

## Oracle® Solaris 관리: 장치 및 파일 시스템

Copyright © 2004, 2012, Oracle and/or its affiliates. All rights reserved.

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

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

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

#### U.S. GOVERNMENT 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 회사는 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 등록 상표입니다.

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

# 목차

---

머리말 .....	17
<b>1 이동식 매체 관리(개요) .....</b>	<b>21</b>
이동식 매체의 새로운 기능 .....	21
이동식 매체 관리에 대해 변경되거나 향상된 기능 .....	21
이동식 매체 관리 작업을 찾을 수 있는 위치 .....	24
이동식 매체의 기능 및 이점 .....	25
수동 및 자동 마운트 비교 .....	25
이동식 매체에 액세스 개요 .....	26
<b>2 이동식 매체 관리(작업) .....</b>	<b>27</b>
이동식 매체 관리(작업 맵) .....	27
이동식 매체 준비 .....	28
이동식 매체 고려 사항 .....	28
▼ 이동식 매체를 로드하는 방법 .....	29
▼ 디스켓을 포맷하는 방법(rmformat) .....	30
▼ 이동식 매체에서 파일 시스템을 만드는 방법 .....	31
▼ DVD-RAM에서 파일 시스템을 만드는 방법 .....	32
▼ 이동식 매체에서 파일 시스템을 검사하는 방법 .....	32
▼ 이동식 매체에서 잘못된 블록을 복구하는 방법 .....	33
이동식 매체에 읽기 또는 쓰기 보호와 암호 보호 적용 .....	33
▼ 이동식 매체에서 쓰기 보호를 사용으로 설정하거나 사용 안함으로 설정하는 방법 .....	33
▼ 이동식 매체에서 읽기 또는 쓰기 보호를 사용으로 설정하거나 사용 안함으로 설정하고 암호를 설정하는 방법 .....	34

<b>3</b>	<b>이동식 매체에 액세스(작업)</b> .....	35
	이동식 매체에 액세스(작업 맵) .....	35
	이동식 매체에 액세스 .....	36
	이동식 매체 이름 사용 .....	36
	이동식 매체 데이터에 액세스하기 위한 지침 .....	36
	▼ 새 이동식 매체 드라이브를 추가하는 방법 .....	36
	▼ 이동식 매체 서비스를 사용으로 설정 또는 사용 안함으로 설정하는 방법 .....	37
	▼ 이동식 매체에 대한 정보에 액세스하는 방법 .....	38
	▼ 이동식 매체에서 정보를 복사하는 방법 .....	38
	▼ 이동식 매체가 아직 사용 중인지 확인하는 방법 .....	39
	▼ 이동식 매체를 꺼내는 방법 .....	39
	원격 시스템에서 이동식 매체에 액세스(작업 맵) .....	40
	▼ 다른 시스템에서 사용할 수 있도록 로컬 매체를 설정하는 방법 .....	40
	▼ 원격 시스템에서 이동식 매체에 액세스하는 방법 .....	41
<b>4</b>	<b>CD 및 DVD에 쓰기(작업)</b> .....	43
	오디오 CD와 데이터 CD 및 DVD 사용 .....	43
	CD/DVD 매체 관련 일반적으로 사용하는 용어 .....	44
	CD 및 DVD 데이터와 오디오 CD에 쓰기 .....	45
	RBAC를 사용하여 원격 매체에 대한 사용자 액세스 제한 .....	46
	▼ RBAC를 사용하여 이동식 매체에 대한 사용자 액세스를 제한하는 방법 .....	46
	▼ CD 또는 DVD 라이터를 식별하는 방법 .....	47
	▼ CD 또는 DVD 매체를 검사하는 방법 .....	47
	데이터 CD 또는 DVD 만들기 .....	48
	▼ 데이터 CD 또는 DVD에 사용할 ISO 9660 파일 시스템을 만드는 방법 .....	48
	▼ 멀티 세션 데이터 CD를 만드는 방법 .....	49
	오디오 CD 만들기 .....	51
	▼ 오디오 CD를 만드는 방법 .....	51
	▼ CD에서 오디오 트랙을 추출하는 방법 .....	52
	▼ CD를 복사하는 방법 .....	53
	▼ CD-RW 매체를 지우는 방법 .....	54
<b>5</b>	<b>장치 관리(개요/작업)</b> .....	55
	장치 관리에 관련된 새로운 기능 .....	56
	드라이버 구성 사용자 정의 .....	56

Solaris PCI 리소스 관리자 .....	56
새로운 InfiniBand 관리 기능 .....	57
새로운 InfiniBand 진단 도구 및 명령 .....	57
새로운 Ethernet Over InfiniBand 장치 .....	57
새로운 핫플러그 기능 .....	57
장치 이름 지정 기능 향상 .....	58
PCIe(PCI Express) 지원 .....	58
추가 장치 관리 작업 검색 위치 .....	59
Oracle Solaris OS에서 장치 관리 .....	60
장치 지원 식별 .....	60
장치 드라이버 정보 .....	60
장치의 자동 구성 .....	61
장치 구성 정보 표시 .....	63
고장난 장치 해결 .....	67
시스템에 주변 장치 추가 .....	69
▼ 주변 장치를 추가하는 방법 .....	69
▼ 장치 드라이버를 추가하는 방법 .....	70
장치 액세스 .....	71
장치 정보를 만드는 방법 .....	71
장치를 관리하는 방법 .....	71
장치 이름 지정 규약 .....	72
논리적 디스크 장치 이름 .....	73
논리적 테이프 장치 이름 .....	75
논리적 이동식 매체 장치 이름 .....	76
<b>6 동적으로 장치 구성(작업) .....</b>	<b>77</b>
동적 재구성 및 핫플러그 기능 사용 .....	77
첨부 지점 .....	79
PCI 또는 PCIe 어댑터 카드 분리 .....	80
PCI 또는 PCIe 어댑터 카드 연결 .....	81
hotplug 명령을 사용하여 PCIe 핫플러그 기능 사용 .....	81
cfgadm 명령을 사용하여 SCSI 핫플러그 기능 사용(작업 맵) .....	83
cfgadm 명령을 사용하여 SCSI 핫플러그 기능 사용 .....	84
▼ SCSI 장치에 대한 정보를 표시하는 방법 .....	84
▼ SCSI 제어기를 구성 해제하는 방법 .....	85

▼ SCSI 제어를 구성하는 방법 .....	85
▼ SCSI 장치를 구성하는 방법 .....	86
▼ SCSI 제어를 연결 해제하는 방법 .....	86
▼ SPARC: SCSI 제어를 연결하는 방법 .....	87
▼ SPARC: SCSI 장치를 SCSI 버스에 추가하는 방법 .....	88
▼ SPARC: SCSI 제어기에서 동일한 장치를 교체하는 방법 .....	89
▼ SPARC: SCSI 장치를 제거하는 방법 .....	90
SCSI 구성 문제 해결 .....	92
▼ 실패한 SCSI 구성 해제 작업을 해결하는 방법 .....	93
cfgadm 명령을 사용하여 PCI 또는 PCIe 핫 플러그 기능 사용(작업 맵) .....	93
cfgadm 명령을 사용하여 PCI 또는 PCIe 핫 플러그 기능 사용 .....	94
PCIe LED 표시기 동작 .....	94
▼ PCI 슬롯 구성 정보를 표시하는 방법 .....	94
▼ PCI 어댑터 카드를 제거하는 방법 .....	96
▼ PCI 어댑터 카드를 추가하는 방법 .....	97
PCI 구성 문제 해결 .....	99
cfgadm 명령을 사용하여 SATA 핫 플러그 기능 사용 .....	99
▼ SATA 장치를 구성 해제하는 방법 .....	100
▼ SATA 장치를 구성하는 방법 .....	100
RCM(Reconfiguration Coordination Manager) 스크립트 개요 .....	101
RCM 스크립트란 무엇입니까? .....	101
RCM 스크립트로 어떤 작업을 수행할 수 있습니까? .....	102
RCM 스크립트 프로세스는 어떻게 작동합니까? .....	102
RCM 스크립트 작업 .....	103
응용 프로그램 개발자 RCM 스크립트(작업 맵) .....	103
시스템 관리자 RCM 스크립트(작업 맵) .....	104
RCM 스크립트 이름 지정 .....	104
RCM 스크립트 설치 또는 제거 .....	104
▼ RCM 스크립트를 설치하는 방법 .....	105
▼ RCM 스크립트를 제거하는 방법 .....	105
▼ RCM 스크립트를 테스트하는 방법 .....	106
테이프 백업 RCM 스크립트의 예 .....	106
<b>7 USB 장치 사용(개요) .....</b>	<b>111</b>
USB 장치의 새로운 기능은 무엇입니까? .....	112

USB IAD(Interface Association Descriptor) 지원 .....	112
EHCI 동시 전송 지원 .....	112
USB 장치 Hotpluggable 동작 .....	112
x86: GRUB 기반 부트 시 USB CD 및 DVD 지원 .....	113
USB 가상 키보드 및 마우스 지원 .....	113
USB 장치용 Oracle Solaris 지원 .....	114
USB 장치 개요 .....	115
일반적으로 사용되는 USB 머리글자어 .....	115
USB 버스 설명 .....	115
Oracle Solaris OS의 USB 정보 .....	120
USB 2.0 기능 .....	120
USB 키보드 및 마우스 장치 .....	122
USB 호스트 제어기 및 허브 .....	123
USB 케이블 지침 .....	124
<b>8 USB 장치 사용(작업) .....</b>	<b>127</b>
Oracle Solaris OS에서 USB 장치 관리(로드맵) .....	127
USB 대용량 저장 장치 사용(작업 맵) .....	128
USB 대용량 저장 장치 사용 .....	129
USB 디스켓 장치 사용 .....	130
비호환 USB 대용량 저장 장치 사용 .....	131
USB 대용량 저장 장치의 핫 플러그 기능 사용 .....	132
USB 대용량 저장 장치 사용 준비 .....	135
▼ USB 장치 정보를 표시하는 방법 .....	136
▼ USB 대용량 저장 장치에 파일 시스템을 만드는 방법 .....	136
▼ USB 대용량 저장 장치에서 분할 영역을 수정하고 PCFS 파일 시스템을 만드는 방법 .....	139
▼ USB 대용량 저장 장치에서 Solaris 분할 영역을 만들고 슬라이스를 수정하는 방법 .....	142
▼ USB 대용량 저장 장치를 마운트 또는 마운트 해제하는 방법 .....	144
USB 대용량 저장 장치 문제 해결 팁 .....	145
특정 USB 드라이버를 사용 안함으로 설정 .....	145
▼ 특정 USB 드라이버를 사용 안함으로 설정하는 방법 .....	146
▼ 사용되지 않은 USB 장치 링크를 제거하는 방법 .....	146
USB 오디오 장치 사용(작업 맵) .....	147
USB 오디오 장치 사용 .....	147

여러 개의 USB 오디오 장치에서 핫 플러그 기능 사용 .....	148
▼ USB 오디오 장치를 추가하는 방법 .....	148
▼ 시스템의 기본 오디오 장치를 식별하는 방법 .....	149
▼ 기본 USB 오디오 장치를 변경하는 방법 .....	150
USB 오디오 장치 문제 해결 .....	150
cfgadm 명령을 사용하여 USB 장치 핫 플러그 기능 사용(작업 맵) .....	151
cfgadm 명령을 사용하여 USB 장치 핫 플러그 기능 사용 .....	151
▼ USB 버스 정보를 표시하는 방법(cfgadm) .....	152
▼ USB 장치를 구성 해제하는 방법 .....	153
▼ USB 장치를 구성하는 방법 .....	154
▼ 논리적으로 USB 장치를 연결 해제하는 방법 .....	154
▼ 논리적으로 USB 장치를 연결하는 방법 .....	154
▼ USB 장치 하위 트리를 논리적으로 연결 해제하는 방법 .....	155
▼ USB 장치를 재설정하는 방법 .....	155
▼ 다중 구성 USB 장치의 기본 구성을 변경하는 방법 .....	156
<b>9 InfiniBand 장치 사용(개요/작업) .....</b>	<b>157</b>
InfiniBand 장치 개요 .....	157
InfiniBand 소프트웨어 패키지 .....	159
동적으로 IB 장치 재구성(작업 맵) .....	159
동적으로 IB 장치 재구성(cfgadm) .....	161
IB 장치 구성 해제 고려 사항 .....	162
▼ IB 장치 정보 표시 방법 .....	162
▼ IB 포트, HCA_SVC 또는 VPPA 장치를 구성 해제하는 방법 .....	164
▼ IB 포트, HCA_SVC 또는 VPPA 장치 구성 방법 .....	164
▼ IB 의사 장치의 구성 해제 방법 .....	165
▼ IB 의사 장치 구성 방법 .....	165
▼ HCA의 커널 IB 클라이언트 표시 방법 .....	166
▼ EoIB 장치가 활성화된 상태로 HCA를 동적으로 재구성하는 방법 .....	166
▼ 작업 중 제거 후 EoIB 인터페이스를 재구성 및 복원하는 방법 .....	167
IB HCA 구성 .....	168
▼ IB_P_key 테이블을 업데이트하는 방법 .....	168
▼ IB 통신 서비스 표시 방법 .....	169
▼ VPPA 통신 서비스 추가 방법 .....	169
▼ 기존 IB 포트, HCA_SVC 또는 VPPA 통신 서비스 제거 방법 .....	170



InfiniBand 장치에서 uDAPL 응용 프로그램 인터페이스 사용 .....	170
▼ uDAPL을 사용으로 설정하는 방법 .....	171
DAT 정적 레지스트리 업데이트 .....	171
IPoIB 장치 관리(dladm) .....	172
▼ 물리적 데이터 링크 정보를 표시하는 방법 .....	173
▼ IB 분할 영역 링크를 만드는 방법 .....	173
▼ IB 분할 영역 링크 정보를 표시하는 방법 .....	174
▼ IB 분할 영역 링크를 제거하는 방법 .....	175
IB 장치 모니터 및 문제 해결 .....	175
<b>10 디스크 관리(개요) .....</b>	<b>179</b>
디스크 관리의 새로운 기능 .....	179
물리적 위치로 장치 식별 .....	180
여러 디스크 섹터 크기 지원 .....	183
Oracle Solaris OS 설치 및 부트 시 2TB 디스크 지원 .....	183
Solaris iSCSI 대상 및 시작 프로그램에서 iSNS 지원 .....	184
Solaris COMSTAR iSCSI 지원 .....	184
x86: GRUB 부트 환경의 디스크 관리 .....	184
2TB보다 큰 SCSI 디스크에 대한 지원 .....	185
디스크 관리 작업 검색 위치 .....	185
디스크 관리 개요 .....	185
디스크 용어 .....	186
디스크 레이블 정보 .....	186
EFI 디스크 레이블 .....	187
디스크 슬라이스 정보 .....	190
format 유틸리티 .....	191
디스크 분할 .....	194
분할 영역 테이블 용어 .....	194
분할 영역 테이블 정보 표시 .....	195
Free Hog 슬라이스 사용 .....	197
<b>11 디스크 관리(작업) .....</b>	<b>199</b>
디스크 관리(작업 맵) .....	199
시스템의 디스크 식별 .....	200
▼ 시스템의 디스크 식별 방법 .....	200

