이지 “[sendmail 구성 변경](#)”을 참조하십시오.

표 14-24 추가 및 개정된 FEATURE() 선언

FEATURE() 이름	설명
<code>compat_check</code>	인수: 다음 단락의 예를 참조하십시오.  이 새로운 FEATURE()를 사용하면 보낸 사람 주소와 받는 사람 주소를 구성하는 액세스 맵에서 키를 찾을 수 있습니다. 이 FEATURE()는 문자열 <@>로 구분됩니다. 예를 들면 <code>sender@sdomain&lt;@&gt;recipient@rdomain</code> 입니다.
<code>delay_checks</code>	인수: 스팸 허용 테스트를 사용으로 설정하는 <code>friend</code> 또는 스팸 방지 테스트를 사용으로 설정하는 <code>hater</code>  모든 검사를 지연시키는 새로운 FEATURE()입니다. FEATURE('delay_checks')를 사용하면 클라이언트가 MAIL 명령을 연결하거나 실행할 경우 규칙 세트 <code>check_mail</code> 및 <code>check_relay</code> 가 호출되지 않습니다. 대신 <code>check_rcpt</code> 규칙 세트에서 이 규칙 세트를 호출합니다. 자세한 내용은 <code>/etc/mail/cf/README</code> 파일을 참조하십시오.
<code>dnsbl</code>	인수: 이 FEATURE()는 두 인수의 최대값을 수락합니다. <ul style="list-style-type: none"> <li>■ DNS 서버 이름</li> <li>■ 거부 메시지</li> </ul> <p>DNS 조회를 위해 여러 번 포함하여 반환 값을 검사할 수 있는 새로운 FEATURE()입니다. 이 FEATURE()를 사용하면 일시적인 조회 오류의 동작을 지정할 수 있습니다.</p>
<code>enhdnsbl</code>	인수: 도메인 이름  DNS 조회를 위해 반환 값을 검사할 수 있게 해주는 <code>dnsbl</code> 의 향상된 버전인 새로운 FEATURE()입니다. 자세한 내용은 <code>/etc/mail/cf/README</code> 를 참조하십시오.
<code>generics_entire_domain</code>	인수: 없음  \$=G의 하위 도메인에 <code>genericstable</code> 를 적용하기 위해 사용할 수도 있는 새로운 FEATURE()입니다.
<code>ldap_routing</code>	인수: 자세한 내용은 <a href="http://www.sendmail.org">http://www.sendmail.org</a> 의 “릴리스 정보”를 참조하십시오.  LDAP 주소 경로 지정을 구현하는 새로운 FEATURE()입니다.
<code>local_lmtp</code>	인수: LMTP 가능 메일러의 경로 이름 기본값은 <code>mail.local</code> 이며 이 Oracle Solaris 릴리스에서는 LMTP 가능합니다.  이제 로컬 메일러의 DSN(배달 상태 통지) 진단 코드 유형을 적절한 SMTP 값으로 설정하는 새로운 FEATURE()입니다.

표 14-24 추가 및 개정된 FEATURE() 선언 (계속)

FEATURE() 이름	설명
local_no_masquerade	인수: 없음  로컬 메일러에 대한 가장을 피하기 위해 사용할 수 있는 새로운 FEATURE()입니다.
lookupdotdomain	인수: 없음  액세스 맵에서 .domain을 조회하기 위해 사용할 수도 있는 새로운 FEATURE()입니다.
nocanonify	인수: canonify_hosts 또는 없음  이제 다음 기능을 포함하는 FEATURE()입니다.  CANONIFY_DOMAIN 또는 CANONIFY_DOMAIN_FILE에서 지정한 대로 도메인 목록을 정규화를 위해 \$[ 및 \$] 연산자에 전달할 수 있게 해줍니다.  canonify_hosts가 매개변수로 지정된 경우 <user@host>와 같이 호스트 이름만 있는 주소를 정규화할 수 있게 해줍니다.  구성 요소가 둘 이상인 주소에 후행 점을 추가합니다.
no_default_msa	인수: 없음  서로 다른 여러 포트에서 RFC 2476의 구현을 “수신”하기 위해 m4 생성 구성 파일에서 sendmail의 기본 설정을 해제하는 새로운 FEATURE()입니다.
nouucp	인수: ! 토큰을 허용하지 않는 reject 또는 ! 토큰을 허용하는 nospecial  주소의 로컬 부분에 ! 토큰을 사용할지 여부를 결정하는 FEATURE()입니다.
nullclient	인수: 없음  이제 스팸 방지 검사를 수행할 수 있도록 일반 구성의 전체 규칙 세트를 제공하는 FEATURE()입니다.
preserve_local_plus_detail	인수: 없음  sendmail이 로컬 배달 에이전트로 주소를 전달할 때 주소의 +detail 부분을 유지할 수 있도록 해주는 새로운 FEATURE()입니다.
preserve_luser_host	인수: 없음  LUSER_RELAY를 사용하는 경우 받는 사람 호스트의 이름을 유지할 수 있도록 해주는 새로운 FEATURE()입니다.
queuegroup	인수: 없음  받는 사람의 전체 전자 메일 주소나 도메인 기반의 대기열 그룹을 선택할 수 있도록 해주는 새로운 FEATURE()입니다.

표 14-24 추가 및 개정된 FEATURE() 선언 (계속)

FEATURE() 이름	설명
relay_mail_from	인수: <i>domain</i> 은 선택적 인수입니다.  메일을 보낸 사람이 액세스 맵에 RELAY로 나열되고 From: 헤더 행으로 태그가 지정되면 중계를 허용하는 새로운 FEATURE()입니다. 선택적 <i>domain</i> 인수가 제공되면 메일 보낸 사람의 도메인 부분도 검사됩니다.
virtuser_entire_domain	인수: 없음  이제 VIRTUSER_DOMAIN이나 VIRTUSER_DOMAIN_FILE로 채울 수 있는 일치하는 virtusertable 항목의 새 클래스인 \${VirtHost}를 적용하는 데 사용할 수 있는 FEATURE()입니다.  FEATURE('virtuser_entire_domain')은 클래스 \${VirtHost}를 전체 하위 도메인에 적용할 수도 있습니다.

다음 FEATURE() 선언은 더 이상 지원되지 않습니다.

표 14-25 지원되지 않는 FEATURE() 선언

FEATURE() 이름	대체
rbl	FEATURE('dnsbl') 및 FEATURE('enhdnsbl')가 이 제거된 FEATURE()를 대체합니다.
remote_mode	MASQUERADE_AS('\${S}')가 /etc/mail/cf/subsidiary.mc에서 FEATURE('remote_mode')를 대체합니다. \$S는 sendmail.cf에서 SMART_HOST 값입니다.
sun_reverse_alias_files	FEATURE('genericstable')
sun_reverse_alias_nis	FEATURE('genericstable')
sun_reverse_alias_nisplus	FEATURE('genericstable')

# sendmail 버전 8.12에서 MAILER() 선언의 변경 사항

MAILER() 선언은 배달 에이전트에 대한 지원을 지정합니다. 배달 에이전트를 선언하려면 다음 구문을 사용하십시오.

MAILER('symbolic-name')

다음 변경 사항에 유의하십시오.

- 이 새 버전의 sendmail에서 MAILER('smtp') 선언은 이제 추가 메일러인 dsmtplib를 포함합니다. 이 메일러는 F=% 메일러 플래그를 사용하여 주문형 배달을 제공합니다. dsmtplib 메일러 정의는 기본값이 IPC \$h인 새로운 DSMTP\_MAILER\_ARGS를 사용합니다.

- MAILER에 사용되는 규칙 세트 수가 제거되었습니다. 이제 uucp-dom과 uucp-uudom을 사용하는 경우 MAILER('smtp') 다음에 와야 하는 MAILER('uucp')를 제외하고 MAILER 나열 순서가 필요 없습니다.

메일러에 대한 자세한 내용은 320 페이지 “메일러 및 sendmail”을 참조하십시오. 새 sendmail.cf 파일을 작성하려면 13 장, “메일 서비스(작업)”의 285 페이지 “sendmail 구성 변경”을 참조하십시오.

## sendmail 버전 8.12의 추가 배달 에이전트 플래그

다음 표에서는 기본적으로 설정되지 않는 추가 배달 에이전트 플래그에 대해 설명합니다. 이 단일 문자 플래그는 부울입니다. 다음 예에 표시된 대로 구성 파일의 F= 문에 포함하거나 제외하여 플래그를 설정하거나 해제할 수 있습니다.

```
Mlocal,    P=/usr/lib/mail.local, F=lsDFMAw5:/|@qSXfmnz9, S=10/30, R=20/40,
Mprog,     P=/bin/sh, F=lsDFMoqeu9, S=10/30, R=20/40, D=$z:/,
Msmtp,     P=[IPC], F=mDFMuX, S=11/31, R=21, E=\r\n, L=990,
Mesmtp,    P=[IPC], F=mDFMuXa, S=11/31, R=21, E=\r\n, L=990,
Msmtp8,    P=[IPC], F=mDFMuX8, S=11/31, R=21, E=\r\n, L=990,
Mrelay,    P=[IPC], F=mDFMuXa8, S=11/31, R=61, E=\r\n, L=2040,
```

표 14-26 추가 메일러 플래그

플래그	설명
%	이 플래그를 사용하는 메일러는 ETRN 요청이나 -qI, -qR 또는 -qS 대기열 옵션 중 하나를 사용하여 대기열에 있는 메시지를 선택하지 않는 한 .메시지의 초기 받는 사람이나 대기열 실행에 메시지를 배달하지 않습니다.
1	이 플래그는 null 문자(예:\0)를 보내는 메일러의 기능을 사용 안함으로 설정합니다.
2	이 플래그는 ESMTP를 사용 안함으로 설정하고 SMTP를 대신 사용하도록 요청합니다.
6	이 플래그를 사용하면 메일러가 헤더를 7비트까지 제거할 수 있습니다.

## sendmail 버전 8.12에서 배달 에이전트에 대한 등식

다음 표에서는 M 배달 에이전트 정의 명령에 사용할 수 있는 추가 등식에 대해 설명합니다. 다음 구문은 새 등식이나 새 인수를 구성 파일에 이미 있는 등식에 추가하는 방법을 보여줍니다.

*Magent-name, equate, equate, ...*

다음 예에는 새로운 W= 등식이 있습니다. 이 등식은 모든 데이터가 전송된 후 반환될 때까지 메일러가 기다릴 최대 시간을 지정합니다.

```
Msmtp, P=[IPC], F=mDFMuX, S=11/31, R=21, E=\r\n, L=990, W=2m
```

m4 구성에 대한 값의 정의를 수정할 때는 다음 예에 제공된 구문을 사용하십시오.

```
define('SMTP_MAILER_MAXMSG', '1000')
```

이전의 예에서는 smtp 메일러의 각 연결에서 배달되는 메시지 수를 1,000개로 제한합니다.

새 sendmail.cf 파일을 작성하려면 13 장, “메일 서비스(작업)”의 285 페이지 “sendmail 구성 변경”을 참조하십시오.

주 - 일반적으로 미세 조정할 때만 mailer 디렉토리에서 등식 정의를 수정합니다.

표 14-27 배달 에이전트의 추가 등식

등식	설명
/=	인수: 디렉토리 경로 메일러 프로그램이 실행되기 전에 chroot()를 적용할 디렉토리를 지정합니다.
m=	인수: 이전에 define() 루틴으로 정의한 다음 m4 값 중 하나 SMTP_MAILER_MAXMSG - smtp 메일러용 LOCAL_MAILER_MAXMSG - local 메일러용 RELAY_MAILER_MAXMSG - relay 메일러용 smtp, local 또는 relay 메일러의 각 연결에서 배달되는 메시지 수를 제한합니다.
w=	인수: 시간 증분 모든 데이터가 전송된 후 메일러가 반환을 기다릴 최대 시간을 지정합니다.

## sendmail 버전 8.12의 추가 대기열 기능

다음 목록에서는 추가 대기열 기능을 자세히 설명합니다.

- 이 릴리스는 여러 대기열 디렉토리를 지원합니다. 여러 대기열을 사용하려면 다음 예에서 표시된 대로 별표(\*)로 끝나는 QueueDirectory 옵션 값을 구성 파일에 제공하십시오.

```
0 QueueDirectory=/var/spool/mqueue/q*
```

옵션 값 /var/spool/mqueue/q\*는 “q”로 시작되는 모든 디렉토리나 디렉토리에 대한 심볼릭 링크를 대기열 디렉토리로 사용합니다. sendmail이 실행되는 동안 대기열 디렉토리 구조를 변경하지 마십시오. Verbose 플래그(-v)가 비데몬 대기열 실행에 사용되지 않으면 대기열을 실행할 때마다 각 대기열 실행을 위한 별도의 프로세스가 생성됩니다. 대기열에 새로운 항목이 임의로 지정됩니다.

- 새로운 대기열 파일 명명 시스템에서는 60년 동안 고유한 파일 이름을 사용합니다. 이 시스템에서는 복잡한 파일 시스템 잠금 없이 대기열 ID를 지정할 수 있으며 대기열 사이에 대기열에 있는 항목을 쉽게 이동할 수 있습니다.



- 8.12부터는 root만 메일 대기열을 실행할 수 있습니다. 자세한 내용은 [mailq\(1\)](#) 매뉴얼 페이지에서 설명하는 변경 사항을 참조하십시오. 새로운 작업 정보는 [300 페이지](#) “대기열 디렉토리 관리(작업 맵)”를 참조하십시오.
- Envelope 분할을 수용하기 위해 이제 대기열 파일 이름 길이는 14자가 아니라 15자입니다. 이름이 14자로 제한된 파일 시스템은 더 이상 지원되지 않습니다.

작업 정보는 [300 페이지](#) “대기열 디렉토리 관리(작업 맵)”를 참조하십시오.

## sendmail 버전 8.12의 LDAP에 대한 변경 사항

다음 목록에서는 sendmail에 LDAP(Lightweight Directory Access Protocol)를 사용하는 경우의 변경 사항에 대해 설명합니다.

- LDAPROUTE\_EQUIVALENT() 및 LDAPROUTE\_EQUIVALENT\_FILE()을 사용하면 LDAP 경로 지정 조회를 위해 가장 도메인 이름으로 대체되는 해당 호스트 이름을 지정할 수 있습니다. 자세한 내용은 [/etc/mail/cf/README](#)를 참조하십시오.
- <ftp://ftp.sendmail.org>의 sendmail 배포에 포함된 릴리스 정보에서 설명한 대로 LDAPX 맵 이름이 LDAP로 바뀌었습니다. LDAP에 다음 구문을 사용하십시오.

Kldap ldap options

- 이 릴리스에서는 단일 LDAP 조회에 대해 복수 값 반환을 지원합니다. 표시된 대로 -v 옵션을 사용하여 쉼표로 구분된 문자열에 반환되도록 값을 배치하십시오.

Kldap ldap -v"mail,more-mail"

- LDAP 속성이 LDAP 맵 선언에 지정되지 않으면 일치 항목에 있는 모든 속성이 반환됩니다.
- 이 버전의 sendmail에서는 따옴표가 사용된 키의 쉼표 및 LDAP 별칭 파일의 사양에 있는 값 문자열의 단일 항목을 여러 항목으로 나눌 수 없습니다.
- 이 버전의 sendmail에는 LDAP 맵을 위한 새로운 옵션이 있습니다. 옵션 -vseparator를 사용하면 조회가 관련 separator로 구분된 속성과 값을 둘 다 반환할 수 있도록 구분자를 지정할 수 있습니다.
- %s 토큰을 사용하여 LDAP 필터 사양을 구문 분석할 뿐 아니라 새 토큰 %0을 사용하여 키 버퍼를 인코딩할 수 있습니다. %0 토큰은 문자 그대로의 의미를 LDAP 특수 문자에 적용합니다.

다음 예에서는 “\*” 조회를 위해 이러한 토큰의 차이를 보여줍니다.

표 14-28 토큰 비교

LDAP 맵 사양	동일한 사양	결과
-k"uid=%s"	-k"uid=*"	사용자 속성에 레코드 일치
-k"uid=%0"	-k"uid=\2A"	이름 “*”에 사용자 일치

다음 표에서는 추가 LDAP 맵 플래그에 대해 설명합니다.

표 14-29 추가 LDAP 맵 플래그

플래그	설명
-1	일치 항목을 하나만 반환해야 합니다. 일치 항목이 둘 이상 반환되면 레코드를 찾지 못한 것과 결과가 동일합니다.
-r never always search find	LDAP 별칭 참조 취소 옵션을 설정합니다.
-Z size	반환할 일치 항목 수를 제한합니다.

## sendmail 버전 8.12의 내장 메일러 변경 사항

이전의 [TCP] 내장 메일러는 사용할 수 없습니다. P=[IPC] 내장 메일러를 대신 사용하십시오. 이제 프로세스 간 통신([IPC]) 내장 메일러를 사용하여 이를 지원하는 시스템에서 UNIX 도메인 소켓으로 배달할 수 있습니다. 명명된 소켓에서 수신하는 LMTP 배달 에이전트와 함께 이 메일러를 사용할 수 있습니다. 메일러의 예는 다음과 같습니다.

```
Mexecmail, P=[IPC], F=lsDFMmqSXzA5@/:|, E=\r\n,  
S=10, R=20/40, T=DNS/RFC822/X-Unix, A=FILE /system/volatile/lmtpd
```

이제 [IPC] 메일러의 첫번째 메일러 인수의 값이 적합한지 검사합니다. 다음 표에는 첫번째 메일러 인수의 가능한 값이 있습니다.

표 14-30 첫번째 메일러 인수의 가능한 값

값	설명
A=FILE	UNIX 도메인 소켓 배달에 사용
A=TCP	TCP/IP 연결에 사용
A=IPC	이제 첫번째 메일러 인수로 사용할 수 없음

## sendmail 버전 8.12의 추가 규칙 세트

다음 표에서는 추가 규칙 세트를 나열하고 규칙 세트의 역할에 대해 설명합니다.

표 14-31 새 규칙 세트

Set	설명
check_eoh	헤더 사이에서 수집한 정보를 상호 연결하고 누락된 헤더가 있는지 검사합니다. 이 규칙 세트는 매크로 저장소 맵에 사용되며 모든 헤더를 수집한 후 호출됩니다.

표 14-31 새 규칙 세트 (계속)

Set	설명
check_etrn	check_rcpt가 RCPT를 사용할 때 ETRN 명령을 사용합니다.
check_expn	check_rcpt가 RCPT를 사용할 때 EXPN 명령을 사용합니다.
check_vrfy	check_rcpt가 RCPT를 사용할 때 VRFY 명령을 사용합니다.

다음 목록에서는 추가 규칙 세트 기능에 대해 설명합니다.

- 번호가 있는 규칙 세트에 이름도 지정할 수 있지만 번호로 규칙 세트에 액세스할 수 있습니다.
- H 헤더 구성 파일 명령을 사용하면 헤더 검사를 위해 기본 규칙 세트를 지정할 수 있습니다. 개별 헤더에 해당 규칙 세트가 지정되지 않은 경우에만 이 규칙 세트가 호출됩니다.
- 구성 파일 버전이 9 이상이면 규칙 세트 안의 주석(괄호 안의 텍스트)이 제거되지 않습니다. 예를 들어, 다음 규칙은 입력 token (1)과 일치하지만 입력 token과는 일치하지 않습니다.  
R\$+ (1)            \$@ 1
- sendmail은 check\_relay 규칙 세트의 TCP 래퍼 때문에 명령을 거부할 경우에도 SMTP RSET 명령을 수락합니다.
- OperatorChars 옵션을 여러 번 설정하면 경고가 나타납니다. 또한 규칙 세트를 정의한 후에는 OperatorChars를 설정하지 마십시오.
- 잘못된 규칙 세트를 선언하면 규칙 세트 행과 이름이 무시됩니다. S0에는 규칙 세트 행이 추가되지 않습니다.

## sendmail 버전 8.12의 파일 변경 사항

다음 변경 사항에 유의하십시오.

- 읽기 전용 /usr 파일 시스템을 지원하기 위해 /usr/lib/mail 디렉토리의 내용이 /etc/mail/cf 디렉토리로 이동했습니다. 자세한 내용은 [331 페이지 “/etc/mail/cf 디렉토리의 내용”](#)을 참조하십시오. 그러나 /usr/lib/mail/sh/check-hostname 및 /usr/lib/mail/sh/check-permissions 셸 스크립트는 이제 /usr/sbin 디렉토리에 있습니다. [334 페이지 “메일 서비스에 사용되는 기타 파일”](#)을 참조하십시오. 역방향 호환성을 위하여 심볼 링크가 각 파일의 새 위치를 가리킵니다.
- /usr/lib/mail/cf/main-v7sun.mc의 새로운 이름은 /etc/mail/cf/cf/main.mc입니다.
- /usr/lib/mail/cf/subsidiary-v7sun.mc의 새로운 이름은 /etc/mail/cf/cf/subsidiary.mc입니다.
- helpfile은 이제 /etc/mail/helpfile에 있습니다. 이전 이름(/etc/mail/sendmail.hf)에는 새 이름으로 연결되는 심볼릭 링크가 있습니다.

- `trusted-users` 파일은 이제 `/etc/mail/trusted-users`에 있습니다. 업그레이드하는 동안 이전 이름(`/etc/mail/sendmail.ct`)만 발견되고 새 이름은 발견되지 않으면 이전 이름에서 새 이름으로 하드 링크가 생성됩니다. 그렇지 않으면 변경되지 않습니다. 기본 콘텐츠는 `root`입니다.
- `local-host-names` 파일은 이제 `/etc/mail/local-host-names`에 있습니다. 업그레이드하는 동안 이전 이름(`/etc/mail/sendmail.cw`)만 발견되고 새 이름은 발견되지 않으면 이전 이름에서 새 이름으로 하드 링크가 생성됩니다. 그렇지 않으면 변경되지 않습니다. 기본 콘텐츠 길이는 0입니다.

## sendmail 버전 8.12 및 구성의 IPv6 주소

sendmail 버전 8.12부터는 구성에 사용되는 IPv6 주소에 `IPv6:` 태그가 접두어로 지정되어 주소를 적절하게 식별해야 합니다. IPv6 주소를 식별하지 않으면 접두어 태그가 사용되지 않습니다.

## 제 5 부

# 직렬 네트워킹 항목

직렬 네트워킹에 대한 이 절에서는 PPP 및 UUCP의 개요, 작업 및 참조 정보를 제공합니다.



## Solaris PPP 4.0(개요)

---

이 절에서는 직렬 네트워킹 관련 항목을 다룹니다. 직렬 네트워킹은 데이터를 전송할 수 있도록 둘 이상의 컴퓨터를 연결하기 위해 RS-232 또는 V.35 포트와 같은 직렬 인터페이스를 사용하는 것을 가리킵니다. 이더넷과 같은 LAN 인터페이스와 달리 이러한 직렬 인터페이스는 서로 멀리 떨어져 있는 여러 시스템을 연결하는데 사용됩니다. PPP(지점 간 프로토콜) 및 UUCP(UNIX 간 복사)는 직렬 네트워킹을 구현하는데 사용할 수 있는 특별한 기술입니다. 네트워킹용으로 구성된 직렬 인터페이스는 이더넷과 같은 다른 네트워크 인터페이스와 거의 같은 방식으로 여러 사용자에게 제공됩니다.

이 장에서는 Solaris PPP 4.0을 소개합니다. 이 버전의 PPP를 사용하면 서로 다른 물리적 위치에 있는 두 컴퓨터가 다양한 매체를 통해 PPP를 사용하여 서로 통신할 수 있습니다. Solaris PPP 4.0은 기본 설치의 구성 요소입니다.

다음 항목을 다룹니다.

- 375 페이지 “Solaris PPP 4.0 기본 사항”
- 378 페이지 “PPP 구성 및 용어”
- 384 페이지 “PPP 인증”
- 386 페이지 “PPPoE를 통한 DSL 사용자 지원”

## Solaris PPP 4.0 기본 사항

Solaris PPP 4.0은 TCP/IP 프로토콜 집합의 구성원이자 데이터 링크 프로토콜인 PPP(지점 간 프로토콜)를 구현합니다. PPP는 전화선과 같은 통신 매체를 통해 두 끝점 시스템 사이에서 데이터가 전송되는 방법을 기술합니다.

1990년대 초반부터 PPP는 통신 링크를 통해 데이터그램을 보내기 위한 인터넷 표준으로 널리 사용되고 있습니다. PPP 표준은 IETF(Internet Engineering Task Force)의 Point-to-Point Working Group에 의해 RFC 1661에 기술되어 있습니다. PPP는 원격 컴퓨터가 인터넷 서비스 제공업체(ISP)나 수신 호출을 받도록 구성된 회사 서버를 호출할 때 일반적으로 사용됩니다.

Solaris PPP 4.0은 공개적으로 사용 가능한 ANU(Australian National University) PPP-2.4를 기반으로 하며 PPP 표준을 구현합니다. 비동기 PPP 링크와 동기 PPP 링크가 모두 지원됩니다.

## Solaris PPP 4.0 호환성

다양한 버전의 표준 PPP가 제공되어 인터넷 커뮤니티에서 널리 사용되고 있습니다. ANU PPP-2.4가 Linux, Tru64 UNIX 및 세 가지 모든 주요 BSD 변형에 대해 가장 많이 사용되고 있습니다.

- FreeBSD
- OpenBSD
- NetBSD

Solaris PPP 4.0은 원하는 대로 구성이 가능한 ANU PPP-2.4의 기능을 Oracle Solaris 운영 체제를 실행하는 시스템에 제공합니다. Solaris PPP 4.0을 실행하는 시스템은 표준 PPP의 구현을 실행하는 모든 시스템에 대한 PPP 링크를 손쉽게 설정할 수 있습니다.

ANU를 기반으로 하지 않는 일부 PPP 구현 중 Solaris PPP 4.0과 성공적으로 상호 운영되는 구현의 예는 다음과 같습니다.

- Solaris PPP(asppp라고도 하며 Solaris 2.4 ~ Solaris 8 릴리스에서 사용 가능함)
- Solstice PPP 3.0.1
- Microsoft Windows 98 DUN
- Cisco IOS 12.0(동기)

## 사용할 Solaris PPP 버전

Solaris PPP 4.0은 지원되는 PPP 구현입니다. Solaris 9 이상 릴리스에는 이전 버전의 비동기 Solaris PPP(asppp) 소프트웨어가 포함되어 있지 않습니다. 자세한 내용은 [23 장, “비동기 Solaris PPP에서 Solaris PPP 4.0으로 마이그레이션\(작업\)”](#)을 참조하십시오.

## Solaris PPP 4.0을 사용하는 이유

현재 asppp를 사용하고 있는 경우 Solaris PPP 4.0으로 마이그레이션해 보십시오. 두 Solaris PPP 기술의 차이점은 다음과 같습니다.

- **전송 모드**  
asppp는 비동기 통신만 지원하고, Solaris PPP 4.0은 비동기 통신과 동기 통신을 모두 지원합니다.
- **구성 프로세스**  
asppp를 설정하려면 asppp.cf 구성 파일, 세 UUCP 파일 및 ipadm 명령을 구성해야 합니다. 또한 시스템에 로그인할 수 있는 모든 사용자에게 대해 인터페이스를 미리 구성해야 합니다.



Solaris PPP 4.0을 설정하려면 PPP 구성 파일에 대한 옵션을 정의하거나 옵션을 사용하여 `pppd` 명령을 실행해야 합니다. 구성 파일과 명령줄 메소드를 조합하여 사용할 수도 있습니다. Solaris PPP는 인터페이스를 동적으로 만들고 제거합니다. 각 사용자에게 대해 직접 PPP 인터페이스를 구성할 필요는 없습니다.

- **asppp에서 사용할 수 없는 Solaris PPP 4.0 기능**
  - MS-CHAPv1 및 MS-CHAPv2 인증
  - ADSL 브릿지를 지원하기 위한 PPPoE(PPP over Ethernet)
  - PAM 인증
  - 플러그인 모듈
  - IPv6 주소 지정
  - Deflate 또는 BSD 압축을 사용하는 데이터 압축
  - Microsoft 클라이언트측 콜백 지원

## Solaris PPP 4.0 업그레이드 경로

기존 `asppp` 구성을 Solaris PPP 4.0으로 변환하는 경우 이 릴리스와 함께 제공되는 변환 스크립트를 사용할 수 있습니다. 자세한 내용은 508 페이지 “`asppp`에서 Solaris PPP 4.0으로 변환하는 방법”을 참조하십시오.

## PPP에 대한 추가 정보

PPP에 대한 정보가 포함된 많은 자원이 인쇄물 및 온라인으로 제공되어 있습니다. 다음 세부절에는 일부 제안 사항이 제공되어 있습니다.

### PPP에 대한 전문 참조 설명서

널리 사용되는 PPP 구현(ANU PPP 포함)에 대한 자세한 내용은 다음 설명서를 참조하십시오.

- Carlson, James. **PPP Design, Implementation, and Debugging**. 제2판 Addison-Wesley, 2000.
- Sun, Andrew. **Using and Managing PPP**. O'Reilly & Associates, 1999.

### PPP에 대한 웹 사이트

PPP에 대한 일반적인 정보를 얻으려면 다음 웹 사이트로 이동하십시오.

- 기술 정보, FAQ, Oracle Solaris 시스템 관리에 대한 설명 및 이전 버전의 PPP를 얻으려면 시스템 관리자 자원인 <http://www.sun.com/bigadmin/home/index.html>로 이동하십시오.
- 다양한 PPP 구현을 위한 모뎀 구성 및 조언은 Stokely Consulting의 Web Project Management & Software Development 웹 사이트인 <http://www.stokely.com/unix.serial.port.resources/ppp.slip.html>을 참조하십시오.

## PPP에 대한 RFC(Request for Comments)

PPP에 대한 일부 유용한 인터넷 RFC는 다음과 같습니다.

- 1661 및 1662 - PPP의 주요 기능을 기술합니다.
- 1334 - PAP(암호 인증 프로토콜) 및 CHAP(Challenge-Handshake 인증 프로토콜)와 같은 인증 프로토콜을 기술합니다.
- 1332 - PPPoE(PPP over Ethernet)를 기술하는 정보 RFC입니다.

PPP RFC 사본을 얻으려면 IETF RFC 웹 페이지(<http://www.ietf.org/rfc.html>)에서 RFC 번호를 지정하십시오.

## PPP에 대한 매뉴얼 페이지

Solaris PPP 4.0 구현에 대한 기술 정보는 다음 매뉴얼 페이지를 참조하십시오.

- [pppd\(1M\)](#)
- [chat\(1M\)](#)
- [pppstats\(1M\)](#)
- [pppoec\(1M\)](#)
- [pppoed\(1M\)](#)
- [sppptun\(1M\)](#)
- [snoop\(1M\)](#)

[pppdump\(1M\)](#)에 대한 매뉴얼 페이지도 참조하십시오. PPP 관련 매뉴얼 페이지는 `man` 명령을 사용하여 찾을 수 있습니다.

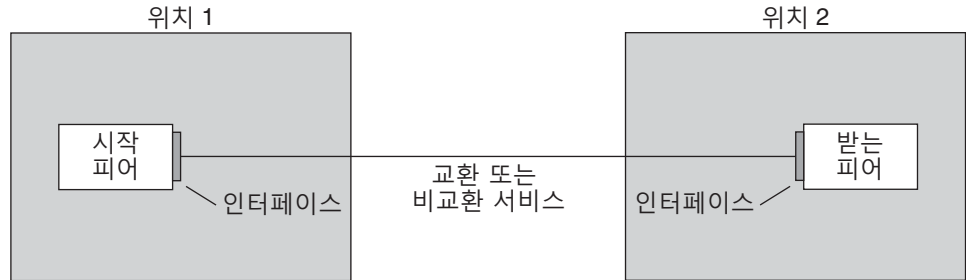
# PPP 구성 및 용어

이 절에서는 PPP 구성을 소개합니다. 이 절에서는 이 설명서에 사용되는 용어도 정의합니다.

Solaris PPP 4.0은 많은 구성을 지원합니다.

- 교환 액세스 또는 **다이얼 업** 구성
- 기계적 연결 또는 **전용 회선** 구성

그림 15-1 PPP 링크의 각 부분



이전 그림에서는 기본 PPP 링크를 보여줍니다. 링크에는 다음과 같은 부분이 있습니다.

- 일반적으로 서로 다른 물리적 위치에 있는 두 시스템(**피어**라고 함). 피어는 사이트 요구 사항에 따라 개인용 컴퓨터, 엔지니어링 워크스테이션, 대규모 서버, 심지어는 상용 라우터일 수 있습니다.
- 각 피어의 직렬 인터페이스. Oracle Solaris 시스템에서 이 인터페이스는 관리자가 비동기 PPP를 구성하는지, 아니면 동기 PPP를 구성하는지에 따라 `cua`, `hihp` 또는 기타 인터페이스일 수 있습니다.
- 직렬 케이블 등의 물리적 링크, 모뎀 연결 또는 네트워크 공급자로부터 임대 받은 T1/T3 회선 등의 전용 회선

## 다이얼 업 PPP 개요

가장 일반적으로 사용되는 PPP 구성은 **다이얼 업 링크**입니다. 다이얼 업 링크에서 로컬 피어는 원격 피어를 **다이얼 업**하여 연결을 설정하고 PPP를 실행합니다. 다이얼 업 프로세스에서 로컬 피어는 원격 피어의 전화번호로 전화를 걸어 링크를 시작합니다.

일반적인 다이얼 업 시나리오로는 수신 호출을 받도록 구성된 ISP에서 피어를 호출하는 홈 컴퓨터를 들 수 있습니다. 다른 시나리오로는 PPP 링크를 통해 다른 빌딩에 있는 피어로 데이터를 전송하는 로컬 시스템을 사용하는 회사 사이트를 들 수 있습니다.

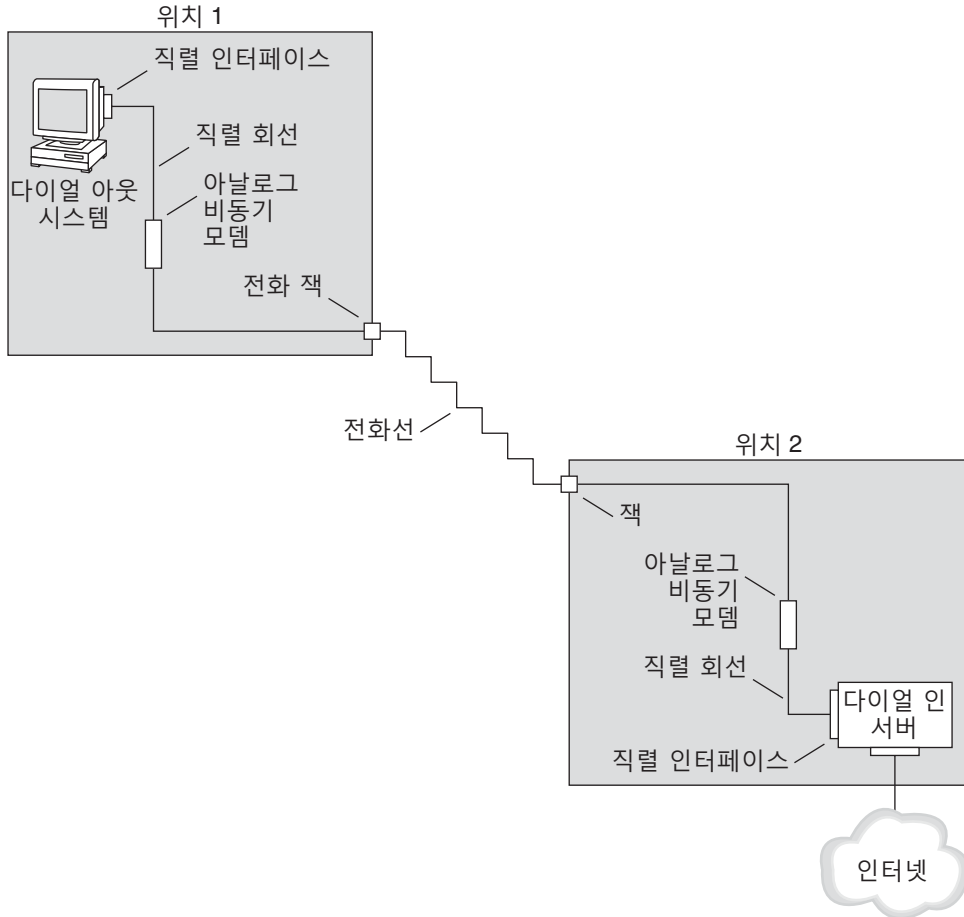
이 설명서에서는 다이얼 업 연결을 시작하는 로컬 피어를 **다이얼 아웃 시스템**이라고 합니다. 또한 수신 호출을 받는 피어를 **다이얼 인 서버**라고 합니다. 이 시스템은 실제로 다이얼 아웃 시스템의 대상 피어이며 실제 서버이거나 실제 서버가 아닐 수 있습니다.

PPP는 클라이언트-서버 프로토콜이 아닙니다. 일부 PPP 문서에서는 "클라이언트" 및 "서버"라는 용어가 전화 호출 설정을 가리키는 데 사용됩니다. 다이얼 인 서버는 파일 서버 또는 이름 서버와 같은 실제 서버가 아닙니다. 다이얼 인 시스템은 종종 둘 이상의 다이얼 아웃 시스템에 네트워크 접근성을 제공하는 데 사용되므로, 다이얼 인 서버라는 말은 널리 사용되는 PPP 용어입니다. 그럼에도 불구하고 다이얼 인 서버는 다이얼 아웃 시스템의 대상 피어입니다.

## 다이얼 업 PPP 링크의 각 부분

다음 그림을 참조하십시오.

그림 15-2 기본 아날로그 다이얼 업 PPP 링크



링크의 다이얼 아웃측인 위치 1에 대한 구성은 다음 요소로 구성되어 있습니다.

- 다이얼 아웃 시스템(일반적으로 사용자 집에 있는 개인용 컴퓨터 또는 워크스테이션)
- 다이얼 아웃 시스템에 있는 직렬 인터페이스. /dev/cua/a 또는 /dev/cua/b는 Oracle Solaris 소프트웨어를 실행하는 시스템에 있는 발신 호출을 위한 표준 직렬 인터페이스입니다.
- 전화 잭에 연결된 비동기 모뎀 또는 ISDN TA(터미널 어댑터)

- 전화 회사의 전화선 및 서비스

링크의 다이얼 인측인 위치 2에 대한 구성은 다음 요소로 구성되어 있습니다.

- 전화 네트워크에 연결된 전화 잭 또는 이와 유사한 커넥터
- 비동기 모뎀 또는 ISDN TA
- 다이얼 인 서버에 있는 직렬 인터페이스(수신 호출용 ttya 또는 ttyb)
- 회사 인트라넷과 같이 네트워크에 연결되어 있거나 전역 인터넷인 ISP의 인스턴스에 있는 다이얼 인 서버

## 다이얼 아웃 시스템에서 ISDN 터미널 어댑터 사용

외부 ISDN TA의 속도는 모뎀보다 빠르지만 TA는 기본적으로 동일하게 구성합니다. ISDN TA를 구성할 때의 주요 차이점은 채트 스크립트에 있습니다. 여기에는 TA 제조업체와 관련된 명령이 필요합니다. ISDN TA의 채트 스크립트에 대한 자세한 내용은 482 페이지 “외부 ISDN TA를 위한 채트 스크립트”를 참조하십시오.

## 다이얼 업 통신 중 발생하는 작업

다이얼 아웃 피어와 다이얼 인 피어 모두에 있는 PPP 구성 파일에 링크 설정 명령이 포함됩니다. 다이얼 업 링크를 시작하면 다음 프로세스가 발생합니다.

1. 다이얼 아웃 시스템에 있는 사용자 또는 프로세스가 `pppd` 명령을 실행하여 링크를 시작합니다.
2. 다이얼 아웃 시스템이 해당 PPP 구성 파일을 읽습니다. 그러면 다이얼 아웃 시스템이 직렬 회선을 통해 해당 모뎀으로 명령을 보냅니다(다이얼 인 서버의 전화 번호 포함).
3. 모뎀이 해당 전화 번호로 전화를 걸어 다이얼 인 서버에 있는 모뎀과 전화 연결을 설정합니다.  
다이얼 아웃 시스템이 모뎀 및 다이얼 인 서버로 보내는 일련의 텍스트 문자열이 **채트 스크립트**라는 파일에 포함됩니다. 필요한 경우 다이얼 아웃 시스템이 다이얼 인 서버에 명령을 보내 서버에서 PPP를 호출합니다.
4. 다이얼 인 서버에 연결된 모뎀이 다이얼 아웃 시스템에 있는 모뎀과 링크 협상을 시작합니다.
5. 모뎀 간 협상이 완료되면 다이얼 아웃 시스템에 있는 모뎀이 "CONNECT(연결)"를 보고합니다.
6. 두 피어 모두의 PPP가 *Establish(설정)* 단계로 들어갑니다. 이 단계에서는 LCP(링크 제어 프로토콜)가 기본 링크 매개변수와 인증 사용을 협상합니다.
7. 필요한 경우 피어가 서로를 인증합니다.
8. PPP의 NCP(Network Control Protocol)가 IPv4 또는 IPv6과 같은 네트워크 프로토콜의 사용을 협상합니다.

그러면 다이얼 아웃 시스템이 다이얼 인 서버를 통해 연결할 수 있는 호스트에 대해 `telnet` 또는 이와 유사한 명령을 실행할 수 있게 됩니다.

## 전용 회선 PPP 개요

기계적으로 연결된 **전용 회선** PPP 구성에는 링크로 연결된 두 피어가 사용됩니다. 이 링크는 공급자로부터 임대 받은 교환 또는 비교환 디지털 서비스로 구성됩니다. Solaris PPP 4.0은 어떠한 전이중 지점 간 전용 회선 매체를 통해서도 작동합니다. 일반적으로 회사는 네트워크 공급자로부터 기계적으로 연결된 링크를 임대 받아 ISP 또는 기타 원격 사이트에 연결합니다.

## 다이얼 업 링크와 전용 회선 링크 비교

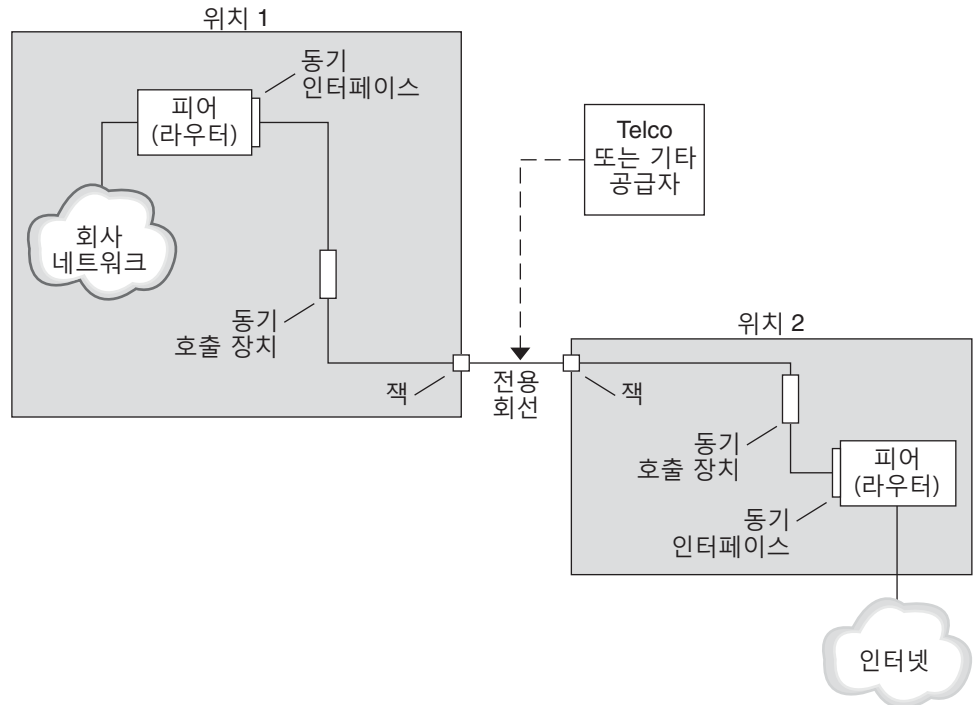
다이얼 업 링크와 전용 회선 링크 모두에는 통신 매체로 연결되는 두 피어가 사용됩니다. 다음 표에는 여러 링크 유형의 차이점이 요약되어 있습니다.

전용 회선	다이얼 업 회선
시스템 관리자나 전원 장애로 인해 전용 회선이 차단되지 않는 한 항상 연결되어 있습니다.	사용자가 원격 피어를 호출하려고 할 때 필요에 따라 시작됩니다.
동기 통신과 비동기 통신을 사용합니다. 비동기 통신의 경우 장거리 모뎀이 자주 사용됩니다.	비동기 통신을 사용합니다.
공급자로부터 임대 받습니다.	기존 전화선을 사용합니다.
동기 장치가 필요합니다.	보다 저렴한 모뎀을 사용합니다.
대부분의 SPARC 시스템에서 혼한 동기 포트가 필요합니다. 그러나 x86 시스템 및 최신 SPARC 시스템에서는 동기 포트가 혼하지 않습니다.	대부분의 컴퓨터에 포함되어 있는 표준 직렬 인터페이스를 사용합니다.

## 전용 회선 PPP 링크의 각 부분

다음 그림을 참조하십시오.

그림 15-3 기본 전용 회선 구성



전용 회선 링크에는 다음 부분이 포함되어 있습니다.

- **두 피어**(각 피어가 링크의 한쪽 끝에 있음). 각 피어는 워크스테이션 또는 서버일 수 있습니다. 피어는 해당 네트워크 또는 인터넷과 반대쪽 피어 간의 라우터 기능을 하는 경우가 많습니다.
- **각 피어의 동기 인터페이스**. Oracle Solaris 소프트웨어를 실행하는 일부 시스템에서는 HSI/P와 같은 동기 인터페이스 카드를 구입해야 전용 회선에 연결할 수 있습니다. UltraSPARC 워크스테이션과 같은 기타 시스템에는 내장 동기 인터페이스가 있습니다.
- **각 피어의 CSU/DSU 동기 디지털 장치**(동기 포트를 전용 회선에 연결함). 위치에 따라 CSU를 DSU에 내장하거나, 직접 소유하거나, 공급자에게서 임대 받을 수 있습니다. DSU는 Oracle Solaris 시스템에 표준 동기 직렬 인터페이스를 제공합니다. FRAD(프레임 릴레이 액세스 장치)는 프레임 릴레이를 사용하여 직렬 인터페이스 적응을 수행합니다.
- **전용 회선**(교환 또는 비교환 디지털 서비스를 제공함). SONET/SDH, 프레임 릴레이 PVC, T1 등을 예로 들 수 있습니다.

## 전용 회선 통신 중 발생하는 작업

대부분의 전용 회선 유형에서 피어는 실제로 서로에게 전화를 걸지 않습니다. 대신 회사가 전용 회선 서비스를 구입하여 두 고정 위치를 명시적으로 연결합니다. 전용 회선의 각 끝에 있는 두 피어가 같은 회사의 서로 다른 물리적 위치에 있는 경우가 있습니다. ISP에 연결되어 있는 전용 회선에서 라우터를 설정하는 회사의 경우도 있습니다.

전용 회선은 다이얼 업 링크보다 덜 일반적으로 사용되지만 기계적으로 연결된 링크는 설정하기가 더 쉽습니다. 기계적으로 연결된 링크에는 채트 스크립트가 필요하지 않습니다. 회선을 임대 받는 경우에는 두 피어가 서로에게 알려져 있으므로 인증이 사용되지 않는 경우가 많습니다. 두 피어가 링크를 통해 PPP를 시작하면 링크가 활성화 상태로 유지됩니다. 전용 회선 링크는 회선에 문제가 발생하거나 둘 중 하나의 피어가 명시적으로 링크를 종료하지 않는 한 활성화 상태로 유지됩니다.

Solaris PPP 4.0을 실행하는 전용 회선의 피어는 다이얼 업 링크를 정의하는 구성 파일과 거의 동일한 구성 파일을 사용합니다.

전용 회선을 통해 통신을 시작하기 위해 발생하는 프로세스는 다음과 같습니다.

1. 각 피어 시스템이 부트 프로세스 또는 다른 관리 스크립트의 일환으로 `pppd` 명령을 실행합니다.
2. 피어가 해당 PPP 구성 파일을 읽습니다.
3. 피어가 통신 매개변수를 협상합니다.
4. IP 링크가 설정됩니다.

## PPP 인증

인증은 개인의 자격을 확인하는 프로세스입니다. UNIX 로그인 절차는 간단한 인증 형태입니다.

1. `login` 명령이 사용자에게 이름 및 암호를 입력하라는 메시지를 표시합니다.
2. 그런 다음 `login`이 암호 데이터베이스에서 입력된 사용자 이름 및 암호를 조회하여 사용자에게 대해 인증을 시도합니다.
3. 데이터베이스에 해당 사용자 이름 및 암호가 포함되어 있으면 사용자가 인증되고 시스템 액세스 권한을 부여받게 됩니다. 데이터베이스에 해당 사용자 이름 및 암호가 포함되어 있지 않으면 사용자에게 대한 시스템 액세스 권한이 거부됩니다.

기본적으로 Solaris PPP 4.0은 기본 경로가 지정되어 있지 않은 시스템에서 인증을 요구하지 않습니다. 따라서 기본 경로가 없는 로컬 시스템은 원격 호출자를 인증하지 않습니다. 반대로, 시스템에 기본 경로가 정의되어 있으면 해당 시스템이 항상 원격 호출자를 인증합니다.



내 시스템에 대한 PPP 링크를 설정하려고 하는 호출자의 ID를 PPP 인증 프로토콜을 사용하여 확인할 수 있습니다. 반대로, 내 로컬 시스템이 호출자를 인증하는 피어를 호출해야 하는 경우 직접 PPP 인증 정보를 구성해야 합니다.

## 인증자 및 피인증자

PPP 링크의 호출 시스템이 **피인증자**로 간주됩니다. 이는 호출자가 자신의 ID를 원격 피어에게 증명해야 하기 때문입니다. 피어가 **인증자**로 간주됩니다. 인증자는 보안 프로토콜의 해당 PPP 파일에서 호출자의 ID를 조회하여 호출자를 인증하거나 인증하지 않습니다.

일반적으로 다이얼 업 링크에 대해 PPP 인증을 구성합니다. 호출이 시작되면 다이얼 아웃 시스템이 피인증자가 됩니다. 다이얼 인 서버는 인증자입니다. 이 서버에는 **암호** 파일의 형태로 데이터베이스가 있습니다. 이 파일에는 서버에 대한 PPP 링크를 설정할 수 있는 권한을 부여받은 모든 사용자의 목록이 나열됩니다. 이러한 사용자를 **신뢰할 수 있는 호출자**로 생각해 보십시오.

일부 다이얼 아웃 시스템에서는 원격 피어가 다이얼 아웃 시스템의 호출에 응답할 때 인증 정보를 제공해야 합니다. 그런 다음에는 역할이 바뀝니다. 즉, 원격 피어가 피인증자가 되고 다이얼 아웃 시스템이 인증자가 됩니다.

---

주 - PPP 4.0은 전용 회선 피어의 인증을 막지 않지만 전용 회선 링크에서는 인증이 자주 사용되지 않습니다. 전용 회선 계약의 특성상 회선 끝의 두 참가자 모두가 서로에게 알려져 있습니다. 일반적으로 두 참가자 모두를 신뢰할 수 있습니다. 그러나 PPP 인증은 관리가 그리 어렵지 않기 때문에 전용 회선에 대해 인증을 구현하는 것을 심각하게 고려해 보아야 합니다.

---

## PPP 인증 프로토콜

PPP 인증 프로토콜은 PAP(암호 인증 프로토콜) 및 CHAP(Challenge-Handshake 인증 프로토콜)입니다. 각 프로토콜은 로컬 시스템에 연결할 수 있는 각 호출자에 대해 ID 정보가 포함된 **암호** 데이터베이스 또는 **보안 자격 증명**을 사용합니다. PAP에 대한 자세한 내용은 [486 페이지 “PAP\(암호 인증 프로토콜\)”](#)를 참조하십시오. CHAP 설명은 [489 페이지 “CHAP\(Challenge-Handshake 인증 프로토콜\)”](#)를 참조하십시오.

## PPP 인증을 사용하는 이유

PPP 링크에서는 반드시 인증을 제공하지 않아도 됩니다. 또한 인증을 통해 피어를 신뢰할 수 있는지 확인할 수 있지만 PPP 인증은 데이터의 기밀성을 보장하지 않습니다. 기밀성 보장을 위해 IPsec, PGP, SSL, Kerberos 및 Secure Shell과 같은 암호화 소프트웨어를 사용하십시오.

주 - Solaris PPP 4.0은 RFC 1968에 기술되어 있는 PPP ECP(암호화 제어 프로토콜)를 구현하지 않습니다.

다음과 같은 경우 PPP 인증을 구현할 수 있습니다.

- 회사가 공개 교환 전화 네트워크를 통해 사용자로부터의 수신 호출을 받는 경우
- 회사 보안 정책상 원격 사용자가 회사 방화벽을 통해 네트워크에 액세스하거나 보안 트랜잭션에 참여할 때 인증 자격 증명을 제공해야 하는 경우
- 호출자를 /etc/passwd, NIS, LDAP 또는 PAM과 같은 표준 UNIX 암호 데이터베이스에 대해 인증하려는 경우. 이 시나리오의 경우 PAP 인증을 사용합니다.
- 회사의 다이얼 인 서버도 네트워크의 인터넷 연결을 제공하는 경우. 이 시나리오의 경우 PAP 인증을 사용합니다.
- 링크의 각 끝에 있는 시스템 또는 네트워크에서 직렬 회선은 암호 데이터베이스보다 덜 안전합니다. 이 시나리오의 경우 CHAP 인증을 사용합니다.

## PPPoE를 통한 DSL 사용자 지원

많은 네트워크 공급자와 채택 근무 사용자가 DSL(디지털 가입자 회선) 기술을 사용하여 빠른 네트워크 액세스를 제공합니다. DSL 사용자를 지원하기 위해 Solaris PPP 4.0에는 PPPoE(PPP over Ethernet) 기능이 포함되어 있습니다. PPPoE 기술을 사용하면 여러 호스트가 하나의 이더넷 링크를 통해 하나 이상의 대상으로 PPP 세션을 실행할 수 있습니다.

다음 중 하나에 해당되는 경우 PPPoE를 사용해야 합니다.

- DSL 사용자(본인도 포함될 수 있음)를 지원합니다. DSL 서비스 공급자가 DSL 회선을 통해 서비스를 받으려는 사용자에게 PPPoE 터널을 구성할 것을 요청할 수 있습니다.
- 사이트가 고객에게 PPPoE를 제공하려는 ISP입니다.

이 절에서는 PPPoE와 관련된 용어를 소개하고 기본 PPPoE 토폴로지에 대한 개요를 제공합니다.

## PPPoE 개요

PPPoE는 RedBack Networks의 독점 프로토콜입니다. PPPoE는 다른 버전의 표준 PPP가 아니라 검색 프로토콜입니다. PPPoE 시나리오에서는 먼저 PPP 통신을 시작하는 시스템이 PPPoE를 실행하는 피어를 찾거나 **발견**해야 합니다. PPPoE 프로토콜은 이더넷 브로드캐스트 패킷을 사용하여 피어를 찾습니다.

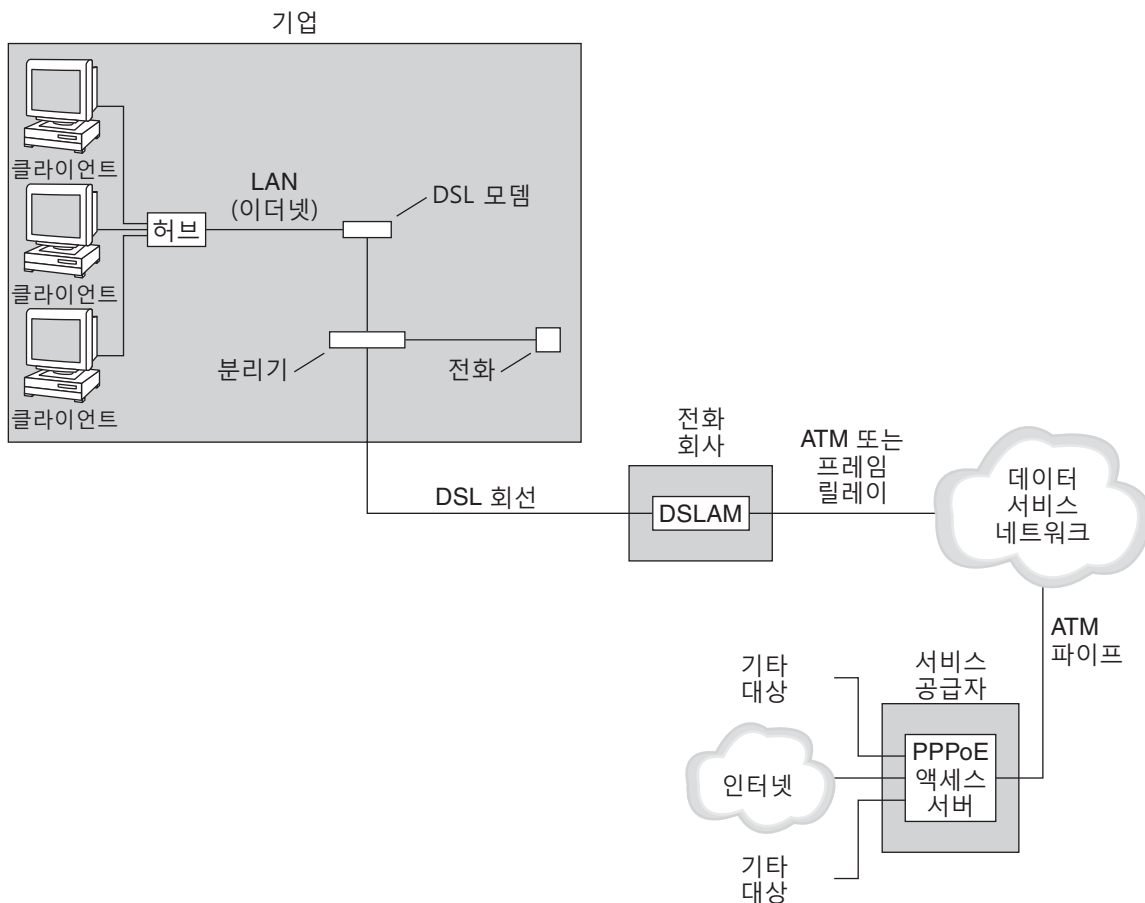
검색 프로세스 후에 PPPoE는 시작 호스트 또는 **PPPoE 클라이언트**에서 피어 또는 **PPPoE 액세스 서버**까지 이더넷 기반 터널을 설정합니다. **터널링**은 한 프로토콜을 기반으로 다른 프로토콜을 실행하는 방법입니다. Solaris PPP 4.0은 PPPoE를 사용하여 PPP over

Ethernet IEEE 802.2를 터널링합니다(둘 다 데이터 링크 프로토콜임). 결과 PPP 연결은 PPPoE 클라이언트와 액세스 서버 간의 전용 링크처럼 동작합니다. PPPoE에 대한 자세한 내용은 [494 페이지 “DSL 지원을 위해 PPPoE 터널 만들기”](#)를 참조하십시오.

## PPPoE 구성의 각 부분

PPPoE 구성에는 다음 그림에서 볼 수 있는 것과 같이 소비자, 전화 회사 및 서비스 공급자라는 세 참가자가 포함됩니다.

그림 15-4 PPPoE 터널의 참가자



## PPPoE 소비자

시스템 관리자는 소비자의 PPPoE 구성을 지원해야 할 수 있습니다. 일반적인 한 PPPoE 소비자 유형에는 DSL 회선을 통해 PPPoE를 실행해야 하는 사용자가 있습니다. 다른 PPPoE 소비자에는 이전 그림에서 볼 수 있는 것과 같이 직원이 PPPoE 터널을 실행할 수 있는 DSL 회선을 구입하는 회사가 있습니다.

기업 사용자는 고속 DSL 장치를 통해 여러 호스트에 PPP 통신을 제공하기 위해 주로 PPPoE를 사용합니다. 단일 PPPoE 클라이언트에 개별 **DSL 모뎀**이 있는 경우가 많습니다. 또는 허브에 있는 클라이언트 그룹이 이더넷 회선으로 역시 허브에 연결되어 있는 DSL 모뎀을 공유할 수 있습니다.

---

**주**-DSL 장치는 원칙적으로 모뎀이 아니라 브릿지입니다. 그러나 보통 이러한 장치를 모뎀이라고 지칭하므로 이 설명서에서는 “DSL 모뎀”이라는 용어를 사용합니다.

---

PPPoE는 DSL 모뎀에 연결되어 있는 이더넷 회선 상의 터널을 통해 PPP를 실행합니다. 해당 회선은 분리기에 연결되어 있고, 분리는 전화선에 연결됩니다.

## 전화 회사의 PPPoE

전화 회사는 PPPoE 시나리오의 중간 계층입니다. 전화 회사는 전화선을 통해 받는 신호를 **DSLAM(Digital Subscriber Line Access Multiplexer)**이라는 장치를 사용하여 분리합니다. DSLAM은 신호를 별도의 전화선, 전화 서비스용 아날로그 전화선 및 PPPoE용 디지털 전화선으로 분리합니다. DSLAM에서 디지털 전화선은 터널을 ATM 데이터 네트워크를 통해 ISP로 확장합니다.

## 서비스 공급자의 PPPoE

ISP는 브릿지를 통해 ATM 데이터 네트워크에서 PPPoE 전송을 받습니다. ISP에서는 PPPoE를 실행하는 액세스 서버가 PPP 링크의 피어 역할을 합니다. 액세스 서버는 기능 면에서 **그림 15-2**에서 소개된 다이얼 인 서버와 매우 유사하지만 액세스 서버는 모뎀을 사용하지 않습니다. 액세스 서버는 개별 PPPoE 세션을 일반 IP 트래픽(예: 인터넷 액세스)으로 변환합니다.

ISP의 시스템 관리자는 액세스 서버를 구성하고 유지 관리해야 할 수 있습니다.

## PPPoE 터널의 보안

PPPoE 터널은 본질적으로 안전하지 않습니다. PAP 또는 CHAP를 사용하여 터널을 통해 실행되는 PPP 링크에 대한 사용자 인증을 제공할 수 있습니다.

## PPP 링크 계획(작업)

PPP 링크를 설정하려면 여러 가지 별개의 작업(계획 작업 및 PPP와 관련되지 않은 기타 작업 포함)을 수행해야 합니다. 이 장에서는 가장 일반적인 PPP 링크, 인증 및 PPPoE를 계획하는 방법에 대해 설명합니다.

16 장, “PPP 링크 계획(작업)” 다음에 오는 작업 장에는 특정 링크를 설정하는 방법을 보여주기 위해 샘플 구성이 사용됩니다. 이러한 샘플 구성은 이 장에서 소개됩니다.

다음과 같은 항목을 다룹니다.

- 390 페이지 “다이얼 업 PPP 링크 계획”
- 393 페이지 “전용 회선 링크 계획”
- 395 페이지 “링크에서 인증 계획”
- 400 페이지 “PPPoE 터널을 통한 DSL 지원 계획”

## 전반적인 PPP 계획(작업 맵)

링크를 설정하려면 PPP에 계획 작업이 필요합니다. 또한 PPPoE 터널링을 사용하려는 경우에는 먼저 PPP 링크를 설정한 다음 터널링을 제공해야 합니다. 다음 작업 맵에는 이 장에 설명되어 있는 대규모 계획 작업이 나열되어 있습니다. 구성할 링크 유형에 대해 일반적인 작업만 사용해야 할 수도 있습니다. 또는 링크, 인증 및 PPPoE에 대한 작업이 필요할 수 있습니다.

표 16-1 PPP 계획 작업 맵

작업	설명	수행 방법
다이얼 업 PPP 링크 계획	다이얼 아웃 시스템 또는 다이얼 인 서버를 설정하는 데 필요한 정보 수집	390 페이지 “다이얼 업 PPP 링크 계획”
전용 회선 링크 계획	전용 회선에서 클라이언트를 설정하는 데 필요한 정보 수집	393 페이지 “전용 회선 링크 계획”

표 16-1 PPP 계획 작업 맵 (계속)

작업	설명	수행 방법
PPP 링크에서 인증 계획	PPP 링크에서 PAP 또는 CHAP 인증을 구성하는 데 필요한 정보 수집	395 페이지 “링크에서 인증 계획”
PPPoE 터널 계획	PPP 링크를 실행할 수 있는 PPPoE 터널을 설정하는 데 필요한 정보 수집	400 페이지 “PPPoE 터널을 통한 DSL 지원 계획”

## 다이얼업 PPP 링크 계획

다이얼업 링크는 가장 일반적으로 사용되는 PPP 링크입니다. 이 절에는 다음과 같은 정보가 포함되어 있습니다.

- 다이얼업 링크에 대한 계획 정보
- 17 장, “다이얼업 PPP 링크 설정(작업)”에 사용될 샘플 링크에 대한 설명

일반적으로 다이얼업 PPP 링크, 다이얼 아웃 시스템 또는 다이얼 인 서버의 한쪽 끝에서만 시스템을 구성합니다. 다이얼업 PPP에 대한 소개는 379 페이지 “다이얼업 PPP 개요”를 참조하십시오.

## 다이얼 아웃 시스템을 설정하기 전에

다이얼 아웃 시스템을 구성하기 전에 다음 표에 나열된 정보를 수집하십시오.

주 - 이 절의 계획 정보에는 인증 또는 PPPoE에 대해 수집할 정보는 포함되어 있지 않습니다. 인증 계획에 대한 자세한 내용은 395 페이지 “링크에서 인증 계획”을 참조하십시오. PPPoE 계획은 400 페이지 “PPPoE 터널을 통한 DSL 지원 계획”을 참조하십시오.

표 16-2 다이얼 아웃 시스템에 대한 정보

정보	작업
최대 모뎀 속도	모뎀 제조업체가 제공하는 설명서를 참조하십시오.
모뎀 연결 명령(AT 명령)	모뎀 제조업체가 제공하는 설명서를 참조하십시오.
링크의 다른 쪽 끝에서 다이얼 인 서버에 사용할 이름	다이얼 인 서버를 식별하는 데 도움이 되는 이름을 만듭니다.
다이얼 인 서버에 필요했던 로그인 절차	다이얼 인 서버의 관리자에게 문의하거나 다이얼 인 서버가 ISP에 있는 경우 ISP 설명서를 참조하십시오.

## 다이얼 인 서버를 설정하기 전에

다이얼 인 서버를 구성하기 전에 다음 표에 나열된 정보를 수집하십시오.

주 - 이 절의 계획 정보에는 인증 또는 PPPoE에 대해 수집할 정보는 포함되어 있지 않습니다. 인증 계획에 대한 자세한 내용은 395 페이지 “링크에서 인증 계획”을 참조하십시오. PPPoE 계획은 400 페이지 “PPPoE 터널을 통한 DSL 지원 계획”을 참조하십시오.

표 16-3 다이얼 인 서버에 대한 정보

정보	작업
최대 모뎀 속도	모뎀 제조업체가 제공하는 설명서를 참조하십시오.
다이얼 인 서버를 호출할 수 있는 사람의 사용자 이름	잠재적 사용자의 홈 디렉토리를 설정하기 전에 414 페이지 “다이얼 인 서버의 사용자를 구성하는 방법”에 설명된 대로 해당 사용자의 이름을 얻으십시오.
PPP 통신을 위한 전용 IP 주소	회사의 IP 주소 위임 담당자에게서 주소를 얻으십시오.

## 다이얼 업 PPP 구성의 예

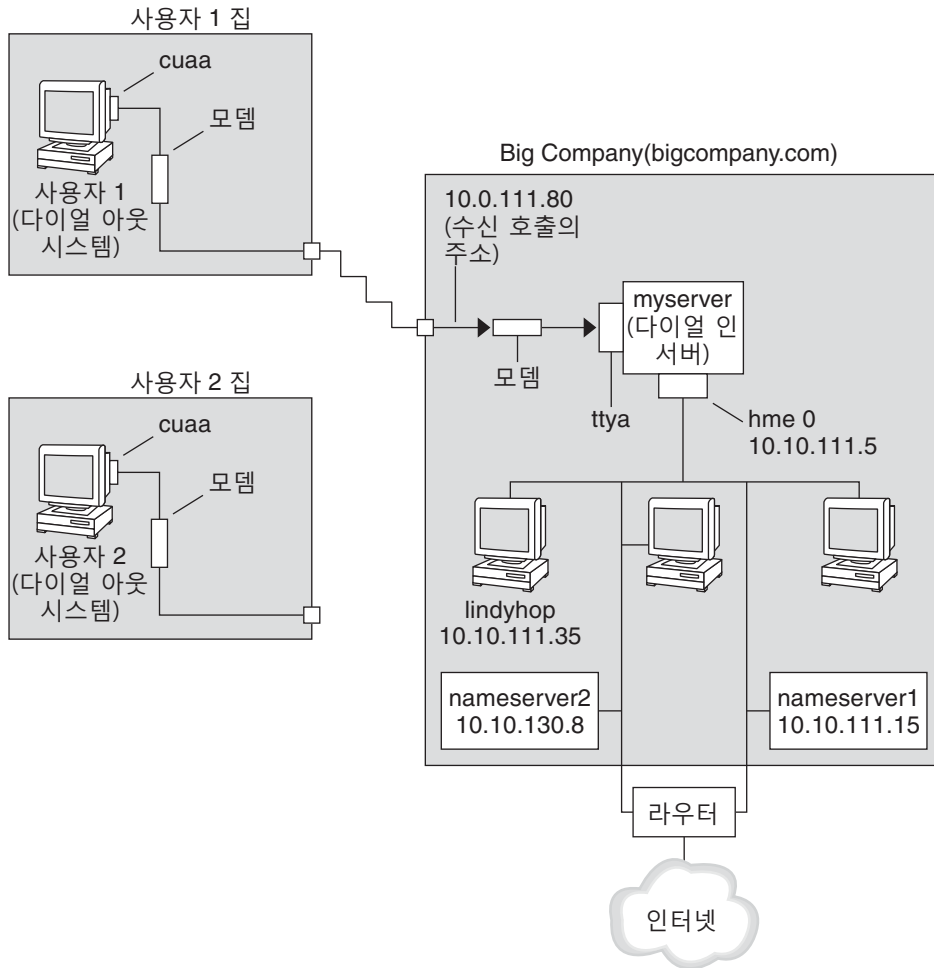
17 장, “다이얼 업 PPP 링크 설정(작업)”에서 소개될 작업은 직원들이 한 주에 몇 번은 집에서 일할 수 있도록 하려는 소기업의 요구 사항을 실행에 옮깁니다. 일부 직원의 경우 홈 시스템에 Oracle Solaris OS가 필요합니다. 또한 이러한 작업자는 회사 인트라넷에서 자신의 시스템으로 원격 로그인해야 합니다.

작업은 다음 기능과의 기본적인 다이얼 업 링크를 설정합니다.

- **다이얼 아웃** 시스템은 회사 인트라넷을 호출해야 하는 직원의 집에 있습니다.
- **다이얼 인** 서버는 회사 인트라넷에서 직원으로부터의 수신 호출을 받도록 구성된 시스템입니다.
- 다이얼 아웃 시스템을 인증하는 데에는 UNIX 스타일의 로그인이 사용됩니다. 회사의 보안 정책상 더 강력한 Solaris PPP 4.0 인증 방법이 필요하지 않습니다.

다음 그림에서는 17 장, “다이얼 업 PPP 링크 설정(작업)”에서 설정된 링크를 보여줍니다.

그림 16-1 샘플 다이얼 업 링크



이 그림에서는 원격 호스트가 전화선을 통해 모뎀을 사용하여 Big Company의 인트라넷으로 다이얼 아웃합니다. 다른 호스트도 Big Company로 다이얼 아웃하도록 구성되어 있지만 현재 비활성 상태입니다. 원격 사용자로부터의 호출은 Big Company에 있는 다이얼 인 서버에 연결된 모뎀이 받는 순서대로 응답됩니다. PPP 연결이 피어 간에 설정됩니다. 그러면 다이얼 아웃 시스템이 인트라넷에 있는 호스트 시스템에 원격 로그인할 수 있습니다.



## 다이얼 업 PPP에 대한 추가 정보

다음을 참조하십시오.

- 다이얼 아웃 시스템을 설정하려면 표 17-2를 참조하십시오.
- 다이얼 인 시스템을 설정하려면 표 17-3을 참조하십시오.
- 다이얼 업 링크에 대한 개요를 보려면 379 페이지 “다이얼 업 PPP 개요”를 참조하십시오.
- PPP 파일 및 명령에 대한 자세한 내용을 보려면 465 페이지 “파일 및 명령줄에서 PPP 옵션 사용”을 참조하십시오.

## 전용 회선 링크 계획

전용 회선 링크를 설정하려면 공급자에게서 임대 받은 교환 또는 비교환 서비스의 한쪽 끝에서 피어를 구성해야 합니다.

이 절에는 다음과 같은 정보가 포함되어 있습니다.

- 전용 회선 링크에 대한 계획 정보
- 그림 16-2에 나와 있는 샘플 링크에 대한 설명

전용 회선 링크에 대한 소개는 382 페이지 “전용 회선 PPP 개요”를 참조하십시오. 전용 회선 설정 작업은 18 장, “전용 회선 PPP 링크 설정(작업)”을 참조하십시오.

## 전용 회선 링크를 설정하기 전에

회사가 네트워크 공급자로부터 전용 회선 링크를 임대 받는 경우에는 일반적으로 링크의 본인 쪽 끝에 있는 시스템만 구성합니다. 링크의 다른 쪽 끝에 있는 피어는 다른 관리자가 유지 관리합니다. 이 사람은 회사의 원격 위치에 있는 시스템 관리자나 ISP에 있는 시스템 관리자일 수 있습니다.

### 전용 회선 링크에 필요한 하드웨어

링크의 본인 쪽 끝에는 링크 매체 외에 다음 하드웨어가 필요합니다.

- 시스템을 위한 동기 인터페이스
- 동기 장치(CSU/DSU)
- 시스템

일부 네트워크 공급자의 경우 라우터, 동기 인터페이스 및 CSU/DSU를 CPE(고객 태내 장치)의 일부로 포함합니다. 그러나 필요한 장비는 공급자와 현지 정부 제한에 따라 달라집니다. 네트워크 공급자가 필요한 장치에 대한 정보를 제공해 줄 수 있습니다(이 장비가 전용 회선에 제공되지 않은 경우).

## 전용 회선 링크에 대해 수집해야 하는 정보

로컬 피어를 구성하기 전에 다음 표에 나열된 항목을 수집해야 할 수 있습니다.

표 16-4 전용 회선 링크 계획

정보	작업
인터페이스의 장치 이름	인터페이스 카드 설명서를 참조하십시오.
동기 인터페이스 카드에 대한 구성 지침	인터페이스 카드 설명서를 참조하십시오. HSI/P 인터페이스를 구성하려면 이 정보가 필요합니다. 다른 유형의 인터페이스 카드를 구성해야 할 수도 있습니다.
(옵션) 원격 피어의 IP 주소	서비스 공급자 설명서를 참조하십시오. 또는 원격 피어의 시스템 관리자에게 문의하십시오. 이 정보는 두 피어 간에 IP 주소가 협상되지 않은 경우에만 필요합니다.
(옵션) 원격 피어의 이름	서비스 공급자 설명서를 참조하십시오. 또는 원격 피어의 시스템 관리자에게 문의할 수 있습니다.
(옵션) 링크의 속도	서비스 공급자 설명서를 참조하십시오. 또는 원격 피어의 시스템 관리자에게 문의할 수 있습니다.
(옵션) 원격 피어에 사용되는 압축	서비스 공급자 설명서를 참조하십시오. 또는 원격 피어의 시스템 관리자에게 문의할 수 있습니다.

## 전용 회선 링크 구성의 예

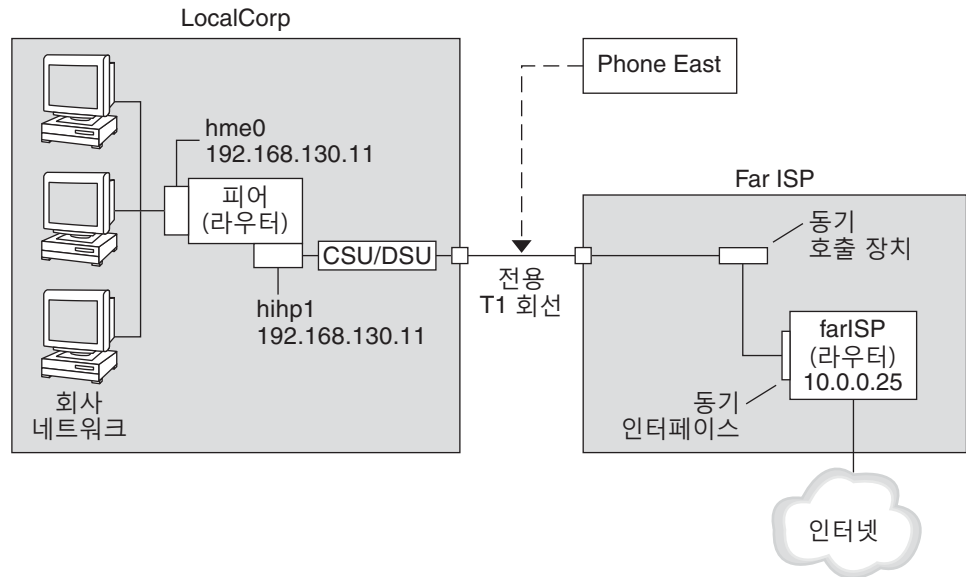
18 장, “전용 회선 PPP 링크 설정(작업)”의 작업에서는 직원에게 인터넷 액세스를 제공하려는 중소 규모 조직(LocalCorp)의 목표를 달성하는 방법을 보여줍니다. 현재는 직원의 컴퓨터가 개인 회사 인트라넷에서 연결되어 있습니다.

LocalCorp는 빠른 트랜잭션을 필요로 하며 인터넷에 있는 많은 자원에 액세스해야 합니다. 이 기업은 자체 전용 회선을 설정할 수 있게 해 주는 서비스 공급자인 Far ISP와 계약을 맺고 있으며, 전화 회사인 Phone East로부터 T1 회선도 임대 받고 있습니다. Phone East는 전용 회선을 LocalCorp와 Far ISP 사이에 놓았습니다. 그리고 Phone East는 이미 LocalCorp로 구성되어 있는 CSU/DSU를 제공합니다.

작업은 다음 특징을 가지는 전용 회선 링크를 설정합니다.

- LocalCorp는 전용 회선을 통해 인터넷 상의 호스트로 패킷을 전달하는 게이트웨이 라우터로 시스템을 설정했습니다.
- Far ISP도 고객의 전용 회선이 연결되는 라우터로 피어를 설정했습니다.

그림 16-2 전용 회선 구성의 예



그림에서는 LocalCorp에서 PPP에 대해 라우터가 설정되어 있습니다. 이 라우터는 해당 hme0 인터페이스를 통해 회사 인트라넷에 연결됩니다. 두번째 연결은 시스템의 HSI/P 인터페이스(hihp1)를 통해 CSU/DSU 디지털 장치로 이루어집니다. 그런 다음 CSU/DSU가 설치된 전용 회선에 연결됩니다. LocalCorp의 관리자가 HSI/P 인터페이스와 PPP 파일을 구성합니다. 그런 다음 관리자는 /etc/init.d/pppd를 입력하여 LocalCorp와 Far ISP 간에 링크를 시작합니다.

## 전용 회선에 대한 추가 정보

다음을 참조하십시오.

- 18 장, “전용 회선 PPP 링크 설정(작업)”
- 382 페이지 “전용 회선 PPP 개요”

## 링크에서 인증 계획

이 절에는 PPP 링크에서 인증을 제공하기 위한 계획 정보가 포함되어 있습니다. 사이트에서 PPP 인증을 구현하는 작업은 19 장, “PPP 인증 설정(작업)”에 있습니다.

PPP는 PAP와 CHAP라는 두 가지 유형의 인증을 제공합니다. PAP는 486 페이지 “PAP(암호 인증 프로토콜)”에 자세히 설명되어 있고, CHAP는 489 페이지 “CHAP(Challenge-Handshake 인증 프로토콜)”에 설명되어 있습니다.

링크에서 인증을 설정하기 전에 사이트의 보안 정책에 가장 잘 맞는 인증 프로토콜이 무엇인지 선택해야 합니다. 그런 다음 다이얼 인 시스템 또는 호출자의 다이얼 아웃 시스템이나 두 시스템 유형 모두에 대해 암호 파일 및 PPP 구성 파일을 설정합니다. 사이트에 적합한 인증 프로토콜을 선택하는 방법에 대한 자세한 내용은 [385 페이지 “PPP 인증을 사용하는 이유”](#)를 참조하십시오.

이 절에는 다음과 같은 정보가 포함되어 있습니다.

- PAP 및 CHAP 인증 모두에 대한 계획 정보
- [그림 16-3](#) 및 [그림 16-4](#)에 나와 있는 샘플 인증 시나리오에 대한 설명

인증 설정 작업은 [19 장, “PPP 인증 설정\(작업\)”](#)을 참조하십시오.

## PPP 인증을 설정하기 전에

사이트에서 인증을 설정하는 작업은 전반적인 PPP 전략의 필수 요소가 되어야 합니다. 인증을 구현하기 전에 하드웨어를 조립하고, 소프트웨어를 구성하고, 링크를 테스트해야 합니다.

표 16-5 인증 구성 전의 필수 조건

정보	수행 방법
다이얼 업 링크 구성 작업	<a href="#">17 장, “다이얼 업 PPP 링크 설정(작업)”</a>
링크 테스트 작업	<a href="#">21 장, “일반적인 PPP 문제 해결(작업)”</a>
사이트의 보안 요구 사항	회사의 보안 정책입니다. 정책이 없는 경우 PPP 인증을 설정하면 보안 정책을 만들 수 있습니다.
사이트에서 PAP를 사용할지, 아니면 CHAP를 사용할지에 대한 제안 사항	<a href="#">385 페이지 “PPP 인증을 사용하는 이유”</a> 이러한 프로토콜에 대한 자세한 내용은 <a href="#">486 페이지 “링크에서 호출자 인증”</a> 을 참조하십시오.

## PPP 인증 구성의 예

이 절에는 [19 장, “PPP 인증 설정\(작업\)”](#)의 절차에 사용될 인증 시나리오의 예가 포함되어 있습니다.

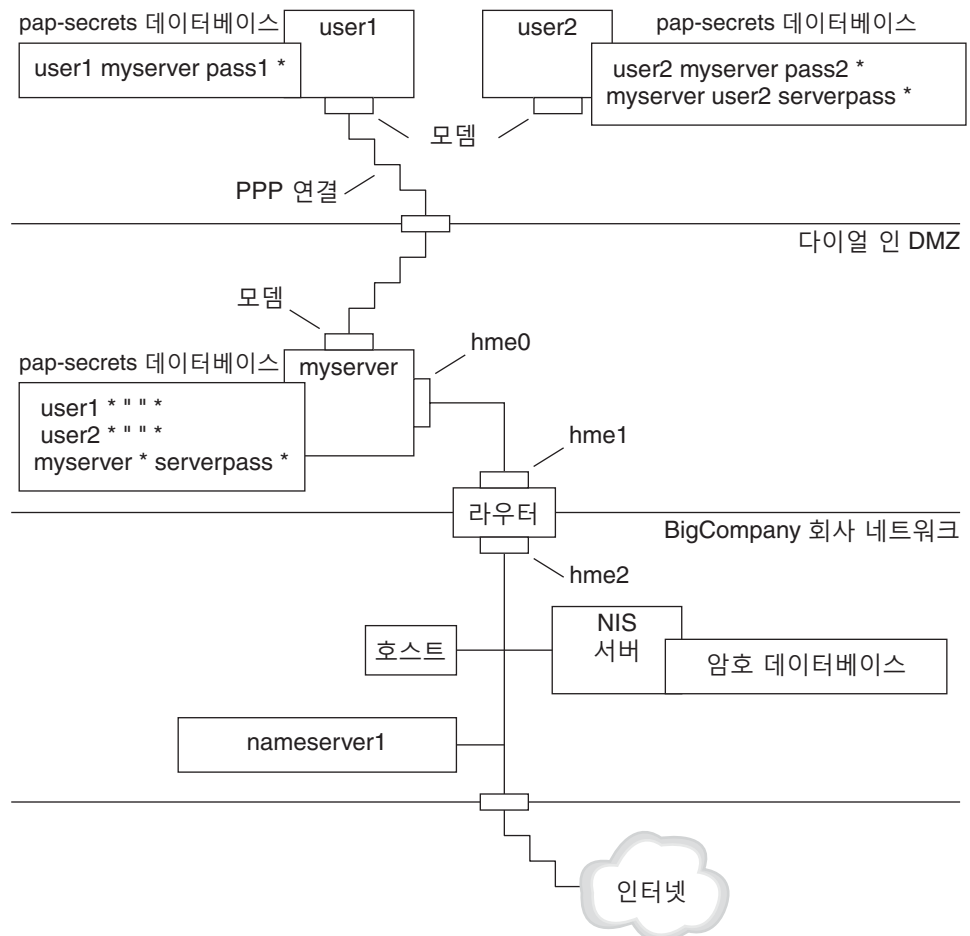
- [396 페이지 “PAP 인증을 사용한 구성의 예”](#)
- [398 페이지 “CHAP 인증을 사용한 구성의 예”](#)

## PAP 인증을 사용한 구성의 예

[426 페이지 “PAP 인증 구성”](#)의 작업에서는 PPP 링크를 통해 PAP 인증을 설정하는 방법을 보여줍니다. 절차에는 [391 페이지 “다이얼 업 PPP 구성의 예”](#)에 나오는 가상 회사 “Big Company”에 대해 만들어진 PAP 시나리오가 예로 사용됩니다.

