

Oracle® Solaris 11.2의 네트워크 파일 시스템 관리

ORACLE®

부품 번호: E53867
2014년 7월

Copyright © 2002, 2014, 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, 디어셈블리 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 END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

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

이 설명서 사용	9
1 네트워크 파일 시스템 정보	11
NFS 서비스 정보	11
autofs 정보	12
NFS 용어	13
NFS 서버 및 클라이언트	13
NFS 파일 시스템	13
NFS 서비스의 기능	13
NFS 버전 2 프로토콜	14
NFS 버전 3 프로토콜	14
NFS 버전 4 프로토콜	14
NFS 버전 제어	15
NFS ACL 지원	15
TCP를 통한 NFS	16
UDP를 통한 NFS	16
RDMA를 통한 NFS 개요	16
네트워크 잠금 관리자 및 NFS	17
NFS 큰 파일 지원	17
NFS 클라이언트 페일오버	17
NFS 서비스에 대한 Kerberos 지원	17
WebNFS 지원	18
RPCSEC_GSS 보안 종류	18
NFS 마운트를 위한 확장	18
WebNFS 서비스의 보안 협상	18
NFS 서버 로깅	19
autofs 기능	19
Oracle Solaris 11 릴리스의 중요한 변경 사항	19
2 네트워크 파일 시스템 기능	21

NFS 서비스의 작동 방식	21
NFS Over RDMA	22
NFS의 버전 협상	24
NFS 버전 4의 기능	24
UDP 및 TCP 협상	33
파일 전송 크기 협상	33
NFS 버전 3의 파일 시스템 마운트 방법	34
마운트 시 <code>-public</code> 옵션과 NFS URL의 효과	35
클라이언트측 페일오버	35
NFS 서버 로깅의 작동 방식	37
WebNFS 서비스의 작동 방식	37
WebNFS 보안 협상의 작동 방식	38
웹 브라우저 사용 시의 WebNFS 제한	39
보안 NFS 시스템	39
미러 마운트의 작동 방식	42
미러 마운트를 사용하여 파일 시스템 마운트	42
미러 마운트를 사용하여 파일 시스템 마운트 해제	43
NFS 참조의 작동 방식	43
NFS 참조를 사용하는 경우	44
NFS 참조 만들기	44
NFS 참조 제거	44
autofs의 작동 방식	44
autofs가 네트워크(맵)를 탐색하는 방법	46
autofs 맵	46
autofs에서 탐색 프로세스를 시작하는 방법(마스터 맵)	51
autofs 마운트 프로세스	52
autofs에서 클라이언트에 대해 가장 가까운 읽기 전용 파일을 선택하는 방법 (여러 위치)	53
autofs 및 가중치	56
autofs 맵 항목의 변수	57
다른 맵을 참조하는 맵	57
실행 가능 autofs 맵	58
이름 서비스에 대한 기본 autofs 동작	59
autofs 참조	61
autofs 및 메타 문자	61
autofs 및 특수 문자	62
3 네트워크 파일 시스템 관리	63
네트워크 파일 시스템 관리 정보	63

자동 파일 시스템 공유	64
파일 시스템 공유(작업 맵)	64
▼ 자동 파일 시스템 공유를 설정하는 방법	64
▼ NFS 서버 로깅을 사용으로 설정하는 방법	65
파일 시스템 마운트	66
파일 시스템 마운트(작업 맵)	66
▼ 부트 시 파일 시스템을 마운트하는 방법	67
▼ 명령줄에서 파일 시스템을 마운트하는 방법	67
자동 마운트를 사용한 마운트	68
▼ 서버에서 모든 파일 시스템을 마운트하는 방법	68
▼ 클라이언트측 페일오버를 사용하는 방법	69
▼ 단일 클라이언트에 대한 마운트 액세스를 사용 안함으로 설정하는 방 법	69
▼ 방화벽을 통해 NFS 파일 시스템을 마운트하는 방법	70
NFS URL을 사용하여 NFS 파일 시스템 마운트	70
마운트할 수 있는 파일 시스템에 대한 정보 표시	71
NFS 서비스 설정	72
NFS 서비스 시작 및 중지	72
자동 마운트 시작 및 중지	72
다른 NFS 버전 선택	73
보안 NFS 시스템 관리	75
▼ DH 인증을 사용하여 보안 NFS 환경을 설정하는 방법	76
WebNFS 관리	77
WebNFS 액세스 계획	78
▼ WebNFS 액세스를 사용으로 설정하는 방법	79
브라우저를 사용하여 NFS URL 액세스	79
방화벽을 통해 WebNFS 액세스를 사용으로 설정	80
NFS 참조 관리	80
▼ NFS 참조를 만들고 액세스하는 방법	80
▼ NFS 참조를 제거하는 방법	81
FedFS 관리	81
FedFS Server에 대한 DNS 레코드 설정	81
▼ 이름 공간 데이터베이스를 만드는 방법	82
▼ NSDB에 대한 보안 연결을 사용하는 방법	82
▼ FedFS 참조를 만드는 방법	83
4 autofs 관리	85
autofs 관리	85
SMF 매개변수를 사용하여 autofs 환경 구성	86

▼ SMF 매개변수를 사용하여 autofs 환경을 구성하는 방법	86
맵 관련 관리 작업	87
맵 수정	87
▼ 마스터 맵을 수정하는 방법	88
▼ 간접 맵을 수정하는 방법	88
▼ 직접 맵을 수정하는 방법	88
마운트 지점 충돌 방지	89
비 NFS 파일 시스템 액세스	89
자동 마운트 사용자 정의	89
/home의 공통 보기 설정	90
▼ 멀티홉 디렉토리 파일 시스템을 사용하여 /home을 설정하는 방법	90
▼ 공통 디렉토리 아래에서 프로젝트 관련 파일을 통합하는 방법	91
▼ 공유 이름 공간에 액세스하도록 서로 다른 구조를 설정하는 방법	93
▼ 호환되지 않는 클라이언트 운영 체제 버전을 지원하는 방법	94
▼ 여러 서버에서 공유 파일을 복제하는 방법	94
autofs 보안 제한	95
▼ autofs와 함께 공용 파일 핸들을 사용하는 방법	95
▼ autofs와 함께 NFS URL을 사용하는 방법	96
autofs 찾아보기 기능 사용 안함으로 설정	96
5 네트워크 파일 시스템 관리 명령	99
NFS 명령	99
automount 명령	100
clear_locks 명령	100
fsstat 명령	101
mount 명령	102
umount 명령	107
mountall 명령	108
umountall 명령	108
sharectl 명령	109
share 명령	111
unshare 명령	116
shareall 명령	116
unshareall 명령	116
showmount 명령	117
nfsref 명령	118
FedFS 명령	118
6 네트워크 파일 시스템 문제 해결	119

NFS 문제 해결 전략	119
NFS 문제 해결용 명령	120
nfsstat 명령	120
pstack 명령	122
rpcinfo 명령	123
snoop 명령	124
truss 명령	125
NFS 문제 해결 절차	126
▼ NFS 클라이언트에서 연결을 확인하는 방법	126
▼ 원격으로 NFS 서버를 확인하는 방법	127
▼ 서버에서 NFS 서비스를 확인하는 방법	128
▼ NFS 서비스를 다시 시작하는 방법	130
NFS 서비스를 제공하는 호스트 식별	130
▼ mount 명령에 사용되는 옵션을 확인하는 방법	130
autofs 문제 해결	131
automount -v를 통해 생성되는 오류 메시지	131
기타 오류 메시지	133
기타 autofs 오류	135
NFS 오류 메시지	135
7 네트워크 파일 시스템 액세스	141
NFS 파일	141
/etc/default/nfslogd 파일	142
/etc/nfs/nfslog.conf 파일	143
NFS 데몬	144
automountd 데몬	145
lockd 데몬	146
mountd 데몬	146
nfs4cbd 데몬	147
nfsd 데몬	147
nfslogd 데몬	148
nfsmapid 데몬	148
reparseid 데몬	154
statd 데몬	154
색인	157

이 설명서 사용

- 개요 - 네트워크 파일 시스템을 관리 및 액세스하는 방법을 설명합니다.
- 대상 - 시스템 관리자
- 필요한 지식 - 기본 및 일부 고급 네트워크 관리 기술

제품 설명서 라이브러리

이 제품에 대한 최신 정보 및 알려진 문제는 설명서 라이브러리(<http://www.oracle.com/pls/topic/lookup?ctx=E36784>)에서 확인할 수 있습니다.

Oracle 지원 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.

피드백

<http://www.oracle.com/goto/docfeedback>에서 이 설명서에 대한 피드백을 보낼 수 있습니다.

네트워크 파일 시스템 정보

이 장에서는 네트워크를 통해 파일 시스템에 액세스하는 데 사용할 수 있는 NFS(네트워크 파일 시스템) 서비스에 대해 간략하게 소개합니다. NFS 서비스를 사용하면 한 시스템이 다른 시스템의 파일 시스템에 액세스할 수 있습니다. 한 시스템이 네트워크에서 특정 시간에 클라이언트나 서버 역할을 할 수도 있고 클라이언트인 동시에 서버가 될 수도 있습니다. autofs는 NFS 서비스를 통해 공유되는 파일 시스템을 마운트하는 데 사용되는 클라이언트측 서비스입니다. autofs는 자동 마운트 기능을 제공하는 파일 시스템 구조입니다. 이 장에는 NFS 서비스를 이해하는 데 필요한 개념에 대한 설명과, NFS 및 autofs의 최신 기능에 대한 설명이 포함되어 있습니다.

이 장의 내용:

- “NFS 용어” [13]
- “NFS 서비스 정보” [11]
- “autofs 정보” [12]
- “NFS 서비스의 기능” [13]
- “Oracle Solaris 11 릴리스의 중요한 변경 사항” [19]

참고 - 시스템에서 영역이 사용으로 설정된 경우 비전역 영역에서 NFS를 사용하려면 “Oracle Solaris 10 영역 만들기 및 사용”의 “NFS 클라이언트인 비전역 영역”을 참조하십시오.

NFS 서비스 정보

NFS 서비스는 구조가 서로 다르며 각각 다른 운영 체제를 실행하는 여러 시스템이 네트워크를 통해 파일 시스템을 공유할 수 있도록 합니다.

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

NFS 서비스에는 다음과 같은 이점이 있습니다.

- 여러 시스템이 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 시스템이 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. 자주 사용되는 파일을 모든 시스템에 보관할 필요 없이 NFS 서버의 파일 시스템에서 원본 파일을 공유할 수 있습니다. 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 autofs는 자동 마운트 기능을 제공하는 파일 시스템 구조입니다. autofs 파일 시스템은 시스템 부트 시 자동으로 실행되는 automount를 통해 초기화됩니다. 자동 마운트 데몬인 automountd는 지속적으로 실행되어 필요에 따라 원격 파일 시스템을 마운트하고 마운트 해제합니다.

automountd를 실행 중인 클라이언트 시스템에서 원격 파일 시스템에 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.

부트 시 마운트가 필요하지 않고, 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.

automountd를 사용하여 일부 파일 시스템을 마운트하는 경우에도 mount 명령을 사용하여 다른 시스템을 마운트할 수 있습니다. 디스크가 없는 컴퓨터는 mount 명령 및 /etc/vfstab 파일을 통해 /(루트), /usr 및 /usr/kvm을 마운트해야 합니다.

autofs 서비스에 대한 자세한 내용은 다음을 참조하십시오.

- [표 4-1. “autofs 관리 작업”](#)
- [“autofs의 작동 방식” \[44\]](#)

NFS 용어

이 절에서는 NFS 서비스를 사용하려면 이해해야 하는 몇 가지 기본적인 용어를 제공합니다. NFS 서비스에 대한 자세한 내용은 [5장. 네트워크 파일 시스템 관리 명령](#)을 참조하십시오.

NFS 서버 및 클라이언트

클라이언트와 서버라는 용어는 파일 시스템을 공유할 때 시스템이 수행하는 역할을 설명합니다. 서버는 네트워크를 통해 해당 파일 시스템을 공유하는 시스템입니다. 이러한 파일 시스템에 액세스하는 시스템이 클라이언트입니다.

클라이언트는 서버의 공유 파일 시스템을 마운트하여 서버의 파일에 액세스합니다. 클라이언트는 원격 파일 시스템을 마운트할 때 파일 시스템 복사본을 만들지 않습니다. 대신 마운트 프로세스에서는 클라이언트가 서버의 공유 파일 시스템에 투명하게 액세스할 수 있도록 하는 일련의 원격 프로시저 호출을 사용합니다. 이러한 마운트는 로컬 마운트와 비슷합니다. 파일 시스템이 로컬인 것처럼 명령을 입력할 수 있습니다. 자세한 내용은 [“파일 시스템 마운트” \[66\]](#)를 참조하십시오.

NFS 작업을 통해 서버에서 공유된 파일 시스템은 클라이언트에서 액세스할 수 있습니다. `autofs`를 사용하여 NFS 파일 시스템을 자동으로 마운트할 수 있습니다. `autofs`에 대한 자세한 내용은 [“autofs 정보” \[12\]](#)를 참조하십시오. `share` 명령 및 `autofs` 관련 작업에 대한 자세한 내용은 [“자동 파일 시스템 공유” \[64\]](#) 및 [표 4-1. “autofs 관리 작업”](#)을 참조하십시오.

NFS 파일 시스템

NFS 서비스와 공유할 수 있는 객체에는 전체 또는 부분 디렉토리 트리나 파일 계층이 포함되며, 단일 파일도 포함됩니다. 시스템은 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 수행하며, UNIX 이외의 다른 환경에서는 파일 시스템 개념이 무의미할 수 있으므로 파일 시스템은 NFS를 사용하여 공유 및 마운트할 수 있는 파일 또는 파일 계층 구조를 나타냅니다.

NFS 서비스의 기능

이 절에서는 NFS 서비스의 중요한 기능에 대해 설명합니다.

NFS 버전 2 프로토콜

NFS 버전 2는 널리 사용된 최초의 NFS 프로토콜 버전입니다. 모든 Oracle Solaris 릴리스에서 NFS 버전 2 프로토콜을 지원합니다.

NFS 버전 3 프로토콜

NFS 버전 2 프로토콜과 달리, NFS 버전 3 프로토콜은 2GB보다 큰 파일을 처리할 수 있습니다. NFS에서 큰 파일 처리에 대한 자세한 내용은 [“NFS 큰 파일 지원” \[17\]](#)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 않으므로 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(원격 프로시저 호출) 수가 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 제거됩니다. 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. Oracle Solaris 10 릴리스부터는 유선 전송 크기 제한이 완화되었습니다. 전송 크기는 기본 전송 기능을 기반으로 합니다.

NFS 버전 4 프로토콜

NFS 버전 4 프로토콜에서는 사용자 ID와 그룹 ID를 문자열로 표시합니다. `nfsmapid` 데몬은 NFS 버전 4 클라이언트 및 서버에서 다음 매핑에 사용됩니다.

- 사용자 ID 및 그룹 ID 문자열을 로컬 숫자 ID에 매핑
- 로컬 숫자 ID를 사용자 ID 및 그룹 ID 문자열에 매핑

`nfsmapid` 데몬에 대한 자세한 내용은 [“NFS 데몬” \[144\]](#)을 참조하십시오.

NFS 버전 4에서는 `nfsmapid` 데몬을 사용하여 서버의 ACL(액세스 제어 목록) 항목에 있는 사용자 ID 또는 그룹 ID를 클라이언트의 ACL 항목에 있는 사용자 ID 또는 그룹 ID로 매핑함

니다. 그 반대의 경우도 마찬가지입니다. 사용자 ID 및 그룹 ID 매핑에 대한 자세한 내용은 [“NFS 버전 4의 ACL 및 nfsmapid” \[31\]](#) 및 [“NFS ACL 지원” \[15\]](#)을 참조하십시오.

NFS 버전 4를 사용하는 경우 파일 시스템 공유를 해제할 때 해당 파일 시스템의 모든 열린 파일 또는 파일 잠금에 대한 상태 정보가 모두 삭제됩니다. NFS 버전 3의 경우에는 서버에서 파일 시스템 공유를 해제하기 전에 클라이언트가 획득한 잠금을 유지합니다. NFS 버전 4의 파일 시스템 공유 해제에 대한 자세한 내용은 [“NFS 버전 4에서 파일 시스템 공유 해제 및 다시 공유” \[25\]](#)를 참조하십시오.

NFS 버전 4 서버에서는 의사 파일 시스템을 사용하여 서버에서 내보낸 객체에 대한 액세스 권한을 클라이언트에 제공합니다. 의사 파일 시스템에 대한 자세한 내용은 [“NFS 버전 4의 파일 시스템 이름 공간” \[25\]](#)을 참조하십시오. NFS 버전 4에서는 휘발성 파일 핸들이 지원됩니다. 자세한 내용은 [“NFS 버전 4의 휘발성 파일 핸들” \[27\]](#)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. 위임에 대한 자세한 내용은 [“NFS 버전 4의 위임” \[30\]](#)을 참조하십시오.

NFS 버전 4에서는 LIPKEY/SPKM 보안을 지원하지 않습니다.

또한 NFS 버전 4에서는 다음 데몬이 사용되지 않습니다.

- lockd
- nfslogd
- statd

NFS 버전 4 기능의 전체 목록은 [“NFS 버전 4의 기능” \[24\]](#)을 참조하십시오.

NFS 서비스 설정에 대한 자세한 내용은 [“NFS 서비스 설정” \[72\]](#)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. 클라이언트 및 서버 매개변수에 대한 자세한 내용은 [“NFS 데몬” \[144\]](#)을 참조하십시오. NFS 데몬의 매개변수 값에 대한 자세한 내용은 [nfs\(4\)](#) 매뉴얼 페이지를 참조하십시오.

NFS ACL 지원

ACL(액세스 제어 목록)은 표준 UNIX 파일 권한을 사용하지 않고 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 구현에서는 이전의 POSIX 드래프트 스타일 ACL이 지원됩니다. POSIX 드래프트 ACL은 UFS에서 기본적으로 지원됩니다. POSIX 드래프트 ACL에 대한 자

세한 내용은 [“Oracle Solaris 11.2의 파일 보안 및 파일 무결성 확인”](#)의 [“액세스 제어 목록을 사용하여 UFS 파일 보호”](#)를 참조하십시오.

NFS 버전 4 프로토콜에서는 NFS 버전 4 스타일 ACL이 지원됩니다. NFS 버전 4 ACL은 기본적으로 Oracle Solaris ZFS에서 지원됩니다. NFS 버전 4 ACL의 모든 기능을 사용하려면 NFS 버전 4 서버에서 ZFS를 기본 파일 시스템으로 사용해야 합니다. NFS 버전 4 ACL에는 다양한 상속 등록 정보 세트와 표준 읽기/쓰기/실행 이외의 권한 비트 세트가 포함되어 있습니다. ACL을 사용하여 ZFS 파일을 보호하는 방법에 대한 자세한 내용은 [“Oracle Solaris 11.2의 ZFS 파일 시스템 관리”](#)의 7 장, [“ACL 및 속성을 사용하여 Oracle Solaris ZFS 파일 보호”](#)를 참조하십시오. NFS 버전 4의 ACL 지원에 대한 자세한 내용은 [“NFS 버전 4의 ACL 및 nfsmapid” \[31\]](#)를 참조하십시오.

TCP를 통한 NFS

NFS 프로토콜에 대한 기본 전송 프로토콜은 TCP(Transmission Control Protocol)입니다. 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 NFS 버전 2, NFS 버전 3 및 NFS 버전 4 프로토콜에서 작동합니다.

참고 - 시스템에서 InfiniBand 하드웨어를 사용할 수 있으면 기본 전송 프로토콜은 TCP에서 RDMA(Remote Direct Memory Access) 프로토콜로 변경됩니다. 자세한 내용은 [“RDMA를 통한 NFS 개요” \[16\]](#) 및 [“NFS Over RDMA” \[22\]](#)를 참조하십시오. `proto=tcp` 마운트 옵션을 사용하는 경우에는 NFS 마운트에서 TCP만 사용하도록 강제 지정됩니다.

UDP를 통한 NFS

Oracle Solaris 11 릴리스부터는 NFS 클라이언트에서 구성 가능한 UDP(User Datagram Protocol) 예약 포트만 사용됩니다. 두 개 이상의 포트를 사용하게 시스템을 구성하여 시스템 성능을 향상할 수 있습니다. 이 기능은 처음부터 이 방식으로 구성 가능했던 TCP를 통한 NFS 지원을 미러합니다. NFS 환경 조정에 대한 자세한 내용은 [“Oracle Solaris 11.2 조정 가능 매개변수 참조 설명서”](#)를 참조하십시오.

RDMA를 통한 NFS 개요

시스템에서 InfiniBand 하드웨어를 사용할 수 있으면 기본 전송 프로토콜은 TCP에서 RDMA 프로토콜로 변경됩니다. RDMA 프로토콜은 고속 네트워크에서 메모리 간 데이터를 전송하는 기술입니다. 구체적으로, RDMA는 CPU를 사용하지 않고 메모리에서/메모리로 직접 원격 데이터를 전송하는 기능을 제공합니다. 이 기능을 제공하기 위해 RDMA는 InfiniBand의 상호 연결 I/O 기능과 Oracle Solaris OS를 결합합니다. 그러나

proto=tcp 마운트 옵션을 사용하는 경우에는 NFS 마운트에서 TCP만 사용하도록 강제 지정됩니다. NFS에 대해 RDMA 프로토콜을 사용하는 방법에 대한 자세한 내용은 “[NFS Over RDMA](#)” [22]를 참조하십시오.

네트워크 잠금 관리자 및 NFS

네트워크 잠금 관리자는 NFS에서 공유되는 모든 파일에 대해 UNIX 레코드 잠금 기능을 제공합니다. 잠금 방식을 통해 클라이언트는 다른 클라이언트와 I/O 요청을 동기화하여 데이터 무결성을 보장할 수 있습니다.

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

NFS 큰 파일 지원

NFS 버전 3 프로토콜은 2GB보다 큰 파일을 처리할 수 있지만 NFS 버전 2 프로토콜은 처리할 수 없습니다.

NFS 클라이언트 페일오버

읽기 전용 파일 시스템의 동적 페일오버는 매뉴얼 페이지, 기타 설명서 및 공유 이진과 같은 이미 복제된 읽기 전용 리소스에 대한 상위 레벨 가용성을 제공합니다. 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 않는다는 점을 제외하면 자동 마운트는 변경되지 않았습니다. 자세한 내용은 [클라이언트측 페일오버를 사용하는 방법](#) [69] 및 “[클라이언트측 페일오버](#)” [35]를 참조하십시오.

NFS 서비스에 대한 Kerberos 지원

NFS 클라이언트 및 서버에서 Kerberos를 지원하도록 구성하면 NFS 서비스는 Kerberos 버전 5 인증, 무결성 및 개인 정보를 지원합니다. 보안 인증으로 Kerberos를 사용할 때 mount 및 share 명령줄 옵션을 사용할 수 있습니다. Kerberos 버전 5 인증에 대한 자세한 내용은 “[Oracle Solaris 11.2의 Kerberos 및 기타 인증 서비스 관리](#)”의 “[Kerberos NFS 서버 구성](#)”을 참조하십시오.

WebNFS 지원

WebNFS는 인터넷의 파일 시스템에 방화벽을 통해 액세스할 수 있게 하는 기능을 제공합니다. 이 기능에는 NFS 프로토콜 확장이 사용됩니다. 인터넷 액세스용으로 WebNFS™ 프로토콜을 사용하는 경우의 한 가지 이점은 안정성입니다. 이 서비스는 NFS 버전 3 및 버전 2 프로토콜의 확장으로 작성되었습니다. 또한 WebNFS에서는 익명 ftp 사이트의 관리 오버헤드 없이 파일을 공유할 수 있습니다. WebNFS에 대한 자세한 내용은 [“WebNFS 서비스의 보안 협상” \[18\]](#) 및 [“WebNFS 관리” \[77\]](#)를 참조하십시오.

참고 - 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 [“NFS 서비스에 대한 Kerberos 지원” \[17\]](#)을 참조하십시오. GSS-API에 대한 자세한 내용은 [“Developer’s Guide to Oracle Solaris 11 Security”](#)를 참조하십시오.

NFS 마운트를 위한 확장

Oracle Solaris에서 NFS 서비스는 mount 및 automountd 명령에 대한 확장을 제공합니다. 이러한 확장을 통해 마운트 요청이 MOUNT 프로토콜 대신 공용 파일 핸들을 사용할 수 있습니다. WebNFS 서비스에서는 MOUNT 프로토콜을 액세스 방법으로 사용합니다. 공용 파일 핸들을 사용하면 방화벽을 통해 마운트를 수행할 수 있습니다. 서버와 클라이언트 간의 트랜잭션 수가 더 적으므로 마운트 속도가 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 서비스에 대한 자세한 내용은 [“WebNFS 지원” \[18\]](#)을 참조하십시오.

WebNFS 서비스의 보안 협상

NFS 서비스를 사용하여 WebNFS 클라이언트가 NFS 서버와 보안 방식을 협상할 수 있습니다. WebNFS 클라이언트에서는 프로토콜을 사용하여 NFS 서버와 보안 방식을 협상합니다. 이 프로토콜을 통해 WebNFS 서비스에서 보안 트랜잭션을 사용할 수 있습니다. WebNFS의 보안 협상에 대한 자세한 내용은 [“WebNFS 보안 협상의 작동 방식” \[38\]](#)을 참조하십시오.

NFS 서버 로깅

참고 - NFS 버전 4에서는 서버 로깅 기능이 지원되지 않습니다.

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 로깅할 작업을 선택할 수도 있습니다. NFS 서버 로깅 기능은 NFS 및 WebNFS 클라이언트가 익명 FTP 아카이브를 사용할 수 있도록 하는 사이트의 경우 특히 유용합니다. 자세한 내용은 [NFS 서버 로깅을 사용으로 설정하는 방법 \[65\]](#)을 참조하십시오.

autofs 기능

autofs는 로컬 이름 공간에 지정된 파일 시스템에서 작동합니다. 이 정보는 NIS(네트워크 정보 서비스) 또는 로컬 파일에서 유지 관리할 수 있습니다. autofs에서 지원되는 기능은 다음과 같습니다.

- 완전한 다중 스레드 버전의 automountd 기능을 통해 autofs의 안정성이 높아집니다. 이 기능을 통해 여러 마운트를 동시에 처리할 수 있어 서버를 사용할 수 없어도 서비스가 정지되지 않습니다.
- automountd 기능은 주문형 마운트 기능도 제공합니다. 최상위 파일 시스템만 마운트됩니다. 이 마운트 지점과 관련된 다른 파일 시스템은 필요한 경우 마운트됩니다.
- autofs 서비스에서는 간접 맵의 "찾아보기 기능"을 지원합니다. 이 기능이 지원되므로 사용자는 각 시스템을 실제로 마운트하지 않고도 마운트 가능한 디렉토리를 확인할 수 있습니다. -nobrowse 옵션을 사용하면 /net 및 /home 등의 대규모 파일 시스템이 자동으로 찾아보기 가능하도록 지정되지 않습니다. 또한 automount 명령에서 -n 옵션을 사용하여 각 클라이언트에서 autofs 찾아보기 기능을 해제할 수도 있습니다. autofs 찾아보기 기능을 사용 안함으로 설정하는 다양한 방법에 대한 자세한 내용은 ["autofs 찾아보기 기능 사용 안함으로 설정" \[96\]](#)을 참조하십시오.

Oracle Solaris 11 릴리스의 중요한 변경 사항

Oracle Solaris 11 릴리스에서는 다음과 같은 기능이 향상되었습니다.

- 새로운 등록 정보 nfs_props/showmount_info가 /network/nfs/server:default 서비스에 추가되었습니다. 이 등록 정보는 showmount 명령이 원격 클라이언트에 표시하는 정보량을 제어합니다. nfs_props/showmount_info 등록 정보에 대한 자세한 내용은 [showmount\(1M\)](#) 매뉴얼 페이지를 참조하십시오.
- FedFS(통합 파일 시스템) 참조에 대한 지원이 추가되었습니다. 이 기능을 사용하여 여러 서버의 참조 정보를 LDAP에서 집중하여 관리할 수 있습니다. FedFS 참조에 대한 자세한 내용은 ["FedFS 관리" \[81\]](#)를 참조하십시오.

- 이전에 `/etc/default/autofs` 및 `/etc/default/nfs` 파일을 편집하여 설정했던 구성 등록 정보를 이제는 SMF(서비스 관리 기능) 저장소에서 설정할 수 있습니다. 새로운 SMF 등록 정보 및 새 SMF 등록 정보를 사용하는 데몬에 대한 자세한 내용은 [“NFS 데몬” \[144\]](#)을 참조하십시오.
- NFS 서비스에서는 미리 마운트를 지원합니다. NFS 버전 4 클라이언트에서는 미리 마운트를 통해 서버 이름 공간에서 공유 파일 시스템 마운트 지점을 순회할 수 있습니다. NFS 버전 4 마운트의 경우 자동 마운트는 서버 이름 공간 루트 마운트를 수행하며 미리 마운트를 통해 해당 파일 시스템에 액세스합니다. 자동 마운트와 비교할 때 미리 마운트가 제공하는 가장 큰 이점은, 미리 마운트를 사용하여 파일 시스템을 마운트하는 경우 자동 마운트 맵 관리 시의 오버헤드가 발생하지 않는다는 것입니다. 미리 마운트에서 제공하는 기능은 다음과 같습니다.
 - 이름 공간 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.

미리 마운트에 대한 자세한 내용은 다음을 참조하십시오.