디스크 포맷 .....	202
▼ 디스크가 포맷되었는지 확인하는 방법 .....	202
▼ 디스크 포맷 방법 .....	203
디스크 슬라이스 표시 .....	204
▼ 디스크 슬라이스 정보 표시 방법 .....	204
디스크 레이블 만들기 및 검사 .....	206
▼ 디스크에 레이블을 지정하는 방법 .....	206
▼ 디스크 레이블 검사 방법 .....	211
손상된 디스크 레이블 복구 .....	212
▼ 손상된 디스크 레이블 복구 방법 .....	213
타사 디스크 추가 .....	215
<b>12 SPARC: 디스크 설정(작업) .....</b>	<b>217</b>
SPARC: ZFS 파일 시스템에 사용할 디스크 설정(작업 맵) .....	217
SPARC: ZFS 파일 시스템에 사용할 디스크 설정 .....	218
▼ SPARC: ZFS 루트 파일 시스템에 사용할 디스크 설정 방법 .....	219
SPARC: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스 만들기 .....	219
▼ SPARC: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만드는 방법 .....	220
▼ SPARC: ZFS 루트 파일 시스템의 부트 블록 설치 방법 .....	223
▼ SPARC: ZFS 파일 시스템에 사용할 디스크 설정 방법 .....	224
<b>13 x86: 디스크 설정(작업) .....</b>	<b>227</b>
x86: ZFS 파일 시스템에 사용할 디스크 설정(작업 맵) .....	227
x86: ZFS 파일 시스템에 사용할 디스크 설정 .....	228
Solaris fdisk 분할 영역 만들기 및 변경 .....	235
x86: fdisk 분할 영역을 만드는 방법에 대한 지침 .....	235
▼ x86: Solaris fdisk 분할 영역을 만드는 방법 .....	236
fdisk 분할 영역 식별자 변경 .....	239
▼ Solaris fdisk 식별자 변경 방법 .....	239
<b>14 COMSTAR를 사용하여 저장 장치 구성 .....</b>	<b>241</b>
COMSTAR 및 iSCSI 기술(개요) .....	241
COMSTAR 소프트웨어 및 하드웨어 요구 사항 .....	243
COMSTAR 구성(작업 맵) .....	243

COMSTAR 구성 .....	245
COMSTAR를 사용하여 iSCSI 장치 구성 .....	247
▼ STMF 서비스를 사용으로 설정하는 방법 .....	247
▼ COMSTAR 구성 백업 및 복원 방법 .....	248
▼ iSCSI LUN을 만드는 방법 .....	248
▼ iSCSI 대상을 만드는 방법 .....	249
▼ iSER에 대해 IB HCA를 구성하는 방법 .....	250
▼ iSCSI 시작 프로그램 구성 방법 .....	251
▼ 검색된 iSCSI 대상 제거 방법 .....	252
iSCSI 대상 포털 그룹 만들기 .....	254
▼ iSCSI 디스크 액세스 방법 .....	255
SCSI 논리 장치를 사용 가능하게 설정 .....	256
▼ 모든 시스템이 논리 장치를 사용할 수 있게 하는 방법 .....	257
▼ 선택한 시스템만 LUN에 액세스하도록 제한하는 방법 .....	257
COMSTAR를 사용하여 Fibre Channel 장치 구성 .....	258
COMSTAR에 대해 Fibre Channel 포트 구성 .....	259
논리 장치를 FC 및 FCoE에 사용할 수 있게 설정 .....	261
COMSTAR를 사용하여 FCoE 장치 구성 .....	263
FCoE 포트 구성 .....	263
이더넷 인터페이스에서 802.3x PAUSE 및 점보 프레임 사용 .....	264
▼ FCoE 대상 포트를 만드는 방법 .....	264
▼ FCoE 대상 포트가 작동하는지 확인하는 방법 .....	264
▼ FCoE 대상 포트 삭제 방법 .....	265
COMSTAR를 사용하여 SRP 장치 구성 .....	266
SRP와 함께 COMSTAR 뷰 사용 .....	266
▼ SRP 대상 서비스를 사용으로 설정하는 방법 .....	267
▼ SRP 대상 상태 확인 방법 .....	267
iSCSI 기반 저장소 네트워크에서 인증 구성 .....	267
▼ iSCSI 시작 프로그램에 대해 CHAP 인증을 구성하는 방법 .....	268
▼ iSCSI 대상에 대해 CHAP 인증을 구성하는 방법 .....	270
타사 RADIUS 서버를 사용하여 iSCSI 구성에서 CHAP 관리 단순화 .....	270
Oracle Solaris에서 iSCSI 다중 경로 장치 설정 .....	273
▼ 대상에 대해 여러 iSCSI 세션을 사용으로 설정하는 방법 .....	274
iSCSI 구성 모니터 .....	276
▼ iSCSI 구성 정보 표시 방법 .....	276
iSCSI 시작 프로그램 및 대상 매개변수 수정 .....	278

iSCSI 매개변수 조정 .....	278
▼ iSCSI 시작 프로그램 및 대상 매개변수 수정 방법 .....	280
iSCSI 구성 문제 해결 .....	283
로컬 시스템이 iSCSI 대상에 연결되어 있지 않음 .....	283
로컬 시스템에서 iSCSI 장치 또는 디스크를 사용할 수 없음 .....	285
iSNS 검색 방법을 사용할 때 LUN 마스킹 사용 .....	285
일반적인 iSCSI 오류 메시지 .....	286
<b>15 Oracle Solaris iSNS(Internet Storage Name Service) 구성 및 관리 .....</b>	<b>291</b>
iSNS 기술(개요) .....	291
iSNS 서버 구성 .....	293
iSNS 관리 설정 지정 .....	294
명령줄 인터페이스를 사용하여 iSNS 구성 .....	296
iSNS 서버 및 클라이언트 관리 .....	299
▼ 검색 도메인 세트의 상태 표시 방법 .....	299
▼ 검색 도메인의 상태 표시 방법 .....	299
▼ 클라이언트 상태 표시 방법 .....	300
▼ 검색 도메인에서 클라이언트를 제거하는 방법 .....	300
▼ 검색 도메인 세트에서 검색 도메인을 제거하는 방법 .....	300
▼ 검색 도메인 세트를 사용 안함으로 설정하는 방법 .....	301
▼ 검색 도메인 세트 제거 방법 .....	301
<b>16 format 유틸리티(참조) .....</b>	<b>303</b>
format 유틸리티 사용을 위한 권장 사항 및 요구 사항 .....	303
format 메뉴 및 명령 설명 .....	304
partition 메뉴 .....	306
x86: fdisk 메뉴 .....	306
analyze 메뉴 .....	307
defect 메뉴 .....	309
format 명령에 대한 입력 규칙 .....	310
format 명령에 숫자 지정 .....	310
format 명령 이름 지정 .....	310
format 명령에 디스크 이름 지정 .....	311
format 유틸리티에 대한 도움말 보기 .....	311

<b>17</b>	<b>파일 시스템 관리(개요)</b> .....	313
	Oracle Solaris 파일 시스템의 새로운 기능 .....	313
	파일 시스템 모니터링 도구(fsstat) .....	313
	Oracle Solaris ZFS 파일 시스템 .....	314
	파일 시스템 관리 작업을 찾을 위치 .....	314
	파일 시스템 개요 .....	315
	Oracle Solaris 파일 시스템 유형 .....	315
	기본 Oracle Solaris 파일 시스템 .....	320
	파일 시스템 마운트 및 마운트 해제에 대한 개요 .....	321
	마운트된 파일 시스템 테이블 .....	322
	가상 파일 시스템 테이블 .....	322
	NFS 환경 .....	323
	자동 마운트(autofs) .....	324
	Oracle Solaris SMB 서비스 .....	325
	파일 시스템 유형 확인 .....	325
	파일 시스템 유형을 확인하는 방법 .....	325
<b>18</b>	<b>파일 시스템 만들기 및 마운트(작업)</b> .....	327
	Solaris 파일 시스템 만들기 .....	327
	ZFS 파일 시스템 만들기 .....	327
	임시 파일 시스템 만들기 .....	328
	LOFS 파일 시스템 만들기 .....	328
	Oracle Solaris 파일 시스템 마운트 및 마운트 해제 .....	328
	/etc/vfstab 파일에 대한 필드 설명 .....	330
	Oracle Solaris 파일 시스템 마운트 해제를 위한 필수 조건 .....	331
	Oracle Solaris 파일 시스템 만들기 및 마운트 .....	332
	▼ ZFS 파일 시스템을 만드는 방법 .....	332
	▼ 레거시 UFS 파일 시스템을 만들고 마운트하는 방법 .....	333
	▼ TMPFS 파일 시스템을 만들고 마운트하는 방법 .....	334
	▼ LOFS 파일 시스템을 만들고 마운트하는 방법 .....	335
	▼ /etc/vfstab 파일에 항목을 추가하는 방법 .....	336
	▼ 파일 시스템(/etc/vfstab 파일)을 마운트하는 방법 .....	337
	▼ NFS 파일 시스템을 마운트하는 방법(mount 명령) .....	338
	▼ x86: 하드 디스크에서 PCFS(DOS) 파일 시스템을 마운트하는 방법(mount 명령) .....	339
	▼ 파일 시스템에 액세스 중인 모든 프로세스를 중지하는 방법 .....	340

▼ 파일 시스템을 마운트 해제하는 방법 .....	340
<b>19 추가 스왑 공간 구성(작업) .....</b>	<b>343</b>
스왑 공간 정보 .....	343
스왑 공간 및 가상 메모리 .....	343
스왑 공간 및 TMPFS 파일 시스템 .....	344
스왑 공간 및 덤프 장치 구성 .....	345
스왑 공간 및 동적 재구성 .....	345
SAN 환경에서 스왑 공간 구성 .....	346
추가 스왑 공간의 필요 여부를 확인하는 방법 .....	346
스왑 관련 오류 메시지 .....	346
TMPFS 관련 오류 메시지 .....	347
스왑 공간을 할당하는 방법 .....	347
스왑 영역 및 /etc/vfstab 파일 .....	347
스왑 공간 계획 .....	348
ZFS 기반 시스템에 스왑 공간 할당 .....	349
스왑 리소스 모니터 .....	349
Oracle Solaris ZFS 루트 환경에서 스왑 공간 추가 및 변경 .....	350
▼ Oracle Solaris ZFS 루트 환경에서 스왑 공간을 추가하는 방법 .....	350
<b>20 파일 및 파일 시스템 복사(작업) .....</b>	<b>353</b>
파일 시스템 복사 명령 .....	353
파일 시스템 간에 디렉토리 복사(cpio 명령) .....	355
▼ 파일 시스템 간에 디렉토리를 복사하는 방법(cpio) .....	356
파일 및 파일 시스템을 테이프에 복사 .....	357
파일을 테이프에 복사(tar 명령) .....	357
▼ 파일을 테이프에 복사하는 방법(tar) .....	357
▼ 테이프에 있는 파일을 나열하는 방법(tar) .....	358
▼ 테이프에서 파일을 검색하는 방법(tar) .....	359
pax 명령을 사용하여 파일을 테이프에 복사 .....	360
▼ 파일을 테이프에 복사하는 방법(pax) .....	360
cpio 명령을 사용하여 파일을 테이프에 복사 .....	361
▼ 디렉토리에 있는 모든 파일을 테이프에 복사하는 방법(cpio) .....	361
▼ 테이프에 있는 파일을 나열하는 방법(cpio) .....	362
▼ 테이프에서 모든 파일을 검색하는 방법(cpio) .....	362

▼ 테이프에서 특정 파일을 검색하는 방법(cpio) .....	363
파일을 테이프 장치에 복사 .....	364
▼ 파일을 원격 테이프 장치에 복사하는 방법(tar 및 dd) .....	364
▼ 원격 테이프 장치에서 파일을 추출하는 방법 .....	365
<b>21 테이프 드라이브 관리(작업) .....</b>	<b>367</b>
사용할 매체 선택 .....	367
백업 장치 이름 .....	368
테이프 드라이브의 되감기 옵션 지정 .....	369
테이프 드라이브에 다양한 밀도 지정 .....	370
테이프 드라이브 상태 표시 .....	370
▼ 테이프 드라이브 상태를 표시하는 방법 .....	370
자기 테이프 카트리지 처리 .....	371
자기 테이프 카트리지 다시 조이기 .....	371
자기 테이프 카트리지 되감기 .....	372
드라이브 유지 관리 및 매체 사용 지침 .....	372
색인 .....	373





# 머리말

---

**시스템 관리 설명서:** 장치 및 파일 시스템은 Oracle Solaris 시스템 관리 정보의 중요한 부분을 다루는 설명서 모음의 일부입니다. 본 설명서에서는 SPARC 기반 시스템과 x86 기반 시스템에 대한 정보를 모두 다룹니다.

본 설명서에서는 다음 작업을 완료한 것으로 가정합니다.

- Oracle Solaris 11 운영 체제 설치
- 사용할 모든 네트워킹 소프트웨어 설정

Oracle Solaris 11 릴리스의 경우 시스템 관리자를 대상으로 한 새로운 기능이 해당 장의 ...의 새로운 기능이라는 섹션에서 설명됩니다.

---

주 - 본 Oracle Solaris 릴리스는 프로세서 아키텍처의 SPARC 및 x86 제품군을 사용하는 시스템을 지원합니다. 지원되는 시스템은 <http://www.oracle.com/webfolder/technetwork/hcl/index.html>에서 **Oracle Solaris 하드웨어 호환성 목록**을 참조하십시오. 이 설명서에서는 플랫폼 유형에 따른 구현 차이가 있는 경우 이에 대하여 설명합니다.