Big Company는 사용자가 집에서 일할 수 있게 만드는 것을 목표로 하고 있습니다. 시스템 관리자는 다이얼 인 서버에 연결되는 직렬 회선을 위한 보안 솔루션을 구현할 계획을 가지고 있습니다. 과거에는 NIS 암호 데이터베이스를 사용하는 UNIX 스타일의 로그인인 Big Company의 네트워크에서 충분히 그 역할을 했습니다. 시스템 관리자는 PPP 링크를 통해 네트워크로 들어오는 호출에 UNIX 스타일의 인증 체계를 구현하려고 합니다. 이에 따라 관리자는 PAP 인증을 사용하는 다음 시나리오를 구현합니다.

그림 16-3 PAP 인증 시나리오의 예(재택 근무)



시스템 관리자는 나머지 회사 네트워크와 라우터로 분리되는 전용 다이얼 인 DMZ를 만듭니다. DMZ라는 용어는 군사 용어인 "비무장 지대"에서 유래한 것입니다. DMZ는 보안을 위해 설정된 격리 네트워크입니다. 일반적으로 DMZ에는 웹 서버, 익명 FTP 서버,

데이터베이스 및 모뎀 서버와 같이 회사가 공개하는 자원이 포함됩니다. 네트워크 디자인에는 종종 방화벽과 회사의 인터넷 연결 사이에 DMZ를 배치합니다.

그림 16-3에서 볼 수 있는 DMZ 점유자는 다이얼 인 서버 `myserver`와 라우터뿐입니다. 다이얼 인 서버를 사용하려면 호출자가 링크를 설정할 때 사용자 이름 및 암호를 포함한 PAP 자격 증명을 제공해야 합니다. 또한 다이얼 인 서버에는 PAP의 `login` 옵션이 사용됩니다. 따라서 호출자의 PAP 사용자 이름 및 암호가 다이얼 인 서버의 암호 데이터베이스에 있는 UNIX 사용자 이름 및 암호와 정확히 일치해야 합니다.

PPP 링크가 설정된 후에는 호출자의 패킷이 라우터로 전달됩니다. 라우터는 회사 네트워크 대상이나 인터넷 대상으로 전송을 전달합니다.

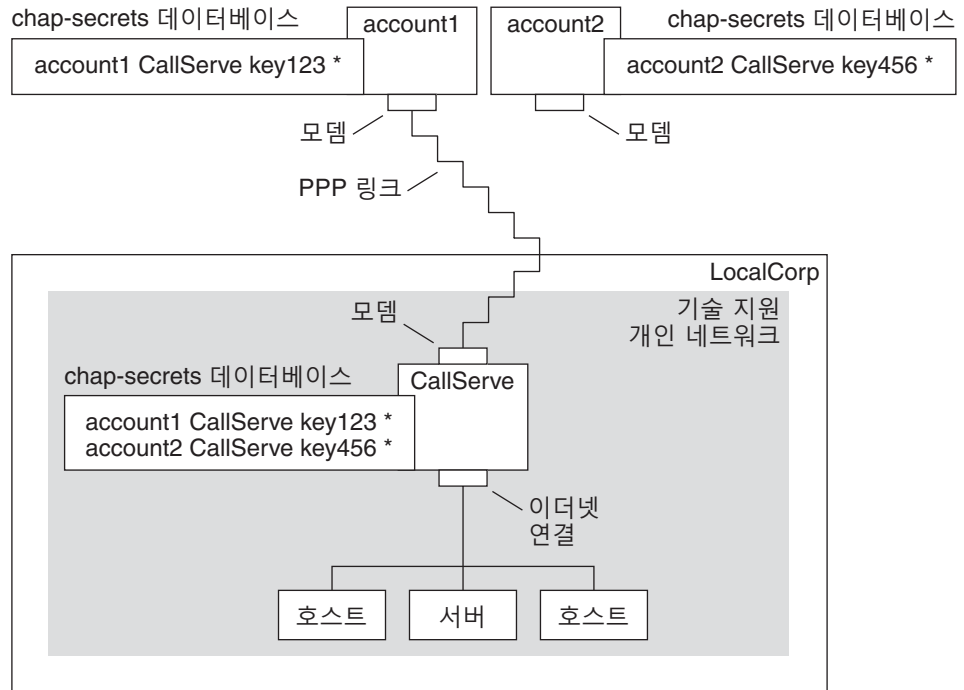
## CHAP 인증을 사용한 구성의 예

433 페이지 “CHAP 인증 구성”의 작업에서는 CHAP 인증을 설정하는 방법을 보여줍니다. 절차에는 394 페이지 “전용 회선 링크 구성의 예”에서 소개된 가상 회사 LocalCorp에 대해 만들어질 CHAP 시나리오가 예로 사용됩니다.

LocalCorp는 ISP에 대한 전용 회선을 통해 인터넷 연결을 제공합니다. LocalCorp의 기술 지원부는 상당한 네트워크 트래픽을 발생시킵니다. 따라서 기술 지원부에는 격리된 자체 개인 네트워크가 필요합니다. 이 부서의 현장 기술자는 광범위한 지역을 다니며, 문제 해결 정보를 얻기 위해 원격 위치에서 기술 지원 네트워크에 액세스해야 합니다. 개인 네트워크의 데이터베이스에 있는 민감한 정보를 보호하기 위해 원격 호출자는 인증을 받아 로그인 권한을 부여받아야 합니다.

이에 따라 시스템 관리자는 다이얼 업 PPP 구성을 위해 다음 CHAP 인증 시나리오를 구현합니다.

그림 16-4 CHAP 인증 시나리오의 예(개인 네트워크 호출)



기술 지원 네트워크에서 외부 세계로 연결되는 링크는 링크의 다이얼 인 서버의 끝에 연결되는 직렬 회선뿐입니다. 시스템 관리자는 각 현장 서비스 담당자의 랩탑 컴퓨터를 CHAP 보안이 적용된 PPP(CHAP 암호 포함)용으로 구성합니다. 다이얼 인 서버에 있는 chap-secrets 데이터베이스에는 기술 지원 네트워크를 호출할 수 있는 모든 시스템의 CHAP 자격 증명이 포함되어 있습니다.

## 인증에 대한 추가 정보

다음 중에서 선택하십시오.

- 426 페이지 “PAP 인증 구성”을 참조하십시오.
- 433 페이지 “CHAP 인증 구성”을 참조하십시오.
- 486 페이지 “링크에서 호출자 인증” 및 [pppd\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

# PPPoE 터널을 통한 DSL 지원 계획

공급자의 DSL 회선 및 고속 디지털 네트워크를 통해 PPP를 실행하기 위해 사이트에 PPPoE 터널링을 설정할 것을 요청하는 DSL 공급자도 있습니다. PPPoE의 개요는 [386 페이지 “PPPoE를 통한 DSL 사용자 지원”](#)을 참조하십시오.

PPPoE 터널에는 소비자, 전화 회사 및 ISP라는 세 참가자가 관련됩니다. 소비자(회사에 있는 PPPoE 클라이언트 또는 개인 소비자)를 위해 PPPoE를 구성하거나 ISP에 있는 서버에서 PPPoE를 구성합니다.

이 절에는 클라이언트와 액세스 서버 모두에서 PPPoE를 실행하기 위한 계획 정보가 포함되어 있습니다. 다음 항목을 다룹니다.

- PPPoE 호스트 및 액세스 서버에 대한 계획 정보
- [401 페이지 “PPPoE 터널 구성의 예”](#)에서 소개된 PPPoE 시나리오에 대한 설명

PPPoE 터널 설정 작업은 [20 장, “PPPoE 터널 설정\(작업\)”](#)을 참조하십시오.

## PPPoE 터널을 설정하기 전에

미리 구성 작업은 터널의 클라이언트측을 구성하는지, 아니면 서버측을 구성하는지에 따라 달라집니다. 어떤 경우든 관리자나 기업은 전화 회사와 계약을 맺어야 합니다. 전화 회사는 클라이언트용 DSL 회선과 어떤 형태로든 브리징을 제공하며 액세스 서버용 ATM 파이프를 제공할 수도 있습니다. 대부분의 계약에서 전화 회사는 사이트에서 장비를 조립합니다.

## PPPoE 클라이언트를 구성하기 전에

PPPoE 클라이언트 구현은 일반적으로 다음 장비로 구성됩니다.

- 개인용 컴퓨터 또는 개인이 사용하는 기타 시스템
- DSL 모뎀(일반적으로 전화 회사 또는 인터넷 액세스 공급자가 설치함)
- (옵션) 허브(기업 DSL 소비자와 같이 둘 이상의 클라이언트를 사용하는 경우)
- (옵션) 분리기(일반적으로 공급자가 설치함)

다양한 DSL 구성이 가능합니다. 구성은 사용자 또는 기업의 요구와 공급자가 제공하는 서비스에 따라 달라집니다.

표 16-6 PPPoE 클라이언트 계획

정보	작업
개인 또는 자신을 위해 홈 PPPoE 클라이언트를 설정하는 경우에는 PPPoE 범위 밖에 있는 모든 설정 정보를 가져와야 합니다.	필요한 설정 절차는 전화 회사나 ISP에 문의하십시오.



표 16-6 PPPoE 클라이언트 계획 (계속)

정보	작업
회사 사이트에서 PPPoE 클라이언트를 설정하는 경우에는 PPPoE 클라이언트 시스템을 지정받는 사용자의 이름을 수집해야 합니다. 원격 PPPoE 클라이언트를 구성하는 경우 홈 DSL 장비 추가에 대한 정보를 사용자에게 제공해 주어야 할 수 있습니다.	권한이 부여된 사용자 목록은 회사 관리부에 문의하십시오.
PPPoE 클라이언트에서 사용 가능한 인터페이스가 무엇인지 알아봅니다.	인터페이스 이름을 가져오려면 각 시스템에서 <code>ipadm show-addr</code> 명령을 실행합니다.
(옵션) PPPoE 클라이언트의 암호를 가져옵니다.	사용자에게 선호하는 암호가 무엇인지 물어보거나 사용자에게 암호를 지정합니다. 이 암호는 UNIX 로그인인 아니라 링크 인증에 사용됩니다.

## PPPoE 서버를 구성하기 전에

PPPoE 액세스 서버를 계획할 때는 데이터 서비스 네트워크에 대한 연결을 제공하는 전화 회사와 협력해야 합니다. 전화 회사는 해당 회선(일반적으로 ATM 파이프)을 사이트에 설치하고 어떤 형태로든 액세스 서버에 브리징을 제공합니다. 관리자는 회사가 제공하는 서비스에 액세스하는 이더넷 인터페이스를 구성해야 합니다. 예를 들어, 관리자는 인터넷 액세스용 인터페이스와 전화 회사 브릿지의 이더넷 인터페이스를 구성해야 합니다.

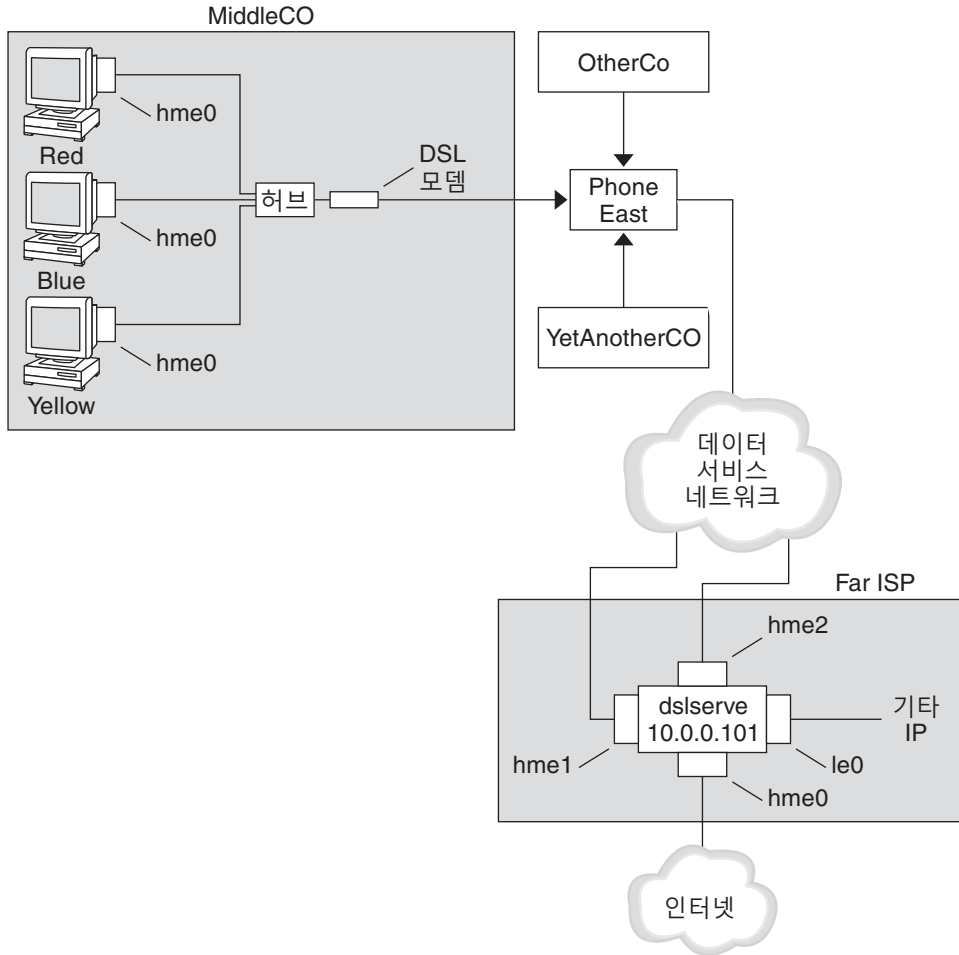
표 16-7 PPPoE 액세스 서버 계획

정보	작업
데이터 서비스 네트워크의 회선에 사용되는 인터페이스	인터페이스를 식별하려면 <code>ipadm show-addr</code> 명령을 실행합니다.
PPPoE 서버에서 제공할 서비스 유형	관리부와 네트워크 기획자에게 요구 사항 및 제안 사항이 있는지 문의하십시오.
(옵션) 소비자에게 제공할 서비스 유형	관리부와 네트워크 기획자에게 요구 사항 및 제안 사항이 있는지 문의하십시오.
(옵션) 원격 클라이언트의 호스트 이름 및 암호	네트워크 기획자와 사이트의 다른 사람에게 계약 협상 담당자가 누구인지 문의하십시오. 호스트 이름 및 암호는 UNIX 로그인인 아니라 PAP 또는 CHAP 인증에 사용됩니다.

## PPPoE 터널 구성의 예

이 절에는 20 장, “PPPoE 터널 설정(작업)”의 작업에 대한 실례로 사용된 PPPoE 터널의 예가 포함되어 있습니다. 그림에는 터널의 모든 참가자가 표시되어 있지만 관리자는 클라이언트측 또는 서버측 중 한쪽 끝만 관리합니다.

그림 16-5 PPPoE 터널의 예



샘플에서는 MiddleCo가 직원들에게 고속 인터넷 액세스를 제공하려고 합니다. MiddleCo는 Phone East로부터 DSL 패키지를 구입하고 Phone East는 서비스 공급자인 Far ISP와 계약을 맺습니다. Far ISP는 Phone East로부터 DSL을 구입하는 고객에게 인터넷 및 기타 IP 서비스를 제공합니다.

## PPPoE 클라이언트 구성의 예

MiddleCo가 사이트에 하나의 DSL 회선을 제공하는 Phone East로부터 패키지를 구입합니다. 패키지에는 MiddleCo의 PPPoE 클라이언트용 ISP에 대한 인증된 전용 연결이 포함되어 있습니다. 시스템 관리자가 잠재적 PPPoE 클라이언트를 허브에 케이블로 연결합니다. Phone East 기술자가 허브를 자사 DSL 장비에 케이블로 연결합니다.

## PPPoE 서버 구성의 예

FarISP가 Phone East와 맺은 사업 계약을 이행하기 위해 FarISP의 시스템 관리자가 액세스 서버 `dslserve`를 구성합니다. 이 서버에는 다음과 같은 네 가지 인터페이스가 있습니다.

- `eri0` – 로컬 네트워크에 연결되는 주 네트워크 인터페이스
- `hme0` – FarISP가 고객에게 인터넷 서비스를 제공하는 데 사용하는 인터페이스
- `hme1` – 인증된 PPPoE 터널을 위해 MiddleCo가 계약한 인터페이스
- `hme2` – PPPoE 터널을 위해 다른 고객이 계약한 인터페이스

## PPPoE에 대한 추가 정보

다음 중에서 선택하십시오.

- 440 페이지 “PPPoE 클라이언트 설정”을 참조하십시오.
- 443 페이지 “PPPoE 액세스 서버 설정”을 참조하십시오.
- 494 페이지 “DSL 지원을 위해 PPPoE 터널 만들기”, `pppoed(1M)`, `pppoec(1M)` 및 `sppptun(1M)` 매뉴얼 페이지를 참조하십시오.



## 다이얼 업 PPP 링크 설정(작업)

이 장에서는 가장 일반적인 PPP 링크인 다이얼 업 링크를 구성하는 작업에 대해 설명합니다. 주요 항목은 다음과 같습니다.

- 406 페이지 “다이얼 아웃 시스템 구성”
- 412 페이지 “다이얼 인 서버 구성”
- 416 페이지 “다이얼 인 서버 호출”

### 다이얼 업 PPP 링크를 설정하는 주요 작업(작업 맵)

모뎀을 구성하고, 네트워크 데이터베이스 파일을 수정하고, 표 22-1에 설명된 PPP 구성 파일을 수정하여 다이얼 업 PPP 링크를 설정합니다.

다음 표에는 다이얼 업 PPP 링크의 양쪽을 구성하는 주요 작업이 나열되어 있습니다. 일반적으로는 링크의 한쪽 끝(다이얼 아웃 시스템 또는 다이얼 인 서버)만 구성합니다.

표 17-1 다이얼 업 PPP 링크 설정 작업 맵

작업	설명	수행 방법
1. 미리 구성 정보 수집	링크를 설정하기 전에 필요한 데이터(예: 피어 호스트 이름, 대상 전화 번호 및 모뎀 속도)를 수집합니다.	390 페이지 “다이얼 업 PPP 링크 계획”
2. 다이얼 아웃 시스템 구성	링크를 통해 호출 작업을 수행하는 시스템에서 PPP를 설정합니다.	406 페이지 “다이얼 아웃 시스템 구성 작업(작업 맵)”
3. 다이얼 인 서버 구성	수신 호출을 받는 시스템에서 PPP를 설정합니다.	412 페이지 “다이얼 인 서버 구성 작업(작업 맵)”
4. 다이얼 인 서버 호출	pppd 명령을 입력하여 통신을 시작합니다.	416 페이지 “다이얼 인 서버를 호출하는 방법”

# 다이얼 아웃 시스템 구성

이 절의 작업에서는 다이얼 아웃 시스템을 구성하는 방법에 대해 설명합니다. 작업에는 [그림 16-1](#)에서 소개된 집에서 다이얼 인 시나리오가 예로 사용됩니다. 잠재적 사용자에게 시스템을 전달하기 전에 회사에서 해당 작업을 수행할 수 있습니다. 또는 경험이 많은 사용자에게 홈 시스템 설정 지침을 제공해 줄 수 있습니다. 다이얼 아웃 시스템을 설정하는 모든 사용자에게는 해당 시스템에 대한 루트 권한이 있어야 합니다.

## 다이얼 아웃 시스템 구성 작업(작업 맵)

표 17-2 다이얼 아웃 시스템 설정 작업 맵

작업	설명	수행 방법
1. 미리 구성 정보 수집	링크를 설정하기 전에 필요한 데이터(예: 피어 호스트 이름, 대상 전화 번호 및 모뎀 속도)를 수집합니다.	390 페이지 “다이얼 업 PPP 링크 계획”
2. 모뎀 및 직렬 포트 구성	모뎀 및 직렬 포트를 설정합니다.	407 페이지 “모뎀 및 직렬 포트를 구성하는 방법(다이얼 아웃 시스템)”
3. 직렬 회선 통신 구성	직렬 회선을 통한 전송의 특징을 구성합니다.	408 페이지 “직렬 회선을 통해 통신을 정의하는 방법”
4. 다이얼 아웃 시스템과 피어 간의 대화 정의	채트 스크립트를 만들 때 사용할 통신 데이터를 수집합니다.	409 페이지 “피어 호출 명령을 만드는 방법”
5. 특정 피어에 대한 정보 구성	개별 다이얼 인 서버를 호출할 PPP 옵션을 구성합니다.	410 페이지 “개별 피어를 사용하여 연결을 정의하는 방법”
6. 피어 호출	pppd 명령을 입력하여 통신을 시작합니다.	416 페이지 “다이얼 인 서버를 호출하는 방법”

## 다이얼 업 PPP 템플리트 파일

Solaris PPP 4.0은 템플리트 파일을 제공합니다. 각 템플리트에는 특정 PPP 구성 파일에 대한 일반적인 옵션이 포함되어 있습니다. 다음 표에는 다이얼 업 링크를 설정하는 데 사용할 수 있는 샘플 템플리트와 해당 Solaris PPP 4.0 파일이 나열되어 있습니다.

템플리트 파일	PPP 구성 파일	수행 방법
/etc/ppp/options.tpl	/etc/ppp/options	470 페이지 “/etc/ppp/options.tpl 템플리트”
/etc/ppp/options.ttya.tpl	/etc/ppp/options.ttyname	472 페이지 “options.ttya.tpl 템플리트 파일”

템플리트 파일	PPP 구성 파일	수행 방법
/etc/ppp/myisp-chat.tpl	채트 스크립트를 포함할 파일(원하는 이름 지정)	478 페이지 “/etc/ppp/myisp-chat.tpl 채트 스크립트 템플리트”
/etc/ppp/peers/myisp.tpl	/etc/ppp/peers/peer-name	475 페이지 “/etc/ppp/peers/myisp.tpl 템플리트 파일”

템플리트 파일 중 하나를 사용할 경우 템플리트의 이름을 해당 PPP 구성 파일의 이름으로 변경해야 합니다. 유일한 예외는 채트 파일 템플리트 /etc/ppp/myisp-chat.tpl입니다. 채트 스크립트의 경우 원하는 이름을 선택할 수 있습니다.

## 다이얼 아웃 시스템에서 장치 구성

다이얼 아웃 PPP 시스템을 설정하는 첫번째 작업은 직렬 회선에서 장치(모뎀 및 직렬 포트)를 구성하는 것입니다.

주 - 모뎀에 적용되는 작업은 일반적으로 ISDN TA에 적용됩니다.

후속 절차를 수행하려면 다음 작업을 완료해야 합니다.

- 다이얼 아웃 시스템에 Oracle Solaris 릴리스 설치
- 최적의 모뎀 속도 결정
- 다이얼 아웃 시스템에서 사용할 직렬 포트 결정
- 다이얼 아웃 시스템에 대한 루트 암호 획득

계획 정보는 390 페이지 “다이얼 아웃 시스템을 설정하기 전에”를 참조하십시오.

## ▼ 모뎀 및 직렬 포트를 구성하는 방법(다이얼 아웃 시스템)

### 1 모뎀을 프로그래밍합니다.

다양한 모뎀 유형을 사용할 수 있지만 대부분의 모뎀은 Solaris PPP 4.0에 맞는 설정으로 제공됩니다. 다음 목록에는 Solaris PPP 4.0을 사용하는 모뎀에 대한 기본적인 매개변수 설정이 나와 있습니다.

- **DCD** - 사업자 지침을 따릅니다.
- **DTR** - 모뎀이 on-hook(전화기를 놓은 상태, Hang-up) 상태가 되도록 낮게 설정합니다.
- **Flow Control(흐름 제어)** - 전이중 하드웨어 흐름 제어를 위해 RTS/CTS로 설정됩니다.

- **Attention Sequences(주의 시퀀스)** - 사용 안함으로 설정됩니다.

링크 설정 시 문제가 있으며 모뎀에 결함이 있다고 판단되는 경우에는 먼저 모뎀 제조업체의 설명서를 참조하십시오. 여러 웹 사이트에서 모뎀 프로그래밍 지원을 받을 수도 있습니다. 마지막으로, [456 페이지 “모뎀 문제를 진단하는 방법”](#)에도 모뎀 문제를 해결하기 위한 제안 사항이 일부 제공되어 있습니다.

- 2 모뎀 케이블을 다이얼 아웃 시스템의 직렬 포트와 전화 잭에 연결합니다.
- 3 다이얼 아웃 시스템에서 관리자가 됩니다.  
자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.
- 4 모뎀 방향을 다이얼 아웃으로만 지정합니다.

## 다이얼 아웃 시스템에서 통신 구성

이 절의 절차에서는 다이얼 아웃 시스템의 직렬 회선을 통해 통신을 구성하는 방법에 대해 설명합니다. 이러한 절차를 사용하려면 먼저 [407 페이지 “모뎀 및 직렬 포트를 구성하는 방법\(다이얼 아웃 시스템\)”](#)에 설명된 대로 모뎀 및 직렬 포트를 구성해야 합니다.

다음 작업에서는 다이얼 아웃 시스템이 다이얼 인 서버와 성공적으로 통신을 시작할 수 있게 만드는 방법을 보여줍니다. 통신은 PPP 구성 파일의 옵션에 정의된 대로 시작됩니다. 다음 파일을 만들어야 합니다.

- /etc/ppp/options
- /etc/ppp/options.ttyname
- 채트 스크립트
- /etc/ppp/peers/peer-name

Solaris PPP 4.0은 PPP 구성 파일 템플릿을 제공합니다. 이 템플릿은 필요에 맞게 사용자 정의할 수 있습니다. 이러한 파일에 대한 자세한 내용은 [406 페이지 “다이얼 업 PPP 템플릿 파일”](#)을 참조하십시오.

## ▼ 직렬 회선을 통해 통신을 정의하는 방법

- 1 다이얼 아웃 시스템에서 관리자가 됩니다.  
자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.
- 2 다음 항목을 사용하여 /etc/ppp/options라는 파일을 만듭니다.  
`lock`



/etc/ppp/options 파일은 로컬 시스템의 모든 통신에 적용되는 전역 매개변수를 정의하는 데 사용됩니다. lock 옵션을 사용하면 /var/spool/locks/LK.xxx.yyy.zzz 형식의 UUCP 스타일 잠금이 가능해집니다.

주 - 다이얼 아웃 시스템에 /etc/ppp/options 파일이 없으면 슈퍼 유저만 pppd 명령을 실행할 수 있습니다. 그러나 /etc/ppp/options는 비워 둘 수 있습니다.

/etc/ppp/options에 대한 전체 설명은 [469 페이지 “/etc/ppp/options 구성 파일”](#)을 참조하십시오.

- 3 (옵션) 특정 직렬 포트에서 통신이 시작되는 방법을 정의하기 위해 /etc/ppp/options.ttyname이라는 파일을 만듭니다.

다음 예에서는 /etc/ppp/options.ttyname 파일을 보여줍니다. 이 파일은 장치 이름이 /dev/cua/a인 포트에 대한 파일입니다.

```
# cat /etc/ppp/options.cua.a
crtstcts
```

PPP 옵션 crtstcts는 직렬 포트 a에 대해 하드웨어 흐름 제어를 켤 것을 pppd 데몬에 지시합니다.

/etc/ppp/options.ttyname 파일에 대한 자세한 내용을 보려면 [471 페이지 “/etc/ppp/options.ttyname 구성 파일”](#)로 이동하십시오.

- 4 [413 페이지 “모뎀 속도를 설정하는 방법”](#)에 설명된 대로 모뎀 속도를 설정합니다.

## ▼ 피어 호출 명령을 만드는 방법

다이얼 아웃 시스템이 PPP 링크를 시작할 수 있게 만들려면 관리자가 먼저 피어가 될 다이얼 인 서버에 대한 정보를 수집해야 합니다. 그런 다음 관리자는 이 정보를 사용하여 채트 스크립트를 만듭니다. 채트 스크립트는 다이얼 아웃 시스템과 피어 간의 실제 대화에 대해 설명합니다.

- 1 다이얼 아웃 시스템의 모뎀이 실행될 속도를 결정합니다.  
자세한 내용은 [476 페이지 “다이얼 업 링크를 위한 모뎀 속도 구성”](#)을 참조하십시오.
- 2 다이얼 인 서버의 사이트에서 다음 정보를 가져옵니다.
  - 서버의 전화 번호
  - 사용된 인증 프로토콜(해당하는 경우)
  - 채트 스크립트를 위해 피어에 필요한 로그인 절차
- 3 다이얼 인 서버의 사이트에서 이름 서버의 이름 및 IP 주소를 가져옵니다.

#### 4 채트 스크립트에서 특정 피어에 대한 호출을 시작하는 명령을 지정합니다.

예를 들어, 다음 채트 스크립트 `/etc/ppp/mychat`을 만들어 다이얼 인 서버 `myserver`를 호출할 수 있습니다.

```
SAY "Calling the peer\n"
TIMEOUT 10
ABORT BUSY
ABORT 'NO CARRIER'
ABORT ERROR
REPORT CONNECT
"" AT&F1&M5S2=255
TIMEOUT 60
OK ATDT1-123-555-1234
CONNECT \c
SAY "Connected; logging in.\n"
TIMEOUT 5
ogin:--ogin: pppuser
TIMEOUT 20
ABORT 'ogin incorrect'
ssword: \qmypassword
"% " \c
SAY "Logged in. Starting PPP on peer system.\n"
ABORT 'not found'
"" "exec pppd"
~ \c
```

스크립트에는 로그인 절차를 필요로 하는 Oracle Solaris 다이얼 인 서버를 호출하는 명령이 포함되어 있습니다. 각 명령에 대한 자세한 내용은 [480 페이지 “UNIX 스타일의 로그인을 위해 향상된 기본적인 채트 스크립트”](#)를 참조하십시오. 채트 스크립트를 만드는 방법에 대한 자세한 내용을 보려면 [476 페이지 “다이얼 업 링크에서 대화 정의”](#) 절을 읽어 보십시오.

---

주 - 채트 스크립트는 직접 호출하지 않습니다. 대신, 채트 스크립트의 파일 이름을 스크립트를 호출하는 `chat` 명령에 대한 인수로 사용합니다.

---

피어가 Oracle Solaris 또는 이와 유사한 운영 체제를 실행하는 경우 이전 채트 스크립트를 다이얼 아웃 시스템에 대한 템플릿으로 사용해 보십시오.

## ▼ 개별 피어를 사용하여 연결을 정의하는 방법

#### 1 다이얼 아웃 시스템에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

#### 2 DNS 및 이름 서비스 스위치 서비스에 대한 저장소 정보를 업데이트합니다.

```
# svccfg
svc:> select network/dns/client
svc:/network/dns/client> setprop config/domain = astring: "bigcompany.com"
svc:/network/dns/client> setprop config/nameserver = net_address: "10.10.111.15"
```

```

svc:/network/dns/client> addpropval config/nameserver "10.10.130.8"
svc:/network/dns/client> select network/dns/client:default
svc:/network/dns/client:default > refresh
svc:/network/dns/client:default > validate
svc:/network/dns/client:default > select system/name-service/switch
svc:/system/name-service/switch > setprop config/host = astring: "files dns"
  svc:/system/name-service/switch:default > select system/name-service/switch:default
svc:/system/name-service/switch:default > refresh
svc:/system/name-service/switch:default > validate
# svcadm enable network/dns/client
# svcadm refresh system/name-service/switch

```

### 3 피어에 대한 파일을 만듭니다.

예를 들어, 다음 파일을 만들어 다이얼 인 서버 myserver를 정의합니다.

```

# cat /etc/ppp/peers/myserver
/dev/cua/a
57600
noipdefault
defaultroute
idle 120
noauth
connect "chat -U 'mypassword' -T 1-123-555-1213 -f /etc/ppp/mychat"

/dev/cua/a

```

/dev/cua/a 장치가 myserver에 대한 호출의 직렬 인터페이스로 사용되도록 지정합니다.

57600

링크의 속도를 정의합니다.

noipdefault

피어 myserver를 사용하는 트랜잭션의 경우 처음에 다이얼 아웃 시스템의 IP 주소가 0.0.0.0으로 설정됨을 지정합니다. myserver는 모든 다이얼 업 세션에서 다이얼 아웃 시스템에 IP 주소를 지정합니다.

idle 120

휴식 기간이 120초가 넘으면 링크 시간이 초과됨을 나타냅니다.

noauth

다이얼 아웃 시스템과 연결을 협상할 때 피어 myserver가 인증 자격 증명을 제공하지 않아도 됨을 지정합니다.

connect "chat -U 'mypassword' -T 1-123-555-1213 -f /etc/ppp/mychat"

connect 옵션과 해당 인수를 지정합니다(피어의 전화 번호와 호출 명령이 있는 채트 스크립트 /etc/ppp/mychat 포함).

**참조** 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 다른 다이얼 아웃 시스템을 구성하려면 [407 페이지 “모뎀 및 직렬 포트를 구성하는 방법\(다이얼 아웃 시스템\)”](#)을 참조하십시오.

- 다른 컴퓨터로 다이얼 아웃하여 모뎀 연결을 테스트하려면 **cu(1C)** 및 **tip(1)** 매뉴얼 페이지를 참조하십시오. 이러한 유틸리티를 사용하면 모뎀이 제대로 구성되었는지 테스트하는 데 도움이 됩니다. 이러한 유틸리티를 사용하여 다른 시스템과 연결을 설정할 수 있는지 테스트할 수도 있습니다.
- 구성 파일 및 옵션에 대한 자세한 내용은 **465 페이지 “파일 및 명령줄에서 PPP 옵션 사용”**을 참조하십시오.
- 다이얼 인 서버를 구성하려면 **412 페이지 “다이얼 인 서버에서 장치 구성”**을 참조하십시오.

## 다이얼 인 서버 구성

이 절의 작업은 다이얼 인 서버를 구성하기 위한 것입니다. 다이얼 인 서버는 다이얼 아웃 시스템에서 PPP 링크를 통해 호출을 받는 피어 시스템입니다. 작업에서는 다이얼 인 서버 **myserver**(**그림 16-1**에서 소개됨)를 구성하는 방법을 보여줍니다.

### 다이얼 인 서버 구성 작업(작업 맵)

표 17-3 다이얼 인 서버 설정 작업 맵

작업	설명	수행 방법
1. 미리 구성 정보 수집	링크를 설정하기 전에 필요한 데이터(예: 피어 호스트 이름, 대상 전화 번호 및 모뎀 속도)를 수집합니다.	<b>390 페이지 “다이얼 업 PPP 링크 계획”</b>
2. 모뎀 및 직렬 포트 구성	모뎀 및 직렬 포트를 설정합니다.	<b>413 페이지 “모뎀 및 직렬 포트를 구성하는 방법(다이얼 인 서버)”</b>
3. 호출 피어 정보 구성	다이얼 인 서버를 호출할 수 있는 모든 다이얼 아웃 시스템에 대해 사용자 환경과 PPP 옵션을 설정합니다.	<b>414 페이지 “다이얼 인 서버의 사용자를 구성하는 방법”</b>
4. 직렬 회선 통신 구성	직렬 회선을 통한 전송의 특징을 구성합니다.	<b>415 페이지 “직렬 회선을 통해 통신을 정의하는 방법(다이얼 인 서버)”</b>

### 다이얼 인 서버에서 장치 구성

다음 절차에서는 다이얼 인 서버에서 모뎀과 직렬 포트를 구성하는 방법에 대해 설명합니다.

후속 절차를 수행하려면 피어 다이얼 인 서버에서 다음 작업을 완료해야 합니다.

- Oracle Solaris 릴리스 설치
- 최적의 모뎀 속도 결정

- 사용할 직렬 포트 결정

## ▼ 모뎀 및 직렬 포트를 구성하는 방법(다이얼 인 서버)

- 1 모뎀 제조업체의 설명서에 제공된 지침대로 모뎀을 프로그래밍합니다.  
기타 제안 사항은 407 페이지 “모뎀 및 직렬 포트를 구성하는 방법(다이얼 아웃 시스템)”을 참조하십시오.
- 2 다이얼 인 서버에 있는 직렬 포트에 모뎀을 연결합니다.
- 3 다이얼 인 서버에서 관리자가 됩니다.  
자세한 내용은 **Oracle Solaris 관리: 보안 서비스**의 “관리 권한을 얻는 방법”을 참조하십시오.
- 4 모뎀 방향을 다이얼 인으로만 지정합니다.

## ▼ 모뎀 속도를 설정하는 방법

다음 절차에서는 다이얼 인 서버에 대해 모뎀 속도를 설정하는 방법에 대해 설명합니다.  
Sun Microsystems 컴퓨터에 사용할 속도에 대한 제안 사항은 476 페이지 “다이얼 업 링크를 위한 모뎀 속도 구성”을 참조하십시오.

- 1 다이얼 인 서버에 로그인합니다.
- 2 **tip** 명령을 사용하여 모뎀에 연결합니다.  
tip을 사용하여 모뎀 속도를 설정하는 지침은 tip(1) 매뉴얼 페이지에 있습니다.
- 3 모뎀에서 고정 DTE 속도를 구성합니다.
- 4 **ttymon**을 사용하여 직렬 포트를 해당 속도로 고정합니다.

참조 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 413 페이지 “모뎀 및 직렬 포트를 구성하는 방법(다이얼 인 서버)”
- 414 페이지 “다이얼 인 서버의 사용자를 구성하는 방법”

## 다이얼 인 서버의 사용자 설정

다이얼 인 서버를 설정할 때는 알려진 각 원격 호출자에 대한 정보도 구성해야 합니다.

이 절의 절차를 시작하려면 먼저 다음 작업을 완료해야 합니다.

- 원격 다이얼 아웃 시스템에서 로그인할 수 있는 모든 사용자의 UNIX 사용자 이름 획득
- 413 페이지 “모뎀 및 직렬 포트를 구성하는 방법(다이얼 인 서버)”에 설명된 대로 모뎀 및 직렬 회선 설정
- 원격 사용자로부터의 수신 호출에 지정할 전용 IP 주소 지정. 잠재적 호출자의 수가 다이얼 인 서버에 있는 모뎀 및 직렬 포트의 수를 초과하는 경우에는 전용 수신 IP 주소를 만드십시오. 전용 IP 주소를 만드는 방법에 대한 자세한 내용을 보려면 492 페이지 “호출자를 위한 IP 주소 지정 체계 만들기”를 참조하십시오.

## ▼ 다이얼 인 서버의 사용자를 구성하는 방법

### 1 다이얼 인 서버에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 각 원격 PPP 사용자에 대해 다이얼 인 서버에서 새 계정을 만듭니다.

새 사용자를 만드는 방법에 대한 자세한 내용은 [Oracle Solaris 관리: 일반 작업의 “사용자 계정 설정 및 관리\(작업 맵\)”](#)를 참조하십시오.

### 3 각 호출자에 대해 사용자의 PPP 세션과 관련된 다양한 옵션이 포함된 \$HOME/.ppprc 파일을 만듭니다.

예를 들어, 다음 .ppprc 파일을 pppuser에 대해 만들 수 있습니다.

```
# cat /export/home/pppuser/.ppprc
noccp
```

noccp는 링크에서 압축 제어를 끕니다.

**참조** 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 414 페이지 “다이얼 인 서버의 사용자를 구성하는 방법”
- 415 페이지 “직렬 회선을 통해 통신을 정의하는 방법(다이얼 인 서버)”

## 다이얼 인 서버를 통한 통신 구성

다음 작업에서는 다이얼 인 서버가 임의의 다이얼 아웃 시스템과 통신을 열 수 있게 만드는 방법을 보여줍니다. 다음 PPP 구성 파일에 정의된 옵션에 따라 통신이 설정되는 방법이 결정됩니다.

- /etc/ppp/options
- /etc/ppp/options.ttyname

이러한 파일에 대한 자세한 내용은 465 페이지 “파일 및 명령줄에서 PPP 옵션 사용”을 참조하십시오.

계속 진행하려면 다음 작업을 완료해야 합니다.

- 413 페이지 “모뎀 및 직렬 포트를 구성하는 방법(다이얼 인 서버)”에 설명된 대로 다이얼 인 서버에서 직렬 포트 및 모뎀 구성
- 414 페이지 “다이얼 인 서버의 사용자를 구성하는 방법”에 설명된 대로 다이얼 인 서버의 잠재적 사용자에 대한 정보 구성

## ▼ 직렬 회선을 통해 통신을 정의하는 방법(다이얼 인 서버)

### 1 다이얼 인 서버에서 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

### 2 다음 항목을 사용하여 /etc/ppp/options 파일을 만듭니다.

**nodefaultroute**

nodefaultroute는 로컬 시스템에 있는 어떤 pppd 세션도 root 권한 없이 기본 경로를 설정할 수 없음을 나타냅니다.

---

주 - 다이얼 인 서버에 /etc/ppp/options 파일이 없으면 슈퍼 유저만 pppd 명령을 실행할 수 있습니다. 그러나 /etc/ppp/options 파일은 비워 둘 수 있습니다.

---

### 3 /etc/options.ttyname 파일을 만들어 직렬 포트 ttyname을 통해 받는 호출이 처리되는 방법을 정의합니다.

다음 /etc/options.ttya 파일은 다이얼 인 서버의 직렬 포트 /dev/ttya에서 수신 호출이 처리되는 방법을 정의합니다.

**:10.0.0.80**  
**xonxoff**

**:10.0.0.80** IP 주소 10.0.0.80을 직렬 포트 ttya를 통해 호출하는 모든 피어에 지정합니다.

**xonxoff** 직렬 회선이 소프트웨어 흐름 제어를 사용으로 설정한 상태에서 모뎀으로부터의 통신을 처리할 수 있도록 허용합니다.

**참조** 이 장의 모든 절차를 따른 경우 다이얼 업 링크의 구성을 완료한 것입니다. 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 다른 컴퓨터로 다이얼 아웃하여 모뎀 연결을 테스트하려면 [cu\(1C\)](#) 및 [tip\(1\)](#) 매뉴얼 페이지를 참조하십시오. 이러한 유틸리티를 사용하면 모뎀이 제대로 구성되었는지 테스트하는 데 도움이 됩니다. 이러한 유틸리티를 사용하여 다른 시스템과 연결을 설정할 수 있는지 테스트할 수도 있습니다.
- 다이얼 인 서버에 대한 옵션을 더 구성하려면 [412 페이지 “다이얼 인 서버 구성”](#)을 참조하십시오.
- 다이얼 아웃 시스템을 더 구성하려면 [406 페이지 “다이얼 아웃 시스템 구성”](#)을 참조하십시오.
- 원격 시스템이 다이얼 인 서버를 호출하게 만들려면 [416 페이지 “다이얼 인 서버 호출”](#)을 참조하십시오.

## 다이얼 인 서버 호출

다이얼 아웃 시스템이 다이얼 인 서버를 호출하게 만들어 다이얼 업 PPP 링크를 설정합니다. 로컬 PPP 구성 파일에서 **demand** 옵션을 지정하여 다이얼 아웃 시스템이 서버를 호출하게 만들 수 있습니다. 그러나 사용자가 다이얼 아웃 시스템에서 **pppd** 명령을 실행하는 것이 링크를 설정하는 가장 일반적인 방법입니다.

후속 작업으로 계속 진행하려면 다음 작업 중 하나 또는 모두를 완료해야 합니다.

- [406 페이지 “다이얼 아웃 시스템 구성”](#)에 설명된 대로 다이얼 아웃 시스템 설정
- [412 페이지 “다이얼 인 서버 구성”](#)에 설명된 대로 다이얼 인 서버 설정

### ▼ 다이얼 인 서버를 호출하는 방법

**1 root가 아니라 일반 사용자 계정을 사용하여 다이얼 아웃 시스템에 로그인합니다.**

**2 pppd 명령을 실행하여 다이얼 인 서버를 호출합니다.**

예를 들어, 다음 명령을 실행하면 다이얼 아웃 시스템과 다이얼 인 서버 **myserver** 간에 링크가 시작됩니다.

```
% pppd 57600 call myserver
```

**pppd** pppd 데몬을 호출하여 호출을 시작합니다.

**57600** 호스트와 모뎀 간 회선의 속도를 설정합니다.

**call myserver** call 옵션(pppd)을 호출합니다. 그러면 pppd가 `/etc/ppp/peers/myserver` 파일([410 페이지 “개별 피어를 사용하여 연결을 정의하는 방법”](#)에서 만들어짐)의 옵션을 읽습니다.



- 3 호스트 lindyhop([그림 16-1](#)에 나와 있음)과 같이 서버의 네트워크에 있는 호스트에 연결합니다.

```
ping lindyhop
```

링크가 올바르게 동작하지 않으면 [21 장](#), “일반적인 PPP 문제 해결(작업)”을 참조하십시오.

- 4 PPP 세션 종료:

```
% pkill -x pppd
```

**참조** 이 장의 모든 절차를 따른 경우 다이얼 업 링크의 구성을 완료한 것입니다. 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 사용자가 자신의 다이얼 아웃 시스템에서 작업을 시작하게 만들려면 [416 페이지](#) “다이얼 인 서버를 호출하는 방법”을 참조하십시오.
- 링크에서 문제를 해결하려면 [21 장](#), “일반적인 PPP 문제 해결(작업)”을 참조하십시오.
- 이 장에 사용되는 파일 및 옵션에 대한 자세한 내용은 [465 페이지](#) “파일 및 명령줄에서 PPP 옵션 사용”을 참조하십시오.



## 전용 회선 PPP 링크 설정(작업)

이 장에서는 피어 간에 전용 회선을 사용하는 PPP 링크를 구성하는 방법에 대해 설명합니다. 주요 절은 다음과 같습니다.

- 420 페이지 “전용 회선에서 동기 장치 구성”
- 421 페이지 “전용 회선에서 시스템 구성”

### 전용 회선 설정(작업 맵)

전용 회선 링크는 다이얼 업 링크에 비해 비교적 설정하기 쉽습니다. 대부분의 경우에서 CSU/DSU, 전화 걸기 서비스 또는 인증을 구성할 필요가 없습니다. CSU/DSU를 구성하지 않아도 되는 경우 제조업체 설명서를 통해 이 복잡한 작업에 대한 지원을 받으십시오.

다음 표의 작업 맵에서는 기본 전용 회선 링크를 설정하는 데 사용되는 모든 작업에 대해 설명합니다.

주 - 일부 전용 회선 유형에서는 CSU/DSU가 반대쪽 피어의 주소로 “전화를 걸어야” 합니다. 예를 들어, 프레임 릴레이는 SVC(가상 교환 회로) 또는 교환식 56 서비스를 사용합니다.

표 18-1 전용 회선 링크 설정 작업 맵

작업	설명	수행 방법
1. 사전 구성 정보 수집	링크를 설정하기 전에 필요한 데이터를 수집합니다.	394 페이지 “전용 회선 링크에 대해 수집해야 하는 정보”
2. 전용 회선 하드웨어 설정	CSU/DSU 및 동기 인터페이스 카드를 조립합니다.	420 페이지 “동기 장치를 구성하는 방법”
3. 필요한 경우 인터페이스 카드 구성	전용 회선 시작 시 사용할 인터페이스 스크립트를 구성합니다.	420 페이지 “동기 장치를 구성하는 방법”

표 18-1 전용 회선 링크 설정 작업 맵 (계속)

작업	설명	수행 방법
4. 원격 피어에 대한 정보 구성	로컬 시스템과 원격 피어 간의 통신이 이루어지는 방법을 정의합니다.	421 페이지 “전용 회선에서 시스템을 구성하는 방법”
5. 전용 회선 시작	시스템이 부트 프로세스의 일환으로 전용 회선을 통해 PPP를 시작하도록 구성합니다.	421 페이지 “전용 회선에서 시스템을 구성하는 방법”

## 전용 회선에서 동기 장치 구성

이 절의 작업에서는 394 페이지 “전용 회선 링크 구성의 예”에서 소개된 전용 회선 토폴로지에 필요한 장비를 구성합니다. 전용 회선에 연결하는 데 필요한 동기 장치에는 인터페이스와 모뎀이 있습니다.

### 동기 장치 설정을 위한 필수 조건

후속 절차를 수행하려면 다음 항목이 있어야 합니다.

- 공급자가 사이트에 설치한 전용 회선(작동해야 함)
- 동기 장치(CSU/DSU)
- 시스템에 설치된 Oracle Solaris 릴리스
- 시스템에 필요한 유형의 동기 인터페이스 카드

### ▼ 동기 장치를 구성하는 방법

- 1 필요한 경우 로컬 시스템에 인터페이스 카드를 물리적으로 설치합니다.  
제조업체 설명서의 지침을 따르십시오.
- 2 케이블로 CSU/DSU와 인터페이스를 연결합니다.  
필요한 경우 케이블로 CSU/DSU와 전용 회선 잭 또는 이와 유사한 커넥터를 연결합니다.
- 3 제조업체 또는 네트워크 공급자의 설명서에 제공된 지침대로 CSU/DSU를 구성합니다.

---

주 - 전용 회선을 임대해 준 공급자가 링크에 대한 CSU/DSU를 제공 및 구성해 줄 수도 있습니다.

---

- 4 필요한 경우 인터페이스 설명서에 제공된 지침대로 인터페이스 카드를 구성합니다.  
인터페이스 카드를 구성할 때는 인터페이스에 대한 시작 스크립트를 생성해야 합니다.  
[그림 16-2](#)에 나와 있는 전용 회선 구성에서 LocalCorp의 라우터에는 HSI/P 인터페이스 카드가 사용됩니다.

다음 스크립트 hsi-conf는 HSI/P 인터페이스를 시작합니다.

```
#!/bin/ksh
/opt/SUNWconn/bin/hsip_init hihp1 speed=1536000 mode=fdx loopback=no \
nrzi=no txc=txc rxc=rxc txd=txd rxd=rxd signal=no 2>&1 > /dev/null

hihp1          HSI/P가 사용되는 동기 포트임을 나타냅니다.

speed=1536000   CSU/DSU의 속도를 나타내기 위해 설정합니다.
```

**참조** 전용 회선에서 로컬 시스템을 구성하려면 421 페이지 “전용 회선에서 시스템을 구성하는 방법”을 참조하십시오.

## 전용 회선에서 시스템 구성

이 절의 작업에서는 전용 회선의 본인 쪽 끝에서 로컬 피어 역할을 하도록 라우터를 설정하는 방법에 대해 설명합니다. 작업에는 394 페이지 “전용 회선 링크 구성의 예”에서 예로 소개된 전용 회선이 사용됩니다.

### 전용 회선에서의 로컬 시스템 구성을 위한 필수 조건

후속 절차를 수행하려면 다음 작업을 완료해야 합니다.

- 420 페이지 “전용 회선에서 동기 장치 구성”에 설명된 대로 링크에 대한 동기 장치 설정 및 구성
- 전용 회선에서 로컬 시스템의 루트 암호 획득
- 전용 회선 공급자의 서비스를 사용하기 위해 네트워크에서 라우터로 실행되도록 로컬 시스템 설정

## ▼ 전용 회선에서 시스템을 구성하는 방법

- 1 로컬 시스템(라우터)에서 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스**의 “관리 권한을 얻는 방법”을 참조하십시오.

- 2 라우터의 /etc/hosts 파일에서 원격 피어에 대한 항목을 추가합니다.

```
# cat /etc/hosts
#
# Internet host table
#
127.0.0.1          localhost
192.168.130.10     local2-peer      loghost
```

```
192.168.130.11 local1-net
10.0.0.25 farISP
```

예제 /etc/hosts 파일은 가상 회사 LocalCorp에 있는 로컬 라우터를 위한 것입니다. 서비스 공급자측에 있는 원격 피어 farISP의 IP 주소 및 호스트 이름을 기록해 둡니다.

**3 공급자의 피어에 대한 정보를 보관할 /etc/ppp/peers/peer-name 파일을 만듭니다.**

이 예제 전용 회선 링크를 위해서는 /etc/ppp/peers/farISP 파일을 만듭니다.

```
# cat /etc/ppp/peers/farISP
init '/etc/ppp/conf_hsi'
local
/dev/hihp1
sync
noauth
192.168.130.10:10.0.0.25
passive
persist
noccp
nopcomp
novj
noaccomp
```

다음 표에는 /etc/ppp/peers/farISP에 사용되는 옵션 및 매개변수가 설명되어 있습니다.

옵션	정의
init '/etc/ppp/conf_hsi'	링크를 시작합니다. 그러면 init가 /etc/ppp/conf_hsi 스크립트의 매개변수를 사용하여 HSI 인터페이스를 구성합니다.
local	DTR(Data Terminal Ready) 신호의 상태를 변경하지 말 것을 pppd 데몬에 지시합니다. DCD(Data Carrier Detect) 입력 신호를 무시할 것도 pppd에 지시합니다.
/dev/hihp1	동기 인터페이스의 장치 이름을 제공합니다.
sync	링크에 대한 동기 인코딩을 설정합니다.
noauth	로컬 시스템이 피어로부터 인증을 요구하지 않아도 됨을 설정합니다. 그러나 피어는 인증을 요구할 수도 있습니다.
192.168.130.10:10.0.0.25	로컬 피어 및 원격 피어의 IP 주소를 콜론으로 구분하여 정의합니다.
passive	LCP 구성-요청을 최대 횟수로 실행한 후 침묵 상태로 전환하여 피어가 시작되기를 기다릴 것을 로컬 시스템에 있는 pppd 데몬에 지시합니다.
persist	연결이 끝난 후 링크를 다시 시작할 것을 pppd 데몬에 지시합니다.
noccp, nopcomp, novj, noaccomp	각각 CCP(Compression Control Protocol), 프로토콜 필드 압축, Van Jacobson 압축 및 주소/컨트롤 필드 압축을 사용 안함으로 설정합니다. 이러한 압축은 다이얼 업 링크에서 전송 속도를 높이지만 전용 회선의 속도를 낮출 수 있습니다.

#### 4 부트 프로세스의 일환으로 PPP 링크를 만드는 demand라는 초기화 스크립트를 만듭니다.

```
# cat /etc/ppp/demand
#!/bin/sh
if [ -f /system/volatile/ppp-demand.pid ] &&
    /usr/bin/kill -s 0 '/bin/cat /system/volatile/ppp-demand.pid'
then
    :
else
    /usr/bin/pppd call farISP
fi
```

demand 스크립트에는 전용 회선 링크를 설정하기 위한 pppd 명령이 포함되어 있습니다. 다음 표에는 \$PPPPDIR/demand의 내용이 설명되어 있습니다.

코드 샘플	설명
if [ -f /system/volatile/ppp-demand.pid ] && /usr/bin/kill -s 0 '/bin/cat /system/volatile/ppp-demand.pid'	이러한 행에서는 pppd가 실행되고 있는지 확인합니다. pppd가 실행되고 있는 경우 이를 시작하지 않아도 됩니다.
/usr/bin/pppd call farISP	이 행에서는 pppd를 시작합니다. pppd는 /etc/ppp/options에서 옵션을 읽습니다. 명령줄에서 call farISP 옵션을 실행하면 /etc/ppp/peers/farISP도 읽습니다

Solaris PPP 4.0 시작 스크립트 /etc/rc2.d/S47pppd는 부트 프로세스의 일환으로 demand 스크립트를 호출합니다. /etc/rc2.d/S47pppd의 다음 행에서는 \$PPPPDIR/demand라는 파일이 있는지 확인하기 위해 검색을 수행합니다.

```
if [ -f $PPPPDIR/demand ]; then
    . $PPPPDIR/demand
fi
```

\$PPPPDIR/demand가 발견되면 실행됩니다. \$PPPPDIR/demand를 실행하는 동안 링크가 설정됩니다.

주-로컬 네트워크 외부에 있는 시스템에 연결하려면 사용자가 telnet, ftp, rsh 또는 이와 유사한 명령을 실행하게 하십시오.

**참조** 이 장의 모든 절차를 따른 경우 전용 회선 링크의 구성을 완료한 것입니다. 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 문제 해결 정보를 찾으려면 [463 페이지](#) “[전용 회선 문제 해결](#)”을 참조하십시오.
- 이 장에 사용되는 파일 및 옵션에 대한 자세한 내용은 [465 페이지](#) “[파일 및 명령줄에서 PPP 옵션 사용](#)”을 참조하십시오.





## PPP 인증 설정(작업)

이 장에는 PPP 인증을 설정하는 작업이 있습니다. 여기서 다루는 항목은 다음과 같습니다.

- 426 페이지 “PAP 인증 구성”
- 433 페이지 “CHAP 인증 구성”

인증을 위해 전용 회선 링크보다는 다이얼 업 링크가 구성되어 있을 확률이 크므로 절차에서는 다이얼 업 링크를 통해 인증을 구현하는 방법에 대해 설명합니다. 회사 보안 정책상 인증이 필요한 경우에는 전용 회선을 통해 인증을 구성할 수 있습니다. 전용 회선 인증의 경우 이 장의 작업을 지침으로 활용하십시오.

PPP 인증을 사용하고 싶지만 어떤 프로토콜을 사용해야 하는지 확실하지 않다면 385 페이지 “PPP 인증을 사용하는 이유” 절을 검토하십시오. PPP 인증에 대한 자세한 내용은 [pppd\(1M\)](#) 매뉴얼 페이지 및 486 페이지 “링크에서 호출자 인증”을 참조하십시오.

## PPP 인증 구성(작업 맵)

이 절에는 PPP 인증 절차에 빠르게 액세스하는 데 도움이 되는 작업 맵이 있습니다.

표 19-1 일반 PPP 인증 작업 맵

작업	설명	수행 방법
PAP 인증 구성	이러한 절차를 사용하면 다이얼 인 서버 및 다이얼 아웃 시스템에서 PAP 인증을 사용으로 설정할 수 있습니다.	426 페이지 “PAP 인증 설정(작업 맵)”
CHAP 인증 구성	이러한 절차를 사용하면 다이얼 인 서버 및 다이얼 아웃 시스템에서 CHAP 인증을 사용으로 설정할 수 있습니다.	433 페이지 “CHAP 인증 설정(작업 맵)”

## PAP 인증 구성

이 절의 작업에서는 PAP(암호 인증 프로토콜)를 사용하여 PPP 링크에 대해 인증을 구현하는 방법에 대해 설명합니다. 작업에는 다이얼 업 링크에 맞는 PAP 시나리오를 보여주기 위해 [396 페이지 “PPP 인증 구성의 예”](#)에 나와 있는 예가 사용됩니다. 지침을 기반으로 사용자 사이트에서 PAP 인증을 구현해 보십시오.

다음 절차를 수행하려면 먼저 다음 작업을 완료해야 합니다.

- 다이얼 인 서버와 신뢰할 수 있는 호출자에 속하는 다이얼 아웃 시스템 간에 다이얼 업 링크를 설정하고 테스트
- 다이얼 인 서버 인증의 경우 네트워크 암호 데이터베이스가 관리되는 시스템(예: LDAP, NIS 또는 로컬 파일)에 대한 슈퍼 유저 사용 권한 획득(이상적인 상황)
- 로컬 시스템(다이얼 인 서버 또는 다이얼 아웃 시스템)에 대한 슈퍼 유저 권한 획득

## PAP 인증 설정(작업 맵)

다음 작업 맵을 사용하면 다이얼 인 서버 및 다이얼 아웃 시스템의 신뢰할 수 있는 호출자에 대한 PAP 관련 작업에 빠르게 액세스할 수 있습니다.

표 19-2 PAP 인증 작업 맵(다이얼 인 서버)

작업	설명	수행 방법
1. 미리 구성 정보 수집	인증에 필요한 사용자 이름 및 기타 데이터를 수집합니다.	<a href="#">395 페이지 “링크에서 인증 계획”</a>
2. 필요한 경우 암호 데이터베이스 업데이트	모든 잠재적 호출자가 서버의 암호 데이터베이스에 있는지 확인합니다.	<a href="#">427 페이지 “PAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)”</a>
3. PAP 데이터베이스 만들기	/etc/ppp/pap-secrets에서 모든 잠재적 호출자에 대한 보안 자격 증명을 만듭니다.	<a href="#">427 페이지 “PAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)”</a>
4. PPP 구성 파일 수정	/etc/ppp/options 및 /etc/ppp/peers/peer-name 파일에 PAP 관련 옵션을 추가합니다.	<a href="#">429 페이지 “PPP 구성 파일에 PAP 지원을 추가하는 방법(다이얼 인 서버)”</a>

표 19-3 PAP 인증 작업 맵(다이얼 아웃 시스템)

작업	설명	수행 방법
1. 미리 구성 정보 수집	인증에 필요한 사용자 이름 및 기타 데이터를 수집합니다.	<a href="#">395 페이지 “링크에서 인증 계획”</a>

표 19-3 PAP 인증 작업 맵(다이얼 아웃 시스템) (계속)

작업	설명	수행 방법
2. 신뢰할 수 있는 호출자의 시스템에 대한 PAP 데이터베이스 만들기	/etc/ppp/pap-secrets에서 신뢰할 수 있는 호출자에 대한 보안 자격 증명을 만들고 필요한 경우 다이얼 아웃 시스템을 호출하는 다른 사용자에게 대한 보안 자격 증명을 만듭니다.	430 페이지 “신뢰할 수 있는 호출자에 대해 PAP 인증 자격 증명을 구성하는 방법”
3. PPP 구성 파일 수정	/etc/ppp/options 및 /etc/ppp/peers/peer-name 파일에 PAP 관련 옵션을 추가합니다.	432 페이지 “PPP 구성 파일에 PAP 지원을 추가하는 방법(다이얼 아웃 시스템)”

## 다이얼 인 서버에서 PAP 인증 구성

PAP 인증을 설정하려면 다음 작업을 수행해야 합니다.

- PAP 자격 증명 데이터베이스 만들기
- PAP 지원을 위해 PPP 구성 파일 수정

### ▼ PAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)

이 절차에서는 링크의 호출자를 인증하는 데 사용되는 PAP 보안 자격 증명이 포함된 /etc/ppp/pap-secrets 파일을 수정합니다. /etc/ppp/pap-secrets는 PPP 링크의 두 시스템 모두에 있어야 합니다.

그림 16-3에 소개된 샘플 PAP 구성에는 PAP의 login 옵션이 사용됩니다. 이 옵션을 사용하려면 네트워크의 암호 데이터베이스를 업데이트해야 할 수도 있습니다. login 옵션에 대한 자세한 내용은 489 페이지 “/etc/ppp/pap-secrets에 login 옵션 사용”을 참조하십시오.

- 1 신뢰할 수 있는 모든 잠재적 호출자의 목록을 어셈블합니다. 신뢰할 수 있는 호출자는 자신의 원격 시스템에서 다이얼 인 서버를 호출할 권한을 부여받을 사람입니다.
- 2 신뢰할 수 있는 각 호출자가 다이얼 인 서버의 암호 데이터베이스에서 이미 UNIX 사용자 이름 및 암호를 가지고 있는지 확인하십시오.

주 - PAP의 login 옵션을 사용하여 호출자를 인증하는 샘플 PAP 구성의 경우 특히 확인 작업이 중요합니다. PAP의 login을 구현하지 않을 경우 호출자의 PAP 사용자 이름이 UNIX 사용자 이름에 해당하지 않아도 됩니다. 표준 /etc/ppp/pap-secrets에 대한 자세한 내용은 486 페이지 “/etc/ppp/pap-secrets 파일”을 참조하십시오.

신뢰할 수 있는 잠재적 호출자에게 UNIX 사용자 이름 및 암호가 없는 경우 다음을 수행하십시오.

- a. 개인적으로 알고 있지 않은 호출자의 경우 해당 호출자의 관리자에게 이들이 다이얼 인 서버에 대한 액세스 권한을 가지고 있는지 확인합니다.
- b. 회사 보안 정책에 명시된 방식으로 이러한 호출자의 UNIX 사용자 이름 및 암호를 만듭니다.

### 3 다이얼 인 서버에서 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

### 4 /etc/ppp/pap-secrets 파일을 편집합니다.

이번 릴리스는 pap-secrets 파일(/etc/ppp에 있음)을 제공합니다. 여기에는 PAP 인증을 사용하는 방법에 대한 설명이 포함되어 있지만 옵션은 포함되어 있지 않습니다. 설명 끝에 다음과 같은 옵션을 추가할 수 있습니다.

```
user1      myserver      ""          *
user2      myserver      ""          *
myserver   user2         serverpass    *
```

login 옵션(/etc/ppp/pap-secrets)을 사용하려면 신뢰할 수 있는 각 호출자의 UNIX 사용자 이름을 입력해야 합니다. 세번째 필드에 큰따옴표(“) 세트가 나타날 때마다 서버의 암호 데이터베이스에서 호출자의 암호가 조회됩니다.

myserver \* serverpass \* 항목에는 다이얼 인 서버의 PAP 사용자 이름 및 암호가 포함되어 있습니다. **그림 16-3**에서 신뢰할 수 있는 호출자인 user2에게는 원격 피어로부터의 인증 작업이 필요합니다. 따라서 user2와 링크가 설정된 경우 myserver의 /etc/ppp/pap-secrets 파일에 사용할 PAP 자격 증명이 포함됩니다.

**참조** 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 428 페이지 “PAP를 위해 PPP 구성 파일 수정(다이얼 인 서버)”
- 430 페이지 “신뢰할 수 있는 호출자에 대해 PAP 인증 구성(다이얼 아웃 시스템)”

## PAP를 위해 PPP 구성 파일 수정(다이얼 인 서버)

이 절의 작업에서는 다이얼 인 서버에서 PAP 인증을 지원하기 위해 기존 PPP 구성 파일을 업데이트하는 방법에 대해 설명합니다.

## ▼ PPP 구성 파일에 PAP 지원을 추가하는 방법(다이얼 인 서버)

절차에는 415 페이지 “직렬 회선을 통해 통신을 정의하는 방법(다이얼 인 서버)”에 소개된 PPP 구성 파일이 예로 사용됩니다.

### 1 다이얼 인 서버에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 /etc/ppp/options 파일에 인증 옵션을 추가합니다.

예를 들어 굵게 표시된 옵션을 기존 /etc/ppp/options 파일에 추가하면 PAP 인증이 구현됩니다.

```
lock
auth
login
nodefaultroute
proxyarp
ms-dns 10.0.0.1
idle 120
```

auth	서버에서 링크를 설정하기 전에 호출자를 인증해야 함을 지정합니다.
login	표준 UNIX 사용자 인증 서비스를 사용하여 원격 호출자를 인증해야 함을 지정합니다.
nodefaultroute	로컬 시스템에 있는 어떤 pppd 세션도 root 권한 없이 기본 경로를 설정할 수 없음을 나타냅니다.
proxyarp	피어의 IP 주소 및 시스템의 이더넷 주소를 지정하는 항목을 시스템의 ARP(주소 결정 프로토콜) 테이블에 추가합니다. 이 옵션을 사용하면 다른 시스템에서 피어가 로컬 이더넷에 있는 것처럼 보입니다.
ms-dns 10.0.0.1	pppd를 사용으로 설정하여 DNS(도메인 이름 서버) 주소인 10.0.0.1을 클라이언트에 제공합니다.
idle 120	유휴 사용자가 2분 후에 연결 해제되도록 지정합니다.

### 3 /etc/ppp/options.cua.a 파일에서 cua/a 사용자에게 대해 다음 주소를 추가합니다.

```
:10.0.0.2
```

### 4 /etc/ppp/options.cua.b 파일에서 cua/b 사용자에게 대해 다음 주소를 추가합니다.

```
:10.0.0.3
```

## 5 /etc/ppp/pap-secrets 파일에서 다음 항목을 추가합니다.

```
*      *      ""      *
```

주 - 앞서 설명한 login 옵션은 필요한 사용자 인증을 제공합니다. /etc/ppp/pap-secrets 파일에 이렇게 입력하는 것이 login 옵션을 사용하여 PAP를 사용으로 설정하는 표준 방법입니다.

**참조** 다이얼 인 서버의 신뢰할 수 있는 호출자에 대해 PAP 인증 자격 증명을 구성하려면 [430 페이지 “신뢰할 수 있는 호출자에 대해 PAP 인증 구성\(다이얼 아웃 시스템\)”](#)을 참조하십시오.

## 신뢰할 수 있는 호출자에 대해 PAP 인증 구성(다이얼 아웃 시스템)

이 절에는 신뢰할 수 있는 호출자의 다이얼 아웃 시스템에서 PAP 인증을 설정하는 작업이 있습니다. 시스템 관리자는 잠재적 호출자에 대한 배포 작업 전에 시스템에서 PAP 인증을 설정할 수 있습니다. 원격 호출자에게 이미 자신의 시스템이 있는 경우에는 이 절의 작업을 해당 호출자에게 할당할 수도 있습니다.

신뢰할 수 있는 호출자에 대해 PAP를 구성하기 위해 수행해야 하는 두 작업은 다음과 같습니다.

- 호출자의 PAP 보안 자격 증명 구성
- PAP 인증을 지원하기 위해 호출자의 다이얼 아웃 시스템 구성

## ▼ 신뢰할 수 있는 호출자에 대해 PAP 인증 자격 증명을 구성하는 방법

이 절차에서는 신뢰할 수 있는 두 호출자에 대해 PAP 자격 증명을 설정하는 방법을 보여줍니다. 둘 중 하나는 원격 피어로부터 인증 자격 증명을 받아야 합니다. 절차 단계에서는 시스템 관리자가 신뢰할 수 있는 호출자의 다이얼 아웃 시스템에서 PAP 자격 증명을 만든다고 가정합니다.

### 1 다이얼 아웃 시스템에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

[그림 16-3](#)에 소개된 샘플 PAP 구성을 사용하여 다이얼 아웃 시스템이 user1에게 속한다고 가정합니다.

## 2 호출자의 pap-secrets 데이터베이스를 수정합니다.

이번 릴리스는 /etc/ppp/pap-secrets 파일을 제공합니다. 여기에는 유용한 설명이 포함되어 있지만 옵션은 포함되어 있지 않습니다. 이 /etc/ppp/pap-secrets 파일에 다음과 같은 옵션을 추가할 수 있습니다.

```
user1    myserver    pass1    *
```

user1의 암호인 pass1은 읽을 수 있는 ASCII 형식으로 링크를 통해 전달됩니다. myserver는 피어를 위한 호출자 user1의 이름입니다.

## 3 다이얼 아웃 시스템에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

PAP 인증 예를 사용하여 이 다이얼 아웃 시스템이 호출자 user2에게 속한다고 가정합니다.

## 4 호출자의 pap-secrets 데이터베이스를 수정합니다.

기존 /etc/ppp/pap-secrets 파일의 끝에 다음 옵션을 추가할 수 있습니다.

```
user2    myserver    pass2    *
myserver user2      serverpass *
```

이 예에서 /etc/ppp/pap-secrets에는 두 항목이 있습니다. 첫번째 항목에는 user2가 인증을 위해 다이얼 인 서버 myserver에 전달하는 PAP 보안 자격 증명이 포함되어 있습니다.

user2에는 링크 협상의 일환으로 다이얼 인 서버에서 가져온 PAP 자격 증명 필요합니다. 따라서 /etc/ppp/pap-secrets에 myserver의 두번째 행에 있어야 하는 PAP 자격 증명도 포함됩니다.

---

주 - 대부분의 ISP는 인증 자격 증명을 제공하지 않으므로 ISP와 통신하는 경우 이전 시나리오가 현실적이지 않을 수 있습니다.

---

**참조** 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 427 페이지 “PAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)”
- 430 페이지 “신뢰할 수 있는 호출자에 대해 PAP 인증 자격 증명을 구성하는 방법”

# PAP를 위해 PPP 구성 파일 수정(다이얼 아웃 시스템)

다음 작업에서는 신뢰할 수 있는 호출자의 다이얼 아웃 시스템에서 PAP 인증을 지원하기 위해 기존 PPP 구성 파일을 업데이트하는 방법에 대해 설명합니다.

절차에서는 다음 매개변수를 사용하여 user2(그림 16-3에 소개됨)에게 속하는 다이얼 아웃 시스템에서 PAP 인증을 구성합니다. user2는 수신 호출자가 다이얼 인 myserver로부터의 호출 등을 인증하도록 요구합니다.

## ▼ PPP 구성 파일에 PAP 지원을 추가하는 방법(다이얼 아웃 시스템)

이 절차에는 408 페이지 “직렬 회선을 통해 통신을 정의하는 방법”에 소개된 PPP 구성 파일이 예로 사용됩니다. 이 절차에서는 user2에게 속하는 다이얼 아웃 시스템을 그림 16-3에 나와 있는 대로 구성합니다.

### 1 다이얼 아웃 시스템에 슈퍼 유저로 로그인합니다.

### 2 /etc/ppp/options 파일을 수정합니다.

다음 /etc/ppp/options 파일에는 PAP 지원을 위한 옵션(굵게 표시됨)이 포함되어 있습니다.

```
# cat /etc/ppp/options
lock
name user2
auth
require-pap
```

**name user2** user2를 로컬 시스템에 있는 사용자의 PAP 이름으로 설정합니다. login 옵션을 사용할 경우 PAP 이름이 암호 데이터베이스에 있는 UNIX 사용자 이름과 동일해야 합니다.

**auth** 다이얼 아웃 시스템이 링크를 설정하기 전에 호출자를 인증해야 함을 지정합니다.

---

주 - 대부분의 다이얼 아웃 시스템은 인증을 요구하지 않지만 이 다이얼 아웃 시스템은 해당 피어로부터 인증을 요구합니다. 둘 모두 허용 가능합니다.

---

**require-pap** 피어로부터 PAP 자격 증명을 요구합니다.

### 3 /etc/ppp/peers/peer-name 파일을 원격 시스템 myserver에 대해 만듭니다.

다음 예에서는 기존 /etc/ppp/peers/myserver 파일(410 페이지 “개별 피어를 사용하여 연결을 정의하는 방법”에서 만들어짐)에 PAP 지원을 추가하는 방법을 보여줍니다.

```
# cat /etc/ppp/peers/myserver
/dev/cua/a
57600
noipdefault
defaultroute
```



```
idle 120
user user2
remotename myserver
connect "chat -U 'mypassword' -f /etc/ppp/mychat"
```

굵게 표시된 새 옵션은 피어 myserver에 대한 PAP 요구 사항을 추가합니다.

user user2                      user2를 로컬 시스템의 사용자 이름으로 정의합니다.

remotename myserver      myserver를 로컬 시스템에서 인증 자격 증명을 가져와야 하는 피어로 정의합니다.

**참조** 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 다이얼 인 서버를 호출하여 PAP 인증 설정을 테스트하려면 [416 페이지 “다이얼 인 서버를 호출하는 방법”](#)을 참조하십시오.
- PAP 인증에 대한 자세한 내용은 [486 페이지 “PAP\(암호 인증 프로토콜\)”](#)를 참조하십시오.

## CHAP 인증 구성

이 절의 작업에서는 CHAP(Challenge-Handshake 인증 프로토콜)를 사용하여 PPP 링크에 대해 인증을 구현하는 방법에 대해 설명합니다. 작업에는 개인 네트워크에 대한 다이얼 업에 맞는 CHAP 시나리오를 보여주기 위해 [그림 16-4](#)에 나와 있는 예가 사용됩니다. 지침을 기반으로 사용자 사이트에서 CHAP 인증을 구현해 보십시오.

후속 절차를 수행하려면 다음 작업을 완료해야 합니다.

- 다이얼 인 서버와 신뢰할 수 있는 호출자에 속하는 다이얼 아웃 시스템 간에 다이얼 업 링크를 설정하고 테스트
- 로컬 시스템(다이얼 인 서버 또는 다이얼 아웃 시스템)에 대한 슈퍼 유저 사용 권한 획득

## CHAP 인증 설정(작업 맵)

**표 19-4** CHAP 인증 작업 맵(다이얼 인 서버)

작업	설명	수행 방법
1. 신뢰할 수 있는 모든 호출자에게 CHAP 암호 지정	CHAP 암호를 만들거나 호출자가 CHAP 암호를 만들게 합니다.	<a href="#">435 페이지 “CHAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)”</a>

표 19-4 CHAP 인증 작업 맵(다이얼 인 서버) (계속)

작업	설명	수행 방법
2. chap-secrets 데이터베이스 만들기	신뢰할 수 있는 모든 호출자에 대한 보안 자격 증명을 /etc/ppp/chap-secrets 파일에 추가합니다.	435 페이지 “CHAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)”
3. PPP 구성 파일 수정	/etc/ppp/options 및 /etc/ppp/peers/peer-name 파일에 CHAP 관련 옵션을 추가합니다.	436 페이지 “PPP 구성 파일에 CHAP 지원을 추가하는 방법(다이얼 인 서버)”

표 19-5 CHAP 인증 작업 맵(다이얼 아웃 시스템)

작업	설명	수행 방법
1. 신뢰할 수 있는 호출자의 시스템에 대한 CHAP 데이터베이스 만들기	/etc/ppp/chap-secrets에서 신뢰할 수 있는 호출자에 대한 보안 자격 증명을 만들고 필요한 경우 다이얼 아웃 시스템을 호출하는 다른 사용자에게 대한 보안 자격 증명을 만듭니다.	435 페이지 “CHAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)”
2. PPP 구성 파일 수정	/etc/ppp/options 파일에 CHAP 관련 옵션을 추가합니다.	438 페이지 “PPP 구성 파일에 CHAP 지원을 추가하는 방법(다이얼 아웃 시스템)”

## 다이얼 인 서버에서 CHAP 인증 구성

CHAP 인증을 설정하는 첫번째 작업은 /etc/ppp/chap-secrets 파일을 수정하는 것입니다. 이 파일에는 링크의 호출자를 인증하는 데 사용되는 CHAP 보안 자격 증명(CHAP 암호 포함)이 포함되어 있습니다.

주 - UNIX 또는 PAM 인증 방식은 CHAP와 함께 작동하지 않습니다. 예를 들어 PPP login 옵션을 427 페이지 “PAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)”에 설명된 대로 사용할 수 없습니다. 인증 시나리오에 PAM 또는 UNIX 스타일의 인증이 필요한 경우에는 대신 PAP를 선택하십시오.

다음 절차에서는 개인 네트워크에 있는 다이얼 인 서버에 대해 CHAP 인증을 구현합니다. PPP 링크가 외부 세계에 대한 유일한 연결입니다. 네트워크 관리자가 네트워크에 액세스할 수 있는 호출자에게만 사용 권한을 부여했습니다(시스템 관리자 포함).

## ▼ CHAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)

- 1 신뢰할 수 있는 모든 호출자의 사용자 이름이 포함된 목록을 어셈블합니다.  
신뢰할 수 있는 호출자에는 개인 네트워크를 호출할 수 있는 권한을 부여받은 모든 사람이 포함됩니다.
- 2 각 사용자에게 CHAP 암호를 지정합니다.

---

주 - 쉽게 추측할 수 없는 CHAP 암호를 선택하십시오. CHAP 암호 내용에 이 이외에 다른 제한은 없습니다.

---

CHAP 암호 지정 방법은 사이트 보안 정책에 따라 달라집니다. 즉, 관리자가 암호를 만들거나 호출자가 자신의 암호를 만들어야 합니다. 관리자가 CHAP 암호를 지정하지 않는 경우 신뢰할 수 있는 각 호출자가 만들었거나 이러한 호출자를 위해 만들어진 CHAP 암호를 받으십시오.

- 3 다이얼 인 서버에서 관리자가 됩니다.  
자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.
- 4 `/etc/ppp/chap-secrets` 파일을 수정합니다.

이번 릴리스에는 `/etc/ppp/chap-secrets` 파일이 포함되어 있습니다. 여기에는 유용한 설명이 포함되어 있지만 옵션은 포함되어 있지 않습니다. 다음과 같은 CallServe 서버 옵션을 기존 `/etc/ppp/chap-secrets` 파일의 끝에 추가할 수 있습니다.

```
account1 CallServe key123 *
account2 CallServe key456 *
```

key123 - 신뢰할 수 있는 호출자 account1의 CHAP 암호

key456 - 신뢰할 수 있는 호출자 account2의 CHAP 암호

참조 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 435 페이지 “CHAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)”
- 436 페이지 “PPP 구성 파일에 CHAP 지원을 추가하는 방법(다이얼 인 서버)”
- 436 페이지 “신뢰할 수 있는 호출자에 대해 CHAP 인증 구성(다이얼 아웃 시스템)”

## CHAP를 위해 PPP 구성 파일 수정(다이얼 인 서버)

이 절의 작업에서는 기존 PPP 구성 파일을 업데이트하여 다이얼 인 서버에서 CHAP 인증을 지원하는 방법에 대해 설명합니다.

## ▼ PPP 구성 파일에 CHAP 지원을 추가하는 방법(다이얼 인 서버)

1 다이얼 인 서버에 슈퍼 유저로 로그인합니다.

2 `/etc/ppp/options` 파일을 수정합니다.

CHAP 지원을 위해 굵게 표시된 옵션을 추가합니다.

```
# cat /etc/ppp/options
lock
nodefaultroute
name CallServe
auth
```

`name CallServe` 이 다이얼 인 서버 인스턴스에서 `CallServe`를 로컬 시스템에 있는 사용자의 CHAP 이름으로 정의합니다.

`auth` 로컬 시스템에서 링크를 설정하기 전에 호출자를 인증하게 만듭니다.

3 신뢰할 수 있는 호출자를 지원하기 위해 나머지 PPP 구성 파일을 만듭니다.

414 페이지 “다이얼 인 서버의 사용자를 구성하는 방법” 및 415 페이지 “직렬 회선을 통해 통신을 정의하는 방법(다이얼 인 서버)”을 참조하십시오.

참조 신뢰할 수 있는 호출자를 위한 CHAP 인증 자격 증명을 구성하려면 435 페이지 “CHAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)”을 참조하십시오.

## 신뢰할 수 있는 호출자에 대해 CHAP 인증 구성(다이얼 아웃 시스템)

이 절에는 신뢰할 수 있는 호출자의 다이얼 아웃 시스템에서 CHAP 인증을 설정하는 작업이 있습니다. 사이트 보안 정책에 따라 관리자나 신뢰할 수 있는 호출자가 CHAP 인증을 설정해야 할 수 있습니다.

CHAP를 구성하는 원격 호출자의 경우 호출자의 로컬 CHAP 암호가 다이얼 인 서버의 `/etc/ppp/chap-secrets` 파일에 있는 호출자의 해당 CHAP 암호와 일치해야 합니다. 그런 다음 호출자에게 이 절에 나와 있는 CHAP 구성 작업을 할당합니다.

신뢰할 수 있는 호출자에 대해 CHAP를 구성하기 위해 수행해야 하는 두 작업은 다음과 같습니다.

- 호출자의 CHAP 보안 자격 증명 만들기
- CHAP 인증을 지원하기 위해 호출자의 다이얼 아웃 시스템 구성

## ▼ 신뢰할 수 있는 호출자에 대해 CHAP 인증 자격 증명을 구성하는 방법

이 절차에서는 신뢰할 수 있는 두 호출자에 대해 CHAP 자격 증명을 설정하는 방법을 보여줍니다. 절차 단계에서는 시스템 관리자가 신뢰할 수 있는 호출자의 다이얼 아웃 시스템에서 CHAP 자격 증명을 만든다고 가정합니다.

### 1 다이얼 아웃 시스템에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

398 페이지 “CHAP 인증을 사용한 구성의 예”의 샘플 CHAP 구성을 사용하여 다이얼 아웃 시스템이 신뢰할 수 있는 호출자 account1에 속한다고 가정합니다.

### 2 호출자 account1의 chap-secrets 데이터베이스를 수정합니다.

이번 릴리스에는 /etc/ppp/chap-secrets 파일이 포함되어 있습니다. 여기에는 유용한 설명이 포함되어 있지만 옵션은 포함되어 있지 않습니다. 다음과 같은 옵션을 기존 /etc/ppp/chap-secrets 파일에 추가할 수 있습니다.

```
account1 CallServe key123 *
```

CallServe는 account1이 연결하려고 하는 피어의 이름입니다. key123은 account1과 CallServer 간의 링크에 사용될 CHAP 암호입니다.