 - [서버에서 모든 파일 시스템을 마운트하는 방법 \[68\]](#)
 - [“미리 마운트의 작동 방식” \[42\]](#)
- NFS 참조가 NFS 서비스에 추가되었습니다. 참조는 NFS 버전 4 클라이언트가 파일 시스템을 찾기 위해 이동할 수 있는 서버 기반 재지정입니다. NFS 서버에서는 `nfsref` 명령으로 생성된 참조를 지원합니다. NFS 버전 4 클라이언트는 이러한 참조를 따라 이동하여 실제 위치에서 파일 시스템을 마운트합니다. 자동 마운트 맵을 편집하는 대신 참조를 만들 수 있습니다. NFS 참조에서 제공하는 기능은 다음과 같습니다.
 - 모든 미리 마운트 기능
 - 자동 마운트와 유사한 기능(자동 마운트에 종속되지는 않음)
 - 클라이언트나 서버에서 설정을 수행하지 않아도 됨

NFS 참조에 대한 자세한 내용은 다음 항목을 참조하십시오.

 - [“NFS 참조 관리” \[80\]](#)
 - [“NFS 참조의 작동 방식” \[43\]](#)
 - `nfsref(1M)` 매뉴얼 페이지
- FedFS 이름 공간의 DNS 도메인 루트별로 마운트하는 기능이 추가되었습니다. 이 마운트 지점을 NFS 참조와 함께 사용하면 파일 서버 간을 연결하여 대형 이름 공간을 원하는 대로 작성할 수 있습니다. FedFS 도메인 루트에 대한 자세한 내용은 다음을 참조하십시오.
 - [“FedFS Server에 대한 DNS 레코드 설정” \[81\]](#)
 - [“마운트 지점 /nfs4” \[48\]](#)
- `sharectl` 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)` 매뉴얼 페이지를 참조하십시오.

◆◆◆ 2 장

네트워크 파일 시스템 기능

이 장에서는 기타 전송 프로토콜에 대한 RDMA(Remote Direct Memory Access) 프로토콜의 관계를 설명합니다. RDMA는 NFS에 대한 기본 전송입니다. 이 장에서는 파일 공유용으로 NFS 버전 4에서 도입된 기능과 버전 협상을 포함하는 NFS 서비스 작동 방식에 대해 설명합니다.

이 장의 내용:

- “NFS Over RDMA” [22]
- “NFS 서비스의 작동 방식” [21]
- “미러 마운트의 작동 방식” [42]
- “NFS 참조의 작동 방식” [43]
- “autofs 맵” [46]
- “autofs의 작동 방식” [44]
- “autofs 참조” [61]

참고 - 시스템에서 영역이 사용으로 설정된 경우 비전역 영역에서 이 기능을 사용하려면 “Oracle Solaris 영역 소개”를 참조하십시오.

NFS 서비스의 작동 방식

다음 절에서는 NFS 소프트웨어의 몇 가지 복잡한 기능에 대해 설명합니다. 이 절의 일부 기능 설명은 NFS 버전 4에만 해당합니다.

- “NFS의 버전 협상” [24]
- “NFS 버전 4의 기능” [24]
- “UDP 및 TCP 협상” [33]
- “파일 전송 크기 협상” [33]
- “NFS 버전 3의 파일 시스템 마운트 방법” [34]
- “마운트 시 -public 옵션과 NFS URL의 효과” [35]
- “클라이언트측 페일오버” [35]

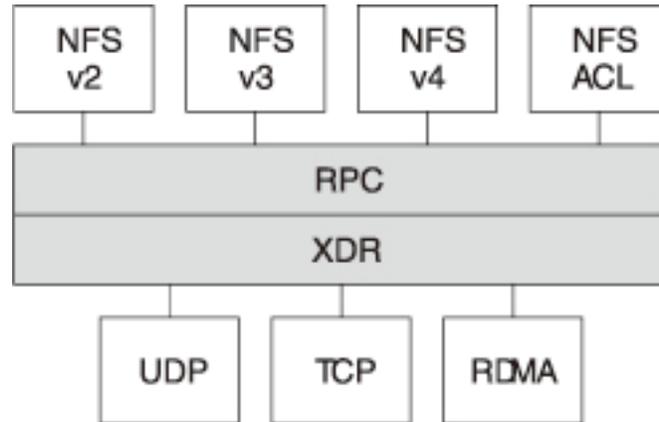
- “NFS 서버 로깅의 작동 방식” [37]
- “WebNFS 서비스의 작동 방식” [37]
- “웹 브라우저 사용 시의 WebNFS 제한” [39]
- “보안 NFS 시스템” [39]
- “보안 RPC” [40]

참고 - 시스템에서 영역이 사용으로 설정된 경우 비전역 영역에서 이 기능을 사용하려면 “Oracle Solaris 영역 소개”를 참조하십시오.

NFS Over RDMA

Oracle Solaris 11.1 릴리스부터 NFS의 기본 전송은 RDMA(Remote Direct Memory Access) 프로토콜입니다. 이 프로토콜은 고속 네트워크를 통해 메모리 간 데이터 전송을 제공합니다. 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 CPU를 해제할 뿐 아니라 호스트 메모리 및 I/O 버스에 대한 경합도 줄여 줍니다. 이 기능을 제공하기 위해 RDMA는 SPARC 및 x86 플랫폼에서 사용할 수 있는 InfiniBand의 상호 연결 I/O 기술을 Oracle Solaris 운영 체제와 결합합니다. 다음 그림에서는 UDP, TCP 등의 다른 프로토콜에 대한 RDMA의 관계를 보여줍니다.

그림 2-1 RDMA와 다른 프로토콜 간의 관계



NFS는RPC위에 배치되는 프로토콜 제품군입니다.
XDR (eXternal Data Representation) 계층은
RPC인수 및 RPC결과를
UDP,TCP, RDMA 등의
여러 RPC전송 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 파일 시스템 공유도 RDMA 전송에서 작동합니다. RDMA 전송을 통한 NFS 마운트는 클라이언트와 서버 간의 InfiniBand 연결이 있으면 투명하게 수행됩니다. InfiniBand 연결 기능은 SPARC 및 x86 플랫폼에서 모두 작동합니다. 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 사용하면 NFS 마운트가 RDMA만 사용하도록 강제 지정됩니다.

TCP 및 UDP만 사용하도록 지정하려면 proto=tcp/udp mount 옵션을 사용합니다. 이 옵션을 사용하면 NFS 클라이언트에서 RDMA가 사용 안함으로 설정됩니다. NFS 마운트 옵션에 대한 자세한 내용은 [mount_nfs\(1M\)](#) 및 [mount\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

참고 - 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.

NFS의 버전 협상

NFS 초기화 프로세스에는 서버 및 클라이언트의 프로토콜 버전 레벨 협상이 포함됩니다. 버전 레벨을 지정하지 않으면 최상의 레벨이 기본적으로 선택됩니다. 예를 들어 클라이언트와 서버가 모두 NFS 버전 3을 지원할 수 있으면 이 버전이 사용됩니다. 클라이언트나 서버 중 하나만 NFS 버전 2를 지원할 수 있으면 NFS 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 매개변수의 기본값을 대체합니다. 클라이언트와 서버 둘 다에 대해 기본 최소값은 2이고 기본 최대값은 4입니다. 서버에서 지원하는 버전을 찾기 위해 NFS 클라이언트는 먼저 `client_versmax`의 값을 확인한 다음 `client_versmin`의 버전 값에 도달할 때까지 각 버전을 계속 시도해 봅니다. 지원되는 버전을 찾는 즉시 프로세스가 종료됩니다. 예를 들어 `client_versmax=4`이고 `client_versmin=2`이면 클라이언트는 NFS 버전 4를 먼저 시도한 후에 NFS 버전 3, NFS 버전 2 순으로 시도합니다. `client_versmin` 및 `client_versmax`를 같은 값으로 설정하면 클라이언트는 항상 해당 버전을 사용하며 다른 버전을 시도하지 않습니다. 서버에서 해당 버전을 제공하지 않으면 마운트가 실패합니다.

참고 - `vers` 옵션을 `mount` 명령과 함께 사용하여 NFS의 버전 협상을 통해 결정되는 값을 대체할 수 있습니다. `mount` 명령의 사용 가능한 옵션에 대한 자세한 내용은 [mount_nfs\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

NFS 서비스 설정에 대한 자세한 내용은 “[NFS 서비스 설정](#)” [72]을 참조하십시오.

NFS 버전 4의 기능

이 절에서는 NFS 버전 4에서 도입된 기능에 대해 설명합니다.

- “[NFS 버전 4에서 파일 시스템 공유 해제 및 다시 공유](#)” [25]
- “[NFS 버전 4의 파일 시스템 이름 공간](#)” [25]
- “[NFS 버전 4의 휘발성 파일 핸들](#)” [27]
- “[NFS 버전 4의 클라이언트 복구](#)” [28]
- “[NFS 버전 4의 OPEN 공유 지원](#)” [29]
- “[NFS 버전 4의 위임](#)” [30]
- “[NFS 버전 4의 ACL 및 nfsmapid](#)” [31]

참고 - Oracle Solaris 10 릴리스부터는 NFS 버전 4가 LIPKEY/SPKM 보안 종류를 지원하지 않습니다. 또한 NFS 버전 4에서는 `mountd`, `nfslogd` 및 `statd` 데몬을 사용하지 않습니다.

NFS 서비스 설정에 대한 자세한 내용은 “[NFS 서비스 설정](#)” [72]을 참조하십시오.

NFS 버전 4에서 파일 시스템 공유 해제 및 다시 공유

NFS 버전 3 및 NFS 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 파일 잠금 또는 열린 파일에 대한 모든 상태 정보가 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 상태 정보가 삭제되지는 않습니다.

NFS 버전 4의 클라이언트 복구에 대한 자세한 내용은 [“NFS 버전 4의 클라이언트 복구” \[28\]](#)를 참조하십시오. `unshare` 명령의 사용 가능한 옵션에 대한 자세한 내용은 [`unshare_nfs\(1M\)` 매뉴얼 페이지](#)를 참조하십시오.

NFS 버전 4의 파일 시스템 이름 공간

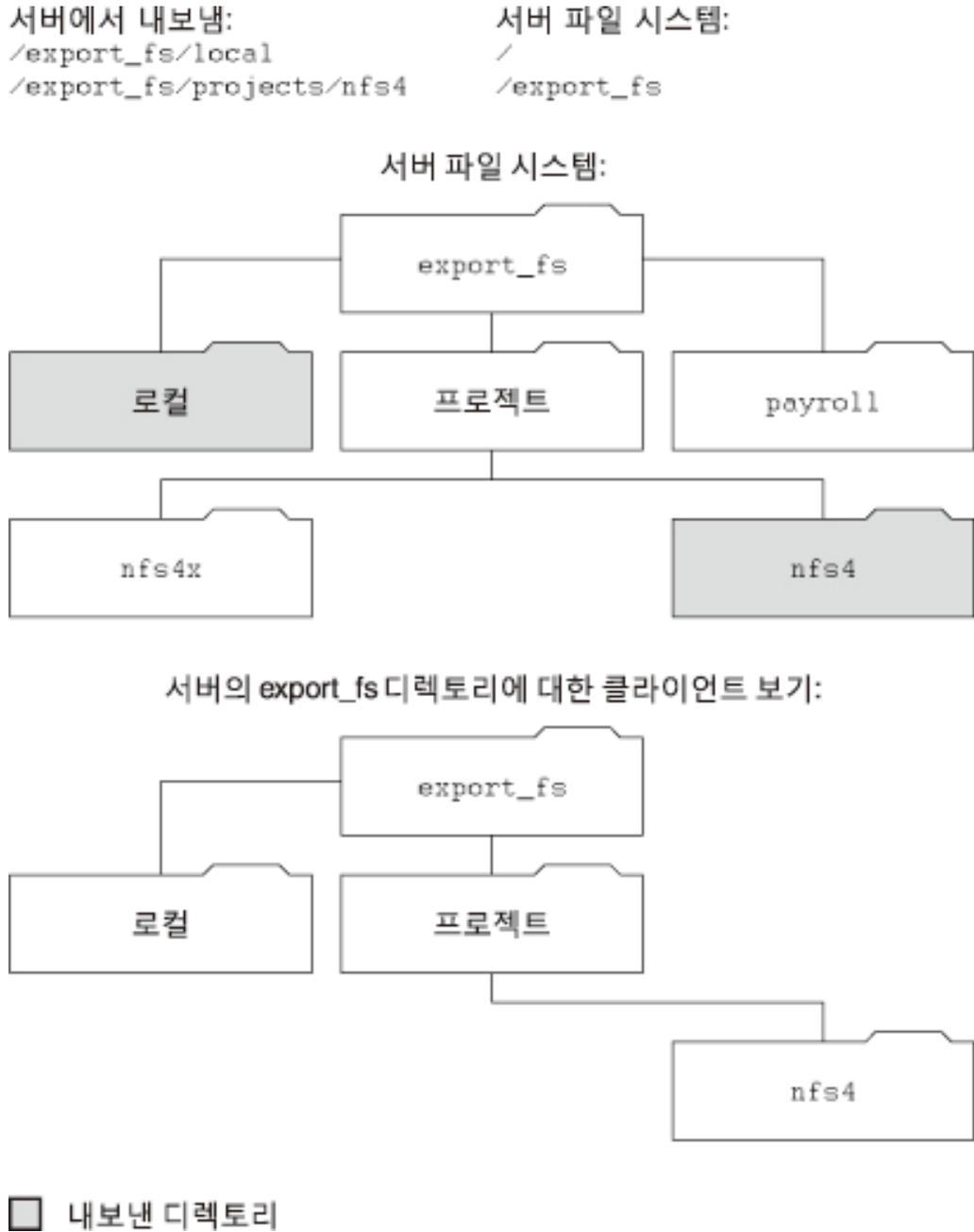
NFS 버전 4 서버는 의사 파일 시스템을 만들고 유지 관리합니다. 따라서 클라이언트가 서버의 모든 내보낸 객체에 원활하게 액세스할 수 있습니다. NFS 버전 4 이전에는 의사 파일 시스템이 없었습니다. 즉, 클라이언트는 액세스할 각 공유 서버 파일 시스템을 강제로 마운트해야 했습니다.

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 이름 공간이 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. 그러나 각 운영 체제에 따라 클라이언트가 각 서버 파일 시스템을 마운트해야 할 수도 있습니다.

그림 2-2 NFS 버전 4의 서버 파일 시스템 및 클라이언트 파일 시스템 보기



그림에 표시된 예에서 클라이언트는 payroll 디렉토리 및 nfs4x 디렉토리를 볼 수 없습니다. 이러한 디렉토리는 내보내지지 않으므로 내보낸 디렉토리에 포함되지 않기 때문입니다. 그러나 local 디렉토리는 클라이언트에 표시됩니다. local은 내보낸 디렉토리가기 때문입니다. projects 디렉토리는 클라이언트에 표시됩니다. projects는 내보낸 디렉토리(nfs4)에 포함되기 때문입니다. 따라서 명시적으로 내보내지 않은 서버 이름 공간 부분은 내보낸 디렉토리 및 서버 내보내기에 포함되는 디렉토리만 표시되는 의사 파일 시스템에 브리지됩니다.

NFS 버전 4의 휘발성 파일 핸들

파일 핸들은 서버에서 만들어지며 파일 및 디렉토리를 고유하게 식별하는 정보를 포함합니다. NFS 버전 2 및 NFS 버전 3에서는 서버에서 영구 파일 핸들을 반환했습니다. 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.
- 서버가 재부트된 경우에도 파일 핸들은 동일하게 유지됩니다.

그러므로 서버에서 파일 핸들을 포함하는 클라이언트로부터의 요청을 받은 경우 파일이 즉시 확인되며 파일 핸들이 항상 올바른 파일을 참조합니다.

대부분 UNIX 기반 서버에서는 영구 파일 핸들을 사용하여 NFS 작업에 대한 파일 및 디렉토리 식별하는 방법으로 충분합니다. 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. 파일 핸들이 변경되면 클라이언트는 새 파일 핸들을 찾아야 합니다.

NFS 버전 2 및 NFS 버전 3 서버 같은 Oracle Solaris NFS 버전 4 서버에서는 항상 영구 파일 핸들을 제공합니다. 그러나 Oracle 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.

참고 - 서버는 영구 파일 핸들과 휘발성 파일 핸들을 항상 클라이언트에 알려 줍니다.

휘발성 파일 핸들은 다음과 같은 상황에서 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 서버에서 둘 다 열린 파일과 파일 잠금에 대한 현재 정보를 유지 관리합니다.

서버가 충돌하여 재부트되면 서버 상태가 손실됩니다. 클라이언트는 서버가 재부트되었음을 감지하고 서버에서 오류 이전에 종료된 열기 및 잠금 상태를 재설정하도록 도와주는 프로세스를 시작합니다. 클라이언트가 프로세스를 진행하므로 이 프로세스를 클라이언트 복구라고 합니다.

클라이언트는 서버가 재부트되었음을 감지하면 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 server-name
```

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 프로세스가 완료될 때까지 기다린 후에 진행됩니다.

클라이언트 복구 프로세스가 완료되면 다음 메시지가 시스템 오류 로그 `/var/adm/messages`에 표시됩니다.

```
NOTICE: Recovery done for server server-name
```

그러면 상태 메시지를 서버로 보내는 클라이언트의 작업이 정상적으로 완료된 것입니다. 그러나 클라이언트가 이 프로세스를 완료했다 해도 다른 클라이언트가 프로세스를 완료하지 않았을 수 있습니다. 따라서 유예 기간이라는 일정 기간에 서버는 모든 클라이언트가 복구를 완료할 수 있도록 열기 또는 잠금 요청을 허용하지 않습니다.

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 기간에는 다음 메시지가 표시됩니다.

```
NFS server recovering
```

유예 기간에도 파일을 열지 않거나 파일 잠금을 설정하지 않는 명령은 계속 실행할 수 있습니다. 예를 들어 `ls` 및 `cd` 명령은 파일을 열거나 파일 잠금을 설정하지 않고 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 분할 영역이 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.

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 n): file : filename
```

복구하는 동안 파일 잠금 재설정에 실패하면 다음 오류 메시지가 표시됩니다.

```
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. 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 EACCES에 매핑됩니다.

프로토콜에서 여러 공유 모드를 제공하는 것은 하지만, 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.

참고 - `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 클라이언트에 동시에 부여할 수 있습니다. 쓰기 위임은 다른 클라이언트의 파일 액세스와 충돌하므로 한 클라이언트에만 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.

클라이언트는 위임을 요청하지 않습니다. 위임을 부여할지 여부는 파일의 액세스 패턴에 따라 전적으로 서버에서 결정합니다. 최근에 여러 클라이언트가 쓰기 모드에서 파일에 액세스한 경우 이 액세스 패턴은 향후 충돌 가능성을 나타내므로 서버는 위임을 부여할 수 없습니다.

클라이언트가 현재 파일에 대해 부여된 위임과 일치하지 않는 방식으로 해당 파일에 액세스하면 충돌이 발생합니다. 예를 들어 클라이언트가 파일에 대한 쓰기 위임을 보유하고 있는데 두 번째 클라이언트가 읽기 또는 쓰기 액세스를 위해 해당 파일을 열면 서버에서 첫 번째 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 두 번째 클라이언트에는 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 해결할 수 없습니다. 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 시작할 수 있습니다. 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 NFS 버전 2와 NFS 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 정지했다가 나중에 다시 시도합니다. 클라이언트는 File unavailable 메시지를 표시합니다.
- NFS 버전 2의 경우에는 JUKEBOX 오류에 해당하는 항목이 없기 때문에 서버에서 응답을 하지 않으며, 클라이언트는 기다렸다가 다시 시도합니다. 클라이언트는 NFS server not responding 메시지를 표시합니다.

위임 충돌이 해결되면 오류 메시지가 제거됩니다.

기본적으로 서버 위임은 사용으로 설정됩니다. `server_delegation` 매개변수를 `off`로 설정하여 위임을 사용 안함으로 설정할 수 있습니다.

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

클라이언트 위임에는 키워드가 필요하지 않습니다. NFS 버전 4 콜백 데몬 `nfs4cbd`는 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 NAT(Network Address Translation)를 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`

ACL(액세스 제어 목록)은 파일 소유자가 파일 소유자, 그룹 및 기타 특정 사용자/그룹의 파일 권한을 정의할 수 있도록 하여 파일 보안을 제공합니다. ZFS 파일 시스템에서 `chmod` 명

령을 사용하여 서버와 클라이언트에서 ACL을 설정할 수 있습니다. UFS 파일 시스템에서는 `setfacl` 명령을 사용할 수 있습니다. 자세한 내용은 `chmod(1)` 및 `setfacl(1)` 매뉴얼 페이지를 참조하십시오. NFS 버전 4에서는 ID 매핑 `nfsmapid`를 사용하여 서버의 ACL 항목에 있는 사용자 ID 또는 그룹 ID를 클라이언트의 ACL 항목에 있는 사용자 ID 또는 그룹 ID로 매핑합니다. 그 반대의 경우도 마찬가지입니다. ACL 항목의 사용자 및 그룹 ID는 클라이언트와 서버에 모두 있어야 합니다.

ACL 및 `nfsmapid`에 대한 자세한 내용은 다음을 참조하십시오.

- “Oracle Solaris 11.2의 ZFS 파일 시스템 관리”의 7 장, “ACL 및 속성을 사용하여 Oracle Solaris ZFS 파일 보호”
- “NFS 데몬” [144]

ID 매핑 문제

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 `unknown`으로 표시됩니다.
예를 들어 이 상황에서 `ls -lv` 또는 `ls -lv` 명령을 실행하면 일부 ACL 항목의 그룹 또는 사용자가 `unknown`으로 표시됩니다.
- 클라이언트에 설정되어 있는 ACL 항목의 사용자 ID 또는 그룹 ID를 서버의 유효한 사용자 ID 또는 그룹 ID에 매핑할 수 없는 경우 `setfacl` 및 `chmod` 명령이 실패할 수 있으며 `Permission denied` 오류 메시지가 반환됩니다.
- 클라이언트와 서버의 `nfsmapid_domain` 값이 일치하지 않으면 ID 매핑에 실패합니다. 자세한 내용은 “NFS 데몬” [144]을 참조하십시오.

ID 매핑 문제를 방지하려면 다음을 수행합니다.

- `nfsmapid_domain`의 값이 올바르게 설정되어 있는지 확인합니다. 현재 선택된 NFSv4 도메인은 `/var/run/nfs4_domain` 파일에서 사용할 수 있습니다.
- ACL 항목의 모든 사용자 ID 및 그룹 ID가 NFS 버전 4 클라이언트와 서버에 모두 있는지 확인합니다.

매핑되지 않은 사용자 ID 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));
}
```

참고 - 이 스크립트에서 사용되는 프로브 이름은 이후 변경될 수 있는 인터페이스입니다. 자세한 내용은 [“Oracle Solaris 11.2 Dynamic Tracing Guide”](#)의 [“Stability Levels”](#)을 참조하십시오.

UDP 및 TCP 협상

NFS 버전 2 및 NFS 버전 3에서 전송 프로토콜에 대한 협상은 마운트될 때 발생합니다. 시작 중에는 전송 프로토콜도 협상합니다. 기본적으로 클라이언트와 서버에서 모두 지원되는 첫 번째 연결 지향 전송이 선택됩니다. 이러한 전송을 선택할 수 없으면 사용 가능한 첫 번째 비연결 프로토콜이 사용됩니다. 시스템에서 지원되는 전송 프로토콜은 `/etc/netconfig` 파일에 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` 명령의 NFS 관련 옵션에 대한 자세한 내용은 [`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 3 프로토콜에서는 전송 크기가 무제한입니다. 클라이언트가 마운트 시에 더 작은 전송 크기를 사용할 수 있지만, 대부분 경우에는 이렇게 할 필요가 없습니다.

NFS 버전 2 프로토콜을 사용하는 시스템과는 전송 크기를 협상하지 않습니다. 최대 전송 크기는 8KB로 설정됩니다.

`mount` 명령에서 `-rsize` 및 `-wsize` 옵션을 사용하여 전송 크기를 수동으로 설정할 수 있습니다. 일부 시스템 클라이언트의 경우에는 전송 크기를 줄여야 할 수 있습니다. NFS 서버가 더 큰 전송 크기를 사용하도록 구성되어 있는 경우에도 전송 크기를 늘릴 수 있습니다.

참고 - 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.

NFS 버전 3의 파일 시스템 마운트 방법

이 절의 정보는 NFS 버전 3 마운트에만 적용됩니다. NFS 버전 4 마운트 프로세스에는 포트 맵 서비스 또는 MOUNT 프로토콜이 포함되지 않습니다.

클라이언트는 서버에서 파일 시스템을 마운트하려고 할 때 서버에서 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/user를 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=33482
client -> server MOUNT3 C Null
server -> client MOUNT3 R Null
client -> server MOUNT3 C Mount /export/home9/user
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/user
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.

참고 - NFS 버전 4 프로토콜에서는 휘발성 파일 핸들을 지원합니다. 자세한 내용은 “NFS 버전 4의 휘발성 파일 핸들” [27]을 참조하십시오.

마운트 시 `-public` 옵션과 NFS URL의 효과

`-public` 옵션을 사용하면 마운트가 실패하는 상황이 발생할 수 있습니다. NFS URL을 추가하면 오류가 발생할 수도 있습니다. 다음 목록에서는 이러한 옵션을 사용하는 경우 파일 시스템을 마운트하는 방법을 설명합니다.

- **NFS URL의 `public` 옵션** - 공용 파일 핸들을 사용합니다. 공용 파일 핸들이 지원되지 않으면 마운트는 실패합니다.
- **정규 경로의 `public` 옵션** - 공용 파일 핸들을 사용합니다. 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

페일오버는 복제된 파일 시스템을 지원하는 서버 목록에서 서버를 선택하는 프로세스입니다. 일반적으로 정렬된 목록에서 다음 서버가 사용됩니다(응답하지 못하는 경우는 제외). NFS 클라이언트는 클라이언트측 페일오버를 사용하여 여러 서버가 동일한 데이터를 제공할 때 이를 감지하고 현재 서버를 사용할 수 없게 되면 대체 서버로 전환할 수 있습니다. 이 전환을 다시 매핑이라고 합니다. 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.

다음 상황의 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. 페일오버는 클라이언트에서 실행 중인 프로세스를 중단하지 않고 언제든지 수행될 수 있습니다.

페일오버를 수행하려면 파일 시스템이 읽기 전용으로 마운트되어야 합니다. 파일 시스템이 동일해야 페일오버가 정상적으로 수행됩니다. 동일한 파일 시스템에 대한 자세한 내용은 [“복제된 파일 시스템이란?” \[36\]](#)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. 최소 2개 서버가 있어야 합니다. 각 서버는 여러 서브넷을 지원합니다. 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.

복제된 파일 시스템이란?

파일 시스템의 크기가 같고 포함된 파일 크기나 파일 유형이 원본 파일 시스템과 같으면 클라이언트측 페일오버를 위해 파일 시스템을 복제로 사용할 수 있습니다. 이때 권한, 생성일 및 기타 파일 속성은 고려하지 않습니다. 파일 크기나 파일 유형이 다른 경우 다시 매핑이 실패하고 이전 서버를 사용할 수 있을 때까지 프로세스가 정지됩니다. NFS 버전 4에서는 동작이 다릅니다. 클라이언트측 페일오버에 대한 자세한 내용은 [“NFS 버전 4의 클라이언트측 페일오버” \[36\]](#)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 이름을 바꿉니다.
- 클라이언트 사용량이 낮은 야간에 업데이트를 실행합니다.
- 업데이트를 소규모로 유지합니다.
- 파일 복사본 수를 최소화합니다.

페일오버 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 감지할 수 없으므로 잠금은 다시 매핑 전체에서 지속됩니다. 파일은 변경되지 않아야 하므로 서버측에서는 파일을 잠글 필요가 없습니다.

NFS 버전 4의 클라이언트측 페일오버

NFS 버전 4에서는 파일 크기나 파일 유형이 달라 복제를 설정할 수 없으면 다음과 같은 현상이 발생합니다.

1. 파일이 사용 불가능으로 표시됩니다.
2. 경고가 표시됩니다.
3. 복제된 마운트의 파일을 사용하는 응용 프로그램에서 시스템 호출 오류를 받습니다.

참고 - 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.

NFS 서버 로깅의 작동 방식

참고 - NFS 버전 4에서는 서버 로깅이 지원되지 않습니다.

NFS 서버 로깅에서는 NFS 읽기 및 쓰기 레코드와 파일 시스템을 수정하는 작업을 제공합니다. 이 레코드를 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. nfslogd를 사용 안함으로 설정한 경우에는 파일 핸들-경로 테이블에서 매핑이 변경되지 않으므로 데몬을 계속 실행해야 합니다.

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. 공용 파일 핸들에는 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 공용 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 전달할 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. 평가를 서버 루트 파일 시스템에 상대적으로 변경하려면 경로 첫 부분에 슬래시를 더 추가합니다. 공용 파일 핸들이 /export/ftp 파일 시스템에서 설정된 경우 다음 두 NFS URL은 동일합니다.

```
nfs://server/junk
nfs://server//export/ftp/junk
```

참고 - 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 공용 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. 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.

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.
- NFS URL을 통해 다양한 파일 형식의 이러한 파일에 액세스하는 경우 파일 이름으로 파일 유형을 확인할 수 있으면 외부 브라우저가 시작됩니다. WebNFS 소프트웨어는 파일 유형을 확인하기 위해 파일 내부를 확인하지 않으므로 파일 이름 확장자를 통해서만 파일 유형을 확인할 수 있습니다. 브라우저는 표준 MIME 유형의 모든 파일 이름 확장자를 인식해야 합니다.
- NFS 검색에서는 서버측 이미지 맵을 사용할 수 없지만, URL이 위치를 사용하여 정의되므로 클라이언트측 이미지 맵을 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 슈퍼 유저로 로그인하고 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. 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 공격

의 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 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 공유하고 사용자는 로그인 암호를 통해 인증됩니다. 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 키가 저장되는 데이터베이스의 항목을 변경하지 않는 것으로 신뢰됩니다.