---

## 이 책의 대상

본 설명서는 Oracle Solaris 11 릴리스가 실행되고 있는 한 대 이상의 시스템을 관리하는 사용자를 대상으로 작성되었습니다. 본 설명서를 사용하려면 1-2년 정도의 UNIX 시스템 관리 경험이 있어야 합니다. UNIX 시스템 관리 교육 과정에 참석하는 것도 도움이 될 수 있습니다.

## 시스템 관리 설명서의 구성

시스템 관리 설명서에서 설명하는 항목 목록은 다음과 같습니다.

책 제목	내용
SPARC 플랫폼에서 Oracle Solaris 부트 및 종료	SPARC 플랫폼에서 시스템 부트 및 종료, 부트 서비스 관리, 부트 동작 수정, ZFS에서 부트, 부트 아카이브 관리 및 부트 문제 해결

---

책 제목	내용
<b>x86 플랫폼에서 Oracle Solaris 부트 및 종료</b>	x86 플랫폼에서 시스템 부트 및 종료, 부트 서비스 관리, 부트 동작 수정, ZFS에서 부트, 부트 아카이브 관리 및 부트 문제 해결
<b>Oracle Solaris 관리: 일반 작업</b>	Oracle Solaris 명령 사용, 시스템 부트 및 종료, 사용자 계정 및 그룹 관리, 서비스, 하드웨어 오류, 시스템 정보, 시스템 리소스 및 시스템 성능 관리, 소프트웨어, 인쇄, 콘솔 및 터미널 관리, 시스템 및 소프트웨어 문제 해결
<b>Oracle Solaris 관리: 장치 및 파일 시스템</b>	이동식 매체, 디스크 및 장치, 파일 시스템, 데이터 백업 및 복원
<b>Oracle Solaris 관리: IP 서비스</b>	TCP/IP 네트워크 관리, IPv4 및 IPv6 주소 관리, DHCP, IPsec, IKE, IP 필터 및 IPQoS
<b>Oracle Solaris Administration: Naming and Directory Services</b>	NIS에서 LDAP으로 전환을 비롯한 DNS, NIS 및 LDAP 이름 지정 및 디렉토리 서비스
<b>Oracle Solaris 관리: 네트워크 인터페이스 및 네트워크 가상화</b>	WiFi 무선을 포함하는 자동 및 수동 IP 인터페이스 구성, 브릿지, VLAN, 통합, LLDP 및 PMP 관리, 가상 NIC 및 리소스 관리
<b>Oracle Solaris 관리: 네트워크 서비스</b>	웹 캐시 서버, 시간 관련 서비스, 네트워크 파일 시스템(NFS 및 Autofs), 메일, SLP, PPP
<b>Oracle Solaris 관리: Oracle Solaris Zones, Oracle Solaris 10 Zones 및 리소스 관리</b>	응용 프로그램이 사용 가능한 시스템 리소스를 사용하는 방식을 제어할 수 있는 리소스 관리 기능, 운영 체제 서비스를 가상화하여 응용 프로그램을 실행하기 위한 격리된 환경을 만드는 Oracle Solaris Zones 소프트웨어 분할 기술, Oracle Solaris 11 커널에서 실행되는 Oracle Solaris 10 환경을 호스트하는 Oracle Solaris 10 영역
<b>Oracle Solaris 관리: 보안 서비스</b>	감사, 장치 관리, 파일 보안, BART, Kerberos 서비스, PAM, 암호화 프레임워크, 키 관리, 권한, RBAC, SASL, 보안 셸 및 바이러스 검사
<b>Oracle Solaris Administration: SMB and Windows Interoperability</b>	SMB 클라이언트가 SMB 공유를 사용할 수 있도록 Oracle Solaris 시스템을 구성할 수 있는 SMB 서비스, SMB 공유에 액세스할 수 있도록 해주는 SMB 클라이언트, 사용자 및 그룹 ID를 Oracle Solaris 시스템과 Windows 시스템 간에 매핑할 수 있도록 해주는 기본 ID 매핑 서비스
<b>Oracle Solaris 관리: ZFS 파일 시스템</b>	ZFS 저장소 풀 및 파일 시스템 만들기/관리, 스냅샷, 복제, 백업, ACL(액세스 제어 목록)을 통한 ZFS 파일 보호, 영역이 설치된 Solaris 시스템에서 ZFS 사용, 에뮬레이트된 볼륨, 문제 해결 및 데이터 복구
<b>Trusted Extensions 구성 및 관리</b>	Trusted Extensions와 관련된 시스템 설치, 구성 및 관리
<b>Oracle Solaris 11 보안 지침</b>	영역, ZFS 및 Trusted Extensions와 같은 보안 기능에 대한 사용 시나리오와 Oracle Solaris 시스템의 보안 설정

책 제목	내용
<b>Oracle Solaris 10에서 Oracle Solaris 11로 전환</b>	설치, 장치, 디스크 및 파일 시스템 관리, 소프트웨어 관리, 네트워킹, 시스템 관리, 보안, 가상화, 데스크탑 기능, 사용자 계정 관리, 사용자 환경 영역에서 Oracle Solaris 10을 Oracle Solaris 11로 전환하는 예제 및 시스템 관리 정보를 제공합니다.

## Oracle Support에 액세스

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

## 활자체 규약의 의미

다음 표는 이 설명서에서 사용되는 활자체 규약에 대해 설명합니다.

표 P-1 활자체 규약

서체 또는 기호	의미	예
AaBbCc123	명령어, 파일 및 디렉토리 이름 등 컴퓨터 화면에 출력되는 내용입니다.	.login 파일을 편집하십시오. 모든 파일 목록을 보려면 <code>ls -a</code> 를 사용합니다. <code>machine_name% you have mail.</code>
<b>AaBbCc123</b>	사용자가 입력하는 내용으로 컴퓨터 화면의 출력 내용과 대조됩니다.	<code>machine_name% su</code> <code>Password:</code>
<i>AaBbCc123</i>	명령줄 자리 표시자: 실제 이름이나 값으로 바꾸십시오.	파일을 삭제하려면 <b>rm</b> 파일 이름을 입력합니다.
<i>AaBbCc123</i>	책 제목, 새로운 단어나 용어 또는 강조할 단어	<b>사용자 설명서</b> 의 6장을 읽으십시오. 이는 <i>class</i> 옵션이라고 합니다. 아직 변경 사항을 저장하지 <b>마</b> 십시오.

## 명령 예의 셸 프롬프트

C 셸, Bourne 셸 및 Korn 셸에 대한 기본 시스템 프롬프트 및 슈퍼유저(또는 관리) 프롬프트는 다음 표와 같습니다.

표 P-2 셸 프롬프트

셸	프롬프트
C 셸 프롬프트	machine_name%
C 셸 슈퍼유저 프롬프트	machine_name#
Bourne 셸 및 Korn 셸 프롬프트	\$
Bourne 셸 및 Korn 셸 슈퍼유저 프롬프트	#

# 이동식 매체 관리(개요)

---

이 장은 Oracle Solaris OS에서 이동식 매체 관리에 대한 일반 지침을 제공합니다.

이 장에서 다루는 개요 정보는 다음과 같습니다.

- 21 페이지 “이동식 매체의 새로운 기능”
- 24 페이지 “이동식 매체 관리 작업을 찾을 수 있는 위치”
- 25 페이지 “이동식 매체의 기능 및 이점”
- 25 페이지 “수동 및 자동 마운트 비교”
- 26 페이지 “이동식 매체에 액세스 개요”

## 이동식 매체의 새로운 기능

### 이동식 매체 관리에 대해 변경되거나 향상된 기능

**Oracle Solaris 11:** 이동식 매체 관리를 위한 이전 기능이 제거되고 더 나은 이동식 매체 관리 기능을 제공하는 서비스 및 방법으로 대체되었습니다.

다음은 새로 제공되는 기능입니다.

- 새 이동식 매체 서비스는 SMF를 통해 관리합니다.

```
online      12:17:54 svc:/system/hal:default
online      12:17:56 svc:/system/filesystem/rmvolmgr:default
online      12:17:26 svc:/system/dbus:default
```

- 이제 /media 디렉토리에 이동식 매체가 자동으로 마운트됩니다. 하지만 호환성을 위해 이전에 사용된 매체 마운트 지점인 /cdrom 및 /rmdisk에서 /media에 대한 심볼릭 링크가 제공됩니다.

다음은 CF(Compact Flash) 메모리 카드(/dev/dsk/c4d0p0:1)를 마운트하는 예입니다.

```
$ ls /media/NIKON
```

다음은 USB 메모리 스틱(/dev/dsk/c3t0d0s0)을 마운트하는 예입니다.

```
$ ls /media/U3
```

- 기본 이동식 매체 볼륨 관리자 `rmvolmgr`는 다음과 같은 작업을 담당합니다.
  - 볼륨 마운트 및 마운트 해제
  - `rmvolmgr`의 루트 인스턴스는 시스템 부트 시 시작됩니다. 하지만 로그인할 때 `rmvolmgr`의 인스턴스를 시작하도록 세션 구성 파일을 구성할 수 있습니다. `rmvolmgr`을 사용자 세션 내에 실행할 경우 현재 사용자 또는 세션에서 소유한 장치만 마운트되므로 루트 인스턴스와 충돌하지 않습니다.
  - `rmvolmgr`을 종료하면 마운트된 모든 매체가 마운트 해제됩니다.
  - `rmvolmgr`은 호환성을 위해 `/cdrom` 또는 `/rmdisk` 디렉토리에 실제 마운트 지점인 `/media`에 대한 심볼릭 링크를 만듭니다.
  - CDE 호환을 위해 특수 `rmvolmgr` 실행 모드를 사용할 수 있습니다.
- HAL(하드웨어 추상화 계층) 데몬인 `hald`는 시스템에 연결된 장치를 표시하는 보기를 제공합니다. 이 보기는 핫 플러그 또는 기타 방식을 통해 하드웨어 구성이 변경되면 자동으로 업데이트됩니다.

HAL은 각 하드웨어를 장치 개체로 나타냅니다. 장치 개체는 UDI(고유 장치 식별자)로 식별되며 장치 등록 정보라는 일련의 키 값 쌍을 갖고 있습니다. 등록 정보는 실제 하드웨어에서 파생되기도 하고, 장치 정보 파일(.fdi 파일)에서 병합되기도 하며, 실제 장치 구성과 관련된 등록 정보도 있습니다.

다음은 제거된 기능입니다.

- 이 릴리스에서는 USB 디스켓만 지원됩니다.
- `vold` 데몬, `volfs` 파일 시스템 및 `volfs` 서비스가 제거되었습니다.

```
svc:/system/filesystem/volfs
```

- `/vol` 디렉토리 아래의 이동식 매체에 대한 논리 장치 이름(예: `/vol/dev/rdisk/...` 또는 `/vol/dev/aliases/...`)은 이제 제공되지 않습니다.

논리 장치 이름으로 이동식 매체에 액세스하려면 `/dev` 장치를 사용해야 합니다. 예를 들면 다음과 같습니다.

```
/dev/rdisk/c0t6d0s2
```

- 일부 `vold` 장치 별명은 이제 사용할 수 없습니다. 다음 `eject -l` 출력을 통해 각 장치에 사용할 수 있는 장치 별명과 예에서 마운트된 매체 경로 이름(`/media/SOL_11_X86_4`)을 확인할 수 있습니다.

```
$ eject -l
/dev/dsk/c2t0d0s2    cdrom,cdrom0,cd,cd0,sr,sr0,SOL_11_X86_4,/media/SOL_11_X86_4
```

각 장치를 꺼내는 데 사용할 수 있는 별명이 쉽표로 구분된 목록으로 표시됩니다.

- `vold.conf` 및 `rmmount.conf` 구성 파일이 이제 존재하지 않기 때문에 이러한 파일에 지정된 사용자 정의를 사용할 수 없습니다. 매체 사용자 정의 관리에 대한 자세한 내용은 24 페이지 “이동식 매체 관리 사용자 정의”를 참조하십시오.
- `volcheck`와 `volrmmount`를 제외한 `vol*`로 시작하는 명령.

## 이전 버전과의 호환성

다음 기능은 이전 Oracle Solaris 이동식 매체 기능과의 역 호환성을 제공합니다.

- 이동식 매체 마운트 지점이 CD-ROM 및 USB 장치와 같은 이동식 매체를 마운트하는 데 사용되는 /media 디렉토리로 이동되었습니다. 하지만 호환성을 위해 /cdrom 및 /rmdisk와 같은 이전 매체 마운트 지점에서 /media에 대한 심블릭 링크가 제공됩니다.
- `rmformat` 명령은 계속 사용할 수 있습니다. 이 명령의 출력은 이전 Solaris 릴리스에서 `vold`를 비활성화했을 때 출력되는 형식과 동일합니다.

예를 들면 다음과 같습니다.

```
# rmformat
Looking for devices...
  1. Logical Node: /dev/rdisk/c0t6d0s2
     Physical Node: /pci@1f,4000/scsi@3/sd@6,0
     Connected Device: TOSHIBA DVD-ROM SD-M1401 1009
     Device Type: DVD Reader
     Bus: SCSI
     Size: 2.9 GB
     Label: <None>
     Access permissions: <Unknown>
```

- 더욱 강력해진 `eject` 명령을 사용할 수 있습니다. 자세한 내용은 23 페이지 “이동식 매체 꺼내기”를 참조하십시오.

## 이동식 매체 마운트 및 마운트 해제

이 릴리스에서는 `vol*`로 시작하는 대부분의 명령이 제거되었습니다. 수정된 버전의 `rmmount` 및 새 `rmumount` 명령을 사용하여 이동식 매체를 마운트 및 마운트 해제할 수 있습니다.

이러한 명령을 통해 장치 이름, 레이블 또는 마운트 지점을 사용하여 마운트할 수 있습니다. 다음은 iPod을 마운트하는 예입니다.

```
% rmmount ipod
```

다음은 DVD에서 파일 시스템을 마운트 해제하는 예입니다.

```
# rmumount cdrom
cdrom /dev/dsk/c0t6d0s5 unmounted
cdrom /dev/dsk/c0t6d0s0 unmounted
```

자세한 내용은 `rmmount(1M)`를 참조하십시오.