### 3 다이얼 아웃 시스템에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

이 시스템이 호출자 account2에게 속한다고 가정합니다.

### 4 호출자 account2에 대한 /etc/ppp/chap-secrets 데이터베이스를 수정합니다.

```
account2 CallServe key456 *
```

이제 account2에 암호 key456이 해당 CHAP 자격 증명으로 설정되었습니다. 이를 피어 CallServe에 대한 링크에 사용할 수 있습니다.

**참조** 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 435 페이지 “CHAP 자격 증명 데이터베이스를 만드는 방법(다이얼 인 서버)”
- 437 페이지 “신뢰할 수 있는 호출자에 대해 CHAP 인증 자격 증명을 구성하는 방법”

## 구성 파일에 CHAP 추가(다이얼 아웃 시스템)

CHAP 인증에 대한 자세한 내용은 [489 페이지](#) “CHAP(Challenge-Handshake 인증 프로토콜)”를 참조하십시오. 다음 작업에서는 호출자 account1에 속하는 다이얼 아웃 시스템을 구성합니다. 이는 [398 페이지](#) “CHAP 인증을 사용한 구성의 예”에 소개되어 있습니다.

### ▼ PPP 구성 파일에 CHAP 지원을 추가하는 방법(다이얼 아웃 시스템)

- 1 다이얼 아웃 시스템에 슈퍼 유저로 로그인합니다.
- 2 `/etc/ppp/options` 파일에 다음과 같은 옵션이 있는지 확인합니다.  

```
# cat /etc/ppp/options
lock
nodefaultroute
```
- 3 `/etc/ppp/peers/peer-name` 파일을 원격 시스템 CallServe에 대해 만듭니다.

```
# cat /etc/ppp/peers/CallServe
/dev/cua/a
57600
noipdefault
defaultroute
idle 120
user account1
connect "chat -U 'mypassword' -f /etc/ppp/mychat"
```

user account1 옵션은 account1을 CallServe에 제공할 CHAP 사용자 이름으로 설정합니다. 이전 파일의 기타 옵션에 대한 자세한 내용은 유사한 `/etc/ppp/peers/myserver` 파일([410 페이지](#) “개별 피어를 사용하여 연결을 정의하는 방법”)을 참조하십시오.

**참조** 다이얼 인 서버를 호출하여 CHAP 인증을 테스트하려면 [416 페이지](#) “다이얼 인 서버를 호출하는 방법”을 참조하십시오.

## PPPoE 터널 설정(작업)

이 장에는 PPPoE 터널의 각 끝에서 참가자(PPPoE 클라이언트 및 PPPoE 액세스 서버)를 설정하는 작업이 포함되어 있습니다. 관련 항목은 다음과 같습니다.

- 439 페이지 “PPPoE 터널을 설정하는 주요 작업(작업 맵)”
- 440 페이지 “PPPoE 클라이언트 설정”
- 443 페이지 “PPPoE 액세스 서버 설정”

작업에는 400 페이지 “PPPoE 터널을 통한 DSL 지원 계획”에서 예로 소개된 시나리오가 사용됩니다. PPPoE의 개요는 386 페이지 “PPPoE를 통한 DSL 사용자 지원”을 참조하십시오.

## PPPoE 터널을 설정하는 주요 작업(작업 맵)

다음 표에는 PPPoE 클라이언트 및 PPPoE 액세스 서버를 구성하는 주요 작업이 나열되어 있습니다. 사이트에서 PPPoE를 구현하려면 PPPoE 터널의 본인 쪽 끝(클라이언트측 또는 액세스 서버측)만 설정하면 됩니다.

표 20-1 PPPoE 클라이언트 설정 작업 맵

작업	설명	수행 방법
1. PPPoE용 인터페이스 구성	PPPoE 터널에 사용할 이더넷 인터페이스를 정의합니다.	440 페이지 “PPPoE 클라이언트용 인터페이스를 구성하는 방법”
2. PPPoE 액세스 서버에 대한 정보 구성	PPPoE 터널의 서비스 공급자 쪽 끝에서 액세스 서버에 대한 매개변수를 정의합니다.	441 페이지 “PPPoE 액세스 서버 피어를 정의하는 방법”
3. PPP 구성 파일 설정	클라이언트용 PPP 구성 파일을 아직 정의하지 않은 경우 하나 정의합니다.	408 페이지 “직렬 회선을 통해 통신을 정의하는 방법”
4. 터널 만들기	액세스 서버를 호출합니다.	441 페이지 “PPPoE 액세스 서버 피어를 정의하는 방법”

표 20-2 PPPoE 액세스 서버 설정 작업 맵

작업	설명	수행 방법
1. PPPoE 액세스 서버 설정	PPPoE 터널에 사용할 이더넷 인터페이스를 정의하고 액세스 서버가 제공하는 서비스를 정의합니다.	443 페이지 “PPPoE 액세스 서버를 설정하는 방법”
2. PPP 구성 파일 설정	클라이언트용 PPP 구성 파일을 아직 정의하지 않은 경우 하나 정의합니다.	414 페이지 “다이얼 인 서버를 통한 통신 구성”
3. (옵션) 인터페이스 사용 제한	PPPoE 옵션 및 PAP 인증을 사용하여 특정 클라이언트만이 특정 이더넷 인터페이스를 사용할 수 있도록 제한합니다.	444 페이지 “특정 클라이언트만 인터페이스를 사용할 수 있도록 제한하는 방법”

## PPPoE 클라이언트 설정

DSL을 통해 클라이언트 시스템에 PPP를 제공하려면 먼저 모뎀 또는 허브에 연결된 인터페이스에서 PPPoE를 구성해야 합니다. 그런 다음 PPP 구성 파일을 변경하여 PPPoE의 반대쪽 끝에서 액세스 서버를 정의해야 합니다.

### PPPoE 클라이언트 설정을 위한 필수 조건

PPPoE 클라이언트를 설정하려면 먼저 다음 작업을 완료해야 합니다.

- PPPoE 터널을 사용할 클라이언트 시스템에 Oracle Solaris 릴리스 설치
- 서비스 공급자에 해당 PPPoE 액세스 서버에 대한 정보 문의
- 전화 회사 또는 서비스 공급자에게 클라이언트 시스템에 사용되는 장치 조립 요청. 이러한 장치에는 관리자가 아니라 전화 회사가 조립하는 DSL 모뎀, 분리기 등이 포함됩니다.

### ▼ PPPoE 클라이언트용 인터페이스를 구성하는 방법

이 절차를 사용하여 PPPoE 터널에 사용할 이더넷 인터페이스를 정의할 수 있습니다.

#### 1 PPPoE 클라이언트에서 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스**의 “관리 권한을 얻는 방법”을 참조하십시오.



- 2 DSL 연결이 있는 이더넷 인터페이스의 이름을 `/etc/ppp/pppoe.if` 파일에 추가합니다.

예를 들어, 다음 항목을 `/etc/ppp/pppoe.if`에 추가합니다. 이는 `hme0`을 DSL 모뎀에 연결된 네트워크 인터페이스로 사용하는 PPPoE 클라이언트의 경우입니다.

```
hme0
```

`/etc/ppp/pppoe.if`에 대한 자세한 내용을 보려면 [495 페이지 “/etc/ppp/pppoe.if 파일”](#)로 이동하십시오.

- 3 PPPoE용 인터페이스를 구성합니다.

```
# /etc/init.d/pppd start
```

- 4 (옵션) 인터페이스가 이제 PPPoE에 연결되었는지 확인합니다.

```
# /usr/sbin/sppptun query
hme0:pppoe
hme0:pppoed
```

`/usr/sbin/sppptun` 명령을 사용하여 인터페이스를 PPPoE에 수동으로 연결할 수도 있습니다. 지침은 [495 페이지 “/usr/sbin/sppptun 명령”](#)을 참조하십시오.

## ▼ PPPoE 액세스 서버 피어를 정의하는 방법

액세스 서버는 `/etc/ppp/peers/peer-name` 파일에서 정의합니다. 액세스 서버에 사용되는 옵션 중 상당수가 다이얼 업 시나리오에서 다이얼 인 서버를 정의하는 데에도 사용됩니다. `/etc/ppp/peers.peer-name`에 대한 자세한 내용은 [474 페이지 “/etc/ppp/peers/peer-name 파일”](#)을 참조하십시오.

- 1 PPPoE 클라이언트에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 서비스 공급자의 PPPoE 액세스 서버를 `/etc/ppp/peers/peer-name` 파일에서 정의합니다.

예를 들어, `/etc/ppp/peers/dslserve` 파일은 Far ISP의 액세스 서버 `dslserve`([401 페이지 “PPPoE 터널 구성의 예”](#)에서 소개됨)를 정의합니다.

```
# cat /etc/ppp/peers/dslserve
sppptun
plugin pppoe.so
connect "/usr/lib/inet/pppoec hme0"
noccp
noauth
user Red
password redsecret
noipdefault
defaultroute
```

이 파일의 옵션에 대한 정의를 보려면 [502 페이지 “액세스 서버 피어를 정의하기 위한 /etc/ppp/peers/peer-name 파일”](#)로 이동하십시오.

### 3 PPPoE 클라이언트에서 다른 PPP 구성 파일을 수정합니다.

a. `/etc/ppp/options`를 구성합니다. 이때 406 페이지 “다이얼 아웃 시스템 구성”에 나와 있는 다이얼 아웃 시스템 구성 지침을 따르십시오.

b. `/etc/ppp/options.sppptun` 파일을 만듭니다. `/etc/ppp/options.sppptun`은 PPPoE에 연결되는 인터페이스가 연결되는 직렬 포트에 대한 PPP 옵션을 정의합니다.

`/etc/ppp/options.ttyname` 파일( 471 페이지 “`/etc/ppp/options.ttyname` 구성 파일”에 설명되어 있음)에 대해 사용할 수 있는 모든 옵션을 사용할 수 있습니다.

`/etc/ppp/options.sppptun` 파일의 이름을 지정해야 합니다. 이는 sppptun이 pppd 구성에서 지정된 장치 이름이기 때문입니다.

### 4 모든 사용자가 클라이언트에서 PPP를 시작할 수 있는지 확인합니다.

```
# touch /etc/ppp/options
```

### 5 PPP를 DSL 회선을 통해 실행할 수 있는지 테스트합니다.

```
% pppd debug updetach call dslserve
```

`dslserve`는 ISP의 액세스 서버( 401 페이지 “PPPoE 터널 구성의 예”에 나와 있음)에 지정되는 이름입니다. `debug updetach` 옵션을 사용하면 디버깅 정보가 단말기 창에 표시됩니다.

PPP가 올바르게 실행되고 있는 경우 단말기 출력에서 링크가 활성 상태가 됩니다. PPP가 계속 실행되지 않는 경우 다음 명령을 실행하여 서버가 올바르게 실행되고 있는지 확인합니다.

```
# /usr/lib/inet/pppoc -i hme0
```

---

주 - 구성된 PPPoE 클라이언트의 사용자는 다음을 입력하여 DSL 회선을 통해 PPP 실행을 시작할 수 있습니다.

```
% pppd call ISP-server-name
```

그런 다음 사용자는 응용 프로그램 또는 서비스를 실행할 수 있습니다.

---

참조 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- 440 페이지 “PPPoE 클라이언트 설정”을 참조하십시오.
- 494 페이지 “DSL 지원을 위해 PPPoE 터널 만들기”를 참조하십시오.
- 21 장, “일반적인 PPP 문제 해결(작업)”을 참조하십시오.
- 443 페이지 “PPPoE 액세스 서버 설정”을 참조하십시오.

# PPPoE 액세스 서버 설정

서비스 공급업체의 경우 DSL 연결을 통해 사이트에 연결하는 클라이언트에 인터넷 및 기타 서비스를 제공할 수 있습니다. 절차에서는 PPPoE 터널에 사용할 서버의 인터페이스를 결정하고 사용자에게 제공할 서비스를 정의합니다.

## ▼ PPPoE 액세스 서버를 설정하는 방법

이 절차를 사용하여 PPPoE 터널에 사용할 이더넷 인터페이스를 정의하고 액세스 서버가 제공하는 서비스를 구성할 수 있습니다.

### 1 액세스 서버에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 PPPoE 터널 전용 이더넷 인터페이스의 이름을 /etc/ppp/pppoe.if 파일에 추가합니다.

예를 들어, /etc/ppp/pppoe.if 파일을 액세스 서버 dslserve(401 페이지 “PPPoE 터널 구성의 예”에 나와 있음)에 사용합니다.

```
# cat /etc/ppp/pppoe.if
hme1
hme2
```

### 3 액세스 서버가 제공하는 전역 서비스를 /etc/ppp/pppoe 파일에서 정의합니다.

/etc/ppp/pppoe 파일은 액세스 서버 dslserve(그림 16-5에 나와 있음)가 제공하는 서비스를 나열합니다.

```
device hme1,hme2
service internet
    pppd "proxyarp 192.168.1.1:"
service debugging
    pppd "debug proxyarp 192.168.1.1:"
```

파일 예에서 인터넷 서비스는 dslserve의 이더넷 인터페이스 hme1 및 hme2에 대해 공지됩니다. 디버깅은 이더넷 인터페이스에서 PPP 링크에 대해 켜집니다.

### 4 PPP 구성 파일을 다이얼 인 서버와 동일한 방식으로 설정합니다.

자세한 내용은 [492 페이지 “호출자를 위한 IP 주소 지정 체계 만들기”](#)를 참조하십시오.

### 5 pppoe 데몬을 시작합니다.

```
# /etc/init.d/pppd start
```

pppd는 /etc/ppp/pppoe.if에 나열되어 있는 인터페이스도 연결합니다.

### 6 (옵션) 서버의 인터페이스가 PPPoE에 연결되었는지 확인합니다.

```
# /usr/sbin/sppptun query
hme1:pppoe
```

```
hme1:pppoed
hme2:pppoe
hme2:pppoed
```

이전 샘플에서는 인터페이스 hme1 및 hme2가 현재 PPPoE에 연결되어 있음을 보여줍니다. /usr/sbin/sppptun 명령을 사용하여 인터페이스를 PPPoE에 수동으로 연결할 수도 있습니다. 지침은 [495 페이지 “/usr/sbin/sppptun 명령”](#)을 참조하십시오.

## ▼ 기존 /etc/ppp/pppoe 파일을 수정하는 방법

- 1 액세스 서버에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 필요에 따라 /etc/ppp/pppoe를 수정합니다.

- 3 pppoe 데몬이 새 서비스를 인식하게 만듭니다.

```
# pkill -HUP pppoe
```

## ▼ 특정 클라이언트만 인터페이스를 사용할 수 있도록 제한하는 방법

다음 절차에서는 한 PPPoE 클라이언트 그룹으로 인터페이스를 제한하는 방법을 보여줍니다. 이 작업을 수행하기 전에 인터페이스에 지정하는 클라이언트의 실제 이더넷 MAC 주소를 가져와야 합니다.

---

주 - 일부 시스템의 경우 이더넷 인터페이스에서 MAC 주소를 변경할 수 있습니다. 그러나 이 기능은 보안 조치가 아닌 편리한 옵션으로 보아야 합니다.

---

[401 페이지 “PPPoE 터널 구성의 예”](#)에 나와 있는 예를 사용하여 이러한 단계에서는 ds1serve의 인터페이스 중 하나인 hme1을 MiddleCo의 클라이언트를 위해 예약하는 방법을 보여줍니다.

- 1 [443 페이지 “PPPoE 액세스 서버를 설정하는 방법”](#)에 나와 있는 대로 액세스 서버의 인터페이스를 구성하고 서비스를 정의합니다.
- 2 클라이언트에 대한 항목을 서버의 /etc/ethers 데이터베이스에 만듭니다.

Red, Blue 및 Yellow라는 클라이언트에 대한 샘플 항목은 다음과 같습니다.

```
8:0:20:1:40:30 redether
8:0:20:1:40:10 yellowether
8:0:20:1:40:25 blueether
```

샘플에서는 redether, yellowether 및 blueether라는 심볼릭 이름을 Red, Yellow 및 Blue라는 클라이언트의 이더넷 주소에 지정합니다. MAC 주소에 심볼릭 이름을 지정하는 작업은 선택 사항입니다.

### 3 /etc/ppp/pppoe.device 파일에서 다음 정보를 정의하여 특정 인터페이스에서 제공되는 서비스를 제한합니다.

이 파일에서 **장치**는 정의할 장치의 이름입니다.

```
# cat /etc/ppp/pppoe.hme1
service internet
    pppd "name dslserve-hme1"
        clients redether,yellowether,blueether
```

dslserve-hme1은 액세스 서버의 이름으로, pap-secrets 파일의 일치하는 항목에 사용됩니다. clients 옵션을 사용하면 hme1 인터페이스의 사용이 redether, yellowether 및 blueether라는 심볼릭 이더넷 이름이 지정된 클라이언트로 제한됩니다.

/etc/ethers에서 클라이언트의 MAC 주소를 위한 심볼릭 이름을 정의하지 않은 경우에는 숫자 주소를 clients 옵션의 인수로 사용할 수 있습니다. 와일드카드도 허용됩니다.

예를 들어, clients 8:0:20:\*:~\*와 같이 숫자 주소를 지정할 수 있습니다. 와일드카드를 사용하면 /etc/ethers에서 일치하는 모든 주소가 허용됩니다.

### 4 액세스 서버를 위한 /etc/ppp/pap-secrets 파일을 만듭니다.

```
Red          dslserve-hme1    redpasswd    *
Blue         dslserve-hme1    bluepasswd   *
Yellow       dslserve-hme1    yellowpasswd *
```

항목은 dslserve의 hme1 인터페이스를 통해 PPP를 실행하도록 허용된 클라이언트의 PAP 이름 및 암호입니다.

PAP 인증에 대한 자세한 내용은 426 페이지 “PAP 인증 구성”을 참조하십시오.

**참조** 다음 목록에는 관련 정보에 대한 참조가 나와 있습니다.

- PPPoE에 대한 자세한 내용은 494 페이지 “DSL 지원을 위해 PPPoE 터널 만들기”를 참조하십시오.
- PPPoE 및 PPP 문제를 해결하려면 451 페이지 “PPP 관련 문제 및 PPPoE 관련 문제 해결”을 참조하십시오.
- PPPoE 클라이언트를 구성하려면 440 페이지 “PPPoE 클라이언트 설정”을 참조하십시오.
- 클라이언트를 위해 PAP 인증을 구성하려면 430 페이지 “신뢰할 수 있는 호출자에 대해 PAP 인증 구성(다이얼 아웃 시스템)”을 참조하십시오.
- 서버에서 PAP 인증을 구성하려면 427 페이지 “다이얼 인 서버에서 PAP 인증 구성”을 참조하십시오.



## 일반적인 PPP 문제 해결(작업)

이 장에는 Solaris PPP 4.0에서 발생하는 일반적인 문제를 해결하는 방법에 대한 정보가 포함되어 있습니다. 다음 항목을 다룹니다.

- 448 페이지 “PPP 문제 해결 도구”
- 451 페이지 “PPP 관련 문제 및 PPPoE 관련 문제 해결”
- 463 페이지 “전용 회선 문제 해결”
- 463 페이지 “인증 문제 진단 및 해결”

*PPP Design, Implementation, and Debugging*(James Carlson 저) 및 Australian National University 웹 사이트와 같은 소스를 통해서도 PPP 문제를 해결하기 위한 상세 조언을 구할 수 있습니다. 자세한 내용은 377 페이지 “PPP에 대한 전문 참조 설명서” 및 377 페이지 “PPP에 대한 웹 사이트”를 참조하십시오.

## PPP 문제 해결(작업 맵)

다음 작업 맵을 사용하면 일반적인 PPP 문제에 대한 조언 및 해결 방법에 빠르게 액세스할 수 있습니다.

표 21-1 PPP 문제 해결 작업 맵

작업	정의	수행 방법
PPP 링크에 대한 진단 정보 획득	PPP 진단 도구를 사용하여 문제 해결을 위한 출력을 얻을 수 있습니다.	449 페이지 “pppd에서 진단 정보를 가져오는 방법”
PPP 링크에 대한 디버깅 정보 획득	pppd debug 명령을 사용하여 문제 해결을 위한 출력을 생성할 수 있습니다.	450 페이지 “PPP 디버깅을 켜는 방법”
네트워크 계층에 발생하는 일반적인 문제 해결	일련의 검사를 통해 네트워크와 관련된 PPP 문제를 식별하고 해결합니다.	451 페이지 “네트워크 문제를 진단하는 방법”

표 21-1 PPP 문제 해결 작업 맵 (계속)

작업	정의	수행 방법
일반적인 통신 문제 해결	PPP 링크에 영향을 주는 통신 문제를 식별하고 해결합니다.	454 페이지 “통신 문제를 진단하고 해결하는 방법”
구성 문제 해결	PPP 구성 파일의 문제를 식별하고 해결합니다.	455 페이지 “PPP 구성을 사용하여 문제를 진단하는 방법”
모뎀 관련 문제 해결	모뎀 문제를 식별하고 해결합니다.	456 페이지 “모뎀 문제를 진단하는 방법”
채트스크립트 관련 문제 해결	다이얼 아웃 시스템에서 채트스크립트 문제를 식별하고 해결합니다.	457 페이지 “채트스크립트에 대한 디버깅 정보를 가져오는 방법”
직렬 회선 속도 문제 해결	다이얼 인 서버에서 회선 속도 문제를 식별하고 해결합니다.	459 페이지 “직렬 회선 속도 문제를 진단하고 해결하는 방법”
전용 회선에 일반적으로 발생하는 문제 해결	전용 회선의 성능 문제를 식별하고 해결합니다.	463 페이지 “전용 회선 문제 해결”
인증 관련 문제 해결	인증 데이터베이스와 관련된 문제를 식별하고 해결합니다.	463 페이지 “인증 문제 진단 및 해결”
PPPoE의 문제 영역 해결	PPP 진단 도구를 사용하여 PPPoE 문제를 식별하고 해결하기 위한 출력을 얻을 수 있습니다.	460 페이지 “PPPoE에 대한 진단 정보를 가져오는 방법”

## PPP 문제 해결 도구

일반적으로 PPP 링크의 경우 다음과 같은 세 영역에서 주로 오류가 발생합니다.

- 설정할 링크에 오류 발생
- 정상적으로 사용 중 링크의 성능 저하
- 링크의 각 측에 있는 네트워크가 원인인 문제

PPP의 작동 여부를 알아내는 가장 쉬운 방법은 링크를 통해 명령을 실행하는 것입니다. ping 또는 traceroute와 같은 명령을 피어의 네트워크에 있는 호스트에 대해 실행한 다음 결과를 관찰하십시오. 그러나 PPP 및 UNIX 디버깅 도구를 사용하여 설정된 링크의 성능을 모니터링하거나 문제가 되는 링크의 문제를 해결해야 합니다.

이 절에서는 `pppd` 및 연관된 해당 로그 파일에서 진단 정보를 가져오는 방법에 대해 설명합니다. 이 장의 나머지 절에서는 PPP 문제 해결 도구를 사용하여 발견하고 해결할 수 있는 일반적인 PPP 문제에 대해 설명합니다.



## ▼ pppd에서 진단 정보를 가져오는 방법

다음 절차에서는 로컬 시스템에서 링크의 현재 작업을 보는 방법을 보여줍니다.

### 1 로컬 시스템에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 PPP에 대해 구성된 직렬 장치를 인수로 사용하여 pppd를 실행합니다.

```
# pppd cua/b debug updetach
```

다음 예에서는 pppd를 전면에서 실행할 때 다이얼 업 링크 및 전용 회선 링크에 대해 표시되는 결과를 보여줍니다. pppd debug를 배경에서 실행하면 생성되는 출력이 /etc/ppp/connect-errors 파일로 전송됩니다.

#### 예 21-1 제대로 작동하는 다이얼 업 링크의 출력

```
# pppd /dev/cua/b debug updetach
have route to 0.0.0.0/0.0.0.0 via 172.21.0.4
serial speed set to 230400 bps
Using interface sppp0
Connect: sppp0 <-> /dev/cua/b
sent [LCP ConfReq id=0x7b <asynmap 0x0> <magic 0x73e981c8> <pcomp> <accomp>]
rcvd [LCP Ident id=0x79 magic=0x0 "ppp-2.4.0b1 (Sun Microsystems, Inc., Oct 6
2004 09:36:22)"]
Peer Identification: ppp-2.4.0b1 (Sun Microsystems, Inc., Oct 6 2004 09:36:22)
rcvd [LCP ConfRej id=0x7b <asynmap 0x0>]
sent [LCP Ident id=0x7c magic=0x0 "ppp-2.4.0b1 (Sun Microsystems, Inc., Sep 15
2004 09:38:33)"]
sent [LCP ConfReq id=0x7d <magic 0x73e981c8> <pcomp> <accomp>]
rcvd [LCP ConfAck id=0x7d <magic 0x73e981c8> <pcomp> <accomp>]
rcvd [LCP ConfAck id=0x78 <magic 0xdd4ad820> <pcomp> <accomp>]
sent [LCP ConfAck id=0x78 <magic 0xdd4ad820> <pcomp> <accomp>]
sent [LCP Ident id=0x7e magic=0x73e981c8 "ppp-2.4.0b1 (Sun Microsystems, Inc.,
Sep 15 2004 09:38:33)"]
sent [IPCP ConfReq id=0x3d <addr 0.0.0.0> <compress VJ 0f 01>]
rcvd [LCP Ident id=0x7a magic=0xdd4ad820 "ppp-2.4.0b1 (Sun Microsystems, Inc.,
Oct 6 2004 09:36:22)"]
Peer Identification: ppp-2.4.0b1 (Sun Microsystems, Inc., Oct 6 2004 09:36:22)
rcvd [IPCP ConfReq id=0x92 <addr 10.0.0.1> <compress VJ 0f 01>]
sent [IPCP ConfAck id=0x92 <addr 10.0.0.1> <compress VJ 0f 01>]
rcvd [IPCP ConfNak id=0x3d <addr 10.0.0.2>]]
sent [IPCP ConfReq id=0x3e <addr 10.0.0.2> <compress VJ 0f 01>]
rcvd [IPCP ConfAck id=0x3e <addr 10.0.0.2> <compress VJ 0f 01>]
local IP address 10.0.0.2
remote IP address 10.0.0.1
```

#### 예 21-2 제대로 작동하는 전용 회선 링크의 출력

```
# pppd /dev/se_hdlc1 default-asynmap debug updetach
pppd 2.4.0b1 (Sun Microsystems, Inc., Oct 24 2004 07:13:18) started by root, uid 0
synchronous speed appears to be 0 bps
```

```

init option: '/etc/ppp/peers/syncinit.sh' started (pid 105122)
Serial port initialized.
synchronous speed appears to be 64000 bps
Using interface sppp0
Connect: sppp0 <-> /dev/se_hdlc1
sent [LCP ConfReq id=0xe9 <magic 0x474283c6><pcomp> <accomp>]
rcvd [LCP ConfAck id=0xe9 <magic 0x474283c6><pcomp> <accomp>]
rcvd [LCP ConfReq id=0x22 <magic 0x8e3a53ff><pcomp> <accomp>]
sent [LCP ConfReq id=0x22 <magic 0x8e3a53ff><pcomp> <accomp>]
sent [LCP Ident id=0xea magic=0x474283c6 "ppp-2.4.0b1 (Sun Microsystems, Inc., Oct
22 2004 14:31:44)"]
sent [IPCP ConfReq id=0xf7 <addr 0.0.0.0> <compress VJ Of ol>]]
sent [CCP ConfReq id=0x3f <deflate 15> <deflate(old#) 15> <bsd v1 15>]
rcvd [LCP Ident id=0x23 magic=0x8e3a53ff "ppp-2.4.0b1 (Sun Microsystems, Inc., Oct
22 2004 14:31:44)"]
Peer Identification: ppp-2.4.0b1 (Sun Microsystems, Inc., Oct 22 2004 14:31:44)
rcvd [IPCP ConfReq id=0x25 <addr 10.0.0.1> <compress VJ Of 01>]
sent [IPCP ConfAck id=0x25 <addr 10.0.0.1> <compress VJ Of 01>]
rcvd [CCP ConfReq id=0x3 <deflate 15> <deflate(old#) 15 <bsd v1 15>]
sent [CCP ConfAck id=0x3 <deflate 15> <deflate(old#) 15 <bsd v1 15>]
rcvd [IPCP ConfNak id=0xf8 <addr 10.0.0.2>]
rcvd [IPCP ConfReq id=0xf7 <addr 10.0.0.2> <compress VJ Of 01>]
rcvd [CCP ConfAck id=0x3f <deflate 15> <deflate(old#) 15 <bsd v1 15>]
Deflate (15) compression enabled
rcvd [IPCP ConfAck id=0xf8 <addr 10.0.0.2> <compress VJ Of 01>]
local IP address 10.0.0.2
remote IP address 10.0.0.1

```

## ▼ PPP 디버깅을 켜는 방법

다음 작업에서는 pppd 명령을 사용하여 디버깅 정보를 가져오는 방법을 보여줍니다.

---

주 - 각 호스트에 대해 1단계에서 3단계까지를 한번씩만 수행하면 됩니다. 그런 다음 4단계를 진행하여 호스트에 대해 디버깅을 켤 수 있습니다.

---

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 pppd의 출력을 보관할 로그 파일을 만듭니다.

```
# touch /var/log/pppdebug
```

### 3 /etc/syslog.conf에서 pppd에 대해 다음 syslog 기능을 추가합니다.

```
daemon.debug;local2.debug          /var/log/pppdebug
```

### 4 syslogd를 다시 시작합니다.

```
# pkill -HUP -x syslogd
```

- 5 다음 `pppd` 구문을 사용하여 특정 피어에 대한 호출에 대해 디버깅을 켭니다.

```
# pppd debug call peer-name
```

`peer-name`은 `/etc/ppp/peers` 디렉토리에 있는 파일의 이름이어야 합니다.

- 6 로그 파일의 내용을 봅니다.

```
# tail -f /var/log/pppdebug
```

로그 파일의 예는 [단계 3](#)을 참조하십시오.

## PPP 관련 문제 및 PPPoE 관련 문제 해결

PPP 관련 문제 및 PPPoE 관련 문제를 해결하는 방법에 대한 자세한 내용은 다음 절을 참조하십시오.

- 451 페이지 “네트워크 문제를 진단하는 방법”
- 453 페이지 “PPP에 영향을 주는 일반적인 네트워크 문제”
- 454 페이지 “통신 문제를 진단하고 해결하는 방법”
- 454 페이지 “PPP에 영향을 주는 일반적인 통신 문제”
- 455 페이지 “PPP 구성을 사용하여 문제를 진단하는 방법”
- 456 페이지 “일반적인 PPP 구성 문제”
- 456 페이지 “모뎀 문제를 진단하는 방법”
- 457 페이지 “채트 스크립트에 대한 디버깅 정보를 가져오는 방법”
- 457 페이지 “일반적인 채트 스크립트 문제”
- 459 페이지 “직렬 회선 속도 문제를 진단하고 해결하는 방법”
- 460 페이지 “PPPoE에 대한 진단 정보를 가져오는 방법”

### ▼ 네트워크 문제를 진단하는 방법

PPP 링크가 활성 상태가 되지만 원격 네트워크에서 연결할 수 있는 네트워크가 거의 없는 경우 네트워크 문제가 표시될 수 있습니다. 다음 절차에서는 PPP 링크에 영향을 주는 네트워크 문제를 격리하고 해결하는 방법을 보여줍니다.

- 1 로컬 시스템에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 문제가 되는 링크를 종료합니다.

- 3 PPP 구성에 다음 옵션을 추가하여 구성 파일에서 모든 선택적 프로토콜을 사용 안함으로 설정합니다.

```
noccp novj nopcomp noaccomp default-asynmap
```

이러한 옵션은 사용 가능한 PPP 중 압축되지 않은 가장 단순한 PPP를 제공합니다. 명령줄에서 `pppd`에 대한 인수로 이러한 옵션을 호출해 보십시오. 이전에 연결할 수 없었던 호스트에 연결할 수 있는 경우 다음 위치 중 하나에 옵션을 추가합니다.

- `/etc/ppp/peers/peer-name`에서 `call` 옵션 뒤
- `/etc/ppp/options`(옵션이 전역적으로 적용되는지 확인)

#### 4 원격 피어를 호출합니다. 그런 다음 디버깅 기능을 사용으로 설정합니다.

```
% pppd debug call peer-name
```

#### 5 `chat`의 `-v` 옵션을 사용하여 `chat` 프로그램에서 상세 정보 로그를 가져옵니다.

예를 들어, 모든 PPP 구성 파일에서 다음 형식을 사용합니다.

```
connect 'chat -v -f /etc/ppp/chatfile'
```

`/etc/ppp/chatfile`은 채트 파일의 이름을 나타냅니다.

#### 6 Telnet 또는 기타 응용 프로그램을 통해 원격 호스트에 연결하여 문제를 재현해 봅니다.

디버깅 로그를 관찰합니다. 계속 원격 호스트에 연결할 수 없으면 PPP 문제가 네트워크 관련 문제일 수 있습니다.

#### 7 원격 호스트의 IP 주소가 등록된 인터넷 주소인지 확인합니다.

일부 조직에서는 로컬 네트워크 내에서 알려져 있지만 인터넷으로 경로를 지정할 수 없는 내부 IP 주소를 지정합니다. 원격 호스트가 회사 내에 있는 경우 NAT(이름-주소 변환) 서버 또는 프록시 서버를 설정해야 인터넷에 연결할 수 있습니다. 원격 호스트가 회사 내에 있지 않은 경우에는 원격 조직에 문제를 보고해야 합니다.

#### 8 경로 지정 테이블을 검사합니다.

a. 로컬 시스템과 피어 모두에서 경로 지정 테이블을 확인합니다.

b. 피어에서 원격 시스템으로 이어지는 경로에 있는 모든 라우터에 대해 경로 지정 테이블을 확인합니다. 또한 다시 피어로 이어지는 경로에 있는 모든 라우터에 대해 경로 지정 테이블을 확인합니다.

중간 라우터가 잘못 구성되지 않았는지 확인합니다. 다시 피어로 이어지는 경로에서 문제가 발견되는 경우가 많습니다.

#### 9 (옵션) 시스템이 라우터인 경우 선택적 기능을 확인합니다.

```
# ndd -set /dev/ip ip_forwarding 1
```

`ndd`에 대한 자세한 내용은 `ndd(1M)` 매뉴얼 페이지를 참조하십시오.

Solaris 10 릴리스에서는 `ndd(1M)` 대신 `routeadm(1M)`을 사용할 수 있습니다.

```
# routeadm -e ipv4-forwarding -u
```

주 - `ndd` 명령은 지속되지 않습니다. 이 명령을 사용하여 설정하는 값은 시스템을 재부트할 때 손실됩니다. `routeadm` 명령은 지속됩니다. 이 명령을 사용하여 설정하는 값은 시스템을 재부트한 후에도 유지됩니다.

#### 10 netstat -s 및 유사한 도구를 통해 얻은 통계를 확인합니다.

netstat에 대한 자세한 내용은 [netstat\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

a. 로컬 시스템에서 통계를 실행합니다.

b. 피어를 호출합니다.

c. netstat -s를 통해 생성된 새로운 통계를 관찰합니다.

자세한 내용은 [453 페이지](#) “PPP에 영향을 주는 일반적인 네트워크 문제”를 참조하십시오.

#### 11 DNS 구성을 확인합니다.

잘못된 이름 서비스 구성을 사용하면 IP 주소를 확인할 수 없기 때문에 응용 프로그램에 오류가 발생합니다.

## PPP에 영향을 주는 일반적인 네트워크 문제

netstat -s에 의해 생성되는 메시지를 사용하여 다음 표에 나와 있는 네트워크 문제를 해결할 수 있습니다. 관련 절차 정보는 [451 페이지](#) “네트워크 문제를 진단하는 방법”을 참조하십시오.

표 21-2 PPP에 영향을 주는 일반적인 네트워크 문제

Message	문제점	해결 방법
IP packets not forwardable(IP 패킷 전달 불가)	로컬 호스트에 경로가 없습니다.	로컬 호스트의 경로 지정 테이블에 누락된 경로를 추가합니다.
ICMP input destination unreachable(ICMP 입력 대상에 연결할 수 없음)	로컬 호스트에 경로가 없습니다.	로컬 호스트의 경로 지정 테이블에 누락된 경로를 추가합니다.
ICMP time exceeded(ICMP 시간 초과)	두 라우터가 동일한 대상 주소를 서로에게 전달하고 있어 TTL(활성 시간) 값이 초과될 때까지 패킷이 앞뒤로 재발송됩니다.	tracert를 사용하여 경로 지정 루프의 소스를 찾은 다음 오류가 발생한 라우터의 관리자에게 문의합니다. tracert에 대한 자세한 내용은 <a href="#">tracert(1M)</a> 매뉴얼 페이지를 참조하십시오.

표 21-2 PPP에 영향을 주는 일반적인 네트워크 문제 (계속)

Message	문제점	해결 방법
IP packets not forwardable(IP 패킷 전달 불가)	로컬 호스트에 경로가 없습니다.	로컬 호스트의 경로 지정 테이블에 누락된 경로를 추가합니다.
ICMP input destination unreachable(ICMP 입력 대상에 연결할 수 없음)	로컬 호스트에 경로가 없습니다.	로컬 호스트의 경로 지정 테이블에 누락된 경로를 추가합니다.

## ▼ 통신 문제를 진단하고 해결하는 방법

통신 문제는 두 피어가 성공적으로 링크를 설정할 수 없을 때 발생합니다. 실제로는 이러한 문제가 잘못 구성된 채트 스크립트로 인해 발생하는 협상 문제인 경우도 있습니다. 다음 절차에서는 통신 문제를 해결하는 방법을 보여줍니다. 잘못된 채트 스크립트로 인해 발생하는 협상 문제를 해결하려면 표 21-5를 참조하십시오.

### 1 로컬 시스템에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 피어를 호출합니다.

### 3 원격 피어를 호출합니다. 그런 다음 디버깅 기능을 사용으로 설정합니다.

```
% pppd debug call peer-name
```

특정 통신 문제를 해결하려면 피어에서 디버깅 정보를 가져와야 할 수도 있습니다.

### 4 결과 로그에서 통신 문제를 확인합니다. 자세한 내용은 [454 페이지 “PPP에 영향을 주는 일반적인 통신 문제”](#)를 참조하십시오.

## PPP에 영향을 주는 일반적인 통신 문제

다음 표에는 [454 페이지 “통신 문제를 진단하고 해결하는 방법”](#) 절차의 로그 출력과 관련된 증상이 설명되어 있습니다.

표 21-3 PPP에 영향을 주는 일반적인 통신 문제

증상	문제점	해결 방법
구성-요청이 너무 많습니다.	한 피어가 다른 피어를 들을 수 없습니다.	다음 문제가 있는지 확인합니다. <ul style="list-style-type: none"> <li>■ 시스템 또는 모뎀의 케이블 연결이 잘못되었을 수 있습니다.</li> <li>■ 모뎀 구성에 잘못된 비트 설정이 있을 수 있습니다. 또는 구성에 중단된 흐름 제어가 있을 수 있습니다.</li> <li>■ 체트 스크립트에 오류가 발생했을 수 있습니다. 이 경우 표 21-5를 참조하십시오.</li> </ul>
pppd debug 출력에 LCP가 시작되지만 상위 레벨 프로토콜이 실패하거나 CRC 오류를 보인다고 표시됩니다.	ACCM(비동기 제어 문자 맵)이 잘못 설정되었습니다.	default-async 옵션을 사용하여 ACCM을 표준 기본값인 FFFFFFFF로 설정합니다. 먼저, 명령줄에서 default-async를 pppd에 대한 옵션으로 사용해 봅니다. 문제가 해결되면 default-async를 /etc/ppp/options 또는 /etc/ppp/peers/peer-name에 추가합니다(call 옵션 뒤).
pppd debug 출력에 IPCP가 시작되지만 즉시 종료된다고 표시됩니다.	IP 주소가 잘못 구성되었을 수 있습니다.	1. 체트 스크립트에 잘못된 IP 주소가 있는지 확인합니다. 2. 체트 스크립트가 올바른 경우 피어에 대한 디버그 로그를 요청하고 피어 로그에서 IP 주소를 확인합니다.
링크의 성능이 현저히 떨어집니다.	모뎀이 잘못 구성되었을 수 있습니다(흐름 제어 구성 오류, 모뎀 설정 오류 및 잘못 구성된 DTE 속도).	모뎀 구성을 확인합니다. 필요한 경우 구성을 조정합니다.

## ▼ PPP 구성을 사용하여 문제를 진단하는 방법

일부 PPP 문제는 PPP 구성 파일의 문제 때문에 발생할 수 있습니다. 다음 절차에서는 일반적인 구성 문제를 격리하고 해결하는 방법을 보여줍니다.

### 1 로컬 시스템에서 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 원격 피어를 호출합니다. 그런 다음 디버깅 기능을 사용으로 설정합니다.

```
% pppd debug call peer-name
```

### 3 결과 로그에서 구성 문제를 확인합니다. 자세한 내용은 456 페이지 “일반적인 PPP 구성 문제”를 참조하십시오.

# 일반적인 PPP 구성 문제

다음 표에는 455 페이지 “PPP 구성을 사용하여 문제를 진단하는 방법” 절차의 로그 출력과 관련된 증상이 설명되어 있습니다.

표 21-4 일반적인 PPP 구성 문제

증상	문제점	해결 방법
pppd debug 출력에 Could not determine remote IP address라는 오류 메시지가 포함되어 있습니다.	/etc/ppp/peers/peer-name 파일에 피어에 대한 IP 주소가 없습니다. 피어가 링크 협상 중 IP 주소를 제공하지 않습니다.	pppd 명령줄 또는 /etc/ppp/peers/peer-name에서 다음 형식을 사용하여 피어에 대한 IP 주소를 제공합니다.  :10.0.0.10
pppd debug 출력에 CCP 데이터 압축이 실패했다고 표시됩니다. 또한 출력에 링크가 삭제되었다고 표시됩니다.	피어의 PPP 압축 구성에서 충돌이 발생했을 수 있습니다.	피어 중 하나에서 noccip 옵션을 /etc/ppp/options에 추가하여 CCP 압축을 사용 안함으로 설정합니다.

## ▼ 모뎀 문제를 진단하는 방법

모뎀은 다이얼 업 링크의 주요 문제 영역이 될 수 있습니다. 흔히 피어로부터 응답이 없을 때 모뎀 구성에 문제가 있다는 사실을 알 수 있습니다. 그러나 링크 문제가 정말로 모뎀 구성 문제의 결과인지는 확인하기가 어려울 수 있습니다.

모뎀 제조업체의 설명서 및 웹 사이트에 특정 장비 문제에 대한 해결 방법이 포함되어 있습니다. 다음 절차는 잘못된 모뎀 구성으로 인해 링크 문제가 발생했는지 여부를 확인하는 데 도움이 됩니다.

- 1 450 페이지 “PPP 디버깅을 켜는 방법”에 설명된 대로 디버깅을 켜 채 피어를 호출합니다.
- 2 결과 /var/log/pppdebug 로그를 표시하여 잘못된 모뎀 구성을 확인합니다.
- 3 ping을 사용하여 다양한 크기의 패킷을 링크를 통해 보냅니다.  
ping에 대한 자세한 내용은 ping(1M) 매뉴얼 페이지를 참조하십시오.  
작은 패킷을 받았지만 더 큰 패킷을 삭제한 경우에는 모뎀 문제가 표시됩니다.
- 4 sppp0 인터페이스에서 오류를 확인합니다.

```
% netstat -ni
Name Mtu Net/Dest Address Ipkts Ierrs Opkts Oerrs Collis Queue
lo0 8232 127.0.0.0 127.0.0.1 826808 0 826808 0 0 0
hme0 1500 172.21.0.0 172.21.3.228 13800032 0 1648464 0 0 0
sppp0 1500 10.0.0.2 10.0.0.1 210 0 128 0 0 0
```

시간이 지남에 따라 인터페이스 오류가 늘어나면 모뎀 구성에 문제가 있을 수 있습니다.



- 일반 오류** 결과 `/var/log/pppdebug` 로그를 표시할 때 출력에 다음 증상이 나타나면 모뎀 구성이 잘못된 것일 수 있습니다. 로컬 시스템은 피어를 들을 수 있지만 피어는 로컬 시스템을 들을 수 없습니다.
- 피어로부터 아무 “recvd” 메시지도 받지 못했습니다.
  - 출력에 피어로부터 받은 LCP 메시지가 포함되어 있지만 로컬 시스템이 보낸 `too many LCP Configure Requests` 메시지와 함께 링크가 실패합니다.
  - 링크가 SIGHUP 신호와 함께 종료됩니다.

## ▼ 채트 스크립트에 대한 디버깅 정보를 가져오는 방법

chat에서 디버깅 정보를 가져오기 위한 다음 절차와 일반적인 문제를 해결하는 방법에 대한 제안 사항을 사용하십시오. 자세한 내용은 [457 페이지 “일반적인 채트 스크립트 문제”](#)를 참조하십시오.

- 1 다이얼 아웃 시스템에서 관리자가 됩니다.  
자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 호출할 피어에 대한 `/etc/ppp/peers/peer-name` 파일을 편집합니다.

- 3 `connect` 옵션에 지정되어 있는 chat 명령에 대한 인수로 `-v`를 추가합니다.  
`connect "/usr/bin/chat -v -f /etc/ppp/chat-script-name"`

- 4 `/etc/ppp/connect-errors` 파일에서 채트 스크립트 오류를 봅니다.

다음은 chat에 대해 발생하는 주요 오류입니다.

```
Oct 31 08:57:13 deino chat[107294]: [ID 702911 local2.info] expect (CONNECT)
Oct 31 08:57:58 deino chat[107294]: [ID 702911 local2.info] alarm
Oct 31 08:57:58 deino chat[107294]: [ID 702911 local2.info] Failed
```

예에서는 (CONNECT) 문자열을 기다리는 동안 시간이 초과되었음을 보여줍니다. chat가 실패하면 pppd로부터 다음 메시지가 반환됩니다.

```
Connect script failed
```

## 일반적인 채트 스크립트 문제

채트 스크립트는 다이얼 업 링크에 대해 문제가 발생하기 쉬운 영역입니다. 다음 표에는 일반적인 채트 스크립트 오류와 오류를 해결하기 위한 제안 사항이 나와 있습니다. 절차 정보는 [457 페이지 “채트 스크립트에 대한 디버깅 정보를 가져오는 방법”](#)을 참조하십시오.

표 21-5 일반적인 채트 스크립트 문제

증상	문제점	해결 방법
pppd debug 출력에 Connect script failed가 포함되어 있습니다.	채트 스크립트가 사용자 이름 및 암호를 제공합니다.  ogin: <i>user-name</i> ssword: <i>password</i>  그러나 연결하려고 했던 피어가 이 정보를 요청하는 메시지를 표시하지 않습니다.	1. 채트 스크립트에서 로그인 및 암호를 삭제합니다.  2. 피어를 다시 호출해 봅니다.  3. 계속 메시지가 반환되면 ISP에 문의합니다. ISP에 올바른 로그인 절차가 무엇인지 문의합니다.
/usr/bin/chat -v 로그에 "expect (login:) "alarm read timed out"이 포함되어 있습니다.	채트 스크립트가 사용자 이름 및 암호를 제공합니다.  ogin: pppuser ssword: \q\U  그러나 연결하려고 하는 피어가 이 정보를 요청하는 메시지를 표시하지 않습니다.	1. 채트 스크립트에서 로그인 및 암호를 삭제합니다.  2. 피어를 다시 호출해 봅니다.  3. 계속 메시지가 반환되면 ISP에 문의합니다. ISP에 올바른 로그인 절차가 무엇인지 문의합니다.
pppd debug 출력에 possibly looped-back이 포함되어 있습니다.	로컬 시스템 또는 해당 피어가 명령줄에서 정지되어 있으며 PPP를 실행하고 있지 않습니다. 잘못 구성된 로그인 이름 및 암호가 채트 스크립트에 있습니다.	1. 채트 스크립트에서 로그인 및 암호를 삭제합니다.  2. 피어를 다시 호출해 봅니다.  3. 계속 메시지가 반환되면 ISP에 문의합니다. 올바른 로그인 절차가 무엇인지 문의합니다.
pppd debug 출력에 LCP가 활성화되지만 링크가 곧 종료된다고 표시됩니다.	채트 스크립트의 암호가 잘못되었을 수 있습니다.	1. 로컬 시스템의 암호가 올바른지 확인합니다.  2. 채트 스크립트에서 암호를 확인합니다. 잘못된 경우 암호를 수정합니다.  3. 피어를 다시 호출해 봅니다.  4. 계속 메시지가 반환되면 ISP에 문의합니다. ISP에 올바른 로그인 절차가 무엇인지 문의합니다.
피어로부터의 텍스트가 틸드(~)로 시작합니다.	채트 스크립트가 사용자 이름 및 암호를 제공합니다.  ogin: pppuser ssword: \q\U  그러나 연결하려고 하는 피어가 이 정보를 요청하는 메시지를 표시하지 않습니다.	1. 채트 스크립트에서 로그인 및 암호를 삭제합니다.  2. 피어를 다시 호출해 봅니다.  3. 계속 메시지가 반환되면 ISP에 문의합니다. 올바른 로그인 절차를 요청합니다.

표 21-5 일반적인 채트 스크립트 문제 (계속)

증상	문제점	해결 방법
모뎀이 정지됩니다.	채트 스크립트에 로컬 시스템이 피어로부터 <b>CONNECT</b> 메시지를 기다리게 만드는 다음 행이 포함되어 있습니다.  <b>CONNECT "</b>	채트 스크립트가 피어로부터 <b>CONNECT</b> 를 기다리게 만들려면 다음 행을 사용합니다.  <b>CONNECT \c</b>  ~\c로 채트 스크립트를 끝냅니다.
pppd debug 출력에 LCP: timeout sending Config-Requests가 포함되어 있습니다.	채트 스크립트에 로컬 시스템이 피어로부터 <b>CONNECT</b> 메시지를 기다리게 만드는 다음 행이 포함되어 있습니다.  <b>CONNECT "</b>	채트 스크립트가 피어로부터 <b>CONNECT</b> 를 기다리게 만들려면 다음 행을 사용합니다.  <b>CONNECT \c</b>  ~\c로 채트 스크립트를 끝냅니다.
pppd debug 출력에 Serial link is not 8-bit clean이 포함되어 있습니다.	채트 스크립트에 로컬 시스템이 피어로부터 <b>CONNECT</b> 메시지를 기다리게 만드는 다음 행이 포함되어 있습니다.  <b>CONNECT "</b>	채트 스크립트가 피어로부터 <b>CONNECT</b> 를 기다리게 만들려면 다음 행을 사용합니다.  <b>CONNECT \c</b>  ~\c로 채트 스크립트를 끝냅니다.
pppd debug 출력에 Loopback detected가 포함되어 있습니다.	채트 스크립트에 로컬 시스템이 피어로부터 <b>CONNECT</b> 메시지를 기다리게 만드는 다음 행이 포함되어 있습니다.  <b>CONNECT "</b>	채트 스크립트가 피어로부터 <b>CONNECT</b> 를 기다리게 만들려면 다음 행을 사용합니다.  <b>CONNECT \c</b>  ~\c로 채트 스크립트를 끝냅니다.
pppd debug 출력에 SIGHUP가 포함되어 있습니다.	채트 스크립트에 로컬 시스템이 피어로부터 <b>CONNECT</b> 메시지를 기다리게 만드는 다음 행이 포함되어 있습니다.  <b>CONNECT "</b>	채트 스크립트가 피어로부터 <b>CONNECT</b> 를 기다리게 만들려면 다음 행을 사용합니다.  <b>CONNECT \c</b>  ~\c로 채트 스크립트를 끝냅니다.

## ▼ 직렬 회선 속도 문제를 진단하고 해결하는 방법

속도 설정이 충돌하여 다이얼 인 서버에 문제가 발생할 수 있습니다. 다음 절차는 링크 문제의 원인을 충돌하는 직렬 회선 속도로 격리하는 데 도움이 됩니다.

다음 동작으로 인해 속도 문제가 발생합니다.

- /bin/login과 같은 프로그램을 통해 PPP를 호출하고 회선의 속도를 지정했습니다.
- mgetty에서 PPP를 시작하고 실수로 비트 속도를 제공했습니다.

pppd는 회선에 대해 원래 설정된 속도를 /bin/login 또는 mgetty에 의해 설정된 속도로 변경합니다. 결과적으로 회선에 문제가 발생합니다.

- 1 다이얼 인 서버에 로그인합니다. 디버깅을 사용으로 설정한 채 피어를 호출합니다.  
지침이 필요하면 [450 페이지 “PPP 디버깅을 켜는 방법”](#)을 참조하십시오.
- 2 결과 `/var/log/pppdebug` 로그를 표시합니다.  
출력에서 다음 메시지를 확인합니다.  
LCP too many configure requests  
이 메시지는 PPP에 대해 구성된 직렬 회선의 속도가 충돌했을 수 있음을 나타냅니다.
- 3 `/bin/login`과 같은 프로그램을 통해 PPP가 호출되었는지 여부와 설정된 회선 속도를 확인합니다.  
그런 경우 `pppd`가 원래 구성된 회선 속도를 `/bin/login`에 지정된 속도로 변경합니다.
- 4 사용자가 `mgetty` 명령에서 PPP를 시작하고 실수로 비트 속도를 지정했는지 여부를 확인합니다.  
이 작업도 직렬 회선 속도가 충돌하게 만듭니다.
- 5 충돌하는 직렬 회선 속도 문제를 다음과 같이 해결합니다.
  - a. 모뎀에서 DTE 속도를 고정합니다.
  - b. 자동 번조를 사용하지 않습니다.
  - c. 구성 후에 회선 속도를 변경하지 않습니다.

## ▼ PPPoE에 대한 진단 정보를 가져오는 방법

PPP 및 표준 UNIX 유틸리티를 사용하여 PPPoE 문제를 식별할 수 있습니다. PPPoE가 링크 문제의 원인으로 의심되는 경우 다음 진단 도구를 사용하여 문제 해결 정보를 얻으십시오.

- 1 PPPoE 터널을 실행하는 시스템(PPPoE 클라이언트 또는 PPPoE 액세스 서버)에서 슈퍼 유저가 됩니다.
- 2 [450 페이지 “PPP 디버깅을 켜는 방법”](#) 절차에 설명된 대로 디버깅을 켭니다.
- 3 로그 파일 `/var/log/pppdebug`의 내용을 봅니다.  
다음 예에서는 PPPoE 터널과의 링크에 대해 생성된 로그 파일의 일부분을 보여줍니다.

```
Sep  6 16:28:45 enyo pppd[100563]: [ID 702911 daemon.info] Plugin
    ppoe.so loaded.
Sep  6 16:28:45 enyo pppd[100563]: [ID 860527 daemon.notice] pppd
    2.4.0b1 (Sun Microsystems, Inc.,
Sep  5 2001 10:42:05) started by troot, uid 0
```

```

Sep  6 16:28:46 enyo pppd[100563]: [ID 702911 daemon.debug] connect option:
'/usr/lib/inet/pppoe'
-v hme0' started (pid 100564)
Sep  6 16:28:46 enyo pppd[100563]: [ID 702911 daemon.info] Serial connection established.
Sep  6 16:28:46 enyo pppd[100563]: [ID 702911 daemon.info] Using interface sppp0
Sep  6 16:28:46 enyo pppd[100563]: [ID 702911 daemon.notice] Connect: sppp0
<--> /dev/sppptun
Sep  6 16:28:46 enyo pppd[100563]: [ID 702911 daemon.debug] /etc/ppp/pap-secrets
is apparently empty
Sep  6 16:28:46 enyo pppd[100563]: [ID 702911 daemon.debug] /etc/ppp/chap-secrets
is apparently empty
Sep  6 16:28:46 enyo pppd[100563]: [ID 702911 daemon.debug] sent
[LCP ConfReq id=0xef <mru 1492>
asynmap 0x0 <magic 0x77d3e953><pcomp><acomp>
Sep  6 16:28:46 enyo pppd[100563]: [ID 702911 daemon.debug] rcvd
[LCP ConfReq id=0x2a <mru 1402>
asynmap 0x0 <magic 0x9985f048><pcomp><acomp>

```

디버깅 출력이 문제를 격리하는 데 도움이 되지 않는 경우 이 절차를 계속하십시오.

#### 4 PPPoE에서 진단 메시지를 가져옵니다.

```
# pppd connect "/usr/lib/inet/pppoe -v interface-name"
```

pppoe는 진단 정보를 stderr로 보냅니다. 전면에서 pppd를 실행하면 화면에 출력이 나타납니다. 배경에서 pppd를 실행하면 출력이 /etc/ppp/connect-errors로 전송됩니다.

다음 예에서는 PPPoE 터널이 협상되면서 생성되는 메시지를 보여줍니다.

```

Connect option: '/usr/lib/inet/pppoe -v hme0' started (pid 100564)
/usr/lib/inet/pppoe: PPPoE Event Open (1) in state Dead (0): action SendPADI (2)
/usr/lib/inet/pppoe: Sending PADI to ff:ff:ff:ff:ff:ff: 18 bytes
/usr/lib/inet/pppoe: PPPoE State change Dead (0) -> InitSent (1)
/usr/lib/inet/pppoe: Received Active Discovery Offer from 8:0:20:cd:c1:2/hme0:pppoe
/usr/lib/inet/pppoe: PPPoE Event rPADO+ (5) in state InitSent (1): action SendPADR+ (5)
/usr/lib/inet/pppoe: Sending PADR to 8:0:20:cd:c1:2: 22 bytes
/usr/lib/inet/pppoe: PPPoE State change InitSent (1) -> ReqSent (3)
/usr/lib/inet/pppoe: Received Active Discovery Session-confirmation from
8:0:20:cd:c1:2/hme0:pppoe
/usr/lib/inet/pppoe: PPPoE Event rPADS (7) in state ReqSent (3): action Open (7)
/usr/lib/inet/pppoe: Connection open; session 0002 on hme0:pppoe
/usr/lib/inet/pppoe: PPPoE State change ReqSent (3) -> Convers (4)
/usr/lib/inet/pppoe: connected

```

진단 메시지가 문제를 격리하는 데 도움이 되지 않는 경우 이 절차를 계속하십시오.

#### 5 snoop을 실행합니다. 그런 다음 파일에 추적을 저장합니다.

snoop에 대한 자세한 내용은 [snoop\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

```
# snoop -o pppoe-trace-file
```

#### 6 snoop 추적 파일을 봅니다.

```
# snoop -i pppoe-trace-file -v pppoe
```

```

ETHER: ----- Ether Header -----
ETHER:
ETHER: Packet 1 arrived at 6:35:2.77

```

```
ETHER: Packet size = 32 bytes
ETHER: Destination = ff:ff:ff:ff:ff:ff, (broadcast)
ETHER: Source      = 8:0:20:78:f3:7c, Sun
ETHER: Ethertype = 8863 (PPPoE Discovery)
ETHER:
PPPoE: ----- PPP Over Ethernet -----
PPPoE:
PPPoE: Version = 1
PPPoE: Type = 1
PPPoE: Code = 9 (Active Discovery Initiation)
PPPoE: Session Id = 0
PPPoE: Length = 12 bytes
PPPoE:
PPPoE: ----- Service-Name -----
PPPoE: Tag Type = 257
PPPoE: Tag Length = 0 bytes
PPPoE:
PPPoE: ----- Host-Uniq -----
PPPoE: Tag Type = 259
PPPoE: Tag Length = 4 bytes
PPPoE: Data = 0x00000002
PPPoE:
.
.
.
ETHER: ----- Ether Header -----
ETHER:
ETHER: Packet 5 arrived at 6:35:2.87
ETHER: Packet size = 60 bytes
ETHER: Destination = 8:0:20:78:f3:7c, Sun)
ETHER: Source      = 0:2:fd:39:7f:7,
ETHER: Ethertype = 8864 (PPPoE Session)
ETHER:
PPPoE: ----- PPP Over Ethernet -----
PPPoE:
PPPoE: Version = 1
PPPoE: Type = 1
PPPoE: Code = 0 (PPPoE Session)
PPPoE: Session Id = 24383
PPPoE: Length = 20 bytes
PPPoE:
PPP: ----- Point-to-Point Protocol -----
PPP:
PPP-LCP: ----- Link Control Protocol -----
PPP-LCP:
PPP-LCP: Code = 1 (Configure Request)
PPP-LCP: Identifier = 80
PPP-LCP: Length = 18
```

## 전용 회선 문제 해결

전용 회선에 가장 흔히 발생하는 문제는 성능 저하입니다. 대부분의 경우 전화 회사에 문의하여 문제를 해결해야 합니다.

표 21-6 일반적인 전용 회선 문제

증상	문제점	해결 방법
링크가 시작되지 않습니다.	CSU 양극성 위반(CSU BPV)이 원인일 수 있습니다. 링크의 한쪽 끝은 AMI 회선용으로 설정되어 있고, 다른 쪽 끝은 ESF Bit 8 제로 대치(B8Zs)로 설정되어 있습니다.	미국 또는 캐나다에 있는 경우 CSU/DSU 메뉴에서 직접 이 문제를 해결할 수 있습니다. 자세한 내용은 CSU/DSU 제조업체의 설명서를 참조하십시오. 다른 위치에서는 공급자가 CSU BPV 문제를 해결해야 할 수 있습니다.
링크의 성능이 저하되었습니다.	링크에 지속 트래픽이 있을 때 pppd debug 출력에 CRC 오류가 표시됩니다. 전화 회사와 사용 중인 네트워크 간의 잘못된 구성으로 인해 회선에 클록킹 문제가 있을 수 있습니다.	전화 회사에 문의하여 "루프 클록킹"이 사용되고 있는지 확인합니다. 구조화되지 않은 일부 전용 회선에서는 클록킹을 제공해야 할 수 있습니다. 북미 사용자는 루프 클록킹을 사용해야 합니다.

## 인증 문제 진단 및 해결

다음 표에는 일반적인 인증 문제에 대한 해결 방법이 설명되어 있습니다.

표 21-7 일반적인 인증 문제

증상	문제점	해결 방법
pppd debug 출력에 Peer is not authorized to use remote address address(피어에게 해당 원격 주소를 사용할 권한이 부여되지 않음) 메시지가 표시됩니다.	PAP 인증을 사용하고 있는데 원격 피어의 IP 주소가 /etc/ppp/pap-secrets 파일에 없습니다.	/etc/ppp/pap-secrets 파일에서 피어에 대한 항목 다음에 별표(*)를 추가합니다.
pppd debug 출력에 LCP가 시작되지만 곧 종료된다고 표시됩니다.	암호가 특정 보안 프로토콜에 대한 데이터베이스에서 잘못되었을 수 있습니다.	/etc/ppp/pap-secrets 또는 /etc/ppp/chap-secrets 파일에서 피어의 암호를 확인합니다.





## Solaris PPP 4.0(참조)

---

이 장에서는 Solaris PPP 4.0에 대한 자세한 개념 정보를 제공합니다. 항목은 다음과 같습니다.

- 465 페이지 “파일 및 명령줄에서 PPP 옵션 사용”
- 473 페이지 “사용자별 옵션 구성”
- 473 페이지 “다이얼 인 서버와의 통신을 위한 정보 지정”
- 476 페이지 “다이얼 업 링크를 위한 모뎀 속도 구성”
- 476 페이지 “다이얼 업 링크에서 대화 정의”
- 486 페이지 “링크에서 호출자 인증”
- 492 페이지 “호출자를 위한 IP 주소 지정 체계 만들기”
- 494 페이지 “DSL 지원을 위해 PPPoE 터널 만들기”

### 파일 및 명령줄에서 PPP 옵션 사용

Solaris PPP 4.0에는 PPP 구성을 정의하는 데 사용할 수 있는 방대한 옵션 세트가 포함되어 있습니다. 이러한 옵션은 PPP 구성 파일 또는 명령줄에서 사용하거나 파일 및 명령줄 옵션을 조합하여 사용합니다. 이 절에는 PPP 옵션을 구성 파일에서 사용하거나 PPP 명령에 대한 인수로 사용하는 방법에 대한 자세한 내용이 포함되어 있습니다.

### PPP 옵션을 정의하는 위치

Solaris PPP 4.0 구성은 매우 유연합니다. PPP 옵션은 다음 위치에서 정의할 수 있습니다.

- PPP 구성 파일
- 명령줄에서 실행되는 PPP 명령
- 두 위치의 조합

다음 표에는 PPP 구성 파일 및 명령이 나열되어 있습니다.

표 22-1 PPP 구성 파일 및 명령 요약

파일 또는 명령	설명	정보
<code>/etc/ppp/options</code>	시스템에 있는 모든 PPP 링크에 기본적으로 적용되는 특징(예: 피어가 자신을 인증할 것을 시스템이 요구하는지 여부)이 포함된 파일입니다. 이 파일이 없으면 비루트 사용자가 PPP를 사용할 수 없습니다.	469 페이지 “ <code>/etc/ppp/options</code> 구성 파일”
<code>/etc/ppp/options.ttyname</code>	직렬 포트 <code>ttyname</code> 을 통한 모든 통신의 특징을 기술하는 파일입니다.	471 페이지 “ <code>/etc/ppp/options.ttyname</code> 구성 파일”
<code>/etc/ppp/peers</code>	다이얼 아웃 시스템이 연결하는 피어에 대한 정보가 일반적으로 포함되어 있는 디렉토리입니다. 이 디렉토리에 있는 파일은 <code>pppd</code> 명령의 <code>call</code> 옵션에 사용됩니다.	473 페이지 “다이얼 인 서버와의 통신을 위한 정보 지정”
<code>/etc/ppp/peers/peer-name</code>	원격 피어 <code>peer-name</code> 의 특징이 포함된 파일입니다. 일반적인 특징에는 원격 피어의 전화 번호 및 피어와 링크를 협상하기 위한 채트 스크립트가 있습니다.	474 페이지 “ <code>/etc/ppp/peers/peer-name</code> 파일”
<code>/etc/ppp/pap-secrets</code>	PAP(암호 인증 프로토콜) 인증에 필요한 보안 자격 증명이 포함된 파일입니다.	486 페이지 “ <code>/etc/ppp/pap-secrets</code> 파일”
<code>/etc/ppp/chap-secrets</code>	CHAP(Challenge-Handshake 인증 프로토콜) 인증에 필요한 보안 자격 증명이 포함된 파일입니다.	489 페이지 “ <code>/etc/ppp/chap-secrets</code> 파일”
<code>~/.ppprc</code>	PPP 사용자의 홈 디렉토리에 있는 파일로, 다이얼 인 서버에 가장 자주 사용됩니다. 이 파일에는 각 사용자 구성과 관련된 정보가 포함되어 있습니다.	473 페이지 “다이얼 인 서버에서 <code>\$HOME/.ppprc</code> 구성”
<code>pppd</code> 옵션	PPP 링크를 시작하고 해당 특징을 기술하기 위한 명령 및 옵션입니다.	466 페이지 “PPP 옵션이 처리되는 방법”

PPP 파일에 대한 자세한 내용은 [pppd\(1M\)](#) 매뉴얼 페이지를 참조하십시오. `pppd(1M)`에는 `pppd` 명령에 사용할 수 있는 모든 옵션에 대한 포괄적인 설명도 포함되어 있습니다. 모든 PPP 구성 파일에 대한 샘플 템플릿은 `/etc/ppp`에 제공되어 있습니다.

## PPP 옵션이 처리되는 방법

1. `pppd` 데몬이 다음의 구문을 분석합니다.

모든 Solaris PPP 4.0 작업은 `pppd` 데몬에 의해 처리되며, 이 데몬은 사용자가 `pppd` 명령을 실행할 때 시작됩니다. 사용자가 원격 피어를 호출하면 다음이 발생합니다.

- `/etc/ppp/options`
  - `$HOME/.ppprc`
  - `/etc/ppp/options` 및 `$HOME/.ppprc`에서 `file` 또는 `call` 옵션으로 열리는 모든 파일
2. `pppd`가 명령줄을 스캔하여 사용 중인 장치를 확인합니다. 데몬은 발견하는 옵션을 아직 해석하지 않습니다.
  3. `pppd`가 다음 조건을 사용하여 직렬 장치를 검색하려고 합니다.
    - 직렬 장치가 명령줄 또는 이전에 처리된 구성 파일에서 지정된 경우 `pppd`는 해당 장치의 이름을 사용합니다.
    - 명명된 직렬 장치가 없는 경우 `pppd`가 명령줄에서 `notty`, `pty` 또는 `socket` 옵션을 검색합니다. 이러한 옵션 중 하나가 지정된 경우에는 `pppd`가 장치 이름이 없다고 가정합니다.
    - 그렇지 않고 표준 입력이 `tty`에 연결되어 있음을 `pppd`가 발견하면 해당 `tty`의 이름이 사용됩니다.
    - `pppd`가 여전히 직렬 장치를 찾을 수 없는 경우 `pppd`는 연결을 종료하고 오류를 발생시킵니다.
  4. 그런 다음 `pppd`가 `/etc/ppp/options.ttyname` 파일이 있는지 확인합니다. 해당 파일이 발견되면 `pppd`가 파일의 구문을 분석합니다.
  5. `pppd`가 명령줄에서 모든 옵션을 처리합니다.
  6. `pppd`가 링크를 설정하기 위해 LCP(링크 제어 프로토콜)를 협상합니다.
  7. (옵션) 인증이 필요한 경우 `pppd`가 반대쪽 피어를 인증하기 위해 `/etc/ppp/pap-secrets` 또는 `/etc/ppp/chap-secrets`를 읽습니다.

`/etc/ppp/peers/peer-name` 파일은 `pppd` 데몬이 명령줄 또는 기타 구성 파일에서 `call peer-name` 옵션을 발견할 때 읽습니다.

## PPP 구성 파일 권한의 작동 방식

Solaris PPP 4.0 구성에는 **권한**이라는 개념이 포함되어 있습니다. 권한은 특히 둘 이상의 위치에서 동일한 옵션이 호출될 때 구성 옵션의 우선 순위를 결정합니다. 권한 있는 소스에서 호출된 옵션이 권한 없는 소스에서 호출된 동일한 옵션보다 우선적으로 사용됩니다.

### 사용자 권한

권한 있는 사용자는 UID가 0인 슈퍼 유저(`root`)뿐입니다. 다른 모든 사용자에게는 권한이 없습니다.

## 파일 권한

다음 구성 파일에 대한 권한은 소유권에 관계없이 부여됩니다.

- /etc/ppp/options
- /etc/ppp/options.ttyname
- /etc/ppp/peers/peer-name

\$HOME/.ppprc 파일은 사용자가 소유합니다. \$HOME/.ppprc 및 명령줄에서 읽는 옵션에 대한 권한은 pppd를 호출하는 사용자가 root인 경우에만 부여됩니다.

file 옵션 뒤에 오는 인수의 경우 권한이 부여됩니다.

## 옵션 권한의 영향

일부 옵션의 경우 호출 사용자 또는 소스에 권한이 있어야 작동합니다. 명령줄에서 호출되는 옵션에는 pppd 명령을 실행하는 사용자의 권한이 지정됩니다. pppd를 호출하는 사용자가 root여야 이러한 옵션에 권한이 부여됩니다.

옵션	상태	설명
도메인	권한이 부여됨	사용하려면 권한이 필요합니다.
linkname	권한이 부여됨	사용하려면 권한이 필요합니다.
noauth	권한이 부여됨	사용하려면 권한이 필요합니다.
nopam	권한이 부여됨	사용하려면 권한이 필요합니다.
pam	권한이 부여됨	사용하려면 권한이 필요합니다.
plugin	권한이 부여됨	사용하려면 권한이 필요합니다.
privgroup	권한이 부여됨	사용하려면 권한이 필요합니다.
allow-ip addresses	권한이 부여됨	사용하려면 권한이 필요합니다.
name hostname	권한이 부여됨	사용하려면 권한이 필요합니다.
plink	권한이 부여됨	사용하려면 권한이 필요합니다.
nopl原因	권한이 부여됨	사용하려면 권한이 필요합니다.
plumbed	권한이 부여됨	사용하려면 권한이 필요합니다.
proxyarp	noproxyarp가 지정된 경우 권한이 부여됨	권한 없는 사용자가 대체할 수 없습니다.
defaultroute	nodefaultroute가 권한이 있는 파일에 설정되거나 권한 있는 사용자에 의해 설정되는 경우 권한이 부여됨	권한 없는 사용자가 대체할 수 없습니다.

옵션	상태	설명
<code>disconnect</code>	권한이 있는 파일에 설정되거나 권한 있는 사용자에 의해 설정되는 경우 권한이 부여됨	권한 없는 사용자가 대체할 수 없습니다.
<code>bsdcomp</code>	권한이 있는 파일에 설정되거나 권한 있는 사용자에 의해 설정되는 경우 권한이 부여됨	권한이 없는 사용자는 권한이 있는 사용자가 지정한 것보다 큰 코드 크기를 지정할 수 없습니다.
<code>deflate</code>	권한이 있는 파일에 설정되거나 권한 있는 사용자에 의해 설정되는 경우 권한이 부여됨	권한이 없는 사용자는 권한이 있는 사용자가 지정한 것보다 큰 코드 크기를 지정할 수 없습니다.
<code>connect</code>	권한이 있는 파일에 설정되거나 권한 있는 사용자에 의해 설정되는 경우 권한이 부여됨	권한이 없는 사용자가 대체할 수 없습니다.
<code>init</code>	권한이 있는 파일에 설정되거나 권한 있는 사용자에 의해 설정되는 경우 권한이 부여됨	권한이 없는 사용자가 대체할 수 없습니다.
<code>pty</code>	권한이 있는 파일에 설정되거나 권한 있는 사용자에 의해 설정되는 경우 권한이 부여됨	권한이 없는 사용자가 대체할 수 없습니다.
<code>welcome</code>	권한이 있는 파일에 설정되거나 권한 있는 사용자에 의해 설정되는 경우 권한이 부여됨	권한이 없는 사용자가 대체할 수 없습니다.
<code>ttyname</code>	권한이 있는 파일에 설정되는 경우 권한이 부여됨  권한이 없는 파일에 설정되는 경우 권한이 부여되지 않음	<code>pppd</code> 를 호출하는 사용자에게 관계없이 루트 권한으로 열립니다.  <code>pppd</code> 를 호출하는 사용자의 권한으로 열립니다.

## /etc/ppp/options 구성 파일

`/etc/ppp/options` 파일을 사용하여 로컬 시스템에 있는 모든 PPP 통신에 대한 전역 옵션을 정의할 수 있습니다. `/etc/ppp/options`는 권한이 있는 파일입니다. `/etc/ppp/options`는 루트가 소유해야 합니다. 그러나 `pppd`에 의해 이 규칙이 강제로 적용되는 것은 아닙니다. `/etc/ppp/options`에 정의하는 옵션은 다른 모든 파일 및 명령줄에 있는 동일한 옵션의 정의보다 우선적으로 사용됩니다.

`/etc/ppp/options`에서 일반적으로 사용할 수 있는 옵션은 다음과 같습니다.

- **lock** – UUCP 스타일의 파일 잠금을 사용으로 설정합니다.
- **noauth** – 시스템이 호출자를 인증하지 않음을 나타냅니다.

주 - Solaris PPP 4.0 소프트웨어에는 기본 `/etc/ppp/options` 파일이 포함되어 있지 않습니다. `pppd`를 작동하는 데에는 `/etc/ppp/options` 파일이 필요하지 않습니다. 시스템에 `/etc/ppp/options` 파일이 없으면 `root`만 해당 시스템에서 `pppd`를 실행할 수 있습니다.

텍스트 편집기를 사용하여 `/etc/ppp/options`를 만들어야 합니다(408 페이지 “직렬 회선을 통해 통신을 정의하는 방법” 참조). 시스템에 전역 옵션이 필요하지 않으면 빈 `/etc/ppp/options` 파일을 만들 수 있습니다. 그러면 `root` 사용자와 일반 사용자가 모두 로컬 시스템에서 `pppd`를 실행할 수 있게 됩니다.

## `/etc/ppp/options.tmpl` 템플리트

`/etc/ppp/options.tmpl`에는 `/etc/ppp/options` 파일에 대한 유용한 설명과 전역 `/etc/ppp/options` 파일에 대한 세 가지 일반적인 옵션이 포함되어 있습니다.

```
lock
nodefaultroute
noproxyarp
```

옵션	정의
<code>lock</code>	UUCP 스타일의 파일 잠금을 사용으로 설정합니다.
<code>nodefaultroute</code>	기본 경로가 정의되어 있지 않음을 지정합니다.
<code>noproxyarp</code>	<code>proxyarp</code> 를 허용하지 않습니다.

`/etc/ppp/options.tmpl`을 전역 옵션 파일로 사용하려면 `/etc/ppp/options.tmpl`의 이름을 `/etc/ppp/options`로 바꾼 다음 사이트의 필요에 따라 파일 내용을 수정하십시오.

## `/etc/ppp/options` 파일의 예를 찾을 수 있는 위치

`/etc/ppp/options` 파일의 예를 찾으려면 다음을 참조하십시오.

- 다이얼 아웃 시스템의 경우 408 페이지 “직렬 회선을 통해 통신을 정의하는 방법”을 참조하십시오.
- 다이얼 인 서버의 경우 415 페이지 “직렬 회선을 통해 통신을 정의하는 방법(다이얼 인 서버)”을 참조하십시오.
- 다이얼 인 서버에서의 PAP 지원의 경우 429 페이지 “PPP 구성 파일에 PAP 지원을 추가하는 방법(다이얼 인 서버)”을 참조하십시오.
- 다이얼 아웃 시스템에서의 PAP 지원의 경우 432 페이지 “PPP 구성 파일에 PAP 지원을 추가하는 방법(다이얼 아웃 시스템)”을 참조하십시오.
- 다이얼 인 서버에서의 CHAP 지원의 경우 436 페이지 “PPP 구성 파일에 CHAP 지원을 추가하는 방법(다이얼 인 서버)”을 참조하십시오.

## /etc/ppp/options.ttyname 구성 파일

`/etc/ppp/options.ttyname` 파일에서 직렬 회선에 대해 통신의 특징을 구성할 수 있습니다. `/etc/ppp/options.ttyname`은 모든 기존 `/etc/ppp/options` 및 기존 `$HOME/.ppprc` 파일의 구문을 분석한 후에 `pppd`가 읽는 권한이 있는 파일입니다. 그렇지 않은 경우 `pppd`는 `/etc/ppp/options.ttyname`을 `/etc/ppp/options`의 구문을 분석한 후에 읽습니다.

`ttyname`은 다이얼 업 링크와 전용 회선 링크 모두에 사용됩니다. `ttyname`은 모뎀 또는 ISDN TA가 연결될 수 있는 `cua/a` 또는 `cua/b`와 같은 시스템의 특정 직렬 포트를 나타냅니다.

`/etc/ppp/options.ttyname` 파일의 이름을 지정할 때는 장치 이름에 있는 슬래시(/)를 점(.)으로 바꿔야 합니다. 예를 들어, `cua/b` 장치의 `options` 파일은 이름을 `/etc/ppp/options.cua.b`로 지정해야 합니다.

---

주 - Solaris PPP 4.0의 경우 올바르게 작동하는 데 `/etc/ppp/options.ttyname` 파일을 필요로 하지 않습니다. 서버에는 PPP용 직렬 회선이 하나만 있을 수 있습니다. 또한 서버에는 옵션이 거의 필요하지 않습니다. 이러한 경우에는 다른 구성 파일 또는 명령줄에서 필요한 모든 옵션을 지정할 수 있습니다.

---

## 다이얼 인 서버에서 /etc/ppp/options.ttyname 사용

다이얼 업 링크의 경우 모뎀이 연결된 다이얼 인 서버에서 모든 직렬 포트에 대해 개별 `/etc/ppp/options.ttyname` 파일을 만들 수 있습니다. 일반적인 옵션은 다음과 같습니다.

- 다이얼 인 서버에 필요한 IP 주소

직렬 포트 `ttyname`의 수신 호출자가 특정 IP 주소를 사용해야 하는 경우 이 옵션을 설정합니다. 주소 공간에는 잠재적 호출자 수에 비해 PPP용으로 사용 가능한 IP 주소가 제한되어 있을 수 있습니다. 이 경우 다이얼 인 서버에서 PPP에 사용되는 각 직렬 인터페이스에 IP 주소를 지정해 보십시오. 이렇게 하면 PPP에 동적 주소 지정이 구현됩니다.

- `asyncmap map-value`

`asyncmap` 옵션은 특정 모뎀 또는 ISDN TA로 직렬 회선을 통해 받을 수 없는 제어 문자를 매핑합니다. `xonxoff` 옵션을 사용하면 `pppd`가 `asyncmap 0xa0000`를 자동으로 설정합니다.

`map-value`는 문제가 되는 제어 문자를 16진수 형식으로 나타냅니다.

- `init "chat -U -f /etc/ppp/mychat"`

`init` 옵션은 `chat -U` 명령의 정보를 사용하여 직렬 회선을 통해 통신을 초기화하도록 모뎀에 지시합니다. 모뎀은 `/etc/ppp/mychat` 파일의 채트 문자열을 사용합니다.

- `pppd(1m)` 매뉴얼 페이지에 나열되어 있는 보안 매개변수

## 다이얼 아웃 시스템에서 /etc/ppp/options.ttyname 사용

다이얼 아웃 시스템의 경우 모뎀에 연결된 직렬 포트에 대해 /etc/ppp/options.ttyname 파일을 만들거나 /etc/ppp/options.ttyname을 사용하지 않도록 선택할 수 있습니다.

주 - Solaris PPP 4.0의 경우 올바르게 작동하는 데 /etc/ppp/options.ttyname 파일을 필요로 하지 않습니다. 다이얼 아웃 시스템에는 PPP용 직렬 회선이 하나만 있을 수 있습니다. 또한 다이얼 아웃 시스템에는 옵션이 거의 필요하지 않습니다. 다른 구성 파일 또는 명령줄에서 필요한 모든 옵션을 지정할 수 있습니다.

## options.ttya.tmpl 템플리트 파일

/etc/ppp/options.ttya.tmpl 파일에는 /etc/ppp/options.tty-name 파일에 대한 유용한 설명이 포함되어 있습니다. 템플리트에는 /etc/ppp/options.tty-name 파일에 대한 세 가지 일반적인 옵션이 포함되어 있습니다.

```
38400
asyncmap 0xa0000
:192.168.1.1
```

옵션	정의
38400	포트 ttya에 대해 이 변조 속도를 사용합니다.
asyncmap 0xa0000	로컬 시스템이 연결이 끊어진 피어와 통신할 수 있도록 asyncmap 값 0xa0000을 지정합니다.
:192.168.1.1	링크를 통해 호출하는 모든 피어에 IP 주소 192.168.1.1을 지정합니다.

사이트에서 /etc/ppp/options.ttya.tmpl을 사용하려면 /etc/ppp/options.tmpl의 이름을 /etc/ppp/options.ttya-name으로 바꾸고, ttya-name을 모뎀이 있는 직렬 포트의 이름으로 바꾼 다음 사이트의 필요에 따라 파일 내용을 수정하십시오.

## /etc/ppp/options.ttyname 파일의 예를 찾을 수 있는 위치

/etc/ppp/options.ttyname 파일의 예를 찾으려면 다음을 참조하십시오.

- 다이얼 아웃 시스템의 경우 408 페이지 “직렬 회선을 통해 통신을 정의하는 방법”을 참조하십시오.
- 다이얼 인 서버의 경우 415 페이지 “직렬 회선을 통해 통신을 정의하는 방법(다이얼 인 서버)”을 참조하십시오.



## 사용자별 옵션 구성

이 절에는 다이얼 인 서버에서 사용자를 설정하는 방법에 대한 자세한 내용이 포함되어 있습니다.

### 다이얼 인 서버에서 \$HOME/.ppprc 구성

\$HOME/.ppprc 파일은 기본 PPP 옵션을 구성하는 사용자를 위한 것입니다. 관리자는 사용자를 위해 \$HOME/.ppprc도 구성할 수 있습니다.

\$HOME/.ppprc의 옵션에는 해당 파일을 호출하는 사용자에게 권한이 있는 경우에만 권한이 부여됩니다.

호출자가 `pppd` 명령을 사용하여 호출을 시작하는 경우에는 .ppprc 파일이 `pppd` 데몬에 의해 확인되는 두번째 파일이 됩니다.

다이얼 인 서버에서 \$HOME/.ppprc를 설정하는 방법에 대한 자세한 내용은 [414 페이지 “다이얼 인 서버의 사용자 설정”](#)을 참조하십시오.

### 다이얼 아웃 시스템에서 \$HOME/.ppprc 구성

\$HOME/.ppprc 파일은 올바른 Solaris PPP 4.0 작동을 위해 다이얼 아웃 시스템에 필요하지 않습니다. 또한 특별한 경우를 제외하고는 \$HOME/.ppprc를 다이얼 아웃 시스템에 둘 필요가 없습니다. 다음과 같은 경우 하나 이상의 .ppprc 파일을 만드십시오.