RPC 인증 시스템에는 자격 증명과 검증기가 사용됩니다. 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, GID 및 그룹 액세스 목록이 포함됩니다. 그러나 검증기가 없으므로 슈퍼 유저는 su와 같은 명령을 사용하여 적절한 자격 증명을 위조할 수 있습니다. 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.

참고 - Kerberos 인증 시스템에 대한 지원이 더 이상 보안 RPC의 일부분으로 제공되지 않지만, 서버측 및 클라이언트측 구현은 릴리스에 포함되어 있습니다. Kerberos 인증 구현에 대한 자세한 내용은 [“Oracle Solaris 11.2의 Kerberos 및 기타 인증 서비스 관리”의 2 장, “Kerberos 서비스 정보”](#)를 참조하십시오.

DH 인증

DH 인증은 DES(데이터 암호화 표준) 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. 맵 설정 방법에 대한 자세한 내용은 “Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”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. 두 에이전트가 현재 시간에 동의해야 하고 보낸 사람과 받는 사람이 같은 암호 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 키라고도 함)를 생성하고 공개 키 암호화를 통해 공통 키를 추론함으로써 동일한 암호 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.

NFS에서 보안 RPC 사용

보안 RPC를 사용하려는 경우 다음 사항에 유의하십시오.

- 정전 이후의 경우와 같이 시스템 관리자가 없는데 서버가 충돌하는 경우 시스템에 저장된 모든 보안 키가 삭제됩니다. 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(keyserv에서 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 강제 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 제

공합니다. 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 `root`에서 소유합니다. `root`의 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 `dave`가 소유하고 있으며 `dave`가 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 키는 해당 컴퓨터의 키 서버로 전달되며, 이 서버에 보안 키가 저장됩니다. 이 프로세스는 원격 컴퓨터를 신뢰하지 않는 경우에만 문제가 됩니다. 의심스러운 부분이 있는 경우 원격 컴퓨터에 암호가 필요하면 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 11 릴리스에는 미러 마운트라는 새로운 마운트 기능이 포함되어 있습니다. 미러 마운트를 사용하면 파일 시스템이 NFS 버전 4 서버에서 공유된 직후에 NFS 버전 4 클라이언트에서 파일 시스템의 파일에 액세스할 수 있습니다. 즉, `mount` 명령을 사용하거나 `autofs` 맵을 업데이트하는 오버헤드 없이도 파일에 액세스할 수 있습니다. 따라서 특정 NFS 버전 4 파일 시스템을 클라이언트에 마운트하고 나면 해당 서버의 다른 파일 시스템도 마운트할 수 있습니다.

일반적으로 미러 마운트 기능을 사용하면 NFS 버전 4 클라이언트에서 최적 상태로 사용할 수 있습니다. 단, 다음 경우는 예외입니다.

- 클라이언트에서 서버에 없는 다른 계층을 사용하는 경우
- 상위 파일 시스템과 다른 마운트 옵션을 사용하는 경우

미러 마운트를 사용하여 파일 시스템 마운트

NFS 버전 4 클라이언트에서 파일 시스템이 수동 마운트 또는 `autofs`를 사용하여 마운트된 경우 마운트된 파일 시스템에 더 추가하는 파일 시스템은 미러 마운트 기능을 사용하여 클라

이엔트에 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 NFS 버전 4 보안 협상이 수행되어 마운트 요청이 성공하도록 마운트 옵션을 조정합니다.

특정 서버 파일 시스템에 대한 자동 마운트 트리거가 있는 경우 자동 마운트 트리거가 미리 마운트보다 우선적으로 사용되므로 해당 파일 시스템에 대해 미리 마운트가 수행되지 않습니다. 이 경우 미리 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 마운트 사용 방법에 대한 자세한 내용은 [서버에서 모든 파일 시스템을 마운트하는 방법 \[68\]](#)을 참조하십시오.

미러 마운트를 사용하여 파일 시스템 마운트 해제

미러 마운트된 파일 시스템은 일정 기간 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. 미러 마운트된 파일 시스템이 활성 상태이면 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 11.1 릴리스에는 NFS 참조라는 새로운 NFS 기능이 포함되어 있습니다. NFS 참조를 사용하면 여러 NFS 버전 4 서버를 통합 이름 공간으로 연결하는 방법으로 NFS 버전 4 서버가 다른 NFS 버전 4 서버에 있는 파일 시스템을 가리킬 수 있습니다.

참조가 심볼릭 링크로 표시되기 때문에 NFS 버전 2, NFS 버전 3 및 기타 유형 클라이언트도 참조를 따를 수 있습니다.

NFS 참조를 사용하는 경우

NFS 참조는 여러 서버에 대해 단일 파일 이름 세트로 표시되는 항목을 만들지만 이 용도로 autofs는 사용하지 않으려는 경우에 유용합니다. 참조를 호스트하려면 NFS 버전 4 서버만 사용할 수 있으며 서버에서는 Oracle Solaris 11.1 릴리스 이상을 실행해야 합니다.

NFS 참조 만들기

`nfsref` 명령을 사용하여 NFS 참조를 만듭니다. 참조가 생성되었는데 마운트 지점이 아직 없으면 심볼릭 링크가 생성됩니다. 이 심볼릭 링크에는 객체를 구문 재분석 지점으로 식별하는 특수 플래그가 포함됩니다. 구문 재분석 지점은 특수 처리가 필요함을 명시하는 데 사용되는 특수 표시자입니다. 구문 재분석 지점이 이미 있으면 경우에 따라 NFS 서비스 데이터가 추가되거나 기존 NFS 서비스 데이터를 대체합니다.

NFS 참조 제거

`nfsref` 명령을 사용하여 NFS 참조를 제거합니다. 이 명령은 지정된 구문 재분석 지점에서 NFS 서비스 데이터를 제거합니다. 다른 서비스 데이터 유형이 없는 경우에도 구문 재분석 지점을 제거합니다.

autofs의 작동 방식

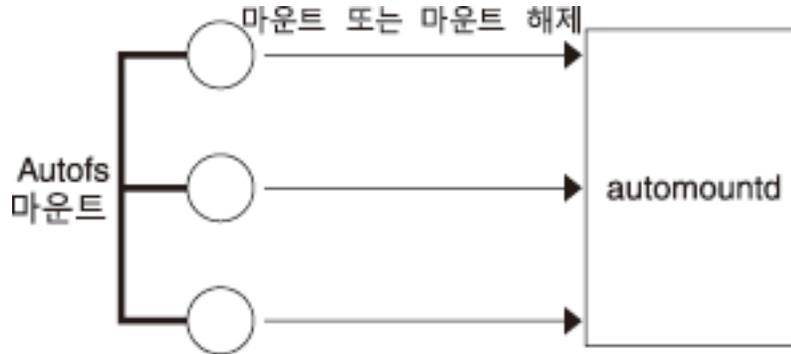
autofs는 자동 마운트 및 마운트 해제를 지원하는 커널 파일 시스템입니다. 자동 마운트를 위해 함께 작동하는 구성 요소는 다음과 같습니다.

- automount 명령
- autofs 파일 시스템
- automountd 데몬

autofs는 해당하는 파일 시스템을 자동으로 마운트하는 클라이언트측 서비스입니다. 시스템 시작 시 호출되는 자동 마운트 서비스인 `svc:/system/filesystem/autofs`는 마스터 맵 파일 `auto_master`를 읽어 초기 autofs 마운트 세트를 만듭니다. 이러한 autofs 마운트는 시작 시 자동으로 마운트되지 않지만 나중에 파일 시스템이 마운트되는 지점으로 사용됩니다. 이러한 지점은 트리거 노드라고도 합니다. 탐색 프로세스 시작에 대한 자세한 내용은 [“autofs에서 탐색 프로세스를 시작하는 방법\(마스터 맵\)” \[51\]](#)을 참조하십시오.

다음 그림에서는 autofs 서비스가 automount 명령을 시작하는 방식을 보여줍니다.

그림 2-3 svc:/system/filesystem/autofs 서비스가 automount 시작



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 마운트 지점에서 파일 시스템 액세스 요청을 하면 다음이 수행됩니다.

1. autofs는 해당 요청을 가로챍니다.
2. autofs는 마운트가 요청된 파일 시스템에 대한 메시지를 automountd 데몬으로 보냅니다.
3. automountd 데몬은 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.

처음에 autofs 마운트를 마운트한 후 automount 명령을 사용하여 필요에 따라 autofs 마운트를 업데이트합니다. 이 명령은 auto_master 맵의 마운트 목록을 마운트 테이블 파일 /etc/mnttab(이전의 /etc/mstab)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/vfstab파일(각 컴퓨터마다 다름)에서 마운트할 파일 시스템 목록을 읽지 않습니다. automount 명령은 도메인 내와 컴퓨터에서 이름 공간 또는 로컬 파일을 통해 제어됩니다.

참고 - 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.

autofs 맵

autofs에서는 세 가지 유형의 맵을 사용합니다.

- 마스터 맵
- 직접 맵
- 간접 맵

마스터 autofs 맵

auto_master 맵은 디렉토리를 맵과 연결합니다. 이 맵은 autofs에서 확인해야 하는 모든 맵을 지정하는 마스터 목록입니다. 다음 예에서는 auto_master 파일에 포함될 수 있는 정보 유형을 보여줍니다.

예 2-1 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]

mount-point 디렉토리의 전체(절대) 경로 이름입니다. 디렉토리가 없으면 autofs에서 가능한 경우 디렉토리를 만듭니다. 디렉토리가 있으며 비어 있지 않

은 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.

map-name

autofs에서 위치에 대한 방향이나 마운트 정보를 찾기 위해 사용하는 맵의 이름입니다. 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 “[마운트 지점 /net](#)” [47]을 참조하십시오.

mount-options

선택적인 심표로 구분된 옵션 목록입니다. 이 목록에 포함된 옵션은 *map-name*의 항목에 다른 옵션이 나열된 경우를 제외하면 *map-name*에 지정된 항목 마운트 시에 적용됩니다. 각각의 특정 파일 시스템 유형에 대한 옵션은 해당 파일 시스템의 mount 매뉴얼 페이지에 나열됩니다. NFS 관련 마운트 옵션에 대한 자세한 내용은 [mount_nfs\(1M\)](#) 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.

참고 - 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.

참고 - autofs는 모든 컴퓨터에서 실행되며 /net 및 /home(자동 마운트된 홈 디렉토리)을 기본적으로 지원합니다. /etc/auto_master 파일을 로컬로 편집하거나 NIS auto.master 맵에서 이러한 기본 항목을 대체할 수 있습니다.

마운트 지점 /net

autofs는 호스트 데이터베이스만 사용하는 특수 내장 맵 -hosts의 모든 항목을 디렉토리 /net 아래에 마운트합니다. system1 컴퓨터가 호스트 데이터베이스에 있고 해당 파일 시스템을 내보낸다고 가정하겠습니다. 다음 명령은 현재 디렉토리를 gumbo 컴퓨터의 루트 디렉토리로 변경합니다.

```
# cd /net/gumbo
```

autofs는 system1 호스트의 내보낸 파일 시스템만 마운트할 수 있습니다. 즉, 로컬 디스크의 파일 시스템이 아닌 네트워크 사용자에게 제공되는 서버의 파일 시스템만 마운트할 수 있습니다. 따라서 system1의 모든 파일과 디렉토리를 /net/system1을 통해 사용할 수는 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.

참고 - NFS 버전 3 및 이전 프로토콜을 사용하여 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. NFS 버전 4를 사용하는 시스템의 경우 미리 마운트는 서버의 내보낸 파일 시스템 목록에 동적으로 변경된 모든 내용을 반영합니다.

마운트 지점 /nfs4

/nfs4 마운트 지점은 의사 맵을 사용하여 FedFS 도메인 루트를 마운트합니다. /nfs4/example.net 파일을 참조하는 경우 DNS 도메인 example.net의 도메인 루트를 찾아 해당 위치에서 마운트하려고 시도합니다. /nfs4 아래에 경로를 마운트하려면 [“FedFS Server에 대한 DNS 레코드 설정” \[81\]](#)에 설명된 대로 DNS 서버에서 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. 다음 예에서는 일반적인 /etc/auto_direct 맵을 보여줍니다.

```
/usr/local      -ro \
  /bin           system1:/export/local/sun4 \
  /share        system1:/export/local/share \
  /src          system1:/export/local/src
/usr/man        -ro system2:/usr/man \
               system3:/usr/man \
               system4:/usr/man
/usr/games      -ro system5:/usr/games
/usr/spool/news -ro system6:/usr/spool/news \
               system4:/var/spool/news
```

직접 맵의 행 구문은 다음과 같습니다.

```
key [ mount-options ] location
```

<i>key</i>	직접 맵에 있는 마운트 지점의 경로 이름입니다.
<i>mount-options</i>	이 특정 마운트에 적용할 옵션입니다. 이러한 옵션은 맵 기본값과 다른 경우에만 필요합니다. 각각의 특정 파일 시스템 유형에 대한 옵션은 해당 파일 시스템의 mount 매뉴얼 페이지에 나열됩니다. NFS 관련 마운트 옵션에 대한 자세한 내용은 <code>mount_nfs(1M)</code> 매뉴얼 페이지를 참조하십시오.
<i>location</i>	파일 시스템 위치입니다. NFS 파일 시스템에 대해 하나 이상의 파일 시스템이 <code>server:pathname</code> 으로 지정됩니다.

참고 - 경로 이름에는 자동 마운트된 마운트 지점을 포함할 수 없습니다. 경로 이름은 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
```

참고 - 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.

직접 autofs 맵의 기능에 대한 자세한 내용은 [“autofs에서 클라이언트에 대해 가장 가까운 읽기 전용 파일을 선택하는 방법\(여러 위치\)” \[53\]](#)을 참조하십시오.

마운트 지점 /-

예 2-1. “샘플 `/etc/auto_master` 파일”에서 /-는 `auto_direct`의 항목을 특정 마운트 지점과 연관시키지 않도록 autofs에 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.

NIS `auto_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*

<i>key</i>	간접 맵에서 슬래시가 없는 이름입니다.
<i>mount-options</i>	이 특정 마운트에 적용할 옵션입니다. 이러한 옵션은 맵 기본값과 다른 경우에만 필요합니다. 각각의 특정 파일 시스템 유형에 대한 옵션은 해당 파일 시스템의 mount 매뉴얼 페이지에 나열됩니다. 예를 들어 NFS 관련 마운트 옵션은 mount_nfs(1M) 매뉴얼 페이지를 참조하십시오.
<i>location</i>	파일 시스템 위치입니다. 하나 이상의 파일 시스템이 <i>server:pathname</i> 으로 지정됩니다.

참고 - 경로 이름에는 자동 마운트된 마운트 지점을 포함할 수 없습니다. 경로 이름은 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 2-1. “[샘플 /etc/auto_master 파일](#)”에는 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.

```
user1      server1:/export/home/user1
user2      server2:/export/home/user2
user3      server3:/export/home/user3
user4      server4:/export/home/user4
user5      server5:/export/home/user5
user6      server6:/export/home/user6
user7      -rw,nosuid  server7:/export/home/user7
```

예를 들어 이전 맵이 master-server 호스트에 있다고 가정해 보겠습니다. user7 사용자의 암호 데이터베이스에 홈 디렉토리를 /home/user7로 지정하는 항목이 있다고 가정하겠습니다. user7이 master-server 컴퓨터에 로그인할 때마다 autofs는 server7 컴퓨터에 있는 /export/home/user7 디렉토리를 마운트합니다. 홈 디렉토리는 읽기/쓰기(nosuid)로 마운트됩니다.

다음으로 사용자 user7의 홈 디렉토리가 암호 데이터베이스에 /home/user7로 포함되어 있다고 가정해 보겠습니다. user7을 비롯한 모든 사용자는 auto_home 맵을 참조하는 마스터 맵을 사용하여 설정된 컴퓨터에서 이 경로에 액세스할 수 있습니다.

이와 같은 조건 하에서 사용자 user7은 이러한 모든 컴퓨터에서 login 또는 rlogin을 실행할 수 있으며 홈 디렉토리를 대신 마운트하도록 할 수 있습니다.

또한 이제 user7은 다음 명령을 입력할 수도 있습니다.

```
# cd ~user1
```

그러면 autofs는 user1의 홈 디렉토리를 user7 대신 마운트합니다(모든 권한이 허용하는 경우).

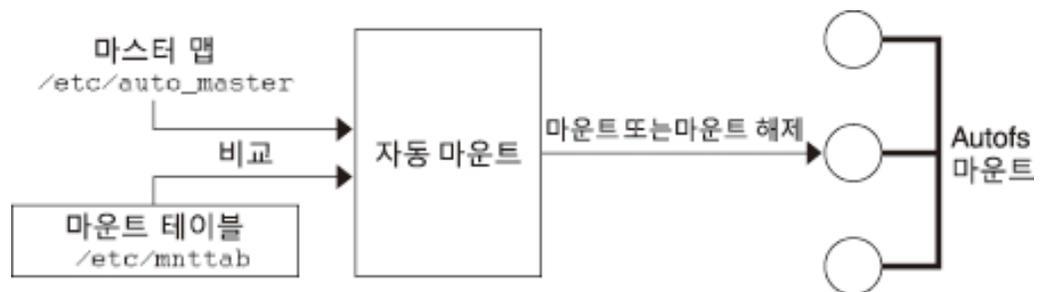
참고 - 자동 마운트 맵 간에 옵션 연결은 수행되지 않습니다. 자동 마운트 맵에 추가되는 모든 옵션은 이전에 검색한 맵에 나열된 모든 옵션을 대체합니다. 예를 들어 auto_master 맵에 포함된 옵션은 다른 맵의 해당하는 항목으로 대체됩니다.

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.

autofs에서 탐색 프로세스를 시작하는 방법(마스터 맵)

automount 명령은 시스템 시작 시 마스터 맵을 읽습니다. 마스터 맵의 각 항목은 직접 맵 이름이나 간접 맵 이름, 경로 및 마운트 옵션입니다. 항목의 특정 순서는 중요하지 않습니다.

그림 2-4 마스터 맵을 통한 탐색



이 그림에서는 automount는 마스터 맵의 항목을 마운트 테이블의 항목과 비교하여 현재 목록을 생성함을 보여줍니다.

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 같이 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. 자세한 내용은 “[autofs 마운트 해제](#)” [53]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 것입니다. 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)

이 절에서는 다음 직접 맵 예를 사용하여 autofs가 클라이언트에 대한 가장 가까운 읽기 전용 파일을 선택하는 방법을 설명합니다.

```
/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 포함). 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. 사용자가 작업을 수행하지 않아도 사용 가능한 최적의 서버가 자동으로 사용된다는 이점도 있습니다.

파일 시스템이 복제(“복제된 파일 시스템이란?” [36] 참조)로 구성된 경우 클라이언트에서는 페일오버를 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. 직접 맵 예에서는 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

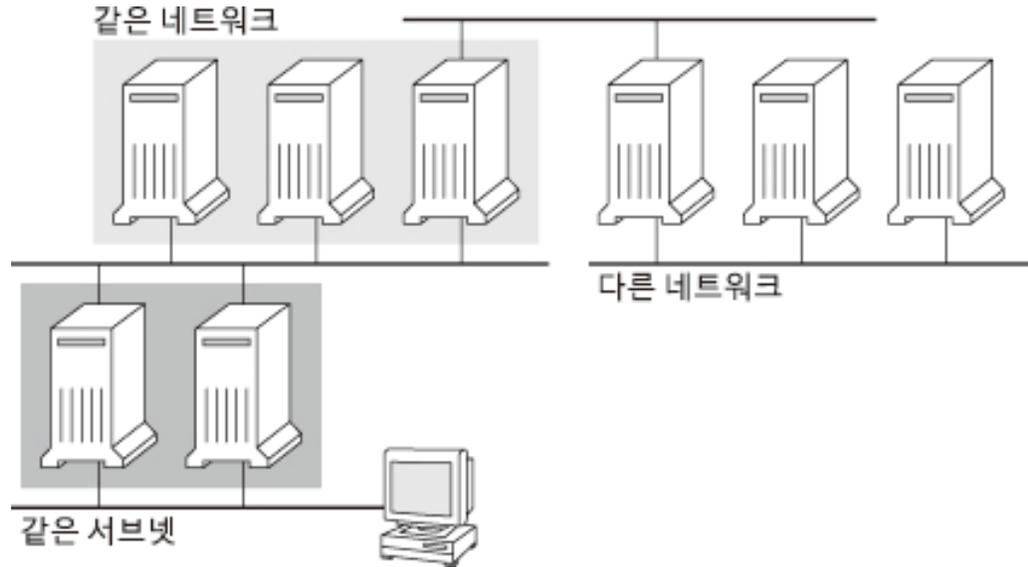
정렬 프로세스 중에는 각 NFS 프로토콜 버전을 지원하는 서버 수를 계산합니다. 가장 많은 서버에서 지원되는 프로토콜 버전이 기본적으로 사용되는 프로토콜이 됩니다. 이러한 선택 방법으로 인해 클라이언트는 가장 많은 수의 서버를 사용할 수 있습니다.

프로토콜 버전이 같은 최대 서버 하위 세트를 찾으면 해당 서버 목록이 인접도에 따라 정렬됩니다. 인접도를 확인하려면 IPv4 주소를 검사하여 각 서브넷에 있는 서버를 확인합니다. 로컬 서브넷의 서버가 원격 서브넷의 서버보다 우선적으로 사용됩니다. 가장 가까운 서버가 우선적으로 사용되므로 대기 시간과 네트워크 트래픽이 줄어듭니다.

참고 - IPv6 주소를 사용하는 복제본의 경우에는 인접도를 확인할 수 없습니다.

그림 2-5. “서버 인접도”에서는 서버 인접도를 보여줍니다.

그림 2-5 서버 인접도



로컬 서브넷에 같은 프로토콜을 지원하는 서버가 여러 대 있는 경우 각 서버에 연결하는 시간을 확인하여 연결 시간이 가장 빠른 서버가 사용됩니다. 가중치를 사용하면 정렬에 영향을 미칠 수도 있습니다. 가중치에 대한 자세한 내용은 [“autofs 및 가중치” \[56\]](#)를 참조하십시오.

예를 들어 NFS 버전 4 서버의 수가 로컬 서브넷에서 가장 많으면 NFS 버전 4가 기본적으로 사용되는 프로토콜이 됩니다. 그러나 로컬 서브넷의 서버가 여러 프로토콜을 지원하면 정렬 프로세스가 더욱 복잡합니다. 아래에 정렬 프로세스의 작동 방식을 보여주는 몇 가지 예제가 나와 있습니다.

- 로컬 서브넷의 서버가 원격 서브넷의 서버보다 우선적으로 사용됩니다. 따라서 NFS 버전 3 서버가 로컬 서브넷에 있고 가장 가까운 NFS 버전 4 서버가 원격 서브넷에 있으면 NFS 버전 3 서버가 우선적으로 사용됩니다. 마찬가지로 로컬 서브넷이 NFS 버전 2 서버로 구성된 경우 NFS 버전 3 및 NFS 버전 4 서버로 구성된 원격 서브넷보다 우선적으로 사용됩니다.
- 로컬 서브넷이 각각 다른 수의 NFS 버전 2, NFS 버전 3, NFS 버전 4 서버로 구성된 경우에는 추가적인 정렬이 필요합니다. 자동 마운트는 로컬 서브넷에서 가장 높은 버전을 우선적으로 사용합니다. 이 경우에는 NFS 버전 4가 가장 높은 버전입니다. 그러나 로컬 서브넷의 NFS 버전 3 또는 NFS 버전 2 서버 수가 NFS 버전 4 서버의 수보다 많은 경우 자동 마운트는 로컬 서브넷의 가장 높은 버전에서 한 버전 낮은 서버를 사용합니다. 예를 들어 로컬 서브넷에 NFS 버전 4 서버 3대, NFS 버전 3 서버 3대, NFS 버전 2 서버 10대가 있는 경우에는 NFS 버전 3 서버가 선택됩니다.

- 마찬가지로 로컬 서브넷이 각각 다른 수의 NFS 버전 2 및 NFS 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 NFS 버전 3 서버보다 NFS 버전 2 서버의 수가 더 많으면 NFS 버전 2 서버가 선택됩니다.

참고 - 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 수 있습니다. 이러한 매개변수에 대한 자세한 내용은 “[NFS 데몬](#)” [144]을 참조하십시오.

파일오버를 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 파일오버가 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.

참고 - 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.

참고 - 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 아키텍처 유형을 포함할 수 있습니다. 중괄호를 사용하여 변수 이름과 추가된 문자/숫자를 구분할 수도 있습니다. 다음 표에서는 미리 정의된 맵 변수를 보여줍니다.

표 2-1 미리 정의된 맵 변수

변수	의미	파생 원본	예
ARCH	아키텍처 유형	uname -m	sun4
CPU	프로세서 유형	uname -p	sparc
HOST	호스트 이름	uname -n	system1
OSNAME	운영 체제 이름	uname -s	SunOS
OSREL	운영 체제 릴리스	uname -r	5.10
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 아키텍처의 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 아키텍처에 적용됩니다.

참고 - sun4 구조용으로 작성된 모든 응용 프로그램은 모든 sun4 플랫폼에서 실행할 수 있습니다. -ARCH 변수는 sun4로 하드 코드됩니다.

다른 맵을 참조하는 맵

파일 맵의 맵 항목에서 맵 이름과 함께 사용되는 특수 문자는 맵 이름 처리 방법에 영향을 미칩니다.

- 파일 맵에서 맵 항목 `+mapname`을 사용하면 automount가 지정된 맵을 현재 파일에 포함된 것처럼 읽습니다.
- `mapname` 앞에 슬래시가 없으면 autofs는 맵 이름을 문자열로 처리하며 이름 서비스 스 위치 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 데

이터베이스를 검색하는 순서를 지정합니다. 특정 config/automount 등록 정보가 지정되어 있지 않으면 config/default 등록 정보에 정의된 순서가 사용됩니다.

예 2-2 automount 명령을 통한 맵 검색 순서 표시

```
# 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 예에서는 로컬 파일의 맵이 NIS 맵보다 먼저 검색된다는 것을 보여줍니다. 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 풀백할 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
```

참고 - 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 맵이 유용합니다. 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
```

다음 예에서는 샘플 실행 가능 맵을 보여줍니다.

```
#!/bin/ksh
#
# executable map for autofs
#

case $1 in
    src) echo '-nosuid,hard bee:/export1' ;;
esac
```

이 예가 작동하려면 파일을 /etc/auto_execute로 설치해야 하며 파일에 실행 가능한 비트 세트가 있어야 합니다. 권한을 744로 설정합니다. 이러한 상황에서 다음 명령을 실행하면 bee의 /export1 파일 시스템이 마운트됩니다.

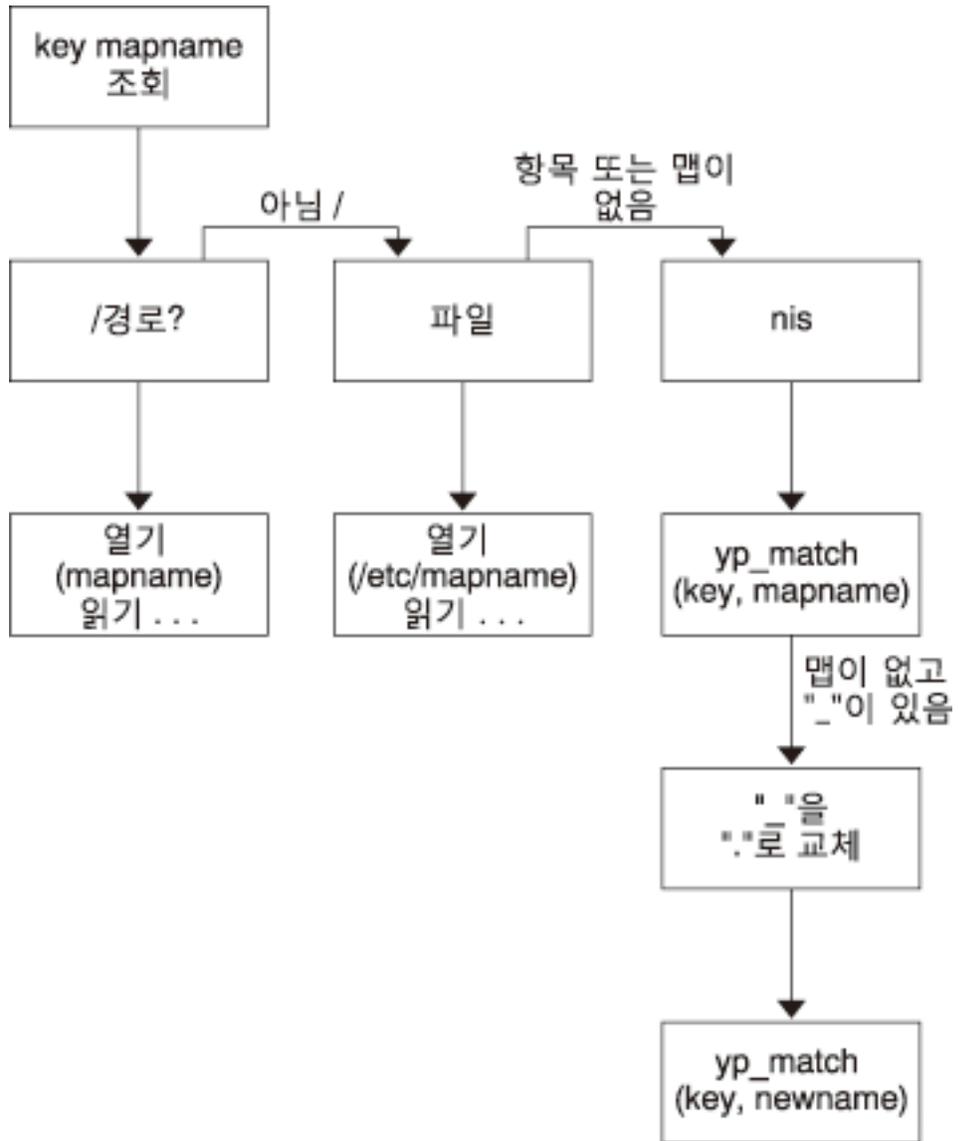
```
# ls /execute/src
```

이름 서비스에 대한 기본 autofs 동작

부트 시에는 autofs가 svc:/system/filesystem/autofs 서비스에 의해 호출되어 마스터 auto_master 맵을 확인합니다.

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 사용할 수 있는 맵은 찾지 못하고 밑줄이 하나 이상 포함된 맵 이름을 찾으면 밑줄이 점으로 변경되므로 기존 NIS 파일 이름이 작동할 수 있습니다. 그러면 autofs는 다음 그림에 나와 있는 것처럼 맵을 다시 확인합니다.

그림 2-6 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 절에서는 고급 autofs 기능 및 항목에 대해 설명합니다.

autofs 및 메타 문자

autofs는 일부 문자를 특별한 의미가 있는 것으로 인식합니다. 예를 들어 대체에 사용되는 문자도 있고, autofs 맵 구문 분석기로부터 다른 문자를 보호하는 데 사용되는 문자도 있습니다.

앰퍼센드(&)

다음 예와 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 다음 텍스트로 변경됩니다.