## 이동식 매체 꺼내기

이전 Solaris 릴리스에서와 마찬가지로 `eject` 명령을 사용하여 이동식 매체를 마운트 해제하고 꺼낼 수 있습니다. 그러나 이번 릴리스에서는 다음과 같은 `eject` 옵션을 사용할 수 있습니다.

`-f`    장치가 사용 중이더라도 강제로 꺼냅니다.

- l    꺼낼 수 있는 장치의 경로 및 별명을 표시합니다.
- t    CD-ROM 트레이 달기 명령이 장치에 제공됩니다. 일부 장치에는 이 옵션이 지원되지 않습니다.

예를 들어, 볼륨 레이블을 꺼내려면 다음과 같이 합니다.

```
% eject mypictures
```

자세한 내용은 `eject(1)`를 참조하십시오.

## 이동식 매체 관리 사용자 정의

`vold.conf` 및 `rmmount.conf` 파일에서 사용할 수 있었던 대부분의 사용자 정의에 대해 데스크탑 볼륨 관리자 기본 설정을 사용하거나 `.fdi` 파일을 수정해야 합니다.

- `rmmount.conf` 작업의 경우 데스크탑 볼륨 관리자 작업인 `gconf` 또는 HAL 호출을 사용해야 합니다.
- 이전에는 일반 사용자를 대신하여 `rmmount.conf` 작업을 루트로 실행할 수 있었습니다. 이제 이 작업은 `/usr/lib/hal` 디렉토리에 호출 실행 파일을 설치함으로써 수행됩니다.

## 이동식 매체 기능 비활성화

볼륨이 사용자 세션 외부에서 마운트되지 않도록 하려면 `rmvolmgr` 서비스를 비활성화합니다. 예:

```
# svcadm disable rmvolmgr
```

# 이동식 매체 관리 작업을 찾을 수 있는 위치

이동식 매체 관리를 위한 단계별 지침을 찾으려면 다음 참조 자료를 사용합니다.

이동식 매체 관리 작업	자세한 정보
이동식 매체 포맷	2 장, “이동식 매체 관리(작업)”
이동식 매체에 액세스	3 장, “이동식 매체에 액세스(작업)”
데이터 CD, DVD 및 음악 CD 쓰기	4 장, “CD 및 DVD에 쓰기(작업)”



## 이동식 매체의 기능 및 이점

Oracle Solaris 릴리스는 사용자 및 소프트웨어 개발자에게 이동식 매체를 다루기 위한 표준 인터페이스를 제공합니다. 이동식 매체 서비스는 다음과 같은 이점을 제공합니다.

- 이동식 매체를 자동으로 마운트합니다. 수동 및 자동 마운트를 비교한 내용은 다음 섹션을 참조하십시오.
- 관리자로 로그인할 필요 없이 이동식 매체에 액세스할 수 있습니다.
- 네트워크의 다른 시스템에 로컬 시스템의 모든 이동식 매체에 대한 자동 액세스 권한을 부여하도록 허용할 수 있습니다. 자세한 내용은 3 장, “이동식 매체에 액세스(작업)”를 참조하십시오.

## 수동 및 자동 마운트 비교

다음 표는 이동식 매체의 수동 마운트(이동식 매체 서비스 기능 없음) 및 자동 마운트(이동식 매체 관리 기능 있음)와 관련된 단계를 비교합니다.

표 1-1 이동식 매체의 수동 및 자동 마운트 비교

단계	수동 마운트	자동 마운트
1	매체를 삽입합니다.	매체를 삽입합니다.
2	관리자로 전환합니다.	USB 디스켓의 경우 <code>volcheck</code> 명령을 사용합니다.
3	매체 장치의 위치를 확인합니다.	이동식 매체 서비스는 이동식 매체를 수동으로 마운트하고 사용하는 데 필요한 여러 작업을 자동으로 수행합니다.
4	마운트 지점을 만듭니다.	
5	마운트 지점 디렉토리에 없는지 확인합니다.	
6	장치를 마운트하고 적절한 <code>mount</code> 옵션을 사용합니다.	
7	관리자 계정을 종료합니다.	
8	매체에서 파일을 사용하여 작업합니다.	매체에서 파일을 사용하여 작업합니다.
9	관리자로 전환합니다.	
10	매체 장치를 마운트 해제합니다.	
11	매체를 꺼냅니다.	매체를 꺼냅니다.
12	관리자 계정을 종료합니다.	

## 이동식 매체에 액세스 개요

근본적으로 이동식 매체 서비스를 사용하여 수동 마운트처럼 이동식 매체에 액세스할 수 있지만 더욱 쉬울 뿐만 아니라 관리자 계정이 없어도 됩니다.

매체에 파일 시스템 및 레이블이 포함된 경우 매체 레이블 이름은 `/media/pathname` 마운트 지점의 이름을 지정하는 데 사용됩니다. 레이블이 존재하지 않는 경우에는 디스크 모델 이름이 매체의 이름을 지정하는 데 사용됩니다(예: `/media/cdrom`). 일반 `nickname`은 레거시 심볼릭 링크에 대해서만 사용됩니다. 예를 들어 `/rmdisk/rmdisk0`입니다.

시스템에 있는 이동식 매체의 유형이 두 개 이상인 경우 매체의 액세스 지점은 다음 표를 참조하십시오.

표 1-2 이동식 매체의 데이터에 액세스하는 방법

액세스	삽입	레이블이 지정되지 않은 매체 경로 이름	레이블이 지정된 매체 경로 이름의 예
이동식 하드 디스크에 있는 파일	이동식 하드 디스크 및 명령줄에 <code>volcheck</code> 입력	<code>/media/usb-disk</code> 또는 레거시 경로 <code>/rmdisk/rmdisk0</code>	<code>/media/00JB-00CRA0</code>
DVD에 있는 파일	DVD 및 몇 초 동안 대기	<code>/media/cdrom</code>	<code>/media/sol_10_sparc</code>
DVD에 있는 파일	DVD 및 몇 초 동안 대기	<code>/media/cdrom</code>	<code>/media/SOL_11_X86</code>

`rmmount -l` 명령을 사용하여 시스템에 마운트된 매체를 식별할 수 있습니다. 예를 들면 다음과 같습니다.

```
# rmmount -l
/dev/dsk/c5t0d0p0      rmdisk6,/media/FD-05PUB
/dev/dsk/c4t0d3p0      rmdisk5,/media/223UHS-SD-MMC
/dev/dsk/c2t0d0s2      cdrom1,cd1,sr1,SOL_11_X86,/media/SOL_11_X86
/dev/dsk/c3t0d0p0      rmdisk2,/media/00JB-00CRA0
```

위 출력에서 마운트된 장치는 다음과 같습니다.

```
/dev/dsk/c5t0d0p0      USB 디스켓
/dev/dsk/c4t0d3p0      USB 카드 리더 내의 CF 카드
/dev/dsk/c2t0d0s2      DVD-ROM
/dev/dsk/c3t0d0p0      이동식 USB 디스크
```

## 이동식 매체 관리(작업)

이 장에서는 Oracle Solaris OS의 명령줄에서 이동식 매체를 관리하는 방법에 대해 설명합니다.

이동식 매체 관리와 관련된 절차에 대한 자세한 내용은 27 페이지 “이동식 매체 관리(작업 맵)”를 참조하십시오. 이동식 매체에 대한 배경 정보는 1 장, “이동식 매체 관리(개요)”를 참조하십시오.

### 이동식 매체 관리(작업 맵)

다음 작업 맵은 이동식 매체를 관리하는 데 필요한 작업에 대해 설명합니다.

작업	설명	수행 방법
1. 매체를 로드합니다.	디스켓을 드라이브에 넣고 <code>volcheck</code> 명령을 입력합니다.	29 페이지 “이동식 매체를 로드하는 방법”
2. (옵션) 디스켓을 포맷합니다.	디스켓을 포맷합니다.	30 페이지 “디스켓을 포맷하는 방법( <code>rmformat</code> )”
3. (옵션) PCFS 파일 시스템을 추가합니다.	파일 전송에 매체를 사용할 수 있도록 PCFS 파일 시스템을 추가합니다.	31 페이지 “이동식 매체에서 파일 시스템을 만드는 방법”
	DVD-RAM 장치에 UDFS 파일 시스템을 추가합니다.	32 페이지 “DVD-RAM에서 파일 시스템을 만드는 방법”
4. (옵션) 매체를 검사합니다.	매체에서 파일 시스템의 무결성을 확인합니다.	32 페이지 “이동식 매체에서 파일 시스템을 검사하는 방법”
5. (옵션) 매체에서 잘못된 블록을 복구합니다.	필요한 경우 매체에서 잘못된 모든 블록을 복구합니다.	33 페이지 “이동식 매체에서 잘못된 블록을 복구하는 방법”

작업	설명	수행 방법
6. (옵션) 읽기 또는 쓰기와 암호 보호를 적용합니다.	필요한 경우 매체에 대해 읽기 또는 쓰기와 암호 보호를 적용합니다.	33 페이지 “이동식 매체에서 쓰기 보호를 사용으로 설정하거나 사용 안함으로 설정하는 방법”

## 이동식 매체 준비

다음 섹션에서는 이동식 매체를 사용하기 위한 준비 방법을 설명합니다.

### 이동식 매체 고려 사항

디스켓으로 작업할 때 다음 고려 사항을 염두에 두십시오.

- SunOS 파일 시스템 형식은 SunOS 파일 시스템을 지원하기 위한 구조 이외에 기본 “비트” 형식으로 구성됩니다. DOS 파일 시스템에 대한 전체 형식은 MS-DOS 또는 NEC-DOS 파일 시스템을 지원하기 위한 구조 이외에 기본 “비트” 형식으로 구성됩니다. 각 유형의 파일 시스템에 대한 매체를 준비하는 데 필요한 절차는 다음과 같습니다. 따라서 디스켓을 포맷하기 전에 따라야 하는 절차를 고려하십시오. 자세한 내용은 27 페이지 “이동식 매체 관리(작업 맵)”를 참조하십시오.
  - 이동식 매체 이름에 대한 자세한 내용은 36 페이지 “이동식 매체 이름 사용”을 참조하십시오.
  - 이름이 지정되지 않은(즉, “레이블”이 없는) 디스켓에는 기본 이름인 `unnamed_floppy`가 지정됩니다.
  - 이름이 지정되지 않은(즉, “레이블”이 없는) 디스켓에는 기본 이름인 `floppy`가 지정됩니다.

Oracle Solaris 시스템에서는 다음 파일 시스템 유형을 포맷할 수 있습니다.

- ZFS 또는 UFS
- MS-DOS 또는 NEC-DOS(PCFS)
- UDFS

Oracle Solaris 시스템(SPARC 또는 x86)에서는 다음 밀도의 디스켓을 포맷할 수 있습니다.

디스켓 크기	디스켓 밀도	용량
3.5"	HD(High density)	1.44MB
3.5"	DD(Double density)	720KB

기본적으로 디스켓 드라이브에서는 디스켓을 같은 밀도로 포맷합니다. 이 기본값은 다르게 지정하지 않는 한 디스켓이 실제로 1.44MB 디스켓인지 여부에 상관없이 1.44MB 드라이브에서 디스켓을 1.44MB로 포맷을 시도함을 의미합니다. 달리 말하면 디스켓은 해당 용량 이하로 포맷될 수 있고, 드라이브에서는 디스켓을 해당 용량 이하로 포맷할 수 있습니다.

## 디스켓 포맷

rmformat 명령을 사용하여 다시 쓰기 가능한 디스켓을 포맷하고 보호할 수 있습니다. 파일 시스템은 자동으로 마운트됩니다. 따라서 매체에 기존 파일 시스템이 포함되어 있는 경우 포맷하기 전에 매체를 마운트 해제해야 할 수 있습니다.

rmformat 명령에는 세 가지 포맷 옵션이 있습니다.

- **quick** - 이 옵션은 매체의 특정 트랙에 대한 검증이나 제한된 검증 없이 디스켓을 포맷합니다.
- **long** - 이 옵션은 디스켓을 완전히 포맷합니다. 일부 장치의 경우 이 옵션을 사용하면 드라이브에서 전체 매체를 검증할 수 있습니다.
- **force** - 이 옵션은 사용자 확인 없이 완전히 포맷합니다. 암호로 보호된 디스켓의 경우 이 옵션은 포맷 전에 암호를 지웁니다. 이 기능은 암호를 잊은 경우 유용합니다. 암호로 보호되지 않은 디스켓에서 이 옵션은 long 포맷을 수행합니다.

## ▼ 이동식 매체를 로드하는 방법

이동식 매체 하드웨어 고려 사항에 대한 자세한 내용은 [28 페이지 “이동식 매체 고려 사항”](#)을 참조하십시오.

### 1 매체를 삽입합니다.

### 2 매체가 포맷되었는지 확인합니다.

확실치 않은 경우 [단계 3](#)에 설명된 대로 매체를 넣고 시스템 콘솔 창에서 상태 메시지를 확인합니다. 매체를 포맷해야 하는 경우 [30 페이지 “디스켓을 포맷하는 방법\(rmformat\)”](#)으로 이동하십시오.

### 3 (옵션) USB가 아닌 레거시 디스켓 장치를 사용 중인 경우 볼륨 관리에 알립니다.

```
$ volcheck -v
```

두 가지 상태 메시지가 가능합니다.

```
media was found
```

볼륨 관리에서 매체를 발견하고 [36 페이지 “이동식 매체 이름 사용”](#)에 설명된 디렉토리에 마운트를 시도합니다.

매체가 제대로 포맷된 경우 콘솔에 오류 메시지가 나타나지 않습니다.

매체가 포맷되지 않은 경우에도 “media was found” 메시지가 표시됩니다. 하지만 다음과 유사한 오류 메시지가 시스템 콘솔 창에 나타납니다.

```
fd0: unformatted diskette or no diskette in the drive
```

```
fd0: read failed (40 1 0)
```

```
fd0: bad format
```

볼륨 관리에서 매체를 마운트하려면 먼저 포맷해야 합니다. 자세한 내용은 2 장, “이동식 매체 관리(작업)”를 참조하십시오.

```
no media was found
```

볼륨 관리에서 매체를 발견하지 못했습니다. 매체를 제대로 넣었는지 확인하고 volcheck를 다시 실행합니다. 실패할 경우 매체가 손상되지 않았는지 확인합니다. 매체 수동 마운트를 시도해 볼 수도 있습니다.