- 다양한 통신 요구를 가진 여러 사용자가 동일한 시스템에서 원격 피어를 호출할 수 있게 허용합니다. 이러한 경우에는 다이얼 아웃해야 하는 각 사용자의 홈 디렉토리에서 개별 .ppprc 파일을 만드십시오.
- 본인의 링크와 관련된 문제를 제어하는 옵션을 지정해야 합니다(예: Van Jacobson 압축을 사용 안함으로 설정). 링크 문제 해결 관련 지원은 James Carlson의 *PPP Design, Implementation, and Debugging* 및 `pppd(1M)` 매뉴얼 페이지를 참조하십시오.

.ppprc 파일은 다이얼 인 서버를 구성할 때 가장 흔히 사용됩니다. .ppprc 구성 지침은 [414 페이지 “다이얼 인 서버의 사용자를 구성하는 방법”](#)을 참조하십시오.

## 다이얼 인 서버와의 통신을 위한 정보 지정

다이얼 인 서버와 통신하려면 서버에 대한 정보를 수집해야 합니다. 그런 다음 몇몇 파일을 편집합니다. 또한 다이얼 아웃 시스템이 호출해야 하는 모든 다이얼 인 서버의 통신 요구 사항을 구성하는 것이 가장 중요합니다. ISP 전화 번호와 같은 다이얼 인 서버에 대한 옵션을 `/etc/ppp/options.ttyname` 파일에 지정할 수 있습니다. 그러나 피어 정보를 구성하기에 가장 좋은 위치는 `/etc/ppp/peers/peer-name` 파일입니다.

## /etc/ppp/peers/peer-name 파일

주 - /etc/ppp/peers/*peer-name* 파일은 Solaris PPP 4.0의 올바른 작동을 위해 다이얼 아웃 시스템에 필요하지 않습니다.

/etc/ppp/peers/*peer-name* 파일을 사용하여 특정 피어와 통신하기 위한 정보를 제공할 수 있습니다. /etc/ppp/peers/*peer-name*을 사용하면 사용자가 설정할 수 없는 미리 선택된 권한이 지정된 옵션을 일반 사용자가 호출할 수 있습니다.

예를 들어, 권한이 부여되지 않은 사용자는 *noauth*가 /etc/ppp/peers/*peer-name* 파일에 지정된 경우 *noauth* 옵션을 대체할 수 없습니다. 사용자가 인증 자격 증명을 제공하지 않는 *peerB*에 대한 링크를 설정하려고 하는 경우 슈퍼 유저는 *noauth* 옵션이 포함된 /etc/ppp/peers/*peerB* 파일을 만들 수 있습니다. *noauth*는 로컬 시스템이 *peerB*로부터의 호출을 인증하지 않음을 나타냅니다.

pppd 데몬은 pppd가 다음 옵션을 발견할 때 /etc/ppp/peers/*peer-name*을 읽습니다.

call *peer-name*

다이얼 아웃 시스템이 통신해야 하는 각 대상 피어에 대해 /etc/ppp/peers/*peer-name* 파일을 만들 수 있습니다. 이는 일반 사용자가 루트 권한 없이 특정 다이얼 아웃 링크를 호출할 수 있게 허용하려는 경우 특히 유용합니다.

/etc/ppp/peers/*peer-name*에서 지정하는 일반적인 옵션은 다음과 같습니다.

- user *user-name*  
PAP 또는 CHAP를 사용하여 인증할 때 다이얼 인 서버에 *user-name*을 다이얼 아웃 시스템의 로그인 이름으로 제공합니다.
- remotename *peer-name*  
*peer-name*을 다이얼 인 시스템의 이름으로 사용합니다. /etc/ppp/pap-secrets 또는 /etc/ppp/chap-secrets 파일을 스캔할 때 remotename이 PAP 또는 CHAP 인증과 함께 사용됩니다.
- connect "chat *chat\_script* ..."  
채트 스크립트의 명령을 사용하여 다이얼 인 서버에 대한 통신을 엽니다.
- noauth  
통신을 시작할 때 피어 *peer-name*을 인증하지 않습니다.
- noipdefault  
피어와 협상할 때 사용되는 초기 IP 주소를 0.0.0.0으로 설정합니다. 피어 간의 IPCP 협상에 도움이 되도록 대부분의 ISP에 대한 링크를 설정할 때 noipdefault를 사용하십시오.
- defaultroute

링크에서 IP가 설정될 때 기본 IPv4 경로를 설치합니다.

특정 대상 피어에 적용될 수 있는 추가 옵션은 [pppd\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## /etc/ppp/peers/myisp.tmpl 템플릿 파일

/etc/ppp/peers/myisp.tmpl 파일에는 /etc/ppp/peers/*peer-name* 파일에 대한 유용한 설명이 포함되어 있습니다. 템플릿은 /etc/ppp/peers/*peer-name* 파일에 대해 사용할 수 있는 일반적인 옵션으로 끝납니다.

```
connect "/usr/bin/chat -f /etc/ppp/myisp-chat"
user myname
remotename myisp
noauth
noipdefault
defaultroute
updetach
noccp
```

옵션	정의
connect "/usr/bin/chat -f /etc/ppp/myisp-chat"	채트 스크립트 /etc/ppp/myisp-chat를 사용하여 피어를 호출합니다.
user myname	로컬 시스템에 이 계정 이름을 사용합니다. myname은 피어의 /etc/ppp/pap-secrets 파일에서 이 시스템의 이름입니다.
remotename myisp	myisp를 로컬 시스템의 /etc/ppp/pap-secrets 파일에서 피어의 이름으로 인식합니다.
noauth	인증 자격 증명을 제공하기 위해 피어를 호출하지 않아도 됩니다.
noipdefault	로컬 시스템에 기본 IP 주소를 사용하지 않습니다.
defaultroute	로컬 시스템에 지정된 기본 경로를 사용합니다.
updetach	오류를 표준 출력 대신 PPP 로그 파일에 기록합니다.
noccp	CCP 압축을 사용하지 않습니다.

사이트에서 /etc/ppp/peers/myisp.tmpl을 사용하려면 /etc/ppp/peers/myisp.tmpl의 이름을 /etc/ppp/peers/.*peer-name*으로 바꾸고, *peer-name*을 호출할 피어의 이름으로 바꾼 다음 사이트의 필요에 따라 파일 내용을 수정하십시오.

## /etc/ppp/peers/peer-name 파일의 예를 찾을 수 있는 위치

/etc/ppp/peers/peer-name 파일의 예를 찾으려면 다음을 참조하십시오.

- 다이얼 아웃 시스템의 경우 [410 페이지](#) “개별 피어를 사용하여 연결을 정의하는 방법”을 참조하십시오.
- 전용 회선에 있는 로컬 시스템의 경우 [421 페이지](#) “전용 회선에서 시스템을 구성하는 방법”을 참조하십시오.
- 다이얼 아웃 시스템에서의 PAP 인증 지원의 경우 [432 페이지](#) “PPP 구성 파일에 PAP 지원을 추가하는 방법(다이얼 아웃 시스템)”을 참조하십시오.
- 다이얼 아웃 시스템에서의 CHAP 인증 지원의 경우 [438 페이지](#) “PPP 구성 파일에 CHAP 지원을 추가하는 방법(다이얼 아웃 시스템)”을 참조하십시오.
- 클라이언트 시스템에서의 PPPoE 지원의 경우 [440 페이지](#) “PPPoE 클라이언트 설정”을 참조하십시오.

## 다이얼 업 링크를 위한 모뎀 속도 구성

모뎀 구성에서의 대표적인 문제는 모뎀 작동 속도를 지정하는 것입니다. 다음 지침은 Sun Microsystems 컴퓨터에 사용되는 모뎀에 적용됩니다.

- 이전 SPARC 시스템 - 시스템과 함께 제공되는 하드웨어 설명서를 확인하십시오. 대부분의 SPARCstation 시스템에서는 모뎀 속도가 38400bps를 넘지 않아야 합니다.
- UltraSPARC 시스템 - 모뎀 속도를 115200bps로 설정합니다. 이는 최신 모뎀에 유용하며 다이얼 업 링크에 사용하기에 충분히 빠릅니다. 압축과 함께 이중 채널 ISDN TA를 사용하려는 경우에는 모뎀 속도를 높여야 합니다. UltraSPARC에 대한 제한은 비동기 링크의 경우 460800bps입니다.

다이얼 아웃 시스템의 경우 /etc/ppp/peers/peer-name과 같은 PPP 구성 파일에서 모뎀 속도를 설정하거나 속도를 pppd에 대한 옵션으로 지정하십시오.

다이얼 인 서버의 경우 [412 페이지](#) “다이얼 인 서버에서 장치 구성”에 설명된 대로 ttymon 기능을 사용하여 속도를 설정해야 합니다.

## 다이얼 업 링크에서 대화 정의

다이얼 아웃 시스템과 해당 원격 피어는 다양한 명령을 협상 및 교환하여 PPP 링크를 통해 통신합니다. 다이얼 아웃 시스템을 구성할 때는 로컬 및 원격 모뎀에 필요한 명령을 확인해야 합니다. 그런 다음 이러한 명령이 포함된 채트 스크립트라는 파일을 만듭니다. 이 절에는 모뎀 구성 및 채트 스크립트 만들기에 대한 정보가 설명되어 있습니다.

## 채트 스크립트의 내용

다이얼아웃 시스템이 연결해야 하는 각 원격 피어에는 자체 채트 스크립트가 필요합니다.

주 - 채트 스크립트는 일반적으로 다이얼업 링크에서만 사용됩니다. 전용 회선 링크의 경우 링크에 시작 구성이 필요한 비동기 인터페이스가 포함되어 있지 않은 한 채트 스크립트가 사용되지 않습니다.

채트 스크립트의 내용은 모뎀 모델 또는 ISDN TA 및 원격 피어의 요구 사항에 따라 결정됩니다. 이러한 내용은 *expect-send* 문자열 세트로 나타납니다. 다이얼아웃 시스템과 해당 원격 피어는 통신 시작 프로세스의 일환으로 문자열을 교환합니다.

*expect* 문자열에는 다이얼아웃 호스트 시스템이 대화를 시작하기 위해 원격 피어에게서 받아야 하는 문자가 포함되어 있습니다. *send* 문자열에는 다이얼아웃 시스템이 *expect* 문자열을 받은 후 원격 피어에게 보내는 문자가 포함되어 있습니다.

채트 스크립트의 정보에는 일반적으로 다음이 포함되어 있습니다.

- 모뎀 명령 - 보통 **AT 명령**이라고 하며, 모뎀이 전화를 통해 데이터를 전송할 수 있게 합니다.
- 대상 피어의 전화 번호  
이 전화 번호는 ISP, 회사 사이트에 있는 다이얼인 서버 또는 개별 시스템에 필요한 번호일 수 있습니다.
- 시간 초과 값(필요한 경우)
- 원격 피어에 필요한 로그인 절차
- 다이얼아웃 시스템이 보낸 로그인 절차

## 채트 스크립트 예

이 절에는 사용자 고유 채트 스크립트를 만들기 위한 참조로 사용할 수 있는 채트 스크립트가 포함되어 있습니다. 모뎀 제조업체의 설명서와 ISP 및 기타 대상 호스트가 제공하는 정보에는 모뎀 및 대상 피어에 대한 채트 요구 사항이 포함되어 있습니다. 또한 많은 PPP 웹 사이트에 샘플 채트 스크립트가 있습니다.

### 기본적인 모뎀 채트 스크립트

다음은 사용자 고유 채트 스크립트를 만들기 위한 템플릿으로 사용할 수 있는 기본적인 채트 스크립트입니다.

```
ABORT BUSY
ABORT 'NO CARRIER'
REPORT CONNECT
```

```
TIMEOUT 10
"" AT&F1M0&M5S2=255
SAY      "Calling myserver\n"
TIMEOUT 60
OK       "ATDT1-123-555-1212"
ogin: pppuser
ssword: \q\U
% pppd
```

다음 표에서는 채트 스크립트의 내용에 대해 설명합니다.

스크립트 내용	설명
ABORT BUSY	모뎀이 반대쪽 피어로부터 이 메시지를 받는 경우 전송을 중단합니다.
ABORT 'NO CARRIER'	전화를 걸 때 모뎀이 ABORT 'NO CARRIER'를 보고하는 경우 전송을 중단합니다. 이 메시지는 일반적으로 전화 걸기 또는 모뎀 협상 실패로 인해 표시됩니다.
REPORT CONNECT	모뎀에서 CONNECT 문자열을 수집합니다. 문자열을 인쇄합니다.
TIMEOUT 10	초기 시간 초과 값을 10초로 설정합니다. 모뎀의 응답이 즉시 이루어집니다.
"" AT&F1M0&M5S2=255	M0 - 연결 중 스피커를 끕니다. &M5 - 모뎀에 오류 제어가 필요하게 만듭니다. S2=255 - TIES “+++” 중단 시퀀스를 사용 안함으로 설정합니다.
SAY "Calling myserver\n"	로컬 시스템에서 Calling myserver 메시지를 표시합니다.
TIMEOUT 60	링크 협상에 더 많은 시간이 사용될 수 있도록 시간 초과 값을 60초로 재설정합니다.
OK "ATDT1-123-555-1212"	전화 번호 123-555-1212를 사용하여 원격 피어를 호출합니다.
ogin: pppuser	UNIX 스타일의 로그인을 사용하여 피어에 로그인합니다. 사용자 이름 pppuser를 제공합니다.
ssword: \q\U	\q - -v 옵션을 사용하여 디버깅하는 경우 기록하지 않습니다. \U - -u(명령줄에 지정) 다음에 오는 문자열의 내용을 이 위치에 삽입합니다. 일반적으로 문자열에는 암호가 포함됩니다.
% pppd	% 셸 프롬프트를 기다렸다가 pppd 명령을 실행합니다.

**/etc/ppp/myisp-chat.tmpl 채트 스크립트 템플리트**

이번 릴리스에는 사용자 사이트에서 사용하기 위해 수정할 수 있는 /etc/ppp/myisp-chat.tmpl이 포함되어 있습니다. /etc/ppp/myisp-chat.tmpl은 템플리트에 로그인 절차가 포함되어 있지 않다는 점을 제외하고 기본적인 모뎀 채트 스크립트와 유사합니다.

```

ABORT    BUSY
ABORT    'NO CARRIER'
REPORT   CONNECT
TIMEOUT  10
""       "AT&F1"
OK       "AT&C1&D2"
SAY      "Calling myisp\n"
TIMEOUT  60
OK       "ATDT1-123-555-1212"
CONNECT  \c
    
```

스크립트 내용	설명
ABORT BUSY	모뎀이 반대쪽 피어로부터 이 메시지를 받는 경우 전송을 중단합니다.
ABORT 'NO CARRIER	전화를 걸 때 모뎀이 ABORT 'NO CARRIER'를 보고하는 경우 전송을 중단합니다. 이 메시지는 일반적으로 전화 걸기 또는 모뎀 협상 실패로 인해 표시됩니다.
REPORT CONNECT	모뎀에서 CONNECT 문자열을 수집합니다. 문자열을 인쇄합니다.
TIMEOUT 10	초기 시간 초과 값을 10초로 설정합니다. 모뎀의 응답이 즉시 이루어집니다.
"" "AT&F1"	모뎀을 출하시의 기본값으로 재설정합니다.
OK "AT&C1&D2"	&C1의 경우 모뎀의 DCD가 반송파를 따르도록 모뎀을 재설정합니다. 어떤 이유로 원격측이 전화를 끊으면 DCD가 감소합니다.  &D2의 경우 DTR이 높음에서 낮음으로 전이될 때 모뎀이 "온후크" 상태가 되거나 정지됩니다.
SAY "Calling myisp\n"	로컬 시스템에서 "Calling myisp" 메시지를 표시합니다.
TIMEOUT 60	링크 협상에 더 많은 시간이 사용될 수 있도록 시간 초과 값을 60초로 재설정합니다.
OK "ATDT1-123-555-1212"	전화 번호 123-555-1212를 사용하여 원격 피어를 호출합니다.
CONNECT \c	반대쪽 피어의 모뎀에서 CONNECT 메시지를 기다립니다.

## ISP 호출을 위한 모뎀 채트 스크립트

다음 채트 스크립트를 U.S. Robotics Courier 모뎀이 설치된 다이얼 아웃 시스템에서 ISP를 호출하기 위한 템플릿으로 사용하십시오.

```

ABORT    BUSY
ABORT    'NO CARRIER'
REPORT   CONNECT
TIMEOUT  10
""       AT&F1M0&M5S2=255
    
```

```
SAY "Calling myisp\n"
TIMEOUT 60
OK "ATDT1-123-555-1212"
CONNECT \c
\r \d\c
SAY "Connected; running PPP\n"
```

다음 표에서는 채트 스크립트의 내용에 대해 설명합니다.

스크립트 내용	설명
ABORT BUSY	모뎀이 반대쪽 피어로부터 이 메시지를 받는 경우 전송을 중단합니다.
ABORT 'NO CARRIER'	모뎀이 반대쪽 피어로부터 이 메시지를 받는 경우 전송을 중단합니다.
REPORT CONNECT	모뎀에서 CONNECT 문자열을 수집합니다. 문자열을 인쇄합니다.
TIMEOUT 10	초기 시간 초과 값을 10초로 설정합니다. 모뎀의 응답이 즉시 이루어집니다.
"" AT&F1M0M0M0M0&M5S2=255	M0 - 연결 중 스피커를 끕니다. &M5 - 모뎀에 오류 제어가 필요하게 만듭니다. S2=255 - TIES “+++” 중단 시퀀스를 사용 안함으로 설정합니다.
SAY "Calling myisp\n"	로컬 시스템에서 Calling myisp 메시지를 표시합니다.
TIMEOUT 60	링크 협상에 더 많은 시간이 사용될 수 있도록 시간 초과 값을 60초로 재설정합니다.
OK "ATDT1-123-555-1212"	전화 번호 123-555-1212를 사용하여 원격 피어를 호출합니다.
CONNECT \c	반대쪽 피어의 모뎀에서 CONNECT 메시지를 기다립니다.
\r \d\c	CONNECT 메시지의 끝까지 기다립니다.
SAY "Connected; running PPP\n"	로컬 시스템에서 Connected; running PPP 정보 메시지를 표시합니다.

UNIX 스타일의 로그인을 위해 향상된 기본적인 채트 스크립트

다음 채트 스크립트는 원격 Oracle Solaris 피어 또는 기타 UNIX 유형의 피어를 호출하기 위해 향상된 기본적인 스크립트입니다. 이 채트 스크립트는 [409 페이지 “피어 호출 명령을 만드는 방법”](#)에 사용됩니다.

```
SAY "Calling the peer\n"
TIMEOUT 10
ABORT BUSY
ABORT 'NO CARRIER'
ABORT ERROR
REPORT CONNECT
"" AT&F1&M5S2=255
TIMEOUT 60
OK ATDT1-123-555-1234
CONNECT \c
SAY "Connected; logging in.\n"
```



```

TIMEOUT 5
ogin:--ogin: pppuser
TIMEOUT 20
ABORT 'ogin incorrect'
ssword: \qmypassword
"% " \c
SAY "Logged in. Starting PPP on peer system.\n"
ABORT 'not found'
"" "exec pppd"
~ \c

```

다음 표에서는 채트 스크립트의 매개변수에 대해 설명합니다.

스크립트 내용	설명
TIMEOUT 10	초기 시간 초과 값을 10초로 설정합니다. 모뎀의 응답이 즉시 이루어집니다.
ABORT BUSY	모뎀이 반대쪽 피어로부터 이 메시지를 받는 경우 전송을 중단합니다.
ABORT 'NO CARRIER'	모뎀이 반대쪽 피어로부터 이 메시지를 받는 경우 전송을 중단합니다.
ABORT ERROR	모뎀이 반대쪽 피어로부터 이 메시지를 받는 경우 전송을 중단합니다.
REPORT CONNECT	모뎀에서 <b>CONNECT</b> 문자열을 수집합니다. 문자열을 인쇄합니다.
"" AT&F1&M5S2=255	<b>&amp;M5</b> - 모뎀에 오류 제어가 필요하게 만듭니다. <b>S2=255</b> - <b>TIES "++"</b> 중단 시퀀스를 사용 안함으로 설정합니다.
TIMEOUT 60	링크 협상에 더 많은 시간이 사용될 수 있도록 시간 초과 값을 60초로 재설정합니다.
OK ATDT1-123-555-1234	전화 번호 123-555-1212를 사용하여 원격 피어를 호출합니다.
CONNECT \c	반대쪽 피어의 모뎀에서 <b>CONNECT</b> 메시지를 기다립니다.
SAY "Connected; logging in.\n"	사용자 상태를 제공하기 위해 <b>Connected; logging in</b> 정보 메시지를 표시합니다.
TIMEOUT 5	로그인 프롬프트를 빠르게 표시할 수 있도록 시간 초과 값을 변경합니다.
ogin:--ogin: pppuser	로그인 프롬프트를 기다립니다. 프롬프트를 받지 못하면 <b>RETURN</b> 을 보내고 기다립니다. 그런 다음 사용자 이름 <b>pppuser</b> 를 피어에게 보냅니다. 후속 절차는 대부분의 ISP가 <b>PAP</b> 로그인으로 참조합니다. 그러나 <b>PAP</b> 로그인은 어떤 방식으로든 <b>PAP</b> 인증과 관련되어 있지 않습니다.
TIMEOUT 20	암호가 더 천천히 확인될 수 있도록 시간 초과 값을 20초로 변경합니다.

스크립트 내용	설명
ssword: \qmysecrethere	피어로부터 암호 프롬프트를 기다립니다. 프롬프트를 받으면 암호 \qmysecrethere를 보냅니다. \q는 시스템 로그 파일에 암호가 기록되지 않게 합니다.
% " \c	피어로부터 셸 프롬프트를 기다립니다. 채트 스크립트에는 C 셸이 사용됩니다. 사용자가 다른 셸을 사용하여 로그인하는 것을 선호하는 경우 이 값을 변경하십시오.
SAY "Logged in. Starting PPP on peer system.\n"	사용자 상태를 제공하기 위해 Logged in. Starting PPP on peer system 정보 메시지를 표시합니다.
ABORT 'not found'	셸에 오류가 발생하는 경우 전송을 중단합니다.
"" "exec pppd"	피어에서 pppd를 시작합니다.
~ \c	피어에서 PPP가 시작되기를 기다립니다.

CONNECT \c 바로 다음에 PPP를 시작하는 것을 ISP는 보통 **PAP 로그인**이라고 부릅니다. 그러나 실제로 PAP 로그인은 PAP 인증의 일부가 아닙니다.

ogin:--ogin: pppuser라는 문구는 다이얼 인 서버의 로그인 프롬프트에 대한 응답으로 사용자 이름 pppuser를 보내도록 모뎀에 지시합니다. pppuser는 다이얼 인 서버에서 원격 user1에 대해 만들어진 특수 PPP 사용자 계정 이름입니다. 다이얼 인 서버에서 PPP 사용자 계정을 만드는 방법에 대한 자세한 내용은 [414 페이지 “다이얼 인 서버의 사용자를 구성하는 방법”](#)을 참조하십시오.

## 외부 ISDN TA를 위한 채트 스크립트

다음은 ZyXEL omni.net ISDN TA를 사용하는 다이얼 아웃 시스템에서 호출하기 위한 채트 스크립트입니다.

```
SAY "Calling the peer\n"
TIMEOUT 10
ABORT BUSY
ABORT 'NO CARRIER'
ABORT ERROR
REPORT CONNECT
"" AT&FB40S83.7=1&K44&J3X7S61.3=1S0=0S2=255
OK ATDI18882638234
CONNECT \c
\r \d\c
SAY "Connected; running PPP\n"
```

다음 표에서는 채트 스크립트의 매개변수에 대해 설명합니다.

스크립트 내용	설명
SAY "Calling the peer"	다이얼 아웃 시스템의 화면에 이 메시지를 표시합니다.

스크립트 내용	설명
TIMEOUT 10	초기 시간 초과 값을 10초로 설정합니다.
ABORT BUSY	모뎀이 반대쪽 피어로부터 이 메시지를 받는 경우 전송을 중단합니다.
ABORT 'NO CARRIER'	모뎀이 반대쪽 피어로부터 이 메시지를 받는 경우 전송을 중단합니다.
ABORT ERROR	모뎀이 반대쪽 피어로부터 이 메시지를 받는 경우 전송을 중단합니다.
REPORT CONNECT	모뎀에서 CONNECT 문자열을 수집합니다. 문자열을 인쇄합니다.
"" AT&FB40S83.7=1&K44&J3X7S61.3=1S0=0S2=255	이 행에 있는 문자의 의미는 다음과 같습니다. <ul style="list-style-type: none"> <li>■ &amp;F - 출하시의 기본값 사용</li> <li>■ B40 - 비동기 PPP 변환 수행</li> <li>■ S83.7=1 - DOSB(Data Over Speech Bearer) 사용</li> <li>■ &amp;K44 - CCP 압축을 사용으로 설정</li> <li>■ &amp;J3 - MP를 사용으로 설정</li> <li>■ X7 - DCE측 속도 보고</li> <li>■ S61.3=1 - 패킷 단편화 사용</li> <li>■ S0=0 - 자동 응답 안함</li> <li>■ S2=255 - TIES 제어를 사용 안함으로 설정</li> </ul>
OK ATDI18882638234	ISDN 호출을 실행합니다. 다중 연결의 경우 두번째 호출이 동일한 전화 번호에 대해 실행됩니다(일반적으로 대부분의 ISP에서 이렇게 요구함). 원격 피어에 다른 두번째 전화 번호가 필요한 경우에는 "+nnnn"을 추가하십시오. nnnn은 두번째 전화 번호를 나타냅니다.
CONNECT \c	반대쪽 피어의 모뎀에서 CONNECT 메시지를 기다립니다.
\r \d\c	CONNECT 메시지의 끝까지 기다립니다.
SAY "Connected; running PPP\n"	다이얼 아웃 시스템의 화면에 이 메시지를 표시합니다.

옵션 설명 및 채트 스크립트에 대한 기타 자세한 내용은 [chat\(1M\)](#) 매뉴얼 페이지를 참조하십시오. expect-send 문자열에 대한 자세한 내용은 [531 페이지 "/etc/uucp/Systems 파일의 Chat-Script 필드"](#)를 참조하십시오.

## 추가 채트 스크립트 예

많은 웹 사이트에서 샘플 채트 스크립트를 얻고 채트 스크립트 만들기 작업에 대한 지원을 받을 수 있습니다. 예는 <http://ppp.samba.org/ppp/index.html>을 참조하십시오.

## 채트 스크립트 호출

채트 스크립트는 `connect` 옵션을 사용하여 호출합니다. `connect "chat ..."`를 모든 PPP 구성 파일 또는 명령줄에서 사용할 수 있습니다.

채트 스크립트는 실행 가능하지 않지만 `connect`로 호출하는 프로그램은 실행 가능해야 합니다. `chat` 유틸리티를 `connect`로 호출하는 프로그램으로 사용할 수 있습니다. 이 경우 `-f` 옵션을 통해 채트 스크립트를 외부 파일에 저장하면 채트 스크립트 파일을 실행할 수 없습니다.

`chat(1m)`에 설명된 `chat` 프로그램은 실제 채트 스크립트를 실행합니다. `pppd` 데몬은 `pppd`가 `connect "chat ..."` 옵션을 발견할 때마다 `chat` 프로그램을 호출합니다.

주 - Perl 또는 Tcl과 같은 임의의 외부 프로그램을 사용하여 고급 채트 스크립트를 만들 수 있습니다. `chat` 유틸리티는 편의상 제공됩니다.

### ▼ 채트 스크립트를 호출하는 방법(작업)

- 1 채트 스크립트를 ASCII 파일로 만듭니다.
- 2 다음 구문을 사용하여 임의의 PPP 구성 파일에서 채트 스크립트를 호출합니다.  

```
connect 'chat -f /etc/ppp/chatfile'
```

`-f` 플래그는 다음에 파일 이름이 나옴을 나타냅니다. `/etc/ppp/chatfile`은 채트 파일의 이름을 나타냅니다.
- 3 `pppd` 명령을 실행하는 사용자에게 외부 채트 파일에 대한 읽기 권한을 부여합니다.



주의 - `chat` 프로그램은 `connect 'chat ...'` 옵션을 권한 있는 소스에서 호출한 경우에도 항상 사용자의 권한으로 실행됩니다. 따라서 `-f` 옵션으로 읽는 별도의 채트 파일은 호출 사용자가 읽을 수 있어야 합니다. 채트 스크립트에 암호나 기타 민감한 정보가 포함되어 있는 경우 이 권한으로 인해 보안 문제가 발생할 수 있습니다.

#### 예 22-1 인라인 채트 스크립트

다음과 같이 전체 채트 스크립트 대화를 한 행에 넣을 수 있습니다.

```
connect 'chat "" "AT&F1" OK ATDT5551212 CONNECT "\c"'
```

전체 채트 스크립트가 `chat` 키워드 다음에 오고 `"\c"`로 종료됩니다. 이 형식은 모든 PPP 구성 파일 또는 명령줄에서 `pppd`에 대한 인수로 사용합니다.

## 자세한 정보 외부 파일의 채트스크립트

특정 피어에 필요한 채트스크립트가 길거나 복잡한 경우 스크립트를 별도의 파일로 만들어 보십시오. 외부 채트 파일은 유지 관리 및 문서화가 쉽습니다. 설명 앞에 해시(#) 기호를 추가하여 채트 파일에 설명을 추가할 수 있습니다.

409 페이지 "피어 호출 명령을 만드는 방법" 절차에서는 외부 파일에 포함되어 있는 채트 스크립트의 사용을 보여줍니다.

## 실행 가능한 채트 파일 만들기

다이얼업 링크가 시작될 때 자동으로 실행할 실행 가능한 스크립트로 채트 파일을 만들 수 있습니다. 이렇게 하면 링크 시작 중 기존 채트스크립트에 포함된 명령 외에 추가 명령(예: 패리티 설정을 위한 `stty`)을 실행할 수 있습니다.

이 실행 가능한 채트 스크립트는 짝수 패리티가 포함된 7비트를 필요로 하는 기존 스타일의 UNIX 시스템에 로그인합니다. 그러면 시스템이 PPP를 실행할 때 패리티가 없는 8비트로 변경됩니다.

```
#!/bin/sh
chat "" "AT&F1" OK "ATDT555-1212" CONNECT "\c"
stty evenp
chat ogin: pppuser ssword: "\q\U" % "exec pppd"
stty -evenp
```

## ▼ 실행 가능한 chat 프로그램을 만드는 방법

- 1 텍스트 편집기를 사용하여 이전 예와 같은 실행 가능한 chat 프로그램을 만듭니다.

- 2 chat 프로그램을 실행 가능하게 만듭니다.

```
# chmod +x /etc/ppp/chatprogram
```

- 3 chat 프로그램을 호출합니다.

```
connect /etc/ppp/chatprogram
```

chat 프로그램은 /etc/ppp 파일 시스템 내에 있지 않아도 되며, 어느 위치에나 저장될 수 있습니다.

## 링크에서 호출자 인증

이 절에서는 PPP 인증 프로토콜의 작동 방법을 설명하고 인증 프로토콜과 연관된 데이터베이스에 대해 설명합니다.

### PAP(암호 인증 프로토콜)

PAP 인증은 작동 면에서 UNIX login 프로그램과 유사합니다. 그러나 PAP는 사용자에게 셸 액세스 권한을 부여하지 않습니다. PAP는 `/etc/ppp/pap-secrets` 파일의 형태로 PPP 구성 파일 및 PAP 데이터베이스를 사용하여 인증을 설정합니다. 또한 PAP는 `/etc/ppp/pap-secrets`를 사용하여 PAP 보안 자격 증명을 정의합니다. 이러한 자격 증명에는 피어 이름, PAP 방식의 “사용자 이름” 및 암호가 포함되어 있습니다. 또한 PAP 자격 증명에는 로컬 시스템에 연결할 수 있는 각 호출자에 대한 관련 정보가 포함되어 있습니다. PAP 사용자 이름 및 암호는 암호 데이터베이스에 있는 UNIX 사용자 이름 및 암호와 같거나 다를 수 있습니다.

#### `/etc/ppp/pap-secrets` 파일

PAP 데이터베이스는 `/etc/ppp/pap-secrets` 파일에 구현됩니다. 인증에 성공하려면 PPP 링크의 양쪽에 있는 시스템의 `/etc/ppp/pap-secrets` 파일에서 PAP 자격 증명을 제대로 구성해야 합니다. 호출자(피인증자)는 사용되지 않는 `+ua` 파일이나 `/etc/ppp/pap-secrets` 파일의 `user` 및 `password` 열에서 자격 증명을 제공합니다. 서버(인증자)는 UNIX `passwd` 데이터베이스를 통해 `/etc/ppp/pap-secrets`의 정보를 기준으로 이러한 자격 증명을 검증하거나 PAM 기능의 정보를 기준으로 자격 증명을 검증합니다.

`/etc/ppp/pap-secrets` 파일의 구문은 다음과 같습니다.

```
myclient ISP-server mypassword *
```

매개변수의 의미는 다음과 같습니다.

<b>myclient</b>	호출자의 PAP 사용자 이름입니다. 이 이름은 호출자의 UNIX 사용자 이름과 동일한 경우가 많으며, 이는 다이얼 인 서버가 PAP의 login 옵션을 사용하는 경우에 특히 그렇습니다.
<b>ISP-server</b>	원격 시스템(보통 다이얼 인 서버)의 이름입니다.
<b>mypassword</b>	호출자의 PAP 암호입니다.
<b>*</b>	호출자에 연결된 IP 주소입니다. IP 주소를 나타낼 때는 별표(*)를 사용하십시오.

## PAP 암호 만들기

PAP 암호는 **암호화되지 않은 형식**, 즉 읽을 수 있는 ASCII 형식으로 링크를 통해 전송됩니다. 호출자(피인증자)의 경우 PAP 암호를 암호화되지 않은 형식으로 다음 위치 중 하나에 저장해야 합니다.

- /etc/ppp/pap-secrets에
- 다른 외부 파일에
- pap-secrets@ 기능을 통해 명명된 파이프에
- pppd에 대한 옵션으로 명령줄 또는 PPP 구성 파일에
- +ua 파일을 통해

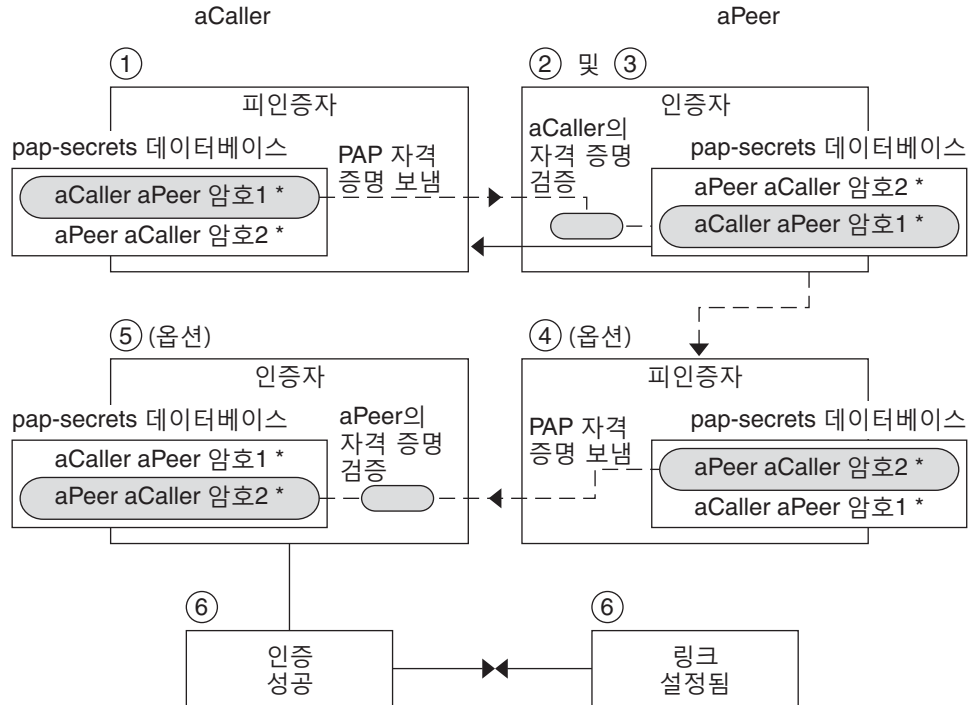
서버(인증자)에서는 다음 중 하나를 수행하여 PAP 암호를 숨길 수 있습니다.

- pap-secrets 파일에서 papcrypt를 지정하고 crypt(3C)로 해시된 암호 사용
- pppd에 login 옵션을 지정하고, 암호 열에 큰따옴표("")를 추가하여 pap-secrets 파일에서 암호 생략. 이 경우 인증은 UNIX passwd 데이터베이스 또는 PAM 방식을 통해 수행됩니다.

## PAP 인증 중 발생하는 작업

PAP 인증은 다음 순서대로 발생합니다.

그림 22-1 PAP 인증 프로세스



1. 호출자(피인증자)가 원격 피어(인증자)를 호출하고 해당 PAP 사용자 이름 및 암호를 링크 협상의 일환으로 제공합니다.
2. 피어가 해당 /etc/ppp/pap-secrets 파일에서 호출자의 ID를 확인합니다. 피어가 PAP의 login 옵션을 사용하는 경우에는 피어가 해당 암호 데이터베이스에서 호출자의 사용자 이름 및 암호를 확인합니다.
3. 인증에 성공하면 피어가 호출자와 링크 협상을 계속합니다. 인증에 실패하면 링크가 삭제됩니다.
4. (옵션) 호출자가 원격 피어의 응답을 인증하는 경우 원격 피어가 자체 PAP 자격 증명을 호출자에게 보내야 합니다. 따라서 원격 피어가 피인증자가 되고 호출자가 인증자가 됩니다.
5. (옵션) 원래 호출자가 자체 /etc/ppp/pap-secrets를 읽어 원격 피어의 ID를 확인합니다.

주 - 원래 호출자가 원격 피어로부터 인증 자격 증명을 요구하는 경우에는 1단계와 4단계가 병렬로 발생합니다.



- 피어가 인증되면 협상이 계속됩니다. 그렇지 않으면 링크가 삭제됩니다.
6. 링크가 성공적으로 설정될 때까지 호출자와 피어 간의 협상이 계속됩니다.

## **/etc/ppp/pap-secrets에 login 옵션 사용**

PAP 자격 증명을 인증하기 위한 login 옵션을 모든 PPP 구성 파일에 추가할 수 있습니다. /etc/ppp/options 등에 login을 지정하면 pppd가 호출자의 PAP 자격 증명(암호 데이터베이스에 있는지 확인합니다. 다음은 login 옵션이 포함된 /etc/ppp/pap-secrets 파일의 형식입니다.

```
joe      *   ""   *
sally    *   ""   *
sue      *   ""   *
```

매개변수의 의미는 다음과 같습니다.

**호출자**     joe, sally 및 sue가 권한이 부여된 호출자의 이름입니다.

**서버**       별표(\*)로, 모든 서버 이름이 유효함을 나타냅니다. PPP 구성 파일에서는 name 옵션이 필요하지 않습니다.

**암호**       큰따옴표로, 모든 암호가 유효함을 나타냅니다.

이 열에 암호가 있는 경우에는 피어의 암호가 PAP 암호 및 UNIX passwd 데이터베이스 모두와 일치해야 합니다.

**IP 주소**     별표(\*)로, 모든 IP 주소가 허용됨을 나타냅니다.

## **CHAP(Challenge-Handshake 인증 프로토콜)**

CHAP 인증에는 **챌린지**와 **응답**이라는 개념이 사용됩니다. 즉, 피어(인증자)가 호출자(피인증자)에게 자신의 ID를 증명하도록 요구합니다. 챌린지에는 인증자가 생성한 고유한 ID와 난수가 포함됩니다. 호출자는 해당 ID, 난수 및 CHAP 보안 자격 증명을 사용하여 피어에게 보낼 적절한 응답(핸드셰이크)을 생성해야 합니다.

CHAP 보안 자격 증명에는 CHAP 사용자 이름 및 CHAP “암호”가 포함됩니다. CHAP 암호는 호출자와 피어가 PPP 링크를 협상하기 전에 양쪽 모두에 알려지는 임의 문자열입니다. CHAP 보안 자격 증명은 CHAP 데이터베이스인 /etc/ppp/chap-secrets에서 구성합니다.

## **/etc/ppp/chap-secrets 파일**

CHAP 데이터베이스는 /etc/ppp/chap-secrets 파일에서 구현됩니다. 인증에 성공하려면 PPP 링크 양쪽에 있는 시스템의 해당 /etc/ppp/chap-secrets 파일에 서로의 CHAP 자격 증명이 있어야 합니다.

---

주-PAP와 달리 공유 암호는 두 피어 모두에서 암호화되지 않은 형식으로 있어야 합니다. CHAP에서는 crypt, PAM 또는 PPP 로그인 옵션을 사용할 수 없습니다.

---

/etc/ppp/chap-secrets 파일의 구문은 다음과 같습니다.

```
myclient myserver secret5748 *
```

매개변수의 의미는 다음과 같습니다.

myclient	호출자의 CHAP 사용자 이름입니다. 이 이름은 호출자의 UNIX 사용자 이름과 같거나 다를 수 있습니다.
myserver	원격 시스템(보통 다이얼 인 서버)의 이름입니다.
secret5748	호출자의 CHAP 암호입니다.

---

주-PAP 암호와 달리 CHAP 암호는 절대 링크를 통해 전송되지 않습니다. 대신 CHAP 암호는 로컬 시스템이 응답을 계산할 때 사용됩니다.

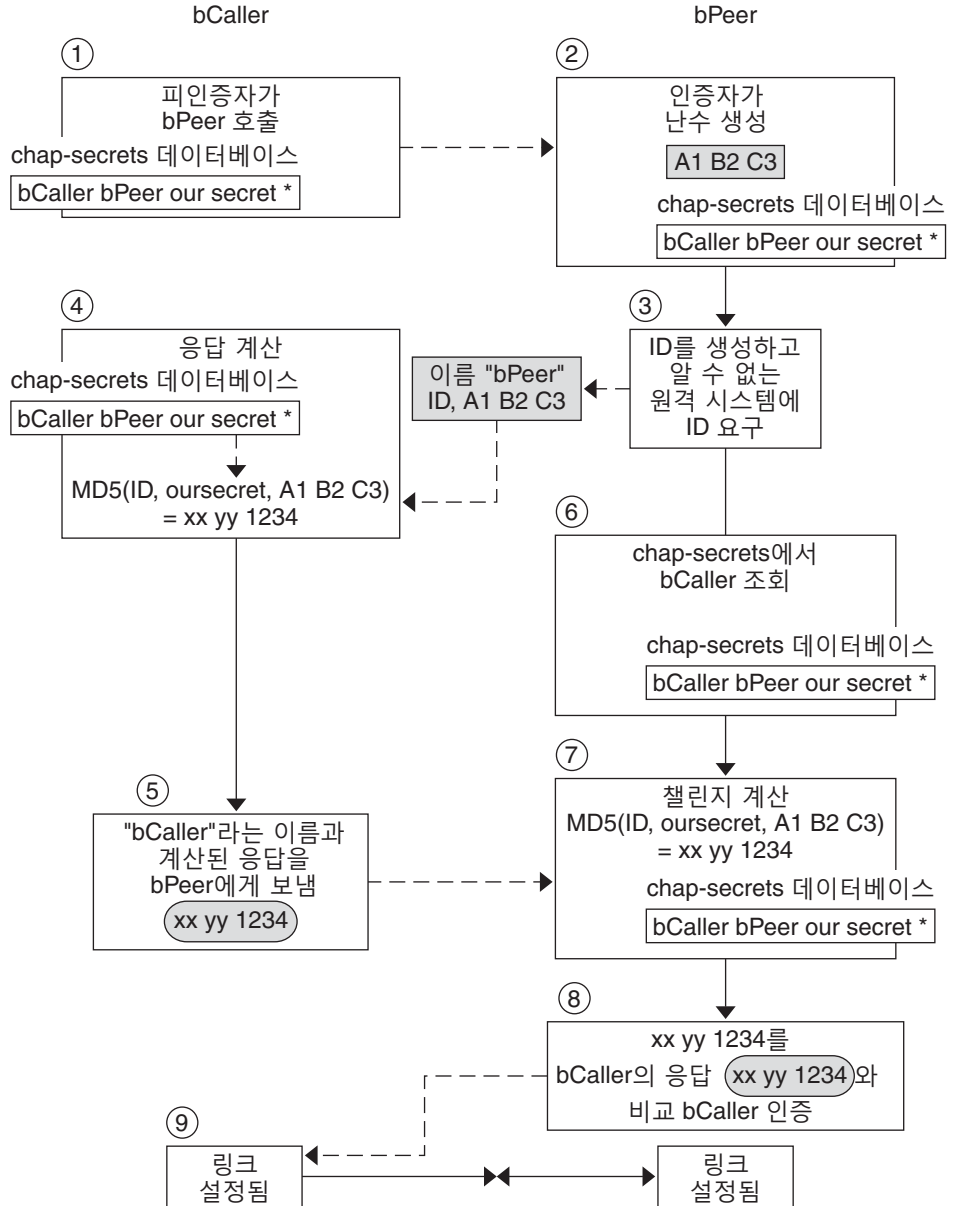
---

\* 호출자에 연결된 IP 주소입니다. IP 주소를 나타낼 때는 별표(\*)를 사용하십시오.

## CHAP 인증 중 발생하는 작업

CHAP 인증은 다음 순서대로 발생합니다.

그림 22-2 CHAP 인증 순서



1. 통신을 시작하려고 하는 두 피어가 PPP 링크 협상 중 인증에 사용될 암호에 대해 합의를 봅니다.

2. 두 시스템의 관리자가 암호, CHAP 사용자 이름 및 기타 CHAP 자격 증명을 해당 시스템의 `/etc/ppp/chap-secrets` 데이터베이스에 추가합니다.
3. 호출자(피인증자)가 원격 피어(인증자)를 호출합니다.
4. 인증자가 난수 및 ID를 생성하고 이 데이터를 피인증자에게 챌린지로 보냅니다.
5. 피인증자가 해당 `/etc/ppp/chap-secrets` 데이터베이스에서 피어의 이름 및 암호를 조회합니다.
6. 피인증자가 암호 및 피어의 난수 챌린지에 MD5 계산 알고리즘을 적용하여 응답을 계산합니다. 그런 다음 피인증자가 결과를 인증자에게 응답으로 보냅니다.
7. 인증자가 해당 `/etc/ppp/chap-secrets` 데이터베이스에서 피인증자의 이름 및 암호를 조회합니다.
8. 인증자가 `/etc/ppp/chap-secrets`에서 챌린지로 생성된 숫자와 피인증자의 암호에 MD5를 적용하여 자체 결과를 계산합니다.
9. 인증자가 자신의 결과를 호출자의 응답과 비교합니다. 두 숫자가 같으면 피어가 호출자를 성공적으로 인증한 것이며 링크 협상이 계속됩니다. 그렇지 않으면 링크가 삭제됩니다.

## 호출자를 위한 IP 주소 지정 체계 만들기

각 원격 사용자에게 고유한 IP 주소를 지정하는 대신 모든 수신 호출에 대해 하나 이상의 IP 주소를 만들어 보십시오. 전용 IP 주소는 잠재적 호출자의 수가 다이얼 인 서버에 있는 직렬 포트 및 모뎀의 수를 초과하는 경우 특히 중요합니다. 사이트의 요구에 따라 다양한 여러 시나리오를 구현할 수 있습니다. 여러 시나리오를 동시에 사용할 수도 있습니다.

### 호출자에게 동적 IP 주소 지정

동적 주소 지정 작업에서는 각 호출자에게 `/etc/ppp/options.ttyname`에 정의되어 있는 IP 주소를 지정합니다. 동적 주소 지정 작업은 직렬 포트별로 발생합니다. 호출이 직렬 회선을 통해 도착하면 호출자가 호출의 직렬 인터페이스에 대한 `/etc/ppp/options.ttyname` 파일에 있는 IP 주소를 받습니다.

수신 호출에 다이얼 업 서비스를 제공하는 직렬 인터페이스 4개가 다이얼 인 서버에 있는 경우를 예로 들어 보겠습니다.

- 직렬 포트 `term/a`의 경우 다음 항목을 사용하여 `/etc/ppp/options.term.a` 파일을 만듭니다.  
:10.1.1.1
- 직렬 포트 `term/b`의 경우 다음 항목을 사용하여 `/etc/ppp/options.term.b` 파일을 만듭니다.

:10.1.1.2

- 직렬 포트 `term/c`의 경우 다음 항목을 사용하여 `/etc/ppp/options.term.c` 파일을 만듭니다.

:10.1.1.3

- 직렬 포트 `term/d`의 경우 다음 항목을 사용하여 `/etc/ppp/options.term.d` 파일을 만듭니다.

:10.1.1.4

이전 주소 지정 체계를 사용하여 직렬 인터페이스 `/dev/term/c`에 있는 수신 호출에 호출 기간 동안 IP 주소 10.1.1.3이 지정됩니다. 첫번째 호출자가 전화를 끊으면 직렬 인터페이스 `/dev/term/c`를 통해 들어오는 후속 호출에도 IP 주소 10.1.1.3이 지정됩니다.

동적 주소 지정의 이점은 다음과 같습니다.

- 직렬 포트까지 PPP 네트워크 사용을 추적할 수 있습니다.
- PPP용으로 최소한의 IP 주소를 지정할 수 있습니다.
- IP 필터링을 보다 간편하게 관리할 수 있습니다.

## 호출자에게 정적 IP 주소 지정

사이트에서 PPP 인증을 구현하는 경우 개별 호출자에게 특정 정적 IP 주소를 지정할 수 있습니다. 이 경우 다이얼 아웃 시스템이 다이얼 인 서버를 호출할 때마다 호출자가 동일한 IP 주소를 받습니다.

정적 주소는 `pap-secrets` 또는 `chap-secrets` 데이터베이스에서 구현합니다. 다음은 정적 IP 주소를 정의하는 `/etc/ppp/pap-secrets` 파일의 예입니다.

```
joe  myserver  joepasswd  10.10.111.240
sally myserver sallypasswd 10.10.111.241
sue   myserver suepasswd   10.10.111.242
```

호출자     joe, sally 및 sue가 권한이 부여된 호출자의 이름입니다.

서버       myserver가 서버의 이름을 나타냅니다.

암호       joepasswd, sallypasswd 및 suepasswd가 각 호출자의 암호를 나타냅니다.

IP 주소     10.10.111.240, 10.10.111.241 및 10.10.111.242가 각 호출자에게 지정된 IP 주소입니다.

다음은 정적 IP 주소를 정의하는 `/etc/ppp/chap-secrets` 파일의 예입니다.

```
account1 myserver secret5748 10.10.111.244
account2 myserver secret91011 10.10.111.245
```

호출자     account1 및 account2가 호출자의 이름을 나타냅니다.

서버        myserver가 각 호출자에 대한 서버의 이름을 나타냅니다.  
암호        secret5748 및 secret91011이 각 호출자의 CHAP 암호를 나타냅니다.  
IP 주소     10.10.111.244 및 10.10.111.245가 각 호출자의 IP 주소입니다.

## sppp 장치 번호별로 IP 주소 지정

PAP 또는 CHAP 인증을 사용하는 경우 sppp 장치 번호별로 호출자에게 IP 주소를 지정할 수 있습니다. 다음은 이러한 사용법의 예입니다.

myclient ISP-server mypassword 10.10.111.240/28+

플러스 기호(+)는 장치 번호가 IP 주소에 추가되었음을 나타냅니다. 다음 사항에 유의하십시오.

- 10.10.111.240부터 10.10.111.255까지의 주소는 원격 사용자에게 지정됩니다.
- sppp0은 IP 주소 10.10.111.240을 받습니다.
- sppp1은 IP 주소 10.10.111.241을 받습니다. 이런 식으로 계속됩니다.

## DSL 지원을 위해 PPPoE 터널 만들기

PPPoE를 사용하면 하나 이상의 DSL 모뎀을 사용하고 있는 여러 클라이언트에게 고속 디지털 서비스를 통해 PPP를 제공할 수 있습니다. PPPoE는 세 참가자(기업, 전화 회사 및 서비스 공급자)를 통해 이더넷 터널을 만들어 이러한 서비스를 구현합니다.

- PPPoE의 작동 방법에 대한 개요 및 설명은 [386 페이지 “PPPoE 개요”](#)를 참조하십시오.
- PPPoE 터널 설정 작업은 [20 장, “PPPoE 터널 설정\(작업\)”](#)을 참조하십시오.

이 절에는 다음 표에 요약되어 있는 PPPoE 명령 및 파일에 대한 자세한 정보가 포함되어 있습니다.

표 22-2 PPPoE 명령 및 구성 파일

파일 또는 명령	설명	수행 방법
/etc/ppp/pppoe	시스템에서 PPPoE에 의해 설정된 모든 터널에 기본적으로 적용되는 특징이 포함된 파일입니다.	<a href="#">497 페이지 “/etc/ppp/pppoe 파일”</a>
/etc/ppp/pppoe.device	PPPoE가 터널에 사용하는 특정 인터페이스의 특징이 포함된 파일입니다.	<a href="#">499 페이지 “/etc/ppp/pppoe.device 파일”</a>
/etc/ppp/pppoe.if	PPPoE에 의해 설정된 터널이 실행되는 이더넷 인터페이스가 나열된 파일입니다.	<a href="#">495 페이지 “/etc/ppp/pppoe.if 파일”</a>

표 22-2 PPPoE 명령 및 구성 파일 (계속)

파일 또는 명령	설명	수행 방법
<code>/usr/sbin/sppptun</code>	PPPoE 터널에 참여하는 이더넷 인터페이스를 구성하기 위한 명령입니다.	495 페이지 “ <code>/usr/sbin/sppptun</code> 명령”
<code>/usr/lib/inet/pppoed</code>	터널 설정을 위해 PPPoE를 사용하기 위한 명령 및 옵션입니다.	497 페이지 “ <code>/usr/lib/inet/pppoed</code> 데몬”

## PPPoE용 인터페이스를 구성하기 위한 파일

PPPoE 터널의 각 끝에서 사용되는 인터페이스를 구성해야 해당 터널이 PPP 통신을 지원할 수 있습니다. 이렇게 하려면 `/usr/sbin/sppptun` 및 `/etc/ppp/pppoe.if` 파일을 사용하십시오. 이러한 도구를 사용하여 모든 Oracle Solaris PPPoE 클라이언트 및 PPPoE 액세스 서버에서 이더넷 인터페이스를 구성해야 합니다.

### `/etc/ppp/pppoe.if` 파일

`/etc/ppp/pppoe.if` 파일에는 PPPoE 터널에 사용될 호스트의 모든 이더넷 인터페이스 이름이 나열됩니다. 이 파일은 나열된 인터페이스가 PPPoE 터널에 사용될 수 있도록 연결될 때 시스템 부트 도중 처리됩니다.

`/etc/ppp/pppoe.if`를 명시적으로 만들어야 합니다. PPPoE용으로 구성할 한 인터페이스의 이름을 각 행에 입력하십시오.

다음 예에서는 PPPoE 터널을 위한 세 가지 인터페이스를 제공하는 서버에 대한 `/etc/ppp/pppoe.if` 파일을 보여줍니다.

```
# cat /etc/ppp/pppoe.if
hme1
hme2
hme3
```

PPPoE 클라이언트에는 일반적으로 `/etc/ppp/pppoe.if`에 나열된 하나의 인터페이스만 있습니다.

### `/usr/sbin/sppptun` 명령

`/usr/sbin/sppptun` 명령을 사용하여 PPPoE 터널에 사용될 이더넷 인터페이스를 수동으로 연결 및 연결 취소할 수 있습니다. 반대로 `/etc/ppp/pppoe.if`는 시스템 부트 시에만 읽힙니다. 이러한 인터페이스는 `/etc/ppp/pppoe.if`에 나열된 인터페이스와 일치해야 합니다.

`sppptun`은 PPPoE 터널에 사용되는 이더넷 인터페이스를 `ipadm` 명령과 유사한 방식으로 연결합니다. 그러나 두 개의 이더넷 프로토콜 번호가 사용되므로 `ipadm`과 달리 PPPoE를 지원하기 위해 인터페이스를 두 번 연결해야 합니다.

`sppptun`의 기본 구문은 다음과 같습니다.

```
# /usr/sbin/sppptun plumb pppoe device-name
device-name:pppoed
# /usr/sbin/sppptun plumb pppoe device-name
device-name:pppoe
```

이 구문에서 *device-name*은 PPPoE에 연결될 장치의 이름입니다.

sppptun 명령을 처음 실행하면 검색 프로토콜 pppoe가 인터페이스에서 연결됩니다. sppptun을 두번째로 실행하면 세션 프로토콜 pppoe가 연결됩니다. sppptun은 방금 연결된 인터페이스의 이름을 인쇄합니다. 필요한 경우 이 이름을 사용하여 인터페이스의 연결을 취소합니다.

자세한 내용은 [sppptun\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## 인터페이스 관리를 위한 sppptun 명령의 예

다음 예에서는 /usr/sbin/sppptun을 사용하여 PPPoE용 인터페이스를 수동으로 연결하는 방법을 보여줍니다.

```
# /usr/sbin/sppptun plumb pppoe hme0
hme0:pppoed
# /dev/sppptun plumb pppoe hme0
hme0:pppoe
```

이 예에서는 PPPoE에 연결된 액세스 서버의 인터페이스를 나열하는 방법을 보여줍니다.

```
# /usr/sbin/sppptun query
hme0:pppoe
hme0:pppoed
hme1:pppoe
hme1:pppoed
hme2:pppoe
hme2:pppoed
```

이 예에서는 인터페이스의 연결을 취소하는 방법을 보여줍니다.

```
# sppptun unplumb hme0:pppoed
# sppptun unplumb hme0:pppoe
```

## PPPoE 액세스 서버 명령 및 파일

DSL 서비스 또는 지원을 고객에게 제공하는 서비스 공급자는 PPPoE를 실행하는 액세스 서버를 사용할 수 있습니다. PPPoE 액세스 서버 및 클라이언트는 기존의 클라이언트-서버 관계에서 동작합니다. 이 관계는 다이얼 업 링크에 있는 다이얼 인 서버와 다이얼 아웃 시스템의 관계와 유사합니다. 한 PPPoE 시스템이 통신을 시작하고 한 PPPoE 시스템이 응답합니다. 반대로 PPP 프로토콜에는 클라이언트-서버 관계라는 개념이 없습니다. PPP는 두 시스템을 모두 동등한 피어로 간주합니다.



PPPoE 액세스 서버를 설정하는 명령 및 파일은 다음과 같습니다.

- 495 페이지 “/usr/sbin/sppptun 명령”
- 497 페이지 “/usr/lib/inet/pppoed 데몬”
- 497 페이지 “/etc/ppp/pppoe 파일”
- 499 페이지 “/etc/ppp/pppoe.device 파일”
- 502 페이지 “pppoe.so 공유 객체”

## /usr/lib/inet/pppoed 데몬

pppoed 데몬은 잠재적 PPPoE 클라이언트로부터 서비스용 브로드캐스트를 받습니다. 또한 pppoeed는 PPPoE 터널의 서버측을 협상하고 PPP 데몬인 pppd를 해당 터널을 통해 실행합니다.

pppoed 서비스는 /etc/ppp/pppoe 및 /etc/ppp/pppoe.device 파일에서 구성합니다. 시스템 부트 시 /etc/ppp/pppoe가 있으면 pppoeed가 자동으로 실행됩니다. 명령줄에서 /usr/lib/inet/pppoed를 입력하여 pppoeed 데몬을 명시적으로 실행할 수도 있습니다.

## /etc/ppp/pppoe 파일

/etc/ppp/pppoe 파일에는 액세스 서버가 제공하는 서비스와 PPP가 PPPoE 터널을 통해 실행되는 방법을 정의하는 방법이 설명되어 있습니다. 개별 인터페이스에 대해 서비스를 정의하거나 액세스 서버에 있는 모든 인터페이스에 대해 전역적으로 서비스를 정의할 수 있습니다. 액세스 서버는 잠재적 PPPoE 클라이언트로부터 받은 브로드캐스트에 대한 응답으로 /etc/ppp/pppoe 파일의 정보를 보냅니다.

/etc/ppp/pppoe의 기본 구문은 다음과 같습니다.

```
global-options
service service-name
    service-specific-options
    device interface-name
```

매개변수의 의미는 다음과 같습니다.

*global-options*

/etc/ppp/pppoe 파일에 대한 기본 옵션을 설정합니다. 이러한 옵션은 pppoeed 또는 pppd를 통해 사용 가능한 모든 옵션일 수 있습니다. 전체 옵션 목록은 **pppoed(1M)** 및 **pppd(1M)** 매뉴얼 페이지를 참조하십시오.

예를 들어, PPPoE 터널에 대해 사용 가능한 이더넷 인터페이스를 *global options*의 일부로 나열해야 합니다. /etc/ppp/pppoe에서 장치를 정의하지 않으면 어떤 인터페이스에서도 서비스가 제공되지 않습니다.

**devices**를 전역 옵션으로 정의하려면 다음 형식을 사용하십시오.

*device interface <,interface>*

*interface*는 서비스가 잠재적 PPPoE 클라이언트에 대해 수신 대기하는 인터페이스를 지정합니다. 둘 이상의 인터페이스가 서비스와 연관되는 경우에는 각 이름을 쉼표로 구분하십시오.

*service service-name* *service-name* 서비스의 정의를 시작합니다. *service-name*은 제공되는 서비스에 적합한 모든 구문일 수 있는 문자열입니다.

*service-specific-options* 이 서비스와 관련된 PPPoE 및 PPP 옵션을 나열합니다.

*device interface-name* 이전에 나열된 서비스를 사용할 수 있는 인터페이스를 지정합니다.

/etc/ppp/pppoe에 대한 추가 옵션은 **pppoed(1M)** 및 **pppd(1M)** 매뉴얼 페이지를 참조하십시오.

일반적인 /etc/ppp/pppoe 파일은 다음과 같을 수 있습니다.

**예 22-2** 기본적인 /etc/ppp/pppoe 파일

```
device hme1,hme2,hme3
service internet
    pppd "name internet-server"
service intranet
    pppd "192.168.1.1:"
service debug
    device hme1
    pppd "debug name internet-server"
```

이 파일에서는 다음 값이 적용됩니다.

hme1,hme2,hme3	PPPoE 터널에 사용될 액세스 서버의 세 인터페이스입니다.
service internet	<b>internet</b> 이라는 서비스를 잠재적 클라이언트에게 알립니다. 서비스를 제공하는 공급자가 <b>internet</b> 의 정의 방법도 결정합니다. 예를 들어, <b>internet</b> 이 다양한 IP 서비스와 인터넷에 대한 액세스를 의미하는 것으로 공급자가 해석할 수 있습니다.
pppd	호출자가 <b>pppd</b> 를 호출할 때 사용되는 명령줄 옵션을 설정합니다. "name internet-server" 옵션은 로컬 시스템(액세스 서버)의 이름을 <b>internet-server</b> 로 제공합니다.
service intranet	<b>intranet</b> 이라는 다른 서비스를 잠재적 클라이언트에게 알립니다.

<code>pppd "192.168.1.1:"</code>	호출자가 <code>pppd</code> 를 호출할 때 사용되는 명령줄 옵션을 설정합니다. 호출자가 <code>pppd</code> 를 호출하면 192.168.1.1이 로컬 시스템(액세스 서버)의 IP 주소로 설정됩니다.
<code>service debug</code>	PPPoE에 대해 정의된 인터페이스에서 세번째 서비스인 디버깅을 알립니다.
<code>device hme1</code>	PPPoE 터널에 대한 디버깅을 <code>hme1</code> 로 제한합니다.
<code>pppd "debug name internet-server"</code>	호출자가 <code>pppd</code> 를 호출할 때 사용되는 명령줄 옵션을 설정합니다. 이 경우에는 로컬 시스템인 <code>internet-server</code> 의 PPP 디버깅입니다.

## **/etc/ppp/pppoe.device 파일**

`/etc/ppp/pppoe.device` 파일에는 PPPoE 액세스 서버의 한 인터페이스에서 제공되는 서비스가 설명되어 있습니다. `/etc/ppp/pppoe.device`에는 PPP가 PPPoE 터널을 통해 실행되는 방법을 정의하는 옵션도 포함되어 있습니다. `/etc/ppp/pppoe.device`는 전역 `/etc/ppp/pppoe`와 동일하게 동작하는 선택적 파일입니다. 그러나 `/etc/ppp/pppoe.device`가 인터페이스에 대해 정의되어 있으면 해당 매개변수가 `/etc/ppp/pppoe`에 정의된 전역 매개변수보다 해당 인터페이스에 대해 우선적으로 사용됩니다.

`/etc/ppp/pppoe.device`의 기본 구문은 다음과 같습니다.

```
service service-name
    service-specific-options
service another-service-name
    service-specific-options
```

이 구문과 `/etc/ppp/pppoe`의 구문은 497 페이지 “`/etc/ppp/pppoe` 파일”에 나와 있는 `device` 옵션을 사용할 수 없다는 점에서만 다릅니다.

## **pppoe.so 플러그인**

`pppoe.so`는 PPPoE 액세스 서버 및 클라이언트가 호출해야 하는 PPPoE 공유 객체 파일입니다. 이 파일은 `pppoed`와 함께 MTU 및 MRU를 1492로 제한하고, 드라이버로부터의 패킷을 필터링하고, PPPoE 터널을 협상합니다. 액세스 서버측에서 `pppoe.so`는 `pppd` 데몬을 통해 자동으로 호출됩니다.

## **PPPoE 및 PPP 파일을 사용하여 액세스 서버 구성**

이 절에는 액세스 서버를 구성하는 데 사용되는 모든 파일의 샘플이 포함되어 있습니다. 액세스 서버는 멀티홈 방식을 사용합니다. 이 서버는 `green`, `orange` 및 `purple`이라는 세 서브넷에 연결됩니다. `pppoed`는 서버에서 `root`로 실행됩니다(기본값).

PPPoE 클라이언트는 hme0 및 hme1 인터페이스를 통해 orange 및 purple 네트워크에 액세스합니다. 클라이언트는 표준 UNIX 로그인을 사용하여 서버에 로그인합니다. 서버는 PAP를 사용하여 클라이언트를 인증합니다.

green 네트워크는 클라이언트에게 알려지지 않습니다. 클라이언트는 직접 “green-net”을 지정하고 CHAP 인증 자격 증명을 제공해야만 green에 액세스할 수 있습니다. 또한 joe 및 mary 클라이언트만 정적 IP 주소를 사용하여 green 네트워크에 액세스할 수 있습니다.

예 22-3 액세스 서버를 위한 /etc/ppp/pppoe 파일

```
service orange-net
    device hme0,hme1
    pppd "require-pap login name orange-server orange-server:"
service purple-net
    device hme0,hme1
    pppd "require-pap login name purple-server purple-server:"
service green-net
    device hme1
    pppd "require-chap name green-server green-server:"
    nowildcard
```

이 샘플에서는 액세스 서버에서 사용 가능한 서비스에 대해 설명합니다. 첫번째 서비스 섹션에서는 orange 네트워크의 서비스에 대해 설명합니다.

```
service orange-net
    device hme0,hme1
    pppd "require-pap login name orange-server orange-server:"
```

클라이언트는 hme0 및 hme1 인터페이스를 통해 orange 네트워크에 액세스합니다. pppd 명령에 제공되는 옵션으로 인해 서버가 잠재적 클라이언트에게 PAP 자격 증명을 요구해야 합니다. 또한 pppd 옵션은 pap-secrets 파일에 사용되는 대로 서버의 이름을 orange-server로 설정합니다.

purple 네트워크의 서비스 섹션은 네트워크 및 서버 이름만 제외하고 orange 네트워크의 서비스 섹션과 동일합니다.

다음 섹션에서는 green 네트워크의 서비스에 대해 설명합니다.

```
service green-net
    device hme1
    pppd "require-chap name green-server green-server:"
    nowildcard
```

이 섹션에서는 클라이언트 액세스를 hme1 인터페이스로 제한합니다. pppd 명령에 제공되는 옵션으로 인해 서버가 잠재적 클라이언트에게 CHAP 자격 증명을 요구해야 합니다. 또한 pppd 옵션은 chap-secrets 파일에 사용될 대로 서버 이름을 green-server로 설정합니다. nowildcard 옵션은 green 네트워크의 존재 여부를 클라이언트에 알리지 않도록 지정합니다.

방금 설명한 이 액세스 서버 시나리오의 경우 다음 `/etc/ppp/options` 파일을 설정할 수 있습니다.

예 22-4 액세스 서버를 위한 `/etc/ppp/options` 파일

```
auth
proxyarp
nodefaultroute
name no-service      # don't authenticate otherwise
```

`name no-service` 옵션은 PAP 또는 CHAP 인증 중 일반적으로 검색되는 서버 이름을 대체합니다. 서버의 기본 이름은 `/usr/bin/hostname` 명령을 통해 검색되는 이름입니다. 이전 예에 있는 `name` 옵션은 서버의 이름을 `no-service`로 변경합니다. 이름 `no-service`는 `pap` 또는 `chap-secrets` 파일에서 검색될 확률이 적습니다. 이 작업을 수행하면 임의 사용자가 `pppd`를 실행하고 `/etc/ppp/options`에 설정된 `auth` 및 `name` 옵션을 대체하지 못하게 됩니다. 그러면 서버 이름이 `no-service`인 클라이언트에 대해 암호가 검색될 수 없기 때문에 `pppd`가 실패합니다.

액세스 서버 시나리오에는 다음 `/etc/hosts` 파일이 사용됩니다.

예 22-5 액세스 서버를 위한 `/etc/hosts` 파일

```
172.16.0.1    orange-server
172.17.0.1    purple-server
172.18.0.1    green-server
172.18.0.2    joes-pc
172.18.0.3    marys-pc
```

`orange` 및 `purple` 네트워크에 액세스하려고 시도하는 클라이언트에 대한 PAP 인증에 사용되는 `/etc/ppp/pap-secrets` 파일은 다음과 같습니다.

예 22-6 액세스 서버를 위한 `/etc/ppp/pap-secrets` 파일

```
* orange-server "" 172.16.0.2/16+
* purple-server "" 172.17.0.2/16+
```

CHAP 인증에 사용되는 `/etc/ppp/chap-secrets` 파일은 다음과 같습니다. `joe` 및 `mary` 클라이언트만 이 파일에 나열됩니다.

예 22-7 액세스 서버를 위한 `/etc/ppp/chap-secrets` 파일

```
joe green-server "joe's secret" joes-pc
mary green-server "mary's secret" marys-pc
```

## PPPoE 클라이언트 명령 및 파일

DSL 모뎀을 통해 PPP를 실행하려면 시스템이 PPPoE 클라이언트가 되어야 합니다. PPPoE를 실행하려면 인터페이스를 연결한 다음 `pppoe` 유틸리티를 사용하여 액세스 서버의 존재 여부를 “발견”해야 합니다. 그러면 클라이언트가 DSL 모뎀을 통해 PPPoE 터널을 만들어 PPP를 실행할 수 있게 됩니다.

PPPoE 클라이언트는 기존의 클라이언트-서버 모델에서 액세스 서버와 관련됩니다. PPPoE 터널은 다이얼 업 링크가 아니지만 거의 동일한 방식으로 구성되어 작동합니다.

PPPoE 클라이언트를 설정하는 명령 및 파일은 다음과 같습니다.

- 495 페이지 “`/usr/sbin/sppptun` 명령”
- 502 페이지 “`/usr/lib/inet/pppoe` 유틸리티”
- 502 페이지 “`pppoe.so` 공유 객체”
- 474 페이지 “`/etc/ppp/peers/peer-name` 파일”
- 469 페이지 “`/etc/ppp/options` 구성 파일”

### `/usr/lib/inet/pppoe` 유틸리티

`/usr/lib/inet/pppoe` 유틸리티는 PPPoE 터널의 클라이언트측을 협상합니다. `pppoe`는 `chat` 유틸리티와 유사합니다. `pppoe`는 직접 호출하지 않습니다. 대신 `pppd`의 `connect` 옵션에 대한 인수로 `/usr/lib/inet/pppoe`를 시작합니다.

### `pppoe.so` 공유 객체

`pppoe.so`는 PPPoE 기능을 액세스 서버 및 클라이언트에 제공하기 위해 PPPoE가 로드해야 하는 PPPoE 공유 객체입니다. `pppoe.so` 공유 객체는 MTU 및 MRU를 1492로 제한하고, 드라이버로부터의 패킷을 필터링하고, 런타임 PPPoE 메시지를 처리합니다.

클라이언트측에서 `pppd`는 사용자가 `plugin pppoe.so` 옵션을 지정할 때 `pppoe.so`를 로드합니다.

### 액세스 서버 피어를 정의하기 위한 `/etc/ppp/peers/peer-name` 파일

`pppoe`를 통해 발견될 액세스 서버를 정의할 때는 `pppoe`와 `pppd` 둘 모두에 적용되는 옵션을 사용합니다. 액세스 서버를 위한 `/etc/ppp/peers/peer-name` 파일에 필요한 매개변수는 다음과 같습니다.

- `sppptun` – PPPoE 터널에 사용되는 직렬 장치의 이름입니다.
- `plugin pppoe.so` – `pppoe.so` 공유 객체를 로드하도록 `pppd`에 지시합니다.
- `connect "/usr/lib/inet/pppoe device"` – 연결을 시작합니다. 그러면 `connect`가 PPPoE에 연결된 인터페이스인 `device`를 통해 `pppoe` 유틸리티를 호출합니다.

`/etc/ppp/peers/peer-name` 파일의 나머지 매개변수는 서버의 PPP 링크에 적용되어야 합니다. 다이얼 아웃 시스템에서 `/etc/ppp/peers/peer-name`에 대해 사용할 것과 같은 옵션을 사용하십시오. PPP 링크에 필요한 옵션의 수를 최소한으로 제한하는 것이 좋습니다.

다음 예는 441 페이지 “PPPoE 액세스 서버 피어를 정의하는 방법”에 소개되어 있습니다.

예 22-8 원격 액세스 서버를 정의하기 위한 `/etc/ppp/peers/peer-name`

```
# cat /etc/ppp/peers/dslserve
sppptun
plugin pppoe.so
connect "/usr/lib/inet/pppoc hme0"
noccp
noauth
user Red
password redsecret
noipdefault
defaultroute
```

이 파일은 액세스 서버 `dslserve`에 대해 PPPoE 터널과 PPP 링크를 설정할 때 사용될 매개변수를 정의합니다. 포함되는 옵션은 다음과 같습니다.

옵션	설명
<code>sppptun</code>	<code>sppptun</code> 을 직렬 장치의 이름으로 정의합니다.
<code>plugin pppoe.so</code>	<code>pppoe.so</code> 공유 객체를 로드하도록 <code>pppd</code> 에 지시합니다.
<code>connect "/usr/lib/inet/pppoc hme0"</code>	<code>pppoc</code> 를 실행하고 <code>hme0</code> 을 PPPoE 터널 및 PPP 링크에 대한 인터페이스로 지정합니다.
<code>noccp</code>	링크에서 CCP 압축을 끕니다.  주 - 많은 ISP가 독점 압축 알고리즘만 사용합니다. 공개적으로 사용 가능한 CCP 알고리즘을 끄면 협상 시간을 절약하고 드물기는 하지만 때로 발생하는 상호 운용성 문제를 방지할 수 있습니다.
<code>noauth</code>	<code>pppd</code> 가 액세스 서버에 인증 자격 증명을 더 이상 요구하지 않게 만듭니다. 대부분의 ISP가 고객에게 인증 자격 증명을 제공하지 않습니다.
<code>user Red</code>	<code>Red</code> 라는 이름을 클라이언트의 사용자 이름으로 설정합니다. 이 이름은 액세스 서버의 PAP 인증에 필요합니다.
<code>password redsecret</code>	<code>redsecret</code> 을 PAP 인증을 위해 액세스 서버에 제공할 암호로 정의합니다.
<code>noipdefault</code>	0.0.0.0을 초기 IP 주소로 지정합니다.

옵션	설명
defaultroute	IPCP 협상 후에 기본 IPv4 경로를 설치하도록 <code>pppd</code> 에 지시합니다. 링크가 인터넷에 대한 시스템의 링크인 경우(PPPoE 클라이언트에 해당) <code>/etc/ppp/peers/peer-name</code> 에 <code>defaultroute</code> 를 포함해야 합니다.



## 비동기 Solaris PPP에서 Solaris PPP 4.0으로 마이그레이션(작업)

---

이전 버전의 Oracle Solaris OS에는 다른 PPP 구현인 비동기 Solaris PPP(asppp)가 포함되어 있었습니다. asppp를 실행하는 피어를 최신 PPP 4.0으로 변환하려면 변환 스크립트를 실행해야 합니다. 이 장에서는 PPP 변환에 대한 다음 항목을 다룹니다.

- 505 페이지 “asppp 파일을 변환하기 전에”
- 508 페이지 “asppp2pppd 변환 스크립트 실행(작업)”

이 장에서는 샘플 asppp 구성을 사용하여 PPP 변환을 수행하는 방법에 대해 설명합니다. Solaris PPP 4.0과 asppp의 차이점에 대한 자세한 내용을 보려면 376 페이지 “사용할 Solaris PPP 버전”으로 이동하십시오.

### asppp 파일을 변환하기 전에

변환 스크립트 /usr/sbin/asppp2pppd를 사용하여 표준 asppp 구성을 이루는 파일을 변환할 수 있습니다.

- /etc/asppp.cf – 비동기 PPP 구성 파일
- /etc/uucp/Systems – 원격 피어의 특징을 설명하는 UUCP 파일
- /etc/uucp/Devices – 로컬 시스템의 모뎀을 설명하는 UUCP 파일
- /etc/uucp/Dialers – /etc/uucp/Devices 파일에 설명된 모뎀에 사용될 로그인 절차가 포함된 UUCP 파일

asppp에 대한 자세한 내용은 *Solaris 8 System Administration Collection, Volume 3*(<http://docs.sun.com>에서 제공)을 참조하십시오.

### /etc/asppp.cf 구성 파일의 예

508 페이지 “asppp에서 Solaris PPP 4.0으로 변환하는 방법”에 나와 있는 절차에는 다음 /etc/asppp.cf 파일이 사용됩니다.

```
#
ipadm create-if ipdptp0
ipadm create-addr -T static -a local=mojave,remote=gobi ipdptp0/ppaddr
path
  inactivity_timeout 120      # Approx. 2 minutes
  interface ipdptp0
  peer_system_name Pgobi      # The name we log in with (also in
                              # /etc/uucp/Systems
```

이 파일에 포함된 매개변수는 다음과 같습니다.

```
ifpadm create-if ipdptp0
```

ifpadm 명령을 실행하여 ipdptp0이라는 인터페이스를 만듭니다.

```
ipadm create-addr -T static -a local=mojave,remote=gobi ipdptp0/ppadd
```

ipadm 명령을 실행하여 PPP 인터페이스 ipdptp0(로컬 시스템 mojave에 있음)에서 원격 피어 gobi로의 링크를 구성합니다.

inactivity\_timeout 120  
2분 동안 작업이 없을 경우 회선을 종료합니다.

```
interface ipdptp0
비동기 PPP를 위해 다이얼 아웃 시스템에서 ipdptp0 인터페이스를 구성합니다.
```

peer\_system\_name Pgobi  
원격 피어의 이름인 Pgobi를 제공합니다.

## `/etc/uucp/Systems` 파일의 예

508 페이지 “asppp에서 Solaris PPP 4.0으로 변환하는 방법”에 나와 있는 절차에는 다음 /etc/uucp/Systems 파일이 사용됩니다.

```
#ident "@(#)Systems 1.5 92/07/14 SMI" /* from SVR4 bnu:Systems 2.4 */
#
```

```
# .
# .
Pgobi Any ACU 38400 15551212 in:--in: mojave word: sand
```

이 파일에 포함된 매개변수는 다음과 같습니다.

Pgobi	원격 피어의 호스트 이름으로 Pgobi를 사용합니다.
Any ACU	다이얼 아웃 시스템 mojava의 모뎀에 하루 중 임의의 시간에 Pgobi에 있는 모뎀과 링크를 설정하도록 지시합니다. Any ACU는 “/etc/uucp/Devices 파일에서 ACU를 찾을 것”을 의미합니다.
38400	38400을 링크의 최대 속도로 설정합니다.
15551212	Pgobi의 전화 번호를 제공합니다.

in:-in: mojave word: sand Pgobi가 다이얼 아웃 시스템 mojave를 인증하는 데 필요한 로그인 스크립트를 정의합니다.

## /etc/uucp/Devices 파일의 예

508 페이지 “asppp에서 Solaris PPP 4.0으로 변환하는 방법”에 나와 있는 절차에는 다음 /etc/uucp/Devices 파일이 사용됩니다.

```
#ident "@(#)Devices 1.6 92/07/14 SMI" /* from SVR4 bnu:Devices 2.7 */

.
.
#

TCP,et - - Any TCP -
.
.
#
ACU cua/b - Any hayes
# 0-7 are on a Magma 8 port card
Direct cua/0 - Any direct
Direct cua/1 - Any direct
Direct cua/2 - Any direct
Direct cua/3 - Any direct
Direct cua/4 - Any direct
Direct cua/5 - Any direct
Direct cua/6 - Any direct
Direct cua/7 - Any direct
# a is the console port (aka "tip" line)
Direct cua/a - Any direct
# b is the aux port on the motherboard
Direct cua/b - Any direct
# c and d are high speed sync/async ports
Direct cua/c - Any direct
Direct cua/d - Any direct
```

이 파일은 직렬 포트 cua/b에 연결된 모든 Hayes 모뎀을 지원합니다.

## /etc/uucp/Dialers 파일의 예

508 페이지 “asppp에서 Solaris PPP 4.0으로 변환하는 방법”에 나와 있는 절차에는 다음 /etc/uucp/Dialers 파일이 사용됩니다.

```
#
# <Much information about modems supported by Oracle Solaris UUCP>

penril      =W-P      "" \d > Q\c : \d- > s\p9\c )-W\p\r\ds\p9\c-) y\c : \E\TP > 9\c OK
ventel      =&-%      "" \r\p\r\c $ k\c ONLINE!
```

```

vadic      =K-K      "" \005\p *- \005\p- * \005\p- * D\p BER? \E\T\e \r\c LINE
develcon   ""      "" \pr\ps\c est:\007 \E\D\e \n\007
micom      ""      "" \s\c NAME? \D\r\c GO
direct
#
#
#
# Hayes Smartmodem -- modem should be set with the configuration
# switches as follows:
#
#      S1 - UP      S2 - UP      S3 - DOWN      S4 - UP
#      S5 - UP      S6 - DOWN      S7 - ?      S8 - DOWN
#
hayes      =, -,      "" \dA\pTE1V1X1Q0S2=255S12=255\r\c OK\r \EATDT\T\r\c CONNECT

```

*<much more information about modems supported by Oracle Solaris UUCP>*

이 파일에는 /etc/uucp/Dialers 파일에서 지원되는 Hayes 모뎀을 비롯한 모든 유형의 모뎀에 대한 채트 스크립트가 포함되어 있습니다.

## asppp2pppd 변환 스크립트 실행(작업)

/usr/sbin/asppp2pppd 스크립트는 /etc/asppp.cf의 PPP 정보와 PPP 관련 UUCP 파일을 Solaris PPP 4.0 파일의 적절한 위치로 복사합니다.

### 작업 필수 조건

그 다음 작업을 수행하기 전에 다음 작업을 완료해야 합니다.

- asppp 및 UUCP 구성 파일도 있는 시스템에 Oracle Solaris 릴리스 설치
- PPP 파일이 있는 시스템(예: mojave 시스템)에서 슈퍼 유저 되기

## ▼ asppp에서 Solaris PPP 4.0으로 변환하는 방법

### 1 변환스크립트를 시작합니다.

```
# /usr/sbin/asppp2pppd
```

변환 프로세스가 시작되고 다음 화면 출력이 제공됩니다.

```

This script provides only a suggested translation for your existing aspppd
configuration.  You will need to evaluate for yourself whether the translation
is appropriate for your operating environment.
Continue [Yn]?

```

## 2 "Y"를 입력하여 계속합니다.

다음 출력이 제공됩니다.

```
Chat cannot do echo checking; requests for this removed.
Adding 'noauth' to /etc/ppp/options
```

Preparing to write out translated configuration:

```
1 chat file:
  1. /etc/ppp/chat.Pgobi.hayes
2 option files:
  2. /etc/ppp/peers/Pgobi
  3. /etc/ppp/options
1 script file:
  4. /etc/ppp/demand
```

새 Solaris PPP 4.0 파일이 생성되었습니다.