```
john      willow:/home/&
mary      willow:/home/&
joe       willow:/home/&
able      pine:/export/&
baker     peach:/export/&
```

직접 맵에서도 다음 예와 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 다음 항목을 포함할 수 없습니다.

```
/progs    &1,&2,&3:/export/src/progs
```

autofs에서는 이 예를 다음과 같이 해석합니다.

```
/progs    /progs1,/progs2,/progs3:/export/src/progs
```

별표(*)

범용 대체 문자인 별표(*)를 사용하여 모든 키와 일치하도록 지정할 수 있습니다. 예를 들어 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 포함되어 있으면 autofs 맵 구문 분석기가 제대로 처리할 수 없는 이름을 가진 디렉토리를 마운트해야 할 수 있습니다. autofs 구문 분석기는 콜론, 심표, 공백 등이 포함된 이름에 민감합니다. 이러한 이름은 다음 예와 같이 큰따옴표로 묶어야 합니다.

```
/vms    -ro    vmserver: - - - "rc0:dk1 - "  
/mac    -ro    gator:/ - "Mr Disk - "
```

◆◆◆ 3 장 3

네트워크 파일 시스템 관리

이 장에서는 NFS 서비스 설정, 공유할 새 파일 시스템 추가, 파일 시스템 마운트 등의 NFS 관리 작업을 수행하는 방법에 대한 정보를 제공합니다. 이 장에는 NFS 및 FedFS 참조를 구성하고 유지 관리하는 데 사용되는 절차도 포함되어 있습니다.

이 장의 내용:

- “자동 파일 시스템 공유” [64]
- “파일 시스템 마운트” [66]
- “NFS 서비스 설정” [72]
- “보안 NFS 시스템 관리” [75]
- “WebNFS 관리” [77]
- “NFS 참조 관리” [80]
- “FedFS 관리” [81]

참고 - 시스템에서 영역이 사용으로 설정된 경우 비전역 영역에서 이 기능을 사용하려면 “Oracle Solaris 영역 소개”를 참조하십시오.

네트워크 파일 시스템 관리 정보

NFS 관리자의 작업은 사이트 요구 사항 및 네트워크 관리자 역할에 따라 달라집니다. 로컬 네트워크에서 모든 시스템을 담당하고 있다면 다음을 결정해야 할 수 있습니다.

- 전용 서버로 사용할 수 있는 시스템
- 서버 및 클라이언트로 둘 다 사용할 수 있는 시스템
- 클라이언트로만 사용할 시스템

서버를 설정한 후에 유지 관리할 때는 다음과 같은 작업을 수행합니다.

- 필요에 따라 파일 시스템 공유 및 공유 해제
- 관리 파일을 수정하여 시스템이 자동으로 마운트하는 파일 시스템 목록 업데이트
- 네트워크 상태 확인

- NFS 관련 문제 발생 시 진단 및 해결
- autof 맵 설정

한 시스템이 서버 및 클라이언트로 둘 다 사용될 수 있습니다. 따라서 시스템을 사용하여 로컬 파일 시스템을 원격 시스템과 공유하고 원격 파일 시스템을 마운트할 수 있습니다.

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. 이 파일은 더는 사용되지 않습니다.

파일 시스템 공유(작업 맵)

다음 작업 맵에서는 NFS 서비스를 사용한 파일 시스템 공유에 대해 설명합니다.

표 3-1 파일 시스템 공유(작업 맵)

작업	설명	지침
자동 파일 시스템 공유 설정	서버를 재부트할 때 파일 시스템이 자동으로 공유되도록 서버를 구성합니다.	자동 파일 시스템 공유를 설정하는 방법 [64]
NFS 서버 로깅 사용으로 설정	선택한 파일 시스템에서 NFS 로깅이 실행되도록 서버를 구성합니다.	NFS 서버 로깅을 사용으로 설정하는 방법 [65]

▼ 자동 파일 시스템 공유를 설정하는 방법

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”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
```

사용 가능한 명령 옵션에 대한 자세한 내용은 [share_nfs\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

3. 지정한 옵션이 나열되어 있는지 확인합니다.

```
# share -F nfs
```

예를 들면 다음과 같습니다.

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

다음 순서 다음 단계에서는 클라이언트가 서버에서 공유하는 파일 시스템에 액세스할 수 있도록 autofs 맵을 설정합니다. autofs 맵 설정에 대한 자세한 내용은 표 4-1. “autofs 관리 작업”을 참조하십시오.

▼ NFS 서버 로깅을 사용하여 설정하는 방법

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. (옵션) 파일 시스템 구성 값을 변경합니다.

다음 중 한 방법으로 /etc/nfs/nfslog.conf 파일의 설정을 변경할 수 있습니다.

- global 태그와 연결된 데이터를 변경하여 모든 파일 시스템에 대한 기본 설정을 편집합니다.
- 파일 시스템에 대한 새 태그를 추가합니다.

/etc/nfs/nfslog.conf 파일 형식에 대한 자세한 내용은 [nfslog.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오.

3. NFS 서버 로깅을 사용할 파일 시스템을 정의합니다.

share 명령을 사용하여 각 파일 시스템을 정의합니다. log=tag 옵션과 함께 사용되는 태그를 /etc/nfs/nfslog.conf 파일에 지정해야 합니다.

다음 예에서는 global 태그의 기본 설정을 사용합니다.

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

4. 지정한 옵션이 나열되어 있는지 확인합니다.

예를 들면 다음과 같습니다.

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

5. NFS 로그 데몬 `nfslogd`가 실행 중인지 확인합니다.

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

6. `nfslogd` 데몬의 상태를 확인합니다.

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

파일 시스템 마운트

시스템 부트 시, 명령줄에서 요구 시 또는 자동 마운트를 통해 파일 시스템을 자동으로 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 마운트(작업 맵)

다음 표에는 파일 시스템 마운트와 관련된 작업이 나와 있습니다.

표 3-2 파일 시스템 마운트(작업 맵)

작업	설명	지침
부트 시 파일 시스템 마운트	시스템을 재부트할 때마다 파일 시스템을 마운트할 수 있게 합니다.	부트 시 파일 시스템을 마운트하는 방법 [67]
명령을 사용하여 파일 시스템 마운트	시스템이 실행 중일 때 파일 시스템을 마운트합니다. 이 절차는 테스트 시에 유용합니다.	명령줄에서 파일 시스템을 마운트하는 방법 [67]
자동 마운트를 사용하여 파일 시스템 마운트	명령줄을 사용하지 않고 요구 시 파일 시스템에 액세스할 수 있게 합니다.	"자동 마운트를 사용한 마운트" [68]
미러 마운트를 통해 모든 파일 시스템 마운트	한 서버에서 모든 파일 시스템을 마운트합니다.	서버에서 모든 파일 시스템을 마운트하는 방법 [68]
클라이언트측 페일오버 시작	서버가 실패하는 경우 작업 파일 시스템으로 자동으로 페일오버할 수 있게 합니다.	클라이언트측 페일오버를 사용하는 방법 [69]
클라이언트에 대해 마운트 액세스 사용 안함으로 설정	단일 클라이언트가 원격 파일 시스템에 액세스하는 기능을 사용 안함으로 설정합니다.	단일 클라이언트에 대한 마운트 액세스를 사용 안함으로 설정하는 방법 [69]
방화벽을 통해 파일 시스템 액세스 제공	WebNFS 프로토콜을 사용하여 방화벽을 통해 파일 시스템에 액세스할 수 있게 합니다.	방화벽을 통해 NFS 파일 시스템을 마운트하는 방법 [70]
NFS URL을 사용하여 파일 시스템 마운트	NFS URL을 사용하여 파일 시스템에 액세스할 수 있게 합니다. 이 프로세스를 통해 MOUNT 프로토콜을	NFS URL을 사용하여 NFS 파일 시스템을 마운트하는 방법 [71]

작업	설명	지침
	사용하지 않고 파일 시스템 액세스를 제공할 수 있습니다.	

▼ 부트 시 파일 시스템을 마운트하는 방법

이 절차에서는 autofs 맵을 사용하지 않고 부트 시 파일 시스템을 마운트하는 방법을 보여줍니다. 원격 파일 시스템에 액세스해야 하는 모든 클라이언트에서 이 절차를 완료해야 합니다.

1. 관리자가 됩니다.

자세한 내용은 [“Oracle Solaris 11.2의 사용자 및 프로세스 보안”](#)의 [“지정된 관리 권한 사용”](#)을 참조하십시오.

2. 파일 시스템에 대한 항목을 /etc/vfstab 파일에 추가합니다.

/etc/vfstab 파일에 포함된 항목의 구문은 다음과 같습니다.

```
#device      device      mount      FS      fsck      mount      mount
#to mount    to fsck     point     type    pass     at boot   options
```

/etc/vfstab 파일 항목에 대한 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 [mount\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

3. NFS 클라이언트 서비스를 사용으로 설정합니다.

```
# svcadm enable network/nfs/client
```

예 3-1 클라이언트 /etc/vfstab 파일의 항목

클라이언트 시스템이 wasp 서버에서 /var/mail 디렉토리를 마운트하려고 한다고 가정합니다. 클라이언트에서 파일 시스템을 /var/mail로 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. umount 명령을 사용하거나 로컬 시스템을 재부트하여 파일 시스템의 마운트를 해제할 수 있습니다.

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. 파일 시스템을 마운트합니다.

```
mount -F nfs -o specific-options resource mount-point
```

예를 들면 다음과 같습니다.

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

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

이 예에서 서버 bee의 /export/share/local 파일 시스템은 로컬 시스템의 /mnt에 읽기 전용으로 마운트됩니다.



주의 - mount 명령은 잘못된 옵션에 대한 경고를 제공하지 않습니다. 명령은 해석할 수 없는 옵션을 자동으로 무시합니다. 예기치 않은 동작을 방지하려면 사용하는 옵션을 모두 확인하십시오.

자동 마운트를 사용한 마운트

일반 시스템을 변경하지 않고 클라이언트에서 /net 마운트 지점을 통해 원격 파일 시스템에 액세스할 수 있습니다. 자동 마운트를 사용한 마운트 설정 및 지원에 대한 자세한 내용은 [표 4-1. “autofs 관리 작업”](#)을 참조하십시오. 다음 명령을 입력하여 /export/share/local 파일 시스템을 마운트합니다.

```
# 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 시스템을 자동으로 마운트 해제하므로 더 이상 액세스할 필요가 없게 된 후 파일 시스템을 수동으로 마운트 해제하지 않아도 됩니다.

▼ 서버에서 모든 파일 시스템을 마운트하는 방법

자동 미리 마운트 기능을 사용하면 특정 서버에서 1회의 마운트가 성공한 이후 클라이언트가 해당 서버에서 NFS를 사용하여 공유되는 모든 사용 가능 파일 시스템에 액세스할 수 있습니다. 미리 마운트는 자동으로 수행되므로 파일 시스템에 액세스하면 됩니다. 자세한 내용은 [“미리 마운트의 작동 방식” \[42\]](#)을 참조하십시오.

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. 서버의 내보낸 이름 공간 루트를 마운트합니다.
이 명령은 클라이언트에서 서버의 파일 시스템 계층을 미리링합니다. 이 예에서는 /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
```

▼ 클라이언트측 페일오버를 사용하는 방법

1. 관리자가 됩니다.
자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.
2. NFS 클라이언트에서 ro 옵션을 사용하여 파일 시스템을 마운트합니다.
명령줄에서, 자동 마운트를 통해 또는 다음과 같은 항목을 /etc/vfstab 파일에 추가하여 마운트할 수 있습니다.

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

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

▼ 단일 클라이언트에 대한 마운트 액세스를 사용 안함으로 설정하는 방법

1. 관리자가 됩니다.
자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.
2. 단일 클라이언트에 대한 마운트 액세스를 사용 안함으로 설정합니다.

```
# share -F nfs -o specific_options pathname
```

예를 들면 다음과 같습니다.

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

ro=-rose:eng rose라는 호스트를 제외하고는 eng 네트워크 그룹에 있는 모든 클라이언트에 대한 읽기 전용 마운트 액세스를 허용하는 액세스 목록입니다.

/export/share/ 공유할 파일 시스템
man

▼ 방화벽을 통해 NFS 파일 시스템을 마운트하는 방법

시작하기 전에 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.

1. 관리자가 됩니다.
자세한 내용은 [“Oracle Solaris 11.2의 사용자 및 프로세스 보안”](#)의 [“지정된 관리 권한 사용”](#)을 참조하십시오.
2. 다음과 같은 명령을 사용하여 파일 시스템을 수동으로 마운트합니다.

```
# mount -F nfs host:pathname mount-point
```

예를 들면 다음과 같습니다.

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

이 예에서는 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.

NFS URL을 사용하여 NFS 파일 시스템 마운트

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.

참고 - 파일 시스템을 마운트할 때 사용되는 NFS 프로토콜 버전은 클라이언트와 서버에서 모두 지원되는 최상위 버전입니다. 그러나 `vers=#` 옵션을 사용하여 특정 NFS 프로토콜 버전을 선택할 수 있습니다.

▼ NFS URL을 사용하여 NFS 파일 시스템을 마운트하는 방법

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. NFS URL을 사용하여 파일 시스템을 수동으로 마운트합니다.

```
# mount -F nfs nfs://host[:port]/pathname mount-point
```

예 3-2 NFS URL을 사용하여 NFS 파일 시스템 마운트

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

이 예에서는 NFS 포트 번호 3000을 사용하여 bee 서버에서 /export/share/local 파일 시스템을 마운트합니다. 포트 번호는 필요하지 않으며 기본적으로 표준 NFS 포트 번호 2049가 사용됩니다.

마운트할 수 있는 파일 시스템에 대한 정보 표시

showmount 명령을 실행하면 원격으로 마운트되었거나 마운트할 수 있는 파일 시스템에 대한 정보가 표시됩니다. -e 옵션을 사용하여 공유 파일 시스템을 나열합니다. 예를 들면 다음과 같습니다.

```
# /usr/sbin/showmount -e bee
export list for bee:
/export/share/local (everyone)
/export/home tulip,lilac
/export/home2 rose
```

기타 옵션에 대한 자세한 내용은 [showmount\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

일부 환경에서는 공유 파일 시스템 및 이 파일 시스템을 마운트한 시스템에 대한 정보가 표시되지 않아야 합니다. sharectl 명령의 showmount_info 등록 정보를 none으로 설정하여 클라이언트에서 다음 파일 시스템 정보가 볼 수 없는지 확인할 수 있습니다.

- 클라이언트가 액세스할 수 없는 파일 시스템 정보
- 모든 공유 파일 시스템에 대한 정보
- 파일 시스템을 마운트한 다른 시스템에 대한 정보

예 3-3 클라이언트에 표시되는 파일 시스템 정보 제한

```
bee# sharectl set -p showmount_info=none nfs
```

다음 정보가 클라이언트 rose에 표시됩니다.

```
# /usr/sbin/showmount -e bee
export list for bee:
/export/share/local  (everyone)
/export/home2       rose
```

/export/home파일 시스템에 대한 정보가 더는 표시되지 않습니다.

NFS 서비스 설정

이 절에서는 NFS 서비스를 설정하는 데 필요한 몇 가지 작업에 대해 설명합니다.

참고 - NFS 버전 4는 Oracle Solaris 11.2에서 지원되는 기본 NFS 버전입니다.

표 3-3 NFS 서비스 설정

작업	설명	지침
NFS 서버 시작 및 중지	NFS 서비스가 자동으로 시작되지 않은 경우 시작합니다. NFS 서비스를 중지합니다. 일반적으로는 서비스를 중지할 필요가 없습니다.	“NFS 서비스 시작 및 중지” [72]
자동 마운트 시작 및 중지	자동 마운트를 시작 및 중지합니다. 자동 마운트 맵 중 일부가 변경된 경우 이 절차를 수행해야 합니다.	“자동 마운트 시작 및 중지” [72]
다른 NFS 버전 선택	서버 및 클라이언트에서 NFS 4 버전 이외의 NFS 버전을 선택합니다.	“다른 NFS 버전 선택” [73]

NFS 서비스 시작 및 중지

관리자로서 `svcadm` 명령을 사용하여 서버에서 NFS 서비스를 사용 및 사용 안함으로 설정합니다.

- 서버에서 NFS 서비스를 사용으로 설정하려면 다음을 수행합니다.

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

- 서버에서 NFS 서비스를 사용 안함으로 설정하려면 다음을 수행합니다.

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

자동 마운트 시작 및 중지

관리자로서 `svcadm` 명령을 사용하여 `autofs` 데몬을 사용 및 사용 안함으로 설정합니다.

- `autofs` 데몬을 사용으로 설정하려면 다음을 수행합니다.

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

- autofs 데몬을 사용 안함으로 설정하려면 다음을 수행합니다.

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

다른 NFS 버전 선택

NFS 버전 4 이외의 NFS 버전을 사용하려면 다른 버전을 선택할 수 있습니다.

- 서버에서 다른 NFS 버전을 선택하려면 [서버에서 다른 NFS 버전을 선택하는 방법 \[73\]](#)을 참조하십시오.
- 클라이언트에서 다른 NFS 버전을 선택하려면 [클라이언트에서 다른 NFS 버전을 선택하는 방법 \[74\]](#)을 참조하십시오.
- 명령줄을 사용하여 클라이언트에서 다른 NFS 버전을 선택하려면 [mount 명령을 사용하여 클라이언트에서 다른 NFS 버전을 선택하는 방법 \[75\]](#)을 참조하십시오.

▼ 서버에서 다른 NFS 버전을 선택하는 방법

기본적으로 설정되는 NFS 버전 4를 사용하지 않게 선택하면 다른 NFS 버전을 선택할 수 있습니다.

1. 관리자가 됩니다.

자세한 내용은 “[Oracle Solaris 11.2의 사용자 및 프로세스 보안](#)”의 “[지정된 관리 권한 사용](#)”을 참조하십시오.

2. SMF 매개변수를 변경하여 NFS 버전 번호를 설정합니다.

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

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

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

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

NFS 서버 위임을 사용하면 NFS 클라이언트에서는 다른 NFS 클라이언트가 같은 파일에 액세스해야 할 때까지 파일을 캐시할 수 있습니다. NFS 버전 4에서는 서버 위임은 기본적으로 사용으로 설정됩니다. 자세한 내용은 “[NFS 버전 4의 위임](#)” [30]을 참조하십시오.

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

클라이언트와 서버의 공통 도메인을 설정하여 클라이언트와 서버 간에 사용자 ID 또는 그룹 ID 매핑을 사용으로 설정합니다.

```
# sharectl set -p nfsmapid_domain=my.example.com nfs
```

여기서 *my.example.com*은 공통 도메인 이름을 제공합니다.

nfsmapid 데몬에 대한 자세한 내용은 “[NFS 데몬](#)” [144]을 참조하십시오.

5. NFS 서비스가 서버에서 실행 중인지 확인합니다.

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

6. 필요한 경우 NFS 서비스를 사용으로 설정합니다.

NFS 서비스가 오프라인 상태이면 다음 명령을 입력하여 서비스를 사용으로 설정합니다.

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

NFS 서비스 구성에 대한 자세한 내용은 [자동 파일 시스템 공유를 설정하는 방법](#) [64]을 참조하십시오.

참조 “[NFS의 버전 협상](#)” [24]

▼ 클라이언트에서 다른 NFS 버전을 선택하는 방법

다음 절차에서는 클라이언트에서 사용되는 NFS 버전을 제어하는 방법을 설명합니다. 기본적으로는 NFS 버전 4가 기본 버전으로 설정됩니다.

1. 관리자가 됩니다.

자세한 내용은 “[Oracle Solaris 11.2의 사용자 및 프로세스 보안](#)”의 “[지정된 관리 권한 사용](#)”을 참조하십시오.

2. SMF 매개변수를 변경하여 NFS 버전 번호를 설정합니다.

예를 들어 NFS 버전 3 프로토콜을 사용하여 모든 파일 시스템이 마운트되도록 하려면 `client_versmax` 및 `client_versmin` 매개변수의 값을 모두 3으로 설정합니다.

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

3. 클라이언트에서 NFS를 마운트합니다.

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

server-name 서버 이름입니다.

/share-point 원격 디렉토리의 경로입니다.

`/local-dir` 로컬 마운트 지점의 경로입니다.

참조 [“NFS의 버전 협상” \[24\]](#)

▼ mount 명령을 사용하여 클라이언트에서 다른 NFS 버전을 선택하는 방법

이 절차에서는 mount 명령을 사용하여 클라이언트에서 특정 마운트에 대해 사용되는 NFS 버전을 제어하는 방법을 설명합니다. 클라이언트에서 마운트하는 모든 파일 시스템에 대한 NFS 버전을 수정하는 방법을 찾으려면 [클라이언트에서 다른 NFS 버전을 선택하는 방법 \[74\]](#)을 참조하십시오.

1. 관리자가 됩니다.
자세한 내용은 [“Oracle Solaris 11.2의 사용자 및 프로세스 보안”](#)의 [“지정된 관리 권한 사용”](#)을 참조하십시오.
2. 클라이언트에서 원하는 NFS 버전을 마운트합니다.

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

`value` NFS 버전 번호

`server-name` 서버 이름

`/share-point` 원격 디렉토리의 경로입니다.

`/local-dir` 로컬 마운트 지점의 경로입니다.

참고 - 이 명령은 SMF 저장소의 클라이언트 설정을 대체합니다.

참조 [“NFS의 버전 협상” \[24\]](#)

보안 NFS 시스템 관리

보안 NFS 시스템을 사용하려면 모든 관리 대상 시스템에 도메인 이름이 있어야 합니다. 일반적으로 도메인은 대규모 네트워크에 속한 여러 시스템의 관리 엔티티입니다. 이름 서비스를 실행 중인 경우에는 도메인에 대해서도 이름 서비스를 설정해야 합니다. 이름 서비스에 대한 자세한 내용은 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)를 참조하십시오.

NFS 서비스에서는 Kerberos V5 인증을 지원합니다. 자세한 내용은 [“Oracle Solaris 11.2의 Kerberos 및 기타 인증 서비스 관리”](#)의 2 장, [“Kerberos 서비스 정보”](#)를 참조하십시오.

Diffie-Hellman 인증을 사용하도록 보안 NFS 환경을 구성할 수도 있습니다. Diffie-Hellman 인증에 대한 자세한 내용은 “Oracle Solaris 11.2의 Kerberos 및 기타 인증 서비스 관리”의 10 장, “네트워크 서비스 인증 구성”을 참조하십시오.

▼ DH 인증을 사용하여 보안 NFS 환경을 설정하는 방법

1. 도메인 이름을 지정합니다.

도메인의 각 시스템에 도메인 이름을 알립니다. 시스템의 NIS 도메인 이름 설정에 대한 자세한 내용은 “Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”의 “시스템의 NIS 도메인 이름을 설정하는 방법”을 참조하십시오.

```
# domainname domain-name
```

2. newkey 명령을 사용하여 클라이언트 사용자용으로 공개 키와 보안 키를 설정합니다.

```
# newkey -u username -s name-service
```

사용자는 chkey 명령을 사용하여 개인 보안 RPC 암호를 설정할 수 있습니다.

```
# chkey -p -s name-service -m mechanism
```

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

이러한 명령에 대한 자세한 내용은 [newkey\(1M\)](#) 및 [chkey\(1\)](#) 매뉴얼 페이지를 참조하십시오.

3. 이름 서비스가 응답하는지 확인합니다.

예를 들면 다음과 같습니다.

- NIS를 실행 중인 경우 ypbind 데몬이 실행 중인지 확인합니다. 자세한 내용은 “Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”의 “ypbind가 클라이언트에서 실행되고 있지 않음”을 참조하십시오.
- LDAP를 실행 중인 경우 ldap_cachemgr 데몬이 실행 중인지 확인합니다. 자세한 내용은 “Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: LDAP”의 “LDAP 클라이언트 상태 모니터링”을 참조하십시오.

4. 키 서버의 keysserv 데몬이 실행 중인지 확인합니다.

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

데몬이 실행 중이지 않으면 다음을 입력하여 키 서버를 시작합니다.

```
# svcadm enable network/rpc/keysserv
```

5. 암호 키의 암호를 해독하고 키를 저장합니다.

일반적으로 로그인 암호는 네트워크 암호와 같습니다. 이 경우 `keylogin`은 필요하지 않습니다. 암호가 다른 경우에는 사용자가 로그인하여 `keylogin`을 실행해야 합니다. `/etc/.rootkey`에 암호가 해독된 암호 키를 저장하려면 `keylogin -r` 명령을 `root`로 사용해야 합니다.

참고 - 루트 암호 키가 변경되거나 `/etc/.rootkey` 파일이 손실된 경우에는 `keylogin -r`을 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.

Diffie-Hellman 인증을 사용 중이면 `auto_master` 데이터를 편집하여 `sec=dh`를 해당 항목의 마운트 옵션으로 포함합니다.

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

시스템을 다시 설치하거나 이동하거나 업그레이드할 때 `root`에 대해 키를 변경하거나 새 키를 설정하지 않는 경우에는 `/etc/.rootkey` 파일을 저장해야 합니다. `/etc/.rootkey` 파일을 삭제하는 경우 다음 명령을 입력합니다.

```
# keylogin -r
```

WebNFS 관리

이 절에서는 WebNFS 시스템 관리 지침을 제공합니다.

표 3-4 WebNFS 관리(작업 맵)

작업	설명	지침
WebNFS 계획	WebNFS 서비스를 사용으로 설정하기 전에 고려해야 하는 문제점입니다.	“WebNFS 액세스 계획” [78]
WebNFS 사용으로 설정	WebNFS 프로토콜을 사용하여 NFS 파일 시스템을 마운트할 수 있도록 설정합니다.	WebNFS 액세스를 사용으로 설정하는 방법 [79]
방화벽을 통해 WebNFS 사용	WebNFS 프로토콜을 사용하여 방화벽을 통해 파일에 액세스할 수 있게 합니다.	“방화벽을 통해 WebNFS 액세스를 사용으로 설정” [80]
NFS URL을 사용하여 찾아보기	웹 브라우저 내에서 NFS URL을 사용합니다.	“브라우저를 사용하여 NFS URL 액세스” [79]
autofs와 함께 공유 파일 핸들 사용	자동 마운트를 사용하여 파일 시스템을 마운트할 때 공유 파일 핸들을 사용합니다.	autofs와 함께 공유 파일 핸들을 사용하는 방법 [95]

작업	설명	지침
autofs와 함께 NFS URL 사용	자동 마운트 맵에 NFS URL을 추가합니다.	autofs와 함께 NFS URL을 사용하는 방법 [96]
방화벽을 통해 파일 시스템 액세스 제공	WebNFS 프로토콜을 사용하여 방화벽을 통해 파일 시스템에 액세스할 수 있게 합니다.	방화벽을 통해 NFS 파일 시스템을 마운트하는 방법 [70]
NFS URL을 사용하여 파일 시스템 마운트	NFS URL을 사용하여 파일 시스템에 액세스할 수 있게 합니다. 이 프로세스를 통해 MOUNT 프로토콜을 사용하지 않고 파일 시스템에 액세스할 수 있습니다.	NFS URL을 사용하여 NFS 파일 시스템을 마운트하는 방법 [71]

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.

- 각 서버에는 기본적으로 서버의 루트 파일 시스템과 연결되는 하나의 공용 파일 핸들이 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 디렉토리와 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.
- 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.
- 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.
- `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. WebNFS 제한에 대한 자세한 내용은 [“웹 브라우저 사용 시의 WebNFS 제한” \[39\]](#)을 참조하십시오.

▼ WebNFS 액세스를 사용하여 설정하는 방법

시작하기 전에 기본적으로 NFS를 마운트할 수 있는 모든 파일 시스템은 자동으로 WebNFS 액세스용으로 제공됩니다. 다음 중 하나의 이유로 이 절차를 사용합니다.