#### 4 내용을 나열하여 매체가 마운트되었는지 확인합니다.

예를 들면 디스켓에 대해 다음을 수행합니다.

```
$ ls /media/floppy
lost+found myfiles
```

## ▼ 디스켓을 포맷하는 방법(rmformat)

rmformat 명령을 사용하여 디스켓을 포맷할 수 있습니다. 기본적으로 이 명령은 매체에 분할 영역 0 및 분할 영역 2의 두 분할 영역을 만듭니다(전체 매체).

#### 1 이동식 매체 서비스가 실행 중인지 확인합니다. 그러한 경우 장치 이름에 대해 짧은 별명을 사용할 수 있습니다.

```
# svcs hal dbus rmvolmgr
STATE      STIME     FMRI
online     Apr_09    svc:/system/dbus:default
online     Apr_09    svc:/system/hal:default
online     Apr_09    svc:/system/filesystem/rmvolmgr:default
```

이동식 매체 서비스 다시 시작에 대한 자세한 내용은 37 페이지 “이동식 매체 서비스를 사용으로 설정 또는 사용 안함으로 설정하는 방법”을 참조하십시오. 매체 장치 이름 식별에 대한 자세한 내용은 36 페이지 “이동식 매체 이름 사용”을 참조하십시오.

#### 2 디스켓을 포맷합니다.

```
$ rmformat -F [ quick | long | force ] device-name
```

rmformat 포맷 옵션에 대한 자세한 내용은 29 페이지 “디스켓 포맷”을 참조하십시오.

rmformat 출력에 잘못된 블록이 있음을 나타낼 경우 33 페이지 “이동식 매체에서 잘못된 블록을 복구하는 방법”을 참조하십시오.

### 3 (옵션) 8자 레이블로 디스켓의 레이블을 지정합니다.

```
$ rmformat -b label device-name
```

DOS 레이블 작성에 대한 자세한 내용은 `mkfs_pcfs(1M)`을 참조하십시오.

## 예 2-1 디스켓 포맷

이 예에서는 디스켓을 포맷하는 방법을 보여 줍니다.

```
$ rmformat -F quick /dev/rdiskette
Formatting will erase all the data on disk.
Do you want to continue? (y/n) y
.....
```

## ▼ 이동식 매체에서 파일 시스템을 만드는 방법

### 1 (옵션) 필요한 경우 디스켓을 포맷합니다.

USB 디스켓을 포맷하려면 다음과 유사한 구문을 사용합니다.

```
$ rmformat -F long /dev/rdisk/c11t0d0p0
```

### 2 (옵션) 대체 Solaris 분할 영역 테이블을 만듭니다.

```
$ rmformat -s slice-file device-name
```

샘플 슬라이스 파일은 다음과 유사하게 나타납니다.

```
slices: 0 = 0, 30MB, "wm", "home" :
         1 = 30MB, 51MB :
         2 = 0, 94MB, "wm", "backup" :
         6 = 81MB, 13MB
```

### 3 관리자로 전환합니다.

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

### 4 적당한 파일 시스템 유형을 결정하고 다음 중 하나를 선택합니다.

- PCFS 파일 시스템을 만듭니다. 예를 들면 다음과 같습니다.

```
# mkfs -F pcfs -o nofdisk,size=9800 /dev/rdisk/c11t0d0p0
```

- UDFS 파일 시스템을 만듭니다. 예를 들면 다음과 같습니다.

```
# mkfs -F udfs /dev/rdisk/c0t1d0p0
```

## ▼ DVD-RAM에서 파일 시스템을 만드는 방법

DVD-RAM에서 파일 시스템을 만들려면 다음 절차를 따릅니다.

### 1 관리자로 로그인합니다.

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

### 2 DVD-RAM 장치에서 파일 시스템을 만듭니다.

- UDFS 파일 시스템을 만듭니다. 예를 들면 다음과 같습니다.

```
# mkfs -F udfs /dev/rdisk/c0t0d0s2
```

### 3 파일 시스템을 마운트합니다.

- UDFS 파일 시스템을 마운트합니다. 예를 들면 다음과 같습니다.

```
# mount -F udfs /dev/dsk/c0t0d0s2 /mnt
```

### 4 파일 시스템에 대해 읽고 쓸 수 있는지 확인합니다.

### 5 해당 작업을 완료하면 DVD-RAM을 꺼냅니다.

## ▼ 이동식 매체에서 파일 시스템을 검사하는 방법

### 1 관리자로 전환합니다.

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

### 2 파일 시스템 유형을 확인하고 다음 중 하나를 선택합니다.

- UDFS 파일 시스템을 검사합니다.

```
# fsck -F udfs device-name
```

- PCFS 파일 시스템을 검사합니다.

```
# fsck -F pcfs device-name
```

### 예 2-2 이동식 매체에서 PCFS 파일 시스템 검사

다음 예는 매체에서 PCFS 파일 시스템의 일관성을 검사하는 방법을 보여 줍니다.

```
# fsck -F pcfs /dev/rdisk/c0t4d0s2
** /dev/rdisk/c0t4d0s2
** Scanning file system meta-data
```



```

** Correcting any meta-data discrepancies
1457664 bytes.
0 bytes in bad sectors.
0 bytes in 0 directories.
0 bytes in 0 files.
1457664 bytes free.
512 bytes per allocation unit.
2847 total allocation units.
2847 available allocation units.

```

## ▼ 이동식 매체에서 잘못된 블록을 복구하는 방법

드라이브가 잘못된 블록 관리를 지원하는 경우 확인하는 동안 발견된 불량 섹터를 확인, 분석 및 복구하려면 `rmformat` 명령만 사용할 수 있습니다. 대부분의 USB 메모리 스틱은 잘못된 블록 관리를 지원하지 않습니다.

드라이브가 잘못된 블록 관리를 지원하는 경우 최선의 방법은 잘못된 블록을 수정하는 것입니다. 최선의 방식에도 불구하고 잘못된 블록을 수정할 수 없는 경우 매체 복구 실패를 나타내는 메시지가 표시됩니다.

### 1 이동식 매체에서 잘못된 블록을 복구합니다.

```
$ rmformat -c block-numbers device-name
```

이전 `rmformat` 세션에서는 10진수, 8진수 또는 16진수 형식으로 블록 수를 표시합니다.

### 2 매체를 확인합니다.

```
$ rmformat -V read device-name
```

## 이동식 매체에 읽기 또는 쓰기 보호와 암호 보호 적용

읽기 또는 쓰기 보호 기능이 제공되는 이동식 매체에서 이러한 기능을 적용하고 암호를 설정할 수 있습니다.

## ▼ 이동식 매체에서 쓰기 보호를 사용으로 설정하거나 사용 안함으로 설정하는 방법

### 1 쓰기 보호의 사용으로 설정 또는 사용 안함으로 설정 여부를 결정하고 다음 중 하나를 선택합니다.

- 쓰기 보호를 사용 안함으로 설정합니다.

```
$ rmformat -w enable device-name
```

- 쓰기 보호를 사용 안함으로 설정합니다.

```
$ rmformat -w disable device-name
```

- 2 매체의 쓰기 보호가 사용으로 설정 또는 사용 안함으로 설정되었는지 확인합니다.

```
$ rmformat -p device-name
```

## ▼ 이동식 매체에서 읽기 또는 쓰기 보호를 사용으로 설정하거나 사용 안함으로 설정하고 암호를 설정하는 방법

이러한 기능이 지원되는 이동식 매체에 대해 최대 32자의 암호를 적용할 수 있습니다.

이 기능이 지원되지 않는 이동식 매체에서 암호를 적용하려고 하면 경고 메시지가 표시됩니다.

- 1 읽기 보호 또는 쓰기 보호를 사용으로 설정하거나 사용 안함으로 설정하고 암호를 설정할지 결정합니다. 다음 중 하나를 선택합니다.

- 읽기 보호 또는 쓰기 보호를 사용으로 설정합니다.

```
$ rmformat -W enable device-name
Please enter password (32 chars maximum): xxx
Please reenter password:
```

```
$ rmformat -R enable device-name
Please enter password (32 chars maximum): xxx
Please reenter password:
```

- 읽기 보호 또는 쓰기 보호를 사용 안함으로 설정하고 암호를 제거합니다.

```
$ rmformat -W disable device-name
Please enter password (32 chars maximum): xxx
```

```
$ rmformat -R disable device-name
Please enter password (32 chars maximum): xxx
```

- 2 매체의 읽기 보호 또는 쓰기 보호가 사용으로 설정되었거나 사용 안함으로 설정되었는지 확인합니다.

```
$ rmformat -p device-name
```

## 이동식 매체에 액세스(작업)

이 장에서는 Oracle Solaris OS의 명령줄에서 이동식 매체에 액세스하는 방법에 대해 설명합니다.

이동식 매체 액세스와 관련된 절차에 대한 자세한 내용은 다음을 참조하십시오.

- 35 페이지 “이동식 매체에 액세스(작업 맵)”
- 40 페이지 “원격 시스템에서 이동식 매체에 액세스(작업 맵)”

이동식 매체에 대한 배경 정보는 1 장, “이동식 매체 관리(개요)”를 참조하십시오.

### 이동식 매체에 액세스(작업 맵)

다음 작업 맵은 이동식 매체에 액세스하는데 필요한 작업에 대해 설명합니다.

작업	설명	수행 방법
1. (옵션) 이동식 매체 드라이브를 추가합니다.	필요한 경우 이동식 매체 드라이브를 시스템에 추가합니다.	36 페이지 “새 이동식 매체 드라이브를 추가하는 방법”
2. (옵션) 볼륨 관리(vold) 사용 여부와 관계없이 이동식 매체를 사용할지 결정합니다.	볼륨 관리(vold)는 기본적으로 실행됩니다. 볼륨 관리 사용 여부와 관계없이 이동식 매체를 사용할지 결정합니다.	37 페이지 “이동식 매체 서비스를 사용하여 설정 또는 사용 안함으로 설정하는 방법”
3. 이동식 매체에 액세스합니다.	볼륨 관리 실행 여부에 관계없이 다른 종류의 이동식 매체에 액세스합니다.	38 페이지 “이동식 매체에 대한 정보에 액세스하는 방법”
4. (옵션) 파일 또는 디렉토리를 복사합니다.	파일 시스템의 다른 위치에서처럼 매체에서 파일 또는 디렉토리를 복사합니다.	38 페이지 “이동식 매체에서 정보를 복사하는 방법”

작업	설명	수행 방법
5. 매체가 아직 사용 중인지 확인합니다.	매체를 꺼내기 전에 매체가 아직 사용 중인지 확인합니다.	39 페이지 “이동식 매체가 아직 사용 중인지 확인하는 방법”
6. 매체를 꺼냅니다.	해당 작업을 완료하면 장치에서 매체를 꺼냅니다.	39 페이지 “이동식 매체를 꺼내는 방법”

## 이동식 매체에 액세스

블룸 관리 사용 여부에 관계없이 이동식 매체에 대한 정보에 액세스할 수 있습니다. GNOME의 File Manager를 사용하여 이동식 매체 관련 정보에 액세스하는 방법에 대한 자세한 내용은 GNOME 데스크탑 설명서를 참조하십시오.

### 이동식 매체 이름 사용

이제 /media 디렉토리에 이동식 매체가 자동으로 마운트됩니다. 하지만 호환성을 위해 이전에 사용된 매체 마운트 지점인 /cdrom 및 /rmdisk에서 /media에 대한 심블릭 링크가 제공됩니다.

다음은 CF(Compact Flash) 메모리 카드(/dev/dsk/c4d0p0:1)를 마운트하는 예입니다.

```
$ ls /media/NIKON
```

다음은 USB 메모리 스틱(/dev/dsk/c3t0d0s0)을 마운트하는 예입니다.

```
$ ls /media/U3
```

### 이동식 매체 데이터에 액세스하기 위한 지침

대부분의 DVD는 이식 가능한 ISO 9660 표준으로 포맷되어 있습니다. 따라서 블룸 관리를 통해 DVD를 마운트할 수 있습니다.

가능한 다른 형식을 수용할 수 있도록 DVD는 슬라이스로 분할됩니다. 슬라이스는 호환 면에서 하드 디스크의 분할 영역과 유사합니다. 9660 부분은 이식할 수 있습니다. DVD를 마운트하는 데 문제가 있는 경우(특히 설치 DVD인 경우) 파일 시스템이 시스템의 아키텍처에 적합한지 확인합니다. 예를 들어 DVD 레이블을 확인할 수 있습니다.

## ▼ 새 이동식 매체 드라이브를 추가하는 방법

일반적으로 대부분의 현대적 버스 유형은 핫 플러그를 지원합니다. 즉, 빈 슬롯에 디스크를 삽입할 수 있고 시스템에서 이러한 디스크를 인식합니다.

핫 플러그 장치에 대한 자세한 내용은 6 장, “동적으로 장치 구성(작업)”을 참조하십시오.

- 1 관리자로 로그인합니다.
- 2 새 매체 드라이브를 연결합니다.  
구체적인 지침은 하드웨어 설명서를 참조하십시오.
- 3 시스템에 새 매체 드라이브가 표시되는지 확인합니다.

```
# rmformat
Looking for devices...
```

## ▼ 이동식 매체 서비스를 사용으로 설정 또는 사용 안함으로 설정하는 방법

간혹 이동식 매체 서비스를 사용하지 않고 매체를 관리하고자 하는 경우도 있습니다. 이 섹션에서는 이동식 매체 서비스를 사용으로 설정 및 사용 안함으로 설정하는 방법을 설명합니다.

이러한 서비스를 비활성화하면 `mount` 명령을 사용하여 모든 매체를 수동으로 마운트해야 합니다.