## ▼ 변환 결과를 보는 방법

변환 프로세스가 끝나면 /usr/sbin/asppp2pppd 변환 스크립트를 통해 만들어진 Solaris PPP 4.0 파일을 볼 수 있습니다. 이 스크립트는 다음 옵션 목록을 표시합니다.

```
Enter option number:
  1 - view contents of file on standard output
  2 - view contents of file using /usr/bin/less
  3 - edit contents of file using /usr/bin/vi
  4 - delete/undelete file from list
  5 - rename file in list
  6 - show file list again
  7 - escape to shell (or "!")
  8 - abort without saving anything
  9 - save all files and exit (default)
```

Option:

### 1 1을 입력하여 화면에서 파일 내용을 봅니다.

스크립트가 표시할 파일 수를 요청합니다.

File number (1 .. 4):

숫자는 이전의 2단계에 나와 있는 것과 같이 변환 프로세스 중 나열되는 변환된 파일의 수를 나타냅니다.

### 2 1을 입력하여 채트 파일 /etc/ppp/chat.Pgobi.hayes를 봅니다.

```
File number (1 .. 4): 1
"" \d\dA\p\pTE1V1X1Q0S2=255S12=255\r\c
OK\r ATDT\T\r\c
CONNECT \c
in:--in: mojave
word: sand
```

채트 스크립트에는 샘플 /etc/uucp/Dialers 파일의 hayes 행에 나타나는 모뎀 “채트” 정보가 포함되어 있습니다. /etc/ppp/chat.Pgobi.hayes에는 Pgobi의 로그인 절차(샘플

/etc/uucp/Systems 파일에 나타남)도 포함되어 있습니다. 채트 스크립트가 이제 /etc/ppp/chat.Pgobi.hayes 파일에 있습니다.

**3 2를 입력하여 피어 파일인 /etc/ppp/peers/Pgobi를 봅니다.**

```
File number (1 .. 4): 2
/dev/cua/b
38400
demand
idle 120
connect "/usr/bin/chat -f /etc/ppp/chat.Pgobi.hayes -T '15551212'"
user NeverAuthenticate
mojave:gobi
```

직렬 포트 정보(/dev/cua/b)는 /etc/uucp/Devices 파일에서 가져온 것입니다. 링크 속도, 유휴 시간, 인증 정보 및 피어 이름은 /etc/asppp.cf 파일에서 가져온 것입니다.

“demand”는 다이얼 아웃 시스템이 피어 Pgobi에 연결하려고 할 때 호출할 “demand” 스크립트를 나타냅니다.

**4 3을 입력하여 /etc/ppp/options 파일(다이얼 아웃 시스템 mojave에 대해 만들어짐)을 봅니다.**

```
File number (1 .. 4): 3
#lock
noauth
```

/etc/ppp/options의 정보는 /etc/asppp.cf 파일에서 가져온 것입니다.

**5 4를 입력하여 demand 스크립트의 내용을 봅니다.**

```
File number (1 .. 4): 4
/usr/bin/pppd file /etc/ppp/peers/Pgobi
```

호출 시 이 스크립트는 pppd 명령을 실행한 다음 /etc/ppp/peers/Pgobi를 읽어 mojave와 Pgobi 간의 링크를 시작합니다.

**6 9를 입력하여 만들어진 파일을 저장합니다.그런 다음 변환 스크립트를 종료합니다.**

## UUCP(개요)

---

이 장에서는 UNIX-to-UNIX Copy Program(UUCP) 및 해당 데몬에 대해 소개합니다. 다음 항목을 다룹니다.

- 511 페이지 “UUCP 하드웨어 구성”
- 512 페이지 “UUCP 소프트웨어”
- 514 페이지 “UUCP 데이터베이스 파일”

UUCP는 컴퓨터에서 파일을 전송하고 서로 메일을 교환할 수 있도록 합니다. 또한 이 프로그램을 통해 컴퓨터에서 Usenet과 같은 대규모 네트워크에 참가할 수 있습니다.

Oracle Solaris OS는 UUCP의 BNU(기본 네트워크 유틸리티) 버전을 제공합니다. 이 버전은 HoneyDanBer UUCP라고도 합니다. *UUCP*라는 용어는 시스템을 구성하는 파일 및 유틸리티의 전체 범위를 나타냅니다. *uucp*는 이 시스템에 포함되는 한 요소입니다. 컴퓨터 간에 파일을 복사하는 데 사용되는 유틸리티(*uucp* 및 *uuto*)에서 원격 로그인 및 명령 실행에 사용되는 유틸리티(*cu* 및 *uux*)에 이르기까지 다양한 UUCP 유틸리티가 있습니다.

## UUCP 하드웨어 구성

UUCP는 다음과 같은 하드웨어 구성을 지원합니다.

**직접 링크**     두 시스템의 직렬 포트 간에 RS-232 케이블을 연결하여 다른 컴퓨터로의 직접 링크를 만들 수 있습니다. 직접 링크는 두 컴퓨터가 정기적으로 통신하며 15미터 이내 등 서로 물리적으로 가까운 거리에 있는 경우 유용합니다. 제한 거리 모뎀을 사용하면 두 컴퓨터 사이의 거리를 어느 정도 늘릴 수 있습니다.

**전화선**       고속 모뎀 등의 ACU(자동 호출 단위)를 사용하면 시스템에서 표준 전화선을 통해 다른 컴퓨터와 통신할 수 있습니다. 모뎀은 UUCP에서 요청하는 전화 번호로 전화를 겁니다. 수신자 시스템에는 수신 전화를 받을 수 있는 모뎀이 있어야 합니다.

네트워크 UUCP는 TCP/IP 또는 다른 프로토콜 제품군을 실행하는 네트워크를 통해서도 통신할 수 있습니다. 네트워크에서 호스트로 설정된 컴퓨터는 네트워크에 연결된 다른 호스트에 연결할 수 있습니다.

이 장에서는 UUCP 하드웨어가 이미 어셈블 및 구성되었다고 가정합니다. 모뎀을 설정해야 하는 경우 **Oracle Solaris 관리: 일반 작업** 및 모뎀과 함께 제공된 매뉴얼을 참조하십시오.

## UUCP 소프트웨어

UUCP 소프트웨어는 Oracle Solaris 설치 프로그램을 실행하고 전체 배포를 선택하면 자동으로 포함됩니다. pkgadd를 사용하여 UUCP 소프트웨어를 추가할 수도 있습니다. UUCP 프로그램은 세 가지 범주인 데몬, 관리 프로그램 및 사용자 프로그램으로 구분할 수 있습니다.

## UUCP 데몬

UUCP 시스템에는 uucico, uuxqt, uusched 및 in.uucpd의 4개 데몬이 있습니다. 이러한 데몬은 UUCP 파일 전송 및 명령 실행을 처리합니다. 필요한 경우 셸에서 명령을 실행할 수도 있습니다.

**uucico** 링크에 사용되는 장치를 선택하고, 원격 컴퓨터에 대한 링크를 설정하고, 필요한 로그인 시퀀스 및 권한 확인을 수행합니다. 또한 uucico는 데이터 파일을, 실행 파일 및 로그로부터의 결과를 전송하고 사용자에게 전송 완료 메일을 보내 통지합니다. uucico는 UUCP 로그인 계정의 “로그인 셸”로 작동합니다. 로컬 uucico 데몬은 원격 시스템을 호출할 때 세션 중에 원격 uucico 데몬과 직접 통신합니다.

모든 필수 파일을 만들고 나면 uucp, uuto 및 uux 프로그램은 uucico 데몬을 실행하여 원격 컴퓨터에 연결합니다. uusched 및 uutry는 모두 uucico를 실행합니다. 자세한 내용은 **uucico(1M)** 매뉴얼 페이지를 참조하십시오.

**uuxqt** 원격 실행 요청을 실행합니다. 이 데몬은 스펠 디렉토리에서 원격 컴퓨터로부터 발송된 실행 파일(이름이 항상 **x.파일**로 지정됨)을 검색합니다. **x.파일** 파일이 있으면 uuxqt는 해당 파일을 열어 실행에 필요한 데이터 파일 목록을 가져옵니다. 그런 다음 uuxqt는 필수 데이터 파일을 사용 및 액세스할 수 있는지를 확인합니다. 파일을 사용할 수 있으면 uuxqt는 Permissions 파일에서 요청된 명령을 실행할 권한이 있는지 확인합니다. uuxqt 데몬은 uudemmon.hour 셸 스크립트를 통해 실행되며, 이 스크립트는 cron을 통해 시작됩니다. 자세한 내용은 **uuxqt(1M)** 매뉴얼 페이지를 참조하십시오.



- uusched** 스펠 디렉토리에서 대기열에 있는 작업을 예약합니다. **uusched**는 부트 시에 **uudemon.hour** 셸 스크립트를 통해 처음 실행되며, 이 스크립트는 **cron**을 통해 시작됩니다. 자세한 내용은 **uusched(1M)** 매뉴얼 페이지를 참조하십시오. **uucico** 데몬을 시작하기 전에 **uusched**는 원격 컴퓨터가 호출되는 순서를 무작위화합니다.
- in.uucpd** 네트워크를 통한 UUCP 연결을 지원합니다. 원격 호스트의 **inetd**는 UUCP 연결을 설정할 때마다 **in.uucpd**를 호출합니다. 그러면 **uucpd**가 로그인 이름 프롬프트를 표시합니다. 호출하는 호스트의 **uucico**는 로그인 이름으로 응답해야 합니다. 그리고 나면 **in.uucpd**가 암호 프롬프트를 표시합니다. 암호가 필요하지 않은 경우에는 제외합니다. 자세한 내용은 **in.uucpd(1M)** 매뉴얼 페이지를 참조하십시오.

## UUCP 관리 프로그램

대부분의 UUCP 관리 프로그램은 **/usr/lib/uucp**에 있습니다. 가장 기본적인 데이터베이스 파일은 **/etc/uucp**에 있습니다. 단, **uulog**는 **/usr/bin**에 있습니다. **uucp** 로그인 ID의 홈 디렉토리는 **/usr/lib/uucp**입니다. **su** 또는 **login**을 통해 관리 프로그램을 실행할 때는 **uucp** 사용자 ID를 사용합니다. 사용자 ID는 프로그램 및 스펠된 데이터 파일을 소유합니다.

- uulog** 지정된 컴퓨터의 로그 파일 콘텐츠를 표시합니다. 로그 파일은 시스템이 통신하는 각 원격 컴퓨터에 대해 만들어집니다. 로그 파일에는 각 **uucp**, **uuto** 및 **uux** 사용이 기록됩니다. 자세한 내용은 **uucp(1C)** 매뉴얼 페이지를 참조하십시오.
- uucleanup** 스펠 디렉토리를 정리합니다. **uucleanup**는 일반적으로 **uudemon.cleanup** 셸 스크립트에서 실행되며, 이 스크립트는 **cron**을 통해 시작됩니다. 자세한 내용은 **uucleanup(1M)** 매뉴얼 페이지를 참조하십시오.
- Uutry** 통화 처리 기능을 테스트하고 적절한 디버깅을 수행합니다. **Uutry**는 **uucico** 데몬을 호출하여 시스템과 지정한 원격 컴퓨터 간에 통신 링크를 설정합니다. 자세한 내용은 **Uutry(1M)** 매뉴얼 페이지를 참조하십시오.
- uuccheck** UUCP 디렉토리, 프로그램 및 지원 파일이 있는지 확인합니다. **uuccheck**는 **/etc/uucp/Permissions** 파일의 특정 부분에 명백한 구문 오류가 있는지도 확인할 수 있습니다. 자세한 내용은 **uuccheck(1M)** 매뉴얼 페이지를 참조하십시오.

## UUCP 사용자 프로그램

UUCP 사용자 프로그램은 **/usr/bin**에 있습니다. 특수한 권한이 없어도 이러한 프로그램을 사용할 수 있습니다.

cu	두 시스템에 동시 로그인할 수 있도록 원격 컴퓨터에서 시스템에 연결합니다. cu를 실행하면 두 시스템 중 하나에서 파일을 전송하거나 명령을 실행할 수 있으며 초기 링크가 삭제되지 않습니다. 자세한 내용은 <a href="#">cu(1C)</a> 매뉴얼 페이지를 참조하십시오.
uucp	시스템 간에 파일을 복사할 수 있습니다. uucp는 작업 파일과 데이터 파일을 만들고, 전송할 작업을 대기열에 넣고, uucico 데몬을 호출합니다. 이 데몬은 원격 컴퓨터 연결을 시도합니다. 자세한 내용은 <a href="#">uucp(1C)</a> 매뉴얼 페이지를 참조하십시오.
uuto	파일을 로컬 시스템에서 원격 시스템의 공개 스푼 디렉토리 /var/spool/uucppublic/receive로 복사합니다. 원격 시스템의 모든 액세스 가능 디렉토리에 파일을 복사하는 데 사용할 수 있는 uucp와는 달리, uuto는 적절한 스푼 디렉토리에 파일을 저장하고 원격 사용자에게 uupick을 사용하여 파일을 선택하라는 메시지를 표시합니다. 자세한 내용은 <a href="#">uuto(1C)</a> 매뉴얼 페이지를 참조하십시오.
uupick	uuto를 사용하여 컴퓨터로 파일을 전송할 때 /var/spool/uucppublic/receive의 파일을 검색합니다. <a href="#">uuto(1C)</a> 매뉴얼 페이지를 참조하십시오.
uux	원격 시스템에서 명령을 실행하는 데 필요한 작업, 데이터 및 실행 파일을 만듭니다. 자세한 내용은 <a href="#">uux(1C)</a> 매뉴얼 페이지를 참조하십시오.
uustat	요청된 전송(uucp, uuto 또는 uux)의 상태를 표시합니다. uustat를 통해 대기열에 있는 전송을 제어할 수도 있습니다. 자세한 내용은 <a href="#">uustat(1C)</a> 매뉴얼 페이지를 참조하십시오.

## UUCP 데이터베이스 파일

UUCP 데이터베이스를 구성하는 파일의 구성은 UUCP 설정에서 중요한 요소입니다. 이러한 파일은 /etc/uucp 디렉토리에 있습니다. 시스템에서 UUCP 또는 asppp를 설정하려면 이러한 파일을 편집해야 합니다. 이 파일에는 다음이 포함됩니다.

Config	변수 매개변수 목록이 포함되어 있습니다. 이러한 매개변수를 수동으로 설정하여 네트워크를 구성할 수 있습니다.
Devconfig	네트워크 통신을 구성하는 데 사용됩니다.
Devices	네트워크 통신을 구성하는 데 사용됩니다.
Dialcodes	Systems 파일 항목의 전화 번호 필드에서 사용할 수 있는 다이얼 코드 약어가 포함되어 있습니다. Dialcodes는 반드시 사용해야 하는 것은 아니지만 asppp와 UUCP에서 모두 사용할 수 있습니다.
Dialers	모뎀과 협상하여 원격 컴퓨터와의 연결을 설정하는 데 필요한 문자열이 포함되어 있습니다. Dialers는 asppp와 UUCP에서 모두 사용됩니다.

Grades	작업 등급 및 각 작업 등급과 연관된 권한을 정의합니다. 사용자는 원격 컴퓨터 대기열에 작업을 넣기 위해 권한을 지정할 수 있습니다.
Limits	시스템에서 허용되는 동시 <code>uucico</code> , <code>uuxqt</code> 및 <code>uusched</code> 의 최대 수를 정의합니다.
Permissions	시스템에서 명령을 실행하거나 파일을 전송하려고 시도하는 원격 호스트에 대해 부여되는 액세스 레벨을 정의합니다.
Poll	시스템에서 폴링할 시스템과 폴링 시간을 정의합니다.
Sysfiles	<code>uucico</code> 및 <code>cu</code> 에서 사용할 서로 다른 파일이나 여러 파일을 <code>Systems</code> , <code>Devices</code> 및 <code>Dialers</code> 파일에 지정합니다.
Sysname	TCP/IP 호스트 이름과 함께 시스템에 대해 고유한 UUCP 이름을 정의할 수 있습니다.
Systems	<code>uucico</code> 데몬, <code>cu</code> 및 <code>asppp</code> 에서 원격 컴퓨터에 대한 링크를 설정하는 데 필요한 정보가 포함되어 있습니다. 이 정보에는 다음이 포함됩니다. <ul style="list-style-type: none"> <li>■ 원격 호스트의 이름</li> <li>■ 원격 호스트와 연관된 연결 장치의 이름</li> <li>■ 호스트에 연결할 수 있는 시간</li> <li>■ 전화 번호</li> <li>■ 로그인 ID</li> <li>■ 암호</li> </ul>

기타 여러 파일은 지원 데이터베이스의 일부분으로 간주될 수는 있지만 링크를 설정하고 파일을 전송하는 데 직접 사용되지는 않습니다.

## UUCP 데이터베이스 파일 구성

UUCP 데이터베이스는 [514 페이지 “UUCP 데이터베이스 파일”](#)에 나와 있는 파일로 구성됩니다. 그러나 기본 UUCP 구성에는 다음과 같은 중요 파일만 포함됩니다.

- `/etc/uucp/Systems`
- `/etc/uucp/Devices`
- `/etc/uucp/Dialers`

`asppp`는 UUCP 데이터베이스 중 일부를 사용하므로 `asppp`를 구성하려면 최소한 이와 같은 중요 데이터베이스 파일에 대해 알고 있어야 합니다. 이러한 데이터베이스 파일을 구성하고 나면 UUCP 관리를 간단하게 수행할 수 있습니다. 일반적으로는 `Systems` 파일을 먼저 편집한 후에 `Devices` 파일을 편집합니다. 기본 파일에 없는 전화 걸기를 추가하려는 경우가 아니면 보통 기본 `/etc/uucp/Dialers` 파일을 사용할 수 있습니다. 또한 기본 UUCP 및 `asppp` 구성에 대해 다음 파일을 사용할 수도 있습니다.

- `/etc/uucp/Sysfiles`
- `/etc/uucp/Dialcodes`

■ /etc/uucp/Sysname

이러한 파일은 서로 긴밀하게 연관되어 함께 사용되므로 해당 콘텐츠를 모두 파악한 후에 파일을 변경해야 합니다. 특정 파일의 항목을 변경하는 경우 다른 파일의 관련 항목을 변경해야 할 수 있습니다. [514 페이지](#) “UUCP 데이터베이스 파일”에 나와 있는 나머지 파일은 크게 영향을 받지 않습니다.

---

주 - asppp는 이 절에서 설명하는 파일만 사용합니다. 즉, asppp는 다른 UCCP 데이터베이스 파일을 사용하지 않습니다.

---

## UUCP 관리(작업)

이 장에서는 시스템과 관련된 데이터베이스 파일을 수정한 후에 UUCP 작업을 시작하는 방법에 대해 설명합니다. 이 장에는 다음과 같이 Oracle Solaris OS를 실행하는 시스템에서 UUCP를 설정 및 유지 관리하기 위한 절차 및 문제 해결 정보가 포함되어 있습니다.

- 517 페이지 “UUCP 관리(작업 맵)”
- 518 페이지 “UUCP 로그인 추가”
- 519 페이지 “UUCP 시작”
- 521 페이지 “TCP/IP를 통해 UUCP 실행”
- 522 페이지 “UUCP 보안 및 유지 관리”
- 523 페이지 “UUCP 문제 해결”

## UUCP 관리(작업 맵)

다음 표에서는 이 장에서 다루는 절차에 대한 포인터와 각 절차의 간단한 설명이 나와 있습니다.

표 25-1 UUCP 관리용 작업 맵

작업	설명	수행 방법
원격 시스템이 시스템에 액세스하도록 허용	시스템 액세스가 허용되는 시스템을 식별하기 위한 항목을 추가하려면 <code>/etc/passwd</code> 파일을 편집합니다.	518 페이지 “UUCP 로그인을 추가하는 방법”
UUCP 시작	제공된 셸 스크립트를 사용하여 UUCP를 시작합니다.	519 페이지 “UUCP를 시작하는 방법”
TCP/IP와 함께 작동하도록 UUCP 설정	TCP/IP에 대해 UUCP를 활성화하려면 <code>/etc/inetd.conf</code> 및 <code>/etc/uucp/Systems</code> 파일을 편집합니다.	521 페이지 “TCP/IP에 대해 UUCP를 활성화하는 방법”
몇 가지 일반적인 UUCP 문제 해결	진단 단계를 수행하여 고장난 모델 또는 ACU를 확인합니다.	523 페이지 “고장난 모델이나 ACU를 확인하는 방법”

표 25-1 UUCP 관리용 작업 맵 (계속)

작업	설명	수행 방법
	진단 단계를 수행하여 전송을 디버그합니다.	524 페이지 “전송을 디버그하는 방법”

## UUCP 로그인 추가

원격 시스템에서 받는 UUCP(uucico) 요청을 올바르게 처리하려면 시스템의 각 시스템에 로그인이 있어야 합니다.

### ▼ UUCP 로그인을 추가하는 방법

원격 시스템의 시스템 액세스를 허용하려면 다음과 같이 `/etc/passwd` 파일에 대한 항목을 추가해야 합니다.

#### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

#### 2 `/etc/passwd` 파일을 편집하여 시스템 액세스가 허용되는 시스템을 식별하기 위한 항목을 추가합니다.

UUCP 연결을 통해 시스템에 액세스할 수 있도록 허용되는 원격 시스템에 대한 `/etc/passwd` 파일에는 일반적으로 다음 항목을 추가합니다.

```
Ugobi:*:5:5:gobi:/var/spool/uucppublic:/usr/lib/uucp/uucico
```

원격 시스템의 로그인 이름은 대문자 `U` 뒤에 시스템 이름이 오는 형식입니다. 이름은 8자 이내여야 합니다. 그렇지 않은 경우에는 이름을 자르거나 축약해야 합니다.

위 항목은 Ugobi의 로그인 요청에 대해 `/usr/lib/uucp/uucico`로 대답함을 보여줍니다. 홈 디렉토리는 `/var/spool/uucppublic`입니다. 암호는 `/etc/shadow` 파일에서 가져옵니다. 원격 시스템의 UUCP 관리자와 함께 암호와 로그인 이름을 조정해야 합니다. 원격 관리자는 원격 시스템의 `Systems` 파일에 로그인 이름과 암호화되지 않은 암호가 포함된 적절한 항목을 추가해야 합니다.

#### 3 시스템 이름은 다른 시스템의 UUCP 관리자와 함께 조정합니다.

마찬가지로, UUCP를 통해 연결하려는 모든 컴퓨터의 UUCP 관리자와 함께 시스템 이름과 암호를 조정해야 합니다.

## UUCP 시작

UUCP에는 원격 시스템을 폴링하고, 전송을 다시 예약하고, 오래된 로그 파일과 실패한 전송을 정리하는 4개의 셸 스크립트가 포함되어 있습니다. 이러한 스크립트는 다음과 같습니다.

- uudemmon.poll
- uudemmon.hour
- uudemmon.admin
- uudemmon.cleanup

이러한 셸 스크립트를 정기적으로 실행하여 UUCP가 원활하게 실행되는지 확인해야 합니다. 스크립트를 실행할 crontab 파일은 전체 설치를 선택하는 경우 Oracle Solaris 설치 프로세스의 일부분으로 /usr/lib/uucp/uudemmon.crontab에서 자동으로 만들어집니다. 그렇지 않으면 UUCP 패키지를 설치할 때 파일이 만들어집니다.

UUCP 셸 스크립트를 수동으로 실행할 수도 있습니다. 특정 시스템용으로 조정할 수 있는 프로토타입 uudemmon.crontab 파일은 다음과 같습니다.

```
#
#ident "@(#)uudemmon.crontab 1.5 97/12/09 SMI"
#
# This crontab is provided as a sample. For systems
# running UUCP edit the time schedule to suit, uncomment
# the following lines, and use crontab(1) to activate the
# new schedule.
#
#48 8,12,16 * * * /usr/lib/uucp/uudemmon.admin
#20 3 * * * /usr/lib/uucp/uudemmon.cleanup
#0 * * * * /usr/lib/uucp/uudemmon.poll
#11,41 * * * * /usr/lib/uucp/uudemmon.hour
```

주 - 기본적으로 UUCP 작업은 사용 안함으로 설정됩니다. UUCP를 사용으로 설정하려면 시간 일정을 편집하고 uudemmon.crontab 파일에서 해당 행의 주석 처리를 해제합니다.

### ▼ UUCP를 시작하는 방법

uudemmon.crontab 파일을 활성화하려면 다음을 수행합니다.

#### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

#### 2 /usr/lib/uucp/uudemmon.crontab 파일을 편집하고 필요한 대로 항목을 변경합니다.

#### 3 다음 명령을 실행하여 uudemmon.crontab 파일을 활성화합니다.

```
crontab < /usr/lib/uucp/uudemmon.crontab
```

## uudemon.poll 셀 스크립트

기본 `uudemon.poll` 셀 스크립트는 `/etc/uucp/Poll` 파일을 1시간마다 읽습니다. Poll 파일의 시스템이 폴링되도록 예약되어 있으면 작업 파일(C. `synxxxx`)이 `/var/spool/uucp/nodename` 디렉토리에 저장됩니다. `nodename`은 시스템의 UUCP 노드 이름을 나타냅니다.

셀 스크립트는 `uudemon.hour` 전에 1시간마다 실행되도록 예약되므로 `uudemon.hour`를 호출하면 작업 파일이 해당 위치에 배치됩니다.

## uudemon.hour 셀 스크립트

기본 `uudemon.hour` 셀 스크립트는 다음을 수행합니다.

- `uusched` 프로그램을 호출하여 처리되지 않은 작업 파일(C.)에 대한 스푼 디렉토리를 검색합니다. 그런 다음 스크립트는 이러한 파일을 원격 파일로 전송하도록 예약합니다.
- `uuxqt` 데몬을 호출하여 컴퓨터로 전송되었으며 전송 시 처리되지 않은 실행 파일(X.)에 대한 스푼 디렉토리를 검색합니다.

기본적으로 `uudemon.hour`는 시간당 두 번 실행됩니다. 원격 시스템에 대한 호출 오류율이 높을 것으로 예상되면 `uudemon.hour` 실행 빈도를 높일 수 있습니다.

## uudemon.admin 셀 스크립트

기본 `uudemon.admin` 셀 스크립트는 다음을 수행합니다.

- `p` 및 `q` 옵션을 포함하여 `uustat` 명령을 실행합니다. `q`는 대기열에 있는 작업 파일(C.), 데이터 파일(D.) 및 실행 파일(X.)의 상태를 보고합니다. `p`는 잠금 파일(`/var/spool/locks`)에 나열된 네트워킹 프로세스에 대한 프로세스 정보를 인쇄합니다.
- `mail` 명령을 사용하여 결과 상태 정보를 `uucp` 관리 로그로 보냅니다.

## uudemon.cleanup 셀 스크립트

기본 `uudemon.cleanup` 셀 스크립트는 다음을 수행합니다.

- `/var/uucp/.Log` 디렉토리에서 개별 시스템의 로그 파일을 수집하여 병합한 다음 기타 오래된 로그 정보와 함께 `/var/uucp/.Old` 디렉토리에 저장합니다.
- 7일 이상 된 작업 파일(C.), 7일 이상 된 데이터 파일(D.) 및 2일 이상 된 실행 파일(X.)을 스푼 파일에서 제거합니다.
- 보낸 사람에게 배달할 수 없는 메일을 반송합니다.



- 현재 날짜 동안 수집된 상태 정보의 요약을 UUCP 관리 로그인(uucp)에게 메일로 보냅니다.

## TCP/IP를 통해 UUCP 실행

TCP/IP 네트워크에서 UUCP를 실행하려면 이 절에서 설명하는 것처럼 몇 가지 사항을 수정해야 합니다.

### ▼ TCP/IP에 대해 UUCP를 활성화하는 방법

#### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

#### 2 `/etc/uucp/Systems` 파일을 편집하여 항목에 다음 필드가 포함되도록 합니다.

*System-Name Time TCP Port networkname Standard-Login-Chat*

일반적인 항목은 다음과 같습니다.

```
rochester Any TCP - ur-seneca login: Umachine password: xxx
```

*networkname* 필드는 TCP/IP 호스트 이름을 명시적으로 지정할 수 있도록 합니다. 일부 사이트에서는 이 기능이 중요합니다. 이전 예에서 사이트에는 UUCP 노드 이름 *rochester*가 있습니다. 이 이름은 해당 TCP/IP 호스트 이름 *ur-seneca*와는 다릅니다. 또한 완전히 다른 시스템에서 UUCP를 쉽게 실행하고 TCP/IP 호스트 이름 *rochester*를 소유할 수 있습니다.

`Systems` 파일의 `Port` 필드에는 - 항목이 있어야 합니다. 이 구문은 항목을 `uucp`로 나열하는 것에 해당합니다. 거의 모든 상황에서 *networkname*은 시스템 이름과 같고 `Port` 필드는 -입니다(즉, `services` 데이터베이스에서 표준 `uucp` 포트 사용). `in.uucpd` 데몬은 원격 시스템이 해당 로그인 및 암호를 인증용으로 보낸다고 가정하며, `in.uucpd`는 `getty` 및 `login`과 비슷하게 로그인 및 암호 프롬프트를 표시합니다.

#### 3 `/etc/inet/services` 파일을 편집하여 UUCP용 포트를 설정합니다.

```
uucp 540/tcp uucpd # uucp daemon
```

항목은 변경할 필요가 없습니다. 그러나 시스템에서 이름 서비스로 NIS를 실행하는 경우에는 `svc:/system/name-service/switch` 서비스의 `config/service`가 `nis` 전에 `files`를 확인하도록 해야 합니다. `config/service` 등록 정보가 정의되어 있지 않으면 `config/default` 등록 정보를 확인합니다.

#### 4 UUCP가 사용으로 설정되어 있는지 확인합니다.

```
# svcs network/uucp
```

UUCP 서비스는 서비스 관리 기능을 통해 관리됩니다. 이 서비스의 상태를 질의하려면 `svcs` 명령을 사용하면 됩니다. 서비스 관리 기능의 개요는 [Oracle Solaris 관리: 일반 작업의 6 장, “서비스 관리\(개요\)”](#)를 참조하십시오.

- 5 (옵션) 필요한 경우 다음을 입력하여 UUCP를 사용으로 설정합니다.

```
# inetadm -e network/uucp
```

## UUCP 보안 및 유지 관리

UUCP를 설정한 후에는 유지 관리를 쉽게 수행할 수 있습니다. 이 절에서는 보안, 유지 관리 및 문제 해결과 관련된 진행 중인 UUCP 작업에 대해 설명합니다.

### UUCP 보안 설정

기본 `/etc/uucp/Permissions` 파일은 UUCP 링크에 대한 최대 보안 레벨을 제공합니다. 기본 `Permissions` 파일에는 항목이 없습니다.

각 원격 시스템에 대해 추가 매개변수를 설정하여 다음을 정의할 수 있습니다.

- 원격 시스템에서 사용자 시스템의 파일을 받을 수 있는 방식
- 원격 시스템에 읽기 및 쓰기 권한이 있는 디렉토리
- 원격 시스템에서 원격 실행에 사용할 수 있는 명령

일반적인 `Permissions` 항목은 다음과 같습니다.

```
MACHINE=datsun LOGNAME=Udatsun VALIDATE=datsun
COMMANDS=rmail REQUEST=yes SENDFILES=yes
```

이 항목은 시스템의 모든 위치가 아닌 "일반" UUCP 디렉토리로 파일을 보내고 해당 디렉토리에서 파일을 받을 수 있도록 합니다. 또한 이 항목은 로그인 시 UUCP 사용자 이름을 검증하도록 합니다.

### 정기 UUCP 유지 관리

UUCP에서는 많은 유지 관리 작업을 수행하지 않아도 됩니다. 그러나 [519 페이지 “UUCP를 시작하는 방법”](#) 절에 설명된 대로 `crontab` 파일이 있는지는 확인해야 합니다. 구체적으로는 메일 파일 및 공개 디렉토리의 확장을 파악해야 합니다.

### UUCP에 대해 전자 메일 보내기

UUCP 프로그램과 스크립트에 의해 생성되는 모든 전자 메일 메시지는 사용자 ID `uucp`에게 발송됩니다. 해당 사용자만큼 자주 로그인하지 않는 경우에는 메일이 누적되어 디스크 공간을 많이 사용하는 것을 모를 수도 있습니다. 이 문제를 해결하려면

/etc/mail/aliases에 별칭을 만들고 전자 메일을 root 또는 UUCP 유지 관리 담당자(자신이나 다른 사람)에게로 재지정합니다. aliases 파일을 수정한 후에는 newaliases 명령을 실행해야 합니다.

## UUCP 공개 디렉토리

/var/spool/uucppublic 디렉토리는 UUCP에서 기본적으로 파일을 복사할 수 있는 모든 시스템의 위치 중 하나입니다. 모든 사용자는 해당 디렉토리에서 /var/spool/uucppublic을 변경하는 권한과 파일을 읽고 쓰는 권한을 가집니다. 그러나 디렉토리에 고정 비트가 설정되어 있으므로 디렉토리 모드는 01777입니다. 따라서 사용자는 해당 디렉토리로 복사한 파일(uucp에 속함)을 제거할 수 없습니다. root 또는 uucp로 로그인한 UUCP 관리자만이 이 디렉토리에서 파일을 제거할 수 있습니다. 이 디렉토리에서 파일이 통제할 수 없을 정도로 누적되지 않도록 하려면 디렉토리의 파일을 정기적으로 제거해야 합니다.

사용자가 이러한 유지 관리 작업을 수행하기 어려운 경우에는 보안상의 이유로 설정되는 고정 비트를 제거하는 대신 uuto 및 uupick을 사용하도록 합니다. uuto 및 uupick 사용 지침은 uuto(1C) 매뉴얼 페이지를 참조하십시오. 디렉토리의 모드를 단일 사용자 그룹으로만 제한할 수도 있습니다. 특정 사용자가 디스크를 모두 사용하지 않도록 하려면 UUCP의 디스크 액세스를 거부할 수도 있습니다.

## UUCP 문제 해결

이 절차에서는 일반적인 UUCP 문제를 해결하는 방법에 대해 설명합니다.

### ▼ 고장난 모뎀이나 ACU를 확인하는 방법

여러 가지 방법으로 모뎀이나 기타 ACU가 정상적으로 작동하지 않는지를 확인할 수 있습니다.

#### 1 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”**을 참조하십시오.

#### 2 다음 명령을 실행하여 연결 실패의 횟수와 이유를 표시합니다.

```
# uustat -q
```

#### 3 특정 전화선을 호출하여 해당 호출 시도에 대한 디버깅 정보를 인쇄합니다.

해당 행은 /etc/uucp/Devices 파일에서 direct로 정의해야 합니다. 전화선이 자동 전화 걸기에 연결되거나 장치를 direct로 설정해야 하는 경우에는 명령줄 끝에 전화 번호를 추가해야 합니다. 다음을 입력합니다.

```
# cu -d -lline
```

*line*은 /dev/cua/a입니다.

## ▼ 전송을 디버그하는 방법

특정 시스템에 연결할 수 없는 경우에는 Uutry 및 uucp를 사용하여 해당 시스템에 대한 통신을 확인할 수 있습니다.

### 1 관리자가 됩니다.

자세한 내용은 **Oracle Solaris 관리: 보안 서비스**의 “관리 권한을 얻는 방법”을 참조하십시오.

### 2 연결을 시도합니다.

```
# /usr/lib/uucp/Uutry -r machine
```

*machine*은 연결할 수 없는 시스템의 호스트 이름으로 대체합니다. 이 명령은 다음을 수행합니다.

- 디버깅과 함께 전송 데몬(uucico)을 시작합니다. 작업을 수행하는 사용자가 root이면 추가 디버깅 정보를 확인할 수 있습니다.
- 디버깅 출력을 /tmp/*machine*으로 보냅니다.
- 다음 명령을 실행하여 디버깅 출력을 터미널로 출력합니다.

```
# tail -f
```

출력을 종료하려면 Ctrl+C를 누릅니다. 출력을 저장하려면 /tmp/*machine*에서 출력을 복사할 수 있습니다.

### 3 Uutry를 실행해도 문제를 확인할 수 없으면 작업을 대기열에 추가해 보십시오.

```
# uucp -r file machine\!dir/file
```

*file*          전송할 파일의 이름을 사용합니다.

*machine*      복사대상 시스템의 이름을 사용합니다.

*/dir/file*    다른 시스템에 대해 파일 위치를 지정합니다.

### 4 다음 명령을 실행합니다.

```
# Uutry
```

그래도 문제를 해결할 수 없으면 해당 지역의 지원 담당자에게 문의해야 할 수 있습니다. 문제 진단에 도움이 되는 디버깅 출력을 저장합니다.

주 - `-xn` 옵션을 통해 `Uutry`에서 제공하는 디버그 레벨을 높이거나 낮출 수도 있습니다. `n`이 디버그 레벨을 나타냅니다. `Uutry`의 기본 디버그 레벨은 5입니다.

디버그 레벨 3에서는 연결 설정 시간 및 방법에 대한 기본적인 정보를 제공하지만, 전송에 대한 많은 정보가 제공되지는 않습니다. 그러나 디버그 레벨 9에서는 전송 프로세스에 대한 모든 정보가 제공됩니다. 디버깅은 전송의 양쪽 끝에서 모두 수행됩니다. 비교적 많은 양의 텍스트에 대해 5보다 높은 레벨을 사용하려는 경우에는 다른 사이트 관리자와 논의하여 레벨 변경 여부를 결정하십시오.

## UUCP /etc/uucp/Systems 파일 확인

특정 시스템에 연결하는 데 문제가 있으면 `Systems` 파일에 최신 정보가 포함되어 있는지 확인합니다. 시스템에 대해 오래되었을 수 있는 일부 정보는 다음과 같습니다.

- 전화 번호
- 로그인 ID
- Password

## UUCP 오류 메시지 확인

UUCP의 오류 메시지는 `ASSERT` 및 `STATUS`의 두 가지 유형입니다.

- 프로세스가 중지되면 `ASSERT` 오류 메시지가 `/var/uucp/.Admin/errors`에 기록됩니다. 이러한 메시지에는 파일 이름(`sccsid`), 행 번호 및 텍스트가 포함됩니다. 보통 시스템 문제가 발생하면 이러한 메시지가 생성됩니다.
- `STATUS` 오류 메시지는 `/var/uucp/.Status` 디렉토리에 저장됩니다. 이 디렉토리에는 컴퓨터에서 통신을 시도하는 각 원격 컴퓨터에 대한 별도의 파일이 포함되어 있습니다. 이러한 파일에는 시도한 통신에 대한 상태 정보와 통신 성공 여부가 포함되어 있습니다.

## 기본 정보 확인

여러 명령을 사용하여 기본 네트워킹 정보를 확인할 수 있습니다.

- `uname` 명령을 사용하여 시스템이 연결할 수 있는 시스템을 나열합니다.
- `uulog` 명령을 사용하여 특정 호스트에 대한 로그 디렉토리의 콘텐츠를 표시합니다.
- `uuccheck -v` 명령을 사용하여 `uucp`에 필요한 파일 및 디렉토리가 있는지 확인합니다. 또한 이 명령은 `Permissions` 파일을 확인하여 사용자가 설정한 권한에 대한 정보를 표시합니다.



## UUCP(참조)

---

이 장에서는 UUCP 작업에 대한 참조 정보를 제공합니다. 다음 항목을 다룹니다.

- 527 페이지 “UUCP /etc/uucp/Systems 파일”
- 534 페이지 “UUCP /etc/uucp/Devices 파일”
- 540 페이지 “UUCP /etc/uucp/Dialers 파일”
- 544 페이지 “다른 기본 UUCP 구성 파일”
- 546 페이지 “UUCP /etc/uucp/Permissions 파일”
- 554 페이지 “UUCP /etc/uucp/Poll 파일”
- 554 페이지 “UUCP /etc/uucp/Config 파일”
- 555 페이지 “UUCP /etc/uucp/Grades 파일”
- 557 페이지 “기타 UUCP 구성 파일”
- 558 페이지 “UUCP 관리 파일”
- 560 페이지 “UUCP 오류 메시지”

### UUCP /etc/uucp/Systems 파일

/etc/uucp/Systems 파일에는 uucico 데몬이 원격 컴퓨터에 대한 통신 링크를 설정하는데 필요한 정보가 포함됩니다. /etc/uucp/Systems는 UUCP를 구성할 때 첫번째로 편집해야 하는 파일입니다.

Systems 파일의 각 항목은 호스트가 통신하는 원격 컴퓨터를 나타냅니다. 특정 호스트의 항목이 둘 이상일 수 있습니다. 추가 항목은 대체 통신 경로를 나타내며 순차적으로 시도됩니다. 또한 기본적으로 UUCP에서는 /etc/uucp/Systems에 표시되지 않은 컴퓨터가 호스트에 로그인하지 못하게 합니다.

Sysfiles 파일을 사용하면 Systems 파일로 사용할 여러 파일을 정의할 수 있습니다. Sysfiles에 대한 설명은 [545 페이지 “UUCP /etc/uucp/Sysfiles 파일”](#)을 참조하십시오.

Systems 파일에 있는 항목의 구문은 다음과 같습니다.

System-Name	Time	Type	Speed	Phone	Chat Script
-------------	------	------	-------	-------	-------------

Systems 파일의 항목에 대한 다음 예를 참조하십시오.

예 26-1 /etc/uucp/Systems의 항목

```
Arabian      Any  ACUEC 38400 111222  ogin: Puucp ssword:beledi
```

Arabian	System-Name 필드의 항목입니다. 자세한 내용은 528 페이지 “/etc/uucp/Systems 파일의 System-Name 필드”를 참조하십시오.
Any	Time 필드의 항목입니다. 자세한 내용은 528 페이지 “/etc/uucp/Systems 파일의 Time 필드”를 참조하십시오.
ACUEC	Type 필드의 항목입니다. 자세한 내용은 529 페이지 “/etc/uucp/Systems 파일의 Type 필드”를 참조하십시오.
38400	Speed 필드의 항목입니다. 자세한 내용은 530 페이지 “/etc/uucp/Systems 파일의 Speed 필드”를 참조하십시오.
111222	Phone 필드의 항목입니다. 자세한 내용은 530 페이지 “/etc/uucp/Systems 파일의 Phone 필드”를 참조하십시오.
ogin: Puucp ssword:beledi	Chat Script 필드의 항목입니다. 자세한 내용은 531 페이지 “/etc/uucp/Systems 파일의 Chat-Script 필드”를 참조하십시오.

## /etc/uucp/Systems 파일의 System-Name 필드

이 필드에는 원격 컴퓨터의 노드 이름이 포함됩니다. TCP/IP 네트워크에서 이 이름은 시스템의 호스트 이름이거나 /etc/uucp/Sysname 파일을 통해 UUCP 통신용으로 특별히 만든 이름일 수 있습니다. 527 페이지 “UUCP /etc/uucp/Systems 파일”을 참조하십시오. 예 26-1에서는 System-Name 필드에 원격 호스트 Arabian에 대한 항목이 포함됩니다.

## /etc/uucp/Systems 파일의 Time 필드

이 필드는 원격 컴퓨터를 호출할 수 있는 요일 및 시간을 지정합니다. Time 필드의 형식은 다음과 같습니다.

```
daytime[;retry]
```



## Time 필드의 day 부분

*day* 부분은 다음 항목 중 일부를 포함하는 목록일 수 있습니다.

Su Mo Tu We Th Fr Sa

개별 요일입니다.

Wk

평일인 경우입니다.

Any

요일을 지정하지 않습니다.

Never

호스트에서 원격 컴퓨터에 대한 호출을 초기화하지 않습니다. 원격 컴퓨터에서 호출을 초기화해야 합니다. 그러면 호스트가 수동 모드로 작동합니다.

## Time 필드의 time 부분

예 26-1에서는 Time 필드에 Any가 표시되어 있습니다. 이는 Arabian 호스트를 언제든지 호출할 수 있음을 나타냅니다.

*time* 부분은 24시간 표기법으로 지정된 시간 범위(예: 오전 8시 30분에서 오후 12시 30분까지인 경우 0800-1230)여야 합니다. *time* 부분을 지정하지 않으면 언제든지 호출할 수 있는 것으로 간주됩니다.

0000을 포함하는 시간 범위를 사용할 수 있습니다. 예를 들어 0800-0600은 오전 6시에서 오전 8시 사이의 시간을 제외한 모든 시간이 허용됨을 의미합니다.

## Time 필드의 retry 부분

*retry* 하위 필드를 사용하여 실패한 시도 이후 재시도 전까지의 최소 시간(분)을 지정할 수 있습니다. 기본 대기는 60분입니다. 하위 필드 구분자는 세미콜론입니다. 예를 들어 Any;9는 언제든지 호출하지만 실패가 발생한 후 9분 이상 기다린 다음 다시 시도하는 것으로 해석됩니다.

*retry* 항목을 지정하지 않으면 지수 백오프 알고리즘이 사용됩니다. 이는 UUCP가 기본 대기 시간으로 시작하고 실패한 시도 수가 늘어남에 따라 대기 시간이 증가함을 의미합니다. 예를 들어 초기 재시도 시간이 5분이라고 가정합니다. 응답이 발생하지 않으면 다음 재시도는 10분 후입니다. 그리고 다음 재시도는 20분 후가 되는 식으로 최대 재시도 시간인 23시간에 도달할 때까지 계속됩니다. *retry*를 지정하는 경우 지정된 값이 항상 재시도 시작이 됩니다. 그렇지 않으면 백오프 알고리즘이 사용됩니다.

## /etc/uucp/Systems 파일의 Type 필드

이 필드에는 원격 컴퓨터에 대한 통신 링크를 설정하는 데 사용해야 하는 장치 유형이 포함됩니다. 이 필드에 사용되는 키워드는 Devices 파일 항목의 첫 번째 필드와 일치합니다.

예 26-2 Type 필드의 키워드

Arabian Any ACUEC, g 38400 1112222 ogin: Puucp ssword:beledi

Type 필드에 프로토콜을 추가하여 시스템에 연결하는 데 사용되는 프로토콜을 정의할 수 있습니다. 앞의 예에서는 g 프로토콜을 장치 유형 ACUEC에 연결하는 방법을 보여줍니다. 프로토콜에 대한 자세한 내용은 [539 페이지 “/etc/uucp/Devices 파일의 프로토콜 정의”](#)를 참조하십시오.

## /etc/uucp/Systems 파일의 Speed 필드

이 필드는 Class 필드라고도 하며 통신 링크를 설정하는 데 사용되는 장치의 전송 속도를 지정합니다. UUCP 속도 필드에는 전화 걸기 클래스를 구분하는 문자와 숫자(예: C1200 또는 D1200)가 포함됩니다. [536 페이지 “/etc/uucp/Devices 파일의 Class 필드”](#)를 참조하십시오.

일부 장치는 모든 속도로 사용할 수 있으므로 Any 키워드를 사용할 수 있습니다. 이 필드는 관련 Devices 파일 항목의 Class 필드와 일치해야 합니다.

예 26-3 Speed 필드의 항목

eagle Any ACU, g D1200 NY3251 ogin: nuucp ssword:Oakgrass

이 필드의 정보가 필요하지 않은 경우 대시(-)를 필드의 개체 틀로 사용합니다.

## /etc/uucp/Systems 파일의 Phone 필드

이 필드에서는 포트 선택기라고 하는 자동 전화 걸기용 원격 컴퓨터의 전화 번호(토큰이라고 함)를 지정할 수 있습니다. 전화 번호는 선택적 영문자 약어와 숫자 부분으로 구성됩니다. 약어를 사용하는 경우 약어가 Dialcodes 파일에 나열되어 있어야 합니다.

예 26-4 Phone 필드의 항목

nubian Any ACU 2400 NY555-1212 ogin: Puucp ssword:Passuan  
eagle Any ACU, g D1200 NY=3251 ogin: nuucp ssword:Oakgrass

Phone 필드에서 등호(=)는 ACU가 2차 발신음이 들릴 때까지 대기한 다음 나머지 숫자를 걸도록 합니다. 문자열의 대시(-)는 ACU가 4초 일시 중지한 후 다음 숫자를 걸도록 합니다.

컴퓨터가 포트 선택기에 연결되어 있으면 해당 선택기에 연결된 다른 컴퓨터에 액세스할 수 있습니다. 이러한 원격 컴퓨터의 Systems 파일 항목에서는 Phone 필드에 전화 번호가 없어야 합니다. 대신 이 필드에는 스위치에 전달되는 토큰이 포함되어

있어야 합니다. 따라서 포트 선택기는 호스트에서 통신할 원격 컴퓨터를 일반적으로 시스템 이름만 압니다. 연결된 Devices 파일 항목에는 끝에 \D가 있어야 합니다. 그래야만 이 필드가 Dialcodes 파일을 사용하여 해석되지 않습니다.

## /etc/uucp/Systems 파일의 Chat-Script 필드

Login 필드라고도 하는 이 필드에는 **채트 스크립트**라는 문자열이 포함됩니다. 채트 스크립트에는 로컬 및 원격 컴퓨터가 초기 대화에서 서로에게 전달해야 하는 문자가 포함됩니다. 채트 스크립트의 형식은 다음과 같습니다.

*expect send [expect send] ....*

*expect*는 로컬 호스트가 대화를 시작하기 위해 원격 호스트에서 받아야 하는 문자열을 나타냅니다. *send*는 로컬 호스트가 원격 호스트에서 *expect* 문자열을 받은 후 보내는 문자열입니다. 채트 스크립트에는 *expect-send* 시퀀스가 둘 이상 있을 수 있습니다.

기본 채트 스크립트에는 다음이 포함될 수 있습니다.

- 로컬 호스트가 원격 컴퓨터에서 받아야 하는 로그인 프롬프트
- 로컬 호스트가 로그인하기 위해 원격 컴퓨터에 보내는 로그인 이름
- 로컬 호스트가 원격 컴퓨터에서 받아야 하는 암호 프롬프트
- 로컬 호스트가 원격 컴퓨터에 보내는 암호

*expect* 필드는 다음 형식의 하위 필드로 구성될 수 있습니다.

*expect[-send-expect]...*

*-send*는 이전 *expect*를 읽지 못한 경우에 전송됩니다. *-send* 다음에 오는 *-expect*는 필요한 다음 문자열입니다.

예를 들어 *login--login* 문자열을 사용하면 로컬 호스트의 UUCP에 *login*이 필요합니다. UUCP가 원격 컴퓨터에서 *login*을 받으면 UUCP는 다음 필드로 이동합니다. UUCP가 *login*을 받지 못하면 UUCP는 캐리지 리턴을 보낸 다음 *login*을 다시 찾습니다. 처음에 로컬 컴퓨터에 아무 문자도 필요하지 않은 경우 *expect* 필드에서 널 문자열에 대해 ""을 사용합니다. *send* 문자열을 \c로 종결하지 않은 경우 모든 *send* 필드는 캐리지 리턴이 첨부되어 전송됩니다.

다음은 *expect-send* 문자열을 사용하는 Systems 파일 항목의 예입니다.

```
sonora Any ACUEC 9600 2223333 "" \r \r ogin:-BREAK-ogin: Puucpx ssword:xyzyz
```

이 예에서는 로컬 호스트의 UUCP가 두 개의 캐리지 리턴을 보내고 *ogin:*(*Login:*에 해당)을 대기하도록 합니다. *ogin:*을 받지 못하면 *BREAK*를 보냅니다. *ogin:*을 받으면 로그인 이름 *Puucpx*를 보냅니다. *ssword:*(*Password:*에 해당)를 받으면 암호 *xyzyz*를 보냅니다.

다음 표에서는 몇 가지 유용한 제어 문자가 나와 있습니다.

표 26-1 Systems 파일의 Chat-Script 필드에서 사용되는 제어 문자

제어 문자	의미
\b	백스페이스 문자를 보내거나 이 문자가 필요합니다.
\c	문자열의 끝에 있는 경우 정상적으로 보낸 캐리지 리턴을 표시하지 않습니다. 다른 경우에는 무시됩니다.
\d	추가 문자를 보내기 전에 1-3초 지연합니다.
\E	에코 검사를 시작합니다. 이 때부터 문자가 전송될 때마다 UUCP는 문자를 받을 때까지 기다린 다음 검사를 계속합니다.
\e	검사 해제를 에코 설정합니다.
\H	행업 하나를 무시합니다. 다이얼 백 모뎀의 경우 이 옵션을 사용합니다.
\K	BREAK 문자를 보냅니다.
\M	CLOCAL 플래그를 켭니다.
\m	CLOCAL 플래그를 끕니다.
\n	개행 문자를 보내거나 이 문자가 필요합니다.
\N	널 문자(ASCII NUL)를 보냅니다.
\p	약 0.25-0.5초 동안 일시 중지합니다.
\r	캐리지 리턴을 보내거나 캐리지 리턴이 필요합니다.
\s	공백 문자를 보내거나 이 문자가 필요합니다.
\t	탭 문자를 보내거나 이 문자가 필요합니다.
EOT	개행 두 번 다음에 EOT를 보냅니다.
BREAK	BREAK 문자를 보냅니다.
\ddd	8진수로 표시되는 문자(ddd)를 보내거나 이 문자가 필요합니다.

## 채트 스크립트를 통해 다이얼 백을 사용으로 설정

일부 회사에서는 원격 컴퓨터에서의 호출을 처리하는 다이얼 인 서버를 설정합니다. 예를 들어 회사에 다이얼 백 모뎀이 포함된 다이얼 인 서버가 있고 직원이 집에 있는 컴퓨터에서 이 서버를 호출할 수 있습니다. 다이얼 인 서버가 원격 컴퓨터를 식별한 후 다이얼 인 서버는 원격 컴퓨터와의 연결을 끊은 다음 원격 컴퓨터를 다시 호출합니다. 그런 다음 통신 링크가 다시 설정됩니다.

**Systems** 파일 채트 스크립트에서 다이얼 백이 있어야 하는 위치에 \H 옵션을 사용하여 다이얼 백을 용이하게 할 수 있습니다. 다이얼 인 서버가 행업해야 하는 위치에서 예상 문자열의 일부로 \H를 포함합니다.

예를 들어 다이얼 인 서버를 호출하는 채트 스크립트에 다음 문자열이 포함되어 있다고 가정합니다.

```
INITIATED\Hogin:
```

로컬 컴퓨터의 UUCP 전화 걸기 기능은 다이얼 인 서버에서 **INITIATED** 문자를 받아야 합니다. **INITIATED** 문자가 일치된 후 전화 걸기 기능은 다이얼 인 서버가 행업할 때까지 전화 걸기 기능이 받는 모든 후속 문자를 비웁니다. 그런 다음 로컬 전화 걸기 기능은 **expect** 문자열의 다음 부분인 **ogin:**을 다이얼 인 서버에서 받을 때까지 기다립니다. **ogin:**을 받으면 전화 걸기 기능은 채트 스크립트를 계속합니다.

앞의 샘플 문자열에 나온 것처럼 문자열이 \H 바로 앞이나 뒤에 오지 않아도 됩니다.

## /etc/uucp/Systems 파일의 하드웨어 흐름 제어

**pseudo-send STTY=** 값 문자열을 사용하여 모뎀 특성을 설정할 수도 있습니다. 예를 들어 **STTY=crtcts**는 하드웨어 흐름 제어를 사용으로 설정합니다. **STTY**에는 모든 **stty** 모드를 사용할 수 있습니다. 자세한 내용은 **stty(1)** 및 **termio(7I)** man 페이지를 참조하십시오.

다음 예에서는 **Systems** 파일 항목에서 하드웨어 흐름 제어를 사용으로 설정합니다.

```
unix Any ACU 2400 12015551212 "" \r ogin: Puucp ssword:Passuan "" \ STTY=crtcts
```

이 **pseudo-send** 문자열은 **Dialers** 파일의 항목에서 사용할 수도 있습니다.

## /etc/uucp/Systems 파일에서 패리티 설정

경우에 따라 호출하는 시스템에서 포트 패리티를 확인하고 잘못된 경우 연결을 끊기 때문에 패리티를 재설정해야 할 수 있습니다. **expect-send** 쌍 "" **P\_ZERO**는 상위 비트(패리티 비트)를 0으로 설정합니다. 다음 예의 **expect-send** 쌍을 참조하십시오.

```
unix Any ACU 2400 12015551212 "" P_ZERO "" \r ogin: Puucp ssword:Passuan
```

다음은 **expect-send** 쌍 "" **P\_ZERO** 다음에 올 수 있는 패리티 쌍입니다.

```
"" P_EVEN    패리티를 짝수로 설정합니다(기본값).
```

```
"" P_ODD     패리티를 홀수로 설정합니다.
```

```
"" P_ONE     패리티 비트를 1로 설정합니다.
```

이러한 패리티 쌍은 채트 스크립트에서 원하는 위치에 삽입할 수 있습니다. 패리티 쌍은 채트 스크립트에서 expect-send 쌍 "" P\_ZERO 다음에 오는 모든 정보에 적용됩니다. 패리티 쌍을 Dialers 파일의 항목에서 사용할 수도 있습니다. 다음 예에서는 패리티 쌍 "" P\_ONE이 포함되어 있습니다.

```
unix Any ACU 2400 12015551212 "" P_ZERO "" P_ONE "" \r ogin: Puucp ssword:Passuan
```

## UUCP /etc/uucp/Devices 파일

/etc/uucp/Devices 파일에는 원격 컴퓨터에 대한 링크를 설정하는 데 사용할 수 있는 모든 장치에 대한 정보가 포함됩니다. 이러한 장치에는 ACU(고속 모뎀 포함), 직접 링크 및 네트워크 연결이 포함됩니다.

/etc/uucp/Devices 파일의 항목은 다음과 같은 구문을 사용합니다.

```
Type Line Line2 Class Dialer-Token-Pairs
```

다음은 포트 A에 연결되어 38,400bps로 실행되는 U.S. Robotics V.32bis 모델에 대한 Devices 파일의 항목입니다.

```
ACUEC cua/a - 38400 usrv32bis-ec
```

ACUEC Type 필드의 항목입니다. 자세한 내용은 [534 페이지 “/etc/uucp/Devices 파일의 Type 필드”](#)를 참조하십시오.

cua/a Line 필드의 항목입니다. 자세한 내용은 [536 페이지 “/etc/uucp/Devices 파일의 Line 필드”](#)를

- Line2 필드의 항목입니다. 자세한 내용은 [536 페이지 “/etc/uucp/Devices 파일의 Line2 필드”](#)를 참조하십시오.

38400 Class 필드의 항목입니다. 자세한 내용은 [536 페이지 “/etc/uucp/Devices 파일의 Class 필드”](#)를 참조하십시오.

usrv32bis-ec Dialer-Token-Pairs 필드의 항목입니다. 자세한 내용은 [537 페이지 “/etc/uucp/Devices 파일의 Dialer-Token-Pairs 필드”](#)를 참조하십시오.

각 필드에 대해서는 다음 절에서 설명합니다.

### /etc/uucp/Devices 파일의 Type 필드

이 필드는 장치에서 설정하는 링크 유형에 대해 설명합니다. UUCP Type 필드는 다음에 이어지는 절에서 설명하는 키워드 중 하나를 포함할 수 있습니다.

## Direct 키워드

Direct 키워드는 주로 cu 연결에 대한 항목에 표시됩니다. 이 키워드는 링크가 다른 컴퓨터나 포트 선택기에 대한 직접 링크임을 나타냅니다. 참조할 각 회선에 대해 cu의 -l 옵션을 통해 별도의 항목을 만드십시오.

## ACU 키워드

ACU 키워드는 cu, UUCP, asppp 또는 Solaris PPP 4.0 중 무엇을 사용하는 원격 컴퓨터와 모뎀을 통해 연결된다는 것을 나타냅니다. 이 모뎀은 컴퓨터에 직접 연결하거나 포트 선택기를 통해 간접적으로 연결할 수 있습니다.

## 포트 선택기

Port Selector는 Type 필드에서 포트 선택기 이름으로 대체되는 변수입니다. 포트 선택기는 네트워크에 연결된 장치로, 호출하는 모뎀의 이름을 묻은 다음 액세스를 허가하는 역할을 합니다. /etc/uucp/Dialers 파일에는 micom 및 develcon 포트 선택기에 대한 호출자 스크립트만 포함되어 있습니다. 사용자 고유의 포트 선택기 항목을 Dialers 파일에 추가할 수 있습니다. 자세한 내용은 [540 페이지 “UUCP/etc/uucp/Dialers 파일”](#)을 참조하십시오.

## System-Name 변수

이 변수는 Type 필드의 시스템 이름으로 대체되며 링크가 이 특정 컴퓨터에 대한 직접 링크임을 나타냅니다. 이 이름 지정 체계는 이 Devices 항목의 행을 System-Name 컴퓨터에 대한 /etc/uucp/Systems의 항목과 연결하는 데 사용됩니다.

## Devices 파일 및 Systems 파일의 Type 필드

예 26-5에서는 /etc/uucp/Devices의 필드와 /etc/uucp/Systems의 필드를 비교하여 표시합니다. Devices 파일의 Type 필드에서 사용되는 키워드는 Systems 파일 항목의 세번째 필드와 일치합니다. Devices 파일에서 Type 필드에는 자동 호출 장치(여기서는 V.32bis 모뎀)를 나타내는 ACUEC 항목이 있습니다. 이 값은 Systems 파일의 Type 필드와 일치합니다. 이 필드에도 ACUEC 항목이 포함되어 있습니다. 자세한 내용은 [527 페이지 “UUCP/etc/uucp/Systems 파일”](#)을 참조하십시오.

예 26-5 Devices 파일 및 Systems 파일의 Type 필드 비교

다음은 Devices 파일에 있는 항목의 예입니다.

```
ACUEC cua/a - 38400 usrv32bis-ec
```

다음은 Systems 파일에 있는 항목의 예입니다.

```
Arabian Any ACUEC 38400 111222 ogin: Puucp ssword:beledi
```

## /etc/uucp/Devices 파일의 Line 필드

이 필드에는 Devices 항목과 연결된 회선(포트라고 함)의 장치 이름이 포함됩니다. 특정 항목과 연관된 모뎀이 /dev/cua/a 장치(직렬 포트 A)에 연결된 경우 이 필드에 입력하는 이름은 cua/a입니다. 선택적 모뎀 제어 플래그 M을 Line 필드에 사용하여 반송파를 기다리지 않고 장치를 열도록 지정할 수 있습니다. 예를 들면 다음과 같습니다.

```
cua/a,M
```

## /etc/uucp/Devices 파일의 Line2 필드

이 필드는 개체 틀입니다. 여기서는 항상 하이픈(-)을 사용하십시오. Oracle Solaris OS에서 지원되지 않는 801 유형 전화 걸기에서 Line2 필드를 사용합니다. 801 외의 전화 걸기에서는 일반적으로 이 구성을 사용하지 않지만 여전히 이 필드에 하이픈이 있어야 합니다.

## /etc/uucp/Devices 파일의 Class 필드

Type 필드에 ACU 또는 Direct 키워드를 사용한 경우 Class 필드에는 장치의 속도가 포함됩니다. 그러나 Class 필드는 Centrex 또는 Dimension PBX와 같은 전화 걸기 클래스를 구분하는 문자와 숫자(예: C1200 또는 D1200)를 포함할 수 있습니다.

많은 대형 사무실에서 두 가지가 넘는 유형의 전화 네트워크를 사용할 수 있으므로 이러한 구분이 필요합니다. 하나의 네트워크는 사무실 내부 통신을 지원하는 데에만 사용하고 다른 네트워크는 외부 통신을 처리할 수 있습니다. 이런 경우 내부 통신에 사용할 회선과 외부 통신에 사용할 회선을 구분해야 합니다.

Devices 파일의 Class 필드에서 사용되는 키워드는 Systems 파일의 Speed 필드와 일치합니다.

예 26-6 Devices 파일의 Class 필드

```
ACU   cua/a   -   D2400   hayes
```

일부 장치는 모든 속도로 사용할 수 있으므로 Class 필드에 Any 키워드를 사용할 수 있습니다. Any를 사용하는 경우 이 행은 Systems 파일의 Speed 필드에 필요한 모든 속도와 일치합니다. 이 필드가 Any이고 Systems 파일 Speed 필드가 Any이면 속도는 기본적으로 2400bps입니다.



## /etc/uucp/Devices 파일의 Dialer-Token-Pairs 필드

Dialer-Token-Pairs(DTP) 필드에는 전화 걸기 이름과 이 전화 걸기에 전달할 토큰이 포함됩니다. DTP 필드의 구문은 다음과 같습니다.

*dialer token [dialer token]*

*dialer* 부분은 포트 모니터인 모뎀의 이름이거나 직접 링크 장치의 경우 *direct* 또는 *uudirect*일 수 있습니다. DTP(Dialer-Token Pairs)를 원하는 수만큼 포함할 수 있습니다. *dialer* 부분이 없으면 *Systems* 파일의 관련 항목에서 가져옵니다. *dialer* 부분 바로 뒤에 *token* 부분을 지정할 수 있습니다.

연결된 전화 걸기에 따라서는 마지막 DTP(Dialer-Token Pairs)가 없을 수도 있습니다. 대부분의 경우 마지막 쌍에는 *dialer* 부분만 포함됩니다. *token* 부분은 연결된 *Systems* 파일 항목의 *Phone* 필드에서 가져옵니다.

*dialer* 부분의 유효한 항목은 *Dialers* 파일에서 정의하거나 여러 특수 전화 걸기 유형 중 하나일 수 있습니다. 이러한 특수 전화 걸기 유형은 소프트웨어에 컴파일되어 있으므로 *Dialers* 파일에 항목을 포함하지 않고도 사용할 수 있습니다. 다음 목록에 특수 전화 걸기 유형이 나와 있습니다.

TCP	TCP/IP 네트워크
TLI	전송 레벨 인터페이스 네트워크(STREAMS 포함 안함)
TLIS	전송 레벨 인터페이스 네트워크(STREAMS 포함)

자세한 내용은 539 페이지 “/etc/uucp/Devices 파일의 프로토콜 정의”를 참조하십시오.

## /etc/uucp/Devices 파일의 Dialer-Token-Pairs 필드 구조

항목과 연결된 장치에 따라 DTP 필드는 네 가지 다른 방법으로 구성할 수 있습니다.

DTP 필드를 구성할 수 있는 첫번째 방법을 참조하십시오.

**직접 연결된 모뎀** - 모뎀이 컴퓨터의 포트에 직접 연결된 경우 연결된 *Devices* 파일 항목의 DTP 필드에는 한 쌍만 포함됩니다. 이 쌍은 일반적으로 모뎀의 이름입니다. 이 이름을 사용하여 특정 *Devices* 파일 항목을 *Dialers* 파일의 항목과 일치시킵니다. 따라서 *Dialer* 필드는 *Dialers* 파일 항목의 첫번째 필드와 일치해야 합니다.

예 26-7 직접 연결 모뎀의 *Dialers* 필드

```
Dialers    hayes =, -, ""          \\dA\pTE1V1X1Q0S2=255S12=255\r\c
                                         \EATDT\T\r\c CONNECT
```

Devices 파일 항목의 DTP 필드에 dialer 부분(hayes)만 있습니다. 이는 전화 걸기에 전달할 *token*(이 경우 전화 번호)을 Systems 파일 항목의 Phone 필드에서 가져온다는 것을 나타냅니다. [예 26-9](#)에서 설명한 대로 \T에 암시되어 있습니다.

DTP 필드를 구성할 수 있는 두번째와 세번째 방법을 참조하십시오.

- **직접 링크** - 특정 컴퓨터에 대한 직접 링크의 경우 연결된 항목의 DTP 필드에는 direct 키워드가 포함됩니다. 이 조건은 직접 링크 항목의 유형인 Direct 및 System-Name 모두에 적용됩니다. [534 페이지 “/etc/uucp/Devices 파일의 Type 필드”](#)를 참조하십시오.
- **동일한 포트 선택기의 컴퓨터** - 통신하려는 컴퓨터가 사용자 컴퓨터와 동일한 포트 선택기 스위치에 있는 경우 먼저 스위치에 액세스해야 합니다. 그런 다음 스위치를 통해 다른 컴퓨터에 연결됩니다. 이러한 유형의 항목에는 쌍이 하나만 있습니다. dialer 부분은 Dialers 파일 항목과 일치시키는 데 사용됩니다.

예 26-8 동일한 포트 선택기에 있는 컴퓨터의 UUCP Dialers 필드

```
Dialers      develcon , ""          ""          \pr\ps\c est:\007 \E\D\e \007
```

표시된 대로 *token* 부분은 비워 둡니다. 이러한 지정은 토큰을 Systems 파일에서 가져온다는 것을 나타냅니다. 이 컴퓨터의 Systems 파일 항목에는 일반적으로 컴퓨터의 전화 번호용으로 예약된 Phone 필드의 토큰이 포함되어 있습니다. 자세한 내용은 [527 페이지 “UUCP /etc/uucp/Systems 파일”](#)을 참조하십시오. 이 유형의 DTP에는 Phone 필드의 내용이 Dialcodes 파일의 유효한 항목으로 해석되지 않도록 하는 제어 문자(\D)가 포함됩니다.

DTP 필드를 구성할 수 있는 네번째 방법을 참조하십시오.

**포트 선택기에 연결된 모뎀** - 고속 모뎀이 포트 선택기에 연결된 경우 컴퓨터는 먼저 포트 선택기 스위치에 액세스해야 합니다. 그런 다음 스위치를 통해 모뎀에 연결됩니다. 이러한 유형의 항목에는 전화 걸기-토큰 쌍이 두 개 필요합니다. 각 쌍의 *dialer* 부분(항목의 다섯번째와 일곱번째 필드)은 다음과 같이 Dialers 파일의 항목과 일치시키는 데 사용됩니다.

예 26-9 포트 선택기에 연결된 모뎀의 UUCP Dialers 필드

```
develcon ""      ""      \pr\ps\c est:\007      \E\D\e      \007
ventel  =&-%      t""      \r\p\r\c $      <K\T%\r>\c ONLINE!
```

첫번째 쌍에서 develcon은 전화 걸기이고 vent는 컴퓨터에 연결할 장치(예: Ventel 모뎀)를 알려 주기 위해 Develcon 스위치에 전달되는 토큰입니다. 각 스위치가 다르게 설정될 수 있으므로 이 토큰은 각 포트 선택기에 대해 고유합니다. Ventel 모뎀이 연결되면 두번째 쌍에 액세스합니다. Ventel은 전화 걸기이고 토큰은 Systems 파일에서 가져옵니다.

다음과 같은 두 개의 제어 문자가 DTP 필드에 표시될 수 있습니다.

- \T - Phone(token) 필드를 /etc/uucp/Dialcodes 파일을 사용하여 해석해야 함을 나타냅니다. 일반적으로 이 제어 문자는 Hayes, U.S. Robotics 등의 모뎀과 연결된 각 호출자 스크립트의 /etc/uucp/Dialers 파일에 있습니다. 따라서 호출자 스크립트에 액세스할 때까지는 해석되지 않습니다.
- \D - Phone(token) 필드를 /etc/uucp/Dialcodes 파일을 사용하여 해석하지 않아야 함을 나타냅니다. Devices 항목 끝에 제어 문자를 지정하지 않으면 \D(기본값)로 간주됩니다. \D도 /etc/uucp/Dialers 파일에서 네트워크 스위치 develcon 및 micom과 연결된 항목에 사용됩니다.

## /etc/uucp/Devices 파일의 프로토콜 정의

/etc/uucp/Devices의 각 장치에서 사용할 프로토콜을 정의할 수 있습니다. 기본값을 사용하거나 호출하는 특정 시스템에서 프로토콜을 정의할 수 있으므로 일반적으로 이 지정은 필요하지 않습니다. 자세한 내용은 [527 페이지 “UUCP/etc/uucp/Systems 파일”](#)을 참조하십시오. 프로토콜을 지정하는 경우 다음과 같은 형식을 사용해야 합니다.

*Type, Protocol [parameters]*

예를 들어 TCP, te를 사용하여 TCP/IP 프로토콜을 지정할 수 있습니다.

다음 표에서는 Devices 파일에서 사용할 수 있는 프로토콜이 나와 있습니다.

표 26-2 /etc/uucp/Devices에서 사용되는 프로토콜

프로토콜	설명
t	이 프로토콜은 일반적으로 TCP/IP를 통한 전송 및 다른 신뢰할 수 있는 연결에 대해 사용됩니다. t에서는 오류 없는 전송을 가정합니다.
g	이 프로토콜은 UUCP의 기본 프로토콜입니다. g는 느리고 신뢰할 수 있으며 잡음이 많은 전화선을 통한 전송에 적합합니다.
e	이 프로토콜에서는 TCP/IP와 같은 바이트 스트림 지향이 아닌 메시지 지향인 오류 없는 채널을 통한 전송을 가정합니다.
f	이 프로토콜은 X.25 연결을 통한 전송에 사용됩니다. f는 데이터 스트림에 대한 흐름 제어를 사용하며 거의 오류가 없다고 보장할 수 있는 링크(특히 X.25/PAD 링크)를 통해 작동하기 위한 것입니다. 체크섬은 전체 파일에 대해서만 시행됩니다. 전송이 실패하는 경우 받는 사람은 재전송을 요청할 수 있습니다.

다음은 장치 항목에 대한 프로토콜 지정을 보여주는 예입니다.

TCP, te - - Any TCP -

이 예에서는 TCP 장치에 대해 t 프로토콜을 사용해 보아야 한다는 것을 나타냅니다. 전송의 다른 한쪽이 거부하면 e 프로토콜을 사용합니다.

e도 t도 모뎀을 통해 사용하는 데에는 적합하지 않습니다. 모뎀에서 오류 없는 전송을 보장하더라도 여전히 모뎀과 CPU 사이에서 데이터가 삭제될 수 있습니다.

## UUCP /etc/uucp/Dialers 파일

/etc/uucp/Dialers 파일에는 일반적으로 사용되는 모뎀의 전화 걸기 명령이 포함됩니다. 비표준 모뎀을 사용하거나 UUCP 환경을 사용자 정의하려는 계획이 없는 경우 이 파일의 항목을 변경하거나 추가할 필요가 없을 것입니다. 그렇지만 파일에 무엇이 있고 이것이 Systems 및 Devices 파일과 어떤 관련이 있는지 알아야 합니다.

텍스트에서는 회선을 데이터 전송에 사용할 수 있으려면 회선에서 수행해야 하는 초기 대화를 지정합니다. 채트 스크립트라고 하는 이 대화는 일반적으로 전송되며 필요한 ASCII 문자열의 시퀀스입니다. 채트 스크립트는 종종 전화 번호로 전화를 거는 데 사용됩니다.

534 페이지 “UUCP /etc/uucp/Devices 파일”의 예에 나온 대로 Devices 파일 항목의 다섯번째 필드는 Dialers 파일의 색인이거나 TCP, TLI, TLIS 등과 같은 특수 전화 걸기 유형입니다. uucico 데몬은 Devices 파일의 다섯번째 필드를 각 Dialers 파일 항목의 첫번째 필드와 일치시키려고 합니다. 또한 일곱번째 위치에서 시작하는 홀수 번호가 매겨진 각 Devices 필드는 Dialers 파일의 인덱스로 사용됩니다. 일치에 성공하면 Dialers 항목은 전화 걸기 대화를 수행하는 것으로 해석됩니다.

Dialers 파일의 각 항목은 다음과 같은 구문을 사용합니다.

```
dialer substitutions expect-send
```

다음 예에서는 U.S. Robotics V.32bis 모뎀의 항목을 보여줍니다.

예 26-10 /etc/uucp/Dialers 파일의 항목

```
usrv32bis-e    =, -, ""    dA\pT&FE1V1X1Q0S2=255S12=255&A1&H1&M5&B2&W\r\c OK\r
                \EATDT\T\r\c CONNECT\s14400/ARQ STTY=crtscs
```

usrv32bis-e

Dialer 필드의 항목입니다. Dialer 필드는 Devices 파일의 다섯번째 필드 및 추가 홀수 번호 필드와 일치합니다.

=, -, ""

Substitutions 필드의 항목입니다. Substitutions 필드는 변환 문자열입니다. 각 문자 쌍의 첫번째는 쌍의 두번째 문자에 매핑됩니다. 이 매핑은 일반적으로 = 및 -를 전화 걸기에 필요한 “wait for dial tone” 및 “pause”와 같은 문자열로 변환하는 데 사용됩니다

```
dA\pT&FE1V1X1Q0S2=255S12=255&A1&H1&M5&B2&W\r\c OK\r
```

Expect-Send 필드의 항목입니다. Expect-Send 필드는 문자열입니다.

\EATDT\T\r\c CONNECT\s14400/ARQ STTY=crtscts  
Expect-Send 필드의 추가 항목입니다.

다음 예에서는 UUCP를 Oracle Solaris 설치 프로그램의 일부로 설치할 때 배포되는 Dialers 파일의 샘플 항목을 보여줍니다.

예 26-11 /etc/uucp/Dialers의 인용구

```
penril      =W-P "" \d > Q\c : \d- > s\p9\c )-W\p\r\ds\p9\c-) y\c : \E\TP > 9\c OK

ventel      =&-% "" \r\p\r\c $ <K\T%\r>\c ONLINE!

vadic       =K-K "" \005\p *-\005\p-*\005\p-* D\p BER? \E\T\e \r\c LINE

develcon     "" "" \pr\ps\c est:\007

\E\D\e \n\007 micom "" "" \s\c NAME? \D\r\c GO

hayes       =,-, "" \dA\pTE1V1X1Q0S2=255S12=255\r\c OK\r \EATDT\T\r\c CONNECT

# Telebit TrailBlazer
tb1200      =W-, "" \dA\pA\pA\pTE1V1X1Q0S2=255S12=255S50=2\r\c OK\r
\EATDT\T\r\c CONNECT\s1200
tb2400      =W-, "" \dA\pA\pA\pTE1V1X1Q0S2=255S12=255S50=3\r\c OK\r
\EATDT\T\r\c CONNECT\s2400
tbfast      =W-, "" \dA\pA\pA\pTE1V1X1Q0S2=255S12=255S50=255\r\c OK\r
\EATDT\T\r\c CONNECT\sFAST

# USrobotics, Codes, and DSI modems

dsi-ec =,-, "" \dA\pTE1V1X5Q0S2=255S12=255*E1*F3*M1*S1\r\c OK\r \EATDT\T\r\c
CONNECT\sEC STTY=crtscts,crtsexoff

dsi-nec =,-, "" \dA\pTE1V1X5Q0S2=255S12=255*E0*F3*M1*S1\r\c OK\r \EATDT\T\r\c CONNECT
STTY=crtscts,crtsexoff

usrv32bis-ec =,-, "" \dA\pT&FE1V1X1Q0S2=255S12=255&A1&H1&M5&B2&W\r\c OK\r \EATDT\T\r\c
CONNECT\s14400/ARQ STTY=crtscts,crtsexoff

usrv32-nec =,-, "" \dA\pT&FE1V1X1Q0S2=255S12=255&A0&H1&M0&B0&W\r\c OK\r \EATDT\T\r\c
CONNECT STTY=crtscts,crtsexoff

codex-fast =,-, "" \dA\pT&C1&D2*MF0*AA1&R1&S1*DE15*FL3S2=255S7=40S10=40*TT5&W\r\c OK\r
\EATDT\T\r\c CONNECT\s38400 STTY=crtscts,crtsexoff

tb9600-ec =W-, "" \dA\pA\pA\pTE1V1X1Q0S2=255S12=255S50=6\r\c OK\r
\EATDT\T\r\cCONNECT\s9600 STTY=crtscts,crtsexoff

tb9600-nec =W-, "" \dA\pA\pA\pTE1V1X1Q0S2=255S12=255S50=6S180=0\r\c OK\r \EATDT\T\r\c
CONNECT\s9600 STTY=crtscts,crtsexoff
```

다음 표에서는 Dialers 파일의 보내기 문자열에서 일반적으로 사용되는 제어 문자가 나와 있습니다.

표 26-3 /etc/uucp/Dialers의 백슬래시 문자

문자	설명
\b	백스페이스 문자를 보내거나 이 문자가 필요합니다.
\c	개행 또는 캐리지 리턴이 없습니다.
\d	약 2초 동안 지연됩니다.
\D	Dialcodes 변환이 없는 전화 번호 또는 토큰입니다.
\e	에코 검사를 사용 안함으로 설정합니다.
\E	느린 장치에 대한 에코 검사를 사용으로 설정합니다.
\K	Break 문자를 삽입합니다.
\n	개행을 보냅니다.
\nnn	8진수를 보냅니다. 사용할 수 있는 추가 제어 문자는 527 페이지 “UUCP /etc/uucp/Systems 파일” 절에 나와 있습니다.
\N	널 문자(ASCII NUL)를 보내거나 이 문자가 필요합니다.
\p	약 12-14초 동안 일시 중지합니다.
\r	반환합니다.
\s	공백 문자를 보내거나 이 문자가 필요합니다.
\T	Dialcodes 변환이 있는 전화 번호 또는 토큰입니다.

다음은 Dialers 파일의 penril 항목입니다.

```
penril =W-P "" \d > Q\c : \d- > s\p9\c )-W\p\r\ds\p9\c-) y\c : \E\TP > 9\c OK
```

먼저 =는 W(발신음 대기)로 대체되고 -는 P(일시 중지)로 대체되도록 전화 번호 인수에 대한 대체 방식이 설정됩니다.

행의 나머지에 주어진 핸드셰이크는 다음과 같이 작동합니다.

- "" - 다음 단계로 진행하는 것을 의미하는 없음을 대기합니다.
- \d - 2초 지연한 다음 캐리지 리턴을 보냅니다.
- >->를 대기합니다.
- Q\c - 캐리지 리턴 없이 Q를 보냅니다.
- :-:이 필요합니다.
- \d- - 2초 지연하고 -와 캐리지 리턴을 보냅니다.
- >->를 대기합니다.
- s\p9\c - s를 보내고 일시 중지하고 캐리지 리턴 없이 9를 보냅니다.

- )-W\p\r\ds\p9\c-) - )를 대기합니다.)를 받지 못하면 뒤에 오는 - 문자 사이의 문자열을 처리합니다.w를 보내고, 일시 중지하고, 캐리지 리턴을 보내고, 지연하고, s를 보내고, 일시 중지하고, 캐리지 리턴 없이 9를 보낸 다음 )를 대기합니다.
- y\c - 캐리지 리턴 없이 y를 보냅니다.
- : - :을 대기합니다.
- \E\TP - \E는 에코 검사를 사용으로 설정합니다. 이때부터 문자가 전송될 때마다 UUCP는 문자를 받을 때까지 대기한 다음 진행합니다. 그런 다음 UUCP는 전화 번호를 보냅니다.\T는 전달된 전화 번호를 인수로 사용한다는 것을 의미합니다.\T는 Dialcodes 변환 및 이 항목의 필드 2에서 지정하는 모뎀 기능 변환에 적용됩니다. 그런 다음 \T는 P와 캐리지 리턴을 보냅니다.
- > ->를 대기합니다.
- 9\c - 개행 없이 9를 보냅니다.
- OK - OK 문자열을 대기합니다.

## /etc/uucp/Dialers 파일에서 하드웨어 흐름 제어를 사용으로 설정

pseudo-send STTY=*value* 문자열을 사용하여 모뎀 특성을 설정할 수도 있습니다. 예를 들어 STTY=crtscts는 아웃바운드 하드웨어 흐름 제어를 사용으로 설정 STTY=crtsexoff는 인바운드 하드웨어 흐름 제어를 사용으로 설정합니다. STTY=crtscts, crtsexoff는 아웃바운드 및 인바운드 하드웨어 흐름 제어를 사용으로 설정합니다.

STTY에는 모든 stty 모드를 사용할 수 있습니다. [stty\(1\)](#) 및 [termio\(7I\)](#) man 페이지를 참조하십시오.

다음 예에서는 Dialers 항목에서 하드웨어 흐름 제어를 사용으로 설정합니다.

```
dsi =, -, "" \dA\pTE1V1X5Q0S2=255S12=255*E1*F3*M1*S1\r\c OK\r \EATDT\T\r\c
CONNECT\sEC STTY=crtscts
```

pseudo-send 문자열을 Systems 파일의 항목에서도 사용할 수 있습니다.

## /etc/uucp/Dialers 파일에서 패리티 설정

경우에 따라 호출하는 시스템에서 포트 패리티를 확인하고 잘못된 경우 연결을 끊기 때문에 패리티를 재설정해야 할 수 있습니다. 다음과 같은 expect-send 쌍 P\_ZERO는 패리티를 0으로 설정합니다.

```
foo =, -, "" P_ZERO "" \dA\pTE1V1X1Q0S2=255S12=255\r\c OK\r\EATDT\T\r\c CONNECT
```

다음은 expect-send 쌍 다음에 올 수 있는 패리티 쌍입니다.

```
"" P_EVEN    패리티를 짝수로 설정합니다(기본값).
"" P_ODD     패리티를 홀수로 설정합니다.
"" P_ONE     패리티를 1로 설정합니다.
```

pseudo-send 문자열을 Systems 파일의 항목에서 사용할 수도 있습니다.