- 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 HTML 파일을 강제로 로드하려는 경우

sharectl 유틸리티를 사용하여 NFS 등의 파일 공유 프로토콜을 구성할 수도 있습니다. 파일 공유 프로토콜 구성에 대한 자세한 내용은 [sharectl\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

WebNFS 서비스를 시작하기 전에 고려할 문제에 대한 자세한 내용은 “[WebNFS 액세스 계획](#)” [78]을 참조하십시오.

1. 관리자가 됩니다.

자세한 내용은 “[Oracle Solaris 11.2의 사용자 및 프로세스 보안](#)”의 “지정된 관리 권한 사용”을 참조하십시오.

2. WebNFS 서비스에서 공유할 파일 시스템을 정의합니다.

share 명령을 사용하여 각 파일 시스템을 정의합니다.

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

share_nfs 명령의 사용 가능한 옵션에 대한 자세한 내용은 [share_nfs\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

3. 지정된 옵션이 나열되어 있는지 확인합니다.

```
# share -F nfs
```

예를 들면 다음과 같습니다.

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

브라우저를 사용하여 NFS URL 액세스

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

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

<i>server</i>	파일 서버의 이름
<i>port</i>	사용할 포트 번호(2049, 기본값)
<i>path</i>	파일 경로(공용 파일 핸들 또는 루트 파일 시스템에 상대적일 수 있음)

참고 - 대부분의 브라우저에서는 `nfs` 또는 `http`와 같은 URL 서비스 유형이 트랜잭션 간에 저장됩니다. 다른 서비스 유형이 포함된 URL을 로드하면 예외가 발생합니다. 예를 들어 NFS URL을 사용한 후 HTTP URL에 대한 참조가 로드되면 후속 페이지는 NFS 프로토콜이 아닌 HTTP 프로토콜을 사용하여 로드됩니다.

방화벽을 통해 WebNFS 액세스를 사용으로 설정

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

NFS 참조 관리

NFS 참조를 사용하면 여러 NFS 버전 4 서버를 통합 이름 공간으로 연결하는 방법으로 NFS 버전 4 서버가 다른 NFS 버전 4 서버에 있는 파일 시스템을 가리킬 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. 예를 들면 다음과 같습니다.

```
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 마운트 지점에 액세스하여 참조를 마운트합니다.

```
client1# ls /share/docs
```

마운트에 실패하면 NFS 클라이언트에서 연결을 확인하고 NFS 서버에서 공유 파일 시스템을 확인합니다. NFS 문제 해결에 대한 자세한 내용은 “[NFS 문제 해결 절차](#)” [126]를 참조하십시오.

예 3-4 기존 NFS 참조 수정

server4:/tank/docs 등의 다른 파일 시스템을 이 절차에서 생성된 기존 참조에 추가하려면 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.

▼ NFS 참조를 제거하는 방법

- NFS 참조를 제거하려면 다음 명령을 입력합니다.

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

그러면 /share/docs에 생성된 단일 참조가 제거됩니다.

FedFS 관리

FedFS 프로토콜을 사용하여 통합 파일 시스템을 구성하고 유지 관리합니다. 이 파일 시스템은 여러 다른 파일 서버를 포함하여 다중 판매자 전역 이름 공간을 만들 수 있습니다.

FedFS Server에 대한 DNS 레코드 설정

적절한 DNS 레코드를 만든 후 마운트 지점에 액세스하면 자동 마운트에서 FedFS를 사용한 파일 시스템 마운트를 완료합니다. 서버에 대한 DNS 레코드는 다음과 같이 표시됩니다.

```
# nslookup -q=svr _nfs-domainroot._tcp.example.com bee.example.com
Server:          bee.example.com
Address:         192.168.1.1
```

```
_nfs-domainroot._tcp.example.com      service = 1 0 2049 bee.example.com.
```

DNS 레코드를 설정한 후 Oracle Solaris에서는 응용 프로그램이 /nfs4/example.com 마운트 지점에 액세스할 때 FedFS 파일 시스템을 자동으로 마운트합니다.

▼ 이름 공간 데이터베이스를 만드는 방법

NSDB(이름 공간 데이터베이스)는 단일 FedFS 이름 공간으로 결합되는 여러 유형 서버의 파일 세트에 대한 정보를 제공하는 데 사용됩니다. 이 절차는 LDAP 서버에서 수행됩니다.

시작하기 전에 LDAP 서버가 설치되어 있어야 합니다.

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. 다음 항목으로 `/etc/openldap/slapd.conf` 파일을 업데이트합니다.

```
include          /usr/lib/fs/nfs/fedfs-11.schema
suffix dc=example,dc=org
rootdn cn=Manager,dc=example,dc=org
rootpw password
```

3. FedFS 데이터에 대한 식별 이름을 만듭니다.

```
# nsdb-update-nci -l NSDB -r port -D bind_DN -w bind-PW nce
```

예를 들면 다음과 같습니다.

```
# nsdb-update-nci -l localhost -r 389 -D cn=Manager -w\
example.org dc=example,dc=org adding new entry "dc=example,dc=org"
NCE entry created
```

설명

-l NSDB를 구현하는 LDAP 서버를 지정합니다.

-r NSDB를 구현하는 LDAP 서버가 수신 중인 포트를 지정합니다.

-D NSDB 정보를 변경하도록 허용된 사용자의 고유 이름을 지정합니다.

-w 바인드 DN 사용자의 암호를 지정합니다.

자세한 내용은 [nsdb-update-nci\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

▼ NSDB에 대한 보안 연결을 사용하는 방법

시작하기 전에 LDAP 서버가 설치되어 있어야 합니다.

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. LDAP 서버에서 인증서를 만듭니다.

LDAP 트래픽을 보호하려면 인증서가 필요합니다.

```
# mkdir /etc/openldap/certs
# mkdir /etc/openldap/certs/keys
# cd /etc/openldap/certs
# openssl req -x509 -nodes -days 3650 -newkey rsa:2048 \
  -keyout keys/ldapskey.pem -out ldapscert.pem
# chown -R openldap:openldap /etc/openldap/certs/*
# chmod 0400 keys/ldapskey.pem
```

3. /etc/openldap/slapd.conf 파일에 선언을 추가합니다.

```
TLSertificateFile /etc/openldap/certs/ldapscert.pem
TLSertificateKeyFile /etc/openldap/certs/keys/ldapskey.pem
```

4. 인증서를 NFS 서버 및 클라이언트에 복사합니다.

```
# scp ldap-server:/etc/openldap/certs/keys/ldapskey.pem \
  /etc/openldap/certs/keys/ldapskey.pem
# chmod 0400 /etc/openldap/certs/keys/ldapskey.pem
```

5. NFS 서버 및 클라이언트에서 연결 항목을 업데이트합니다.

```
# nsdbparams update -f ldapscert.pem -t FEDFS_SEC_TLS localhost
```

nsdbparams 명령의 사용 가능한 옵션에 대한 자세한 내용은 [nsdbparams\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

▼ FedFS 참조를 만드는 방법

시작하기 전에 NFS 서버가 설치되어 있어야 합니다.

1. 관리자가 됩니다.

2. NSDB에 대한 연결 항목을 만듭니다.

이 명령은 LDAP 서버와 NFS 서버에 정의된 NSDB 간 연결 항목을 만듭니다.

```
# nsdbparams update -D cn=Manager,dc=example,dc=org -w example.org nsdb.example.org
```

3. FedFS 참조를 만듭니다.

```
# nfsref -t svc-type add path location
```

`-t svc-type` 참조의 서비스 유형을 지정합니다.

예를 들면 다음과 같습니다.

```
# nfsref -t nfs-fedfs add /share/docs server2:/usr/local/docs server3:/tank/docs  
Created reparse point /share/doc
```

◆◆◆ 4 장

autofs 관리

이 장에서는 파일 시스템 액세스, autofs 맵 수정, autofs에서 보안 제한 사항 사용과 같은 autofs 관리 작업을 수행하는 방법에 대한 정보를 제공합니다.

이 장의 내용:

- “autofs 관리” [85]
- “SMF 매개변수를 사용하여 autofs 환경 구성” [86]
- “맵 수정” [87]
- “자동 마운트 사용자 정의” [89]

참고 - 시스템에서 영역이 사용으로 설정된 경우 비전역 영역에서 이 기능을 사용하려면 “Oracle Solaris 영역 소개”를 참조하십시오.

autofs 관리

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

표 4-1 autofs 관리 작업

작업	설명	지침
autofs 시작 및 중지	시스템을 재부트하지 않고 자동 마운트 서비스 시작 및 중지	“자동 마운트 시작 및 중지” [72]
autofs SMF 매개변수를 통해 autofs 환경 구성	SMF 저장소에서 매개변수에 값 지정	“SMF 매개변수를 사용하여 autofs 환경 구성” [86]
autofs를 사용하여 파일 시스템 액세스	자동 마운트 서비스를 사용하여 파일 시스템 액세스	“자동 마운트를 사용한 마운트” [68]
autofs 맵 수정	다른 맵을 나열하는 데 사용되는 마스터 맵 수정 대부분 맵에 사용되는 간접 맵 수정 서버와 클라이언트의 마운트 지점 간에 직접 연결을 설정하는 데 사용되는 직접 맵 수정	마스터 맵을 수정하는 방법 [88] 간접 맵을 수정하는 방법 [88] 직접 맵을 수정하는 방법 [88]
비 NFS 파일 시스템 액세스를 위해 autofs 맵 수정 /home 맵 사용	CD-ROM 응용 프로그램의 항목으로 autofs 맵 설정 공통 /home 맵 설정	“비 NFS 파일 시스템 액세스” [89] “/home의 공통 보기 설정” [90]

작업	설명	지침
	여러 파일 시스템을 참조하는 /home 맵 설정	멀티홈 디렉토리 파일 시스템을 사용하여 /home을 설정하는 방법 [90]
새 autofs 마운트 지점 사용	프로젝트 관련 autofs 맵 설정 다른 클라이언트 구조를 지원하는 autofs 맵 설정 다른 운영 체제를 지원하는 autofs 맵 설정	공동 디렉토리 아래에서 프로젝트 관련 파일을 통합하는 방법 [91] 공유 이름 공간에 액세스하도록 서로 다른 구조를 설정하는 방법 [93] 호환되지 않는 클라이언트 운영 체제 버전을 지원하는 방법 [94]
autofs를 사용하여 파일 시스템 복제	폐일오버되는 파일 시스템 액세스 제공	여러 서버에서 공유 파일을 복제하는 방법 [94]
autofs를 통해 보안 제한 사용	파일에 대한 원격 root 액세스를 제한하면서 파일 시스템 액세스를 제공합니다.	autofs 보안 제한을 적용하는 방법 [95]
autofs와 함께 공용 파일 핸들 사용	파일 시스템을 마운트할 때 공용 파일 핸들을 강제로 사용합니다.	autofs와 함께 공용 파일 핸들을 사용하는 방법 [95]
autofs와 함께 NFS URL 사용	자동 마운트에서 사용할 수 있도록 NFS URL을 추가합니다.	autofs와 함께 NFS URL을 사용하는 방법 [96]
autofs 찾아보기 기능 사용 안함으로 설정	단일 클라이언트에서 autofs 마운트 지점이 자동으로 채워지지 않도록 찾아보기 기능을 사용 안함으로 설정 모든 클라이언트에서 autofs 마운트 지점이 자동으로 채워지지 않도록 찾아보기 기능을 사용 안함으로 설정 단일 클라이언트에서 특정 autofs 마운트 지점이 자동으로 채워지지 않도록 찾아보기 기능을 사용 안함으로 설정	단일 NFS 클라이언트에서 autofs 찾아보기 기능을 완전히 사용 안함으로 설정하는 방법 [96] 모든 클라이언트에 대해 autofs 찾아보기 기능을 사용 안함으로 설정하는 방법 [97] 선택한 파일 시스템에 대해 autofs 찾아보기 기능을 사용 안함으로 설정하는 방법 [97]

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. sharectl 명령을 사용하여 명령 줄에서 지정한 것과 같게 지정할 수 있습니다. 명령을 통해 지정하려면 키워드에 값을 제공하면 됩니다.

▼ SMF 매개변수를 사용하여 autofs 환경을 구성하는 방법

1. 관리자가 됩니다.
자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”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 automountd 명령의 -n 옵션에 해당합니다. autofs에 대해 지원되는 매개 변수에 대한 자세한 내용은 [autofs\(4\)](#)를 참조하십시오.

3. autofs 데몬을 다시 시작합니다.

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

맵 관련 관리 작업

선택하는 맵과 이름 서비스 유형은 autofs 맵을 변경할 때 사용해야 하는 방식에 영향을 줍니다.

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

맵의 유형 및 해당 용도는 다음과 같습니다.

- 마스터 - 디렉토리를 맵과 연결합니다.
- 직접 - autofs를 특정 파일 시스템에 연결합니다.
- 간접 - autofs를 참조 방식 파일 시스템에 연결합니다.

autofs 환경을 변경하는 방법은 이름 서비스를 기반으로 합니다. 로컬 파일을 이름 서비스로 사용 중인 경우 변경을 수행하려면 텍스트 편집기를 사용합니다. 이름 서비스가 NIS이면 make 파일을 사용합니다.

맵 유형에 대해 수행하는 수정에 따라 automount 명령을 실행해야 할 수 있습니다. 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. 마스터 맵을 변경하는 경우 항상 automount 명령을 실행해야 합니다. 간접 맵을 변경하는 경우 automount 명령을 실행하지 않아야 합니다.

맵 수정

이 절에서는 여러 자동 마운트 맵 유형을 업데이트하는 방법을 설명합니다.

▼ 마스터 맵을 수정하는 방법

1. 사용 중인 이름 서비스를 기반으로 맵을 변경할 권한이 있는 사용자로 로그인합니다. 로컬 맵 파일을 사용 중이면 root 역할을 맡습니다.
2. 마스터 맵을 변경합니다.
맵을 변경하기 위해 수행해야 하는 특정 단계는 사용 중인 이름 서비스에 따라 다릅니다. 로컬 파일을 이름 서비스로 사용 중이면 텍스트 편집기를 사용합니다. 이름 서비스가 NIS이면 make 파일을 사용합니다.
3. 각 클라이언트에 대해 관리자 권한을 얻습니다.
자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.
4. 각 클라이언트에 대해 automount 명령을 실행하여 변경 사항을 적용합니다.
5. 마스터 맵의 새로운 정보를 통합하려고 자신의 시스템에서 슈퍼 유저로 automount 명령을 실행하도록 사용자에게 알립니다.

▼ 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 시스템에서 슈퍼 유저로 automount 명령을 실행할 수 있도록 알림을 제공해야 합니다.

참고 - 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.

마운트 지점 충돌 방지

로컬 디스크 분할 영역이 `/src`에 마운트되어 있는 상태에서 `autofs` 서비스를 사용하여 다른 소스 디렉토리를 마운트하려는 경우에는 로컬 분할 영역에 연결할 때마다 NFS 서비스가 로컬 분할 영역을 숨깁니다. 따라서 약간 다른 위치에 분할 영역을 마운트해야 합니다.

예를 들어 `/export/src`에 분할 영역을 마운트하려면 다음과 같이 `/etc/vfstab` 파일에 항목을 추가합니다.

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

`auto_src`에도 항목을 추가해야 합니다. 이 예에서 시스템 이름은 `terra`입니다.

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

비 NFS 파일 시스템 액세스

`autofs`는 NFS 파일 외의 파일도 마운트할 수 있습니다(예: CD-ROM 또는 USB 플래시 드라이브 같은 이동식 매체의 파일).

서버에서 파일 시스템을 마운트하는 대신 매체를 드라이브에 배치하고 맵에서 파일 시스템을 참조합니다. 예를 들어 CD-ROM 응용 프로그램에 액세스하려면 관리자로 로그인하고 `autofs` 맵에서 다음 예와 같이 CD-ROM 장치 이름과 콜론을 차례로 사용하여 CD-ROM 파일 시스템의 항목을 추가합니다.

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

자동 마운트 사용자 정의

이 절에서는 쉽게 사용할 수 있는 디렉토리 구조를 제공하기 위해 자동 마운트 맵을 사용자 정의하는 방법을 설명합니다.

/home의 공통 보기 설정

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

모든 Oracle Solaris 설치 시에는 /etc/auto_master 마스터 맵이 기본적으로 설치됩니다.

```
# Master map for autofs
#
+auto_master
/net      -hosts      -nosuid,nobrowse
/home     auto_home  -nobrowse
/nfs4     -fedfs      -ro,nosuid,nobrowse
```

auto_home의 맵도 /etc 아래에 설치됩니다. 새 로컬 사용자가 생성되면 /etc/auto_home에 항목이 자동으로 추가됩니다. 예를 들면 다음과 같습니다.

```
# Home directory map for autofs
#
rusty     dragon:/export/home/&
+auto_home
```

dragon이라는 이름의 서버에서 rusty에 대한 홈 디렉토리를 /export/home/rusty 및 /home/rusty를 통해 액세스할 수 있습니다.

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

따라서 사용자는 실제 홈 디렉토리 위치를 몰라도 됩니다. 사용자가 디스크 공간이 더 필요하여 홈 디렉토리를 다른 서버로 옮겨야 하는 경우에는 auto_home 맵에서 사용자 항목을 변경하여 새 위치를 반영하면 됩니다. 다른 사용자가 계속해서 /home/user 경로를 사용할 수 있습니다.

참고 - 사용자가 자신의 홈 디렉토리에서 setuid 실행 파일을 실행하도록 허용하지 마십시오. 이 제한을 적용하지 않으면 모든 사용자가 모든 시스템에 대한 슈퍼 유저 권한을 가질 수 있습니다.

▼ 멀티홈 디렉토리 파일 시스템을 사용하여 /home을 설정하는 방법

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. **/export/home 아래에 홈 디렉토리 분할 영역을 설치합니다.**
시스템에 분할 영역이 여러 개 있는 경우에는 /export/home1, /export/home2 등과 같은 별도의 디렉토리 아래에 분할 영역을 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.

```
user1      system1:/export/home1/&
user2      system1:/export/home1/&
user3      system2:/export/home2/&
user4      system1:/export/home3/&
```

여기서는 &(앰퍼센드)를 맵 키 대신 사용했습니다. 앰퍼센드는 다음 예에서 두번째 user1 항목의 약어입니다.

```
user1      system1:/export/home1/user1
```

▼ 공통 디렉토리 아래에서 프로젝트 관련 파일을 통합하는 방법

autofs를 사용하여 여러 시스템에서 공통으로 사용되는 디렉토리에 파일을 통합할 수 있습니다. 프로젝트 관련 파일의 디렉토리 구조를 공통 디렉토리에 대한 autofs 맵에 추가할 수 있습니다. 이 디렉토리 구조를 통해 사용자는 시스템의 물리적 및 하드웨어 변경 사항에 관계 없이 프로젝트 파일을 사용할 수 있습니다.

1. **관리자가 됩니다.**
자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.
2. **공통 디렉토리에 대한 항목을 auto_master 맵에 추가합니다.**

```
/common-dir      auto_common-dir      -nosuid
```


auto_common-dir 맵에서 공통 디렉토리의 내용이 결정됩니다.
3. **-nosuid 옵션을 예방 조치로 추가하여 사용자가 작업 공간에 있을 수 있는 setuid 프로그램을 실행하지 못하게 합니다.**
4. **auto_common-dir 맵에 항목을 추가합니다.**
auto_common-dir 맵은 각 항목이 하위 프로젝트를 설명하는 방식으로 구성됩니다. 첫번째 시도에서는 다음과 같은 맵이 생성됩니다.

```
project1 system1:/export/common-dir/&
project2 system1:/export/common-dir/&
app1     system2:/export/common-dir/&
```

각 항목 끝의 앰퍼센드(&)는 항목 키의 약어입니다.

예 4-1 /ws 아래에서 프로젝트 관련 파일 통합

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

/ws 디렉토리에 대한 항목을 사이트 auto_master 맵에 추가합니다.

```
/ws auto_ws -nosuid
```

auto_ws 맵에서 /ws 디렉토리의 내용이 결정됩니다. -nosuid는 작업 영역에 있을 수 있는 setuid 프로그램을 사용자가 실행할 수 없도록 합니다. 각 항목이 하위 프로젝트를 설명하도록 auto_ws 맵에 항목을 추가합니다. 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 첫 시도에서는 간단하게 나타나지만 추가적인 구체화가 필요한 맵이 제공됩니다. 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/compiler`를 참조하려면 `vers1.0`, `vers2.0` 및 `man` 디렉토리에 대한 3개의 마운트가 필요합니다.

각 행의 끝에 오는 백슬래시는 입력 내용이 다음 행으로 이어진다는 것을 나타냅니다. 효율성을 위해 한 행에 모두 입력했지만 가독성을 위해 줄바꿈과 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 사항이 `auto_ws` 맵에 반영되어도 사용자에게 알릴 필요가 없습니다. `/ws` 아래의 디렉토리 계층은 변경되지 않기 때문입니다.

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

▼ 공유 이름 공간에 액세스하도록 서로 다른 구조를 설정하는 방법

스프레드시트 응용 프로그램, 워드 프로세싱 패키지 등의 응용 프로그램과 로컬 실행 파일에 대해 공유 이름 공간을 어셈블해야 합니다. 이 이름 공간의 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.**
이름 지정 서비스에 대한 자세한 내용은 “Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”를 참조하십시오.
2. **공유 이름 공간에 대해 사이트 특정 이름을 하나 선택합니다.**
이 이름을 통해 해당 이름 공간에 속하는 파일과 디렉토리를 쉽게 식별할 수 있습니다. 예를 들어 이름으로 `/usr/local`을 사용하는 경우 `/usr/local/bin` 경로는 이 이름 공간에 속합니다.
3. **autofs 직접 맵을 만들어 사용자가 특정 파일 시스템에 액세스할 수 있게 합니다.**
`/usr/local`에서 이 맵을 마운트합니다. NIS `auto_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
```

여기서 aa는 서버 이름입니다.

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

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

여기서 aa는 서버 이름입니다.

- SPARC 클라이언트의 경우에는 /export/local/bin/sparc에 실행 파일을 저장합니다.
- x86 클라이언트의 경우에는 /export/local/bin/i386에 실행 파일을 저장합니다.

▼ 호환되지 않는 클라이언트 운영 체제 버전을 지원하는 방법

1. 클라이언트의 운영 체제 유형을 결정하는 변수와 아키텍처 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 자세한 내용은 “클라이언트측 페일오버” [35]를 참조하십시오.

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. autofs 맵에서 심표로 구분된 모든 복제 서버 목록을 만듭니다.

예를 들면 다음과 같습니다.

```
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 제한

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

이 항목은 일반 로컬 /etc/auto_master 파일에서 /home에 대한 항목을 대체합니다. 일반 로컬 /etc/auto_master 파일에 대한 자세한 내용은 “/home의 공통 보기 설정” [90]을 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.

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

▼ autofs 보안 제한을 적용하는 방법

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. 이름 서비스 auto_master 파일에서 -nosuid 항목을 만듭니다.

```
/home    auto_home    -nosuid
```

▼ autofs와 함께 공용 파일 핸들을 사용하는 방법

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. autofs 맵에서 다음 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 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. autofs 맵에서 다음 항목을 만듭니다.

```
/usr/local -ro nfs://server-name/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 버전에서는 -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.

이 절에서는 단일 클라이언트, 모든 클라이언트, 선택한 파일 시스템에 대한 찾아보기 기능을 해제하는 방법을 설명합니다.

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

1. NFS 클라이언트에서 관리자 권한을 얻습니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”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 실행하여 새 동작을 적용합니다.
재부트한 후에도 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 정보는 자동 마운트 정보의 검색 순서를 보여줍니다.

```
# 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 이름 공간의 항목보다 우선적으로 검색되도록 하려면 +auto_master 항목을 /net 뒤로 이동해야 합니다.

```
# Master map for automounter
```

```
#  
/net    -hosts    -nosuid  
/home   auto_home  
/nfs4   -fedfs     -ro,nosuid,nobrowse  
+auto_master
```

표준 구성에서는 +auto_master 항목이 파일의 맨 위에 배치되어 로컬 변경 사항이 사용되지 않게 합니다.

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
```

◆◆◆ 5 장

네트워크 파일 시스템 관리 명령

이 장에서는 네트워크 파일 시스템을 관리하는 데 사용되는 명령줄 유틸리티에 대해 설명합니다.

이 장의 내용:

- “NFS 명령” [99]
- “FedFS 명령” [118]

참고 - 시스템에서 영역이 사용으로 설정된 경우 비전역 영역에서 이 기능을 사용하려면 “Oracle Solaris 영역 소개”를 참조하십시오.

NFS 명령

이러한 명령이 완전히 효과를 보려면 root로 실행되어야 합니다. 하지만 정보 요청은 모든 사용자가 할 수 있습니다.

- “automount 명령” [100]
- “clear_locks 명령” [100]
- “fsstat 명령” [101]
- “mount 명령” [102]
- “mountall 명령” [108]
- “nfsref 명령” [118]
- “sharectl 명령” [109]
- “share 명령” [111]
- “shareall 명령” [116]
- “showmount 명령” [117]
- “umount 명령” [107]
- “umountall 명령” [108]
- “unshare 명령” [116]
- “unshareall 명령” [116]

또한 FedFS 서비스와 관련된 명령은 “FedFS 명령” [118]을 참조하십시오.

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 상세 정보 표시 모드를 선택합니다. 이 명령을 상세 정보 표시 모드로 실행하면 문제를 보다 쉽게 해결할 수 있습니다.

구체적으로 설정하지 않는 경우 기간의 값은 5분으로 설정됩니다. 대부분의 경우에는 기본 값을 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 사용하거나(df(1M) 매뉴얼 페이지 참조) egrep를 사용하여 df에서 출력을 필터링할 수 있습니다.

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 “[맵 수정](#)” [87]을 참조하십시오.

sharectl 명령을 사용하여 명령줄에서 지정한 것과 같게 지정할 수 있습니다. 그러나 명령줄 옵션과는 달리 SMF 저장소에서는 서비스 다시 시작, 시스템 재부트 및 시스템 업그레이드 시에도 지정 사항이 보존됩니다. automount 명령에 대해 다음 매개변수를 설정할 수 있습니다.

timeout

파일 시스템 마운트를 해제할 때까지 파일 시스템이 유훘 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 automountd용 -v 인수와 동등합니다. 기본값은 FALSE입니다.

clear_locks 명령

이 명령을 사용하면 NFS 클라이언트에 대한 모든 파일, 레코드 및 공유 잠금을 제거할 수 있습니다. 이 명령은 root에서 실행해야 합니다. NFS 서버에서 특정 클라이언트에 대한 잠금을 지울 수 있습니다. NFS 클라이언트에서 특정 서버의 해당 클라이언트에 대한 잠금을 지울

수 있습니다. 다음 예에서는 현재 시스템에서 이름이 tulip인 NFS 클라이언트에 대한 잠금을 지웁니다.

```
# clear_locks tulip
```

-s 옵션을 사용하여 잠금을 지울 NFS 호스트를 지정합니다. 이 옵션은 잠금을 만든 NFS 클라이언트에서 실행해야 합니다. 이 경우 클라이언트의 잠금은 bee라는 NFS 서버에서 제거됩니다.

```
# clear_locks -s bee
```



주의 - 이 명령은 클라이언트가 충돌하여 잠금을 지울 수 없는 경우에만 실행해야 합니다. 데이터 손실 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.

-i 마운트 지점의 I/O 작업에 대한 통계를 표시합니다.

-n 마운트 지점의 이름 지정 작업에 대한 통계를 표시합니다.