- 1 매체가 사용 중이 아닌지 확인합니다.  
매체의 모든 사용자를 찾았는지 확실하지 않은 경우 `fuser` 명령을 사용합니다(39 페이지 “이동식 매체가 아직 사용 중인지 확인하는 방법” 참조).
- 2 관리자로 전환합니다.
- 3 다음 중 하나를 선택합니다.
  - 이 릴리스에서는 이동식 매체 관련 기능 중 일부 또는 전체를 비활성화할 수 있습니다.
    - 볼륨이 사용자 세션 외부에서 마운트되지 않도록 하려면 `rmvolmgr` 서비스를 비활성화합니다. 예를 들면 다음과 같습니다.
 

```
# svcadm disable rmvolmgr
```
    - 모든 볼륨 관리를 차단하려면 `dbus`, `hal` 및 `rmvolmgr` 서비스를 사용 안함으로 설정합니다.
 

```
# svcadm disable rmvolmgr
# svcadm disable dbus
# svcadm disable hal
```
  - 이러한 서비스를 비활성화하면 `mount` 명령을 사용하여 모든 매체를 수동으로 마운트해야 합니다.
  - 이동식 매체 서비스를 사용으로 설정합니다.

```
# svcadm enable rmvolmgr
# svcadm enable dbus
# svcadm enable hal
```

## ▼ 이동식 매체에 대한 정보에 액세스하는 방법

- 1 매체를 삽입합니다.  
몇 초 뒤에 매체가 마운트됩니다.
- 2 매체의 내용이 나열됩니다.

```
% ls /media
```

### 예 3-1 이동식 매체에 대한 정보에 액세스

다음 예는 USB 메모리 스틱에 대한 정보에 액세스하는 방법을 보여 줍니다.

```
$ ls /media/usb-name
```

다음 예는 DVD에 대한 정보에 액세스하는 방법을 보여 줍니다.

```
$ ls /media
sol_10_811_sparc cdrom
```

## ▼ 이동식 매체에서 정보를 복사하는 방법

임의의 다른 파일 시스템에서처럼 이동식 매체에서 파일 및 디렉토리에 액세스 할 수 있습니다. 유일한 중요 제한 사항은 소유권 및 권한과 관련이 있습니다.

예를 들어 DVD에서 파일 시스템으로 파일을 복사하면 소유자가 됩니다. 그러나 DVD에 있는 파일에는 쓰기 권한이 없으므로 쓰기 권한을 가질 수 없습니다. 사용자가 직접 권한을 변경해야 합니다.

- 1 매체가 마운트되었는지 확인합니다.

```
$ ls /media
```

ls 명령은 마운트된 매체의 내용을 표시합니다. 표시된 내용이 없는 경우 [38 페이지](#) “이동식 매체에 대한 정보에 액세스하는 방법”을 참조하십시오.

- 2 (옵션) 파일 또는 디렉토리를 복사합니다.

예를 들어 DVD의 경우 다음을 수행합니다.

```
$ cp /media/sol_sparc/Solaris_11/file .
$ ls -l
-rwxr-xr-x  1 pmorph  gelfs  64065 Aug 2  2010 file
```

## ▼ 이동식 매체가 아직 사용 중인지 확인하는 방법

- 1 관리자 로그인합니다.
- 2 매체에 액세스 중인 프로세스를 식별합니다.  

```
# fuser -u /media
```

-u는 매체의 사용자를 표시합니다.  
자세한 내용은 [fuser\(1M\)](#)를 참조하십시오.
- 3 (옵션) 매체에 액세스 중인 프로세스를 종료합니다.  

```
# fuser -u -k /media
```

-k는 매체에 액세스 중인 프로세스를 종료시킵니다.



주의 - 매체에 액세스 중인 프로세스 종료는 비상 상황에만 사용해야 합니다.

- 4 프로세스가 종료되었는지 확인합니다.  

```
# pgrep process-ID
```

### 예 3-2 매체가 아직 사용 중인지 확인

다음 예는 /media/sol\_10\_1008\_sparc/Solaris\_10/Tools 디렉토리에 액세스하는 사용자 pmorph를 보여 줍니다.

```
# fuser -u /media/sol_10_1008_sparc/Solaris_10/Tools
/media/sol_10_1008_sparc/Solaris_10/Tools:      723c(pmorph)      316c(pmorph)
```

## ▼ 이동식 매체를 꺼내는 방법

- 1 매체가 사용 중이 아닌지 확인합니다.  
셀 또는 응용 프로그램이 매체의 파일 또는 디렉토리 중 하나에 액세스하고 있으면 매체가 “사용 중”입니다. DVD의 사용자를 모두 찾았는지 확실하지 않은 경우(예를 들어 데스크탑 도구 뒤에 숨겨진 셀이 DVD에 액세스 중일 수 있음) [fuser](#) 명령을 사용합니다. [39 페이지 “이동식 매체가 아직 사용 중인지 확인하는 방법”](#)을 참조하십시오.
- 2 매체를 꺼냅니다.  

```
# eject media
```

예를 들어 DVD의 경우 다음을 수행합니다.  

```
# eject cdrom
```

예를 들어 USB 메모리 스틱의 경우 다음을 수행합니다.

```
# eject rmdisk0
```

참고 - eject -l 명령을 사용하면 이동식 장치 이름을 볼 수 있습니다.

## 원격 시스템에서 이동식 매체에 액세스(작업 맵)

다음 작업 맵은 원격 시스템에서 이동식 매체에 액세스하는 데 필요한 작업에 대해 설명합니다.

작업	설명	수행 방법
1. 원격 시스템에서 사용할 수 있도록 로컬 매체를 설정합니다.	매체 드라이브를 공유하도록 시스템을 구성하여 다른 시스템에서 매체 드라이브의 모든 매체를 사용하도록 설정합니다.	40 페이지 “다른 시스템에서 사용할 수 있도록 로컬 매체를 설정하는 방법”
2. 원격 시스템에서 이동식 매체에 액세스합니다.	로컬 시스템에서 원격 매체에 액세스합니다.	38 페이지 “이동식 매체에 대한 정보에 액세스하는 방법”

### ▼ 다른 시스템에서 사용할 수 있도록 로컬 매체를 설정하는 방법

매체 드라이브를 공유하도록 시스템을 구성하여 다른 시스템에서 매체 드라이브의 모든 매체를 사용하도록 설정할 수 있습니다. 음악 CD는 예외입니다. 매체 드라이브를 공유하면 이러한 드라이브를 마운트하여 다른 시스템에서 해당 드라이브에 포함된 매체에 액세스할 수 있습니다. 지침은 41 페이지 “원격 시스템에서 이동식 매체에 액세스하는 방법”을 참조하십시오.

- 1 관리자로 전환합니다.
- 2 매체가 로드되었는지 확인합니다.
- 3 매체를 공유합니다.  
예를 들면 다음과 같습니다.  

```
# share -F nfs -o ro /media/so1_10_811_sparc
```
- 4 NFS 서버 서비스가 실행 중인지 확인합니다.  

```
# svcs *nfs*
```



NFS 서버 서비스가 실행되면 `svcs` 명령으로부터 다음 출력이 반환됩니다.

```
online          14:28:43 svc:/network/nfs/server:default
```

**5 NFS 서버 상태를 식별하고 다음 중 하나를 선택합니다.**

- NFS 서버 서비스가 실행된 경우 **단계 7**로 이동합니다.
- NFS 서버 서비스가 실행되지 **않은** 경우 다음 단계로 이동합니다.

**6 NFS 서버 서비스를 시작합니다.**

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

NFS 데몬이 실행 중인지 확인합니다.

예를 들면 다음과 같습니다.

```
# svcs -p svc:/network/nfs/server:default
STATE      STIME      FMRI
online     Aug_30    svc:/network/nfs/server:default
           Aug_30    319 mountd
           Aug_30    323 nfsd
```

**7 다른 시스템에서 해당 매체를 사용할 수 있는지 확인합니다.**

매체를 사용할 수 있는 경우 공유 구성이 표시됩니다.

```
# share
-          /media/sol_10_811_sparc  sec=sys,ro  ""
```

### 예 3-3 다른 시스템에서 사용할 수 있도록 로컬 DVD 설정

다음 예는 네트워크의 다른 시스템에서 사용할 수 있도록 로컬 DVD를 설정하는 방법을 보여 줍니다.

```
# share -F nfs -o ro /media
# svcs *nfs*
# svcadm enable network/nfs/server
# svcs -p svc:/network/nfs/server:default
# share
-          /media/sol_10_811_sparc  ro  ""
```

## ▼ 원격 시스템에서 이동식 매체에 액세스하는 방법

파일 시스템으로 매체를 마운트하여 원격 시스템에서 매체에 액세스할 수 있습니다. 또한 원격 시스템에서 **40 페이지** “다른 시스템에서 사용할 수 있도록 로컬 매체를 설정하는 방법”의 지침에 따라 매체를 공유해야 합니다.

**1 마운트 지점의 역할을 할 기존 디렉토리를 선택합니다. 또는 마운트 지점을 만듭니다.**

```
$ mkdir /directory
```

여기서 */directory*는 원격 시스템 DVD의 마운트 지점 역할을 하도록 만든 디렉토리의 이름입니다.

**2** 마운트하려는 매체의 이름을 찾습니다.

```
$ showmount -e system-name
```

**3** 관리자로 매체를 마운트합니다.

```
# mount -F nfs -o ro system-name:/media/media-name local-mount-point
```

*system-name:*            마운트할 매체의 시스템 이름입니다.

*media-name*            마운트하려는 매체의 이름입니다.

*local-mount-point*    원격 매체를 마운트할 로컬 디렉토리입니다.

**4** 관리자로 로그아웃합니다.

**5** 매체가 마운트되었는지 확인합니다.

```
$ ls /media
```

**예 3-4** 원격 시스템에서 DVD 또는 CD에 액세스

다음 예는 autofs를 사용하는 원격 시스템 starbug에서 sol\_10\_811\_sparc로 명명된 원격 DVD에 자동으로 액세스하는 방법을 보여 줍니다.

```
$ showmount -e starbug
export list for starbug:
/media/sol_10_811_sparc (everyone)
$ ls /net/starbug/media/
sol_10_811_sparc
```

## CD 및 DVD에 쓰기(작업)

---

이 장에서는 `cdwr` 명령을 사용하여 데이터 CD 및 DVD와 오디오 CD에 쓰고 복사하는 단계별 지침을 제공합니다.

- 46 페이지 “RBAC를 사용하여 이동식 매체에 대한 사용자 액세스를 제한하는 방법”
- 47 페이지 “CD 또는 DVD 라이터를 식별하는 방법”
- 47 페이지 “CD 또는 DVD 매체를 검사하는 방법”
- 48 페이지 “데이터 CD 또는 DVD에 사용할 ISO 9660 파일 시스템을 만드는 방법”
- 49 페이지 “멀티 세션 데이터 CD를 만드는 방법”
- 51 페이지 “오디오 CD를 만드는 방법”
- 52 페이지 “CD에서 오디오 트랙을 추출하는 방법”
- 53 페이지 “CD를 복사하는 방법”
- 54 페이지 “CD-RW 매체를 지우는 방법”

## 오디오 CD와 데이터 CD 및 DVD 사용

`cdwr` 명령을 사용하여 CD 및 DVD에서는 Rock Ridge를 사용한 ISO 9660 형식으로 또는 CD-R, CD-RW, DVD-RW나 DVD+RW 매체 장치에서는 Joliet 확장자로 파일 시스템을 쓸 수 있습니다.

`cdwr` 명령을 사용하면 다음 작업을 수행할 수 있습니다.

- 데이터 CD 및 DVD 만들기
- 오디오 CD 만들기
- 오디오 CD에서 오디오 데이터 추출
- CD 및 DVD 복사
- CD-RW 매체 지우기

`cdwr` 명령은 다음 릴리스에서 사용할 수 있습니다.

- Oracle Solaris 10 릴리스
- Oracle Solaris 11 릴리스, `media/cdwr` 패키지

권장 CD-R CD-RW 장치에 대한 정보를 확인하려면 다음 페이지로 이동하십시오.

<http://www.oracle.com/us/sun/index.html>

## CD/DVD 매체 관련 일반적으로 사용하는 용어

이 섹션에서는 CD/DVD 매체와 관련하여 일반적으로 사용하는 용어를 정의합니다.

용어	설명
CD-R	한 번만 쓸 수 있고 그 후에는 읽기만 가능한 CD 읽기 매체입니다.
CD-RW	쓰고 지울 수 있는 CD 쓰기 가능 매체입니다. CD-RW 매체는 CD-RW 장치에서는 읽을 수만 있습니다.
DVD-R	한 번만 쓸 수 있고 그 후에는 읽기만 가능한 디지털 비디오 디스크입니다(기록 가능). 이러한 장치의 용량은 CD-R 매체보다 훨씬 큼니다.
DVD+R	한 번만 쓸 수 있고 그 후에는 읽기만 가능한 디지털 비디오 디스크입니다(기록 가능). DVD+R 장치에는 DVD-R보다 더 완벽한 오류 관리 시스템이 있어 매체 품질에 관계없이 매체에 더욱 정확하게 구울 수 있습니다.
DVD-RW	DVD-R과 동일한 저장소 용량을 갖춘 디지털 비디오 디스크입니다(쓰기 가능). 전체 디스크를 처음으로 지우면 이 매체에 다시 기록할 수 있습니다.
DVD+RW	DVD+R과 동일한 저장소 용량을 갖춘 디지털 비디오 디스크입니다(임의 액세스 쓰기 가능). 이 매체에서는 전체 디스크를 지우지 않고 개별 블록을 덮어 쓸 수 있습니다.
DVD-RAM	나선형 트랙이 아닌 원형 트랙 및 하드 섹터링 기능을 갖춘 디지털 비디오 디스크입니다.
ISO 9660	ISO는 Industry Standards Organization의 머리글자어로, 컴퓨터 저장소 형식을 설정하는 기구입니다.  ISO 9660 파일 시스템은 주요 컴퓨터 플랫폼에서 동일한 CD 또는 DVD를 읽을 수 있도록 하는 표준 CD 또는 DVD 파일 시스템입니다. 1988년에 제정된 이 표준은 네바다 주의 High Sierra Hotel의 이름을 딴 High Sierra 산업 그룹이 작성했습니다. CD 또는 DVD 드라이브가 장착된 거의 모든 컴퓨터에서 ISO 9660 파일 시스템의 파일을 읽을 수 있습니다.