## 다른 기본 UUCP 구성 파일

기본 UUCP 구성을 수행할 때 Systems, Devices 및 Dialers 파일과 더불어 이 절에서 설명하는 파일을 사용할 수 있습니다.

### UUCP /etc/uucp/Dialcodes 파일

/etc/uucp/Dialcodes 파일을 사용하여 /etc/uucp/Systems 파일의 Phone 필드에서 사용할 수 있는 다이얼 번호 약어를 정의할 수 있습니다. Dialcodes 파일을 사용하여 동일한 사이트의 여러 시스템에서 사용하는 기본 전화 번호에 대한 추가 정보를 제공할 수 있습니다.

각 항목의 구문은 다음과 같습니다.

**Abbreviation**    **Dial-Sequence**

**Abbreviation**    이 필드에서는 Systems 파일의 Phone 필드에서 사용되는 약어를 제공합니다.

**Dial-Sequence**    이 필드에서는 특정 Systems 파일 항목에 액세스할 때 전화 걸기에 전달되는 전화 걸기 시퀀스를 제공합니다.

두 파일의 필드를 비교하십시오. 다음은 Dialcodes 파일의 필드입니다.

**Abbreviation**    **Dial-Sequence**

다음은 Systems 파일의 필드입니다.

**System-Name**    **Time**    **Type**    **Speed**    **Phone**    **Chat**    **Script**

다음 표에는 Dialcodes 파일에 있는 필드의 샘플 내용이 나와 있습니다.

표 26-4 Dialcodes 파일의 항목

약어	Dial-Sequence
NY	1=212



표 26-4 Dialcodes 파일의 항목 (계속)

약어	Dial-Sequence
jt	9+847

첫번째 행에서 NY는 Systems 파일의 Phone 필드에 표시되는 약어입니다. 예를 들어 Systems 파일에는 다음과 같은 항목이 있을 수 있습니다.

NY5551212

uucico가 Systems 파일에서 NY를 읽으면 uucico는 Dialcodes 파일에서 NY를 검색하고 전화 걸기 시퀀스 1=212 를 가져옵니다. 1=212는 뉴욕에 거는 모든 전화에 필요한 전화 걸기 시퀀스입니다. 이 시퀀스에는 숫자 1, 일시 중지하고 두번째 발신음을 대기하는 것을 의미하는 “등호”(=) 및 지역 번호 212가 포함됩니다. uucico는 이 정보를 전화 걸기에 보낸 다음 Systems 파일로 돌아와 나머지 전화 번호 5551212를 찾습니다.

jt 9=847- 항목은 Systems 파일의 Phone 필드(예: jt7867)와 함께 작동합니다. uucico가 Systems 파일에서 jt7867을 포함하는 항목을 읽으면 uucico는 DTP(Dialer-Token Pairs)의 토큰이 \T인 경우 9=847-7867 시퀀스를 전화 걸기에 보냅니다.

## UUCP /etc/uucp/Sysfiles 파일

/etc/uucp/Sysfiles 파일에서는 uucp 및 cu에서 사용할 Systems, Devices 및 Dialers 파일과 같은 여러 파일을 지정할 수 있습니다. cu에 대한 자세한 내용은 [cu\(1C\)](#) man 페이지를 참조하십시오. Sysfiles를 다음 파일에 대해 사용할 수 있습니다.

- 다른 Systems 파일. uucp 서비스와 다른 주소로 로그인 서비스를 요청할 수 있습니다.
- 다른 Dialers 파일. cu 및 uucp에 대해 다른 핸드셰이크를 지정할 수 있습니다.
- 여러 Systems, Dialers 및 Devices 파일. 특히 Systems 파일은 커질 수 있으므로 여러 작은 파일로 분할하면 더 간편해집니다.

Sysfiles 파일의 구문은 다음과 같습니다.

```
service=w systems=x:x dialers=y:y devices=z:z
```

w uucico, cu 또는 두 명령 모두를 콜론으로 구분하여 나타냅니다

x Systems 파일로 사용할 하나 이상의 파일을 나타내며, 각 파일 이름을 콜론으로 구분하고 제공되는 순서대로 읽습니다

y Dialers 파일로 사용할 하나 이상의 파일을 나타냅니다.

z Devices 파일로 사용할 하나 이상의 파일을 나타냅니다.

각 파일 이름은 전체 경로를 제공하지 않은 경우 /etc/uucp 디렉토리에 대한 상대 경로로 간주됩니다.

다음 샘플 /etc/uucp/Sysfiles에서는 로컬 Systems 파일(Local\_Systems) 및 표준 /etc/uucp/Systems 파일을 정의합니다.

```
service=uucico:cu systems=Systems :Local_Systems
```

이 항목이 /etc/uucp/Sysfiles에 있으면 uucico 및 cu는 모두 먼저 표준 /etc/uucp/Systems를 확인합니다. 호출되는 시스템의 항목이 이 파일에 없거나 파일의 항목이 실패하는 경우 두 명령은 모두 /etc/uucp/Local\_Systems를 확인합니다.

이전 항목에 지정된 대로 cu 및 uucico는 Dialers 및 Devices 파일을 공유합니다.

uucico 및 cu 시스템에 대해 서로 다른 Systems 파일을 정의한 경우 시스템에서는 두 가지 다른 Systems 목록을 저장합니다. uucico 목록을 인쇄하려면 uuname 명령을 사용하고 cu 목록을 인쇄하려면 uuname -C 명령을 사용합니다. 다음은 먼저 대체 파일을 참조하고 필요한 경우 기본 파일을 참조하는 것을 보여주는 파일의 다른 예입니다.

```
service=uucico systems=Systems.cico:Systems
dialers=Dialers.cico:Dialers \
devices=Devices.cico:Devices
service=cu systems=Systems.cu:Systems \
dialers=Dialers.cu:Dialers \
devices=Devices.cu:Devices
```

## UUCP /etc/uucp/Sysname 파일

UUCP를 사용하는 모든 시스템에는 식별 이름(노드 이름이라고도 함)이 있어야 합니다. 노드 이름은 원격 시스템의 /etc/uucp/Systems 파일에 채트 스크립트 및 다른 식별 정보와 함께 표시됩니다. 일반적으로 UUCP에서는 uname -n 명령으로 반환되는 것과 동일한 노드 이름을 사용하며 이 이름은 TCP/IP에서도 사용됩니다.

/etc/uucp/Sysname 파일을 만들면 UUCP 노드 이름을 TCP/IP 호스트 이름과 별개로 지정할 수 있습니다. 파일에는 시스템의 UUCP 노드 이름을 포함하는 한 행 항목이 있습니다.

## UUCP /etc/uucp/Permissions 파일

/etc/uucp/Permissions 파일에서는 원격 컴퓨터의 로그인, 파일 액세스 및 명령 실행을 위한 사용 권한을 지정합니다. 일부 옵션은 원격 컴퓨터가 파일을 요청하고 로컬 컴퓨터에 대기된 파일을 받을 수 있는 기능을 제한합니다. 다른 옵션은 원격 컴퓨터가 로컬 컴퓨터에서 실행할 수 있는 명령을 지정하는 데 사용할 수 있습니다.

## UUCP 구성 항목

각 항목은 논리적 행이며, 연속을 나타내는 백슬래시(\)로 끝나는 물리적 행을 포함합니다. 항목은 공백으로 구분된 옵션으로 구성됩니다. 각 옵션은 다음과 같은 형식의 이름-값 쌍입니다.

*name=value*

*Values*는 콜론으로 구분된 목록일 수 있습니다. 옵션 지정 내에 공백은 사용할 수 없습니다.

주석 행은 파운드 기호(#)로 시작하며 개행 문자까지의 전체 행을 사용합니다. 여러 행 항목 내에 있는 것을 포함하여 빈 행은 무시됩니다.

Permissions 파일 항목의 유형은 다음과 같습니다.

- **LOGNAME** - 원격 컴퓨터가 사용자 컴퓨터에 로그인(호출)할 때 적용되는 사용 권한을 지정합니다.

---

주 - 원격 컴퓨터가 호출할 때 고유한 로그인 및 검증 가능한 암호를 사용하지 않는 경우 신원이 의심스러울 수 있습니다.

---

- **MACHINE** - 사용자 컴퓨터가 원격 컴퓨터에 로그인(호출)할 때 적용되는 사용 권한을 지정합니다.

LOGNAME 항목에는 LOGNAME 옵션이 있고, MACHINE 항목에는 MACHINE 옵션이 있습니다. 하나의 항목에 두 옵션을 모두 포함할 수 있습니다.

## UUCP 고려 사항

Permissions 파일을 사용하여 원격 컴퓨터에 부여되는 액세스 레벨을 제한하려면 다음을 고려해야 합니다.

- 원격 컴퓨터가 UUCP 통신을 위해 로그인하는 데 사용하는 로그인 ID는 하나의 LOGNAME 항목에만 표시되어야 합니다.
- MACHINE 항목에 표시되지 않은 이름으로 호출되는 모든 사이트에는 다음과 같은 기본 사용 권한 또는 제한 사항이 적용됩니다.
  - 로컬 보내기 및 수신 요청이 실행됩니다.
  - 원격 컴퓨터는 파일을 사용자 컴퓨터의 /var/spool/uucppublic 디렉토리로 보낼 수 있습니다.
  - 원격 컴퓨터가 사용자 컴퓨터에서 실행하기 위해 보내는 명령은 기본 명령 중 하나(일반적으로 rmail)여야 합니다.

## UUCP REQUEST 옵션

원격 컴퓨터가 사용자 컴퓨터를 호출하여 파일 수신을 요청하면 이 요청을 허용하거나 거부할 수 있습니다. **REQUEST** 옵션은 원격 컴퓨터가 사용자 컴퓨터에서 파일 전송을 설정하도록 요청할 수 있는지 여부를 지정합니다. **REQUEST=yes** 문자열은 원격 컴퓨터가 사용자 컴퓨터에서 파일 전송을 요청할 수 있도록 지정합니다. **REQUEST=no** 문자열은 원격 컴퓨터가 사용자 컴퓨터에서 파일 전송을 요청할 수 없도록 지정합니다.

**REQUEST=no**(기본값)는 **REQUEST** 옵션을 지정하지 않는 경우에 사용됩니다. **REQUEST** 옵션은 원격 컴퓨터가 사용자를 호출할 수 있도록 **LOGNAME** 항목에 표시되거나 사용자가 원격 컴퓨터를 호출할 수 있도록 **MACHINE** 항목에 표시될 수 있습니다.

## UUCP SENDFILES 옵션

원격 컴퓨터가 사용자 컴퓨터를 호출하고 작업을 완료하면 원격 컴퓨터는 사용자 컴퓨터의 대기열에 있는 작업을 검색하려고 할 수 있습니다. **SENDFILES** 옵션은 사용자 컴퓨터가 원격 컴퓨터의 대기열에 있는 작업을 보낼 수 있는지 여부를 지정합니다.

**SENDFILES=yes** 문자열은 사용자 컴퓨터가 **LOGNAME** 옵션의 이름 중 하나로 로그인한 경우 원격 컴퓨터의 대기열에 있는 작업을 보낼 수 있는지 여부를 지정합니다.

**/etc/uucp/Systems**의 **Time** 필드에 **Never**를 입력한 경우 이 문자열은 필수입니다. 이렇게 지정하면 로컬 컴퓨터가 수동 모드로 설정되지만 이 특정 원격 컴퓨터에 대한 호출을 시작할 수 없습니다. 자세한 내용은 [527 페이지 “UUCP /etc/uucp/Systems 파일”](#)을 참조하십시오.

**SENDFILES=call** 문자열은 사용자 컴퓨터의 대기열에 있는 파일이 사용자 컴퓨터에서 원격 컴퓨터를 호출할 때만 전송되도록 지정합니다. **call** 값은 **SENDFILES** 옵션의 기본값입니다. **MACHINE** 항목은 호출이 원격 컴퓨터로 전송될 때 적용되므로 이 옵션은 **LOGNAME** 항목에서만 중요합니다. **MACHINE** 항목에 사용하는 경우 이 옵션은 무시됩니다.

## UUCP MYNAME 옵션

이 옵션을 사용하여 컴퓨터의 고유한 UUCP 노드 이름과 **hostname** 명령으로 반환되는 TCP/IP 호스트 이름을 지정할 수 있습니다. 예를 들어 실수로 호스트 이름을 다른 시스템과 동일하게 지정한 경우 **Permissions** 파일의 **MYNAME** 옵션을 설정할 수 있습니다. 자신의 조직을 **widget**으로 알리려 한다고 가정합니다. 모든 모뎀이 호스트 이름이 **gadget**인 시스템에 연결된 경우 **gadget**의 **Permissions** 파일의 항목은 다음과 같을 수 있습니다.

```
service=uucico systems=Systems.cico:Systems
  dialers=Dialers.cico:Dialers \
  devices=Devices.cico:Devices
service=cu systems=Systems.cu:Systems \
```

```
dialers=Dialers.cu:Dialers \
devices=Devices.cu:Devices
```

이제 시스템 world가 widget에 로그인하는 것처럼 컴퓨터 gadget에 로그인할 수 있습니다. 사용자가 컴퓨터 world를 호출할 때 사용자도 가칭된 이름 widget으로 인식되도록 하려면 다음과 같은 항목이 있을 수 있습니다.

```
MACHINE=world MYNAME=widget
```

MYNAME 옵션은 컴퓨터가 자신을 호출하는 것도 허용하므로 테스트 용도로도 사용할 수 있습니다. 그러나 이 옵션은 컴퓨터의 실제 신원을 마스킹하는 데 사용될 수 있으므로 [552 페이지 “UUCP VALIDATE 옵션”](#)에 설명된 VALIDATE 옵션을 사용해야 합니다.

## UUCP READ 및 WRITE 옵션

이러한 옵션은 uucico가 읽거나 쓸 수 있는 파일 시스템의 여러 부분을 지정합니다. READ 및 WRITE 옵션은 MACHINE 또는 LOGNAME 항목에 지정할 수 있습니다.

READ 및 WRITE 옵션의 기본값은 모두 다음 문자열에 표시된 uucppublic 디렉토리입니다.

```
READ=/var/spool/uucppublic WRITE=/var/spool/uucppublic
```

READ=/ 및 WRITE=/ 문자열은 기타 권한을 가진 로컬 사용자가 액세스할 수 있는 모든 파일에 액세스할 수 있는 권한을 지정합니다.

이러한 항목의 값은 콜론으로 구분한 경로 이름 목록입니다. READ 옵션은 파일을 요청하기 위한 것이고 WRITE 옵션은 파일을 배치하기 위한 것입니다. 값 중 하나는 입력하는 파일이나 기존 파일에 대한 전체 경로 이름의 접두어여야 합니다. 파일을 /usr/news 및 공용 디렉토리에 배치할 수 있는 권한을 부여하려면 WRITE 옵션에 다음 값을 사용합니다.

```
WRITE=/var/spool/uucppublic:/usr/news
```

READ 및 WRITE 옵션을 사용하는 경우 경로 이름은 기본 목록에 추가되지 않으므로 경로 이름을 모두 지정해야 합니다. 예를 들어 /usr/news 경로 이름만 WRITE 옵션에 지정한 경우 파일을 공용 디렉토리에 배치하는 권한은 거부됩니다.

원격 시스템이 액세스하여 읽고 쓸 수 있게 할 디렉토리를 주의해서 선택해야 합니다. 예를 들어 /etc 디렉토리에는 중요한 시스템 파일이 많이 포함되어 있습니다. 원격 사용자는 이 디렉토리에 파일을 배치할 수 있는 권한이 없어야 합니다.

## UUCP NOREAD 및 NOWRITE 옵션

NOREAD 및 NOWRITE 옵션은 READ 및 WRITE 옵션 또는 기본값에 대한 예외를 지정합니다. 다음 항목은 /etc 디렉토리와 그 하위 디렉토리에 있는 파일을 제외한 모든 파일의 읽기를 허용합니다. 이러한 옵션은 점두어입니다.

READ=/ NOREAD=/etc WRITE=/var/spool/uucppublic

이 항목은 기본 /var/spool/uucppublic 디렉토리에만 쓰기를 허용합니다. NOWRITE 옵션은 NOREAD 옵션과 작동 방식이 동일합니다. NOREAD 및 NOWRITE 옵션은 LOGNAME 및 MACHINE 항목 모두에 지정할 수 있습니다.

## UUCP CALLBACK 옵션

LOGNAME 항목에서 콜백 옵션을 사용하여 호출 시스템이 콜백할 때까지 트랜잭션이 발생하지 않도록 지정할 수 있습니다. CALLBACK을 설정하는 이유는 다음과 같습니다.

- 보안 목적 - 컴퓨터에 콜백하는 경우 올바른 컴퓨터인지 확인할 수 있습니다.
- 회계 목적 - 긴 데이터 전송을 수행하는 경우 장거리 전화에 대해 청구되는 컴퓨터를 선택할 수 있습니다.

CALLBACK=yes 문자열은 사용자 컴퓨터가 원격 컴퓨터에 콜백한 이후에만 파일 전송이 이루어질 수 있도록 지정합니다.

CALLBACK 옵션의 기본값은 CALLBACK=no입니다. CALLBACK을 yes로 설정하는 경우 나머지 대화에 영향을 주는 권한을 호출자에 해당하는 MACHINE 항목에 지정해야 합니다. LOGNAME 또는 원격 컴퓨터가 사용자 호스트에 대해 설정했을 수 있는 LOGNAME 항목에는 이러한 권한을 지정하지 마십시오.

---

주 - 두 사이트가 서로에 대해 콜백 옵션을 설정한 경우 대화가 시작되지 않습니다.

---

## UUCP COMMANDS 옵션




---

주의 - COMMANDS 옵션은 시스템의 보안을 손상할 수 있습니다. 이 옵션은 특히 주의해서 사용하십시오.

---

MACHINE 항목에서 COMMANDS 옵션을 사용하면 원격 컴퓨터가 사용자 컴퓨터에서 실행할 수 있는 명령을 지정할 수 있습니다. uux 프로그램은 원격 실행 요청을 생성하고 원격

컴퓨터에 전송할 요청을 대기열에 대기시킵니다. 파일 및 명령은 대상 컴퓨터로 전송되어 원격 실행됩니다. 이는 시스템이 콜아웃할 때만 MACHINE 항목이 적용된다는 규칙에 대한 예외입니다.

COMMANDS는 LOGNAME 항목에서 사용되지 않습니다. MACHINE 항목의 COMMANDS는 사용자가 원격 시스템을 호출하거나 원격 시스템이 사용자를 호출하는 두 경우 모두에 명령 권한을 정의합니다.

COMMANDS=rmail 문자열은 원격 컴퓨터가 사용자 컴퓨터에서 실행할 수 있는 기본 명령을 지정합니다. MACHINE 항목에서 명령 문자열을 사용하는 경우 기본 명령이 대체됩니다. 예를 들어 다음 항목은 COMMAND 기본값을 대체하여 owl, raven, hawk 및 dove라는 컴퓨터가 사용자 컴퓨터에서 rmail, rnews 및 lp를 실행할 수 있게 합니다.

```
MACHINE=owl:raven:hawk:dove COMMANDS=rmail:rnews:lp
```

방금 지정한 이름뿐 아니라 명령의 전체 경로 이름도 포함할 수 있습니다. 예를 들어 다음 항목은 rmail 명령에서 기본 검색 경로를 사용하도록 지정합니다.

```
COMMANDS=rmail:/usr/local/rnews:/usr/local/lp
```

UUCP의 기본 검색 경로는 /bin 및 /usr/bin입니다. 원격 컴퓨터에서 rnews 또는 /usr/local/rnews를 실행할 명령으로 지정하는 경우 /usr/local/rnews는 기본 경로와 상관없이 실행됩니다. 마찬가지로 /usr/local/lp는 실행되는 lp 명령입니다.

목록에 ALL 값을 포함하면 항목에 지정된 원격 컴퓨터의 모든 명령이 실행됩니다. 이 값을 사용하는 경우 원격 컴퓨터에게 사용자 컴퓨터에 대한 모든 권한을 부여합니다.



**주의** - 이 값을 사용하면 일반 사용자보다 더 많은 액세스 권한이 허용됩니다. 이 값은 두 컴퓨터가 동일한 사이트에 있고 밀접하게 연결되어 있으며 사용자를 신뢰할 수 있는 경우에만 사용해야 합니다.

다음은 ALL 값이 추가된 문자열입니다.

```
COMMANDS=/usr/local/rnews:ALL:/usr/local/lp
```

이 문자열에서는 다음 두 가지를 보여줍니다.

- ALL 값은 문자열에서 아무 곳이나 표시될 수 있습니다.
- 요청된 명령에 rnews 또는 lp의 전체 경로 이름이 포함되지 않은 경우 rnews 및 lp에 대해 지정한 경로 이름이 기본값 대신 사용됩니다.

COMMANDS 옵션에 cat 및 uucp와 같은 위험할 수 있는 명령을 사용하는 경우에는 항상 VALIDATE 옵션을 함께 사용해야 합니다. 파일을 읽거나 쓰는 모든 명령은 UUCP 원격 실행 데몬(uuxqt)에서 실행할 경우 로컬 보안에 잠재적으로 위험할 수 있습니다.



## UUCP VALIDATE 옵션

컴퓨터 보안에 잠재적으로 위험할 수 있는 명령을 지정할 때는 항상 **VALIDATE** 옵션을 **COMMANDS** 옵션과 함께 사용합니다. **VALIDATE**는 **COMMANDS** 옵션의 맨 위에서 보안 레벨을 추가할 뿐이지만 **ALL**보다 안전한 명령 액세스를 여는 방법입니다.

**VALIDATE**를 사용하면 호출 컴퓨터의 호스트 이름과 사용하는 로그인 이름을 교차 확인하여 호출자의 신원을 일정 수준으로 검증할 수 있습니다. 다음 문자열은 **widget** 또는 **gadget** 외의 컴퓨터가 **Uwidget**으로 로그인하려고 하면 연결을 거부하도록 합니다.

```
LOGNAME=Uwidget VALIDATE=widget:gadget
```

**VALIDATE** 옵션을 사용하면 권한 있는 컴퓨터가 UUCP 트랜잭션을 위해 고유한 로그인 및 암호를 사용해야 합니다. 이 검증의 중요한 측면은 이 항목과 연관된 로그인 및 암호가 보호된다는 것입니다. 외부인이 이 정보를 얻게 되면 이 특정 **VALIDATE** 옵션은 더 이상 안전하다고 볼 수 없습니다.

UUCP 트랜잭션에 대한 권한이 있는 로그인 및 암호를 부여할 원격 컴퓨터를 신중히 선택하십시오. 원격 컴퓨터에 파일 액세스 및 원격 실행 기능이 있는 특별 로그인 및 암호를 제공하는 것은 해당 컴퓨터의 모든 사람에게 사용자 컴퓨터의 일반 로그인 및 암호를 제공하는 것과 같습니다. 따라서 원격 컴퓨터에 신뢰할 수 없는 사람이 있으면 해당 컴퓨터에 권한이 있는 로그인과 암호를 제공하지 마십시오.

다음 **LOGNAME** 항목은 **eagle**, **owl** 또는 **hawk**라고 주장하는 원격 컴퓨터 중 하나가 컴퓨터에 로그인하려고 하는 경우 **uucpfriend** 로그인을 사용했어야 한다고 지정합니다.

```
LOGNAME=uucpfriend VALIDATE=eagle:owl:hawk
```

외부인이 **uucpfriend** 로그인과 암호를 얻는 경우 쉽게 가장할 수 있습니다.

하지만 이 항목은 **MACHINE** 항목에만 표시되는 **COMMANDS** 옵션과 어떤 관련이 있습니까? 이 항목은 **MACHINE** 항목 및 **COMMANDS** 옵션을 권한 있는 로그인과 연관된 **LOGNAME** 항목에 연결합니다. 원격 컴퓨터가 로그인한 동안에는 실행 데몬이 실행되지 않기 때문에 이 연결이 필요합니다. 실제로 이 링크는 실행 요청을 보낸 컴퓨터를 알지 못하는 비동기 프로세스입니다. 따라서 사용자 컴퓨터에서 실행 파일의 출처를 어떻게 아는지가 실질적인 질문이 됩니다.

각 원격 컴퓨터는 사용자의 로컬 컴퓨터에 자체의 스푼 디렉토리가 있습니다. 이러한 스푼 디렉토리에는 UUCP 프로그램에만 부여된 쓰기 권한이 있습니다. 원격 컴퓨터의 실행 파일은 사용자 컴퓨터로 전송된 후 스푼 디렉토리에 저장됩니다. **uuxqt** 데몬이 실행되면 스푼 디렉토리 이름을 사용하여 **Permissions** 파일에서 **MACHINE** 항목을 찾고 **COMMANDS** 목록을 가져올 수 있습니다. 또는 컴퓨터 이름이 **Permissions** 파일에 표시되어 있지 않으면 기본 목록이 사용됩니다.

이 예에서는 **MACHINE** 및 **LOGNAME** 항목 간의 관계를 보여줍니다.



```
MACHINE=eagle:owl:hawk REQUEST=yes \
COMMANDS=rmail:/usr/local/rnews \
READ=/ WRITE=/
LOGNAME=uucpz VALIDATE=eagle:owl:hawk \
REQUEST=yes SENDFILES=yes \
READ=/ WRITE=/
```

COMMANDS 옵션의 값은 원격 사용자가 rmail 및 /usr/local/rnews를 실행할 수 있다는 것을 나타냅니다.

첫번째 항목에서는 나열된 컴퓨터 중 하나를 호출하려면 eagle, owl 또는 hawk를 호출한다고 가정해야 합니다. 따라서 eagle, owl 또는 hawk 스푼 디렉토리 중 하나로 저장되는 모든 파일은 이러한 컴퓨터 중 하나에 의해 거기에 저장됩니다. 원격 컴퓨터가 로그인하여 이러한 세 컴퓨터 중 하나라고 주장하는 경우 해당 실행 파일도 권한이 있는 스푼 디렉토리에 저장됩니다. 따라서 컴퓨터에 권한이 있는 로그인 uucpz가 있는지 확인해야 합니다.

## OTHER에 대한 UUCP MACHINE 항목

특정 MACHINE 항목에서 언급하지 않은 원격 컴퓨터에 대해 다른 옵션 값을 지정할 수도 있습니다. 여러 컴퓨터가 사용자 호스트를 호출하고 명령 세트가 가끔 변경되는 경우에 이렇게 할 필요가 생길 수 있습니다. 다음 예에 나온 것처럼 이 항목에서는 컴퓨터 이름에 OTHER 이름을 사용합니다.

```
MACHINE=OTHER \
COMMANDS=rmail:rnews:/usr/local/Photo:/usr/local/xp
```

MACHINE 항목에 사용할 수 있는 다른 모든 옵션도 다른 MACHINE 항목에서 언급하지 않은 컴퓨터에 대해 설정할 수 있습니다.

## UUCP의 MACHINE 및 LOGNAME 항목 결합

공통 옵션이 동일한 경우 MACHINE 및 LOGNAME 항목을 결합할 수 있습니다. 예를 들어 다음에 나오는 두 항목 세트는 동일한 REQUEST, READ 및 WRITE 옵션을 공유합니다.

```
MACHINE=eagle:owl:hawk REQUEST=yes \
READ=/ WRITE=/
```

및

```
LOGNAME=uupz REQUEST=yes SENDFILES=yes \
READ=/ WRITE=/
```

표시된 대로 이들 항목을 병합할 수 있습니다.

```
MACHINE=eagle:owl:hawk REQUEST=yes \
logname=uucpz SENDFILES=yes \
READ=/ WRITE=/
```

MACHINE 및 LOGNAME 항목을 결합하면 Permissions 파일을 보다 효율적으로 관리할 수 있습니다.

## UUCP 전달

일련의 컴퓨터를 통해 파일을 보낼 경우 중개 컴퓨터의 COMMANDS 옵션에 uucp 명령이 포함되어 있어야 합니다. 다음 명령을 입력하는 경우 willow 컴퓨터에서 oak 컴퓨터가 uucp 프로그램을 실행할 수 있도록 허용하는 경우에만 전달 작업이 작동합니다.

```
% uucp sample.txt oak\!willow\!pine\!/usr/spool/uucppublic
```

또한 oak 컴퓨터에서도 사용자 컴퓨터가 uucp 프로그램을 실행할 수 있도록 허용해야 합니다. 마지막 컴퓨터로 지정된 pine 컴퓨터는 전달 작업을 수행하지 않으므로 uucp 명령을 허용하지 않아도 됩니다. 일반적으로는 컴퓨터를 이와 같이 설정하지 않습니다.

## UUCP /etc/uucp/Poll 파일

/etc/uucp/Poll 파일에는 원격 컴퓨터 폴링에 필요한 정보가 포함됩니다. Poll 파일의 각 항목에는 호출할 원격 컴퓨터 이름, 탭 문자 또는 공백, 컴퓨터를 호출할 시간이 차례로 포함됩니다. Poll 파일에 있는 항목의 형식은 다음과 같습니다.

```
sys-name hour ...
```

예를 들어 **eagle 0 4 8 12 16 20** 항목을 사용하면 eagle 컴퓨터를 네 시간마다 폴링할 수 있습니다.

uudemon.poll 스크립트는 Poll 파일을 처리하지만 폴링을 실제로 수행하지는 않습니다. 이 스크립트는 스푼 디렉토리에서 항상 *c.file*이라는 폴링 작업 파일을 설정할 뿐입니다. uudemon.poll 셸 스크립트에서 스케줄러를 시작하고 스케줄러는 스푼 디렉토리의 모든 작업 파일을 검사합니다.

## UUCP /etc/uucp/Config 파일

/etc/uucp/Config 파일을 사용하여 특정 매개변수를 수동으로 대체할 수 있습니다. Config 파일에 있는 각 항목의 형식은 다음과 같습니다.

```
parameter=value
```

구성 가능한 매개변수 이름의 전체 목록은 시스템과 함께 제공되는 **Config** 파일을 참조하십시오.

다음 **Config** 항목은 기본 프로토콜 순서를 **Gge**로 설정하고 **G** 프로토콜 기본값을 7개 창과 512바이트 패킷으로 변경합니다.

Protocol=G(7,512)ge

## UUCP/etc/uucp/Grades 파일

**/etc/uucp/Grades** 파일에는 작업을 원격 컴퓨터의 대기열에 넣는 데 사용할 수 있는 작업 등급에 대한 정의가 포함됩니다. 이 파일에는 각 작업 등급에 대한 사용 권한도 포함됩니다. 이 파일의 각 항목은 사용자가 작업을 대기열에 넣을 수 있도록 하는 관리자 정의 작업 등급에 대한 정의를 나타냅니다.

**Grades** 파일에 있는 각 항목의 형식은 다음과 같습니다.

*User-job-grade System-job-grade Job-size Permit-type ID-list*

각 항목에는 공백으로 구분된 필드가 포함되어 있습니다. 항목의 마지막 필드는 역시 공백으로 구분된 하위 필드로 구성되어 있습니다. 한 항목이 둘 이상의 행을 차지하는 경우 백슬래시를 사용하여 항목을 다음 행으로 연결할 수 있습니다. 주석 행은 파운드 기호(#)로 시작하며 전체 행을 사용합니다. 빈 행은 항상 무시됩니다.

### UUCP User-job-grade 필드

이 필드에는 최대 64자로 이루어진 관리자 정의 사용자 작업 등급 이름이 포함됩니다.

### UUCP System-job-grade 필드

이 필드에는 *User-job-grade*가 매핑되는 단일 문자 작업 등급이 포함됩니다. 유효한 문자 목록은 A-Z, a-z이며, 우선 순위는 A가 가장 높고 z가 가장 낮습니다.

### 사용자 및 시스템 작업 등급 간 관계

사용자 작업 등급은 둘 이상의 시스템 작업 등급에 바인딩될 수 있습니다. **Grades** 파일에서 사용자 작업 등급 항목을 순서대로 검색합니다. 따라서 최대 작업 크기에 대한 제한에 따라 여러 시스템 작업 등급 항목이 나열되어야 합니다.

사용자 작업 등급에 대한 최대 개수는 없지만 허용되는 최대 시스템 작업 등급 수는 52개입니다. 둘 이상의 *User-job-grade*를 하나의 *System-job-grade*에 매핑할 수 있지만 각 *User-job-grade*는 파일에서 별도의 행에 있어야 하기 때문입니다. 다음은 예입니다.

```
mail N Any User Any netnews N Any User Any
```

이 구성이 Grades 파일에 있는 경우 이러한 두 *User-job-grade* 필드는 동일한 *System-job-grade*를 공유합니다. *Job-grade*에 대한 사용 권한은 *System-job-grade*가 아닌 *User-job-grade*와 연결되므로 두 *User-job-grade*는 동일한 *System-job-grade*를 공유하지만 사용 권한 세트는 서로 다를 수 있습니다.

## 기본 등급

기본 *User-job-grade*와 시스템 작업 등급의 바인딩을 정의할 수 있습니다. 키워드 기본값을 Grades 파일의 *User-job-grade* 필드에 있는 사용자 작업 등급 및 이 등급이 바인딩된 시스템 작업 등급으로 사용해야 합니다. 모든 사용자 및 모든 크기의 작업을 이 등급의 대기열에 넣을 수 있도록 Restrictions 및 ID 필드는 Any로 정의해야 합니다. 다음은 예입니다.

```
default a Any User Any
```

기본 사용자 작업 등급을 정의하지 않으면 내장 기본 등급인 z가 사용됩니다. 제한 필드 기본값은 Any이므로 기본 등급 항목이 여러 개인지는 확인하지 않습니다.

## UUCP Job-size 필드

이 필드는 대기열에 입력할 수 있는 최대 작업 크기를 지정합니다. *Job-size*는 바이트 단위로 측정되며 다음 목록에 설명된 옵션 목록일 수 있습니다.

<i>nnnn</i>	이 작업 등급의 최대 작업 크기를 지정하는 정수입니다.
<i>nK</i>	킬로바이트 수를 나타내는 10진수입니다(K는 킬로바이트의 약어).
<i>nM</i>	메가바이트 수를 나타내는 10진수입니다(M는 메가바이트의 약어).
Any	최대 작업 크기가 없음을 지정하는 키워드입니다.

다음은 몇 가지 예입니다.

- 5000은 5000바이트를 나타냅니다.
- 10K는 10KB를 나타냅니다.
- 2M은 2MB를 나타냅니다.

## UUCP Permit-type 필드

이 필드에는 ID 목록을 해석할 방법을 나타내는 키워드가 포함됩니다. 다음 표에서는 키워드와 그 의미가 나와 있습니다.

표 26-5 Permit-type 필드

키워드	ID 목록 내용
User	이 작업 등급을 사용할 수 있는 사용자의 로그인 이름
Non-user	이 작업 등급을 사용할 수 없는 사용자의 로그인 이름
Group	구성원이 이 그룹을 사용할 수 있는 그룹 이름
Non-group	구성원이 이 작업 등급을 사용할 수 없는 그룹 이름

## UUCP ID-list 필드

이 필드에는 이 작업 등급에 대한 대기열 할당이 허용되거나 정의된 로그인 이름 또는 그룹 이름의 목록이 포함됩니다. 이름 목록은 공백으로 구분되며 개행 문자로 끝납니다. Any 키워드를 사용하면 누구나 이 작업 등급의 대기열에 넣을 수 있도록 지정할 수 있습니다.

## 기타 UUCP 구성 파일

이 절에서는 자주 수정되지 않으며 UUCP 기능 사용에 영향을 주는 세 가지 파일에 대해 설명합니다.

### UUCP /etc/uucp/Devconfig 파일

/etc/uucp/Devconfig 파일을 사용하여 uucp 또는 cu와 같은 서비스별로 장치를 구성할 수 있습니다. Devconfig 항목은 특정 장치에 사용되는 STREAMS 모듈을 정의합니다. 이러한 항목의 형식은 다음과 같습니다.

```
service=x device=y push=z[:z...]
```

x는 cu, uucico 또는 두 서비스 모두일 수 있으며 각 서비스는 콜론으로 구분됩니다. y는 네트워크 이름이며 Devices 파일의 항목과 일치해야 합니다. z는 STREAMS 모듈의 이름으로 대체되며, 그 순서는 모듈이 스트림에 푸시되는 순서와 같습니다. cu 및 uucp 서비스에 대해 모듈과 장치를 다르게 정의할 수 있습니다.

다음 항목은 STARLAN 네트워크에 적용되며 이 파일에서 가장 일반적으로 사용됩니다.

```
service=cu      device=STARLAN    push=ntty:tirdwr
service=uucico  device=STARLAN    push=ntty:tirdwr
```

이 예에서는 ntty, tirdwr을 순서대로 푸시합니다.

## UUCP /etc/uucp/Limits 파일

/etc/uucp/Limits 파일은 uucp 네트워킹에서 실행 중인 동시 uucico, uuxqt 및 uusched의 최대 수를 제어합니다. 대부분의 경우 기본값을 사용 가능하며 변경할 필요가 없습니다. 변경하려면 텍스트 편집기를 사용하십시오.

Limits 파일의 형식은 다음과 같습니다.

```
service=x max= y:
```

$x$ 는 uucico, uuxqt 또는 uusched일 수 있고  $y$ 는 해당 서비스에 허용되는 한계입니다. 필드는 순서에 상관없이 없고 소문자입니다.

다음 항목은 Limits 파일에서 가장 일반적으로 사용됩니다.

```
service=uucico max=5
service=uuxqt max=5
service=uusched max=2
```

이 예는 컴퓨터에서 uucico 5개, uuxqt 5개 및 uusched 2개를 실행할 수 있게 합니다.

## UUCP remote.unknown 파일

통신 기능 사용에 영향을 주는 다른 파일은 remote.unknown 파일입니다. 이 파일은 Systems 파일에 없는 컴퓨터가 대화를 시작할 때 실행되는 이진 프로그램입니다. 이 프로그램은 대화 시도를 기록하고 연결을 끊습니다.



**주의** - remote.unknown 파일의 사용 권한을 변경하여 파일을 실행할 수 없도록 하는 경우 시스템에서 모든 시스템의 연결을 허용합니다.

이 프로그램은 Systems에 없는 컴퓨터가 대화를 시작하면 실행됩니다. 이 프로그램은 대화를 시도했지만 연결하지 못한 경우를 기록합니다. 이 파일의 사용 권한을 변경하여 파일을 실행할 수 없도록 하는 경우(chmod 000 remote.unknown) 시스템에서 모든 대화 요청을 허용합니다. 이러한 변경은 단순하지 않습니다. 변경할 충분한 이유가 있어야 합니다.

## UUCP 관리 파일

다음에서는 UUCP 관리 파일에 대해 설명합니다. 이러한 파일은 스푼 디렉토리에 만들며 장치를 잠그거나, 임시 데이터를 보관하거나, 원격 전송 또는 실행에 대한 정보를 보관하는 데 사용됩니다.

- *Temporary data files(TM)* - 이러한 데이터 파일은 다른 컴퓨터에서 파일을 받을 때 UUCP 프로세스가 스푼 디렉토리 `/var/spool/uucp/x` 아래에 만듭니다. `x` 디렉토리의 이름은 파일을 보내는 원격 컴퓨터와 같습니다. 임시 데이터 파일의 이름은 다음 형식을 사용합니다.

*TM. pid.ddd*

*pid*는 프로세스 ID이고 *ddd*는 0에서 시작하는 순차적 세 자리 숫자입니다.

전체 파일을 받으면 *TM. pid.ddd* 파일은 전송의 원인이 된 *C.sysnxxxx* 파일(나중에 설명)에 지정된 경로 이름으로 이동됩니다. 처리가 비정상적으로 종료되면

*TM. pid.ddd* 파일이 `x` 디렉토리에 유지될 수 있습니다. 이러한 파일은 `uucleanup`에 의해 자동으로 제거됩니다.

- *Lock files(LCK)* - 잠금 파일은 사용 중인 각 장치의 `/var/spool/locks` 디렉토리에 만들어집니다. 잠금 파일은 중복된 대화와 동일한 호출 장치를 여러 번 사용하려는 시도를 방지합니다. 다음 표에서는 여러 유형의 UUCP 잠금 파일을 보여줍니다.

표 26-6 UUCP 잠금 파일

파일 이름	설명
LCK.sys	<i>sys</i> 는 파일을 사용 중인 컴퓨터의 이름을 나타냅니다.
LCK.dev	<i>dev</i> 는 파일을 사용 중인 장치의 이름을 나타냅니다.
LCK.LOG	<i>LOG</i> 는 잠긴 UUCP 로그 파일을 나타냅니다.

컴퓨터가 충돌하는 경우와 같이 통신 링크가 예기치 않게 끊기는 경우 이러한 파일이 스푼 디렉토리에 남을 수 있습니다. 부모 프로세스가 더 이상 활성 상태가 아니면 잠금 파일은 무시(제거)됩니다. 잠금 파일에는 잠금을 만든 프로세스의 프로세스 ID가 포함됩니다.

- *Work file(C.)* - 작업 파일은 파일 전송, 원격 명령 실행 등의 작업을 원격 컴퓨터의 대기열에 넣은 경우 스푼 디렉토리에 만들어집니다. 작업 파일의 이름은 다음 형식을 사용합니다.

*C.sysnxxxx*

*sys*는 원격 컴퓨터의 이름이고, *n*은 작업의 등급(우선 순위)을 나타내는 ASCII 문자이고, *xxxx*는 UUCP에서 지정하는 4자리 작업 시퀀스 번호입니다. 작업 파일에는 다음 정보가 포함됩니다.

- 보내거나 요청할 파일의 전체 경로 이름
- 대상, 사용자 또는 파일 이름의 전체 경로 이름
- 사용자 로그인 이름
- 옵션 목록
- 스푼 디렉토리에 있는 관련 데이터 파일의 이름. `uucp -C` 또는 `uuto -p` 옵션을 지정한 경우 임시 이름(`D.0`)이 사용됩니다.

- 소스 파일의 모드 비트
- 전송 완료 시 알림을 받는 원격 사용자의 로그인 이름
- *Data file(D.)* - 데이터 파일은 명령줄에 지정하여 소스 파일을 스푼 디렉토리로 복사할 때 만들어집니다. 데이터 파일의 이름은 다음 형식을 사용합니다.  
*D.systmxxxxyyy* - *systm*은 원격 컴퓨터 이름에서 처음 5개 문자입니다. *xxxx*는 *uucp*에서 지정하는 4자리 작업 시퀀스 번호입니다. 4자리 작업 시퀀스 번호 다음에서 후속 번호가 올 수 있습니다. *yyy*는 작업(C.) 파일에 대해 여러 *D.* 파일을 만드는 경우에 사용됩니다.
- *X.(execute file)* - 실행 파일은 원격 명령 실행 전에 스푼 디렉토리에 만들어집니다. 실행 파일의 이름은 다음 형식을 사용합니다.  
*X.synxxxx*  
*syn*는 원격 컴퓨터의 이름이고, *n*은 작업의 등급(우선 순위)을 나타내는 문자이고, *xxxx*는 *UUCP*에서 지정하는 4자리 작업 시퀀스 번호입니다. 실행 파일에는 다음 정보가 포함됩니다.
  - 요청자의 로그인 및 컴퓨터 이름
  - 실행해야 하는 파일의 이름
  - 명령 문자열의 표준 입력으로 사용할 입력
  - 명령 실행의 표준 출력을 받을 컴퓨터 및 파일 이름
  - 명령 문자열
  - 반환 상태 요청에 대한 옵션 행

## UUCP 오류 메시지

이 절에서는 UUCP와 관련된 오류 메시지를 나열합니다.

### UUCP ASSERT 오류 메시지

다음 표에서는 ASSERT 오류 메시지를 나열합니다.

표 26-7 ASSERT 오류 메시지

오류 메시지	설명 또는 작업
CAN'T OPEN	<code>open()</code> 또는 <code>fopen()</code> 이 실패했습니다.
CAN'T WRITE	<code>write()</code> , <code>fwrite()</code> , <code>fprint()</code> 또는 유사한 명령이 실패했습니다.
CAN'T READ	<code>read()</code> , <code>fgets()</code> 또는 유사한 명령이 실패했습니다.
CAN'T CREATE	<code>creat()</code> 호출이 실패했습니다.
CAN'T ALLOCATE	동적 할당이 실패했습니다.



표 26-7 ASSERT 오류 메시지 (계속)

오류 메시지	설명 또는 작업
CAN'T LOCK	LCK(잠금) 파일을 만들지 못했습니다. 경우에 따라 치명적인 오류일 수 있습니다.
CAN'T STAT	stat() 호출이 실패했습니다.
CAN'T CHMOD	chmod() 호출이 실패했습니다.
CAN'T LINK	link() 호출이 실패했습니다.
CAN'T CHDIR	chdir() 호출이 실패했습니다.
CAN'T UNLINK	unlink() 호출이 실패했습니다.
WRONG ROLE	내부 논리 문제입니다.
CAN'T MOVE TO CORRUPTDIR	일부 잘못된 C. 또는 X. 파일을 /var/spool/uucp/.Corrupt 디렉토리로 이동하지 못했습니다. 디렉토리가 없거나 모드 또는 소유자가 잘못되었습니다.
CAN'T CLOSE	close() 또는 fclose() 호출이 실패했습니다.
FILE EXISTS	C. 또는 D. 파일을 생성하려고 했지만 파일이 이미 있습니다. 일반적으로 소프트웨어 오류를 나타내는 시퀀스 파일 액세스 관련 문제가 발생하는 경우 이 오류가 발생합니다.
NO uucp SERVICE NUMBER	TCP/IP 호출을 시도했지만 /etc/services에서 UUCP에 대한 항목이 없습니다.
BAD UID	사용자 ID가 암호 데이터베이스에 없습니다. 이름 서비스 구성을 확인하십시오.
BAD LOGIN_UID	이전 설명과 동일합니다.
BAD LINE	Devices 파일에 잘못된 행이 있습니다. 하나 이상의 행에 인수가 부족합니다.
SYSLST OVERFLOW	genome.c의 내부 테이블이 오버플로우되었습니다. 단일 작업에서 30개 이상의 시스템과 대화하려고 했습니다.
TOO MANY SAVED C FILES	이전 설명과 동일합니다.
RETURN FROM fixline ioctl	실패하면 안 되는 ioctl(2)이 실패했습니다. 시스템 드라이버 문제가 발생했습니다.
BAD SPEED	Devices 또는 Systems 파일(Class 또는 Speed 필드)에 잘못된 회선 속도가 표시됩니다.
BAD OPTION	Permissions 파일에 잘못된 행 또는 옵션이 있습니다. 이 오류는 바로 수정해야 합니다.
PKCGET READ	원격 컴퓨터가 행업된 것 같습니다. 아무 작업도 필요하지 않습니다.
PKXSTART	원격 컴퓨터가 복구할 수 있는 방식으로 중단되었습니다. 이 오류는 일반적으로 무시할 수 있습니다.
TOO MANY LOCKS	내부 문제가 발생했습니다. 시스템 공급업체에 문의하십시오.
XMV ERROR	일부 파일 또는 디렉토리에 문제가 발생했습니다. 스푼 디렉토리가 원인일 수 있으므로 이 프로세스를 시도하기 전에 대상 모드를 확인해야 합니다.
CAN'T FORK	fork 및 exec를 수행하지 못했습니다. 현재 작업은 손실되지 않으며 나중에 시도됩니다(uuxqt). 아무 작업도 필요하지 않습니다.

# UUCP STATUS 오류 메시지

다음 표에서는 일반적인 STATUS 오류 메시지를 나열합니다.

표 26-8 UUCPSTATUS 메시지

오류 메시지	설명/작업
OK	상태가 적합합니다.
NO DEVICES AVAILABLE	호출에 사용할 수 있는 장치가 현재 없습니다. 특정 시스템의 <b>Devices</b> 파일에 유효한 장치가 있는지 확인하십시오. 시스템 호출에 사용할 장치가 있는지 <b>Systems</b> 파일에 있는지 확인하십시오.
WRONG TIME TO CALL	<b>Systems</b> 파일에 지정된 시간과 다른 시간에 시스템을 호출했습니다.
TALKING	설명이 필요하지 않습니다.
LOGIN FAILED	특정 컴퓨터에 로그인하지 못했습니다. 로그인 또는 암호가 잘못되었거나, 번호가 잘못되었거나, 컴퓨터가 느리거나, <b>Dialer-Token-Pairs</b> 스크립트를 실행하지 못했기 때문일 수 있습니다.
CONVERSATION FAILED	성공적인 시작 후 대화에 실패했습니다. 일반적으로 한 쪽이 다운되었거나, 프로그램이 중단되었거나, 회선(링크)이 끊긴 것을 나타냅니다.
DIAL FAILED	원격 컴퓨터가 응답하지 않습니다. 전화 걸기가 잘못되었거나 전화 번호가 틀릴 수 있습니다.
BAD LOGIN/MACHINE COMBINATION	<b>Permissions</b> 파일과 일치하지 않는 로그인/컴퓨터 이름을 사용하여 컴퓨터를 호출했습니다. 이 오류는 가장하려는 시도일 수 있습니다.
DEVICE LOCKED	사용할 호출 장치가 현재 잠겨 있고 다른 프로세스에서 사용되고 있습니다.
ASSERT ERROR	ASSERT 오류가 발생했습니다. <b>/var/uucp/.Admin/errors</b> 파일의 오류 메시지를 확인하고 <a href="#">560 페이지 “UUCP ASSERT 오류 메시지”</a> 절을 참조하십시오.
SYSTEM NOT IN Systems FILE	시스템이 <b>Systems</b> 파일에 없습니다.
CAN'T ACCESS DEVICE	시도된 장치가 없거나 모드가 잘못되었습니다. <b>Systems</b> 및 <b>Devices</b> 파일에서 적절한 항목을 확인하십시오.
DEVICE FAILED	장치를 열 수 없습니다.
WRONG MACHINE NAME	호출된 컴퓨터가 예상과 다른 이름을 보고합니다.
CALLBACK REQUIRED	호출된 컴퓨터가 사용자 컴퓨터를 호출해야 합니다.
REMOTE HAS A LCK FILE FOR ME	원격 컴퓨터에 사용자 컴퓨터의 <b>LCK</b> 파일이 있습니다. 호출된 컴퓨터가 사용자 컴퓨터를 호출하려고 할 수 있습니다. 원격 컴퓨터에 이전 버전의 UUCP가 있는 경우 사용자 컴퓨터와 대화하던 프로세스가 실패하여 <b>LCK</b> 파일이 남았을 수 있습니다. 원격 컴퓨터가 최신 버전의 UUCP가 있고 사용자 컴퓨터와 통신하고 있지 않은 경우 <b>LCK</b> 파일을 사용하는 프로세스가 중단된 것입니다.
REMOTE DOES NOT KNOW ME	원격 컴퓨터의 <b>Systems</b> 파일에 사용자 컴퓨터의 노드 이름이 없습니다.

표 26-8 UUCP STATUS 메시지 (계속)

오류 메시지	설명/작업
REMOTE REJECT AFTER LOGIN	컴퓨터가 로그인할 때 사용한 로그인이 원격 컴퓨터가 예상한 것과 일치하지 않습니다.
REMOTE REJECT, UNKNOWN MESSAGE	원격 컴퓨터가 알 수 없는 이유로 사용자 컴퓨터와의 통신을 거부했습니다. 원격 컴퓨터에서 표준 버전의 UUCP를 실행하고 있지 않을 수 있습니다.
STARTUP FAILED	로그인에 성공했지만 초기 핸드셰이크가 실패했습니다.
CALLER SCRIPT FAILED	이 오류는 일반적으로 DIAL FAILED와 동일합니다. 그러나 이 오류가 자주 발생하는 경우 Dialers 파일의 호출자 스크립트가 원인일 수 있습니다. Uutry를 사용하여 확인하십시오.

## UUCP 숫자 오류 메시지

다음 표에서는 /usr/include/sysexits.h 파일에서 생성하는 오류 상태 메시지의 종료 코드 번호를 나열합니다. 일부는 현재 uucp에서 사용되지 않습니다.

표 26-9 번호별 UUCP 오류 메시지

메시지 번호	설명	의미
64	오류 메시지의 기준 값	오류 메시지가 이 값에서 시작됩니다.
64	명령줄 사용 오류	명령을 잘못 사용했습니다. 예를 들어 잘못된 인수 수, 잘못된 플래그 또는 잘못된 구문을 사용했습니다.
65	데이터 형식 오류	입력 데이터가 잘못되었습니다. 이 데이터 형식은 사용자 데이터에만 사용해야 하며 시스템 파일에 사용하면 안 됩니다.
66	입력을 열 수 없음	시스템 파일이 아닌 입력 파일이 없거나 읽지 못했습니다. 이 문제는 메일러의 “No message(메시지가 없습니다)”와 같은 오류도 포함할 수 있습니다.
67	알 수 없는 주소	지정된 사용자가 없습니다. 이 오류는 메일 주소 또는 원격 로그인에 사용할 수 있습니다.
68	알 수 없는 호스트 이름	호스트가 없습니다. 이 오류는 메일 주소 또는 네트워크 요청에서 사용됩니다.
69	사용할 수 없는 서비스	서비스를 사용할 수 없습니다. 지원 프로그램 또는 파일이 없는 경우 오류가 발생할 수 있습니다. 또한 이 메시지는 일부 서비스가 작동하지 않지만 현재 그 이유를 알 수 없음을 나타내기도 합니다.
70	내부 소프트웨어 오류	내부 소프트웨어 오류가 감지되었습니다. 이 오류는 가급적 운영 체제와 관련된 없는 오류로 제한됩니다.
71	시스템 오류	운영 체제 오류가 감지되었습니다. 이 오류는 “포크할 수 없음”, “파이프를 만들 수 없음”과 같은 상태에 사용하기 위한 것입니다. 예를 들어 이 오류에는 passwd 파일에 없는 사용자의 getuid 반환이 포함됩니다.

표 26-9 번호별 UUCP 오류 메시지 (계속)

메시지 번호	설명	의미
72	중요한 OS 파일 누락	/etc/passwd 또는 /var/admin/utmpx와 같은 시스템 파일이 없거나 열 수 없거나 구문 오류 같은 오류가 있습니다.
73	출력 파일을 만들 수 없음	사용자 지정 출력 파일을 만들 수 없습니다.
74	입출력 오류	일부 파일에서 I/O 수행 시 오류가 발생했습니다.
75	임시 오류. 사용자 재시도 유도	실제 오류가 아닌 임시 오류입니다. sendmail에서는 예를 들어 메일러에서 연결을 만들 수 없어 요청을 나중에 다시 시도해야 함을 나타냅니다.
76	프로토콜의 원격 오류	원격 시스템이 프로토콜 교환 중 "가능하지 않은" 항목을 반환했습니다.
77	사용 권한 거부	권한이 부족하여 작업을 수행할 수 없습니다. 이 메시지는 NOINPUT 또는 CANTCREAT를 사용해야 하는 시스템 문제에 적용되지 않고 상위 레벨 권한에 적용됩니다. 예를 들어 kre는 이 메시지를 사용하여 메일을 보낼 수 있는 학생을 제한할 수 있습니다.
78	구성 오류	구성에서 오류가 감지되었습니다.
79	항목을 찾을 수 없음	항목을 찾지 못했습니다.
79	나열된 최대 값	오류 메시지의 최상위 값입니다.

## 제 6 부

# 원격 시스템 작업 항목

이 절에서는 Oracle Solaris 환경에서 FTP 서버를 관리하고 원격 시스템에 액세스하기 위한 지침을 제공합니다.



## 원격 시스템 작업(개요)

---

이 장에는 원격 파일 작업에 대한 정보가 나옵니다.

- 567 페이지 “FTP 서버란?”
- 567 페이지 “원격 시스템이란?”
- 568 페이지 “이 릴리스의 FTP 서버 정보”
- 568 페이지 “표준 ProFTPD와의 차이점”
- 568 페이지 “ProFTPD 구성 요소”

### FTP 서버란?

FTP(File Transfer Protocol) 서버는 ProFTPD 프로젝트를 기반으로 합니다. 이 소프트웨어는 대규모 FTP 사이트의 기본 표준이며 인터넷을 통한 벌크 데이터의 배포에 널리 사용되는 FTP 프로토콜의 서버측을 구현합니다. ProFTPD 프로젝트에 대한 자세한 내용은 <http://www.proftpd.org>를 참조하십시오.

### 원격 시스템이란?

이 장에서 **원격 시스템**은 임의의 물리적 네트워크를 사용하여 로컬 시스템에 연결되어 있고 TCP/IP 통신을 위해 구성된 워크스테이션 또는 서버입니다.

Oracle Solaris 릴리스를 실행하는 시스템에서 TCP/IP 구성은 시작될 때 자동으로 설정됩니다. 자세한 내용은 **Oracle Solaris 관리: IP 서비스**를 참조하십시오.

# 이 릴리스의 FTP 서버 정보

wu-ftpd 배포를 기반으로 하는 이전 FTP 서버가 **proftpd** 서버로 대체되었습니다. 이전 서비스에서 새 서비스로의 구성 정보 마이그레이션은 `/usr/share/doc/proftpd/proftpd_migration.txt`에 설명되어 있습니다.

## 표준 ProFTPD와의 차이점

다음 목록에서는 ProFTPD의 Oracle Solaris 11 구현에서 다른 항목에 대해 설명합니다.

- ProFTPD의 Oracle Solaris 버전은 독립형 모드로만 실행됩니다.
- 이 릴리스에서는 `logrotate.d` 명령을 사용하여 서비스 로그를 회전시키지 않습니다.

## ProFTPD 구성 요소

다음 절에서는 ProFTPD 서비스의 명령, 파일 및 중요한 기타 구성 요소에 대한 정보를 제공합니다.

## ProFTPD 명령

다음 표에서는 ProFTPD 서비스와 연관된 명령 및 데몬에 대해 설명합니다.

표 27-1 ProFTPD 명령

파일 이름	기능
<code>/usr/bin/ftp</code>	ProFTPD 서비스에 대한 사용자 인터페이스를 제공합니다. 자세한 내용은 <code>ftp(1)</code> 매뉴얼 페이지를 참조하십시오.
<code>/usr/bin/ftpcount</code>	서버뿐 아니라 가상 호스트 또는 익명 구성당 현재 연결 수를 표시합니다. 자세한 내용은 <code>ftpcount(1)</code> 매뉴얼 페이지를 참조하십시오.
<code>/usr/bin/ftpdctl</code>	<b>proftpd</b> 서비스 데몬을 제어합니다. 자세한 내용은 <code>ftpdctl(8)</code> 매뉴얼 페이지를 참조하십시오.
<code>/usr/bin/ftptop</code>	FTP 세션의 현재 상태를 계속 업데이트되는 형식으로 표시합니다. 자세한 내용은 <code>ftptop(1)</code> 매뉴얼 페이지를 참조하십시오.
<code>/usr/bin/ftpwho</code>	모든 활성 <b>proftpd</b> 연결에 대한 프로세스 정보 및 각 서버에 연결된 모든 사용자의 수를 표시합니다. 자세한 내용은 <code>ftpwho(1)</code> 매뉴얼 페이지를 참조하십시오.
<code>/usr/sbin/ftprestart</code>	<code>ftpshtut - R</code> 명령을 사용하여 FTP 연결을 다시 시작합니다. 자세한 내용은 <code>ftpshtut(8)</code> 매뉴얼 페이지를 참조하십시오.



표 27-1 ProFTPD 명령 (계속)

파일 이름	기능
/usr/sbin/ftpscrub	스코어보드 파일에서 더 이상 라이브 상태가 아닌 프로세스를 요구 시 제거합니다. 자세한 내용은 ftpscrub(8) 매뉴얼 페이지 및 <a href="http://www.proftpd.org/docs/howto/Scoreboard.html">http://www.proftpd.org/docs/howto/Scoreboard.html</a> 을 참조하십시오.
/usr/sbin/ftpsht	지정된 시간에 FTP 연결을 종료합니다. 자세한 내용은 ftpshut(8) 매뉴얼 페이지를 참조하십시오.
/usr/lib/inet/proftpd	FTP 서비스를 제공합니다. 자세한 내용은 proftpd (8) 매뉴얼 페이지를 참조하십시오.

## ProFTPD 파일

시스템에서 ProFTPD 서비스를 지원하려면 여러 파일이 필요합니다. 다음 표에서는 이러한 여러 파일과 그 기능을 나열합니다.

표 27-2 ProFTPD 파일

파일 이름	기능
~/.ftpaccess	각 가상 호스트에 대한 추가 제어 방식을 제공합니다. 이 파일은 가상 호스트의 홈 디렉토리에 있어야 합니다. 자세한 내용은 <a href="http://www.proftpd.org/localsite/Userguide/linked/x1021.html">http://www.proftpd.org/localsite/Userguide/linked/x1021.html</a> 을 참조하십시오.
/etc/proftpd.conf	ProFTPD 서비스가 작동하려면 정의해야 하는 구성 매개변수를 대부분 포함합니다.
/etc/shutmsg	ftpshut 명령에서 사용하는 정보를 포함합니다.
/etc/ftpd/ftpusers	FTP 로그인 권한이 허용되지 않을 사용자를 나열합니다. wu-ftp 서비스와의 역방향 호환 가능성을 위해 제공됩니다.
/var/log/xferlog	ProFTPD에 대한 로그 정보를 나열합니다.
/var/run/proftpd.scoreboard	ftpcount, ftpdtop 및 ftpwho와 같은 명령에서 사용되는 각 현재 세션에 대한 추적 정보를 포함합니다. 자세한 내용은 <a href="http://www.proftpd.org/docs/howto/Scoreboard.html">http://www.proftpd.org/docs/howto/Scoreboard.html</a> 을 참조하십시오.

## ProFTPD 사용자

ftp 사용자와 ftp 그룹은 ProFTPD 설치 프로세스에서 만듭니다. ProFTPD 서버는 이러한 자격 증명으로 실행됩니다.



## FTP 서버 관리(작업)

이 장에는 FTP 서버를 설정 및 관리하는 작업이 포함되어 있습니다.

- 571 페이지 “FTP 서버 관리(작업 맵)”
- 572 페이지 “FTP 서버 관리(작업)”

### FTP 서버 관리(작업 맵)

다음 표에는 FTP 서버를 사용하기 위해 따라야 하는 절차가 설명되어 있습니다.

표 28-1 작업 맵: FTP 서버 관리

작업	설명	수행 방법
FTP 서버를 시작합니다.	<code>proftpd.conf</code> 파일을 변경한 후에 이 절차를 따릅니다.	572 페이지 “SMF를 사용하여 FTP 서버를 시작하는 방법”
FTP 서버를 중지합니다.	<code>proftpd.conf</code> 파일을 변경하기 전에 이 절차를 따릅니다.	572 페이지 “SMF를 사용하여 FTP 서버를 종료하는 방법”
FTP 서버 연결을 종료합니다.	파일 시스템 유지 관리 중이나 서비스를 중지하지 않아도 되지만 파일에 대한 액세스는 거부해야 하는 기타 이벤트 중 <code>/etc/shutmsg</code> 파일을 사용하고 <code>ftpsht</code> 를 실행하여 FTP 연결을 종료합니다.	572 페이지 “FTP 연결을 종료하는 방법”
FTP 서버를 재구성합니다.	<code>proftpd.conf</code> 파일을 변경할 때 이 절차를 따릅니다.	573 페이지 “ProFTPD 구성을 변경하는 방법”

## FTP 서버 관리(작업)

다음 절차에서는 FTP 서버를 시작 및 중지하는 방법, FTP 연결을 사용 안함으로 설정하는 방법 및 ProFTPD 구성 파일을 변경하는 방법을 보여줍니다.

### ▼ SMF를 사용하여 FTP 서버를 시작하는 방법

- 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 FTP 서버를 시작합니다.

```
# svcadm enable network/ftp
```

### ▼ SMF를 사용하여 FTP 서버를 종료하는 방법

- 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

- 2 FTP 서버를 중지합니다.

```
# svcadm disable network/ftp
```

### ▼ FTP 연결을 종료하는 방법

ftpshtut(8) 명령은 특정 시간에 FTP 서버를 종료합니다. FTP 지원만 중지하고 데몬은 중지하지 않으려면(클라이언트에 서비스 사용 불가능 상태를 보고할 수 있도록) 이 절차를 사용하십시오. ftpshut 명령은 연결을 차단하고 현재 연결을 중지하지만 서버 데몬 자체는 종료하지 않습니다.

ftpshtut을 실행하면 종료가 발생하는 때, 새 연결이 거부되는 시점 및 기존 연결이 삭제되는 때를 지정하는 명령줄 옵션에서 파일이 생성됩니다. 사용자에게는 이 정보를 바탕으로 서버 종료 사실이 알려집니다. ftpshut을 통해 만들어지는 파일의 위치는 /etc/shutmsg입니다.

- 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

## 2 ftpshut 명령을 실행합니다.

```
ftpshut [ -l min] [ -d min] time [warning-message...]
```

ftpshut FTP 서버가 종료되고 있다는 사실을 사용자에게 알리기 위한 절차를 제공하는 명령입니다.

-l FTP 서버에 대한 새로운 연결이 거부되는 시간을 조정하는 데 사용되는 플래그입니다.

-d FTP 서버에 대한 기존 연결이 해제되는 시간을 조정하는 데 사용되는 플래그입니다.

time 종료 시간으로, 즉시 종료를의 경우 now라는 단어로 지정되고 이후 종료를의 경우 두 형식(+ *number* 또는 *HHMM*) 중 하나로 지정됩니다.

[warning-message...] 종료 알림 메시지입니다. 자세한 내용은 ftpshut(8) 매뉴얼 페이지를 참조하십시오.

## 3 파일에 대한 액세스를 복원합니다.

ftpprestart 명령을 사용하여 FTP 서버에 대한 연결을 다시 시작합니다. 자세한 내용은 ftpshut(8) 및 ftpprestart(8)를 참조하십시오.

# ▼ ProFTPD 구성을 변경하는 방법

대부분의 구성 변형은 /etc/proftpd.conf 파일을 변경하여 발생합니다. 이 파일을 변경할 때는 다음 단계를 사용하십시오.

## 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스](#)의 “관리 권한을 얻는 방법”을 참조하십시오.

## 2 구성 파일을 변경합니다.

구성 파일에 추가할 정보에 대한 제안 사항은 아래의 간단한 예를 참조하십시오.

## 3 FTP 서버를 다시 시작합니다.

```
# svcadm restart network/ftp
```

### 예 28-1 가상 호스트를 위한 ProFTPD 구성 파일 변경 사항

고정 IP 주소를 사용하는 가상 호스트의 경우 다음 지시어를 사용하십시오. 필요한 경우 여러 IP 주소를 공백으로 구분하여 추가할 수 있습니다.

```
<VirtualHost 10.0.0.1>  
  ServerName "My virtual FTP server"  
</VirtualHost>
```

### 예 28-2 익명 액세스를 위한 ProFTPD 구성 파일 변경 사항

사이트에 대한 익명 ftp 액세스를 제공하려면 다음 지시어를 사용하십시오.

```
# Deny login access  
<Limit LOGIN>  
  DenyAll  
</Limit>  
  
<Anonymous ~ftp>  
  
# Allow anonymous logins  
<Limit LOGIN>  
  AllowAll  
</Limit> ....  
</Anonymous>
```

## 원격 시스템 액세스(작업)

이 장에서는 원격 시스템에 로그인하고 해당 파일을 사용하는 데 필요한 모든 작업에 대해 설명합니다. 본 장에 포함된 단계별 지침 목록은 다음과 같습니다.

- 575 페이지 “원격 시스템 액세스(작업 맵)”
- 576 페이지 “원격 시스템에 로그인(rlogin)”
- 583 페이지 “원격 시스템에 로그인(ftp)”
- 589 페이지 “rcp를 사용한 원격 복사”

## 원격 시스템 액세스(작업 맵)

이 장에서는 원격 시스템에 로그인하고 해당 시스템에서 파일을 복사하기 위해 다음 표에 설명된 작업을 제공합니다.

표 29-1 작업 맵: 원격 시스템 액세스

작업	설명	수행 방법
원격 시스템에 로그인(rlogin)	<ul style="list-style-type: none"> <li>■ .rhosts 파일을 제거합니다.</li> <li>■ rlogin 명령을 사용하여 원격 시스템에 액세스합니다.</li> </ul>	<p>580 페이지 “.rhosts 파일을 검색하여 제거하는 방법”</p> <p>580 페이지 “원격 시스템이 작동 중인지 알아보는 방법”</p> <p>581 페이지 “원격 시스템에 로그인한 사용자를 알아보는 방법”</p> <p>582 페이지 “원격 시스템에 로그인하는 방법(rlogin)”</p> <p>582 페이지 “원격 시스템에서 로그아웃하는 방법(exit)”</p>
원격 시스템에 로그인(ftp)	<ul style="list-style-type: none"> <li>■ ftp 연결을 열고 닫습니다.</li> <li>■ 원격 시스템에 파일을 복사하고 해당 시스템에서 파일을 복사해 옵니다.</li> </ul>	<p>584 페이지 “원격 시스템에 대한 ftp 연결을 여는 방법”</p> <p>585 페이지 “원격 시스템에 대한 ftp 연결을 닫는 방법”</p> <p>585 페이지 “원격 시스템에서 파일을 복사하는 방법(ftp)”</p> <p>587 페이지 “원격 시스템으로 파일을 복사하는 방법(ftp)”</p>

표 29-1 작업 맵: 원격 시스템 액세스 (계속)

작업	설명	수행 방법
rcp를 사용하여 원격 파일 복사	rcp 명령을 사용하여 원격 시스템에 파일을 복사하고 해당 시스템에서 파일을 복사해 옵니다.	591 페이지 “로컬 시스템과 원격 시스템 간에 파일을 복사하는 방법(rcp)”

## 원격 시스템에 로그인(rlogin)

rlogin 명령을 사용하면 원격 시스템에 로그인할 수 있습니다. 로그인되면 원격 파일 시스템을 탐색하여 해당 내용을 조작하거나(권한에 따라 달라짐), 파일을 복사하거나, 원격 명령을 실행할 수 있습니다.

로그인 대상 시스템이 원격 도메인에 있는 경우 시스템 이름에 도메인 이름을 추가하십시오. 이 예에서는 SOLAR가 원격 도메인의 이름입니다.

```
rlogin pluto.SOLAR
```

또한 Ctrl-d를 누르면 언제든지 원격 로그인 작업을 인터럽트할 수 있습니다.

## 원격 로그인을 위한 인증(rlogin)

rlogin 작업을 위한 인증(신원 입증)은 원격 시스템 또는 네트워크 환경에서 수행할 수 있습니다.

이러한 인증 형식의 주요 차이점은 사용자가 관리자에게 요구하는 상호 작용의 유형과 인증 형식이 설정되는 방법입니다. 원격 시스템이 사용자를 인증하려고 하는 경우 사용자가 /etc/hosts.equiv 또는 .rhosts 파일을 설정하지 않았으면 암호를 입력하라는 메시지가 표시됩니다. 네트워크가 사용자를 인증하려고 하는 경우에는 네트워크에 사용자의 신원이 이미 알려져 있으므로 암호를 입력하라는 메시지가 표시되지 않습니다.

원격 시스템은 사용자를 인증하려고 할 때 해당 로컬 파일에 있는 정보에 의존합니다. 이는 특히 다음 중 하나가 참인 경우 그렇습니다.

- 시스템 이름 및 사용자 이름이 원격 시스템의 /etc/hosts.equiv 파일에 나타납니다.
- 시스템 이름 및 사용자 이름이 원격 사용자의 .rhosts 파일에서 원격 사용자의 홈 디렉토리 아래에 나타납니다.

네트워크 인증은 다음 두 방법 중 하나에 의존합니다.

- 로컬 네트워크 정보 서비스 및 자동 마운트를 통해 설정된 “신뢰할 수 있는 네트워크 환경”
- 원격 시스템의 svc:/system/name-service/switch 서비스가 지시한 네트워크 정보 서비스 중 하나에 사용자에 대한 정보가 포함되어 있습니다.



주 - 네트워크 인증은 일반적으로 시스템 인증을 대체합니다.

## **/etc/hosts.equiv 파일**

/etc/hosts.equiv 파일에는 원격 시스템에 대해 신뢰할 수 있는 호스트 목록이 행당 하나씩 포함되어 있습니다. 사용자가 이 파일에 나열된 호스트 중 하나에서 rlogin을 사용하여 원격으로 로그인하려고 하는 경우 원격 시스템이 사용자의 암호 항목에 액세스할 수 있으면 원격 시스템에서 사용자가 암호 없이 로그인할 수 있습니다.

일반적인 hosts.equiv 파일의 구조는 다음과 같습니다.

```
host1
host2 user_a
+@group1
-@group2
```

hosts.equiv에서 host1에 대한 이전 항목과 같이 호스트에 대한 단순한 항목이 만들어지면 해당 호스트와 해당 시스템의 모든 사용자를 신뢰할 수 있다는 의미입니다.

예의 두번째 항목에서와 같이 사용자 이름도 언급되면 지정된 사용자가 액세스를 시도할 경우에만 해당 호스트를 신뢰할 수 있습니다.

플러스 기호(+)가 앞에 오는 그룹 이름은 해당 넷 그룹에 있는 모든 시스템을 신뢰할 수 있는 것으로 간주함을 의미합니다.

마이너스 기호(-)가 앞에 오는 그룹 이름은 해당 넷 그룹에 있는 시스템 중 신뢰할 수 있는 것으로 간주되는 시스템이 없음을 의미합니다.

## **/etc/hosts.equiv 파일 사용 시의 보안 위험**

/etc/hosts.equiv 파일은 보안 위험을 초래합니다. /etc/hosts.equiv 파일을 시스템에서 유지 관리하는 경우 네트워크에서 신뢰할 수 있는 호스트만 포함해야 합니다. 이 파일에는 다른 네트워크에 속하는 호스트나 공공 장소에 있는 시스템을 포함해서는 안 됩니다. 예를 들어, 단자실에 있는 호스트는 포함하지 마십시오.

신뢰할 수 없는 호스트를 사용하면 심각한 보안 문제가 발생할 수 있습니다.

/etc/hosts.equiv 파일을 올바르게 구성된 파일로 교체하거나 파일을 완전히 제거하십시오.

/etc/hosts.equiv 파일에 단일 + 행이 있으면 알려진 모든 호스트를 신뢰할 수 있다는 의미입니다.

## **.rhosts 파일**

.rhosts 파일은 사용자에게 해당하는 /etc/hosts.equiv 파일입니다. 이 파일에는 일반적인 호스트 대신 호스트-사용자 조합 목록이 포함되어 있습니다. 호스트-사용자 조합이 이 파일에 나열되고, 지정된 사용자에게 암호 없이 지정 호스트에서 원격으로 로그인할 수 있는 권한이 부여됩니다.

.rhosts 파일은 사용자의 홈 디렉토리에서 최상위 레벨에 상주해야 합니다. 하위 디렉토리에 있는 .rhost 파일은 참조되지 않습니다.

사용자는 자신의 홈 디렉토리에서 .rhosts 파일을 만들 수 있습니다. .rhosts 파일을 사용해도 /etc/hosts.equiv 파일 없이 다른 시스템에 있는 사용자 자신의 여러 계정 간에 신뢰할 수 있는 액세스를 허용할 수 있습니다.

## .rhosts 파일 사용 시의 보안 위험

.rhosts 파일은 큰 보안 문제를 초래합니다. /etc/hosts.equiv 파일이 시스템 관리자의 제어 하에 있어 효과적으로 관리될 수 있는 동안에는 모든 사용자가 시스템 관리자에게 알리지 않고 원하는 모든 사용자에게 액세스 권한을 부여하는 .rhosts 파일을 만들 수 있습니다.

모든 사용자의 홈 디렉토리가 단일 서버에 있으며 특정 사용자만 해당 서버에 대한 수퍼 유저 액세스 권한을 가지는 경우 빈 파일을 해당 홈 디렉토리에서 수퍼 유저로 만들면 사용자가 .rhosts 파일을 사용하는 것을 막을 수 있습니다. 그런 다음 이 파일의 사용 권한을 000으로 변경하면 수퍼 유저라도 이를 쉽게 변경할 수 없게 됩니다. 이렇게 변경하면 사용자가 .rhosts 파일을 무책임하게 사용하여 시스템 보안에 위험을 초래하는 것을 실질적으로 막을 수 있습니다. 그러나 사용자가 유효 경로를 자신의 홈 디렉토리로 변경할 수 있으면 이렇게 변경한다고 해서 문제가 해결되지 않습니다.

.rhosts 파일을 안전하게 관리하는 유일한 방법은 해당 파일을 아예 허용하지 않는 것입니다. 자세한 내용은 580 페이지 “[.rhosts 파일을 검색하여 제거하는 방법](#)”을 참조하십시오. 시스템 관리자는 시스템을 자주 검사하여 이 정책이 위반되었는지 확인할 수 있습니다. 이 정책의 한 가지 예외가 될 수 있는 것은 루트 계정입니다. 네트워크 백업 및 기타 원격 서비스를 수행하려면 .rhosts 파일이 있어야 할 수 있습니다.

## 원격 로그인 링크 만들기

시스템이 적절히 구성된 경우 원격 로그인을 링크할 수 있습니다. 예를 들어, earth에 있는 사용자가 jupiter에 로그인하고, 여기에서 pluto에 로그인하기로 결정합니다.

사용자는 jupiter에서 로그아웃한 다음 pluto에 직접 로그인할 수도 있지만 링크를 만들면 더 편리합니다.

암호를 제공할 필요 없이 원격 로그인을 링크하려면 /etc/hosts.equiv 또는 .rhosts 파일을 올바르게 설정해야 합니다.

## 직접 또는 간접 원격 로그인

rlogin 명령을 사용하면 원격 시스템에 직접 또는 간접적으로 로그인할 수 있습니다.

직접 원격 로그인은 기본 사용자 이름, 즉 현재 로컬 시스템에 로그인되어 있는 개인의 사용자 이름으로 시도됩니다. 이것이 가장 흔한 원격 로그인 방식입니다.

간접 원격 로그인은 원격 로그인 작업 중 제공되는 다른 사용자 이름으로 시도됩니다. 이것은 잠시 빌린 워크스테이션에서 시도할 수 있는 원격 로그인 방식입니다. 예를 들어, 동료의 사무실에 있는데 홈 디렉토리에 있는 파일을 검사해야 하는 경우 동료의 시스템에서 자신의 시스템에 원격으로 로그인해야 할 수 있습니다. 이때에는 자신의 사용자 이름을 제공하여 간접 원격 로그인을 수행할 것입니다.

다음 표에는 직접 및 간접 로그인과 인증 방법 간의 종속성이 요약되어 있습니다.

표 29-2 로그인 방법과 인증 방법 간의 종속성(rlogin)

로그인 유형	사용자 이름 제공자	인증	암호
직접	시스템	네트워크	없음
		시스템	필수
간접	사용자	네트워크	없음
		시스템	필수

## 원격으로 로그인한 후 수행되는 작업

원격 시스템에 로그인하면 `rlogin` 명령이 사용자의 홈 디렉토리를 찾으려고 합니다. `rlogin` 명령이 사용자의 홈 디렉토리를 찾지 못하면 사용자가 원격 시스템의 루트(/) 디렉토리에 지정됩니다. 예를 들면 다음과 같습니다.

```
Unable to find home directory, logging in with /
```

그러나 `rlogin` 명령이 사용자의 홈 디렉토리를 찾으면 `.cshrc` 파일과 `.login` 파일이 모두 소스가 됩니다. 따라서 원격 로그인 후에는 사용자 프롬프트가 표준 로그인 프롬프트가 되고 현재 디렉토리는 로컬로 로그인할 때와 동일하게 유지됩니다.

예를 들어, 평상시의 프롬프트에 시스템 이름 및 작업 디렉토리가 표시되는 경우 로그인하면 작업 디렉토리가 홈 디렉토리이고 로그인 프롬프트는 다음과 같습니다.

```
earth(/home/smith):
```

그런 다음 원격 시스템에 로그인하면 유사한 프롬프트가 표시되고 `rlogin` 명령을 입력한 디렉토리에 관계없이 작업 디렉토리가 홈 디렉토리입니다.

```
earth(/home/smith): rlogin pluto
```

```
·
·
·
```

```
pluto(/home/smith):
```

유일한 차이점은 프롬프트 시작 시 원격 시스템의 이름이 로컬 시스템 대신 사용된다는 것입니다. 원격 파일 시스템은 사용자의 홈 디렉토리와 병렬로 나타납니다.

실질적으로, 디렉토리를 `/home`으로 변경한 다음 `ls`를 실행하면 다음이 표시됩니다.

```
earth(home/smith): cd ..
earth(/home): ls
smith jones
```

## ▼ .rhosts 파일을 검색하여 제거하는 방법

### 1 관리자가 됩니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### 2 **find(1)** 명령을 통해 .rhosts 파일을 검색하여 제거합니다.

```
# find home-directories -name .rhosts -print -exec rm {} \;
```

*home-directories*     사용자의 홈 디렉토리가 있는 디렉토리에 대한 경로를 식별합니다. 여러 경로를 입력하여 한 번에 둘 이상의 홈 디렉토리를 입력할 수 있습니다.

*-name .rhosts*     파일 이름을 식별합니다.

*-print*     현재 경로 이름을 인쇄합니다.

*-exec rm {} \;*     일치하는 파일 이름을 사용하여 식별된 모든 파일에 rm 명령을 적용할 것을 find 명령에 지시합니다.

find 명령은 지정된 디렉토리에서 시작하여 .rhosts라는 모든 파일을 검색합니다. 해당 파일이 검색되면 find가 경로를 화면에 인쇄하고 제거합니다.

### 예 29-1 .rhosts 파일을 검색하여 제거

다음 예에서는 /export/home 디렉토리에 있는 모든 사용자의 홈 디렉토리에서 .rhosts 파일을 검색하여 제거합니다.

```
# find /export/home -name .rhosts -print | xargs -i -t rm {} \;
```

## 원격 시스템이 작동 중인지 알아보는 방법

ping 명령을 사용하여 원격 시스템이 작동 중인지 알아보십시오.

```
$ ping system-name | ip-address
```

*system-name*     원격 시스템의 이름

*ip-address*     원격 시스템의 IP 주소

ping 명령은 다음 메시지 중 하나를 반환합니다.

상태 메시지	설명
<i>system-name</i> is alive	네트워크를 통해 시스템에 액세스할 수 있습니다.
ping: unknown host <i>system-name</i>	시스템 이름을 알 수 없습니다.
ping: no answer from <i>system-name</i>	시스템이 알려져 있지만 현재 작동 중이 아닙니다.

"핑"하는 시스템이 다른 도메인에 있는 경우 반환 메시지에 경로 지정 정보도 포함될 수 있습니다. 이 정보는 무시할 수 있습니다.

ping 명령의 시간 초과 값은 20초입니다. 실질적으로, 20초 안에 응답을 받지 못하면 세번째 메시지가 반환됩니다. **시간 초과** 값(초)을 입력하여 ping이 더 오래 또는 더 짧게 기다리게 만들 수 있습니다.

```
$ ping system-name | ip-address time-out
```

자세한 내용은 **ping(1M)**을 참조하십시오.

## 원격 시스템에 로그인한 사용자를 알아보는 방법

**rusers(1)** 명령을 사용하여 원격 시스템에 로그인한 사용자를 알아보십시오.

```
$ rusers [-l] remote-system-name
```

**rusers** (옵션 없음) 시스템 이름과 시스템에 현재 로그인되어 있는 사용자(루트 포함)의 이름을 차례대로 표시합니다.

**-l** 각 사용자에 대한 추가 정보(사용자의 로그인 창, 로그인 시간 및 날짜, 로그인 기간, 사용자가 로그인한 원격 시스템의 이름)를 표시합니다.

**예 29-2** 원격 시스템에 로그인한 사용자 알아보기

다음 예에서는 **rusers**의 짧은 출력을 보여줍니다.

```
$ rusers pluto
pluto smith jones
```

다음 예에서 긴 버전의 **rusers**는 원격 시스템 **starbug**에 두 사용자가 로그인되어 있음을 보여줍니다. 첫번째 사용자는 9월 10일에 시스템 콘솔에서 로그인하여 137시간 15분 동안 로그인된 상태를 유지하고 있습니다. 두번째 사용자는 9월 14일에 원격 시스템 **mars**에서 로그인했습니다.

```
$rusers -l starbug
root          starbug:console      Sep 10 16:13  137:15
rimmer        starbug:pts/0          Sep 14 14:37      (mars)
```

## 원격 시스템에 로그인하는 방법(rlogin)

**rlogin(1)** 명령을 사용하여 원격 시스템에 로그인하십시오.

```
$ rlogin [-l user-name] system-name
```

**rlogin** (옵션 없음) 현재 사용자 이름을 사용하여 원격 시스템에 효과적으로 **직접** 로그인합니다.

**-l user-name** 제공되는 사용자 이름을 사용하여 원격 시스템에 효과적으로 **간접** 로그인합니다.

네트워크가 사용자를 인증하려고 하는 경우 암호를 입력하라는 메시지가 표시되지 않습니다. 원격 시스템이 사용자를 인증하려고 하는 경우에는 암호를 제공하라는 메시지가 표시됩니다.