다음 예에서는 NFS 버전 3, NFS 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  /
```

다음 예에서는 -i 옵션을 사용하여 NFS 버전 3, NFS 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  /
```

다음 예에서는 -n 옵션을 사용하여 NFS 버전 3, NFS 버전 4 및 root 마운트 지점의 이름 지정 작업에 대한 통계를 제공합니다.

```
% 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 mount는 컴퓨터에 현재 마운트되어 있는 파일 시스템 목록을 표시합니다.

표준 Oracle Solaris 설치에 포함된 각 파일 시스템 유형에는 mount 명령에 대한 특정 옵션이 있습니다. NFS 파일 시스템 옵션에 대해서는 [mount_nfs\(1M\)](#) 매뉴얼 페이지를 참조하십시오. UFS 파일 시스템 옵션에 대해서는 [mount_ufs\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

표준 `server:/pathname` 구문 대신 NFS URL을 사용하여 NFS 서버에서 마운트할 경로 이름을 선택할 수 있습니다. 자세한 내용은 [NFS URL을 사용하여 NFS 파일 시스템을 마운트하는 방법 \[71\]](#)을 참조하십시오.



주의 - mount 명령은 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

이 절에서는 NFS 파일 시스템을 마운트할 때 `-o` 플래그 뒤에 사용할 수 있는 일부 옵션에 대해 설명합니다. 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`이며, 이 옵션을 선택하면 마운트가 완료될 때까지 추가 처리를 수행할 수 없기 때문에 사용 가능해야 하는 파일 시스템에서 가장 적합한 선택입니다. 중요하지 않은 파일 시스템의 경우에는 `bg`를 선택하는 것이 좋는데, 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(`noforcedirectio`).

응용 프로그램이 클라이언트의 단일 파일에 대해 동시 쓰기 및 읽기 및 쓰기를 실행하게 허용하려면 `forcedirectio` 마운트 옵션을 사용합니다. 이 옵션은 마운트된 파일 시스템 내의 모든 파일에 대해 이 기능을 사용으로 설정합니다. 또한 `directio()` 인터페이스

스를 사용하여 클라이언트의 단일 파일에서 이 기능을 사용으로 설정할 수 있었습니다. 이 기능을 사용으로 설정하지 않으면 파일에 대한 쓰기가 일련화됩니다. 또한 동시 쓰기 나 동시 읽기 및 쓰기가 발생하는 경우 해당 파일에 대해 더 이상 POSIX 의미가 지원되지 않습니다.

이 옵션을 사용하는 방법의 예제는 “[mount 명령 사용](#)” [105]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 largefiles를 사용하여 마운트됩니다. NFS 버전 2 프로토콜을 사용하는 마운트의 경우 largefiles 옵션을 사용하면 마운트가 실패하고 오류가 발생합니다.

nolargefiles

UFS 마운트에 대해 이 옵션을 사용하면 파일 시스템에 큰 파일이 없도록 할 수 있습니다. 큰 파일의 존재 여부는 NFS 서버에서만 제어할 수 있으므로 NFS 마운트를 사용할 때는 nolargefiles에 대한 옵션이 없습니다. 이 옵션을 사용하여 파일 시스템 NFS 마운트를 시도하면 작업이 거부되고 오류가 발생합니다.

nosuid|suid

nosuid 옵션은 noisetuid 옵션과 함께 nodevices 옵션을 지정하는 것에 해당합니다. nodevices 옵션이 지정된 경우에는 마운트된 파일 시스템에서 장치 특정 파일을 열 수 없습니다. noisetuid 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 noisetuid와 결합하는 경우의 효과에 대해 설명합니다. 각 옵션을 결합할 때는 가장 제한적인 옵션에 따라 동작이 결정됩니다.

결합된 옵션의 동작	옵션	옵션	옵션
noisetuid와 nodevices를 결합한 경우에 해당하는 옵션	nosuid	noisetuid	nodevices
noisetuid와 nodevices를 결합한 경우에 해당하는 옵션	nosuid	noisetuid	devices
noisetuid와 nodevices를 결합한 경우에 해당하는 옵션	nosuid	setuid	nodevices

결합된 옵션의 동작	옵션	옵션	옵션
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 신뢰할 수 없는 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 사용 가능한 값은 다음과 같습니다.

- Kerberos 버전 5 인증 서비스의 경우 krb5
- Kerberos 버전 5(무결성 포함)의 경우 krb5i
- Kerberos 버전 5(프라이버시 포함)의 경우 krb5p
- 인증이 없을 경우 none
- Diffie-Hellman(DH) 인증의 경우 dh
- 표준 UNIX 인증의 경우 sys

/etc/nfssec.conf에서도 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 hard로, 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 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 예에서는 다른 시나리오를 보여줍니다.

- NFS 버전 2 또는 NFS 버전 3에서는 다음 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 NFS 3에서 다음 명령의 -o 옵션은 /usr/man이 이미 마운트되었어도 bee 서버의 매뉴얼 페이지가 로컬 시스템에 마운트되도록 강제 지정합니다.

```
# mount -F nfs -O bee:/export/share/man /usr/man
```

NFS 버전 4에서는 다음 명령이 동일한 마운트를 수행합니다.

```
# mount -F nfs -o vers=4 -O bee:/export/share/man /usr/man
```

- NFS 버전 2 또는 NFS 버전 3에서 다음 명령은 클라이언트 페일오버를 사용합니다.

```
# mount -F nfs -r bee,wasp:/export/share/man /usr/man
```

NFS 버전 4에서 다음 명령은 클라이언트 페일오버를 사용합니다.

```
# mount -F nfs -o vers=4 -r bee,wasp:/export/share/man /usr/man
```

참고 - 명령줄에서 사용하는 경우 나열된 서버는 동일한 NFS 프로토콜 버전을 지원해야 합니다. 명령줄에서 mount를 실행할 때는 NFS 버전 2 및 NFS 버전 3 서버를 둘 다 사용해서는 안 됩니다. autofs는 NFS 버전 2 또는 NFS 버전 3 서버의 가장 적절한 하위 세트를 자동으로 선택하므로 autofs와 함께 두 서버를 모두 사용할 수 있습니다.

- 다음 예에서는 NFS 버전 2 또는 NFS 버전 3에서 mount 명령과 함께 NFS URL을 사용하는 방법을 보여줍니다.

```
# mount -F nfs nfs://bee//export/share/man /usr/man
```

다음 예에서는 NFS 버전 4에서 mount 명령과 함께 NFS URL을 사용하는 방법을 보여줍니다.

```
# mount -F nfs -o vers=4 nfs://bee//export/share/man /usr/man
```

- 다음 예에서는 forcedirectio 마운트 옵션을 사용하여 클라이언트가 파일에 대한 동시 쓰기 및 동시 읽기 및 쓰기를 허용하게 하는 방법을 보여줍니다.

```
# 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 사용하여 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 부속 시스템에서 보다 효율적인 I/O 시퀀스를 제공하여 I/O를 최적화할 수 있습니다. 동기 쓰기에서는 저장소 부속 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.

참고 - 동시 쓰기 지원을 사용하여 설정하지 않으면 모든 쓰기 요청이 일련화됩니다. 쓰기 요청이 진행 중이면 두번째 쓰기 요청은 첫번째 쓰기 요청이 완료될 때까지 기다린 후에 진행됩니다.

- 다음 예에서는 인수를 포함하지 않고 `mount` 명령을 사용하여 클라이언트에 마운트된 파일 시스템을 표시하는 방법을 보여줍니다.

```
% 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

`umount` 명령을 사용하면 현재 마운트된 원격 파일 시스템을 제거할 수 있습니다. `umount` 명령과 함께 다음 옵션을 사용할 수 있습니다.

- v 테스트를 사용으로 설정합니다.
- a 여러 파일 시스템을 한 번에 마운트 해제합니다. `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 사용 중인 파일 시스템을 강제로 마운트 해제합니다. 이 옵션을 사용하여 마운트할 수 없는 파일 시스템 마운트를 시도하는 동안 정지된 클라이언트의 정지를 해제할 수 있습니다.



주의 - 파일 시스템에 파일을 쓰는 중인 경우 해당 시스템을 강제로 마운트 해제하면 데이터가 손실될 수 있습니다.

예 5-1 파일 시스템 마운트 해제

다음 예제에서는 `/usr/man`에 마운트되어 있는 파일 시스템 마운트를 해제합니다.

```
# umount /usr/man
```

예 5-2 umount에서 옵션 사용

다음 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

mountall 명령을 사용하여 파일 시스템 표에 나열되어 있는 특정 파일 시스템 그룹이나 모든 파일 시스템을 마운트합니다. 이 명령은 다음 옵션을 제공합니다.

- F *FSType* 액세스할 파일 시스템 유형을 선택합니다.
- r 파일 시스템 표에 나열되어 있는 모든 원격 파일 시스템을 선택합니다.
- l 모든 로컬 파일 시스템을 선택합니다.

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

umountall 명령을 사용하여 파일 시스템 그룹 마운트를 해제합니다. umountall 명령과 함께 다음 옵션을 사용할 수 있습니다.

- k `fuser -k mount-point` 명령을 실행하여 *mount-point*와 연관된 모든 프로세스를 강제 종료합니다.
- s 마운트 해제가 병렬로 수행되지 않음을 나타냅니다.
- l 로컬 파일 시스템만 사용하도록 지정합니다.
- r 원격 파일 시스템만 사용하도록 지정합니다.

`-h host` 이름 지정된 호스트의 모든 파일 시스템을 마운트 해제하도록 지정합니다. `-h` 옵션을 `-l` 또는 `-r`과 결합하여 사용할 수는 없습니다.

다음 예에서는 원격 호스트에서 마운트된 모든 파일 시스템을 마운트 해제합니다.

```
# umountall -r
```

다음 예에서는 `bee` 서버에서 현재 마운트되어 있는 모든 파일 시스템을 마운트 해제합니다.

```
# 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.

`set` 파일 공유 프로토콜의 등록 정보를 정의합니다. 등록 정보 및 등록 정보 값 목록은 [nfs\(4\)](#) 매뉴얼 페이지에 설명되어 있는 매개변수를 참조하십시오.

`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
- [“set 하위 명령” \[109\]](#)
- [“get 하위 명령” \[110\]](#)
- [“status 하위 명령” \[111\]](#)

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
```

set 하위 명령을 사용하려면 root 권한이 있어야 합니다.

각 추가 등록 정보 값에 대해 이 명령을 반복할 필요는 없습니다. -p 옵션을 여러 번 사용하여 같은 명령에서 여러 등록 정보를 정의할 수 있습니다.

다음 예제에서는 클라이언트의 최소 NFS 프로토콜 버전을 3으로 설정합니다.

```
# sharectl set -p 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
```

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=example.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 표시합니다. 이 하위 명령은 온라인 도움말 설명을 제공하는 -h 옵션을 지원합니다.

status 하위 명령은 다음 구문을 사용합니다.

```
# sharectl status [-h] [protocol]
```

다음 예에서는 NFS 프로토콜의 상태를 보여줍니다.

```
# sharectl status nfs
nfs    enabled
```

share 명령

share 명령을 사용하여 NFS 서버의 로컬 파일 시스템을 마운트 가능하도록 지정합니다. share 명령을 사용하여 시스템에서 현재 공유 중인 파일 시스템 목록을 표시할 수도 있습니다. NFS 서버가 실행 중이어야 share 명령이 작동합니다.

공유할 수 있는 객체에는 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/local이 한 디스크 슬라이스에 있는 경우 /usr을 공유하거나 /usr/local을 공유할 수 있습니다. 그러나 서로 다른 공유 옵션을 사용하여 두 디렉토리를 모두 공유해야 하는 경우에는 /usr/local을 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. 더 안전한 방법을 사용하려면 읽기/쓰기로 지정해야 하는 파일 시스템을 읽기 전용으로 공유해야 하는 파일 시스템과 다른 별도의 분할 영역이나 디스크 슬라이스에 배치합니다.

참고 - 파일 시스템 공유를 해제했다가 다시 공유할 때 NFS 버전 4가 작동하는 방식에 대한 자세한 내용은 “[NFS 버전 4에서 파일 시스템 공유 해제 및 다시 공유](#)” [25]를 참조하십시오.

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=access-list`

파일 시스템이 나열된 클라이언트에 한해 읽기/쓰기로 공유됩니다. 다른 모든 요청은 거부됩니다. 자세한 내용은 “[share 명령을 사용하여 액세스 목록 설정](#)” [114]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. ACL(액세스 제어 목록)에 대한 자세한 내용은 “[Oracle Solaris 11.2의 파일 보안 및 파일 무결성 확인](#)”의 “[액세스 제어 목록을 사용하여 UFS 파일 보호](#)”를 참조하십시오.

참고 - 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`

`anon`을 사용하여 인증되지 않은 사용자의 사용자 ID를 선택합니다. `anon`을 `-1`로 설정하면 서버에서 인증되지 않은 사용자의 액세스를 거부합니다. `anon=0`을 설정하여 루트 액세스 권한을 부여하면 인증되지 않은 사용자가 루트 액세스 권한을 가질 수 있으므로 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 httpd에 대해 DirectoryIndex 옵션을 설정하는 것과 동등합니다. 예를 들어 share 명령의 결과가 다음과 같다고 가정합니다.

```
export_web /export/web nfs sec=sys,public,index=index.html,ro
```

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.conf에서 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=access-list

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 목록에 클라이언트 호스트 이름만 포함할 수 있습니다. 자세한 내용은 “[share 명령을 사용하여 액세스 목록 설정](#)” [114]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=hostname

AUTH_SYS 또는 RPCSEC_GSS와 같은 보안 NFS 모드에서는 서버가 액세스 목록에서 파생되는 호스트 기반 기본 이름 목록에 대해 클라이언트의 기본 이름을 확인합니다. 클라이언트 기본 이름의 일반 구문은 root@hostname입니다. Kerberos V의 경우 구문은

`root/hostname.fully.qualified@REALM`입니다. `hostname` 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 11.2의 Kerberos 및 기타 인증 서비스 관리”의 “Kerberos 클라이언트 구성”을 참조하십시오.

`-sec=mode[:mode]`

이 옵션은 파일 시스템 액세스 권한을 얻는 데 필요한 보안 모드를 설정합니다. 기본 보안 모드는 UNIX 인증입니다. 여러 모드를 지정할 수는 있지만 명령줄당 각 보안 모드를 한 번씩만 사용해야 합니다. 각 `-sec=` 옵션은 다른 `-sec=` 옵션이 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. 기본값은 30000초(8.3시간)입니다.

share 명령을 사용하여 액세스 목록 설정

`share` 명령을 제공하는 액세스 목록에는 도메인 이름, 서브넷 번호 또는 액세스를 거부할 항목은 물론 표준 `-ro=`, `-rw=` 또는 `-root=` 옵션도 포함될 수 있습니다. 이러한 확장을 사용하면 긴 클라이언트 목록을 유지 관리하거나 이름 공간을 변경할 필요 없이 단일 서버에서 파일 액세스를 간편하게 제어할 수 있습니다.

다음 예에서 대부분의 시스템에 대해서는 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 예에서는 `eng` 넷 그룹의 모든 호스트에 대한 읽기 전용 액세스 권한을 지정합니다. `rose` 클라이언트에만 읽기/쓰기 액세스 권한이 부여됩니다.

```
# share -F nfs -o ro=eng,rw=rose /usr/src
```

참고 - 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 객체에 대해 `share` 명령을 여러 번 호출하면 마지막으로 실행된 명령만 적용됩니다. 다음 예에서는 3개 클라이언트 시스템에 대한 읽기/쓰기 권한을 사용으로 설정하지만, `rose` 및 `tulip`에만 `root`로서 파일 시스템 액세스 권한이 제공됩니다.

```
# 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`, `-ro=`, `-rw`, `-rw=`, `-root` 및 `-window` 옵션을 포함하십시오. 이 예에서는 이름이 `eng`인 넷 그룹의 모든 호

스트에 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 lilac은 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 예에서는 eng.example.com 도메인의 모든 호스트에 대한 마운트 액세스를 허용합니다.

```
# share -F nfs -o ro=.:.eng.example.com /export/share/man
```

이 예에서는 단일 점이 NIS 이름 공간을 통해 일치되는 모든 호스트와 일치합니다. 이러한 이름 서비스에서 반환되는 결과에는 도메인 이름이 포함되지 않습니다. .eng.example.com 항목은 이름 공간 확인을 위해 DNS를 사용하는 모든 호스트와 일치합니다. DNS는 항상 정규화된 호스트 이름을 반환하므로 DNS와 다른 이름 공간을 결합하여 사용하는 경우에는 더 긴 항목이 필요합니다.

실제 네트워크 번호 또는 네트워크 이름과 @ 기호를 앞에 붙여 액세스 목록에서 서브넷 번호를 사용할 수 있습니다. 이 문자는 네트워크 이름을 넷 그룹이나 정규화된 호스트 이름과 구분합니다. 서브넷은 /etc/networks 또는 NIS 이름 공간에서 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 자세한 내용은 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

unshare 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.

참고 - 파일 시스템 공유를 해제했다가 다시 공유할 때 NFS 버전 4가 작동하는 방식에 대한 자세한 내용은 [“NFS 버전 4에서 파일 시스템 공유 해제 및 다시 공유” \[25\]](#)를 참조하십시오.

다음 예에서는 파일 시스템 /usr/src를 공유 해제합니다.

```
# unshare /usr/src
```

shareall 명령

shareall 명령을 사용하여 여러 파일 시스템을 공유할 수 있습니다. 옵션을 포함하지 않고 사용하는 경우 이 명령은 SMF 저장소의 모든 항목을 공유합니다. 파일 이름을 포함하여 share 명령줄이 나열되는 파일 이름을 지정할 수 있습니다.

다음 예에서는 로컬 파일에 나열되는 모든 파일 시스템을 공유합니다.

```
# shareall /etc/dfs/special_dfstab
```

unshareall 명령

unshareall 명령은 모든 현재 공유된 리소스를 사용할 수 없게 지정합니다. -F *FSType* 옵션은 /etc/dfs/fstypes에 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/fstypes에서 정의됩니다. 특정 파일 시스템을 선택하려면 unshare 명령을 사용합니다.

다음 예에서는 모든 NFS 유형 파일 시스템을 공유 해제합니다.

```
# unshareall -F nfs
```

showmount 명령

showmount 명령을 사용하여 다음 정보를 표시합니다.

- NFS 서버에서 공유되는 원격으로 마운트된 파일 시스템을 포함하는 모든 클라이언트
- 클라이언트가 마운트하는 파일 시스템 전용
- 클라이언트 액세스 정보를 포함하는 공유되는 파일 시스템

참고 - 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 예에서는 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 예에서는 마운트된 디렉토리를 나열합니다.

```
# showmount -d bee
/export/share/man
/usr/src
```

다음 예에서는 공유된 파일 시스템을 나열합니다.

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

/network/nfs/server:default 서비스의 `nfs_props/showmount_info` 등록 정보는 `showmount` 명령으로 클라이언트에 표시되는 정보의 양을 제어합니다. 기본값은 `full`입니다. 이 값을 `none`으로 설정하면 클라이언트는 클라이언트가 마운트할 수 있는 서버의 원격 파일 시스템만 볼 수 있습니다. 다른 클라이언트에 대한 정보는 표시되지 않습니다. 이 등록 정보를 변경하는 방법에 대한 예는 예 3-3. “클라이언트에 표시되는 파일 시스템 정보 제한”을 참조하십시오.

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 확인합니다.

location 구문 재분석 지점과 연관시킬 NFS 또는 SMB 공유 파일 시스템을 하나 이상 식별합니다.

FedFS 명령

다음 명령은 FedFS 서비스와 관련됩니다.

`nsdb-list` LDAP 서버에 저장된 모든 FedFS 데이터를 나열합니다.

`nsdb-nces` LDAP 서버의 이름 지정 컨텍스트 및 관련 고유 이름을 나열합니다.

`nsdb-resolve-fsn` 선택한 파일 세트 이름의 파일 세트 위치를 보여줍니다.

`nsdb-update-nci` FedFS 데이터의 고유 이름을 관리합니다.

`nsdbparams` FedFS 연결을 관리합니다.

명령 사용 방법의 예를 보려면 “FedFS 관리” [81]를 참조하십시오.

◆◆◆ 6 장 6

네트워크 파일 시스템 문제 해결

이 장에서는 NFS 문제 해결 전략, 절차 및 명령에 대한 정보를 제공합니다. 이 장에는 autofs 문제 해결에 대한 정보와 NFS 오류 메시지 및 해당 의미의 목록도 포함되어 있습니다.

이 장의 내용:

- “NFS 문제 해결 전략” [119]
- “NFS 문제 해결용 명령” [120]
- “NFS 문제 해결 절차” [126]
- “autofs 문제 해결” [131]
- “NFS 오류 메시지” [135]

NFS 문제 해결 전략

NFS 문제를 추적할 때는 가능한 주요 오류 지점(클라이언트/네트워크)을 기억하십시오. 각 구성 요소를 격리하여 작동하지 않는 구성 요소를 찾아보십시오. `mountd` 및 `nfsd` 데몬이 항상 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 여러 번 시도한 후에 클라이언트 시스템 호출에서 오류가 반환됩니다. 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)
```



주의 - 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 해결용 명령

이 절에서는 NFS 문제 해결에 사용할 수 있는 명령에 대해 설명합니다.

nfsstat 명령

이 명령은 NFS 및 RPC 연결에 대한 통계 정보를 표시합니다. 다음 구문을 사용하여 NFS 서버 및 클라이언트 통계를 표시합니다.

```
# nfsstat [ -cmnrzs ]
```

- c 클라이언트측 정보를 표시합니다.
- m NFS 마운트된 각 파일 시스템의 통계를 표시합니다.
- n 클라이언트측과 서버측에 모두 NFS 정보를 표시합니다.
- r RPC 통계를 표시합니다.
- s 서버측 정보를 표시합니다.
- z 통계가 0으로 설정되도록 지정합니다.

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.

예 6-1 NFS 서버 통계 표시

```
# 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%
    
```

이 예에서는 RPC 및 NFS 작업에 대한 통계를 표시하는 방법을 보여줍니다. 두 통계 세트에서 badcalls 또는 calls의 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 rmdir입니다. NFS 버전 3 및 NFS 버전 4 통계의 경우 확인해야 하는 값은 commit입니다. 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

pstack 명령은 각 프로세스에 대한 스택 추적을 표시합니다. pstack 명령은 프로세스 소유자가 실행하거나 root에서 실행해야 합니다. pstack 명령을 사용하여 프로세스가 정지된 위치를 확인할 수 있습니다. 이 명령에 사용할 수 있는 옵션은 확인할 프로세스의 프로세스 ID뿐입니다. pstack 명령에 대한 자세한 내용은 [proc\(1\)](#) 매뉴얼 페이지를 참조하십시오.

예 6-2 NFS 프로세스에 대한 스택 추적 표시

```

# /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. 정지된 프로세스 해결에 대한 자세한 내용은 [NFS 서비스를 다시 시작하는 방](#)

법 [130]을 참조하십시오. NFS 문제 해결에 대한 자세한 내용은 “NFS 문제 해결 절차” [126]를 참조하십시오.

rpcinfo 명령

rpcinfo 명령은 시스템에서 실행 중인 RPC 서비스에 대한 정보를 생성합니다. 다음 명령 구문을 사용하여 RPC 서비스에 대한 정보를 표시합니다.

```
# 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 프로빙합니다.

-u UDP를 사용하는 RPC 프로그램을 프로빙합니다.

transport 서비스용 전송 또는 프로토콜을 지정합니다.

hostname 서버의 호스트 이름을 지정합니다.

progname RPC 프로그램의 이름을 지정합니다.

사용 가능한 옵션에 대한 자세한 내용은 [rpcinfo\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

*hostname*에 대해 값을 지정하지 않으면 로컬 호스트 이름이 사용됩니다. *progname*을 RPC 프로그램 번호로 대체할 수는 있지만 일반적으로 이름이 더 많이 사용됩니다. 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

예 6-3 RPC 서비스 정보 표시

```
# 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                                    smsrvertd   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
```

이 예에서는 서버에서 실행 중인 RPC 서비스에 대한 정보를 보여줍니다. 명령에 의해 생성되는 출력은 정보를 보다 쉽게 읽을 수 있도록 프로그램 번호별로 sort 명령을 통해 필터링됩니다. 예에서는 RPC 서비스가 나열되는 여러 행이 삭제되었습니다.

서버에서 특정 전송을 선택하여 특정 RPC 서비스에 대한 정보를 수집할 수 있습니다. 다음 예에서는 TCP를 통해 실행 중인 mountd 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
```

다음 예에서는 UDP를 통해 실행 중인 NFS 서비스를 확인합니다.

```
# rpcinfo -u bee nfs
program 100003 Version 2 ready and waiting
program 100003 Version 3 ready and waiting
```

snoop 명령

snoop 명령은 네트워크에서 패킷을 모니터링하는 데 사용됩니다. snoop 명령은 root 사용자로 실행해야 합니다. 이 명령을 사용하면 클라이언트와 서버에서 네트워크 하드웨어가 작동하는지를 효율적으로 확인할 수 있습니다.

다음 명령 구문을 사용하여 네트워크에서 패킷을 모니터링합니다.

```
# 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 `grep`와 결합하여 사용하면 유용하게 활용 가능하도록 구체적인 데이터가 생성되는 경우가 많습니다. 사용 가능한 옵션에 대한 자세한 내용은 [snoop\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

문제를 해결할 때는 패킷이 적절한 호스트로 들어가고 적절한 호스트에서 나오는지 확인하십시오. 또한 오류 메시지도 확인하십시오. 파일에 패킷을 저장하면 데이터를 간편하게 검토할 수 있습니다.

truss 명령

`truss` 명령을 사용하여 프로세스가 정지되었는지 확인합니다. `truss` 명령은 프로세스 소유자가 실행하거나 `root`로 실행해야 합니다.

다음 명령 구문을 사용하여 프로세스가 정지되었는지 확인합니다.

```
# truss [ -t syscall ] -p pid
```

`-t syscall` 추적할 시스템 호출을 선택합니다.

`-p pid` 추적할 프로세스의 PID를 나타냅니다.

`syscall`은 추적할 시스템의 심프로로 구분된 목록입니다. ! 문자가 포함된 목록을 시작하면 나열된 시스템 호출이 추적에서 제외됩니다. 사용 가능한 옵션에 대한 자세한 내용은 [truss\(1\)](#) 매뉴얼 페이지를 참조하십시오.

예 6-4 프로세스 상태 표시

```
# /usr/bin/truss -p 243
poll(0x00024D50, 2, -1) (sleeping...)
```

이 예에서는 프로세스에서 다른 연결 요청(정상 응답)을 대기하고 있음을 보여줍니다. 새 연결 요청을 수행한 후에도 응답이 변경되지 않으면 프로세스가 정지될 수 있습니다.

NFS 서비스 다시 시작에 대한 자세한 내용은 [NFS 서비스를 다시 시작하는 방법 \[130\]](#)을 참조하십시오. 정지된 프로세스 문제 해결에 대한 자세한 내용은 [“NFS 문제 해결 절차” \[126\]](#)를 참조하십시오.

NFS 문제 해결 절차

NFS 서비스에 오류가 발생했는지 확인하려면 몇 가지 절차를 수행하여 오류를 격리시켜야 합니다. 다음을 확인합니다.

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

이 프로세스 중에 네트워크의 다른 부분이 작동하지 않음을 확인할 수도 있습니다. 예를 들어 이름 서비스 또는 물리적 네트워크 하드웨어가 작동하지 않을 수 있습니다. 이름 지정 서비스에 대한 자세한 내용은 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)를 참조하십시오. 또한 프로세스 중에 문제가 클라이언트측에 있지 않음을 확인할 수 있습니다(예: 작업 영역의 모든 서브넷에서 문제가 수신된 경우). 이 경우에는 문제가 서버 또는 서버 근처의 네트워크 하드웨어라고 가정하고 클라이언트가 아니라 서버에서 디버깅을 시작합니다.

▼ NFS 클라이언트에서 연결을 확인하는 방법

1. 클라이언트에서 NFS 서버에 연결할 수 있는지 확인합니다.

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

명령에서 서버가 활성 상태임이 보고되면 NFS 서버를 원격으로 확인합니다. 원격으로 NFS 서버를 확인하는 방법에 대한 자세한 내용은 [원격으로 NFS 서버를 확인하는 방법 \[127\]](#)을 참조하십시오.

2. 클라이언트에서 서버에 연결할 수 없는 경우에는 로컬 이름 서비스가 클라이언트에서 실행 중인지 확인합니다.

예를 들면 다음과 같습니다.

- NIS 이름 서비스를 사용 중인 경우 ypbind 데몬이 실행 중인지 확인합니다. 자세한 내용은 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)의 [“ypbind가 클라이언트에서 실행되고 있지 않음”](#)을 참조하십시오.
- LDAP 이름 서비스를 사용 중인 경우 ldap_cachemgr 데몬이 실행 중인지 확인합니다. 자세한 내용은 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: LDAP”](#)의 [“LDAP 클라이언트 상태 모니터링”](#)을 참조하십시오.

- 이름 서비스가 실행 중인 경우 클라이언트가 올바른 호스트 정보를 받았는지 확인합니다.

```
# /usr/bin/getent hosts system
```

예를 들면 다음과 같습니다.

```
# /usr/bin/getent hosts bee
192.168.83.117 bee.eng.example.com
```

- 호스트 정보가 정확하데 서버가 클라이언트에 연결할 수 없는 경우에는 다른 클라이언트에서 ping 명령을 실행합니다.

두번째 클라이언트에서 실행한 명령이 실패하면 NFS 서비스가 서버에서 사용으로 설정되어 있는지 확인합니다. 자세한 내용은 [서버에서 NFS 서비스를 확인하는 방법 \[128\]](#)을 참조하십시오.

- 서버가 두번째 클라이언트에서 연결할 수 있는 경우에는 ping을 사용하여 첫번째 클라이언트가 로컬 네트워크의 다른 시스템에 연결할 수 있는지 확인합니다.

ping 명령이 실패하면 클라이언트에서 네트워킹 소프트웨어 구성을 확인합니다(예: /etc/netmasks 및 svc:/system/name-service/switch 서비스와 연결된 등록 정보).

- (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 3-3. "NFS 서비스 설정"](#)을 참조하십시오.

- 소프트웨어가 올바른 경우에는 네트워킹 하드웨어를 확인합니다.

클라이언트를 다른 물리적 네트워크 연결로 이동해 보십시오.

▼ 원격으로 NFS 서버를 확인하는 방법

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

- NFS 데몬이 NFS 서버에서 시작되었는지 확인합니다.

```
# rpcinfo -s server-name | egrep 'nfs|mountd'
```

예를 들면 다음과 같습니다.

```
# 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 않았으면 NFS 서비스를 다시 시작합니다. 자세한 내용은 [NFS 서비스를 다시 시작하는 방법 \[130\]](#)을 참조하십시오.

2. 클라이언트에서 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
```

참고 - NFS 버전 4에서는 UDP를 지원하지 않습니다.

서버가 실행 중인 경우 rpcinfo 명령은 UDP 프로토콜과 연결된 버전 번호 및 프로그램을 나열합니다. -t 옵션을 rpcinfo 명령과 함께 사용하여 TCP 연결을 확인합니다. rpcinfo 명령이 실패하면 NFS 서비스가 서버에서 사용으로 설정되어 있는지 확인합니다. 자세한 내용은 [서버에서 NFS 서비스를 확인하는 방법 \[128\]](#)을 참조하십시오.

3. 서버의 mountd 데몬이 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 경우 rpcinfo 명령은 UDP 프로토콜과 연결된 버전 번호 및 프로그램을 나열합니다. -t 옵션을 사용하여 TCP 연결을 테스트합니다. nfsd 및 mountd 데몬이 실행 중인지 확인합니다.

4. 제대로 작동하는 /net 또는 /home 마운트 지점으로 변경하여 로컬 autofs 서비스가 클라이언트에서 사용되는지 확인합니다.