용어	설명
Joliet 확장자	Windows 파일 시스템 정보를 추가합니다.
Rock Ridge 확장자	UNIX 파일 시스템 정보를 추가합니다. Rock Ridge는 영화 Blazing Saddles(브레이징 새들스)에 등장하는 마을 이름을 딴 것입니다.  주 - 이러한 확장자는 배타적이지 않습니다. 두 시스템과의 호환성을 위해 mkisofs -R 및 -j 옵션을 지정할 수 있습니다. 자세한 내용은 mkisofs(1M)를 참조하십시오.
MMC 호환 레코더	Multi Media Command의 머리글자어로 이러한 레코더가 일반적인 명령 세트를 준수한다는 의미입니다. MMC 호환 레코더에 쓸 수 있는 프로그램은 다른 모든 레코더에도 쓸 수 있어야 합니다.
Red Book CDDA	Compact Disc Digital Audio의 머리글자어로, 콤팩트 디스크에 디지털 오디오를 저장하기 위한 산업 표준 방법입니다. “Red Book” 형식이라고도 합니다. 이러한 공식 산업 규격은 16비트 스테레오 사운드에서 44.1kHz(킬로헤르츠)의 샘플링 속도로 샘플링된 오디오 파일을 하나 이상 호출합니다.

CD 매체에 쓸 때 일반적으로 사용되는 용어는 다음과 같습니다.

용어	설명
비우기	CD-RW 매체에서 데이터를 지우는 프로세스입니다.
mkisofs	CD에서 ISO 파일 시스템을 만드는 명령입니다.
세션	리드인 및 리드아웃 정보가 포함된 전체 트랙입니다.
트랙	전체 데이터 또는 오디오 단위입니다.

## CD 및 DVD 데이터와 오디오 CD에 쓰기

CD 또는 DVD에 쓰기 프로세스는 중단할 수 없으며 지속적인 데이터 스트림을 필요로 합니다. cdrw -S 옵션으로 매체에 쓰기를 시뮬레이트하여 해당 시스템이 CD 또는 DVD에 쓰기에 충분한 속도로 데이터를 제공할 수 있는지 확인해 보십시오.

다음 문제 중 하나로 인해 쓰기 오류가 발생할 수 있습니다.

- 매체에서 드라이브 속도를 처리할 수 없습니다. 예를 들어 일부 매체의 경우 2x 또는 4x 속도에 대해서만 인증되었습니다.
- 시스템에 쓰기 프로세스를 필요로 하는, 메모리를 많이 사용하는 프로세스가 너무 많이 실행 중입니다.
- 네트워크 혼잡으로 인해 원격 시스템에 있는 이미지 읽기가 지연될 수 있습니다.

- 원본 드라이브가 대상 드라이브보다 느립니다.

이러한 문제가 발생하면 `cdwr -p` 옵션을 사용하여 장치의 쓰기 속도를 낮출 수 있습니다. 예를 들어 다음 명령은 4x 속도로 쓰기를 시뮬레이션하는 방법을 보여 줍니다.

```
$ cdwr -iS -p 4 image.iso
```

---

주 - CD-R, CD-RW(MRW 포맷이 아님), DVD-R, 및 DVD-RW 매체는 시뮬레이션 모드(-S)를 지원하지 않지만 DVD-RAM, DVD+R, DVD+RW 및 MRW 포맷 매체 및 기타 다른 매체는 시뮬레이션 모드를 지원하지 않습니다. 시뮬레이션 모드가 지원되지 않는 경우 다음 메시지가 표시됩니다.

```
Media does not support simulated writing
```

매체 유형에 대한 자세한 내용은 [44 페이지](#) “CD/DVD 매체 관련 일반적으로 사용하는 용어”를 참조하십시오.

---

자세한 내용은 [cdwr\(1\)](#)를 참조하십시오.

## RBAC를 사용하여 원격 매체에 대한 사용자 액세스 제한

기본적으로 모든 사용자가 이동식 매체에 액세스할 수 있습니다. 그러나 RBAC(역할 기반 액세스 제어)를 통해 역할을 설정하여 이동식 매체에 대한 사용자 액세스를 제한할 수 있습니다. 제한된 사용자 집합에 역할을 할당하여 이동식 매체에 대한 액세스를 제한합니다.

역할 사용에 대한 설명은 [Oracle Solaris 관리: 보안 서비스의 “역할 기반 액세스 제어\(개요\)”](#)를 참조하십시오.

### ▼ RBAC를 사용하여 이동식 매체에 대한 사용자 액세스를 제한하는 방법

- 1 관리자로 전환합니다.

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

- 2 장치 관리 권한이 포함된 역할을 설정합니다.

자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 9 장, “역할 기반 액세스 제어 사용\(작업\)”](#)을 참조하십시오.

```
# roleadd -m -d /export/home/muser -c "mediauser role" -A solaris.device.cdwr -P All muser
```

- 3 새로 만든 역할에 `cdwr` 명령을 사용해야 하는 사용자를 추가합니다.

```
# usermod -R muser joe
```

- 4 `/etc/security/policy.conf` 파일의 다음 행에 주석을 겁니다.

```
AUTHS_GRANTED=solaris.device.cdwr
```

이 단계를 수행하지 않으면 장치 관리 역할의 구성원만이 아니라 모든 사용자가 `cdwr` 명령에 대한 액세스 권한을 계속 가지고 있게 됩니다.

이 파일을 수정하면 `cdwr` 명령을 사용할 수 있는 사용자만 장치 관리 역할 구성원이 됩니다. 그 외 모든 사용자는 액세스가 거부되고 다음과 같은 메시지가 표시됩니다.

```
Authorization failed, Cannot access disks.
```

## ▼ CD 또는 DVD 라이터를 식별하는 방법

- 1 시스템에서 CD 또는 DVD 라이터를 식별합니다.

예를 들면 다음과 같습니다.

```
$ cdwr -l
Looking for CD devices...
  Node | Connected Device | Device type
-----+-----+-----
cdrom0 | YAMAHA CRW8824S | 1.0d | CD Reader/Writer
```

- 2 특정 CD 또는 DVD 라이터를 식별합니다.

예를 들면 다음과 같습니다.

```
$ cdwr -a filename.wav -d cdrom2
```

- 3 매체가 비어 있는지 또는 매체에 콘텐츠가 있는지 식별합니다.

예를 들면 다음과 같습니다.

```
$ cdwr -M

Device : YAMAHA CRW8824S
Firmware : Rev. 1.00 (26/04/00)
Media is blank
%
```

## ▼ CD 또는 DVD 매체를 검사하는 방법

`cdwr` 명령은 이동식 매체 서비스 실행 여부에 관계없이 작동합니다. 이동식 매체 서비스 사용으로 설정 또는 사용 안함으로 설정에 대한 자세한 내용은 37 페이지 “이동식 매체 서비스를 사용으로 설정 또는 사용 안함으로 설정하는 방법”을 참조하십시오.

- 1 CD 또는 DVD를 드라이브에 삽입합니다.

CD 또는 DVD는 드라이브가 읽을 수 있는 CD 또는 DVD가 될 수 있습니다.

- 2 드라이브를 나열하여 해당 드라이브가 제대로 연결되어 있는지 확인합니다.

```
$ cdw -l
Looking for CD devices...
      Node                Connected Device                Device type
-----+-----+-----+-----+-----+-----+-----
cdrom1                | YAMAHA    CRW8824S                1.0d | CD Reader/Writer
```

- 3 (옵션) 목록에 해당 드라이브가 표시되지 않으면 시스템에서 해당 드라이브를 인식할 수 있도록 다음 중 하나를 선택합니다.

- 시스템을 재부트하지 않고 드라이브 추가

```
# drvconfig
# disks
```

그런 다음 이동식 매체 서비스를 다시 시작합니다.

## 데이터 CD 또는 DVD 만들기

mkisofs 명령을 사용하여 파일 및 파일 정보를 CD 또는 DVD에서 사용되는 High Sierra 형식으로 변환하여 데이터를 먼저 준비합니다.

### ▼ 데이터 CD 또는 DVD에 사용할 ISO 9660 파일 시스템을 만드는 방법

- 1 빈 CD 또는 DVD를 드라이브에 삽입합니다.
- 2 새 CD 또는 DVD에서 ISO 9660 파일 시스템을 만듭니다.

```
$ mkisofs -r /pathname > cd-file-system
-r                Rock Ridge 정보를 만들고 파일 소유권을 0으로 재설정합니다.
/pathname        ISO 9660 파일 시스템을 만드는 데 사용한 경로 이름을 식별합니다.
> cd-file-system CD 또는 DVD에 저장한 파일 시스템의 이름을 식별합니다.
```

- 3 CD 또는 DVD로 파일 시스템을 복사합니다.

```
$ cdw -i cd-file-system
-i cd-file-system은 데이터 CD 또는 DVD를 만드는 데 필요한 이미지 파일을 지정합니다.
```

#### 예 4-1 데이터 CD 또는 DVD에 사용할 ISO 9660 파일 시스템 만들기

다음 예는 데이터 CD 또는 DVD에 사용할 ISO 9660 파일 시스템을 만드는 방법을 보여줍니다.



```
$ mkisofs -r /home/dubs/dir > dubs_cd
Total extents actually written = 56
Total translation table size: 0
Total rockridge attributes bytes: 329
Total directory bytes: 0
Path table size(bytes): 10
Max brk space used 8000
56 extents written (0 Mb)
```

그런 다음 CD로 파일 시스템을 복사합니다.

```
$ cdrw -i dubs_cd
Initializing device...done.
Writing track 1...done.
Finalizing (Can take several minutes)...done.
```

## ▼ 멀티 세션 데이터 CD를 만드는 방법

이 절차는 CD에 세션을 두 개 이상 설정하는 방법에 대해 설명합니다. 이 절차에는 infoA 및 infoB 디렉토리를 CD로 복사하는 예가 포함되어 있습니다.

### 1 첫번째 CD 세션에 사용할 파일 시스템을 만듭니다.

```
$ mkisofs -o infoA -r -V my_infoA /data/infoA
Total translation table size: 0
Total rockridge attributes bytes: 24507
Total directory bytes: 34816
Path table size(bytes): 98
Max brk space used 2e000
8929 extents written (17 Mb)
```

-o infoA        ISO 파일 시스템의 이름을 식별합니다.

-r              Rock Ridge 정보를 만들고 파일 소유권을 0으로 재설정합니다.

-V my\_infoA    이동식 매체 서비스에서 마운트 지점으로 사용할 볼륨 레이블을 식별합니다.

/data/infoA    만들 ISO 이미지 디렉토리를 식별합니다.

### 2 CD로 첫번째 세션의 ISO 파일 시스템을 복사합니다.

```
$ cdrw -i0 infoA
Initializing device...done.
Writing track 1...done.
done.
Finalizing (Can take several minutes)...done.
```

-i infoA        CD에 쓸 이미지 파일의 이름을 식별합니다.

-0              쓸 수 있도록 CD를 열어 둡니다.

### 3 CD를 꺼낸 후 다시 삽입합니다.

4 다음 쓰기 세션에 포함할 CD 매체의 경로 이름을 식별합니다.

```
$ eject -n
.
.
.
cdrom0 -> /vol/dev/rdisk/c2t4d0/my_infoA
/vol/dev/... 경로 이름을 적어 둡니다.
```

5 CD에서 다음 쓰기 가능 주소를 식별하고 다음 세션을 씁니다.

```
% cdrw -M /cdrom
Device : YAMAHA CRW8424S
Firmware : Rev. 1.0d (06/10/99)

Track No. |Type      |Start address
-----+-----+-----
1         |Audio     |0
2         |Audio     |33057
3         |Data      |60887
4         |Data      |68087
5         |Data      |75287
Leadout   |Data      |84218

Last session start address: 75287
Next writable address: 91118
```

다음 세션을 쓸 때 제공할 수 있도록 Next writable address 출력에 포함된 주소를 적어 둡니다.

6 다음 CD 세션에 사용할 다음 ISO 파일 시스템을 만들어 CD에 씁니다.

```
$ mkisofs -o infoB -r -C 0,91118 -M /vol/dev/rdisk/c2t4d0/my_infoA
/data/infoB
Total translation table size: 0
Total rockridge attributes bytes: 16602
Total directory bytes: 22528
Path table size(bytes): 86
Max brk space used 20000
97196 extents written (189 Mb)

-o infoB
-r
-C 0,91118
-M /vol/dev/rdisk/c2t4d0/my_infoA
/data/infoB
```

ISO 파일 시스템의 이름을 식별합니다.

Rock Ridge 정보를 만들고 파일 소유권을 0으로 재설정합니다.

첫번째 세션의 시작 주소와 다음 쓰기 가능 주소를 식별합니다.

병합할 기존 ISO 이미지 경로를 지정합니다.

만들 ISO 이미지 디렉토리를 식별합니다.

## 오디오 CD 만들기

`cdwr` 명령을 사용하여 개별 오디오 트랙 또는 `.au` 및 `.wav` 파일에서 오디오 CD를 만들 수 있습니다.

지원되는 오디오 형식은 다음 표에 설명되어 있습니다.

형식	설명
<code>sun</code>	Red Book CDDA 형식의 데이터가 포함된 Oracle <code>.au</code> 파일입니다.
<code>wav</code>	Red Book CDDA 형식의 데이터가 포함된 RIFF( <code>wav</code> ) 파일입니다.
<code>cda</code>	원시 CD 오디오 데이터가 포함된 <code>.cda</code> 파일로, 리틀 엔디언 바이트 순서로 정렬된 44.1kHz의 샘플링 속도의 16비트 PCM 스테레오입니다.
<code>aur</code>	빅 엔디언 바이트 순서로 정렬된 원시 CD 데이터가 포함된 <code>.aur</code> 파일입니다.

지정된 오디오 형식이 없으면 `cdwr` 명령은 파일 확장자를 기준으로 오디오 파일 형식을 결정하려고 합니다. 확장자의 대소문자는 무시됩니다.

### ▼ 오디오 CD를 만드는 방법

이 절차에서는 CD로 오디오 파일을 복사하는 방법에 대해 설명합니다.

- 1 CD-RW 드라이브로 빈 CD를 삽입합니다.
- 2 오디오 파일이 포함된 디렉토리로 복사합니다.

```
$ cd /myaudiodir
```

- 3 CD로 오디오 파일을 복사합니다

```
$ cdwr -a track1.wav track2.wav track3.wav
```

`-a` 옵션은 오디오 CD를 만듭니다.