작업이 성공하면 **rlogin** 명령이 해당 시스템에 대한 사용자의 최신 원격 로그인에 대한 간략한 정보, 원격 시스템에서 실행되고 있는 운영 체제의 버전, 홈 디렉토리에 확인이 필요한 메일이 있는지 여부를 표시합니다.

**예 29-3** 원격 시스템에 로그인(rlogin)

다음 예에서는 **pluto**에 대한 직접 원격 로그인의 출력을 보여줍니다. 사용자는 네트워크에 의해 인증되었습니다.

```
$ rlogin starbug
```

```
Last login: Mon Jul 12 09:28:39 from venus
Sun Microsystems Inc. SunOS 5.8 February 2000
starbug:
```

다음 예에서는 **pluto**에 대한 간접 원격 로그인의 출력을 보여줍니다. 사용자는 원격 시스템에 의해 인증됩니다.

```
$ rlogin -l smith pluto
```

```
password: user-password
Last login: Mon Jul 12 11:51:58 from venus
Sun Microsystems Inc. SunOS 5.8 February 2000
starbug:
```

## 원격 시스템에서 로그아웃하는 방법(exit)

**exit(1)** 명령을 사용하여 원격 시스템에서 로그아웃하십시오.

```
$ exit
```

예 29-4 원격 시스템에서 로그아웃(exit)

이 예에서는 사용자 smith가 pluto 시스템에서 로그아웃하는 것을 보여줍니다.

```
$ exit
pluto% logout
Connection closed.
earth%
```

## 원격 시스템에 로그인(ftp)

ftp 명령은 인터넷의 FTP(File Transfer Protocol)에 대한 사용자 인터페이스를 엽니다. 명령 인터프리터라고 하는 이 사용자 인터페이스를 사용하면 원격 시스템에 로그인하여 해당 파일 시스템에서 다양한 작업을 수행할 수 있습니다. 주요 작업은 다음 표에 요약되어 있습니다.

rlogin 및 rcp와 비교했을 때 ftp의 주요 장점은 ftp를 사용하기 위해 원격 시스템에서 UNIX를 실행하지 않아도 된다는 점입니다. 원격 시스템을 TCP/IP 통신용으로 구성하는 작업은 필요합니다. 그러나 rlogin이 ftp보다 더 많은 파일 조작 명령 세트에 대한 액세스를 제공합니다.

## 원격 로그인에 대한 인증(ftp)

ftp 원격 로그인 작업에 대한 인증은 다음 방법 중 하나를 사용하여 설정할 수 있습니다.

- 원격 시스템의 /etc/passwd 파일이나 이에 상응하는 네트워크 정보 서비스 맵 또는 테이블에 암호 항목 포함
- 원격 시스템에서 익명 ftp 계정 설정

## 필수 ftp 명령

표 29-3 필수 ftp 명령

명령	설명
ftp	ftp 명령 인터프리터에 액세스합니다.
ftp remote-system	원격 시스템에 대한 ftp 연결을 설정합니다. 자세한 내용은 <a href="#">584 페이지 “원격 시스템에 대한 ftp 연결을 여는 방법”</a> 을 참조하십시오.
open	명령 인터프리터에서 원격 시스템에 로그인합니다.
close	원격 시스템에서 로그아웃하여 명령 인터프리터로 돌아갑니다.

표 29-3 필수 ftp 명령 (계속)

명령	설명
bye	ftp 명령 인터프리터를 종료합니다.
help	모든 ftp 명령을 나열하거나 명령의 효과를 간략하게 설명(명령 이름이 제공된 경우)합니다.
reset	원격 ftp 서버와 명령-회신 시퀀싱을 재동기화합니다.
ls	원격 작업 디렉토리의 내용을 나열합니다.
pwd	원격 작업 디렉토리의 이름을 표시합니다.
cd	원격 작업 디렉토리를 변경합니다.
lcd	로컬 작업 디렉토리를 변경합니다.
mkdir	원격 시스템에서 디렉토리를 만듭니다.
rmdir	원격 시스템에서 디렉토리를 삭제합니다.
get, mget	원격 작업 디렉토리에서 로컬 작업 디렉토리로 하나 이상의 파일을 복사합니다.
put, mput	로컬 작업 디렉토리에서 원격 작업 디렉토리로 하나 이상의 파일을 복사합니다.
delete, mdelete	원격 작업 디렉토리에서 하나 이상의 파일을 삭제합니다.

자세한 내용은 [ftp\(1\)](#) 매뉴얼 페이지를 참조하십시오.

## ▼ 원격 시스템에 대한 ftp 연결을 여는 방법

### 1 ftp 인증이 있는지 확인합니다.

583 페이지 “원격 로그인에 대한 인증(ftp)”에 설명된 대로 ftp 인증이 있어야 합니다.

### 2 ftp 명령을 사용하여 원격 시스템에 대한 연결을 엽니다.

```
$ ftp remote-system
```

연결에 성공하면 확인 메시지 및 프롬프트가 표시됩니다.

### 3 사용자 이름을 입력합니다.

```
Name (remote-system:user-name): user-name
```

### 4 메시지가 표시되면 암호를 입력합니다.

```
331 Password required for user-name:
```

```
Password: password
```



액세스하는 시스템에 설정된 익명 ftp 계정이 있는 경우 암호에 대한 전자 메일 주소를 입력하라는 메시지가 표시됩니다. ftp 인터페이스에서 암호가 수락되면 확인 메시지 및 (ftp>) 프롬프트가 표시됩니다.

이제 help를 비롯하여 ftp 인터페이스에 제공되는 모든 명령을 사용할 수 있습니다. 주요 명령은 표 29-3에 요약되어 있습니다.

### 예 29-5 원격 시스템에 대한 ftp 연결 열기

이 ftp 세션은 원격 시스템 pluto에서 사용자 smith가 설정한 것입니다.

```
$ ftp pluto
Connected to pluto.
220 pluto FTP server ready.
Name (pluto:smith): smith
331 Password required for smith:
Password: password
230 User smith logged in.
ftp>
```

## 원격 시스템에 대한 ftp 연결을 닫는 방법

bye 명령을 사용하여 원격 시스템에 대한 ftp 연결을 닫으십시오.

```
ftp> bye
221-You have transferred 0 bytes in 0 files.
221-Total traffic for this sessions was 172 bytes in 0 transfers.
221-Thanks you for using the FTP service on spdev.
221 Goodbye.
```

연결 종료 메시지와 평상시의 셸 프롬프트가 차례로 나타납니다.

## ▼ 원격 시스템에서 파일을 복사하는 방법(ftp)

- 1 원격 시스템에서 가져온 파일을 복사해 넣을 로컬 시스템의 디렉토리로 변경합니다.

```
$ cd target-directory
```

- 2 ftp 연결을 설정합니다.

584 페이지 “원격 시스템에 대한 ftp 연결을 여는 방법”을 참조하십시오.

- 3 소스 디렉토리로 변경합니다.

```
ftp> cd source-directory
```

시스템에 자동 마운트가 사용되는 경우 원격 시스템 사용자의 홈 디렉토리가 /home 아래에서 현재 사용자의 홈 디렉토리와 병렬로 나타납니다.

- 4 소스 파일에 대한 읽기 권한이 있는지 확인합니다.

```
ftp> ls -l
```

- 5 전송 유형을 **binary**로 설정합니다.

```
ftp> binary
```

- 6 단일 파일을 복사하려면 **get** 명령을 사용합니다.

```
ftp> get filename
```

- 7 다중 파일을 한 번에 복사하려면 **mget** 명령을 사용합니다.

```
ftp> mget filename [filename ...]
```

일련의 개별 파일 이름을 제공할 수 있으며 와일드카드 문자를 사용할 수 있습니다. **mget** 명령은 각 파일을 개별적으로 복사하며 매번 확인 메시지를 표시합니다.

- 8 **ftp** 연결을 닫습니다.

```
ftp> bye
```

#### 예 29-6 원격 시스템에서 파일 복사(ftp)

이 예에서는 사용자 **kryten**이 **pluto** 시스템에 대한 **ftp** 연결을 열고 **get** 명령을 사용하여 **/tmp** 디렉토리에서 단일 파일을 복사합니다.

```
$ cd $HOME
ftp pluto
Connected to pluto.
220 pluto FTP server (SunOS 5.8) ready.
Name (pluto:kryten): kryten
331 Password required for kryten.
Password: xxx
230 User kryten logged in.
ftp> cd /tmp
250 CWD command successful.
ftp> ls
200 PORT command successful.
150 ASCII data connection for /bin/ls (129.152.221.238,34344)
(0 bytes).
filea
files
ps_data
226 ASCII Transfer complete.
53 bytes received in 0.022 seconds (2.39 Kbytes/s)
ftp> get filea
200 PORT command successful.
150 ASCII data connection for filea (129.152.221.238,34331)
(0 bytes).
221 Goodbye.
```

이 예에서는 동일한 사용자 **kryten**이 **mget** 명령을 사용하여 **/tmp** 디렉토리에서 자신의 홈 디렉토리로 파일 세트를 복사합니다. **kryten**은 세트에 포함된 개별 파일을 수락하거나 거부할 수 있습니다.

```

$ ftp> cd /tmp
250 CWD command successful.
ftp> ls files
200 PORT command successful.
150 ASCII data connection for /bin/ls (129.152.221.238,34345)
(0 bytes).
fileb
filec
filed
remote: files
21 bytes received in 0.015 seconds (1.36 Kbytes/s)
ftp> cd files
250 CWD command successful.
ftp> mget file*
mget fileb? y
200 PORT command successful.
150 ASCII data connection for fileb (129.152.221.238,34347)
(0 bytes).
226 ASCII Transfer complete.
mget filec? y
200 PORT command successful.
150 ASCII data connection for filec (129.152.221.238,34348)
(0 bytes).
226 ASCII Transfer complete.
mget filed? y
200 PORT command successful.
150 ASCII data connection for filed (129.152.221.238,34351)
(0 bytes).
226 ASCII Transfer complete.200 PORT command successful.
ftp> bye
221 Goodbye.

```

## ▼ 원격 시스템으로 파일을 복사하는 방법(ftp)

### 1 로컬 시스템의 소스 디렉토리로 변경합니다.

ftp 명령을 입력하는 디렉토리가 로컬 작업 디렉토리이므로 이 작업의 소스 디렉토리입니다.

### 2 ftp 연결을 설정합니다.

584 페이지 “원격 시스템에 대한 ftp 연결을 여는 방법”을 참조하십시오.

### 3 대상 디렉토리로 변경합니다.

```
ftp> cd target-directory
```

이때 시스템에 자동 마운트가 사용되는 경우 원격 시스템 사용자의 홈 디렉토리가 /home 아래에서 현재 사용자의 홈 디렉토리와 병렬로 나타납니다.

### 4 대상 디렉토리에 대한 쓰기 권한이 있는지 확인합니다.

```
ftp> ls -l target-directory
```

**5 전송 유형을 binary로 설정합니다.**

```
ftp> binary
```

**6 단일 파일을 복사하려면 put 명령을 사용합니다.**

```
ftp> put filename
```

**7 다중 파일을 한 번에 복사하려면 mput 명령을 사용합니다.**

```
ftp> mput filename [filename ...]
```

일련의 개별 파일 이름을 제공할 수 있으며 와일드카드 문자를 사용할 수 있습니다. mput 명령은 각 파일을 개별적으로 복사하며 매번 확인 메시지를 표시합니다.

**8 ftp 연결을 닫으려면 bye를 입력합니다.**

```
ftp> bye
```

**예 29-7 원격 시스템으로 파일 복사(ftp)**

이 예에서는 사용자 kryten이 pluto 시스템에 대한 ftp 연결을 열고 put 명령을 사용하여 자신의 시스템에서 pluto 시스템의 /tmp 디렉토리로 파일을 복사합니다.

```
$ cd /tmp
ftp pluto
Connected to pluto.
220 pluto FTP server (SunOS 5.8) ready.
Name (pluto:kryten): kryten
331 Password required for kryten.
Password: xxx
230 User kryten logged in.
ftp> cd /tmp
250 CWD command successful.
ftp> put filef
200 PORT command successful.
150 ASCII data connection for filef (129.152.221.238,34356).
226 Transfer complete.
ftp> ls
200 PORT command successful.
150 ASCII data connection for /bin/ls (129.152.221.238,34357) (0 bytes).
filea
filef
files
ps_data
226 ASCII Transfer complete.
60 bytes received in 0.058 seconds (1.01 Kbytes/s)
ftp> bye
221 Goodbye.
```

이 예에서는 동일한 사용자 kryten이 mput 명령을 사용하여 자신의 홈 디렉토리에서 pluto의 /tmp 디렉토리로 파일 세트를 복사합니다. kryten은 세트에 포함된 개별 파일을 수락하거나 거부할 수 있습니다.

```
$ cd $HOME/testdir
$ ls
```

```

test1 test2 test3
$ ftp pluto
Connected to pluto.
220 pluto FTP server (SunOS 5.8) ready.
Name (pluto:kryten): kryten
331 Password required for kryten.
Password: xxx
230 User kryten logged in.
ftp> cd /tmp
250 CWD command successful.
ftp> mput test*
mput test1? y
200 PORT command successful.
150 ASCII data connection for test1 (129.152.221.238,34365).
226 Transfer complete.
mput test2? y
200 PORT command successful.
150 ASCII data connection for test2 (129.152.221.238,34366).
226 Transfer complete.
mput test3? y
200 PORT command successful.
150 ASCII data connection for file (129.152.221.238,34356).
226 Transfer complete.
ftp> bye
221 Goodbye.

```

## rcp를 사용한 원격 복사

rcp 명령은 로컬 및 원격 시스템 간이나 두 원격 시스템 간에 파일 또는 디렉토리를 복사합니다. 원격 시스템(rlogin 명령을 사용하여 로그인한 후) 또는 로컬 시스템(원격 시스템에 로그인 안함)에서 이 명령을 사용할 수 있습니다.

rcp를 사용하면 다음과 같은 원격 복사 작업을 수행할 수 있습니다.

- 사용자 시스템에서 원격 시스템으로 파일 또는 디렉토리 복사
- 원격 시스템에서 로컬 시스템으로 파일 또는 디렉토리 복사
- 로컬 시스템에서 원격 시스템 간에 파일 또는 디렉토리 복사

자동 마운트가 실행되고 있는 경우 cp 명령을 사용하여 이러한 원격 작업을 수행할 수 있습니다. 그러나 cp의 범위는 자동 마운트를 통해 만들어진 가상 파일 시스템과 사용자의 홈 디렉토리에 상대적인 작업으로 제한됩니다. rcp는 이러한 제약 조건 없이 동일한 작업을 수행하므로 이 절에서는 해당 작업의 rcp 버전만 설명합니다.

## 복사 작업에 대한 보안 고려 사항

시스템 간에 파일 또는 디렉토리를 복사하려면 로그인하고 파일을 복사할 수 있는 권한이 있어야 합니다.



주의 - cp 명령과 rcp 명령 모두 주의 메시지 없이 파일을 덮어쓸 수 있습니다. 명령을 실행하기 전에 파일 이름이 올바른지 확인하십시오.

## 소스 및 대상 지정

rcp 명령을 C 셸에 포함하면 절대 경로 이름이나 축약 경로 이름을 사용하여 소스(복사할 파일 또는 디렉토리) 및 대상(파일 또는 디렉토리를 복사해 넣을 위치)을 지정할 수 있습니다.

상황	절대 경로 이름	축약 경로 이름
로컬 시스템에서	<code>mars:/home/jones/myfile.txt</code>	<code>~jones/myfile.txt</code>
원격 로그인 후에	<code>/home/jones/myfile.txt</code>	<code>~jones/myfile.txt</code>

절대 경로 이름은 특정 시스템에 마운트되어 있는 파일 또는 디렉토리를 식별합니다. 이전 예에서 첫번째 절대 경로 이름은 mars 시스템에 있는 (myfile.txt) 파일을 식별합니다. 축약 경로 이름은 상주 위치에 관계없이 사용자의 홈 디렉토리에 상대적인 파일 또는 디렉토리를 식별합니다. 이전 첫번째 예에서 축약 경로 이름은 동일한 파일 myfile.txt를 식별하지만 “~” 기호를 사용하여 jones 홈 디렉토리를 나타냅니다.

~ = mars:/home/jones

두번째 행의 예에서는 원격 로그인 후에 절대 및 축약 경로 이름을 사용한 것을 보여줍니다. 축약 경로 이름의 경우 차이점이 뚜렷하지 않습니다. 그러나 원격 로그인 작업이 jones 홈 디렉토리를 로컬 사용자의 홈 디렉토리와 병렬로 로컬 시스템에 마운트했기 때문에 절대 경로 이름에 더 이상 시스템 이름 mars가 필요하지 않습니다. 원격 로그인 작업이 다른 사용자의 홈 디렉토리를 마운트하는 방법에 대한 자세한 내용은 579 페이지 “원격으로 로그인한 후 수행되는 작업”을 참조하십시오.

다음 표에는 C 셸에서 인식되는 절대 및 축약 경로 이름의 샘플이 나와 있습니다. 샘플에는 다음 용어가 사용됩니다.

- 작업 디렉토리 - rcp 명령이 입력되는 디렉토리입니다. 원격 또는 로컬일 수 있습니다.
- 현재 사용자 - rcp 명령 입력 시 사용되는 사용자 이름입니다.

표 29-4 디렉토리 및 파일 이름에 허용되는 구문

로그인 대상	구문	설명
로컬 시스템	<code>.</code>	로컬 작업 디렉토리
	<code>path/filename</code>	로컬 작업 디렉토리에 있는 path 및 filename

표 29-4 디렉토리 및 파일 이름에 허용되는 구분 (계속)

로그인 대상	구분	설명
원격 시스템	~	현재 사용자의 홈 디렉토리
	~/path/filename	현재 사용자의 홈 디렉토리 아래에 있는 <i>path</i> 및 <i>filename</i>
	~user	<i>user</i> 의 홈 디렉토리
	~user/path/filename	<i>user</i> 의 홈 디렉토리 아래에 있는 <i>path</i> 및 <i>filename</i>
	remote-system:path/filename	원격 작업 디렉토리에 있는 <i>path</i> 및 <i>filename</i>
	.	원격 작업 디렉토리
	filename	원격 작업 디렉토리에 있는 <i>filename</i>
	path/filename	원격 작업 디렉토리에 있는 <i>path</i> 및 <i>filename</i>
	~	현재 사용자의 홈 디렉토리
	~/path/filename	현재 사용자의 홈 디렉토리에 있는 <i>path</i> 및 <i>filename</i>
	~user	<i>user</i> 의 홈 디렉토리
	~/user/path/filename	<i>user</i> 의 홈 디렉토리 아래에 있는 <i>path</i> 및 <i>filename</i>
	local-system:path/filename	로컬 작업 디렉토리에 있는 <i>path</i> 및 <i>filename</i>

## ▼ 로컬 시스템과 원격 시스템 간에 파일을 복사하는 방법(rcp)

### 1 복사할 수 있는 권한이 있는지 확인합니다.

최소한 소스 시스템에서 읽기 권한, 대상 시스템에서 쓰기 권한이 있어야 합니다.

### 2 소스 및 대상의 위치를 결정합니다.

소스 또는 대상의 경로를 모르는 경우 582 페이지 “원격 시스템에 로그인하는 방법(rlogin)”에 설명된 대로 먼저 rlogin 명령을 사용하여 원격 시스템에 로그인할 수 있습니다. 그런 다음 해당 위치를 찾을 때까지 원격 시스템을 탐색합니다. 그런 후 로그아웃하지 않고 다음 단계를 수행할 수 있습니다.

### 3 파일 또는 디렉토리를 복사합니다.

```
$ rcp [-r] source-file|directory target-file|directory
```

rcp (옵션 없음) 소스에서 대상으로 단일 파일을 복사합니다.

-r 소스에서 대상으로 디렉토리를 복사합니다.

이 구문은 사용자가 원격 시스템에 로그인했는지, 아니면 로컬 시스템에 로그인했는지에 관계없이 적용됩니다. 표 29-4 및 다음 예에 설명된 것과 같이 파일 또는 디렉토리의 경로 이름만 변경됩니다.

“~” 및 “.” 문자를 사용하여 로컬 파일 또는 디렉토리 이름의 경로 부분을 지정할 수 있습니다. 그러나 “~”는 원격 시스템이 아니라 현재 사용자에게 적용되고 “.”은 현재 로그인된 시스템에 적용됩니다. 이러한 기호에 대한 설명은 표 29-4를 참조하십시오.

#### 예 29-8 rcp를 사용하여 원격 파일을 로컬 시스템으로 복사

이 예에서 rcp는 원격 시스템 pluto의 /home/jones 디렉토리에서 로컬 시스템 earth에 있는 작업 디렉토리(/home/smith)로 letter.doc 파일을 복사하는 데 사용됩니다.

```
earth(/home/smith): rcp pluto:/home/jones/letter.doc .
```

이 경우 rcp 작업은 원격 로그인 없이 수행됩니다. 여기서 명령줄 끝에 있는 “.” 기호는 원격 시스템이 아니라 로컬 시스템을 가리킵니다.

대상 디렉토리는 로컬 사용자의 홈 디렉토리이기도 하므로 “~” 기호를 사용하여 지정할 수도 있습니다.

#### 예 29-9 rlogin 및 rcp를 사용하여 원격 파일을 로컬 시스템으로 복사

이 예에서 rcp 작업은 원격 시스템에서 로컬 시스템으로 파일을 복사하기 위해 rlogin 명령이 실행된 후에 실행됩니다. 작업 흐름은 이전 예와 같지만 경로는 원격 로그인을 허용하기 위해 변경됩니다.

```
earth(/home/smith): rlogin pluto
.
.
.
pluto(/home/jones): rcp letter.doc ~
```

이 경우에는 명령줄 끝에서 “.” 기호를 사용하는 것이 부적절합니다. 원격 로그인 때문에 이 기호는 단순히 원격 시스템을 가리켜 본질적으로 rcp에 중복 파일을 만들도록 지시하게 됩니다. 그러나 “~” 기호는 로그인 대상이 원격 시스템이더라도 현재 사용자의 홈 디렉토리를 가리킵니다.

#### 예 29-10 rcp를 사용하여 로컬 파일을 원격 시스템으로 복사

이 예에서 rcp는 로컬 시스템 earth의 홈 디렉토리(/home/smith)에서 원격 시스템 pluto의 /home/jones 디렉토리로 notice.doc 파일을 복사하는 데 사용됩니다.

```
earth(/home/smith): rcp notice.doc pluto:/home/jones
```



제공된 원격 파일 이름이 없기 때문에 `notice.doc` 파일이 동일한 이름을 사용하여 `/home/jones` 디렉토리로 복사됩니다.

이 경우 이전 예의 `rcp` 작업이 반복되지만 `rcp`는 로컬 시스템(`/tmp`)의 다른 작업 디렉토리에서 입력됩니다. 현재 사용자의 홈 디렉토리를 가리키기 위해 “~” 기호를 사용하는 방법은 다음과 같습니다.

```
earth(/tmp): rcp ~/notice.doc pluto:/home/jones
```

### 예 29-11 rlogin 및 rcp를 사용하여 로컬 파일을 원격 시스템으로 복사

이 예에서 `rcp` 작업은 로컬 파일을 원격 디렉토리로 복사하기 위해 `rlogin` 명령이 실행된 후에 실행됩니다. 작업 흐름은 이전 예와 같지만 경로는 원격 로그인을 허용하기 위해 변경됩니다.

```
earth(/home/smith): rlogin pluto
.
.
.
pluto(/home/jones): rcp ~/notice.doc .
```

이 경우 “~” 기호를 사용하여 현재 사용자의 홈 디렉토리가 로컬 시스템에 있더라도 해당 디렉토리를 나타낼 수 있습니다. “.” 기호는 사용자가 원격 시스템에 로그인되어 있으므로 원격 시스템에 있는 작업 디렉토리를 가리킵니다. 다음은 동일한 작업을 수행하는 대체 구문입니다.

```
pluto(/home/jones): rcp earth:/home/smith/notice.doc /home/jones
```



## 제 7 부

# 네트워크 서비스 모니터링 항목

이 절에서는 네트워크 서비스 모니터링에 대한 단계별 지침을 제공합니다.



## 네트워크 성능 모니터링(작업)

이 장에서는 네트워크 성능을 모니터링하는 방법에 대해 설명합니다. 다음은 이 장의 단계별 지침 목록입니다.

- 598 페이지 “네트워크에서 호스트 응답을 확인하는 방법”
- 598 페이지 “네트워크에서 호스트로 패킷을 보내는 방법”
- 599 페이지 “네트워크에서 패킷을 캡처하는 방법”
- 599 페이지 “네트워크 상태를 확인하는 방법”
- 602 페이지 “NFS 서버 및 클라이언트 통계를 표시하는 방법”

## 네트워크 성능 모니터링

다음 표는 네트워크 성능을 모니터링하는 데 사용할 수 있는 명령에 대해 설명합니다.

표 30-1 네트워크 모니터링 명령

명령	설명
ping	네트워크에서 호스트 응답을 확인합니다.
spray	패킷 크기의 안정성을 테스트합니다. 이 명령은 네트워크에서 패킷이 지연되고 있는지 또는 삭제되고 있는지 알려줍니다.
snoop	네트워크에서 패킷을 캡처하고 각 클라이언트의 각 서버 호출을 추적합니다.
netstat	TCP/IP 트래픽에 사용되는 인터페이스의 상태, IP 경로 지정 테이블 및 UDP, TCP, ICMP 및 IGMP에 대한 프로토콜당 통계를 포함하는 네트워크 상태를 표시합니다.
nfsstat	NFS 문제를 식별하는 데 사용할 수 있는 서버 및 클라이언트 통계의 요약을 표시합니다.

## 네트워크에서 호스트 응답을 확인하는 방법

ping 명령을 사용하여 네트워크에서 호스트 응답을 확인합니다.

```
$ ping hostname
```

물리적 문제가 예상되는 경우 ping 명령을 사용하여 네트워크에서 일부 호스트의 응답 시간을 확인할 수 있습니다. 한 호스트의 응답이 예상한 바와 다를 경우 해당 호스트를 조사할 수 있습니다. 물리적 문제는 다음과 같은 이유로 발생할 수 있습니다.

- 느슨하게 연결된 케이블 또는 커넥터
- 고르지 못한 지면 상태
- 종단점 없음
- 신호 반사

이 명령에 대한 자세한 내용은 [ping\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

### 예 30-1 네트워크에서 호스트 응답 확인

ping의 가장 간단한 버전은 단일 패킷을 네트워크의 호스트로 보냅니다. ping 명령이 올바른 응답을 받는 경우 명령은 *host is alive*라는 메시지를 출력합니다.

```
$ ping elvis
elvis is alive
```

-s 옵션을 사용하면 ping은 초당 하나의 데이터그램을 호스트로 보냅니다. 그러면 명령은 각 응답 및 라운드 트립에 필요한 시간을 출력합니다. 예는 다음과 같습니다.

```
$ ping -s pluto
64 bytes from pluto (123.456.78.90): icmp_seq=0. time=3.82 ms
64 bytes from pluto (123.456.78.90): icmp_seq=5. time=0.947 ms
64 bytes from pluto (123.456.78.90): icmp_seq=6. time=0.855 ms
^C
---pluto PING Statistics---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip (ms) min/avg/max/sttdev = 0.855/1.87/3.82/1.7
```

## 네트워크에서 호스트로 패킷을 보내는 방법

spray 명령을 사용하여 패킷 크기의 안정성을 테스트합니다.

```
$ spray [ -c count -d interval -l packet-size] hostname
```

-i count      보낼 패킷 수입니다.

-d interval      패킷을 보내는 중에 일시 중지할 마이크로초의 수입니다. 지연을 사용하지 않으면 버퍼가 대폭 줄어들 수 있습니다.

-l packet-size      패킷 크기입니다.

*hostname* 패킷을 보낼 시스템입니다.

이 명령에 대한 자세한 내용은 [spray\(1M\)](#)를 참조하십시오.

**예 30-2** 네트워크에서 호스트로 패킷 보내기

다음 예에서는 2048바이트 크기의 패킷(-l 2048)으로 100 패킷(-c 100)을 호스트로 보냅니다. 각 버스트 사이에 20마이크로초의 지연 시간으로 패킷을 보냅니다(-d 20).

```
$ spray -c 100 -d 20 -l 2048 pluto
sending 100 packets of length 2048 to pluto ...
no packets dropped by pluto
279 packets/sec, 573043 bytes/sec
```

## 네트워크에서 패킷을 캡처하는 방법

네트워크에서 패킷을 캡처하고 각 클라이언트의 각 서버 호출을 추적하려면 `snoop` 명령을 사용합니다. 이 명령은 일부 네트워크 성능 문제를 빨리 처리할 수 있는 정확한 시간 기록을 제공합니다. 자세한 내용은 [snoop\(1M\)](#)를 참조하십시오.

**# snoop**

패킷 삭제는 부족한 버퍼 공간 또는 CPU 과부하 문제로 인해 발생할 수 있습니다.

## 네트워크 상태를 확인하는 방법

네트워크 인터페이스 상태, 경로 지정 테이블 및 다양한 프로토콜에 대한 통계와 같은 네트워크 상태 정보를 표시하려면 `netstat` 명령을 사용합니다.

```
$ netstat [-i] [-r] [-s]
```

-i TCP/IP 인터페이스 상태를 표시합니다.

-r IP 경로 지정 테이블을 표시합니다.

-s UDP, TCP, ICMP 및 IGMP 프로토콜에 대한 통계를 표시합니다.

자세한 내용은 [netstat\(1M\)](#)을 참조하십시오.

### 예 - 네트워크 상태 확인

다음 예에서는 TCP/IP 트래픽에 사용되는 인터페이스 상태를 표시하는 `netstat -i` 명령의 출력을 보여줍니다.

```
$ netstat -i
Name Mtu Net/Dest Address Ipkts Ierrs Opkts Oerrs Collis Queue
```

```
lo0    8232 software    localhost    1280    0    1280    0    0    0
eri0   1500 loopback      venus       1628480  0    347070   16   39354   0
```

이 화면에서는 각 인터페이스에서 시스템이 전송하고 수신한 패킷 수를 보여줍니다. 네트워크 트래픽이 활성화된 시스템에서는 **Ipkts**와 **Opkts** 모두 지속적으로 증가하는 것을 보여줍니다.

충돌 카운트의 수(**Collis**)를 출력 패킷(**Opkts**)의 수로 나누어 네트워크 충돌률을 계산합니다. 이전 예에서 충돌률은 11%입니다. 네트워크상에서 5~10%보다 큰 충돌률은 문제가 있음을 나타낼 수 있습니다.

입력 오류의 수를 입력 패킷의 총 수로 나누어 입력 패킷의 오류율을 계산합니다(**Ierrs/Ipkts**). 출력 패킷의 오류율은 출력 오류 수(**Oerrs/Opkts**)를 총 출력 패킷의 수로 나눈 값입니다. 입력 오류율이 0.25%보다 높으면 호스트에서 패킷을 삭제할 수 있습니다.

다음 예에서는 UDP, TCP, ICMP 및 IGMP 프로토콜의 프로토콜당 통계를 표시하는 **netstat -s** 명령의 출력을 보여줍니다.

```
UDP
  udpInDatagrams    =196543    udpInErrors        =      0
  udpOutDatagrams    =187820

TCP
  tcpRtoAlgorithm    =      4    tcpRtoMin           =    200
  tcpRtoMax          =   60000    tcpMaxConn          =     -1
  tcpActiveOpens     =   26952    tcpPassiveOpens     =    420
  tcpAttemptFails    =    1133    tcpEstabResets      =      9
  tcpCurrEstab       =     31    tcpOutSegs          =3957636
  tcpOutDataSegs     =2731494    tcpOutDataBytes     =1865269594
  tcpRetransSegs     =   36186    tcpRetransBytes     =3762520
  tcpOutAck          =1225849    tcpOutAckDelayed    =165044
  tcpOutUrg          =      7    tcpOutWinUpdate     =    315
  tcpOutWinProbe     =      0    tcpOutControl       =   56588
  tcpOutRsts         =     803    tcpOutFastRetrans   =    741
  tcpInSegs          =4587678
  tcpInAckSegs       =2087448    tcpInAckBytes       =1865292802
  tcpInDupAck        =109461    tcpInAckUnsent      =      0
  tcpInInorderSegs   =3877639    tcpInInorderBytes   =-598404107
  tcpInUnorderSegs   =   14756    tcpInUnorderBytes   =17985602
  tcpInDupSegs       =     34    tcpInDupBytes       =   32759
  tcpInPartDupSegs   =    212    tcpInPartDupBytes   =134800
  tcpInPastWinSegs   =      0    tcpInPastWinBytes   =      0
  tcpInWinProbe      =    456    tcpInWinUpdate      =      0
  tcpInClosed        =     99    tcpRttNoUpdate      =    6862
  tcpRttUpdate       =435097    tcpTimRetrans       =   15065
  tcpTimRetransDrop  =     67    tcpTimKeepalive     =    763
  tcpTimKeepaliveProbe=    1    tcpTimKeepaliveDrop =      0

IP
  ipForwarding       =      2    ipDefaultTTL        =    255
  ipInReceives       =11757234    ipInHdrErrors       =      0
  ipInAddrErrors     =      0    ipInChecksumErrs    =      0
  ipForwDatagrams    =      0    ipForwProhibits     =      0
  ipInUnknownProtos  =      0    ipInDiscards        =      0
```



```

ipInDelivers      =4784901  ipOutRequests     =4195180
ipOutDiscards     =    0    ipOutNoRoutes     =    0
ipReasmTimeout    =    60    ipReasmReqds      =   8723
ipReasmOKs        =   7565    ipReasmFails      =   1158
ipReasmDuplicates =    7     ipReasmPartDups   =    0
ipFragOKs         =  19938    ipFragFails       =    0
ipFragCreates     =116953    ipRoutingDiscards =    0
tcpInErrs         =    0     udpNoPorts        =6426577
udpInCksumErrs    =    0     udpInOverflows    =   473
rawipInOverflows  =    0

```

## ICMP

```

icmpInMsgs        =490338    icmpInErrors      =    0
icmpInCksumErrs   =    0     icmpInUnknowns    =    0
icmpInDestUnrechs =   618    icmpInTimeExcds   =   314
icmpInParmProbs   =    0     icmpInSrcQuenchs  =    0
icmpInRedirects   =   313    icmpInBadRedirects =    5
icmpInEchos       =   477    icmpInEchoReps    =   20
icmpInTimestamps  =    0     icmpInTimestampReps =    0
icmpInAddrMasks   =    0     icmpInAddrMaskReps =    0
icmpInFragNeeded  =    0     icmpOutMsgs        =   827
icmpOutDrops      =   103    icmpOutErrors      =    0
icmpOutDestUnrechs =   94    icmpOutTimeExcds   =  256
icmpOutParmProbs  =    0     icmpOutSrcQuenchs  =    0
icmpOutRedirects  =    0     icmpOutEchos       =    0
icmpOutEchoReps   =   477    icmpOutTimestamps  =    0
icmpOutTimestampReps =    0    icmpOutAddrMasks   =    0
icmpOutAddrMaskReps =    0    icmpOutFragNeeded  =    0
icmpInOverflows   =    0

```

## IGMP:

```

0 messages received
0 messages received with too few bytes
0 messages received with bad checksum
0 membership queries received
0 membership queries received with invalid field(s)
0 membership reports received
0 membership reports received with invalid field(s)
0 membership reports received for groups to which we belong
0 membership reports sent

```

다음 예에서는 IP 경로 지정 테이블을 표시하는 `netstat - r` 명령의 출력을 보여줍니다.

## Routing Table:

Destination	Gateway	Flags	Ref	Use	Interface
localhost	localhost	UH	0	2817	lo0
earth-bb	pluto	U	3	14293	eri0
224.0.0.0	pluto	U	3	0	eri0
default	mars-gate	UG	0	14142	

`netstat - r` 보고서의 필드는 다음 표에 설명되어 있습니다.

표 30-2 netstat -r 명령의 출력

필드 이름	설명
Flags	<div>U 경로가 작동 중입니다.</div> <div>G 게이트웨이를 통한 경로입니다.</div> <div>H 호스트로 가는 경로입니다.</div> <div>D 경로가 재지정을 사용하여 동적으로 만들어졌습니다.</div>
Ref	동일 링크 계층을 공유하는 현재 경로의 수를 표시합니다.
Use	전송한 패킷 수를 나타냅니다.
Interface	경로에 사용된 네트워크 인터페이스를 나열합니다.

## NFS 서버 및 클라이언트 통계를 표시하는 방법

NFS 분산 파일 서비스는 로컬 명령을 원격 호스트에 대한 요청으로 변환하는 원격 프로시저 호출(RPC) 기능을 사용합니다. 원격 프로시저 호출은 동기식입니다. 클라이언트 응용 프로그램은 서버가 호출을 완료하고 결과를 반환할 때까지 차단되거나 일시 중지됩니다. NFS 성능에 영향을 미치는 주요 요인 중 하나는 재전송률입니다.

파일 서버가 클라이언트 요청에 응답할 수 없는 경우 클라이언트는 중지하기 전에 지정된 횟수의 요청을 재전송합니다. 각 재전송은 시스템 과부하 및 네트워크 트래픽 증가를 유발합니다. 과도한 재전송은 네트워크 성능 문제를 유발할 수 있습니다. 재전송률이 높은 경우, 다음과 같은 이유로 인한 것일 수 있습니다.

- 서버에 과부하가 걸려 요청을 완료하기에 너무 느림
- 이더넷 인터페이스에서 패킷을 삭제하는 중
- 네트워크 혼잡, 패킷 전송 속도 느려짐

다음 표에서는 클라이언트 및 서버 통계를 표시하는 `nfsstat` 옵션에 대해 설명합니다.

표 30-3 클라이언트/서버 통계 표시 명령

명령	표시
<code>nfsstat -c</code>	클라이언트 통계
<code>nfsstat -s</code>	서버 통계
<code>netstat -m</code>	각 파일 시스템에 대한 네트워크 통계

`nfsstat -c`를 사용하여 클라이언트 통계를 표시하고 `nfsstat -s`를 사용하여 서버 통계를 표시합니다. `netstat -m`을 사용하여 각 파일 시스템의 네트워크 통계를 표시합니다. 자세한 내용은 `nfsstat(1M)`을 참조하십시오.

## 예 - NFS 서버 및 클라이언트 통계 표시

다음 예에서는 클라이언트 pluto에 대한 RPC 및 NFS 데이터를 표시합니다.

```
$ nfsstat -c

Client rpc:
Connection oriented:
calls      badcalls  badxids  timeouts  newcreds  badverfs  timers
1595799    1511      59        297        0          0          0
cantconn  nomem      interrupts
1198       0          7
Connectionless:
calls      badcalls  retrans  badxids  timeouts  newcreds  badverfs
80785      3135      25029    193       9543       0          0
timers     nomem      cantsend
17399      0          0

Client nfs:
calls      badcalls  clgets   cltoomany
1640097    3112      1640097  0
Version 2: (46366 calls)
null       getattr  setattr  root      lookup    readlink  read
0 0%       6589 14%  2202 4%  0 0%      11506 24%  0 0%      7654 16%
wrcache    write    create    remove    rename    link      symlink
0 0%       13297 28%  1081 2%  0 0%      0 0%      0 0%
mkdir      rmdir    readdir  statfs
24 0%      0 0%     906 1%   3107 6%
Version 3: (1585571 calls)
null       getattr  setattr  lookup    access    readlink  read
0 0%       508406 32%  10209 0%  263441 16%  400845 25%  3065 0%  117959 7%
write      create    mkdir     symlink    mknod     remove    rmdir
69201 4%  7615 0%   42 0%     16 0%      0 0%     7875 0%  51 0%
rename     link      readdir  readdir+   fsstat    fsinfo    pathconf
929 0%     597 0%    3986 0%  185145 11%  942 0%    300 0%    583 0%
commit
4364 0%

Client nfs_acl:
Version 2: (3105 calls)
null       getacl   setacl    getattr   access
0 0%       0 0%      0 0%      3105 100%  0 0%
Version 3: (5055 calls)
null       getacl   setacl
0 0%       5055 100%  0 0%
```

nfsstat -c 명령의 출력은 다음 표에 설명되어 있습니다.

표 30-4 nfsstat -c 명령의 출력

필드	설명
calls	보낸 총 호출 수입니다.
badcalls	RPC에서 거부된 총 호출 수입니다.

표 30-4 nfsstat -c 명령의 출력 (계속)

필드	설명
retrans	총 재전송 수입니다. 이 클라이언트에서는 재전송 수가 1% 미만이거나 6888 호출 중 약 10건의 시간 초과가 발생하였습니다. 이러한 재전송은 일시적인 오류로 인해 발생할 수 있습니다. 보다 높은 재전송률은 문제가 있음을 나타낼 수 있습니다.
badxid	단일 NFS 요청에서 수신된 중복 긍정 응답의 수입니다.
timeout	시간 초과된 호출 수입니다.
wait	클라이언트 핸들을 사용할 수 없기 때문에 기다려야 하는 호출 횟수입니다.
newcred	인증 정보를 새로 고침한 횟수입니다.
timers	호출에 대해 지정된 시간 초과 값보다 크거나 같은 시간 초과 값의 횟수입니다.
readlink	심볼릭 링크에 대해 수행된 read의 횟수입니다. 이 숫자가 큰 경우(10% 이상) 심볼릭 링크가 너무 많은 것일 수 있습니다.

다음 예에서는 nfsstat -m 명령의 출력을 보여줍니다.

```
pluto$ nfsstat -m
/usr/man from pluto:/export/svr4/man
Flags: vers=2,proto=udp,auth=unix,hard,intr,dynamic,
       rsize=8192, wsize=8192,retrans=5
Lookups: srtt=13 (32ms), dev=10 (50ms), cur=6 (120ms)
All:      srtt=13 (32ms), dev=10 (50ms), cur=6 (120ms)
```

밀리초로 표시되는 이 nfsstat -m 명령의 출력은 다음 표에서 설명합니다.

표 30-5 nfsstat -m 명령의 출력

필드	설명
srtt	라운드 트립 시간의 평균
dev	평균 분산
cur	현재 “예상되는” 응답 시간

네트워크의 하드웨어 구성 요소에서 문제가 발생하고 있다고 예상되는 경우 케이블과 커넥터를 주의 깊게 살펴봐야 합니다.

## 용어집

---

<b>asppp</b>	Solaris 2.4에서 Solaris 8 릴리스까지의 운영 체제에 포함된 PPP의 버전입니다. asppp는 비동기 PPP 통신만 지원합니다.
<b>CBCP</b> (콜백 제어 프로토콜)	콜백 세션을 협상하기 위해 사용되는 독점적 Microsoft PPP 확장자입니다. Solaris PPP 4.0은 이 프로토콜의 클라이언트(초기 호출자)측만 지원합니다.
<b>CCP</b> (압축 제어 프로토콜)	링크에서 데이터 압축 사용을 협상하는 PPP의 하위 프로토콜입니다. 헤더 압축과 달리 CCP는 링크에서 전송되는 패킷의 모든 데이터를 압축합니다.
<b>CHAP</b>	Challenge-Handshake 인증 프로토콜은 PPP 링크에서 호출자의 ID를 확인하는 데 사용할 수 있는 인증 프로토콜입니다. CHAP 인증은 <b>챌린지</b> 및 <b>응답</b> 의 개념을 사용하며 이때 호출을 수신하는 시스템이 호출자로 하여금 ID를 제공하도록 요구합니다.  <b>PAP(암호 인증 프로토콜)</b> 를 참조하십시오.
<b>CHAP 보안</b>	식별 목적으로 사용되며 PPP 링크에서 두 피어에 모두 알려지는 ASCII 또는 이진 문자열입니다. CHAP 보안은 시스템의 <code>/etc/ppp/chap-secrets</code> 파일에 일반 텍스트로 저장되지만 암호화된 형태여도 PPP 링크로 전송되지 않습니다. CHAP 프로토콜은 호출자가 사용하는 CHAP 보안의 해시가 수신자의 <code>/etc/ppp/chap-secrets</code> 파일에 있는 호출자에 대한 CHAP 보안 항목의 해시와 일치하는지 확인합니다.
<b>CSU/DSU</b>	CSU 및 DSU 장치를 결합하고 전용 회선 PPP 링크에서 사용되는 동기식 전자 통신 장치입니다. CSU/DSU는 피어에서 전용 회선으로 신호를 변환합니다. 대부분의 CSU/DSU는 링크를 설정하기 위해 채트 스크립트가 필요하지 않습니다. 종종 전용 회선 공급자가 CSU/DSU를 구성합니다.  <b>CSU(채널 서비스 장치)</b> 및 <b>DSU(데이터 서비스 장치)</b> 를 참조하십시오.
<b>CSU</b> (채널 서비스 장치)	전용 전자 통신 회선에 로컬 인터페이스를 제공하고 해당 회선을 종료하는 동기식 전자 통신 장치입니다. 미국에서는 CSU가 T1 회선을 종료하고 DS1 또는 DSX 인터페이스를 제공합니다. 기타 국가에서 대개 전자 통신 회사 공급자가 CSU를 소유합니다.  <b>CSU/DSU 및 DSU(데이터 서비스 장치)</b> 를 참조하십시오.
<b>DA</b> (디렉토리 에이전트)	SA(서비스 에이전트)가 전송하는 서비스 알림 캐시를 저장 및 유지 관리하는 선택적 SLP 에이전트입니다. 배치 후, DA는 UA(사용자 에이전트) 서비스 요청을 해결합니다. DA는 디렉토리 알림을 위한 SA와 UA의 활성 요청에 응답합니다. 따라서 SA와 UA는 연결된 DA와 <b>범위</b> 를 발견합니다. DA는 UA와 SA가 공유 범위에서 DA를 발견하는 요청하지 않은 알림을 주기적으로 보냅니다.

**DSU**  
(데이터 서비스 장치) 전용 회선 PPP 링크에서 사용되는 동기식 전자 통신 장치입니다. DSU는 전자 통신 회선에서 사용되며 표준 데이터 통신 인터페이스를 제공하는 데이터 프레임 형식입니다.

CSU(채널 서비스 장치) 및 CSU/DSU를 참조하십시오.

**expect-send** PPP 및 UUCP 채트 스크립트에 사용되는 스크립트 형식입니다. 채트 스크립트는 원격 피어로부터 *expect*할 텍스트나 명령으로 시작합니다. 다음 행에는 로컬 호스트가 피어에서 정확한 예상 문자열을 수신한 후 *sent*할 응답이 있습니다. 후속 행에서는 통신을 설정하는 데 필요한 모든 명령이 성공적으로 협상될 때까지 로컬 호스트와 피어 사이에 *expect-send* 명령을 반복합니다.

**IPCP**  
(인터넷 프로토콜 제어 프로토콜) 링크에서 피어의 IP 주소를 협상하는 PPP의 하위 프로토콜입니다. 또한 IPCP는 링크의 헤더 압축을 협상하며 네트워크 계층 프로토콜을 사용할 수 있도록 합니다.

**IPV6CP**  
(인터넷 프로토콜 버전 6 제어 프로토콜) **IPCP(인터넷 프로토콜 제어 프로토콜)**를 참조하십시오.

**ISDN TA**  
(터미널 어댑터) ISDN 네트워크에서 다이얼 업 PPP 링크를 위해 모뎀과 유사한 인터페이스를 제공하는 신호 적용 장치입니다. 표준 모뎀 구성에 사용하는 것과 같이 동일한 Solaris PPP 4.0 구성 파일을 사용하여 ISDN TA를 구성합니다.

**LCP**  
(링크 제어 프로토콜) 피어 간의 초기 링크 매개변수 세트를 협상하는 데 사용되는 PPP의 하위 프로토콜입니다. 링크 무결성 테스트가 LCP 기능 중 일부이므로 여러 링크 관련 문제가 LCP 실패로 매니페스트됩니다.

**MS-CHAP**  
(Microsoft CHAP) PPP용 독점 Microsoft 인증 프로토콜입니다. Solaris PPP 4.0에서는 클라이언트 모드와 서버 모드 모두에서 이 프로토콜의 버전 1과 2가 지원됩니다.

**PAP**  
(암호 인증 프로토콜) PPP 링크에서 호출자의 ID를 확인하는 데 사용할 수 있는 인증 프로토콜입니다. PAP는 링크를 통해 전달되는 일반 텍스트 암호를 사용하며 끝점 시스템 중 하나에 암호를 저장할 수 있게 해줍니다. 예를 들어, PAP는 호출을 수신하는 시스템의 UNIX *passwd* 데이터베이스에 있는 로그인 및 암호 항목을 사용하여 호출자의 ID를 확인합니다.

**CHAP**를 참조하십시오.

**PPP**  
(지점 간 프로토콜) 지점 간 매체를 통해 데이터그램을 전송하기 위한 표준 방법을 제공하는 데이터 링크 계층 프로토콜입니다. PPP 구성은 **피어**라고 하는 끝점 컴퓨터 두 대와 전화선 또는 통신을 위해 피어에 사용되는 다른 양방향 링크로 구성됩니다. 두 피어 사이의 하드웨어와 소프트웨어 연결이 **PPP 링크**입니다.

PPP는 PAP, CHAP, LCP 및 CCP를 포함하여 여러 하위 프로토콜로 구성됩니다. 여러 PPP 구현을 사용할 수 있습니다.

**PPPoE**  
(PPP over Ethernet) 호스트가 이더넷 링크를 통해 PPP 세션을 실행할 수 있게 해주는 RedBack Networks의 독점적 프로토콜입니다. PPPoE는 공통적으로 DSL(디지털 가입자 회선) 서비스와 함께 사용됩니다.

SA (서비스 에이전트)	네트워크 서비스를 위해 서비스 알림을 유지 관리하는 SLP 에이전트입니다. 사용 가능한 DA가 없을 경우 SA가 UA의 멀티캐스트 서비스 요청에 응답합니다. DA를 사용할 수 있는 경우 SA는 해당 범위를 지원하는 DA를 사용하여 서비스를 등록하고 선택적으로 등록 취소하기도 합니다.
SLP 데몬 (slpd)	SLP의 Oracle Solaris 구현에서 DA 또는 SA 서버 역할을 하는 데몬 프로세스입니다. 호스트의 서비스 프로세스는 알림을 개별적으로 유지 관리하는 대신 slpd를 사용하여 서비스 알림을 등록합니다. 데몬이 SA 서버로 구성되면 각 프로세스에 slpd와 통신하는 SA 클라이언트 라이브러리가 포함됩니다. SLP 데몬은 모든 등록과 등록 취소를 DA에 전달합니다. 데몬은 만료된 서비스 알림을 시간 초과로 설정하고, 활성 및 수동 DA 검색을 수행하여 사용 가능한 DA의 테이블을 유지 관리합니다. 이러한 방식을 통해 DA 정보가 UA 클라이언트에 제공됩니다. UA 클라이언트는 DA 정보만을 위해 호스트에서 slpd를 사용합니다. 선택적으로 slpd를 DA로 구성할 수 있습니다.
UA (사용자 에이전트)	사용자 응용 프로그램 대신 작동하는 SLP 에이전트입니다. 에이전트는 해당 범위, 디렉토리 에이전트 및 서비스 알림의 ID를 질의합니다.
다이얼 아웃 시스템	다이얼 업 PPP 링크를 설정하기 위한 호출을 시작하는 피어입니다. 구성 후, 다이얼 아웃 시스템은 다이얼 인 서버를 임의의 수만큼 호출할 수 있습니다. 다이얼 아웃 시스템은 대개 다이얼 업 링크가 설정되기 전에 인증 자격 증명을 제공합니다.
다이얼 업 PPP 링크	전화선이나 유사한 통신 매체(예: ISDN에서 제공하는 매체) 끝에서 피어 및 모뎀이 작동하는 PPP 연결입니다. “다이얼 업”이라는 용어는 피어의 전화 번호를 사용하여 로컬 모뎀이 원격 피어로 전화를 걸 때 링크 협상의 시퀀스를 말합니다. 다이얼 업 링크는 가장 일반적이고 비용이 적게 드는 PPP 구성입니다.
다이얼 인 서버	다이얼 아웃 시스템에서 호출을 받은 후 다이얼 업 PPP 링크의 수신자 끝을 협상하고 설정하는 피어입니다. “다이얼 인 서버”라는 용어는 공통으로 사용되지만 다이얼 인 서버는 클라이언트 서버 패러다임에 따라 작동하지 않습니다. 단순히 다이얼 업 링크를 설정하라는 요청에 응답하는 피어입니다. 구성 후, 다이얼 업 서버는 임의의 수만큼의 다이얼 아웃 시스템에서 호출을 수신할 수 있습니다.
동기식 PPP	데이터를 원시 비트의 지속적 스트림으로 전송하는 동기식 디지털 회선에서 실행되는 PPP 형식입니다. 전용 회선 PPP 링크는 동기식 PPP를 사용합니다.
레거시 서비스	SLP가 사용으로 설정되지 않은 네트워크 서비스입니다. 프로시 등록을 만들어 SLP를 사용하여 레거시 서비스를 등록할 수 있습니다. 그러면 SLP 기반 클라이언트가 레거시 서비스를 검색할 수 있습니다(10 장, “레거시 서비스 통합” 참조).
링크	PPP에서 두 피어 간에 협상 및 설정된 통신 연결입니다. Solaris PPP 4.0에서는 다이얼 업과 전용 회선이라는 두 가지 링크 유형이 지원됩니다.
멀티캐스트	데이터그램 패킷을 IP 네트워크의 여러 시스템으로 보내는 데 사용되는 네트워크 계층 프로시저입니다. 브로드캐스트 경로 지정이 있는 경우처럼 패킷은 모든 시스템에서 처리되지 않습니다. 멀티캐스트를 사용하려면 특수한 경로 지정 프로토콜로 라우터를 구성해야 합니다.
범위	관리를 위한 방식, 토폴로지별 방식 또는 다른 방식으로 배열되는 UA 및 SA의 그룹화입니다. 범위를 사용하여 기업에서 서비스에 대한 액세스를 제공하는 방법을 수정할 수 있습니다.

브로드캐스트	서브넷에서 모든 시스템에 패킷을 전송하는 데 사용되는 데이터 링크 계층 프로시저입니다. 일반적으로 브로드캐스트 패킷은 서브넷을 벗어나 경로가 지정되지 않습니다.
비동기 PPP	한 번에 한 문자씩 전송하는 비동기 직렬 회선에서 실행되는 PPP 형식입니다. 가장 일반적인 형식의 PPP인 다이얼 업 링크에는 비동기 PPP 통신이 사용됩니다.
서비스 URL	서비스의 네트워크 위치를 알리는데 사용되는 URL입니다. URL에는 서비스 유형, 호스트 이름 또는 서비스 호스트의 네트워크 주소가 포함됩니다. URL은 포트 번호 및 서비스 사용에 필요한 기타 정보도 포함합니다.
서비스 알림	서비스를 설명하는 SA에 의해 배포되는 정보입니다. 서비스 알림은 URL 및 서비스를 설명하는 속성/값 목록 쌍의 모음으로 구성됩니다. 모든 서비스 알림에는 수명이 있습니다. 서비스 알림은 수명이 만료된 후 등록하지 않을 경우 더 이상 유효하지 않습니다.
신뢰할 수 있는 호출자	PPP에서 다이얼 인 서버가 서버의 PAP 또는 CHAP 보안 데이터베이스에 있는 피어의 보안 자격 증명을 포함시켜 액세스를 부여하는 원격 피어입니다.
인증	원격 사용자 또는 프로그램 등의 엔티티를 통해 네트워크에서 제공되는 ID를 검증하는 작업입니다. 일부 인증 프로토콜을 사용하면 잠재적 사용자의 인증 자격 증명 데이터베이스를 구축할 수 있습니다. 다른 인증 프로토콜은 인증 용도로 인증 기관에서 생성한 트러스트의 인증서 체인을 사용합니다. 이러한 자격 증명은 관리자나 통신하거나 관리자 사이트의 서비스를 사용하는 사용자를 인증할 수 있습니다.
전용 회선 PPP 링크	공급자가 임대한 동기식 네트워크 매체에 연결된 호스트 및 CSU/DSU를 포함하는 PPP 연결입니다. OC3 및 T1은 전용 회선 매체의 일반적인 예입니다. 전용 회선 링크는 다이얼 업 PPP 링크에 비해 관리가 쉬워도 비용이 많이 들기 때문에 일반적이지 않습니다.
채트 스크립트	원격 피어와의 사이에 통신 링크를 설정하는 방법을 모뎀에 지시하는 명령입니다. PPP 및 UUCP 프로토콜 모두 다이얼 업 링크와 다이얼 백 호출을 설정하기 위해 채트 스크립트를 사용합니다.
피어	PPP에서 PPP 통신 링크의 한 쪽 끝에 있는 개별 컴퓨터이며 통신 매체로 연결되는 피어 두 개로 구성됩니다. 워크스테이션, PC, 라우터, 메인프레임 등의 여러 유형의 컴퓨팅 장비를 피어로 구성할 수 있습니다.
확장된 계정	작업 또는 프로세스에 따라 자원 소비를 기록하는 유연한 방법입니다.



# 색인

---

## 번호와 기호

- (대시)
  - autofs 맵 이름, 209
  - Line2 필드 개체 틀, 536
  - Speed 필드 개체 틀, 530
  - 다이얼 번호 약어, 530
- + (더하기 기호)
  - autofs 맵 이름, 209, 210
- = (등호), 다이얼 번호 약어, 530
- \* (별표), autofs 맵, 213
- / (슬래시)
  - /- as 마스터 맵 마운트 지점, 196, 199
  - 루트 디렉토리
    - 디스크가 없는 클라이언트를 통한 마운트, 74
  - 마스터 맵 이름 앞에 붙음, 196
- & (앰퍼센드), autofs 맵, 212
- .(점)
  - rcp 명령 구문, 592, 593
- ~ (틸드)
  - rcp 명령 구문, 592, 593
  - 축약 경로 이름, 590
- # (파운드 기호)
  - 간접 맵의 주석, 199
  - 마스터 맵(auto\_master)의 주석, 196
  - 직접 맵의 주석, 198
- + (플러스 기호)
  - /etc/hosts.equiv 파일 구문, 577
- ~/ .ftpaccess 파일, 설명, 569
- 8.12의 명령줄 옵션, sendmail 명령, 360
- 8진수 제어 문자, 542

## A

- Ac 옵션, sendmail 명령, 359
- Am 옵션, sendmail 명령, 359
- a option, showmount 명령, 164
- a 옵션, umount 명령, 153
- ACU(자동 호출 단위)
  - UUCP 하드웨어 구성, 511
  - 문제 해결, 523
- ACU(자동 호출 장치), Devices 파일 Type 필드, 535
- aliases.db 파일, 297, 330
- aliases.dir 파일, 297, 330
- aliases.pag 파일, 297, 330
- aliases 파일, 330, 522
- anon 옵션, share 명령, 159
- ANU(Australian National University) PPP, Solaris PPP
  - 4.0과의 호환성, 376
- Any Time 필드 항목, 528
- Any 키워드
  - Grades 파일(UUCP), 556, 557
  - Speed 필드(UUCP), 530
- ARCH 맵 변수, 208
- asppp, 참조 비동기 PPP(asppp)
- asppp2pppd 변환 스크립트
  - Solaris PPP 4.0으로 변환, 508-509
  - Solaris PPP 4.0으로 변환된 파일 보기, 509
  - 표준 asppp 구성, 505
- ASSERT 오류 메시지(UUCP), 525, 560, 561
- asyncmap 옵션(PPP), 471
- auth 옵션(PPP), 429
- auto\_direct 파일, 280
- auto\_home 맵
  - /home 디렉토리, 107

## auto\_home 맵 (계속)

/home 디렉토리 서버 설정, 107

/home 마운트 지점, 195, 196

## autofs

/home 디렉토리, 107

NFS URL 및, 112

개요, 73

공용 파일 핸들 및, 112

공유 네임스페이스 액세스, 110

기능, 80

네임스페이스 데이터, 80

마운트 프로세스, 203, 204

마운트 해제 프로세스, 204

## 맵

CD-ROM 파일 시스템, 106

hsfs 옵션, 106

PC-DOS 파일 시스템, 106

pcfs 옵션, 106

types, 102

간접, 199, 200

네트워크 탐색, 202

다른 맵 참조, 209, 210

마스터, 195, 196

변수, 208, 209

일기 전용 파일 선택, 205

읽기 전용 파일 선택, 207

직접, 197, 198

찾아보기 기능 및, 80

탐색 프로세스 시작, 196, 203

맵 관리, 102

메타 문자, 212, 213

문제 해결, 122

비 NFS 파일 시스템 액세스, 105, 106

시작, 92-93

여러 서버에서 공유 파일 복제, 111

운영 체제

호환되지 않는 버전 지원, 111

중지, 93

참조, 212, 213

찾아보기 기능, 80, 112

특수 문자, 213

파일 시스템 마운트, 87

프로젝트 관련 파일 통합, 108

홈 디렉토리 서버 설정, 107

automount 명령, 146-147

autofs 마스터 맵 수정(auto\_master), 104

autofs 및, 73

-v 옵션, 122

개요, 201

실행해야 하는 경우, 103

오류 메시지, 122

automountd 데몬, 135

autofs 및, 73

description, 80

mounting and, 80

개요, 201

## B

-bP 옵션, sendmail 명령, 359

b 제어 문자, Dialers 파일, 542

bad argument specified with index option, 125

bg 옵션, mount 명령, 149

Break 제어 문자, Dialers 파일, 542

bye 명령(FTP), 585

## C

C. UUCP 작업 파일

설명, 559, 560

정리, 520

c 제어 문자, Dialers 파일, 542

call 옵션(PPP), 다이얼 인 서버 호출, 416

CD-ROM 응용 프로그램, autofs를 사용하여  
액세스, 106

CHAP(Challenge-Handshake 인증 프로토콜)

/etc/ppp/chap-secrets의 구문, 489

구성 작업 맵, 433-434

예제 구성, 398

인증 프로세스, 492

정의, 489

CHAP 자격 증명 데이터베이스

만들기

다이얼 인 서버, 435

신뢰할 수 있는 호출자, 437

Chat Script 필드, /etc/uucp/Systems 파일, 531

Chat Script 필드의 expect 필드, 531

check\_eoh 규칙 세트, sendmail 명령, 370  
 check\_etrn 규칙 세트, sendmail 명령, 371  
 check\_expn 규칙 세트, sendmail 명령, 371  
 check-hostname 스크립트, 282, 283, 334  
 check-permissions 스크립트, 334  
 check\_vrfy 규칙 세트, sendmail 명령, 371  
 Class 필드, Devices 파일, 536  
 clear\_locks 명령, 147  
 client\_versmax 매개변수, 137  
 client\_versmin parameter, 137  
 clientmqueue 디렉토리, 334  
 COMMANDS 옵션의 ALL 값, 551  
 compat\_check FEATURE() 선언, 364  
 confFORWARD\_PATH 정의, 305  
 connect 옵션 (PPP)  
   예, 411  
   채트 스크립트 호출, 484  
 CPU 맵 변수, 208  
 crontab 파일, UUCP용, 519  
 crtscts 옵션 (PPP), 409  
 CSU/DSU  
   구성, 420  
   일반적인 문제 해결, 463  
   정의, 383  
 cu 명령  
   Systems 목록 인쇄, 546  
   모뎀 또는 ACU 확인, 523  
   설명, 514  
   여러 또는 다른 구성 파일, 545  
   여러 파일 또는 서로 다른 구성 파일, 515

## D

D. UUCP 데이터 파일, 정리, 520  
 D 제어 문자, 539  
 d 제어 문자, Dialers 파일, 542  
 -d 옵션  
   cu 명령, 523  
   showmount 명령, 164  
 DA\_BUSY\_NOW, 248  
 DA(SLP)  
   DA 로깅, 246  
   검색, 230, 234, 243  
   다이얼 업 네트워크 검색, 231, 233

DA(SLP) (계속)  
   다중 DA, 248-249  
   멀티캐스트, 234  
   멀티캐스트 없음, 249  
   멀티캐스트 제거, 231  
   배포, 234, 245-246  
   수동 검색 사용 안함으로 설정, 231  
   알림, 230, 231, 233, 234  
   제거, 233  
   하트비트, 233, 234, 235  
   활성 검색 사용 안함으로 설정, 231  
 DA 검색 (SLP), 239  
 DA 하트비트, 빈도, 230  
 delay\_checks FEATURE() declaration, 364  
 /dev/nca 파일, NCA 및, 57  
 Devconfig 파일  
   설명, 514, 557  
   형식, 557  
 Devices 파일  
   Class 필드, 536  
   Dialer-Token-Pairs 필드, 537, 539  
   Line 필드, 536  
   Line2 필드, 536  
   Systems 파일 Speed 필드 및, 530  
   Systems 파일 Type 필드 및, 535  
   Type 필드, 534  
   설명, 514, 534  
   여러 또는 다른 파일, 545  
   프로토콜 정의, 539, 540  
   형식, 534  
 Devices 파일의 e 프로토콜, 539  
 Devices 파일의 f 프로토콜, 539  
 Devices 파일의 g 프로토콜, 539  
 Devices 파일의 Line 필드, 536  
 Devices 파일의 Line2 필드, 536  
 Devices 파일의 t 프로토콜, 539  
 Devices 파일의 포트 선택기 변수, 535  
 Devices 파일의 프로토콜 정의, 539, 540  
 DH 인증  
   개요, 191  
   보안 NFS 및, 96  
   사용자 인증, 189  
   암호 보호, 190  
 Dialcodes 파일, 514, 544

## Dialer-Token-Pairs 필드

## Devices 파일

구문, 537

동일한 포트 선택기, 538

전화 걸기 유형, 537

포트 선택기 연결, 538

## Dialers 파일

설명, 514, 540

예, 541

Dialers 파일의 penril 항목, 542

DNS 레코드, 통합 파일 시스템, 91

DNS 이름 서비스, sendmail 프로그램 및, 284

dnsbl FEATURE() 선언, 364, 366

domain 디렉토리, 332

DOS 파일, autofs를 사용하여 액세스, 106

drift 파일, 65

DSL, 참조 PPPoE

DSL 모뎀, 388

DSLAM(Digital Subscriber Line Access Multiplexer),

PPPoE, 388

DTP 필드의 direct 키워드, 537

DTP 필드의 uudirect 키워드, 537

## E

E 제어 문자, Dialers 파일, 542

e 제어 문자, Dialers 파일, 542

-e 옵션, showmount 명령, 164

editmap 명령, 334

enhdnsbl FEATURE() 선언, 364, 366

error 디렉토리(UUCP), 525

/etc/asppp.cf 구성 파일, 505

/etc/auto\_direct 파일, 280

/etc/default/autofs 파일, autofs 환경 구성, 102

/etc/default/nfslogd 파일, 132-133

/etc/default/sendmail 파일, 343

/etc/ftpd/ftpusers 파일, 설명, 569

/etc/hostname.interface 파일, NCA 및, 57

/etc/hosts.equiv 파일, 577

/etc/hosts 파일, 57, 276, 277

/etc/inet/ntp.client 파일, 64

/etc/inet/ntp.conf 파일, 64

/etc/inet/ntp.keys 파일, 65

/etc/inet/ntp.leap 파일, 65

/etc/inet/ntp.server 파일, 65

/etc/inet/services 파일, UUCP 확인, 521

/etc/inet/slp.conf 파일

DA 배포, 247

DA 알림, 232

DA 하트비트, 233

SA 재등록, 235

개요, 223

구성 변경, 229

로드 균형 조정, 248

멀티캐스트 활성 시간, 236

브로드캐스트 전용 경로 지정, 238

새 범위, 242, 244

시간 초과, 240

요소, 228

인터페이스 변경, 251

임의 대기 한도, 241

정적 DA, 231

패킷 크기, 237

프록시 등록, 257

/etc/init.d/ncakmod 스크립트, 58

/etc/init.d/ncalogd 스크립트, 58

/etc/init.d/slpsd 스크립트, 257

/etc/mail/aliases.db 파일, 297, 330

/etc/mail/aliases.dir 파일, 297, 330

/etc/mail/aliases.pag 파일, 297, 330

/etc/mail/aliases 파일, 324, 330, 340, 341

UUCP 및, 522

/etc/mail/cf/cf/main.cf 파일, 331

/etc/mail/cf/cf/main.mc 파일, 332

/etc/mail/cf/cf/Makefile 파일, 332

/etc/mail/cf/cf/sendmail.mc 파일, 332

/etc/mail/cf/cf/submit.cf 파일, 332

/etc/mail/cf/cf/submit.mc 파일, 332

/etc/mail/cf/cf/subsidiary.cf 파일, 332

/etc/mail/cf/cf/subsidiary.mc 파일, 332

/etc/mail/cf/domain/generic.m4 파일, 332

/etc/mail/cf/domain/solaris-antispam.m4  
파일, 332/etc/mail/cf/domain/solaris-generic.m4  
파일, 332

/etc/mail/cf/domain 디렉토리, 332

/etc/mail/cf/feature 디렉토리, 332

/etc/mail/cf/m4 디렉토리, 333

- /etc/mail/cf/mailler 디렉토리, 333
- /etc/mail/cf/main-v7sun.mc 파일, 333
- /etc/mail/cf/ostype/solaris2.m4 파일, 333
- /etc/mail/cf/ostype/solaris2.ml.m4 파일, 333
- /etc/mail/cf/ostype/solaris2.pre5.m4 파일, 333
- /etc/mail/cf/ostype/solaris8.m4 파일, 333
- /etc/mail/cf/ostype 디렉토리, 333
- /etc/mail/cf/README 파일, 331
- /etc/mail/cf/sh/check-hostname 스크립트, 334
- /etc/mail/cf/sh/check-permissions 스크립트, 334
- /etc/mail/cf/subsidiary-v7sun.mc 파일, 333
- /etc/mail/cf 디렉토리, 내용, 331
- /etc/mail/helpfile 파일, 331, 371
- /etc/mail/local-host-names 파일, 331, 372
- /etc/mail/Mail.rc 파일, 330
- /etc/mail/mailx.rc 파일, 330
- /etc/mail/main.cf 파일, 330
- /etc/mail/relay-domains 파일, 330
- /etc/mail/sendmail.cf 파일, 330
- /etc/mail/sendmail.ct 파일, 372
- /etc/mail/sendmail.cw 파일, 372
- /etc/mail/sendmail.hf 파일, 371
- /etc/mail/sendmail.pid 파일, 331
- /etc/mail/statistics 파일, 331
- /etc/mail/submit.cf 파일, 331, 358
- /etc/mail/subsidiary.cf 파일, 275, 331
- /etc/mail/trusted-users 파일, 331, 372
- /etc/mail 디렉토리, 내용, 330
- /etc/mnttab 파일
  - auto\_master 맵과 비교, 201
  - 만들기, 164
- /etc/nca/nca.if 파일, 58
- /etc/nca/ncakmod.conf 파일, 58
- /etc/nca/ncalogd.conf 파일, 58
- /etc/nca/ncaport.conf 파일, 58
- /etc/netconfig 파일, 설명, 132
- /etc/nfs/nfslog.conf 파일, 133-134
- /etc/nsswitch.conf 파일, 576
- /etc/passwd 파일
  - ftp 및, 583
  - UUCP 로그인 사용, 518
- /etc/ppp/chap-secrets 파일
  - 구문, 489
- /etc/ppp/chap-secrets 파일 (계속)
  - 만들기
    - 신뢰할 수 있는 호출자, 437
  - 예, PPPoE 액세스 서버, 501
  - 정의, 466
  - 주소 지정
    - sppp 장치 번호별, 494
  - 정적, 493
- /etc/ppp/myisp-chat.tmpl 템플리트, 478-479
- /etc/ppp/options.tmpl 템플리트, 470
- /etc/ppp/options.ttya.tmpl 템플리트, 472
- /etc/ppp/options.ttyname 파일
  - 권한, 468
  - 다이얼 아웃 시스템, 409, 472
  - 다이얼 인 서버, 415, 471
  - 동적 주소 지정, 492
  - 예 목록, 472
  - 정의, 466, 471
- /etc/ppp/options 파일
  - CHAP 인증을 위한 name 옵션, 436
  - /etc/ppp/options.tmpl 템플리트, 470
  - PAP 인증을 위해 수정, 432
  - 권한, 468
  - 만들기
    - 다이얼 아웃 시스템, 408-409
    - 다이얼 인 서버, 415
  - 예 목록, 470
  - 예제 PPPoE, 501
  - 정의, 466, 469
- /etc/ppp/pap-secrets 파일
  - 구문, 486
  - 만들기
    - PPPoE 액세스 서버, 445
    - 다이얼 인 서버, 428
  - 신뢰할 수 있는 호출자에 대해 만들기, 431
  - 예, PPPoE 액세스 서버, 501
  - 정의, 466
  - 주소 지정
    - sppp 장치 번호별, 494
  - 정적, 493
- /etc/ppp/peers/myisp.tmpl 템플리트, 475
- /etc/ppp/peers/peer-name 파일
  - 권한, 468

**/etc/ppp/peers/peer-name 파일 (계속)**

만들기

전용 회선 링크의 끝점, 422

수정

PAP 인증, 432

PPPoE 클라이언트, 441

예, PPPoE 클라이언트, 502

예 목록, 476

유용한 옵션, 474

정의, 466, 474-475

**/etc/ppp/peers 디렉토리, 466****/etc/ppp/pppoe.device 파일**

구문, 499

액세스 서버, 445

정의, 499

**/etc/ppp/pppoe.if 파일**

만들기

PPPoE 클라이언트, 441

액세스 서버, 443

예, 495

정의, 495

**/etc/ppp/pppoe 파일**

구문, 497

목록 서비스, 443

수정, 444

예, 498, 500

**/etc/proftpd.conf 파일, 설명, 569****/etc/services 파일, nfsd 항목, 125****/etc/shells 파일, 306****/etc/shutmsg 파일, 설명, 569****/etc/syslog.conf 파일, 310****/etc/uucp/Config 파일**

설명, 514, 554

형식, 554

**/etc/uucp/Devconfig 파일**

설명, 514, 557

형식, 557

**/etc/uucp/Devices 파일**

Class 필드, 536

Dialer-Token-Pairs 필드, 537, 539

Line 필드, 536

Line2 필드, 536

Systems 파일 Speed 필드 및, 530

Systems 파일 Type 필드 및, 535

**/etc/uucp/Devices 파일 (계속)**

Type 필드, 534

설명, 514, 534

예, asppp 구성, 507

프로토콜 정의, 539, 540

형식, 534

**/etc/uucp/Dialcodes 파일, 514, 544****/etc/uucp/Dialers 파일**

설명, 514, 540

예, 541

예, asppp 구성, 507

**/etc/uucp/Grades 파일**

ID-list 필드, 556, 557

Job-size 필드, 556

keywords, 556

Permit-type 필드, 556

System-job-grade 필드, 555, 556

User-job-grade 필드, 555

기본 등급, 556

설명, 515, 555

키워드, 556

**/etc/uucp/Limits 파일**

설명, 515, 558

형식, 558

**/etc/uucp/Permissions 파일**

COMMANDS 옵션, 550, 551, 554

LOGNAME

MACHINE과 결합, 553

설명, 547

원격 컴퓨터의 로그인 ID, 547

MACHINE

LOGNAME과 결합, 553

OTHER 옵션, 553

기본 사용 권한 또는 제한 사항, 547

설명, 547

MYNAME 옵션, 548

NOREAD 옵션, 550

NOWRITE 옵션, 550

OTHER 옵션, 553

READ 옵션, 549

REQUEST 옵션, 548

SENDFILES 옵션, 548

uuccheck 명령 및, 513

uuxqt 데몬 및, 512

**/etc/uucp/Permissions 파일 (계속)**

VALIDATE 옵션, 552, 553  
 WRITE 옵션, 549  
 고려 사항, 547  
 구성 항목, 547  
 노드 이름 변경, 548  
 다이얼 백 권한, 550  
 보안 설정, 522  
 설명, 515, 546  
 원격 실행 권한, 550, 553  
 전달 작업, 554  
 콜백 옵션, 550  
 파일 전송 권한, 548, 550  
 형식, 547

**/etc/uucp/Poll 파일**

설명, 515, 554  
 형식, 554

**/etc/uucp/Sysfiles 파일**

Systems 목록 인쇄, 546  
 샘플, 546  
 설명, 515, 545  
 형식, 545

**/etc/uucp/Sysname 파일, 515, 546****/etc/uucp/Systems 파일**

Chat Script 필드, 531, 533  
 Devices 파일 Class 필드 및, 536  
 Devices 파일 Type 필드 및, 535  
 Phone 필드, 530  
 Speed 필드, 530  
 System-Name 필드, 528  
 TCP/IP 구성, 521  
 Time 필드  
   Never 항목, 548  
   설명, 528  
 Type 필드, 529  
 다이얼-코드 약어, 514  
 문제 해결, 525  
 설명, 515, 527  
 여러 또는 다른 파일, 527, 545  
 여러 파일 또는 서로 다른 파일, 515  
 예, asppp 구성, 506  
 제어 문자, 532  
 패리티 설정, 533  
 하드웨어 흐름 제어, 533

**/etc/uucp/Systems 파일 (계속)**

형식, 527

**/etc/vfstab 파일**

automount 명령 및, 201  
 NFS 서버 및, 86  
 nolargefiles 옵션, 89  
 디스크가 없는 클라이언트를 통한 마운트, 74  
 부트 시에 파일 시스템 마운트, 86  
 클라이언트측 파일오버 사용으로 설정, 89

etrn 스크립트, 334

exit 명령, 582, 583

**F**

-F 옵션, unshareall 명령, 163

feature 디렉토리, 332

fg 옵션, mount 명령, 149

find 명령, .rhosts 파일 검색, 580

forcedirectio 옵션, mount 명령, 149

.forward+detail 파일, 343

.forward.hostname 파일, 343

.forward 파일

  검색 경로 변경, 305

  관리, 304

  사용 안함, 304

  사용자용, 342

ftp 명령

  rlogin 및 rcp와 원격 로그인 비교, 583

  로그인 인터럽트, 576

ftp 명령, 설명, 568

ftp 명령

  원격 로그인 인증, 583

  원격 시스템 연결 열기, 584, 585

ftp 세션

  원격 시스템 연결 닫기, 585

  원격 시스템 연결 열기, 585

  익명 ftp 계정, 583

  파일 복사

    원격 시스템에서, 585

    원격 시스템으로, 587

ftp 아카이브, WebNFS 및, 99

ftp 하위 명령, 설명, 584

ftpcount 명령, 설명, 568

ftpdctl 명령, 설명, 568

ftprestart 명령, 설명, 568  
 ftpscrub 명령, 설명, 569  
 ftpshut 명령, 설명, 569  
 ftptop 명령, 설명, 568  
 ftpusers 파일, 설명, 569  
 ftpwho 명령, 설명, 568  
 fuser 명령, umountall 명령 및, 155

## G

-G 옵션, sendmail 명령, 360  
 -g 옵션, lockd 데몬, 136  
 gen-etc-shells 스크립트, 306  
 generic.m4 파일, 332  
 generics\_entire\_domain FEATURE() 선언, 364  
 genericstable FEATURE() 선언, 366  
 get 명령(FTP), 예, 586  
 gethostbyname 명령, 347  
 grace\_period 매개변수, lockd 데몬, 136  
 Grades 파일  
   ID-list 필드, 556, 557  
   Job-size 필드, 556  
   Permit-type 필드, 556  
   System-job-grade 필드, 555, 556  
   User-job-grade 필드, 555  
   기본 등급, 556  
   설명, 515, 555  
   키워드, 556  
 Grades 파일의 ID-list 필드, 556, 557  
 Grades 파일의 Job-size 필드, 556  
 Grades 파일의 Permit-type 필드, 556  
 Grades 파일의 System-job-grade 필드, 555, 556  
 Grades 파일의 User-job-grade 필드, 555  
 GSS-API, 및 NFS, 79

## H

-h 옵션, umountall 명령, 155  
 hard 옵션, mount 명령, 151  
 helpfile 파일, 331  
   sendmail 명령, 371  
 /home 디렉토리 및 NFS 서버 설정, 107  
 /home 마운트 지점, 195, 196

HOST 맵 변수, 208  
 hostname.interface 파일, NCA 및, 57  
 hosts.equiv 파일, 577  
 hosts 파일, 57  
 hsf s 옵션, autofs 맵, 106  
 HTML 파일, WebNFS 및, 99  
 httpd 명령  
   NCA 및, 58-59  
   방화벽 액세스 및 WebNFS, 100

## I

ICMP 프로토콜, 600  
 ID 매핑 실패, 이유, 181  
 IGMP 프로토콜, 600  
 in.comsat 데몬, 334  
 in.uucpd 데몬, 513  
 index 옵션  
   (share 명령 포함), 83  
   WebNFS 및, 99  
   잘못된 인수 오류 메시지, 125  
 inetd 데몬, in.uucpd 호출, 513  
 init 명령, PPP 및, 422  
 -intr 옵션, mount 명령, 116  
 IP 경로 지정 테이블, 601  
 IPv6 주소 및 버전 8.12, sendmail 명령, 372

## K

K 제어 문자, Dialers 파일, 542  
 -k 옵션, umountall 명령, 155  
 KERB 인증, NFS 및, 78  
 /kernel/fs 파일, 확인, 132  
 keylogin 명령, 원격 로그인 보안 문제, 192  
 keylogout 명령, 보안 NFS 및, 192

## L

-L tag 옵션, sendmail 명령, 360  
 -l 옵션  
   cu 명령, 523  
   umountall 명령, 155



LAN(Local Area Network), UUCP 구성, 512  
 largefiles 옵션  
   mount 명령, 149  
   오류 메시지, 128  
 LCK UUCP 잠금 파일, 559  
 ldap\_routing FEATURE() 선언, 364  
 leap 파일, NTP, 65  
 libslp.so 라이브러리, 220  
 Limits 파일  
   설명, 515, 558  
   형식, 558  
 LOCAL\_DOMAIN() m4 구성 매크로, 363  
 local-host-names 파일, 331, 372  
 local\_lmtp FEATURE() 선언, 364  
 local\_no\_masquerade FEATURE() 선언, 365  
 local 옵션(PPP), 422  
 LOCKD\_GRACE\_PERIOD 매개변수, lockd 데몬, 136  
 lockd\_retransmit\_timeout 매개변수, lockd 데몬, 136  
 lockd\_servers 매개변수, lockd 데몬, 136  
 lockd 데몬, 136  
 log 옵션, share 명령, 159  
 login 명령, 보안 NFS 및, 192  
 login 옵션(PPP)  
   /etc/ppp/options 다이얼 인 서버, 429  
   /etc/ppp/pap-secrets, 432, 489  
 LOGNAME Permissions 파일  
   MACHINE과 결합, 553  
   SENDFILES 옵션, 548  
   VALIDATE 옵션, 552, 553  
   설명, 547  
   원격 컴퓨터의 로그인 ID, 547  
 lookupdotdomain FEATURE() 선언, 365  
 ls 명령, ACL 항목 및, 181

## M

m4 디렉토리, 333  
 MACHINE Permissions 파일  
   COMMANDS 옵션, 551  
   LOGNAME과 결합, 553  
   OTHER 옵션, 553  
   기본 사용 권한 또는 제한 사항, 547  
   설명, 547

Mail.rc 파일, 330  
 mail 명령, 329  
 mailboxes, 파일, 334  
 mailcompat 필터, 329  
 MAILER-DAEMON 메시지, 311  
 mailer 디렉토리, 333  
 mailq 명령, 329  
 .mailrc 별칭, 340  
 .mailrc 파일, 326  
 mailstats 명령, 330  
 mailx.rc 파일, 330  
 mailx 명령, 330  
 main.cf 파일, 330, 331, 338  
 main.mc 파일, 332, 371  
 main-v7sun.mc 파일, 333, 371  
 Makefile 파일, 332  
 makemap 명령, 334  
 MASQUERADE\_EXCEPTION() m4 구성 매크로, 363  
 MAXBADCOMMANDS macro, sendmail command, 362  
 MAXETRNCOMMANDS macro, sendmail command, 363  
 MAXHELOCOMMANDS macro, sendmail command, 362  
 MAXNOOPCOMMANDS macro, sendmail command, 362  
 MAXVRFYCOMMANDS macro, sendmail command, 363  
 mconnect 명령, 309-310, 330  
 mget 명령(FTP), 예, 586  
 MILTER, 메일 필터 API, 317  
 mnttab 파일  
   auto\_master 맵과 비교, 201  
   만들기, 164  
 mount 명령, 148-153  
   (NFS URL 포함), 90  
   autofs 및, 74  
   NFS URL, 152  
   디스크가 없는 클라이언트의 요구, 74  
   사용, 151  
   옵션  
   nolargefiles, 88  
   public, 90  
   설명, 148-151  
   인수 없음, 153  
   큰 파일 생성을 사용 안함으로 설정, 88  
   파일 시스템 수동 마운트, 86  
   페일오버, 152  
 mountall 명령, 154-155