```
# cd /net/eng
```

이 명령이 실패하면 클라이언트에서 root 사용자로 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 이름 공간도 확인합니다. 이 예에서 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 파일과 모든 /etc/auto_* 파일이 포함됩니다.

▼ 서버에서 NFS 서비스를 확인하는 방법

1. 관리자가 됩니다.

자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.

2. 서버가 클라이언트에 연결할 수 있는지 확인합니다.

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

3. 서버에서 클라이언트에 연결할 수 없는 경우에는 로컬 이름 서비스가 클라이언트에서 실행 중인지 확인합니다.
4. 이름 서비스가 실행 중인 경우 서버에서 네트워킹 소프트웨어 구성을 확인합니다. 예를 들어 /etc/netmasks 및 svc:/system/name-service/switch 서비스와 연결된 등록 정보를 확인합니다.
5. rpcbind 데몬이 서버에서 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 rpcinfo 명령은 UDP 프로토콜에 연결된 버전 번호 및 프로그램을 나열합니다.

6. nfsd 데몬이 서버에서 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 101328      0  0   Jul 12 ?          303:25 nfsd_kproc
root 101327      1  0   Jul 12 ?          2:54 /usr/lib/nfs/nfsd
root 263149 131084  0 13:59:19 pts/17      0:00 grep nfsd
```

참고 - NFS 버전 4에서는 UDP를 지원하지 않습니다.

서버가 실행 중인 경우 rpcinfo 명령은 UDP 프로토콜과 연결된 버전 번호 및 프로그램을 나열합니다. 또한 rpcinfo에서 -t 옵션을 사용하여 TCP 연결을 확인합니다. 이러한 명령이 실패하면 NFS 서비스를 다시 시작합니다. 자세한 내용은 NFS 서비스를 다시 시작하는 방법 [130]을 참조하십시오.

7. mountd 데몬이 서버에서 실행 중인지 확인합니다.

```
# /usr/bin/rpcinfo -t 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 경우 rpcinfo 명령은 UDP 프로토콜과 연결된 버전 번호 및 프로그램을 나열합니다. 또한 rpcinfo에서 -t 옵션을 사용하여 TCP 연결을 확인합니다. 이러한 명령이 실패하면 NFS 서비스를 다시 시작합니다. 자세한 내용은 [NFS 서비스를 다시 시작하는 방법 \[130\]](#)을 참조하십시오.

▼ NFS 서비스를 다시 시작하는 방법

1. 관리자가 됩니다.
자세한 내용은 “Oracle Solaris 11.2의 사용자 및 프로세스 보안”의 “지정된 관리 권한 사용”을 참조하십시오.
2. 서버에서 NFS 서비스를 다시 시작합니다.

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

NFS 서비스를 제공하는 호스트 식별

nfsstat 명령을 -m 옵션과 함께 사용하여 현재 NFS 정보를 표시합니다. 현재 서버의 이름은 “currserver=” 다음에 출력됩니다.

예를 들면 다음과 같습니다.

```
# 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,wsz=32678,retrans=5
Failover: noresponse=0, failover=0, remap=0, currserver=bee
```

▼ mount 명령에 사용되는 옵션을 확인하는 방법

mount 명령과 함께 제공되는 잘못된 옵션에 대한 경고는 표시되지 않습니다. 이 절차를 수행하면 명령줄에서 또는 /etc/vfstab 파일을 통해 제공된 옵션이 유효한지를 확인할 수 있습니다.

예를 들어 이 절차에서는 다음 명령이 실행되었다고 가정합니다.

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

1. 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
```

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

2. `/etc/mnttab` 파일의 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 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.

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
/blat      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: 도움이 필요하면 My Oracle Support에 문의하십시오. 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: 도움이 필요하면 My Oracle Support에 문의하십시오. 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: 도움이 필요하면 My Oracle Support에 문의하십시오. 이 오류 메시지는 거의 표시되지 않으며 직접 해결할 수 있는 방법도 없습니다.

mountpoint: Not a directory

설명: *mountpoint*는 디렉토리가 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: 도움이 필요하면 My Oracle Support에 문의하십시오. 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: 도움이 필요하면 My Oracle Support에 문의하십시오. 이 오류 메시지는 거의 표시되지 않으며 직접 해결할 수 있는 방법도 없습니다.

nfscast: select: *reason*

설명: 이 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: 도움이 필요하면 My Oracle Support에 문의하십시오. 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: 구성 가능한 경로 이름에 대한 자세한 내용은 [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 재설정합니다.

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

NFS 오류 메시지

이 절에는 오류 메시지와 오류를 유발할 수 있는 상황에 대한 설명, 그리고 가능한 해결 방법이 나와 있습니다.

index 옵션에 대해 잘못된 인수가 지정됨 - 파일이어야 함

해결책: index 옵션을 사용하여 파일 이름을 포함해야 합니다. 디렉토리 이름은 사용할 수 없습니다.

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

설명: 이 메시지는 이름 공간의 서비스 정보가 업데이트되지 않았을 때 생성됩니다. 또한 UDP에 대해서도 이 메시지가 보고될 수 있습니다.

해결책: 이 문제를 해결하려면 이름 공간에서 서비스 데이터를 업데이트해야 합니다.

NIS 및 /etc/services의 경우 항목은 다음과 같아야 합니다.

```
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: 도움이 필요하면 My Oracle Support에 문의하십시오. 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: 공용 파일 핸들을 사용하지 않고 마운트 요청을 시도하거나, `public` 파일 핸들을 지원하도록 NFS 서버를 재구성하십시오.

`daemon` running already with pid `pid`

설명: 데몬이 이미 실행 중입니다.

해결책: 새 프로세스를 실행하려면 현재 버전을 종료하고 새 버전을 시작합니다.

error locking `lock-file`

설명: 이 메시지는 데몬과 연결된 `lock-file`을 제대로 잠글 수 없을 때 표시됩니다.

해결책: 도움이 필요하면 My Oracle Support에 문의하십시오. 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 제대로 열 수 없을 때 표시됩니다.

해결책: 도움이 필요하면 My Oracle Support에 문의하십시오. 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: 필요한 작업이 없습니다.

`filename`: File too large

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

해결책: 가능하면 NFS 버전 2를 사용하지 마십시오. NFS 버전 3 또는 NFS 버전 4를 사용하여 파일 시스템을 마운트합니다. 또한 `mount(1M)` 매뉴얼 페이지의 `noLargefiles` 옵션에 대한 설명을 참조하십시오.

mount: ... server not responding:RPC_PMAP_FAILURE - RPC_TIMED_OUT

설명: 마운트하려는 파일 시스템을 공유하는 서버가 다운된 상태이거나, 연결할 수 없거나, 실행 레벨이 잘못되었거나, 해당 `rpcbind` 프로세스가 사용 불가능 상태이거나 정지되었습니다.

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

mount: ... server not responding: RPC_PROG_NOT_REGISTERED

설명: 마운트 요청은 rpcbind 프로세스로 등록되었는데 NFS 마운트 데몬 mountd는 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 ls를 실행합니다.

mount: ...: Permission denied

설명: 마운트를 시도한 파일 시스템에 액세스하도록 허용되는 클라이언트 또는 넷 그룹 목록에 시스템 이름이 없을 수 있습니다.

해결책: showmount -e 명령을 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 [“NFS 버전 4의 위임” \[30\]](#)을 참조하십시오.

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

설명: 이 오류는 다양한 상황에서 발생할 수 있습니다. 디버그하기 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, ACL(액세스 제어 목록)을 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 NFS 버전 3 또는 NFS 버전 4를 사용해야 합니다.

NFS server *hostname* not responding still trying

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

해결책: NFS 페일오버를 사용 중인 경우 *hostname*은 서버 목록입니다. [NFS 클라이언트에서 연결을 확인하는 방법 \[126\]](#)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: 필요한 작업이 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 ACL(액세스 제어 목록) 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.

NFSdml ACL 항목에 대한 자세한 내용은 ["NFS 버전 4의 ACL 및 nfsmapid" \[31\]](#)를 참조하십시오.

해결책: 다음을 수행합니다.

- ACL 항목의 모든 사용자 ID 및 그룹 ID가 클라이언트와 서버에 모두 있는지 확인합니다.

- SMF 저장소에서 `nfsmapid_domain` 등록 정보의 값이 올바르게 설정되어 있는지 확인합니다.

사용자 또는 그룹을 서버나 클라이언트에서 매핑할 수 없는지 확인하는 데 사용되는 스크립트에 대한 자세한 내용은 “[매핑되지 않은 사용자 ID 또는 그룹 ID 확인](#)” [32]을 참조하십시오.

`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 실행하지 마십시오.

`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 NFS 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: `share` 명령을 사용하여 `-public` 옵션을 통해 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 최신 OS 버전으로 서버를 업그레이드하십시오.

◆◆◆ 7 장

네트워크 파일 시스템 액세스

이 장에서는 NFS 서비스를 지원하는 파일 및 데몬 목록을 제공합니다.

이 장의 내용:

- “NFS 파일” [141]
- “NFS 데몬” [144]

참고 - 시스템에서 영역이 사용으로 설정된 경우 비전역 영역에서 이 기능을 사용하려면 “Oracle Solaris 영역 소개”를 참조하십시오.

NFS 파일

시스템에서 NFS 작업을 지원하려면 여러 파일이 필요합니다. 이러한 파일은 대부분 ASCII 형식이지만 일부는 데이터 파일입니다. 다음 표에서는 NFS 파일과 그 기능을 나열합니다.

표 7-1 NFS 파일

파일 이름	기능	매뉴얼 페이지
/etc/default/fs	로컬 파일 시스템의 기본 파일 시스템 유형을 지정합니다. /kernel/fs의 파일을 확인하여 클라이언트나 서버에서 지원되는 파일 시스템 유형을 파악할 수 있습니다.	fs(4) 매뉴얼 페이지.
/etc/default/nfslogd	NFS 서버 로깅 데몬 nfslogd에 대한 구성 정보를 지정합니다.	nfslogd(1M) 매뉴얼 페이지.
/etc/dfs/dfstab	사용되지 않음: 공유할 로컬 리소스를 지정합니다.	dfstab(4) 매뉴얼 페이지.
/etc/dfs/fstypes	원격 파일 시스템의 기본 파일 시스템 유형을 지정합니다. 첫번째 항목은 NFS 파일 시스템 유형을 기본값으로 정의합니다.	fstypes(4) 매뉴얼 페이지.
/etc/dfs/sharetab	공유할 로컬 및 원격 리소스를 지정합니다. 이 파일을 편집하지 마십시오.	sharetab(4) 매뉴얼 페이지.
/etc/mnttab	현재 마운트되어 있는 파일 시스템(자동 마운트된 디렉토리 포함)을 지정합니다. 이 파일을 편집하지 마십시오.	mnttab(4) 매뉴얼 페이지.
/etc/netconfig	전송 프로토콜을 지정합니다. 이 파일을 편집하지 마십시오.	netconfig(4) 매뉴얼 페이지.

파일 이름	기능	매뉴얼 페이지
/etc/nfs/nfslog.conf	NFS 서버 로깅에 대한 일반 구성 정보를 지정합니다.	nfslog.conf(4) 매뉴얼 페이지.
/etc/nfs/nfslogtab	nfslogd 데몬에 의한 로그 사후 처리 정보를 지정합니다. 이 파일을 편집하지 마십시오.	
/etc/nfssec.conf	NFS 보안 서비스를 지정합니다.	nfssec.conf(4) 매뉴얼 페이지.
/etc/rmtab	NFS 클라이언트에 의해 원격으로 마운트된 파일 시스템을 지정합니다. 이 파일을 편집하지 마십시오.	rmtab(4) 매뉴얼 페이지.
/etc/vfstab	로컬로 마운트할 파일 시스템을 정의합니다.	vfstab(4) 매뉴얼 페이지.

/etc/default/nfslogd 파일

이 파일은 NFS 서버 로깅 사용 시 사용되는 일부 매개변수를 정의합니다.

참고 - NFS 버전 4에서는 NFS 서버 로깅이 지원되지 않습니다.

다음 매개변수를 정의할 수 있습니다.

CYCLE_FREQUENCY

로그 파일을 순환하기 전에 경과해야 하는 시간을 결정합니다. 기본값은 24시간입니다. 이 옵션을 사용하여 로그 파일이 너무 커지지 않도록 방지합니다.

IDLE_TIME

버퍼 파일에서 추가 정보를 확인하기 전에 nfslogd가 일시 정지 상태로 유지되는 시간(초)을 설정합니다. 이 매개변수는 구성 파일 확인 빈도도 결정합니다. 이 매개변수는 MIN_PROCESSING_SIZE와 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_TIME과 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

nfslogd에서 만드는 로그 파일의 파일 모드 생성 마스크를 지정합니다. 기본값은 0137입니다.

/etc/nfs/nfslog.conf 파일

이 파일은 nfslogd에서 사용할 경로, 파일 이름 및 로깅 유형을 정의합니다. 각 정의는 태그와 연관됩니다. NFS 서버 로깅을 시작하려면 각 파일 시스템에 대해 tag를 식별해야 합니다. 전역 태그는 기본값을 정의합니다.

참고 - NFS 버전 4에서는 NFS 서버 로깅이 지원되지 않습니다.

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/nfs입니다.

log=*path/filename*

로그 파일의 경로 및 파일 이름을 설정합니다. 기본값은 /var/nfs/nfslog입니다.

fhtable=*path/filename*

파일 핸들-경로 데이터베이스 파일의 경로 및 파일 이름을 선택합니다. 기본값은 /var/nfs/fhtable입니다.

buffer=*path/filename*

버퍼 파일의 경로 및 파일 이름을 결정합니다. 기본값은 /var/nfs/nfslog_workbuffer입니다.

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 defaultdir에 의해 정의되는 경로가 사용됩니다. 절대 경로를 사용하여 defaultdir을 대체할 수도 있습니다.

파일을 보다 쉽게 식별하려면 별도의 디렉토리에 파일을 저장하십시오. 다음 예에서는 필요한 변경 사항을 보여줍니다.

예 7-1 샘플 NFS 서버 로깅 구성 파일

```
# 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 예에서는 log=publicftp와 공유되는 파일 시스템을 보여줍니다. log=publicftp와 공유되는 파일 시스템에서는 다음 값을 사용합니다.

- 기본 디렉토리는 /var/nfs입니다.
- 로그 파일은 /var/nfs/logs/nfslog*에 저장됩니다.
- 파일 핸들-경로 데이터베이스 테이블은 /var/nfs/fh/fhtables에 저장됩니다.
- 버퍼 파일은 /var/nfs/buffers/workbuffer에 저장됩니다.

NFS 서버 로깅을 사용으로 설정하는 방법에 대한 자세한 내용은 [NFS 서버 로깅을 사용으로 설정하는 방법 \[65\]](#)을 참조하십시오.

NFS 데몬

시스템이 실행 레벨 또는 다중 사용자 모드로 진입하면 NFS 작업을 지원하기 위해 여러 데몬이 시작됩니다. mountd 및 nfsd 데몬은 서버 시스템에서 실행됩니다. 서버 데몬의 자동 시작은 최소 하나의 NFS가 존재하는지 여부에 달려 있습니다. 현재 NFS 공유 목록을 표시하려면 share -F nfs 명령을 실행합니다. NFS 파일 잠금을 지원하기 위해 lockd 및 statd 데몬이 NFS 클라이언트 및 서버에서 실행됩니다. 그러나 이전 NFS 버전과는 달리 NFS 버전 4에서는 lockd, statd 및 nfslogd 데몬이 사용되지 않습니다.

이 절에서는 다음 데몬에 대해 설명합니다.

- “automountd 데몬” [145]
- “lockd 데몬” [146]
- “mountd 데몬” [146]
- “nfs4cbd 데몬” [147]
- “nfsd 데몬” [147]
- “nfslogd 데몬” [148]
- “nfsmapid 데몬” [148]
- “reparsed 데몬” [154]

■ “statd 데몬” [154]

automountd 데몬

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

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

설명

- T 추적을 사용으로 설정합니다.
- n 모든 autofs 노드에서 찾아보기를 사용 안함으로 설정합니다.
- v 모든 상태 메시지를 콘솔에 로깅합니다.
- D name=value name으로 표시되는 자동 마운트 맵 변수를 value로 대체합니다.

자동 마운트 맵의 기본값은 /etc/auto_master입니다. 문제 해결 시에는 -T 옵션을 사용합니다.

sharectl 명령을 사용하여 명령줄에서 지정한 것과 같게 지정할 수 있습니다. 그러나 명령줄 옵션과는 달리 SMF 저장소에서는 서비스 다시 시작, 시스템 재부트 및 시스템 업그레이드 시에도 지정 사항이 보존됩니다. automountd 데몬에 대해 다음 매개변수를 설정할 수 있습니다.

automountd_verbose

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

nobrowse

모든 autofs 마운트 지점에 대해 찾아보기를 설정하거나 해제하며, automountd용 -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

lockd 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. sharectl 명령을 사용하여 매개변수를 설정하거나 명령줄을 통해 이러한 옵션을 설정할 수 있습니다. 자세한 내용은 [lockd\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

참고 - LOCKD_GRACE_PERIOD 키워드 및 -g 옵션은 더 이상 사용되지 않습니다. 사라진 키워드는 새 grace_period 매개변수로 교체되었습니다. 두 키워드가 모두 설정된 경우 grace_period의 값이 LOCKD_GRACE_PERIOD의 값을 대체합니다.

LOCKD_GRACE_PERIOD와 마찬가지로, grace_period=*graceperiod* 매개변수는 서버가 재부트된 후 클라이언트가 NLM에서 제공하는 NFS 버전 3 잠금과 NFS 버전 4 잠금을 모두 재생 이용해야 하는 시간(초)을 설정합니다.

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 5초입니다. *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 통해 데몬을 시작하여 명령줄에서도 같은 매개변수를 사용할 수 있습니다.

lockd_servers=*number* 매개변수는 최대 동시 lockd 요청 수를 지정합니다. 기본값은 1024입니다.

nthreads 매개변수는 서버에서 처리할 수 있는 최대 동시 스레드 수를 지정합니다. 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

mountd 데몬은 원격 시스템으로부터의 파일 시스템 마운트 요청을 처리하며 액세스 제어 기능을 제공합니다. mountd 데몬은 /etc/dfs/sharetab를 확인하여 원격 마운트에 사용할 수 있는 파일 시스템 및 원격 마운트 수행이 허용되는 시스템을 결정합니다. 자세한 내용은 [mountd\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

- v 상세 정보 표시 모드에서 명령을 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 클라이언트로부터의 모든 이후 마운트 요청을 거부합니다. 이 옵션은 이미 파일 시스템이 마운트된 클라이언트에는 영향을 주지 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. “[NFS 서비스 설정](#)” [72]을 참조하십시오.

`client_versmax`

NFS 클라이언트에서 사용할 최대 NFS 프로토콜 버전을 설정합니다. 기본값은 4입니다. 2와 3도 값으로 사용할 수 있습니다. “[NFS 서비스 설정](#)” [72]을 참조하십시오.

nfs4cbd 데몬

NFS 버전 4 클라이언트 전용으로 사용되는 `nfs4cbd` 데몬은 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

`nfsd` 데몬은 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` NFS 및 TCP에 대한 연결 지향 전송의 연결 대기열 길이를 설정합니다. 기본값은 32개 항목입니다. `-l` 옵션을 포함해 `nfsd`를 시작하여 명령줄에서도 같은 항목을 선택할 수 있습니다.

`max_connections=#-conn` 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` 서버에서 처리할 수 있는 최대 동시 요청 수를 선택합니다. `nservers`의 기본값은 1024입니다. `nservers` 옵션을 포함하여 `nfsd`를 시작하면 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.

다음 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. [“NFS 서비스 설정” \[72\]](#)을 참조하십시오.

`server_versmax`

서버에서 등록 및 제공할 최대 NFS 프로토콜 버전을 설정합니다. 기본값은 4입니다. 2와 3도 값으로 사용할 수 있습니다. [“NFS 서비스 설정” \[72\]](#)을 참조하십시오.

`server_delegation`

CNFS 버전 4 위임 기능이 서버에 대해 사용으로 설정되는지를 제어합니다. 이 기능이 사용으로 설정된 경우 서버는 NFS 버전 4 클라이언트에 위임 제공을 시도합니다. 기본적으로 서버 위임은 사용으로 설정됩니다. 서버 위임을 사용 안함으로 설정하려면 [서버에서 다른 NFS 버전을 선택하는 방법 \[73\]](#)을 참조하십시오. 자세한 내용은 [“NFS 버전 4의 위임” \[30\]](#)을 참조하십시오.

nfslogd 데몬

참고 - NFS 버전 4에서는 이 데몬이 사용되지 않습니다.

`nfslogd` 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 다음 `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.

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. 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.

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)를 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 4의 경우 파일 속성을 수정하는 작업이 실패합니다.
- 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.

`sharectl` 명령을 `nfsmapid_domain` 옵션과 함께 사용하여 클라이언트 및 서버의 도메인 이름을 변경할 수 있습니다. 이 옵션은 클라이언트 및 서버에 대해 공통 도메인을 설정합니다. 로컬 DNS 도메인 이름을 사용하는 기본 동작을 대체합니다. 작업 정보는 [“NFS 서비스 설정” \[72\]](#)을 참조하십시오.

구성 파일 및 `nfsmapid` 데몬

`nfsmapid` 데몬은 다음과 같이 `svc:system/name-service/switch` 및 `svc:/network/dns/client`에 있는 SMF 구성 정보를 사용합니다.

- `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 RR(리소스 레코드)인 `_nfsv4idmapdomain`의 구성을 사용합니다. `_nfsv4idmapdomain` 리소스 레코드 구성에 대한 자세한 내용은 [“nfsmapid 및 DNS TXT 레코드” \[151\]](#)를 참조하십시오. 또한 다음 사항을 확인하십시오.
 - 원하는 도메인 정보를 사용하여 DNS 서버에서 DNS TXT RR을 명시적으로 구성해야 합니다.
 - `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.
- [“우선 순위 규칙” \[150\]](#)
 - [“NFS 버전 4 기본 도메인 구성” \[153\]](#)
 - [`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 [“NFS 서비스 설정” \[72\]](#)을 참조하십시오.

참고 - `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 `resolver`의 루틴 세트에 사용되는 `/etc/resolv.conf` 파일의 지시어를 사용합니다. `resolver`는 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.

다음 예에서는 `domain` 및 `search` 지시어가 둘 다 사용됩니다. `nfsmapid` 데몬은 `search` 지시어 뒤에 나열된 첫번째 도메인(`example.com`)을 사용합니다.

```
domain company.example.com
search example.com abc.def.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 저장소 및 배포 방식을 제공합니다. 또한, DNS는 기본적으로 확장이 가능하므로 대규모 배치에서는 NFS 버전 4 도메인 이름을 구성할 때 DNS TXT 리소스 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.

다음 예에서는 DNS 서버가 NFS 버전 4 도메인 이름을 제공할 수 있도록 설정하는 데 기본적으로 사용되는 항목을 보여줍니다.

```
_nfsv4idmapdomain IN TXT "abc.def"
```

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 이름 공간 레벨에서 레코드를 다음과 같이 읽을 수 있습니다.

```
_nfsv4idmapdomain.subnet.example.com. IN TXT "abc.def"
_nfsv4idmapdomain.example.com. IN TXT "abc.def"
```

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 RR(리소스 레코드)을 정의할 수 있습니다. 이 기능을 통해 더 낮은 레벨의 DNS 하위 도메인이 최상위 레벨 DNS 도메인에서 정의한 TXT 레코드를 대체할 수 있습니다.

참고 - 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.

다음 예에서는 `nslookup` 명령의 샘플 출력을 보여줍니다.

```
# nslookup -q=txt _nfsv4idmapdomain
Server:      10.255.255.255
Address:     10.255.255.255#53

_nfsv4idmapdomain.company.example.com text = "example.com"
```

다음 예에서는 `dig` 명령의 샘플 출력을 보여줍니다.

```
# dig +domain=company.example.com -t TXT _nfsv4idmapdomain
...
;; QUESTION SECTION:
;_nfsv4idmapdomain.company.example.com. IN      TXT

;; ANSWER SECTION:
_nfsv4idmapdomain.company.example.com. 21600 IN TXT    "example.com"

;; AUTHORITY SECTION:
...
```

DNS TXT RR을 설정하는 방법에 대한 자세한 내용은 [“nfsmapid 및 DNS TXT 레코드” \[151\]](#)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 company.example.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
example.com
```

- NIS 등의 다른 이름 지정 서비스를 사용하는 경우 다음 명령을 사용하여 네트워크에 대해 구성된 이름 지정 서비스의 도메인을 식별합니다.

```
# domainname
it.company.example.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 “[Oracle Solaris 11 릴리스에서 NFS 버전 4 기본 도메인 구성](#)” [153]을 참조하십시오.
- 초기 Solaris 10 릴리스의 경우에는 “[Solaris 10 릴리스에서 NFS 버전 4 기본 도메인 구성](#)” [153]을 참조하십시오.

Oracle Solaris 11 릴리스에서 NFS 버전 4 기본 도메인 구성

Oracle Solaris 11 릴리스에서는 다음 명령을 입력하여 기본 NFS 도메인 버전을 설정합니다.