- mountd 데몬, 137
  - rpcbind로 등록되지 않음, 127
  - 서버의 응답 확인, 119
  - 실행 중인지 확인, 120, 127
- mput 명령(FTP), 예, 588
- mqueue 디렉토리, 334
- MS-DOS 파일, autofs를 사용하여 액세스, 106
- MX(메일 교환기) 레코드, 284
  
- N**
- N 제어 문자, Dialers 파일, 542
- n 제어 문자, Dialers 파일, 542
- name 옵션(PPP)
  - CHAP 인증, 436
  - /etc/ppp/pap-secrets, 432
  - noservice 사용, 501
- names/naming
  - 노드 이름
  - UUCP 별칭, 515
- NCA
  - httpd 및, 58-59
  - 개요, 45-46
  - 구조, 58-59
  - 로깅 변경, 52
  - 사용 안함으로 설정, 51
  - 사용으로 설정, 48-51
  - 새 기능, 46
  - 소켓, 48
  - 소켓 라이브러리, 52
  - 요구 사항, 47-48
  - 작업 목록, 47
  - 커널 모듈, 58-59
  - 파일 설명, 57
- nca\_addr.so 라이브러리, 58
- nca\_httpd\_1.door 파일, 58
- nca.if 파일, 49, 58
- NCA 로그 파일, 58
- ncab2clf 명령, 58
- ncaconfd 명령, 58
- ncakmod.conf 파일, 49, 51, 58
- ncakmod 모듈, 58-59
- ncalogd.conf 파일, 49, 51, 58
- ncalogd 스크립트, 58
- ncaport.conf 파일, 58
- NCA에 대한, 로그 파일, 58
- net.slp.DAActiveDiscoveryInterval 등록 정보, 231
  - 정의, 230
- net.slp.DAAddresses 등록 정보, 233, 244, 248
  - 정의, 230
- net.slp.DAAttributes 등록 정보, 235
- net.slp.DAHeartBeat 등록 정보, 234, 235
  - 정의, 230
- net.slp.interfaces 등록 정보
  - DA 및, 248
  - 경로가 지정되지 않은 인터페이스 및, 253
  - 구성, 250
  - 멀티홈 호스트 및, 253
  - 인터페이스 변경, 252
- net.slp.isBroadcastOnly 등록 정보, 238, 249, 250
- net.slp.isDA 등록 정보, 229
- net.slp.MTU 등록 정보, 237
- net.slp.multicastTTL 등록 정보, 235
- net.slp.passiveDADetection 등록 정보, 231
  - 정의, 230
- net.slp.randomWaitBound 등록 정보, 241
- net.slp.serializedRegURL 등록 정보, 256
- net.slp.useScopes 등록 정보, 243-244, 244, 258
  - 정의, 242
- /net 마운트 지점, 197
- netconfig 파일, 설명, 132
- netstat 명령, 224, 599, 601
  - i 옵션(인터페이스), 599, 600
  - r 옵션(IP 경로 지정 테이블), 601
  - s 옵션(프로토콜당), 600
  - 개요, 597, 599
- networks
  - 문제 해결
    - 하드웨어 구성 요소, 604
  - 패킷
    - 네트워크에서 캡처, 597
- Never Time 필드 항목, 548
- newaliases 링크, 334
- newaliases 명령, UUCP 및, 522
- NFS
  - 데몬, 134-146
  - 명령, 146
  - 버전 협상, 172-173

- NFS ACL
  - 설명, 76, 181-182
  - 오류 메시지, Permission denied, 128
- NFS URL
  - autofs 및, 112
  - mount 명령 예제, 152
  - WebNFS 및, 98
  - 구문, 99-100
  - 마운트, 79
  - 파일 시스템 마운트, 90
- NFS V2에서 largefiles를 지원할 수 없음 메시지, 128
- NFS 관리, 관리자 작업, 82
- NFS 마운트된 파일 시스템
  - 메일 서비스 및, 278
  - 메일 클라이언트 및, 278, 280
- NFS 문제 해결
  - NFS 서비스가 실패한 위치 확인, 120
  - 서버 문제, 117
  - 원격 마운트 문제, 127
  - 전략, 116
  - 정지된 프로그램, 128
- NFS 버전 4, 기능, 173-182
- NFS 서버
  - autofs 파일 선택, 207
  - 공유 파일 복제, 111
  - 맵의 가중치, 208
  - 문제 해결
    - 문제 해결, 117
    - 원격 마운트 문제, 117, 127
    - 원격 마운트에 데몬 필요, 116
    - 유지 관리, 82
    - 현재 항목 식별, 121
- NFS 서버 로깅
  - 개요, 79
  - 사용으로 설정, 84
- NFS 서비스
  - 다시 시작, 120-121
  - 서버에서 다른 버전 선택, 93-94
  - 시작, 92
  - 작업 맵, 91
  - 중지, 92
  - 클라이언트에서 다른 버전 선택
    - mount 명령 사용, 95-96
- NFS 서비스, 클라이언트에서 다른 버전 선택 (계속)
  - SMF 등록 정보 변경, 94-95
- NFS 잠금 및, 클라이언트측 페일오버 및, 186
- NFS 참조
  - 개요, 194-195
  - 만들기, 115
  - 제거, 116
- NFS 클라이언트
  - NFS 서비스, 72
  - 호환되지 않는 운영 체제 지원, 111
- NFS 환경, 보안 NFS 시스템, 189
- /nfs4 마운트 지점, 195, 197
- nfs4cbd 데몬, 137
- nfscast: 선택 메시지, 125
- nfscast: 패킷을 보낼 수 없음 메시지, 124
- nfscast: 회신을 받을 수 없음 메시지, 124
- nfsd 데몬, 137-138
  - 마운트 및, 183-184
  - 서버의 응답 확인, 118
  - 실행 중인지 확인, 120
- nfslog.conf 파일, 설명, 133-134
- nfslogd 데몬, 설명, 138-139
- nfslogd 파일, 132-133
- nfsmapid\_domain 매개변수, 140
- NFSMAPID\_DOMAIN 키워드, 181
- nfsmapid 데몬
  - ACL 및, 181-182
  - DNS TXT 레코드 및, 141-142
  - NFSv4 기본 도메인 구성, 143-144
  - NFSv4 도메인 식별, 142-143
  - 관련 추가 정보, 144-145
  - 구성 파일 및, 140-141
  - 설명, 75, 139-145
  - 우선 순위 규칙 및, 141
- nfsref 명령, 165
- nfsstat 명령, 121, 165-167, 602, 604
  - c 옵션(클라이언트), 602, 603
  - m 옵션(파일 시스템당), 602, 604
  - s 옵션(서버), 602
  - 개요, 597, 602
- NFS에서 nolargefiles를 지원할 수 없음 메시지, 128
- NFS의 ACL 관련 문제, 방지, 181
- NFS의 ACL 관련 문제 방지, 181

NISaliases 맵, 341  
 NISmail.aliases 맵, 설정, 295  
 NIS 이름 서비스, autofs 맵 업데이트, 103  
 nnn 제어 문자, 542  
 no\_default\_msa FEATURE() 선언, 365  
 noauth 옵션(PPP), 411, 422  
 nocanonify FEATURE() 선언, 365  
 noccip 옵션(PPP), 414  
 noipdefault 옵션(PPP), 411  
 nolargefiles 옵션  
     in\_vfstab 파일, 89  
     mount 명령, 88, 149  
     오류 메시지, 128  
 noservice 옵션(PPP), 501  
 nosuid 옵션, share 명령, 160  
 nouucp FEATURE() 선언, 365  
 nsswitch.conf 파일, 576  
 nthreads 옵션, lockd 데몬, 136  
 ntp.conf 파일, 62, 63  
 ntp-keygen 명령, 65  
 NTP 서버, 설정, 62  
 NTP 클라이언트, 설정, 62-63  
 NTP 파일, 64  
 ntpd 데몬, 62, 63, 65  
 ntpdate 명령, 65  
 ntpdc 명령, 65  
 ntpq 명령, 65  
 ntpstats 디렉토리, 65  
 ntptime 명령, 65  
 ntptrace 명령, 65  
 nullclient FEATURE() 선언, 365

## O

-O 옵션, mount 명령, 152  
 -o 옵션  
     mount 명령, 151  
     share 명령, 158, 161  
 OPEN 공유 지원, NFS 버전 4, 178-179  
 openssl 명령 및 sendmail, 289  
 options.ttyname 파일(PPP), 참조  
     /etc/ppp/options.ttyname  
 options 파일, PPP, 408-409  
 Oracle Solaris, UUCP 버전, 527

OSNAME 맵 변수, 208  
 OSREL 맵 변수, 208  
 ostype 디렉토리, 333  
 OSVERS 맵 변수, 208  
 owner- 접두어, 메일 별칭, 325  
 owner- 접두어 및 우편함 이름, 324  
 owner-owner 및 우편함 이름, 324

## P

p 제어 문자, Dialers 파일, 542  
 PAP(암호 인증 프로토콜)  
     /etc/ppp/pap-secrets 파일, 486  
     login 옵션 사용, 489  
     PAP 자격 증명 데이터베이스 만들기, 427-428  
     계획, 426  
     구성  
         다이얼 인 서버, 429-430  
         신뢰할 수 있는 호출자, 430-431, 431, 432  
     암호 제안 사항, 487  
     예제 구성, 396  
     인증 프로세스, 487  
     작업 맵, 426-427  
     정의, 486  
 PAP 인증을 위해 구성, 427, 430-431, 431, 432  
 PAP 자격 증명 데이터베이스  
     다이얼 인 서버에 대해 만들기, 427-428  
     만들기  
         다이얼 인 서버, 428  
         신뢰할 수 있는 호출자, 430-431  
 passive 옵션(PPP), 422  
 passwd 파일, UUCP 로그인 사용, 518  
 pathconf: 서버가 응답하지 않음 메시지, 125  
 pathconf: 정보 없음 메시지, 125  
 PC-DOS 파일, autofs를 사용하여 액세스, 106  
 pcfs 옵션, autofs 맵, 106  
 Perl 5, 소개, 42-43  
 Permissions 파일  
     COMMANDS 옵션, 550, 551, 554  
     LOGNAME  
         MACHINE과 결합, 553  
     설명, 547  
     원격 컴퓨터의 로그인 ID, 547

## Permissions 파일 (계속)

## MACHINE

LOGNAME과 결합, 553

OTHER 옵션, 553

기본 사용 권한 또는 제한 사항, 547

설명, 547

MYNAME 옵션, 548

NOREAD 옵션, 550

NOWRITE 옵션, 550

OTHER 옵션, 553

READ 옵션, 549

REQUEST 옵션, 548

SENDFILES 옵션, 548

uucheck 명령 및, 513

VALIDATE 옵션, 552, 553

WRITE 옵션, 549

고려 사항, 547

구성 항목, 547

노드 이름 변경, 548

다이얼 백 권한, 550

보안 설정, 522

설명, 515, 546

원격 실행 권한, 550, 553

전달 작업, 554

콜백 옵션, 550

파일 전송 권한, 548, 550

형식, 547

Permissions 파일의 COMMANDS 옵션, 550-551, 554

VALIDATE 옵션, 553

Permissions 파일의 MYNAME 옵션, 548

Permissions 파일의 NOREAD 옵션, 550

Permissions 파일의 NOWRITE 옵션, 550

Permissions 파일의 OTHER 옵션, 553

Permissions 파일의 READ 옵션, 549

Permissions 파일의 REQUEST 옵션, 548

Permissions 파일의 SENDFILES 옵션, 548

Permissions 파일의 VALIDATE 옵션, 552, 553

COMMANDS 옵션, 550, 551

Permissions 파일의 WRITE 옵션, 549

Permissions 파일의 콜백 옵션, 550

Permit-type 필드의 Group 키워드, 557

Permit-type 필드의 Non-group 키워드, 557

Permit-type 필드의 Non-user 키워드, 557

Permit-type 필드의 User 키워드, 557

persist 옵션(PPP), 422

PidFile 옵션, sendmail 명령, 360

ping 명령, 239, 581, 597, 598

Poll 파일

설명, 515, 554

형식, 554

postmaster 별칭, 만들기, 298

postmaster 우편함

만들기, 299

설명, 324

테스트, 308

pound sign (#), 간접 맵의 주석, 199

PPP

asppp와 다른 점, 376

DSL 지원, 386

ISDN 지원, 381

PPP 계획 작업 맵, 389

pppd

참조 pppd 명령

PPPoE, 386

개요, 375

관련 RFC, 378

구성 파일 옵션

참조 옵션(PPP)

구성 파일 요약, 465

다이얼 업 링크, 379

링크의 각 부분, 378-384, 387-388

문제 해결

참조 PPP 문제 해결

비동기 PPP에서 변환, 508-509

인증, 384, 385

일반적인 문제, 448

자원, 외부, 377

전용 회선 링크, 382

채트 스크립트 예, 410

파일 권한, 467

호환성, 376

PPP 구성 작업

PPPoE 터널, 439

구성 문제 진단, 455

다이얼 업 링크, 405

인증, 425-426

전용 회선, 419

## PPP 디버깅

- PPPoE 문제 진단, 460
- 네트워크 문제 진단, 451
- 디버깅 켜기, 450
- 모뎀 문제 해결, 456
- 직렬 회선 문제 진단, 459
- 채트 스크립트 디버깅, 457
- 통신 문제 해결, 454, 455

## PPP 링크의 ISDN, 381

## PPP 문제 해결

- 일반적인 문제, 448
  - PPP 구성 사용, 456
  - 네트워크, 453
  - 인증, 463
  - 일반 통신, 454
  - 전용 회선 링크, 463
  - 직렬 회선, 459
  - 채트 스크립트, 457, 458, 459
- 작업 맵, 447
- 진단 정보 획득, 449-450, 450

## pppd 명령

- DSL 회선 테스트, 442
- 구문 분석 옵션, 467
- 디버깅 켜기, 450
- 정의, 466
- 진단 정보 획득, 449, 461
- 호출 시작, 416

## pppdebug 로그 파일, 460

## PPPoE

- DSLAM, 388
- snoop 추적 획득, 461
- 개요, 386
- 구성 작업 맵, 439
- 명령 및 파일 목록, 494
- 액세스 서버 구성, 443, 444, 445
- 액세스 서버에서 서비스 제공, 497-499, 499
- 일반적인 문제 해결, 460, 461
- 터널 계획, 400, 401, 403

## pppoe.so 공유 객체, 499, 502

## PPPoE 클라이언트

- /etc/ppp/peers/peer-name 파일
  - 사용법(PPPoE), 502
- 계획, 400, 440
- 구성, 440-441

## PPPoE 클라이언트 (계속)

- 구성 작업 맵, 439
- 명령, 502
- 액세스 서버 및, 502
- 액세스 서버 정의, 441
- 장비, 400
- 정의, 386
- 파일, 502

## pppoecl 유틸리티

- 정의, 502
- 진단 정보 획득, 461

## pppoed 데몬

- 시작, 443
- 정의, 497

## .ppprc 파일

- 권한, 468
- 만들기, 414
- 정의, 466

## PPP에 대한 -debug 옵션, 450

## PPP에 대한 demand 초기화 스크립트, 423

## PPP에 대한 진단 정보

- debug 옵션, 450
- PPPoE 터널에 대한 로그 파일, 460
- 다이얼 업 링크, 449
- 전용 회선 링크, 449
- 켜기

## pppd 사용, 449-450

## PPP의 chat 프로그램, 참조 채트 스크립트

## PPP의 구성 예

- CHAP 인증, 398
- PAP 인증, 396
- PPPoE 터널, 401
- 다이얼 업 링크, 391
- 전용 회선 링크, 394

## PPP의 링크 유형

- 다이얼 업, 379
- 다이얼 업과 전용 회선 비교, 382
- 링크의 각 부분, 379
- 물리적 링크 매체, 379
- 전용 회선, 382

## PPP의 암호 파일, 참조 /etc/ppp/pap-secrets 파일

## praliases 명령, 330

## preserve\_local\_plus\_detail FEATURE() 선언, 365

## preserve\_luser\_host FEATURE() 선언, 365

ProcessTitlePrefix 옵션, sendmail 명령, 360  
 proftpd.conf 파일, 설명, 569  
 proftpd.scoreboard 파일, 설명, 569  
 proftpd 데몬, 설명, 569  
 pstack 명령, 167-168  
 public 옵션  
   (dfstab 파일), 83  
   mount 명령, 90, 151  
   WebNFS 및, 99  
   공유 오류 메시지, 129  
 put 명령(FTP), 예, 588

## Q

-qf 옵션, sendmail 명령, 360  
 -qGname 옵션, sendmail 명령, 360  
 -qptime 옵션, sendmail 명령, 360  
 -q[!]Isubstring 옵션, sendmail 명령, 360  
 -q[!]Rsubstring 옵션, sendmail 명령, 360  
 -q[!]Ssubstring 옵션, sendmail 명령, 360  
 -q 옵션, uustat 명령, 523  
 queuegroup FEATURE() 선언, 365

## R

r 제어 문자, Dialers 파일, 542  
 -r 옵션  
   mount 명령, 151  
   umountall 명령, 155  
   uucp 명령, 524  
   Uutry 명령, 524  
 rbl FEATURE() 선언, 366  
 rcv 명령, 589, 593  
   경로 이름  
     구문 옵션, 590  
     절대 또는 축약, 590  
   디렉토리 복사, 591  
   로컬 시스템과 원격 시스템 간에 복사, 591, 593  
   보안 문제, 589  
   설명, 589  
   소스 및 대상 지정, 590  
   예, 593  
 rdate 명령, 64

relay\_mail\_from FEATURE() 선언, 366  
 relay-domains 파일, 330  
 remote\_mode FEATURE() 선언, 366  
 remote.unknown 파일, 558  
 reparsed 데몬, 145  
 -request 접미어 및 우편함 이름, 324  
 RFC(Request for Comments), PPP, 378  
 .rhosts 파일  
   검색, 580  
   보안 문제, 578  
   삭제, 580  
   설명, 578  
   원격 시스템 인증 프로세스, 576, 577-578  
 rlogin 명령  
   로그인 인터럽트, 576  
   로그인 후의 프로세스, 579  
   보안 NFS 및, 192  
   사용, 582  
   설명, 576  
   인증, 576, 578  
     /etc/hosts.equiv 파일, 577  
     .rhosts 파일, 578  
     네트워크 또는 원격 시스템 인증, 576, 577  
     직접 또는 간접 로그인, 578, 579  
 rm 명령, 578  
 rmail 명령, 330  
 ro 옵션  
   mount 명령, 151  
   mount 명령(-o 플래그 포함), 151  
   share 명령, 158, 161  
 root 옵션, share 명령, 160  
 RPC, 602, 603  
   보안  
     DH 인증 문제, 192, 193  
     개요, 190  
     인증, 191  
 rpcbind 데몬  
   mountd 데몬이 등록되지 않음, 127  
   사용 불능 상태이거나 정지됨, 127  
 rpcinfo 명령, 168-169  
 RPCSEC\_GSS, 79  
 RS-232 전화선, UUCP 구성, 511  
 rusers 명령, 581  
 rw=client 옵션, umountall 명령, 158



## rw 옵션

- mount 명령, 151
- share 명령, 158, 161

## S

- s 제어 문자, Dialers 파일, 542
- s 옵션, umountall 명령, 155
- SA(SLP), 243, 251, 256
- SA 서버(SLP), 241
- security
  - UUCP
    - Permissions 파일의 VALIDATE 옵션, 552
- sendmail.cf 파일, 330
  - 공급업체 설정, 318
  - 구성 파일 작성, 285
  - 대체 구성, 293-294
  - 로그 레벨, 339
  - 메일 게이트웨이 및, 328
  - 메일 도메인 및, 345
  - 메일 서버 및, 339
  - 메일 호스트 및, 339
  - 메일러, 설명, 320
  - 버전 레벨, 318
  - 설명, 338-339
- sendmail command, 버전 8.12의 명령줄 옵션, 359
- sendmail.ct 파일, 372
- sendmail.cw 파일, 372
- sendmail.hf 파일, 371
- sendmail.mc 파일, 332
- sendmail.pid 파일, 331, 334
- sendmail.st 파일, 참조 statistics 파일
- sendmail 명령
  - 8.12의 명령줄 옵션, 360
    - /etc/mail/helpfile 파일, 371
    - /etc/mail/local-host-names 파일, 372
    - /etc/mail/sendmail.ct 파일, 372
    - /etc/mail/sendmail.cw 파일, 372
    - /etc/mail/submit.cf, 358
    - /etc/mail/trusted-users 파일, 372
  - FEATURE() 선언
    - 버전 8.12의 변경 사항, 363
  - .forward 파일, 342
  - helpfile 파일, 371

## sendmail 명령 (계속)

- IPv6 주소 및 버전 8.12, 372
- local-host-names 파일, 372
- main.mc 파일, 371
- main-v7sun.mc 파일, 371
- NIS aliases 맵, 341
- NIS 및 DNS와의 상호 작용, 348
- NIS의 상호 작용, 347
- sendmail.ct 파일, 372
- sendmail.cw 파일, 372
- submit.cf 파일, 358
- subsidiary.mc 파일, 371
- subsidiary-v7sun.mc 파일, 371
- TCP 래퍼 및, 357
- trusted-users 파일, 372
- 기능, 338
- 대체 명령, 318
- 매크로
  - 버전 8.12의 m4 구성 매크로, 363
  - 버전 8.12의 MAX 매크로, 362
  - 버전 8.12의 정의된 매크로, 361
- 메일러, 내장
  - [TCP] 및 [IPC], 370
- 버전 8.12에서 변경된 사항, 357
- 버전 8.12의 FEATURE() 선언
  - supported, 364
  - 지원되지 않음d, 366
- 버전 8.12의 LDAP, 369
- 버전 8.12의 MAILER() 선언, 366
- 버전 8.12의 규칙 세트, 370
- 버전 8.12의 대기열 기능, 368
- 버전 8.12의 명령줄 옵션, 358
- 버전 8.12의 배달 에이전트 플러그, 367
- 버전 8.12의 배달 에이전트에 대한 등식, 367
- 버전 8.12의 파일 이름 또는 파일 위치 변경
  - 사항, 371
- 버전 8.13의 FEATURE() 선언, 356-357
- 버전 8.13의 구성 파일 옵션, 354-356
- 버전 8.13의 명령줄 옵션, 354
- 버전 8.13의 변경 사항, 349-357
- 설명, 335
- 오류 메시지, 311
- 이름 서비스 및, 346
- 컴파일 플러그, 316



- sendmail 명령의 옵션
  - 8.12의 명령줄 옵션, 360
  - PidFile 옵션, 360
  - ProcessTitlePrefix 옵션, 360
  - 버전 8.12의 명령줄 옵션, 358, 359
  - 버전 8.13의 구성 파일 옵션, 354-356
  - 버전 8.13의 명령줄 옵션, 354
- sendmail 버전 8.13의 FEATURE() 선언, 356-357
- server\_delegation 매개변수, 138
- server\_versmax 매개변수, 138
- server\_versmin 매개변수, 138
- setfacl 명령, NFS 및, 181
- setgid 모드, share 명령, 160
- setmnt 명령, 164
- setuid 모드
  - share 명령, 160
  - 보안 RPC 및, 192
- share 명령
  - WebNFS 서비스 사용으로 설정, 83
  - 보안 문제, 160
  - 설명, 158-162
  - 옵션, 158
- shareall 명령, 163
- showmount 명령, 164
- shutmsg 파일, 설명, 569
- slash (/), /- as 마스터 맵 마운트 지점, 195
- SLP
  - snoop slp 추적 분석, 225
  - 검색 요청, 239
  - 구성, 223-224
  - 구성 등록 정보, 228
  - 구성 파일, 227, 228-229
  - 구조, 217
  - 구현, 220
  - 데몬, 220
  - 로깅, 217
  - 배포 계획, 223-224
  - 브로드캐스트 경로 지정, 238
  - 성능 조정, 234
  - 알림, 246
  - 에이전트 및 프로세스, 218-219
  - 패킷 크기, 237
- slp.conf 파일, 주석, 228
- slp.jar 라이브러리, 220
- SLP 메시지 유형, 262-263
- SLP 상태 코드, 261-262
- SLP 성능 조정, 234
- slpd.conf 파일, 230, 243-244
- slpd 데몬, 255, 256, 259
  - DA, 241
  - DA 제거, 233
  - SA 서버, 241
  - 멀티홈 시스템 및, 249
  - 범위 및, 243
  - 인터페이스 변경, 250
  - 정적 DA 및, 230
  - 프록시 알림 및, 252
  - 하트비트, 233
- SLPv2, SLPv1을 통한 상호 운용성, 246
- SMART\_HOST() m4 구성 매크로, 363
- SMTP(Simple Mail Transfer Protocol)
  - sendmail.cf 파일, 359
  - 메일러, 320
- SMTP 및 TLS
  - 관련 보안 고려 사항, 354
  - 구성 파일 옵션, 350-352
  - 규칙 세트, 353-354
  - 매크로, 352-353
  - 설명, 349-354
  - 작업 정보, 289-293
- snoop 명령, 169-170, 597, 599
  - SLP 서비스 등록 및, 235
  - SLP 트래픽 및, 248
  - SLP에서 사용, 224, 225
  - 다중 SLP 요청 및, 250
  - 재전송 모니터링, 241
- snoop 추적, PPPoE, 461
- soft 옵션, mount 명령, 151
- solaris-antispam.m4 파일, 332
- solaris-generic.m4 파일, 305, 332
- Solaris PPP 4.0, 참조 PPP
- solaris2.m4 파일, 333
- solaris2.ml.m4 파일, 333
- solaris2.pre5.m4 파일, 333
- solaris8.m4 파일, 333
- Speed 필드
  - Devices 파일 Class 필드 및, 536
  - Systems 파일, 530

sPPP 장치 번호, PPP 주소 지정, 494  
 spray 명령, 597, 598  
 statd 데몬, 145-146  
 statistics 파일, 331  
 .Status 디렉토리, 525  
 STATUS 오류 메시지(UUCP), 525, 562, 563  
 STREAMS, 장치 구성, 557  
 STTY 흐름 제어, 533, 543  
 submit.cf 파일, 331, 332, 358  
 submit.mc 파일, 332  
 subsidiary.cf 파일, 275, 331, 332  
 subsidiary.mc 파일, 332, 371  
 subsidiary-v7sun.mc 파일, 333, 371  
 sun\_reverse\_alias\_files FEATURE() 선언, 366  
 sun\_reverse\_alias\_nis FEATURE() 선언, 366  
 sun\_reverse\_alias\_nisplus FEATURE() 선언, 366  
 sync 옵션(PPP), 422  
 Sysfiles 파일  
     Systems 목록 인쇄, 546  
     샘플, 546  
     설명, 515, 545  
     형식, 545  
 syslog.conf 파일, 310  
 syslogd 명령, 334  
 Sysname 파일, 515, 546  
 /system/volatile/nca\_httpd\_1.door 파일, 58  
 /system/volatile/sendmail.pid 파일, 334  
 Systems 파일  
     Chat Script 필드, 531, 533  
     Devices 파일 Class 필드 및, 536  
     Devices 파일 Type 필드 및, 535  
     Phone 필드, 530  
     Speed 필드, 530  
     System-Name 필드, 528  
     TCP/IP 구성, 521  
     Time 필드  
         Never 항목, 548  
         설명, 528  
     Type 필드, 529  
     다이얼 번호 약어, 530  
     다이얼-코드 약어, 514  
     문제 해결, 525  
     설명, 515, 527  
     여러 또는 다른 파일, 527, 545

## Systems 파일 (계속)

여러 파일 또는 서로 다른 파일, 515  
 제어 문자, 532  
 패리티 설정, 533  
 하드웨어 흐름 제어, 533  
 형식, 527  
 Systems 파일의 Phone 필드, 530  
 Systems 파일의 System-Name 필드, 528  
 Systems 파일의 Time 필드, 528, 548  
 Systems 파일의 전화 번호, 530

## T

### T 제어 문자

Devices 파일, 539  
 Dialers 파일, 539, 542  
 -t 옵션, lockd 데몬, 136  
 TA(터미널 어댑터)를 위한 채트 스크립트, 482-483, 483  
 TCP, NFS 버전 3 및, 76  
 TCP/IP 네트워크  
     UUCP, 521  
 TCP/IP 트래픽, 597, 599, 600  
 TCP 래퍼, sendmail 명령 및, 357  
 TCP 프로토콜, 600  
 telnet 명령, 보안 NFS 및, 192  
 Time 필드의 day 항목, 529  
 Time 필드의 retry 하위 필드, 529  
 TLS 및 SMTP  
     관련 보안 고려 사항, 354  
     구성 파일 옵션, 350-352  
     규칙 세트, 353-354  
     매크로, 352-353  
     설명, 349-354  
     작업 정보, 289-293  
 TLS(전송 계층 보안) 및 SMTP  
     관련 보안 고려 사항, 354  
     구성 파일 옵션, 350-352  
     규칙 세트, 353-354  
     매크로, 352-353  
     설명, 349-354  
     작업 정보, 289-293  
 TLS를 사용하는 SMTP 실행  
     규칙 세트, 353-354

## TLS를 사용하는 SMTP 실행 (계속)

설명, 349-354

TLS를 사용하도록 SMTP 설정, 289-293

## TLS를 사용하여 SMTP 실행

관련 보안 고려 사항, 354

구성 파일 옵션, 350-352

매크로, 352-353

작업 정보, 289-293

TM UUCP 임시 데이터 파일, 559

truss 명령, 170

trusted-users 파일, 331, 372

## Type 필드

Devices 파일, 534

Systems 파일, 529

Type 필드의 ACU 키워드, 535

Type 필드의 Direct 키워드, 535

Type 필드의 Sys-Name 변수, 535

## U

-U 옵션, sendmail 명령, 360

UA, 요청, 235

UA(SLP), 224, 246

요청 시간 초과, 248

UDP, NFS 및, 77

UDP/TCP 유니캐스트(SLP), 249

UDP 프로토콜, 600

## umount 명령

autofs 및, 74

설명, 153-154

umountall 명령, 155

uname -n 명령, 546

UNIX 인증, 189, 191

unshare 명령, 162-163

unshareall 명령, 163

URL 서비스 유형, WebNFS 및, 99

Usenet, 511, 527

User-job-grade 필드의 기본 키워드, 556

## /usr/bin/cu 명령

Systems 목록 인쇄, 546

모뎀 또는 ACU 확인, 523

설명, 514

여러 또는 다른 구성 파일, 545

여러 파일 또는 서로 다른 구성 파일, 515

/usr/bin/ftp 명령, 설명, 568

/usr/bin/ftpcount 명령, 설명, 568

/usr/bin/ftpdctl 명령, 설명, 568

/usr/bin/ftptop 명령, 설명, 568

/usr/bin/ftpwho 명령, 설명, 568

/usr/bin/mail 명령, 329

/usr/bin/mailcompat 필터, 329

/usr/bin/mailq 명령, 329

/usr/bin/mailstats 명령, 330

/usr/bin/mailx 명령, 330

/usr/bin/mconnect 명령, 309-310, 330

/usr/bin/ncab2clf 명령, 58

/usr/bin/praliases 명령, 330

/usr/bin/rmail 명령, 330

## /usr/bin/uucp 명령

uucico 실행, 512

로그인 ID의 홈 디렉토리, 513

설명, 514

전달 작업에 대한 권한, 554

전송 디버그, 524

/usr/bin/uulog 명령, 513, 525

/usr/bin/uupick 명령, 514, 523

/usr/bin/uustat 명령, 514, 523

## /usr/bin/uuto 명령

uucico 실행, 512

공개 디렉토리 파일 제거, 523

설명, 514

## /usr/bin/uux 명령

uucico 실행, 512

설명, 514

/usr/bin/vacation 명령, 330, 338

/usr/bin 디렉토리, 내용, 329

/usr/kvm 디렉토리, 디스크가 없는 클라이언트를

통한 마운트, 74

/usr/lib/inet/ntpd 데몬, 설명, 65

/usr/lib/inet/proftpd 데몬, 설명, 569

/usr/lib/ncsa\_addr.so 라이브러리, 58

/usr/lib/net/ncacnfd 명령, 58

/usr/lib/uucp/uuccheck 명령, 513, 525

/usr/lib/uucp/uucleanup 명령, 513

/usr/lib/uucp/Uutry 명령, 513, 524, 525

/usr/lib 디렉토리, 내용, 333

/usr/ntp/ntpstats 디렉토리, 65

/usr/sbin/editmap 명령, 334

/usr/sbin/etern 스크립트, 334  
 /usr/sbin/ftprestart 명령, 설명, 568  
 /usr/sbin/ftpscrub 명령, 설명, 569  
 /usr/sbin/ftpsht 명령, 설명, 569  
 /usr/sbin/in.comsat 데몬, 334  
 /usr/sbin/inetd 데몬, in.uucpd 호출, 513  
 /usr/sbin/makemap 명령, 334  
 /usr/sbin/mount 명령, 참조 mount 명령  
 /usr/sbin/newaliases 링크, 334  
 /usr/sbin/ntp-keygen 명령, 65  
 /usr/sbin/ntpddate 명령, 65  
 /usr/sbin/ntpdcc 명령, 65  
 /usr/sbin/ntpq 명령, 65  
 /usr/sbin/ntpstime 명령, 65  
 /usr/sbin/ntpstrace 명령, 65  
 /usr/sbin/showmount 명령, 164  
 /usr/sbin/sppptun 명령, 정의, 495  
 /usr/sbin/syslogd 명령, 334  
 /usr/sbin/unshareall 명령, 163  
 /usr 디렉토리, 디스크가 없는 클라이언트를 통한  
   마운트, 74  
 uucheck 명령, 513, 525  
 uucico 데몬  
   Dialcodes 파일 및, 545  
   Systems 목록 인쇄, 546  
   Systems 파일 및, 527  
   UUCP 로그인 추가, 518  
   uusched 데몬 및, 513  
   Uutry 명령 및, 513  
   설명, 512  
   여러 또는 다른 구성 파일, 527, 545  
   여러 파일 또는 서로 다른 구성 파일, 515  
   최대 동시 실행, 515  
   최대 동시 실행 수, 558  
 uucleanup 명령, 513  
 UUCP  
   Oracle Solaris 버전, 511, 527  
   STREAMS 구성, 557  
   공개 디렉토리 유지 관리, 523  
   관리 명령, 513  
   관리 파일, 558, 560  
   구성  
     TCP/IP를 통해 UUCP 실행, 521  
     UUCP 로그인 추가, 518

## UUCP (계속)

권한 있는 로그인 및 암호, 552  
 노드 이름  
   별칭, 515, 548  
   원격 컴퓨터, 528, 546  
 데몬  
   개요, 512, 513  
 데이터베이스 파일, 514, 558  
   asppp 구성, 515  
   기본 구성 파일, 515  
   설명, 514, 515  
   여러 또는 다른 파일, 527, 545  
   여러 파일 또는 서로 다른 파일, 515  
 디렉토리  
   공개 디렉토리 유지 관리, 523  
   관리, 513  
   오류 메시지, 525  
 로그 파일  
   정리, 520  
   표시, 513  
 로그 파일 표시, 513  
 로그인  
   권한, 552  
   추가, 518  
 “로그인 셀”, 512  
 메일 누적, 522  
 문제 해결, 523, 563  
   ACU 고장, 523  
   ASSERT 오류 메시지, 525, 560, 561  
   STATUS 오류 메시지, 525, 562, 563  
   Systems 파일 확인, 525  
   기본 정보 확인, 525  
   모뎀 고장, 523  
   문제 해결 명령, 525  
   오류 메시지 확인, 525, 563  
   전송 디버그, 524, 525  
 보안  
   Permissions 파일의 COMMANDS 옵션, 550,  
     551  
   Permissions 파일의 VALIDATE 옵션, 552, 553  
   공개 디렉토리 파일의 고정 비트, 523  
   설정, 522  
 사용자 명령, 513, 514  
 설명, 511, 527

**UUCP(계속)**

- 셸 스크립트, 519, 521
- 수동 모드, 548
- 수동으로 매개변수 대체, 554
- 스폴
  - 데몬 예약, 513
  - 작업 등급 정의, 555, 557
  - 정리 명령, 513
- 원격 실행
  - 데몬, 512
  - 명령, 547, 550, 553
  - 작업 파일 C., 559, 560
- 원격 컴퓨터 폴링, 515, 554
- 유지 관리, 522, 523
- 전달 작업, 554
- 전송 속도, 530, 536
- 콜백 옵션, 550
- 파일 전송
  - 데몬, 512
  - 문제 해결, 524, 525
  - 사용 권한, 548, 550
  - 작업 파일 C., 559, 560
- 하드웨어 구성, 511
- UUCP(UNIX-to-UNIX Copy command), 메일러, 321
- UUCP(UNIX-to-UNIX Copy 명령), 연결
  - 테스트, 308
- uucp 명령
  - uucico 실행, 512
  - 로그인 ID의 홈 디렉토리, 513
  - 설명, 514
  - 전달 작업에 대한 권한, 554
  - 전송 디버그, 524
- UUCP 유지 관리
  - 공개 디렉토리 유지 관리, 523
  - 로그인 추가, 518
  - 메일, 522
  - 셸 스크립트, 521
  - 정기 유지 관리, 522, 523
- UUCP 통신 링크의 장치 유형, 529
- UUCP 통신 링크의 전송 속도, 530, 536
- uucppublic 디렉토리 유지 관리, 523
- UUCP에 대해 데몬 예약, 513
- UUCP 유지 관리, 셸 스크립트, 519
- uudemon.admin 셸 스크립트, 520

- uudemon.cleanup 셸 스크립트, 520
- uudemon.crontab 파일, 519
- uudemon.hour 셸 스크립트
  - uusched 데몬 실행, 513
  - uuxqt 데몬 실행, 512
  - 설명, 520
- uudemon.poll 셸 스크립트, 520, 554
- uulog 명령, 513, 525
- uname 명령, 525
- uupick 명령
  - 공개 디렉토리 파일 제거, 523
  - 설명, 514
- uusched 데몬
  - uudemon.hour 셸 스크립트 호출, 520
  - 설명, 513
  - 최대 동시 실행, 515
  - 최대 동시 실행 수, 558
- uustat 명령
  - uudemon.admin 셸 스크립트, 520
  - 모뎀 또는 ACU 확인, 523
  - 설명, 514
- uuto 명령
  - uucico 실행, 512
  - 공개 디렉토리 파일 제거, 523
  - 설명, 514
- Uutry 명령, 513, 524, 525
- uux 명령
  - uucico 실행, 512
  - 설명, 514
- uuxqt 데몬
  - uudemon.hour 셸 스크립트 호출, 520
  - 설명, 512
  - 최대 동시 실행, 515
  - 최대 동시 실행 수, 558

**V**

- V 옵션, umount 명령, 153
- v 옵션
  - automount 명령, 122
  - uucheck 명령, 525
- vacation 명령, 329, 330, 338
- /var/log/xferlog 파일, 설명, 569
- /var/mail 디렉토리, 275, 276

/var/mail 디렉토리 (계속)  
   메일 클라이언트 구성 및, 280  
   자동 마운트, 280  
 /var/mail 파일, 323  
 /var/nca/log 파일, 58  
 /var/ntp/ntp.drift 파일, 65  
 /var/run/proftpd.scoreboard 파일, 설명, 569  
 /var/spool/clientmqueue 디렉토리, 334  
 /var/spool/mqueue 디렉토리, 334  
 /var/spool/uucppublic 디렉토리 유지 관리, 523  
 /var/uucp/.Admin/errors 디렉토리, 525  
 /var/uucp/.Status 디렉토리, 525  
 vfstab 파일  
   automount 명령 및, 201  
   NFS 서버 및, 86  
   nolargefiles 옵션, 89  
   디스크가 없는 클라이언트를 통한 마운트, 74  
   부트 시에 파일 시스템 마운트, 86  
   클라이언트측 페일오버 사용으로 설정, 89  
 VIRTUSER\_DOMAIN\_FILE() m4 구성 매크로, 363  
 VIRTUSER\_DOMAIN() m4 구성 매크로, 363  
 virtuser\_entire\_domain FEATURE() 선언, 366

## W

WAN(Wide Area Network)  
   Usenet, 511, 527  
 WebNFS 서비스  
   URL 서비스 유형 및, 99  
   개요, 78  
   계획, 98-99  
   방화벽 및, 100  
   보안 협상 및, 79  
   사용으로 설정, 83-84  
   설명, 187-188  
   작업 맵, 98  
   찾아보기, 99-100

## X

X. UUCP 실행 파일  
   uuxqt 실행, 512  
   설명, 560

X. UUCP 실행 파일 (계속)  
   정리, 520  
 xonxoff 옵션(PPP), 415

## 가

가상 호스트, 설정, 287

## 간

간접 맵(autofs)  
   automount 명령을 실행해야 하는 경우, 103  
   syntax, 199  
   개요, 199, 200  
   구문, 199  
   설명, 103  
   예제, 200  
   주석, 199  
 간접 원격 로그인, 578, 579

## 개

개행 제어 문자, 542

## 검

검색  
   .rhosts 파일, 580  
   원격 시스템에 로그인한 사용자, 581  
 검색 요청(SLP), 239  
 검증기, RPC 인증 시스템, 190

## 경

경고: 마운트 지점 이미 마운트됨 메시지, 123  
 경로 이름  
   rcp 명령  
     구문 옵션, 590  
     절대 또는 축약, 590  
   틸드(~), 590

**계**

계층 마운트 지점 메시지, 123  
계층적 마운트(다중 마운트), 204

**공**

공개 디렉토리 유지 관리(UUCP), 523  
공개 디렉토리 파일의 고정 비트, 523  
공개 키 데이터베이스  
    보안 키  
        데이터베이스, 191  
공개 키 맵, DH 인증, 191  
공개 키 암호화  
    DH 인증, 191  
    공개 키 데이터베이스, 190, 191  
    공통 키, 191  
    대화 키, 191  
    보안 키  
        원격 서버에서 삭제, 192  
    시간 동기화, 191  
공급업체 설정, sendmail.cf 파일에 지정, 318  
공백 제어 문자, 542  
공용 파일 핸들  
    autofs 및, 112  
    NFS 마운트, 79  
    WebNFS 및, 98  
    마운트 및, 183  
공용 파일 핸들을 사용할 수 없음 메시지, 126  
공유 해제 및 다시 공유, NFS 버전 4, 173

**관**

관리 명령(UUCP), 513  
관리 파일(UUCP)  
    cleanup, 520  
    실행 파일(X.), 512, 560  
    임시 데이터 파일(TM), 559  
    작업 파일(C.), 559, 560  
    잠금 파일(LCK), 559

**구**

구성  
    asppp UUCP 데이터베이스에 대한 링크, 515  
    UUCP  
        TCP/IP 네트워크, 521  
        데이터베이스 파일, 515  
        로그인 추가, 518  
        셸 스크립트, 519, 521  
        메일 게이트웨이, 327  
구성 파일  
    sendmail 명령, 339  
    UUCP, 554

**권**

권한, NFS 버전 3의 개선 사항, 74  
권한 거부됨 메시지, 127  
권한 파일, uuxqt 데몬 및, 512

**규**

규칙 세트  
    버전 8.12 sendmail, 370  
    테스트, 308

**끄**

끄기, 에코 검사, 542

**날**

날짜, 다른 시스템과 동기화, 64

**널**

널 제어 문자, 542

## 네

## 네임스페이스

autofs 및, 80

공유 액세스, 110

## 네트워크

## 문제 해결

높은 재전송률, 602

성능 모니터링을 위한 명령, 597

성능 정보 표시, 597, 598, 599, 604

IP 경로 지정 테이블, 601

서버 통계, 602, 604

인터페이스 통계, 599, 601

충돌률, 600

클라이언트 통계, 602, 604

호스트 응답, 598

클라이언트의 서버 호출 추적, 597, 599

## 패킷

네트워크에서 캡처, 599

삭제, 599

안정성 테스트, 597, 598

오류율, 600

전송된 수, 600

호스트로 보내기, 598

네트워크 데이터베이스 서비스, UUCP 포트, 521

네트워크 인터페이스(SLP), 경로를 지정하지 않을  
경우의 고려 사항, 253

네트워크 잠금 관리자, 77

네트워크 정보 표시, 597, 598, 599, 604

네트워크 캐시 및 가속기, 참조 NCA

## 노

## 노드 이름

UUCP 별칭, 515, 548

UUCP 원격 컴퓨터, 528, 546

## 다

다른 시스템과의 메일 연결, 테스트, 309-310

다시 마운트 메시지, 123

## 다이얼 백

Permissions 파일의 콜백 옵션, 550

채트 스크립트를 통해 사용으로 설정, 532

다이얼 번호 약어, 530

다이얼 번호 약어의 등호(=), 530

## 다이얼 아웃 시스템

/etc/ppp/options.ttyname을 사용하여 직렬 회선  
구성, 472

계획 정보, 390

## 구성

CHAP 인증, 436, 438

PAP 인증, 430-431

모뎀, 407-408

직렬 포트, 407-408

직렬 회선 통신, 408-409

피어를 사용한 연결, 410-412

구성 작업 맵, 406

원격 피어 호출, 416-417

정의, 379

주소 지정

동적, 492

정적, 493

채트 스크립트 만들기, 409

## 다이얼 업 링크

계획, 390, 391

구성 파일 템플릿, 406

다이얼 업 프로세스, 381

링크에 대한 인증, 385

링크의 각 부분, 380-381

예, 391

## 일반적인 문제 진단

pppd 사용, 449

네트워크, 451

직렬 회선, 459

작업 맵, 405

정의, 379

## 채트 스크립트

ISDN TA, 482-483

UNIX 스타일의 로그인, 480-482

예, 477-478, 479-480, 483

템플릿, 478-479

채트 스크립트 만들기, 476

피어에 대한 호출 시작, 416-417

## 다이얼 인 서버

UUCP, 532

계획 정보, 390, 414



**다이얼 인 서버 (계속)**

## 구성

CHAP 인증, 434, 436

PAP 인증, 427-428, 428, 429-430

모뎀, 413

직렬 포트, 413

직렬 회선 통신, 415-416, 471

구성 작업 맵, 412

정의, 379

호출 받기, 416-417

다이얼-코드 약어, 514

**단**

단일 사용자 모드 및 보안, 192

**대**

## 대기열(UUCP)

uusched 데몬

설명, 513

최대 동시 실행, 515

최대 동시 실행 수, 558

관리 파일, 558, 560

데몬 예약, 513

스폴 디렉토리, 559

작업 등급 정의, 555, 557

정리 명령, 513

## 대시(-)

autofs 맵 이름, 209

Line2 필드 개체 틀, 536

Speed 필드 개체 틀, 530

다이얼 번호 약어, 530

대체 명령, sendmail 명령, 318

대화 키, 191

**더**

더하기 기호(+)

autofs 맵 이름, 209, 210

**데**

## 데몬

automountd, 135

autofs 및, 73

개요, 201

lockd, 136

mountd, 137

rpcbind로 등록되지 않음, 127

서버의 응답 확인, 119

실행 중인지 확인, 120, 127

nfs4cbd, 137

nfsd

서버의 응답 확인, 118

설명, 137-138

실행 중인지 확인, 120

nfslogd, 138-139

nfsmapid, 139-145

reparsed, 145

rpcbind

마운트 오류 메시지, 127

statd, 145-146

원격 마운트에 필요, 116

데몬 이미 실행 중 메시지, 126

데스크탑 게시 파일, 우편함 공간 요구 사항 및, 327

데이터(D.) UUCP 파일, 정리, 520

**도**

## 도메인

원격 로그인 및, 576

정의, 96

하위 도메인 및, 322

도메인 이름, 보안 NFS 시스템 및, 96

**동**

## 동기 PPP

참조 전용 회선 링크

동기 장치 구성, 420

동적 주소 지정, PPP, 492

## 등

등록 수명(SLP), 225

## 디

디렉토리(UUCP)

공개 디렉토리 유지 관리, 523

관리, 513

오류 메시지, 525

디렉토리 아님 메시지, 124

디렉토리 에이전트(SLP)

DA 주소, 230

SLP 구조 및, 217

네트워크 혼잡 및, 234

로드 균형 조정, 248-249

배치 위치, 248-249

배포 시기, 247

디렉토리는 '/'로 시작해야 함 메시지, 123

디버그

UUCP 전송, 524, 525

디스크가 없는 클라이언트

부트 프로세스 중의 보안, 192

수동 마운트 요구 사항, 74

## 레

레거시 서비스(SLP)

알림, 255, 259

정의, 255

## 로

로그 레벨, sendmail.cf 파일, 339

로그아웃(원격 시스템), 582, 583

로그인

원격 로그인

ftp 명령, 584

ftp 연결 닫기, 585

ftp 연결 열기, 584, 585

rlogin 사용, 582

rlogin 사용, 576

로그인 링크 만들기, 578

로그인, 원격 로그인(계속)

로그인한 사용자 알아보기, 581

인증(rlogin), 576, 578

인터럽트, 576

직접 또는 간접(rlogin), 578, 579

로그인(UUCP)

권한 있음, 552

추가, 518

로그

UUCP 로그 파일 정리, 520

UUCP 로그 파일 표시, 513

로컬 메일 주소, 324

로컬 배달 에이전트, 메일 서비스, 320

로컬 캐시 및 NFS 버전 3, 74

로컬 파일, autofs 맵 업데이트, 103

로컬 파일 시스템, 그룹 마운트 해제, 155

로컬 편지 별칭 파일, 설정, 296

## 루

루트 디렉토리, 디스크가 없는 클라이언트를 통한

마운트, 74

루프, 별칭, 308

## 마

마스터 맵(auto\_master)

/- 마운트 지점, 195, 199

automount 명령을 실행해야 하는 경우, 103

/etc/mnttab 파일과 비교, 201

개요, 195, 196

구문, 195

보안 제한, 111

사전 설치됨, 107

설명, 103

주석, 196

컨텐츠, 195, 197

마운트

autofs 및, 73, 204

nfsd 데몬 및, 183-184

/var/mail 디렉토리, 280

공용 파일 핸들 및, 183

디스크가 없는 클라이언트 요구 사항, 74

**마운트 (계속)**

- 미러 마운트 및, 193-194
- 백그라운드 채시도, 149
- 소프트와 하드, 116
- 예제, 151
- 원격 마운트
  - 데몬 필요, 116
  - 문제 해결, 117-118, 120
- 이미 마운트된 파일 시스템 오버레이, 152
- 읽기/쓰기 사양, 151
- 읽기 전용 사양, 151
- 전경 채시도, 149
- 직접 I/O 강제 적용, 149
- 키보드를 통한 중단, 116
- 통합 파일 시스템, 91
- 포트매퍼 및, 183-184
- 표의 모든 파일 시스템, 154

**마운트 지점**

- /- as 마스터 맵 마운트 지점, 195, 199
- /home, 195, 196
- /net, 197
- /nfs4, 195, 197
- 충돌 방지, 105

**마운트 지점을 만들 수 없음, 122****마운트 지점을 만들 수 없음 메시지, 122****마운트 해제**

- autofs 및, 73, 204
- 미러 마운트 및, 194
- 예제, 154
- 파일 시스템 그룹, 155

**마이너스 기호(-), /etc/hosts.equiv 파일 구문, 577****만****만들기**

- /etc/shells 파일, 306
- NFS 참조, 115, 195
- postmaster 별칭, 298
- postmaster 우편함, 299
- 키 맵 파일, 298

**매**

매핑되지 않은 사용자 또는 그룹 ID, 확인, 181-182

매핑되지 않은 사용자 또는 그룹 ID 확인, 181-182

**맵****맵(autofs)****automount 명령**

실행해야 하는 경우, 103

간접, 199, 200

관리 작업, 102

긴 행 분할, 196, 198, 199

네트워크 탐색, 202

다른 맵 참조, 209, 210

다중 마운트, 204

마스터, 195, 196

마운트 충돌 방지, 105

변수, 208, 209

실행 가능, 210

유지 관리 방법, 103

유형 및 용도, 102

주석, 196, 198, 199

직접, 197, 198

클라이언트에 대해 읽기 전용 파일 선택, 205, 207

탐색 프로세스 시작, 196, 203

특수 문자, 213

맵 키가 잘못됨 메시지, 124

맵 항목 메시지의 선행 공백, 123

맵 항목의 변수, 208, 209

맵을 사용한 탐색

개요, 202

프로세스 시작, 196, 203

맵의 \ (백슬래시), 196, 198, 199

맵의 백슬래시(\), 196, 198, 199

맵의 서버 가중치, 208

맵의 특수 문자, 213

**멀****멀티캐스트(SLP)**

DA, 231, 233

멀티홈 시스템 및, 249

## 멀티캐스트(SLP) (계속)

- 사용 안함으로 설정된 경우, 249
- 서비스 요청, 246
- 인터페이스 변경, 250
- 전파, 236
- 트래픽, 245
- 활성 시간 등록 정보, 235

## 멀티홈 호스트(SLP)

- 구성, 249
- 멀티캐스트 없음, 246
- 범위 및, 252
- 브로드캐스트 전용 경로 지정, 238
- 유티캐스트 경로 지정 사용 안함으로 설정, 251
- 인터페이스 변경, 250
- 프록시 알림, 252

## 메

## 메시지

## UUCP

- ASSERT 오류 메시지, 560, 561
- STATUS 오류 메시지, 562, 563
- 오류 메시지 확인, 525

## 메시지 유형, SLP, 262-263

## 메일 게이트웨이

- sendmail.cf 파일 및, 328
- 구성, 327
- 메일 게이트웨이 설정, 283
- 정의, 327
- 테스트, 308

## 메일 경로 지정, 메일 주소 및, 344

## 메일 구성

- 로컬 메일 및 원격 연결, 276
- 로컬만, 275
- 일반, 270
- 테스트, 307

## 메일 대기열

- 대기열 디렉토리 관리, 300
- 메일 대기열 이동, 303
- 메일 대기열 처리 강제 실행, 302
- 이전 메일 대기열 실행, 303
- 일부 실행, 302

## 메일 도메인

- sendmail.cf 파일 및, 345

## 메일 도메인 (계속)

- 이름 서비스 도메인 및, 346

## 메일 명령, 상호 작용, 335

## 메일 별칭 파일

- /etc/mail/aliases 파일, 340
- .mailrc 별칭, 340
- 설명, 339

## 메일 사용자 에이전트, 319

## 메일 서버, 327

- 공간 요구 사항, 327
- 메일 서버 설정, 303
- 백업 및, 327
- 설명, 327
- 우편함, 324, 327

## 메일 서비스

- sendmail 버전 8.13의 변경 사항, 349-357

## 메일 시스템 계획, 274

## 버전 8.12에서 sendmail 변경 사항, 357

## 소프트웨어 구성 요소, 319

- 로컬 배달 에이전트, 320
- 메일 사용자 에이전트, 319
- 메일 전송 에이전트, 319
- 메일 주소, 321
- 메일 칭, 325
- 메일러, 320
- 우편함 파일, 324

## 작업 맵

- 관리 .forward 파일, 304
- 대기열 디렉토리 관리, 300
- 메일 서비스 설정, 277
- 문제 해결 절차 및 팁, 306
- 편지 별칭 파일 관리, 294
- 포괄적인 작업 맵, 273

## 하드웨어 구성 요소

- 메일 게이트웨이, 327
- 메일 서버, 327
- 메일 클라이언트, 327
- 메일 호스트, 326
- 필수 요소, 326

## 메일 전송 에이전트, 319

## 메일 주소

- %, 324
- 대소문자 구분, 322
- 도메인 및 하위 도메인, 322

**메일 주소 (계속)**

- 로컬, 324
- 메일 경로 지정 및, 344
- 설명, 321

**메일 클라이언트**

- NFS 마운트된 파일 시스템 및, 280
- 메일 클라이언트 설정, 279
- 정의, 327

**메일 필터 APIMILTER, 317****메일 호스트**

- 메일 호스트 설정, 281
- 설명, 326

**메일러**

- SMTP(Simple Mail Transfer Protocol) 메일러, 320
- Solaris 메일러, 320
- UUCP(UNIX-to-UNIX Copy command)
  - 메일러, 321
- 내장(sendmail)
  - [TCP] 및 [IPC], 370
- 정의, 320

**명****명령**

- UUCP 문제 해결, 525
- UUCP를 사용한 원격 실행, 547, 550, 553
- 실행(X.) UUCP 파일, 512, 560
- 정지된 프로그램, 128

**모****모뎀, 모뎀 문제 해결, 456****모뎀(PPP)**

- DSL, 388
- 구성
  - 다이얼 아웃 시스템, 407-408
  - 다이얼 인 서버, 413
- 모뎀 속도 설정, 413
- 채트 스크립트
  - ISDN TA, 482-483
  - UNIX 스타일의 로그인, 480-482
  - 예, 410, 477-478, 479-480, 483
  - 템플릿, 478-479

**모뎀(PPP) (계속)**

- 채트 스크립트 만들기, 476

**모뎀(UUCP)**

- UUCP 데이터베이스
  - Devices 파일의 DTP 필드, 539
- UUCP 데이터베이스, Devices 파일의 DTP 필드, 538
- UUCP 하드웨어 구성, 511
- 문제 해결, 523
- 설정 특성, 533
- 직접 연결, 538
- 특성 설정, 543
- 포트 선택기 연결, 538, 539

**목****목록**

- 공유 파일 시스템, 161
- 마운트된 파일 시스템, 153
- 원격으로 마운트된 파일 시스템을 포함하는 클라이언트, 164

**문****문제 해결**

- autofs, 122
  - automount -v를 통해 생성되는 오류 메시지, 122
  - 기타 오류 메시지, 123
  - 마운트 지점 충돌 방지, 105
- MAILER-DAEMON 메시지 및, 311
- NFS
  - NFS 서비스가 실패한 위치 확인, 120
  - 서버 문제, 117
  - 원격 마운트 문제, 117, 127
  - 전략, 116
  - 정지된 프로그램, 128
- UUCP, 523, 563
  - ASSERT 오류 메시지, 525, 560, 561
  - STATUS 오류 메시지, 525, 562, 563
  - Systems 파일 확인, 525
  - 고장난 모뎀 또는 ACU, 523
  - 기본 정보 확인, 525

## 문제 해결, UUCP (계속)

- 문제 해결 명령, 525
- 오류 메시지 확인, 525, 563
- 전송 디버그, 524, 525
- 규칙 세트, 308
- 네트워크, 602, 604
- 다른 시스템과의 메일 연결, 309-310
- 메일 서비스, 306
- 배달되지 않은 메일, 308
- 편지 별칭, 308

## 미

## 미러 마운트

- 개요, 193-194
- 단일 서버에서 모든 파일 시스템 마운트, 87-88
- 하나 또는 여러 개의 파일 시스템 마운트, 87

## 반

## 반환 제어 문자, 542

## 방

## 방화벽

- NFS 액세스, 79
- WebNFS 액세스, 100
- 통해 파일 시스템 마운트, 90

## 배

## 배달되지 않은 메시지, 문제 해결, 308

## 백

- 백그라운드 파일 마운트 옵션, 149
- 백스페이스 제어 문자, 542
- 백슬래시 제어 문자
  - Dialers 파일 보내기 문자열, 541
  - Systems 파일 채트 스크립트, 532

## 백업, 메일 서버 및, 327

## 버

## 버전 8.12의 FEATURE() 선언

- 지원, 364
- 지원되지 않음, 366
- 버전 8.12의 LDAP, sendmail 명령 및, 369
- 버전 8.12의 MAILER() 선언, 366
- 버전 8.12의 대기열 기능, sendmail 명령, 368
- 버전 8.12의 매크로
  - m4 구성 매크로(sendmail), 363
  - MAX 매크로(sendmail), 362
  - 정의된 매크로(sendmail), 361
- 버전 8.12의 명령줄 옵션
  - sendmail 명령, 358, 359
- 버전 8.12의 배달 에이전트 플러그인, sendmail 명령, 367
- 버전 8.12의 배달 에이전트에 대한 등식, sendmail 명령, 367
- 버전 레벨, sendmail.cf 파일에 지정, 318
- 버전 협상, NFS, 172-173

## 범

## 범위(SLP)

- DA 및, 233, 246
- default 범위, 244
- 고려 사항, 243-244
- 구성 시기, 243
- 멀티홈 호스트 및, 252
- 배포, 242-245
- 정의, 217
- 프록시 등록 및, 256

## 변

## 변경

- /etc/shells 파일, 306
- .forward-파일 검색 경로, 305

**별**

별, 루프, 308

**별칭**

/etc/mail/aliases 파일, 340, 341

NISaliases 맵, 341

만들기, 325

정의, 325

확인, 308

별표(\*), autofs 맵, 213

**보****보안**

autofs 제한 적용, 111

DH 인증

개요, 191

사용자 인증, 189

암호 보호, 190

/etc/hosts.equiv 파일 문제, 577

NFS 버전 3 및, 74

.rhosts 파일 문제, 578, 580

UNIX 인증, 189, 191

UUCP

Permissions 파일의 COMMANDS 옵션, 550, 551

Permissions 파일의 VALIDATE 옵션, 553

공개 디렉토리 파일의 고정 비트, 523

설정, 522

보안 NFS 시스템

개요, 189

관리, 96

보안 RPC

DH 인증 문제, 192, 193

overview, 190

복사 작업 문제, 589

파일 공유 문제, 158, 160

보안 NFS 시스템

DH 인증 및, 96

개요, 189

관리, 96

도메인 이름, 96

보안 RPC

DH 인증 문제, 192, 193

개요, 190

보안 모드 선택 및 mount 명령, 151

보안 및 NFS

설명, 76, 181-182

오류 메시지, Permission denied, 128

보안 종류, 79

보안 키

데이터베이스, 191

서버 충돌 및, 192

원격 서버에서 삭제, 192

**복**

복제된 마운트, 소프트웨어 옵션 및, 129

복제된 마운트는 소프트웨어가 아니어야 함, 129

복제된 파일 시스템, 185

복제본 마운트는 읽기 전용이어야 함, 129

복제본의 버전이 같아야 함, 129

**부**

부트, 디스크가 없는 클라이언트 보안, 192

부팅, 파일 시스템 마운트, 86

**브**

브로드캐스트(SLP), 238, 246, 249

**비**

비동기 PPP(asppp)

Solaris PPP 4.0과 다른 점, 376

Solaris PPP 4.0으로 변환, 508-509

UUCP 데이터베이스 구성, 515

구성의 파일, 505

설명서, 376

**사**

사서함 이름의 밑줄(\_), 324

사용 권한, 요구 사항 복사, 591

사용 안함, .forward 파일, 304

사용 안함으로 설정

autofs 찾아보기 기능

개요, 112

작업, 112

NCA, 51

NCA 로깅, 52

큰 파일 만들기, 88-89

사용안 함, 단일 클라이언트에 대한 마운트

액세스, 89

사용으로 설정

NCA, 48-51

NCA 로깅, 52

NFS 서버 로깅, 84

WebNFS 서비스, 83-84

클라이언트측 파일오버, 89

사용자 에이전트(SLP), 230

사용자 이름

원격 시스템에 로그인한 사용자 알아보기, 581

직접 또는 간접 로그인(rlogin), 578

현재 사용자, 590

사용자 이름, 우편함 이름 및, 324

## 삭

삭제, .rhosts 파일, 578

삭제된 패킷, 599

## 상

상태 코드, SLP, 261-262

## 서

서버

참조 NFS 서버

autofs 파일 선택, 205

NFS 서버 및 vfstab 파일, 86

NFS 서비스, 72

정보 표시, 597, 602, 604

충돌 및 보안 키, 192

클라이언트 호출 추적, 597, 599

서버 (계속)

홈 디렉토리 서버 설정, 107

서버:경로 이름 마운트 오류, 124

서버 및 클라이언트, NFS 서비스, 72

서버가 응답하지 않음 메시지, 123, 125

원격 마운트 문제, 127

정지된 프로그램, 128

키보드를 통한 중단, 116

서비스 URL

프록시 등록(SLP), 256, 258

서비스 검색(SLP), 238, 239, 245

서비스 관리(SLP), 257

서비스 데이터베이스, UUCP 포트, 521

서비스 알림(SLP), 235

서비스 에이전트(SLP), 230, 235

서비스 요청(SLP), 246

## 설

설정

NISmail.aliases 맵, 295

가상 호스트, 287

로컬 편지 별칭 파일, 296

메일 게이트웨이, 283

메일 서버, 303

메일 클라이언트, 279

메일 호스트, 281

## 셀

셀 스크립트(UUCP), 519, 521

uudemon.admin, 520

uudemon.cleanup, 520

uudemon.hour

uusched 데몬 실행, 513

uuxqt 데몬 실행, 512

uudemon.hour

설명, 520

uudemon.poll, 520, 554

수동 실행, 519

자동 실행, 519



**소**

소켓, NCA 및, 48

**수**

수동 모드, 548

수정, NFS 참조, 115

수퍼 유저, autofs 및 암호, 74

**숫**

숫자 기호(#)

간접 맵의 주석, 199

마스터 맵(auto\_master)의 주석, 196

직접 맵의 주석, 198

**스**

스크립트

셸 스크립트(UUCP), 519, 521

채트 스크립트(UUCP), 533

expect 필드, 531

기본 스크립트, 531

다이얼 백을 사용으로 설정, 532

제어 문자, 532

형식, 531

스폴(UUCP)

uusched 데몬

description, 513

최대 동시 실행, 515

최대 동시 실행 수, 558

관리 파일, 558, 560

디렉토리, 559

작업 등급 정의, 555, 557

정리 명령, 513

슬래시(/) (계속)

루트 디렉토리, 디스크가 없는 클라이언트를  
통한 마운트, 74

마스터 맵 이름 앞에 붙임, 196

**시**

시간

다른 시스템과 동기화, 64

시간 동기화, 191

다른 시스템과, 64

시간 초과(SLP), 239, 246

시스템 Permissions 파일, COMMANDS 옵션, 550

시작

autofs 서비스, 92-93

NFS 서비스, 92

UUCP 셸 스크립트, 519, 521

채트 스크립트를 통해 다이얼 백을 사용으로  
설정, 532

켜기

에코 검사, 542

**신**

신뢰할 수 있는 네트워크 환경

원격 로그인

로그인 후의 프로세스, 579

신뢰할 수 있는 호출자, 385

CHAP 인증을 위해 구성, 437

**실**

실행(X.) UUCP 파일

uuxqt 실행, 512

설명, 560

정리, 520

실행 가능 맵, 210

**쓰**

쓰기 오류, NFS 및, 74

**슬**

슬래시(/)

/- as 마스터 맵 마운트 지점, 199

**압**

## 암호

autofs 및 수퍼 유저 암호, 74  
 DH 암호 보호, 190  
 UUCP 권한 있음, 552  
 원격 로그인에 대한 인증  
   ftp 명령, 583, 585  
 원격 로그인을 위한 인증  
   rlogin 명령, 576, 579, 582

**액**

액세스, NFS 참조, 115

## 액세스 서버(PPP)

/etc/ppp/chap-secrets 파일, 501  
 /etc/ppp/options 파일, 501  
 /etc/ppp/pap-secrets 파일, 501  
 PPPoE 클라이언트로 인터페이스 제한, 444  
 계획 작업 맵, 401  
 구성, PPPoE, 443, 444, 499-501  
 구성 작업 맵, 439-440  
 구성을 위한 명령 및 파일, 496, 497-499  
 정의, 386

## 액세스 제어 목록(ACL) 및 NFS

설명, 76, 181-182  
 오류 메시지, Permission denied, 128

**앰**

앰 퍼센드(&), autofs 맵, 212

**예**

예코 검사, 542

**여**

여러 서버에서 공유 파일 복제, 111  
 여러 파일(ftp), 585

**열**

열기 오류, NFS 및, 74

**예**

예, PPP 구성, 참조 PPP의 구성 예

**오**

오디오 파일, 우편함 공간 요구 사항 및, 327

## 오류 메시지

automount -v를 통해 생성됨, 122  
 sendmail 프로그램, 311  
 권한 거부됨, 127  
 기타 automount 메시지, 123  
 서버가 응답하지 않음  
   원격 마운트 문제, 127, 128  
   정지된 프로그램, 128  
   키보드를 통한 중단, 116  
 쓰기 오류  
   NFS 및, 74  
 열기 오류  
   NFS 및, 74  
 파일 또는 디렉토리 없음, 127

**옵**

## 옵션(PPP)

asynmap, 471  
 auth, 429  
 call, 416, 474  
 connect, 411, 484  
 crtscts, 409  
 debug, 450  
 init, 422, 471  
 local, 422  
 login, 429, 489  
 name, 432  
 noauth, 411, 422  
 noccp, 414  
 noipdefault, 411  
 noservice, 501

## 옵션(PPP) (계속)

passive, 422  
 persist, 422  
 pppd 데몬의 구문 분석, 467  
 sync, 422  
 xonxoff, 415  
 사용 지침, 465-472  
 옵션 권한, 468

## 우

## 우편함

공간 요구 사항, 327  
 메일 서버 및, 327  
 파일, 324

우편함 이름, 324

우편함 이름의 퍼센트 기호(%), 324

## 운

## 운영 체제

맵 변수, 208  
 호환되지 않는 버전 지원, 111

## 원

## 원격 로그인

ftp 명령, 584  
 ftp 연결 닫기, 585  
 ftp 연결 열기, 584, 585  
 .rhosts 파일 제거, 580  
 rlogin 명령 사용, 582  
 도메인, 576  
 로그인 링크 만들기, 578  
 로그인한 사용자 알아보기, 581  
 원격 시스템 작동 확인, 580  
 인증(ftp), 583  
 인증(rlogin), 576, 578  
 /etc/hosts.equiv 파일, 577  
 .rhosts 파일, 578  
 네트워크 인증 또는 원격 시스템 인증, 576, 577

## 원격 로그인 (계속)

인터럽트, 576  
 직접 또는 간접(rlogin), 578, 579  
 원격 로그인 링크 만들기, 578  
 원격 로그인 인터럽트, 576  
 원격 로그인을 위한 네트워크 인증, 576, 577, 579  
 원격 로그인을 위한 시스템 인증, 576, 577  
 원격 마운트  
 데몬 필요, 116  
 문제 해결, 117, 120  
 원격 복사  
 ftp 사용, 584  
 rcp 사용, 589, 593  
 원격 시스템  
 로그아웃(exit), 582, 583  
 로그인, 576, 585  
 원격 복사  
 rcp 사용, 589, 593  
 원격 파일 복사  
 ftp 명령 사용, 584  
 작동 확인, 580  
 정의, 567  
 원격 시스템 연결 닫기, 585  
 원격 시스템 연결 열기, 584, 585  
 원격 실행(UUCP)  
 데몬, 512  
 명령, 547, 550, 553  
 작업 파일 C., 559, 560  
 원격 컴퓨터 폴링(UUCP), 515, 554  
 원격 파일 시스템  
 그룹 마운트 해제, 155  
 원격으로 마운트된 파일 시스템을 포함하는  
 클라이언트 목록, 164

## 위

위임, NFS 버전 4, 179-180

## 유

유니캐스트 경로 지정(SLP), 249  
 사용 안함으로 설정, 251

## 응

응용 프로그램, 정지됨, 128

## 이

이더넷, 메일 구성 테스트, 307  
 이름 서비스, autofs 맵 유지 관리 방법, 103  
 이름 서비스 도메인, 메일 도메인 및, 346  
 이름/이름 지정  
   노드 이름  
     UUCP 별칭, 548  
     UUCP 원격 컴퓨터, 528, 546  
 이미 마운트된 파일 시스템 오버레이, 152  
 이미 마운트됨 메시지, 123

## 익

익명 ftp, 계정, 583

## 인

인바운드 통신  
   UUCP 채트 스크립트를 통해 사용으로 설정, 532  
   콜백 보안, 550  
 인쇄  
   공유되거나 내보낸 파일 목록, 164  
   원격으로 마운트된 디렉토리 목록, 164  
 인증  
   참조 인증(PPP)  
   DH, 191  
   ftp 명령을 사용한 원격 로그인, 583, 584, 585  
   rlogin 명령을 사용한 원격 로그인, 576, 578, 582  
     /etc/hosts.equiv 파일, 577  
     .rhosts 파일, 578  
     네트워크 또는 원격 시스템 인증, 576, 577, 579  
     직접 또는 간접 로그인, 579  
   RPC, 191  
   UNIX, 189, 191  
   일반적인 문제 해결, 463

## 인증(PPP)

CHAP 구성  
   참조 CHAP(Challenge-Handshake 인증  
     프로토콜)  
     다이얼 아웃 시스템, 438  
     다이얼 인 서버, 434, 436  
 CHAP 자격 증명 구성, 437  
 CHAP 자격 증명 데이터베이스 구성, 435  
 CHAP의 예, 398  
 PAP 구성  
   참조 PAP(암호 인증 프로토콜)  
 PAP의 예, 396  
 계획, 395, 398  
 구성 작업 맵, 425-426, 426-427, 433-434  
 구성 전의 필수 조건, 396  
 기본 정책, 384  
 신뢰할 수 있는 호출자, 385  
 암호 파일  
   PAP, 428  
   PPP, 385  
 인증자, 385  
 전용 회선에 대한 지원, 385  
 프로세스 다이어그램  
   PAP, 487  
   피인증자, 385  
 인증 네트워크 환경  
   원격 로그인  
     인증 프로세스, 576  
 인증자(PPP), 385  
 인터페이스(PPP)  
   HSI/P 구성 스크립트, 420  
   PPP 다이얼 아웃을 위한 비동기 인터페이스, 380  
   PPP 다이얼 인을 위한 비동기 인터페이스, 381  
   PPPoE 액세스 서버를 위해 구성, 443, 495  
   PPPoE 클라이언트로 인터페이스 제한, 444  
   PPPoE 클라이언트를 위해 구성, 440-441  
   참조 /etc/ppp/pppoe.if 파일  
   /usr/sbin/sppptun을 사용하여 PPPoE  
     인터페이스 연결, 495  
   전용 회선을 위한 동기, 383

**읽**

- 읽기/쓰기 유형
  - 파일 시스템 공유, 158, 161
  - 파일 시스템 마운트, 151
- 읽기 전용 유형
  - autofs의 파일 선택, 207
  - 파일 시스템 공유, 158, 161
  - 파일 시스템 마운트, 151
- 읽기 전용 파일, autofs의 파일 선택, 205

**임**

- 임시(TM) UUCP 데이터 파일, 559

**자**

- 자격 증명
  - CHAP 인증, 435
  - PAP 인증, 427-428
  - UNIX 인증, 191
  - 설명, 190
- 자동 마운트
  - /var/mail 디렉토리, 280, 327

**작**

- 작업(C.) UUCP 파일
  - 설명, 559, 560
  - 정리, 520
- 작업 디렉토리, rcp 명령에 대한 정의, 590
- 작업 목록, NCA, 47

**잘**

- 잘못된 키 메시지, 122

**잠**

- 잠금, NFS 버전 3의 개선 사항, 77
- 잠금(LCK) UUCP 파일, 559

- 잠금 오류 메시지, 126
- 잠금 제거, 147

**장**

- 장치 전송 프로토콜, 539, 540

**전**

- 전경 파일 마운트 옵션, 149
- 전달 작업(UUCP), 554
- 전송 설정 문제, 오류 메시지, 125
- 전송 프로토콜, NFS 협상, 182
- 전용 회선 링크
  - CSU/DSU, 383
  - demand 스크립트, 423
  - 계획, 393, 394, 395, 421
  - 구성, 394
  - 구성 작업 맵, 419
  - 동기 인터페이스 구성, 420-421
  - 링크에 대한 인증, 385
  - 링크의 각 부분, 382-383
  - 매체, 383
  - 예제 구성, 394
  - 일반적인 문제 진단
    - 개요, 463
    - 네트워크, 451
  - 정의, 382
  - 통신 프로세스, 384
  - 하드웨어, 393
- 전자 메일, UUCP 유지 관리, 522
- 전화선, UUCP 구성, 511

**점**

- 점(.)
  - rcp 명령 구문, 592, 593
  - 도메인 주소, 323
  - 사서함 이름, 324

**정**

정보 없음 메시지, 125  
 정적 주소 지정, PPP, 493  
 정지된 프로그램, 128

**제**

제거  
   NFS 참조, 116  
   참조, 195  
 제어 문자  
   Dialers 파일 보내기 문자열, 541  
   Systems 파일 채트 스크립트, 532

**주**

주석  
   간접 맵, 199  
   마스터 맵(auto\_master), 196  
   직접 맵, 198  
 주소 지정  
   PPP, 492, 493, 494

**중**

중지  
   autofs 서비스, 93  
   NFS 서비스, 92  
   끄기  
     에코 검사, 542

**지**

지연 제어 문자, 542  
 지점 간 프로토콜, 참조 PPP

**직**

직렬 마운트 해제, 155

**직렬 포트**

구성  
   다이얼 아웃 시스템, 407-408  
   다이얼 인 서버, 413  
   다이얼 인 서버에서 구성, 471  
 직접 I/O 마운트 옵션, 149  
 직접 링크, UUCP 구성, 511  
 직접 맵(autofs)  
   automount 명령을 실행해야 하는 경우, 103  
   개요, 198  
   구문, 198  
   설명, 103  
   예제, 197  
   주석, 198  
 직접 원격 로그인  
   rlogin 명령 사용, 582  
   간접 로그인 또는  
     rlogin 명령, 578, 579

**참**

참조, 참조 NFS 참조

**찾**

찾아보기, NFS URL 사용, 99-100  
 찾아보기 기능  
   개요, 80  
   사용 안함으로 설정, 112  
 찾을 수 없음 메시지, 123

**채**

채트 스크립트  
   실행 가능한 chat 프로그램 만들기, 485  
   예(PPP)  
     ISDN TA, 482-483, 483  
     ISP 호출을 위한 스크립트, 479-480  
     UNIX 스타일의 로그인 채트 스크립트, 410,  
       480-482  
     기본적인 모뎀 채트 스크립트, 477-478  
   채트 스크립트 설계, 477

## 채트 스크립트 (계속)

호출, PPP, 484-485

## 총

총돌률(네트워크), 600

## 취

취소, 원격 로그인, 576

## 캐

캐리지 리턴 제어 문자, 542

캐시 및 NFS 버전 3, 74

## 커

커널, 서버의 응답 확인, 117

## 컴

컴파일 플래그, sendmail 명령, 316

## 켜

켜기

에코 검사, 542

채트 스크립트를 통해 다이얼 백을 사용으로  
설정, 532

## 콜

콜백

Permissions 파일 옵션, 550

채트 스크립트를 통해 다이얼 백을 사용으로  
설정, 532

## 큰

큰 파일

NFS 지원, 78

개요, 186

생성을 사용 안함으로 설정, 88-89

## 클

클라이언트

참조 메일 클라이언트, NFS 클라이언트, NTP  
클라이언트 및 PPPoE 클라이언트

서버 호출 추적, 597, 599

정보 표시, 597, 602

정보 표시t, 604

클라이언트 복구, NFS 버전 4, 176-178

클라이언트측 파일오버

NFS 버전 4, 186

NFS 잠금 및, 186

NFS 지원, 78

개요, 184-186

복제된 파일 시스템, 185

사용으로 설정, 89

용어, 185

## 키

키 맵 파일, 만들기, 298

키 파일, NTP, 65

키보드를 통한 마운트 중단, 116

키워드

Devices 파일 Type 필드, 534

Grades 파일, 556, 557

NFS 버전 협상, 172-173

## 터

터널

구성 작업 맵, 439

예제 구성, 401, 403

정의(PPP), 386

## 테

## 테스트

- 규칙 세트, 308
- 다른 시스템과의 메일 연결, 309-310
- 메일 구성, 307
- 패킷 안정성, 597
- 편지 별칭, 308

## 템

## 템플리트 파일(PPP)

- /etc/ppp/myisp-chat.tmpl, 478-479
- /etc/ppp/options.tmpl, 470
- /etc/ppp/peers/myisp.tmpl, 475
- options.ttya.tmpl, 472
- 템플리트 목록, 406

## 토

- 토큰(DTP(Dialer-Token Pairs)), 539
- 토큰(전화 걸기-토큰 쌍), 537

## 통

## 통합 파일 시스템

- DNS 레코드, 91
- 마운트, 91
- 마운트 지점, 197

## 틸

## 틸드(~)

- rcp 명령 구문, 592, 593
- 축약 경로 이름, 590

## 파

## 파운드 기호(#)

- 마스터 맵(auto\_master)의 주석, 196
- 직접 맵의 주석, 198

## 파일 공유

- NFS 버전 3의 개선 사항, 74, 77
- 개요, 158
- 공유 해제, 163
- 나열된 클라이언트만, 158
- 루트 액세스 권한 부여, 160
- 보안 문제, 158, 160, 189
- 여러 서버에서 공유 파일 복제, 111
- 여러 파일 시스템, 163
- 예제, 161
- 인증되지 않은 사용자 및, 159
- 읽기/쓰기 액세스, 158, 161
- 읽기 전용 액세스, 158, 161

## 파일 공유 옵션, 158

## 파일 권한

- NFS 버전 3의 개선 사항, 74
- WebNFS 및, 99
- 파일 또는 디렉토리 없음 메시지, 127
- 파일 및 파일 시스템

## autofs 액세스

- 비 NFS 파일 시스템, 105, 106
- autofs 파일 선택, 205, 207
- NFS ASCII 파일 및 해당 기능, 132
- NFS 파일 및 해당 기능, 131
- NFS의, 72
- NFS의 처리, 72

## 로컬 파일 시스템

- 그룹 마운트 해제, 155
- 원격 파일 시스템
- 그룹 마운트 해제, 155
- 원격으로 마운트된 파일 시스템을 포함하는 클라이언트 목록, 164
- 파일 시스템 표에서 마운트, 155

## 정의된 파일 시스템, 72

## 축약 경로 이름, 590

## 프로젝트 관련 파일 통합, 108

## 파일 복사(원격)

## ftp 사용, 584

## rcp 사용, 589, 593

## 파일 속성 및 NFS 버전 3, 74

## 파일 시스템

- 네트워크 통계, 603, 604
- 파일 시스템 공유 해제
- unshare 명령, 162-163



**파일 시스템 공유 해제 (계속)**

unshareall 명령, 163

파일 시스템 네임스페이스, NFS 버전 4, 174-175

**파일 시스템 마운트**

autofs 및, 87

NFS URL, 90

개요, 85

단일 서버에서 전체 마운트, 87-88

단일 클라이언트에 대한 액세스를 사용 안함으로  
설정, 89

미러 마운트 및, 87

방화벽을 통해, 90

부트 시 방법, 86

수동으로(즉시), 86

작업 맵, 85

파일 시스템 및 NFS 처리, 72

**파일 전송(UUCP)**

daemon, 512

데몬, 512

문제 해결, 524, 525

사용 권한, 548, 550

작업 파일 C., 559, 560

파일 전송 크기, 협상, 182-183

파일이 너무 큼 메시지, 126

**패****패리티**

Dialers 파일, 543

Systems 파일, 533

패킷 크기, SLP 구성, 237

패킷을 보낼 수 없음 메시지, 124

**페****페일오버**

mount 명령 예제, 152

NFS 지원, 78

오류 메시지, 126

**편**

편지 별칭 파일, 관리, 294

**포****포트**

Devices 파일 항목, 536

UUCP, 521

포트매퍼, 마운트 및, 183-184

**프**

프레임 릴레이, 383, 419

프로그램, 정지됨, 128

프로세서 유형 맵 변수, 208

프로세스 다이어그램, CHAP, 490

프로젝트, 파일 통합, 108

프로젝트 관련 파일 통합, 108

프록시 등록(SLP), 256, 258

멀티홈 호스트, 252

프록시 알림(SLP), 255, 257

**플**

플러스 기호(+)

/etc/hosts.equiv 파일 구문, 577

**피****피어**

PPPoE 클라이언트, 386, 400

다이얼 아웃 시스템, 379

다이얼 인 서버, 379

액세스 서버, 386, 401

인증자, 385

전용 회선 피어, 383

정의, 379

피인증자, 385

피인증자(PPP), 385

- 하**
  - 하드웨어
    - UUCP
      - 구성, 511
      - 포트 선택기, 535
    - 흐름 제어
      - Dialers 파일, 543
      - Systems 파일, 533
  - 하이픈(-)
    - Line2 필드 개체 틀, 536
    - Speed 필드 개체 틀, 530
    - 다이얼 번호 약어, 530
- 회**
  - 회발성 파일 핸들, NFS 버전 4, 175–176

- 현**
  - 현재 사용자, 590

- 협**
  - 협상
    - WebNFS 보안, 79
    - 파일 전송 크기, 182–183

- 호**
  - 호스트
    - /etc/hosts.equiv 파일, 577
    - 모든 파일 시스템 마운트 해제, 155
    - 응답 확인, 598
    - 패킷 보내기, 598
    - 호스트가 응답하지 않음 메시지, 123

- 화**
  - 확인, 원격 시스템 작동, 580
  - 확인 오류 메시지, 126

- 회**
  - 회신을 받을 수 없음 메시지, 124