```
# sharectl set -p nfsmapid_domain=example.com nfs
```

참고 - DNS는 기본적으로 다양한 용도로 사용되며 확장이 가능하므로, 대규모 NFS 버전 4 배치의 도메인을 구성할 때는 DNS TXT 레코드가 계속 사용되고 있으며 사용하는 것이 좋습니다. “[nfsmapid 및 DNS TXT 레코드](#)” [151]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 이름 공간은 하나뿐이면 모든 클라이언트는 `nfsmapid_domain`에 대해 하나의 값을 사용해야 했습니다. DNS를 사용하는 사이트의 경우 `nfsmapid`는 `_nfsv4idmapdomain`에 지정된 값에서 도메인 이름을 가져오으로써 이 문제를 해결합니다. 자세한 내용은 “[nfsmapid 및 DNS TXT 레코드](#)” [151]를 참조하십시오. 네트워크가 DNS를 사용하도록 구성되어 있지 않은 경우 첫번째 시스템 부트 중에 운영 체제에서 `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 다음 메시지가 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 []:

참고 - nfsmapid_domain의 값이 SMF 저장소에 있는 경우 사용자가 제공하는 도메인 이름이 해당 값을 대체합니다.

nfsmapid 관련 추가 정보

nfsmapid에 대한 자세한 내용은 다음을 참조하십시오.

- [nfsmapid\(1M\) 매뉴얼 페이지](#)
- [nfs\(4\) 매뉴얼 페이지](#)
- <http://www.ietf.org/rfc/rfc1464.txt>
- [“NFS 버전 4의 ACL 및 nfsmapid” \[31\]](#)

reparsed 데몬

reparsed 데몬은 구문 재분석 지점과 연결된 데이터를 해석합니다. 이 지점은 SMB 및 NFS 파일 서버의 DFS 및 NFS 참조에서 사용됩니다. 이 서비스는 SMF에서 관리하며 수동으로 시작해서는 안 됩니다.

statd 데몬

참고 - NFS 버전 4에서는 이 데몬이 사용되지 않습니다.

statd 데몬은 lockd와 함께 작동하여 잠금 관리자에 대해 충돌 및 복구 기능을 제공합니다. statd 데몬은 NFS 서버에 대한 잠금을 보유한 클라이언트를 추적합니다. 서버가 충돌하면 재부트 시 서버의 statd가 클라이언트의 statd에 연결합니다. 그러면 클라이언트 statd는 서

버에서 잠금 재생 이용을 시도할 수 있습니다. 또한 클라이언트가 충돌하면 서버의 클라이언트 잠금을 지울 수 있도록 클라이언트 `statd`가 서버 `statd`에 알림을 보냅니다. 이 데몬에는 옵션이 없습니다. 자세한 내용은 [statd\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

색인

번호와 기호

- (대시)
 - autofs 맵 이름, 57
 - * (별표)
 - autofs 맵, 62
 - / (슬래시)
 - as 마스터 맵 마운트 지점, 46, 49
 - 루트 디렉토리
 - 디스크가 없는 클라이언트를 통한 마운트, 12
 - 마스터 맵 이름 앞에 붙임, 47
 - & (앰퍼센드)
 - autofs 맵, 61
 - # (파운드 기호)
 - 간접 맵의 주석, 50
 - 마스터 맵(auto_master)의 주석, 47
 - 직접 맵의 주석, 49
 - + (더하기 기호)
 - autofs 맵 이름, 57, 58
 - a 옵션
 - showmount 명령, 117
 - umount 명령, 107
 - ACL(액세스 제어 목록) 및 NFS
 - 설명, 15, 31
 - 오류 메시지, Permission denied, 138
 - already mounted 메시지, 132
 - anon 옵션
 - share 명령, 112
 - ARCH 맵 변수, 57
 - ausofs 맵의 \ (백슬래시), 50
 - auto_home 맵
 - /home 디렉토리, 90
 - /home 디렉토리 서버 설정, 91
 - /home 마운트 지점, 46, 47
 - auto_master 파일
 - nobrowse 옵션, 97
 - autofs
 - /home 디렉토리, 90
 - NFS URL 및, 96
 - nobrowse 옵션, 97
 - 개요, 12
 - 공용 파일 핸들 및, 96
 - 공유 이름 공간 액세스, 93
 - 기능, 19
 - 마운트 프로세스, 52, 53
 - 마운트 해제 프로세스, 53
 - 맵
 - 간접, 50, 51
 - 네트워크 탐색, 46
 - 다른 맵 참조, 57, 58
 - 마스터, 46, 46
 - 변수, 57, 57
 - 유형, 87
 - 읽기 전용 파일 선택, 53, 56
 - 직접, 48, 49
 - 찾아보기 기능 및, 19
 - 탐색 프로세스 시작, 47, 51
 - 맵 관리, 87
 - 메타 문자, 61, 62
 - 문제 해결, 131
 - 비 NFS 파일 시스템 액세스, 89
 - 여러 서버에서 공유 파일 복제, 95
 - 이름 공간 데이터, 19
 - 참조 정보, 61, 62
 - 찾아보기 기능, 19, 96
 - 특수 문자, 62
 - 파일 시스템 마운트, 68
 - 프로젝트 관련 파일 통합, 91
 - 호환되지 않는 클라이언트 OS 버전 지원, 94
 - 홈 디렉토리 서버 설정, 91
- autofs 맵의 \ (백슬래시), 47, 49
- automount 명령, 100
 - autofs 마스터 맵 수정(auto_master), 88

- autofs 및, 12
- v 옵션, 131
- 개요, 44
- 실행해야 하는 경우, 87
- 오류 메시지, 131
- automountd 데몬, 145
 - autofs 및, 12
 - description, 19
 - 개요, 45
 - 마운트 및, 19
- automountd 명령
 - 개요, 44
- bad argument specified with index option, 135
- bad key 메시지, 131
- bg 옵션
 - mount 명령, 102
- cannot receive reply 메시지, 134
- cannot send packet 메시지, 134
- clear_locks 명령, 100
- client_versmax 매개변수, 147
- client_versmin 매개변수, 147
- could not use public filehandle 메시지, 136
- couldn't create mount point 메시지, 132
- CPU 맵 변수, 57
- d 옵션
 - showmount 명령, 117
- daemon running already 메시지, 136
- DH 인증
 - 개요, 41, 41
 - 보안 NFS 및, 76
 - 사용자 인증, 39
 - 암호 보호, 40
- dir must start with '/' 메시지, 133
- DNS 레코드
 - FedFS, 81
- /etc/default/autofs 파일
 - autofs 환경 구성, 86
- /etc/default/nfslogd 파일, 142
- /etc/mnttab 파일
 - auto_master 맵과 비교, 45
- /etc/netconfig 파일, 141
- /etc/nfs/nfslog.conf 파일, 143
- /etc/services 파일
 - nfsd 항목, 135
- /etc/vfstab 파일
 - automount 명령 및, 45
 - NFS 서버 및, 67
 - 디스크가 없는 클라이언트를 통한 마운트, 12
 - 부트 시에 파일 시스템 마운트, 67
 - 클라이언트측 페일오버 사용으로 설정, 69
- e 옵션
 - showmount 명령, 117
- error checking 메시지, 136
- error locking 메시지, 136
- F 옵션
 - unshareall 명령, 116
- FedFS
 - DNS 레코드, 81
 - LDAP 스키마, 82
 - 관리, 81
 - 마운트, 81
 - 마운트 지점, 48
- FedFS 명령, 118
- fg 옵션
 - mount 명령, 102
- file too large 메시지, 136
- forcedirectio 옵션
 - mount 명령, 102
- ftp 아카이브
 - WebNFS 및, 78
- fuser 명령
 - umountall 명령 및, 108
- g 옵션
 - lockd 데몬, 146
- grace_period 매개변수
 - lockd 데몬, 146
- GSS-API
 - 및 NFS, 18
- /home 디렉토리 및 NFS 서버 설정, 91
- /home 마운트 지점, 46, 47
- h 옵션
 - umountall 명령, 108
- hard 옵션
 - mount 명령, 105
- hierarchical mount points 메시지, 133
- HOST 맵 변수, 57
- host not responding 메시지, 133
- HTML 파일
 - WebNFS 및, 78
- httpd 명령
 - 방화벽 액세스 및 WebNFS, 80
- ID 매핑 실패

- 이유, 32
- index 옵션
 - share 명령 사용, 79
 - WebNFS 및, 78
 - 잘못된 인수 오류 메시지, 135
- intr 옵션
 - mount 명령, 119
- /kernel/fs 파일
 - 확인, 141
- k 옵션
 - umountall 명령, 108
- KERB 인증
 - NFS 및, 17
- keylogin 명령
 - 원격 로그인 보안 문제, 42
- keylogout 명령
 - 보안 NFS 및, 42
- l 옵션
 - umountall 명령, 108
- largefiles 옵션
 - mount 명령, 103
 - 오류 메시지, 138
- LDAP 스키마
 - FedFS용, 82
- leading space in map entry 메시지, 132
- lockd 데몬, 146
- LOCKD_GRACE_PERIOD 매개변수
 - lockd 데몬, 146
- lockd_retransmit_timeout 매개변수
 - lockd 데몬, 146
- lockd_servers 매개변수
 - lockd 데몬, 146
- log 옵션
 - share 명령, 113
- login 명령
 - 보안 NFS 및, 42
- ls 명령
 - ACL 항목 및, 32
- map key bad 메시지, 133
- mnttab 파일
 - auto_master 맵과 비교, 45
- mount 명령, 102
 - autofs 및, 12
 - NFS URL, 71, 106
 - 디스크가 없는 클라이언트의 요구, 12
 - 사용, 105
- 옵션
 - public, 70
 - 설명, 102
 - 인수 없음, 107
 - 파일 시스템 수동 마운트, 68
 - 페일오버, 105, 105
- mount of *server:pathname* 오류, 133
- mountall 명령, 108
- mountd 데몬, 146
 - rpcbind로 등록되지 않음, 137
 - 서버의 응답 확인, 128
 - 실행 중인지 확인, 129, 137
- /net 마운트 지점, 47
- /nfs4 마운트 지점, 46, 48
- netconfig 파일
 - 설명, 141
- NFS
 - 데몬, 144
 - 명령, 99
 - 버전 협상, 24
- NFS 관리
 - 관리자 책임, 63, 85
- NFS 문제 해결
 - NFS 서비스가 실패한 위치 확인, 129
 - 서버 문제, 126
 - 원격 마운트 문제, 137
 - 전략, 119
 - 정지된 프로그램, 138
- NFS 버전 4
 - 기능, 24
- NFS 서버
 - autofs 파일 선택, 56
 - 공유 파일 복제, 95
 - 맵의 가중치, 56
 - 문제 해결
 - 문제 해결, 126
 - 원격 마운트 문제, 126, 137
 - 원격 마운트에 데몬 필요, 119
 - 유지 관리, 63, 85
 - 현재 식별, 130
 - NFS 서버 로깅
 - 개요, 19
 - 사용으로 설정, 65
 - NFS 서비스
 - 다시 시작, 130
 - 서버에서 다른 버전 선택, 73

- 작업 맵, 72
- 클라이언트에서 다른 버전 선택
 - mount 명령 사용, 75
 - SMF 등록 정보 변경, 74
- NFS 잠금 및
 - 클라이언트측 페일오버 및, 36
- NFS 참조
 - 개요, 43
 - 만들기, 80, 83
 - 제거, 81
- NFS 클라이언트
 - NFS 서비스, 13
 - 호환되지 않는 운영 체제 지원, 94
- NFS 환경
 - 보안 NFS 시스템, 39
- NFS ACL
 - 설명, 15, 31
 - 오류 메시지, Permission denied, 138
- NFS can't support nolargefiles 메시지, 137
- NFS URL
 - autofs 및, 96
 - mount 명령 예, 106
 - WebNFS 및, 78
 - 구문, 79
 - 마운트, 18
 - 파일 시스템 마운트, 71
- NFS V2 can't support largefiles 메시지, 138
- nfs4cbd 데몬, 147
- NFS의 ACL 관련 문제
 - 방지, 32
- NFS의 ACL 관련 문제 방지, 32
- nfscast: cannot receive reply 메시지, 134
- nfscast: cannot send packet 메시지, 134
- nfscast: select 메시지, 134
- nfsd 데몬, 147
 - 마운트 및, 34
 - 서버의 응답 확인, 128
 - 실행 중인지 확인, 129
- nfslog.conf 파일, 143
- nfslogd 데몬
 - 설명, 148
- nfslogd 파일, 142
- nfsmapid 데몬
 - ACL 및, 31
 - DNS TXT 레코드 및, 151
 - NFSv4 기본 도메인 구성, 153
 - NFSv4 도메인 식별, 152
 - 관련 추가 정보, 154
 - 구성 파일 및, 149
 - 설명, 14, 148
 - 우선 순위 규칙 및, 150
- nfsmapid_domain 매개변수, 149
- NFSMAPID_DOMAIN 키워드, 32
- nfsref 명령
 - 설명, 118
 - 예, 83
- nfsstat 명령, 120, 130
- NIS 이름 서비스
 - autofs 맵 업데이트, 87
- no info 메시지, 134
- No such file or directory 메시지, 137
- nobrowse 매개변수
 - 설정, 96
- nobrowse 옵션
 - auto_master 파일, 97
- nolargefiles 옵션
 - mount 명령, 103
 - 오류 메시지, 137
- nosuid 옵션
 - share 명령, 113
- Not a directory 메시지, 134
- Not found 메시지, 132
- nsdb-list 명령
 - 설명, 118
- nsdb-nces 명령
 - 설명, 118
- nsdb-resolve-fsn 명령
 - 설명, 118
- nsdb-update-nci 명령
 - 설명, 118
 - 예, 82
- nsdbparams 명령
 - 설명, 118
 - 예, 82
- nthreads 옵션
 - lockd 데몬, 146
- o 옵션
 - mount 명령, 105
 - share 명령, 112, 114
- O 옵션
 - mount 명령, 105
- OPEN 공유 지원

- NFS 버전 4, 29
- OSNAME 맵 변수, 57
- OSREL 맵 변수, 57
- OSVERS 맵 변수, 57
- pathconf: no info 메시지, 134
- pathconf: *server* not responding 메시지, 134
- Permission denied 메시지, 137
- pound sign (#)
 - 간접 맵의 주석, 50
- pstack 명령, 122
- public 옵션
 - dfstab 파일, 79
 - mount 명령, 70, 104
 - WebNFS 및, 78
 - 공유 오류 메시지, 139
- r 옵션
 - mount 명령, 105
 - umountall 명령, 108
- remount 메시지, 132
- reparsed 데몬, 154
- replicas must have the same version 메시지, 139
- replicated mounts must be read-only 메시지, 139
- replicated mounts must not be soft 메시지, 139
- rlogin 명령
 - 보안 NFS 및, 42
- ro 옵션
 - mount 명령, 104
 - mount 명령(-o 플래그 포함), 105
 - share 명령, 112, 114
- root 옵션
 - share 명령, 113
- RPC
 - 보안
 - DH 권한 부여 문제, 41, 42
 - 인증, 40
 - RPC(원격 프로시저 호출)
 - 보안
 - 개요, 40
- rpcbind 데몬
 - mountd 데몬이 등록되지 않음, 137
 - 사용 불가능 상태 또는 정지됨, 136
- rpcinfo 명령, 123
- RPCSEC_GSS, 18
- rw 옵션
 - mount 명령, 104
 - share 명령, 112, 114
- rw=client 옵션
 - umountall 명령, 112
- s 옵션
 - umountall 명령, 108
- server not responding 메시지, 132, 134
 - 원격 마운트 문제, 136
 - 정지된 프로그램, 138
- server_delegation 매개변수, 148
- server_versmax 매개변수, 148
- server_versmin 매개변수, 148
- setfacl 명령
 - NFS 및, 31
- setgid 모드
 - share 명령, 113
- setuid 모드
 - share 명령, 113
 - 보안 RPC 및, 42
- share 명령
 - WebNFS 서비스 사용으로 설정, 79
 - 보안 문제, 113
 - 설명, 111
 - 옵션, 112
- shareall 명령, 116
- showmount 명령, 117
 - 예, 71
- showmount_info 등록 정보, 71
- snoop 명령, 124
- soft 옵션
 - mount 명령, 105
- statd 데몬, 154
- t 옵션
 - lockd 데몬, 146
- TCP
 - NFS 버전 3 및, 16
- telnet 명령
 - 보안 NFS 및, 42
- transport setup problem
 - 오류 메시지, 135
- truss 명령, 125
- /usr 디렉토리
 - 디스크가 없는 클라이언트를 통한 마운트, 12
- /usr/kvm 디렉토리
 - 디스크가 없는 클라이언트를 통한 마운트, 12
- /usr/lib/fs/nfs/fedfs-11.schema 파일, 82

- `/usr/sbin/mount` 명령 살펴볼 내용 `mount` 명령
 - `/usr/sbin/nsdb-list` 명령
 - 설명, 118
 - `/usr/sbin/nsdb-nces` 명령
 - 설명, 118
 - `/usr/sbin/nsdb-resolve-fsn` 명령
 - 설명, 118
 - `/usr/sbin/nsdb-update-nci` 명령
 - 설명, 118
 - `/usr/sbin/nsdbparams` 명령
 - 설명, 118
 - `/usr/sbin/showmount` 명령, 117
 - `/usr/sbin/unshareall` 명령, 116
 - UDP
 - NFS 및, 16
 - `umount` 명령
 - `autofs` 및, 12
 - 설명, 107
 - `umountall` 명령, 108
 - UNIX 인증, 39, 40
 - `unshare` 명령, 116
 - `unshareall` 명령, 116
 - URL 서비스 유형
 - WebNFS 및, 80
 - v 옵션
 - `automount` 명령, 131
 - v 옵션
 - `umount` 명령, 107
 - `vfstab` 파일
 - `automount` 명령 및, 45
 - NFS 서버 및, 67
 - 디스크가 없는 클라이언트를 통한 마운트, 12
 - 부트 시에 파일 시스템 마운트, 67
 - 클라이언트측 페일오버 사용으로 설정, 69
 - WARNING: *mountpoint* already mounted on 메시지, 132
 - WebNFS 서비스
 - URL 서비스 유형 및, 80
 - 개요, 18
 - 계획, 78
 - 방화벽 및, 80
 - 보안 협상 및, 18
 - 사용으로 설정, 79
 - 설명, 37
 - 작업 맵, 77
 - 찾아보기, 79
- ㄱ
- 간접 맵(`autofs`)
 - `automount` 명령을 실행해야 하는 경우, 87
 - 개요, 50, 51
 - 구문, 50, 50
 - 설명, 87
 - 예, 50, 51
 - 주석, 50
 - 검증기
 - RPC 인증 시스템, 40
 - 계층적 마운트(다중 마운트), 52
 - 공개 키 데이터베이스
 - 보안 키
 - 데이터베이스, 41
 - 공개 키 맵
 - DH 인증, 41
 - 공개 키 암호화
 - DH 인증, 41, 41
 - 공개 키 데이터베이스, 40, 41
 - 공통 키, 41
 - 대화 키, 41
 - 보안 키
 - 원격 서버에서 삭제, 41
 - 시간 동기화, 41
 - 공용 파일 핸들
 - `autofs` 및, 96
 - NFS 마운트, 18
 - WebNFS 및, 78
 - 마운트 및, 34
 - 공유 살펴볼 내용 파일 공유
 - 공유 해제 및 다시 공유
 - NFS 버전 4, 25
 - 권한
 - NFS 버전 3의 개선 사항, 14
- ㄴ
- 네트워크 잠금 관리자, 17
- ㄷ
- 단일 사용자 모드 및 보안, 41
 - 대시(-)
 - `autofs` 맵 이름, 57
 - 대화 키, 41
 - 더하기 기호(+)

- autofs 맵 이름, 57, 58
 - 데몬
 - automountd, 145
 - autofs 및, 12
 - 개요, 44, 45
 - lockd, 146
 - mountd, 146
 - rpcbind 데몬으로 등록되지 않음, 137
 - 서버의 응답 확인, 128
 - 실행 중인지 확인, 129, 137
 - nfs4cbd, 147
 - nfsd
 - 서버의 응답 확인, 128
 - 설명, 147
 - 실행 중인지 확인, 129
 - nfslogd, 148
 - nfsmapid, 148
 - reparsed, 154
 - rpcbind
 - 마운트 오류 메시지, 136, 137
 - statd, 154
 - 원격 마운트에 필요, 119
 - 도메인
 - 정의, 75
 - 도메인 이름
 - 보안 NFS 시스템 및, 75
 - 디스크가 없는 클라이언트
 - 부트 프로세스 중의 보안, 41
 - 수동 마운트 요구 사항, 12
- ㄹ**
- 로컬 캐시 및 NFS 버전 3, 14
 - 로컬 파일
 - autofs 맵 업데이트, 87
 - 로컬 파일 시스템
 - 그룹 마운트 해제, 108
 - 루트 디렉토리
 - 디스크가 없는 클라이언트를 통한 마운트, 12
- ㄴ**
- 마스터 맵(auto_master)
 - /- 마운트 지점, 46, 49
 - /etc/mnttab 파일과 비교, 45
 - automount 명령을 실행해야 하는 경우, 87
 - 개요, 46, 46
 - 구문, 46
 - 보안 제한, 95
 - 사전 설치됨, 90
 - 설명, 87
 - 주석, 47
 - 컨텐츠, 46, 48
 - 마운트
 - autofs 및, 12, 53
 - FedFS, 81
 - nfsd 데몬 및, 34
 - 공용 파일 핸들 및, 34
 - 디스크가 없는 클라이언트 요구 사항, 12
 - 미러 마운트 및, 42
 - 백그라운드 재시도, 102
 - 소프트 및 하드 비교, 119
 - 예, 105
 - 원격 마운트
 - 데몬 필요, 119
 - 문제 해결, 126, 129
 - 이미 마운트된 파일 시스템 오버레이, 105
 - 읽기 전용 사양, 104
 - 읽기 전용 지정, 105
 - 읽기/쓰기 사양, 104
 - 전경 재시도, 102
 - 직접 I/O 강제 적용, 103
 - 키보드를 통한 중단, 119
 - 포트매퍼 및, 34
 - 표의 모든 파일 시스템, 108
 - 마운트 미러링
 - 한 서버에서 모든 파일 시스템 마운트, 68
 - 마운트 지점
 - /- as 마스터 맵 마운트 지점, 46, 49
 - /home, 46, 47
 - /net, 47
 - /nfs4, 46, 48
 - 충돌 방지, 89
 - 마운트 지점을 만들 수 없음, 132
 - 마운트 해제
 - autofs 및, 12, 53
 - 미러 마운트 및, 43
 - 예제, 107
 - 파일 시스템 그룹, 108
 - 만들기
 - NFS 참조, 44, 80, 83
 - 보안 연결(FedFS), 82

- 이름 공간 데이터베이스(FedFS), 82
- 매핑되지 않은 사용자 또는 그룹 ID
 - 확인, 32
- 매핑되지 않은 사용자 또는 그룹 ID 확인, 32
- 맵 항목의 변수, 57, 57
- 맵(autofs)
 - automount 명령
 - 실행해야 하는 경우, 87
 - 간접, 50, 51
 - 관리 작업, 87
 - 긴 행 분할, 47, 49, 50
 - 네트워크 탐색, 46
 - 다른 맵 참조, 57, 58
 - 다중 마운트, 52
 - 마스터, 46, 46
 - 마운트 충돌 방지, 89
 - 변수, 57, 57
 - 실행 가능, 58
 - 유지 관리 방법, 87
 - 유형 및 해당 용도, 87
 - 주석, 47, 49, 50
 - 직접, 48, 49
 - 클라이언트에 대해 읽기 전용 파일 선택, 53, 56
 - 탐색 프로세스 시작, 47, 51
 - 특수 문자, 62
- 맵을 사용한 탐색
 - 개요, 46
 - 프로세스 시작, 47, 51
- 맵의 백슬래시(₩), 47, 49, 50
- 맵의 서버 가중치, 56
- 맵의 특수 문자
 - 따옴표로 묶기, 62
- 명령
 - FedFS, 118
 - NFS, 99
 - 정지된 프로그램, 138
- 목록
 - 공유 파일 시스템, 114
 - 마운트된 파일 시스템, 107
 - 원격으로 마운트된 파일 시스템을 포함하는 클라이언트, 117
- 문제 해결
 - autofs, 131
 - automount -v 명령을 통해 생성되는 오류 메시지, 131
 - 기타 오류 메시지, 133
 - 마운트 지점 충돌 방지, 89
- NFS
 - NFS 서비스가 실패한 위치 확인, 129
 - 서버 문제, 126
 - 원격 마운트 문제, 126, 137
 - 전략, 119
 - 정지된 프로그램, 138
- 미러 마운트
 - 개요, 42
- ㅂ**
- 방화벽
 - NFS 액세스, 18
 - WebNFS 액세스, 80
 - 파일 시스템 마운트, 70
- 백그라운드 파일 마운트 옵션, 102
- 버전 협상
 - NFS, 24
- 별표(*)
 - autofs 맵, 62
- 보안
 - autofs 제한 적용, 95
 - DH 인증
 - 개요, 41, 41
 - 사용자 인증, 39
 - 암호 보호, 40
 - NFS 버전 3 및, 14
 - UNIX 인증, 39, 40
 - 보안 NFS 시스템
 - 개요, 39
 - 관리, 75
 - 보안 RPC
 - DH 권한 부여 문제, 41, 42
 - 개요, 40
 - 파일 공유 문제, 111, 113
- 보안 모드 선택 및 mount 명령, 104
- 보안 및 NFS
 - 설명, 15, 31
 - 오류 메시지, Permission denied, 138
- 보안 종류, 18
- 보안 키
 - 데이터베이스, 41
 - 서버 충돌 및, 41, 41
 - 원격 서버에서 삭제, 41
- 보안 NFS 시스템

- DH 인증 및, 76
 - 개요, 39
 - 관리, 75
 - 도메인 이름, 75
- 보안 RPC
 - DH 권한 부여 문제, 41, 42
 - 개요, 40
- 복제된 마운트
 - 소프트 옵션 및, 139
- 복제된 파일 시스템, 36
- 부트
 - 디스크가 없는 클라이언트 보안, 41
 - 파일 시스템 마운트, 67
- ㅅ
- 사용 안함으로 설정
 - autofs 찾아보기 기능
 - 개요, 96
 - 단일 클라이언트에 대한 마운트 액세스, 69
- 사용으로 설정
 - NFS 서버 로깅, 65
 - WebNFS 서비스, 79
 - 클라이언트측 페일오버, 69
- 서버, 95
 - 살펴볼 다른 내용 NFS 서버
 - autofs 파일 선택, 53
 - NFS 서버 및 vfstab 파일, 67
 - NFS 서비스, 13
 - 충돌 및 보안 키, 41, 41
 - 홈 디렉토리 서버 설정, 91
- 서버 및 클라이언트
 - NFS 서비스, 13
- 서버가 응답하지 않음 메시지
 - 키보드를 통한 중단, 119
- 설정
 - nobrowse 매개변수, 96
- 수정
 - NFS 참조, 81
- 수퍼 유저
 - autofs 및 암호, 12
- 숫자 기호(#)
 - 간접 맵의 주석, 50
 - 마스터 맵(auto_master)의 주석, 47
 - 직접 맵의 주석, 49
- 슬래시(/)
 - /- as 마스터 맵 마운트 지점, 46, 49
 - 루트 디렉토리, 디스크가 없는 클라이언트를 통한
 - 마운트, 12
 - 마스터 맵 이름 앞에 붙음, 47
 - 시간 동기화, 41, 41
 - 실행 가능 맵, 58
 - 쓰기 오류
 - NFS 및, 14
- ㅇ
- 암호
 - autofs 및 수퍼 유저 암호, 12
 - DH 암호 보호, 40
- 액세스
 - NFS 참조, 80
- 앰퍼센드(&)
 - autofs 맵, 61
- 여러 서버에서 공유 파일 복제, 95
- 열기 오류
 - NFS 및, 14
- 오류 메시지
 - automount -v 명령을 통해 생성됨 , 131
 - No such file or directory, 137
 - Permission denied, 137
 - server not responding
 - 원격 마운트 문제, 136, 138
 - 정지된 프로그램, 138
 - 기타 automount 메시지, 133
 - 서버가 응답하지 않음
 - 키보드를 통한 중단, 119
 - 쓰기 오류
 - NFS 및, 14
 - 열기 오류
 - NFS 및, 14
- 운영 체제
 - 맵 변수, 57
 - 호환되지 않는 버전 지원, 94
- 원격 마운트
 - 데몬 필요, 119
 - 문제 해결, 126, 129
- 원격 파일 시스템
 - 그룹 마운트 해제, 108
 - 원격으로 마운트된 파일 시스템을 포함하는 클라이언트 목록, 117
- 위임

- NFS 버전 4, 30
 - 응용 프로그램
 - 정지됨, 138
 - 이름 공간
 - autofs 및, 19
 - 공유 액세스, 93
 - 이름 지정 서비스
 - autofs 맵 유지 관리 방법, 87
 - 이미 마운트된 파일 시스템 오버레이, 105
 - 인증
 - 공유되거나 내보낸 파일 목록, 117
 - 원격으로 마운트된 디렉토리 목록, 117
 - 인증
 - DH, 41
 - DH(Diffie-Hellman), 41
 - RPC, 40
 - UNIX, 39, 40
 - 읽기 전용 유형
 - autofs의 파일 선택, 56
 - 파일 시스템 공유, 111, 112, 114
 - 파일 시스템 마운트, 104, 105
 - 읽기 전용 파일
 - autofs의 파일 선택, 53
 - 읽기/쓰기 유형
 - 파일 시스템 공유, 112, 114
 - 파일 시스템 마운트, 104
- ㄹ**
- 자격 증명
 - UNIX 인증, 40
 - 설명, 40
 - 잠금
 - NFS 버전 3의 향상된 기능, 17
 - 잠금 제거, 100
 - 전경 파일 마운트 옵션, 102
 - 전송 프로토콜
 - NFS 협상, 33
 - 정지된 프로그램, 138
 - 제거
 - NFS 참조, 81
 - 참조, 44
 - 제한
 - 표시된 파일 시스템 정보, 71
 - 주석
 - 간접 맵, 50
 - 마스터 맵(auto_master), 47
 - 직접 맵, 49
 - 직렬 마운트 해제, 108
 - 직접 맵(autofs)
 - automount 명령을 실행해야 하는 경우, 87
 - 개요, 49
 - 구문, 48
 - 설명, 87
 - 예제, 48
 - 주석, 49
 - 직접 I/O 마운트 옵션, 102
- ㅋ**
- 참조 살펴볼 내용 NFS 참조
 - 찾아보기
 - NFS URL 사용, 79
 - 찾아보기 기능
 - 개요, 19
 - 사용 안함으로 설정, 96
- 크**
- 캐시 및 NFS 버전 3, 14
 - 커널
 - 서버의 응답 확인, 126
 - 큰 파일
 - NFS 지원, 17
 - 클라이언트 복구
 - NFS 버전 4, 28
 - 클라이언트측 파일오버
 - NFS 버전 4, 36
 - NFS 잠금 및, 36
 - NFS 지원, 17
 - 개요, 35
 - 복제된 파일 시스템, 36
 - 사용으로 설정, 69
 - 키보드를 통한 마운트 중단, 119
 - 키워드
 - NFS 버전 협상, 24
- ㅋ**
- 통합 파일 시스템 살펴볼 내용 FedFS

ㅁ

파운드 기호(#)

- 마스터 맵(auto_master)의 주석, 47
- 직접 맵의 주석, 49

 파일 공유

- 예, 114

 파일 공유

- NFS 버전 3의 개선 사항, 14
- NFS 버전 3의 향상된 기능, 17
- 개요, 111
- 공유 해제, 116
- 나열된 클라이언트만, 112
- 루트 액세스 권한 부여, 113
- 보안 문제, 39, 111, 113
- 여러 서버에서 공유 파일 복제, 95
- 여러 파일 시스템, 116
- 인증되지 않은 사용자 및, 112
- 읽기 전용 액세스, 111, 112, 114
- 읽기/쓰기 액세스, 112, 114
- 자동, 64

 파일 공유 옵션, 112

- 파일 권한
 - NFS 버전 3의 개선 사항, 14
 - WebNFS 및, 78

 파일 및 파일 시스템

- autofs 액세스
 - 비 NFS 파일 시스템, 89
- autofs 파일 선택, 53, 56
- NFS 파일 및 해당 기능, 141
- NFS의 , 13
- NFS의 처리, 13
- 로컬 파일 시스템
 - 그룹 마운트 해제, 108
- 원격 파일 시스템
 - 그룹 마운트 해제, 108
 - 원격으로 마운트된 파일 시스템을 포함하는 클라이언트 목록, 117
 - 파일 시스템 표에서 마운트, 108
- 정의, 13
- 파일 시스템, 13
- 프로젝트 관련 파일 통합, 91

 파일 속성 및 NFS 버전 3, 14

- 파일 시스템 공유 해제
 - unshare 명령, 116
 - unshareall 명령, 116
- 파일 시스템 마운트

- autofs 및, 68
- NFS URL 사용, 71
- 개요, 66
- 단일 클라이언트에 대한 액세스 사용 안함으로 설정, 69
- 방화벽 통과, 70
- 부트 시 방법, 67
- 수동으로(즉시), 67
- 작업 맵, 66
- 한 서버에서 모두 마운트, 68
- 파일 시스템 및 NFS 처리, 13
- 파일 시스템 이름 공간
 - NFS 버전 4, 25
- 파일 전송 크기
 - 협상, 33
- 페일오버
 - mount 명령 예제, 105, 105
 - NFS 지원, 17
 - 오류 메시지, 136
- 포트매퍼
 - 마운트 및, 34
- 표시
 - 마운트 가능한 파일 시스템, 71
 - 제한된 파일 시스템 정보, 71
- 프로그램
 - 정지됨, 138
- 프로세서 유형 맵 변수, 57
- 프로젝트
 - 파일 통합, 91
 - 프로젝트 관련 파일 통합, 91

ㅎ

협상

- WebNFS 보안, 18
- 파일 전송 크기, 33

 호스트

- 모든 파일 시스템 마운트 해제, 108

 휘발성 파일 핸들

- NFS 버전 4, 27