#### 예 4-2 오디오 CD 만들기

다음 예는 오디오 CD를 만드는 방법을 보여 줍니다.

```
$ cdwr -a bark.wav chirp.au meow.wav
Initializing device...done.
Writing track 1...done.
done.
Writing track 2...done.
```

```
Writing track 3...done.
done.
Finalizing (Can take several minutes)...done.
```

다음 예는 다중 세션 오디오 CD를 만드는 방법을 보여 줍니다. 첫번째 세션을 쓰면 CD가 배출됩니다. 다음 쓰기 세션 이전에 CD를 다시 삽입해야 합니다.

```
$ cdwr -a0 groucho.wav chico.au harpo.wav
Initializing device...done.
Writing track 1...done.
done.
Writing track 2...done.
Writing track 3...done.
done.
Finalizing (Can take several minutes)...done.
<Re-insert CD>
$ cdwr -a zeppo.au
Initializing device...done.
Writing track 1...done.
done.
Finalizing (Can take several minutes)...done.
```

## ▼ CD에서 오디오 트랙을 추출하는 방법

다음 절차를 수행하여 CD에서 오디오 트랙을 추출하고 새 CD로 오디오 트랙을 복사합니다.

`cdwr -T` 옵션을 사용하여 오디오 파일 형식을 지정하지 않은 경우 `cdwr` 명령은 파일 이름 확장자를 사용하여 오디오 파일 형식을 결정합니다. 예를 들어 `cdwr` 명령은 이 파일이 `.wav` 파일인지 감지합니다.

```
$ cdwr -x 1 testme.wav
```

1 CD-RW 드라이브로 오디오 CD를 삽입합니다.

2 오디오 트랙을 추출합니다.

```
$ cdwr -x -T audio-type 1 audio-file
```

`-x` 오디오 CD에서 오디오 데이터를 추출합니다.

`T audio-type` 추출할 오디오 파일 형식을 식별합니다. 지원되는 오디오 유형은 `sun`, `wav`, `cda` 또는 `aur`입니다.

`audio-file` 추출할 오디오 트랙을 식별합니다.

3 새 CD로 트랙을 복사합니다.

```
$ cdwr -a audio-file
```

### 예 4-3 오디오 CD 추출 및 만들기

다음 예는 오디오 CD에서 첫번째 트랙을 추출한 다음 파일의 이름을 `song1.wav`로 지정하는 방법을 보여 줍니다.

```
$ cdwr -x -T wav 1 song1.wav
Extracting audio from track 1...done.
```

이 예에서는 오디오 CD로 트랙을 복사하는 방법을 보여 줍니다.

```
$ cdwr -a song1.wav
Initializing device...done.
Writing track 1...done.
Finalizing (Can take several minutes)...done.
```

## ▼ CD를 복사하는 방법

이 절차에서는 오디오 CD에서 디렉토리로 모든 트랙을 추출한 다음 CD로 복사하는 방법에 대해 설명합니다.

---

주 - 기본적으로 `cdwr` 명령은 CD를 `/tmp` 디렉토리로 복사합니다. 복사하려면 최대 700MB의 사용 가능한 공간이 필요합니다. `/tmp` 디렉토리에 CD를 복사하는 데 필요한 공간이 부족한 경우 `-m` 옵션을 사용하여 대체 디렉토리를 지정합니다.

---

- 1 CD-RW 드라이브로 오디오 CD를 삽입합니다.
- 2 오디오 파일을 저장할 디렉토리를 만듭니다.  

```
$ mkdir /music_dir
```
- 3 오디오 CD에서 트랙을 추출합니다.  

```
$ cdwr -c -m music_dir
```

각 트랙에 대해 `Extracting audio ...` 메시지가 표시됩니다.  
 트랙이 모두 추출되면 CD가 배출됩니다.
- 4 빈 CD를 삽입하고 `Return`을 누릅니다.  
 트랙이 추출되면 오디오 CD가 배출됩니다. 빈 CD를 삽입하라는 메시지가 표시됩니다.

### 예 4-4 CD 복사

이 예는 CD를 다른 CD로 복사하는 방법에 대해 설명합니다. 이 작업을 수행하려면 CD-RW 장치가 두 개 필요합니다.

```
$ cdrw -c -s cdrom0 -d cdrom1
```

## ▼ CD-RW 매체를 지우는 방법

CD에 다시 쓸려면 기존 CD-RW 데이터를 지워야 합니다.

- 다음 중 하나를 선택하여 전체 매체 또는 CD에서 마지막 세션만 지웁니다.

- 마지막 세션만 지웁니다.

```
$ cdrw -d cdrom0 -b session
```

-b session 옵션을 사용하여 마지막 세션만 지우는 것이 -b all 옵션을 사용하여 전체 매체를 지우는 것보다 빠릅니다. cdrw 명령을 사용하여 하나의 세션에만 데이터 또는 오디오 CD를 만들었다더라도 -b session 옵션을 사용할 수 있습니다.

- 전체 매체를 지웁니다.

```
$ cdrw -d cdrom0 -b all
```

## 장치 관리(개요/작업)

---

이 장에서는 Oracle Solaris 릴리스에서 디스크, DVD 드라이브 및 테이프 장치 등의 주변 장치를 관리하는 방법에 대한 개요 정보와 단계별 지침을 제공합니다.

이 장에서 다루는 개요 정보는 다음과 같습니다.

- 56 페이지 “장치 관리에 관련된 새로운 기능”
- 59 페이지 “추가 장치 관리 작업 검색 위치”
- 60 페이지 “장치 드라이버 정보”
- 61 페이지 “장치의 자동 구성”
- 63 페이지 “장치 구성 정보 표시”
- 67 페이지 “고장난 장치 해결”
- 71 페이지 “장치 액세스”

이 장에서 다루는 단계별 지침은 다음과 같습니다.

- 61 페이지 “드라이버 구성을 사용자 정의하는 방법”
- 64 페이지 “시스템 구성 정보를 표시하는 방법”
- 68 페이지 “고장난 장치를 해결하는 방법”
- 70 페이지 “장치 드라이버를 추가하는 방법”
- 69 페이지 “주변 장치를 추가하는 방법”

Oracle Solaris 릴리스에서 장치 관리의 경우, 일반적으로 시스템의 주변 장치 추가 및 제거, 장치 지원을 위한 타사 장치 드라이버 추가 및 시스템 구성 정보 표시 등이 포함됩니다.

## 장치 관리에 관련된 새로운 기능

이 섹션에서는 Oracle Solaris 릴리스의 새로운 장치 관리 기능에 대한 정보를 제공합니다.

- 56 페이지 “드라이버 구성 사용자 정의”
- 56 페이지 “Solaris PCI 리소스 관리자”
- 57 페이지 “새로운 InfiniBand 관리 기능”
- 57 페이지 “새로운 InfiniBand 진단 도구 및 명령”
- 57 페이지 “새로운 Ethernet Over InfiniBand 장치”
- 57 페이지 “새로운 핫 플러그 기능”
- 58 페이지 “장치 이름 지정 기능 향상”
- 58 페이지 “PCIe(PCI Express) 지원”

### 드라이버 구성 사용자 정의

**Oracle Solaris 11:** 이 Solaris 릴리스에서는 `/etc/driver/drv` 디렉토리에 보조적인 드라이버 구성 파일인 `driver.conf`를 제공할 수 있습니다. 이전 Solaris 릴리스에서는 공급업체의 `driver.conf` 파일을 직접 수정해야 했습니다.

이 릴리스에서는 시스템에서 드라이버 공급업체의 `/kernel` 또는 `/platform` `driver.conf` 파일을 사용자 정의 `/etc/driver/drv` `driver.conf` 파일과 자동으로 병합하므로, 드라이버에서 두 개의 등록 정보 값 세트를 모두 사용할 수 있습니다.

`prtconf -u` 명령을 사용하면 원래 공급업체와 사용자 정의 등록 정보 값을 표시할 수 있습니다. 공급업체의 `driver.conf` 파일과 로컬 사용자 정의 `driver.conf` 파일을 분리하면, 관리용 사용자 정의 설정을 덮어쓰지 않으면서 새로운 공급업체 `driver.conf` 파일로 업그레이드할 수 있습니다.

`/kernel/drv` 디렉토리에 있는 이전 Solaris 릴리스 또는 시스템의 드라이버 구성 파일에서 수정한 사항이 있는 경우에는 `/etc/driver/drv` 디렉토리로 복사해야 합니다.

`/etc/driver/drv/` `driver.conf` 파일 만들기에 대한 자세한 내용은 61 페이지 “드라이버 구성을 사용자 정의하는 방법”, `driver.conf(4)` 및 `driver(4)`를 참조하십시오.

### Solaris PCI 리소스 관리자

**Oracle Solaris 11:** 이 릴리스에서는 시스템을 부트할 때 PCIe(PCI Express) 장치를 지원하기에 충분한 리소스가 할당되도록 PCI 리소스 균형을 조정하는 PCIRM(PCI 리소스 관리자)이 제공됩니다. 예를 들어 SR-IOV(Single Root I/O Virtualization) 장치는 가상화 기능 때문에 PCI 메모리 리소스가 더 많이 필요합니다. 현재 사용되는 대부분의 펌웨어는 SR-IOV 표준이 나오기 전에 디자인되었기 때문에 가상 장치용 리소스를 예약해야 하는 요구 사항을 인식하지 못합니다. 리소스 균형 재조정 프로세스는 부트 시에 자동으로 진행되며 따로 관리할 필요가 없습니다.



## 새로운 InfiniBand 관리 기능

**Oracle Solaris 11:** IPoIB 관리가 이전 릴리스와 비교해 달라졌습니다. 이 릴리스에서는 `dladm` 하위 명령인 `create-part`, `delete-part`, `view-part` 및 `show-ib`를 사용하여 IPoIB 데이터 링크 정보 만들기, 삭제 및 보기를 수행할 수 있습니다. 이러한 새 `dladm` 하위 명령은 IPoIB 데이터 링크를 더 쉽게 관리할 수 있는 방법을 제공하며 일부 IB 문제를 보고 해결하는 데에도 도움이 됩니다. 예를 들어 `show-ib` 하위 명령은 시스템에 있는 HCA의 수와 해당되는 HCA GUID를 보여 줍니다. 새 관리 모델에는 두 가지 유형의 IP over IB 데이터 링크가 있습니다.

- 물리적 데이터 링크 - 일반적인 NIC(네트워크 인터페이스 카드) 데이터 링크와 비슷합니다.
- IB 분할 데이터 링크 - VNIC(가상 NIC)와 비슷합니다.

새로 설치된 시스템에는 기본적으로 물리적 링크가 자동 설치됩니다. 그 후에 `dladm` 명령을 사용하여 IB 물리적 링크에서 IB 분할 링크를 만들 수 있습니다. NIC 데이터 링크와 달리 IB 물리적 링크는 플럼 또는 링크 사용을 위한 IP 주소 할당 등의 방법으로 구성할 수 없습니다.

`dladm` 하위 명령을 이용한 IB 구성 요소 관리와 관련된 자세한 내용은 [172 페이지 “IPoIB 장치 관리\(dladm\)”](#)를 참조하십시오.

## 새로운 InfiniBand 진단 도구 및 명령

**Oracle Solaris 11:** `system/io/infiniband/open-fabrics` 패키지는 OFA(Open Fabrics Alliance) OFED(Open Fabrics Enterprise Distribution) 버전 1.5.3에 있는 명령 및 유틸리티 중 일부를 제공합니다. 이러한 도구 모음은 IB 장치를 나열 및 질의하고, IB 패브릭 문제를 진단 및 해결하며, 다양한 IB 사용자 동사 마이크로 벤치마크 모음을 사용하여 IB 성능을 측정하는 기능을 제공합니다. 자세한 내용은 [175 페이지 “IB 장치 모니터 및 문제 해결”](#)을 참조하십시오.

## 새로운 Ethernet Over InfiniBand 장치

**Oracle Solaris 11:** EoIB(Ethernet over InfiniBand) 드라이버는 Oracle Network QDR InfiniBand 게이트웨이 스위치에 연결된 시스템에서 DLPI(Data Link Provider Interface)와 전체적인 IB 포트를 지원하는 다중 스레드 방식의 로드 가능, 복제 가능한 GLD 기반 STREAMS 드라이버입니다.

## 새로운 핫 플러그 기능

**Oracle Solaris 11:** 이 릴리스에서는 `hotplug` 명령을 사용하여 PCIe(PCI Express) 및 PCI SHPC(표준 핫 플러그 컨트롤러) 장치에서 핫 플러그 가능한 연결을 관리할 수 있습니다. 이 기능은 USB 및 SCSI와 같은 다른 버스 유형에서는 지원되지 않습니다.

이전 Solaris 릴리스와 마찬가지로 `cfgadm`을 계속 사용하여 핫 플러그 가능한 USB 및 SCSI 장치를 관리할 수 있습니다. 이 릴리스에서 핫 플러그 기능을 사용할 경우의 이점은 작업을 자동으로 설정하고 사용 안함으로 설정할 수 있다는 것과, `hotplug` 명령을 사용하여 지원되는 PCI 장치에 대한 온라인 및 오프라인 기능을 이용할 수 있다는 것입니다.

자세한 내용은 81 페이지 “`hotplug` 명령을 사용하여 PCIe 핫 플러그 기능 사용”을 참조하십시오.

## 장치 이름 지정 기능 향상

**Oracle Solaris 11:** `/dev` 네임스페이스는 필요에 따라 여러 파일 시스템 인스턴스를 지원합니다. 시스템을 부트할 때 `/dev` 파일 시스템의 전역 인스턴스가 자동으로 만들어집니다. 그 후의 `/dev` 인스턴스는 비전역 영역에 장치를 추가하는 등의 필요한 경우에 만들어지고 마운트됩니다. 비전역 영역이 종료되면 사용 가능한 `/dev` 인스턴스가 마운트 해제되어 사용할 수 없게 됩니다.

또한, 장치 구성이 다음과 같이 향상되었습니다.

- **재구성 부트 제거** - 이전의 Solaris 릴리스에서는 전원이 꺼진 시스템에 장치를 연결한 경우에 재구성 부트가 필요했습니다.

이 릴리스에서는 장치를 전원이 꺼진 시스템에 연결할 경우 재구성 부트가 필요 없습니다. 시스템을 재부트하면 새로 연결한 장치가 자동으로 인식되고 적절한 장치 링크가 만들어집니다.

자세한 내용은 `dev(7FS)`를 참조하십시오.

- **영역 장치 지원 간소화** - 위에서 설명한 것처럼 비전역 영역에 대한 `/dev` 디렉토리의 특정 인스턴스를 제공함으로써 Oracle Solaris 영역에 대한 장치 지원이 향상되었습니다. 또한, 영역이 이제 영역에서 `devfsadm` 데몬과 독립적으로 장치를 재구성할 수 있습니다.

- **의사 장치 만들기 향상** - 이 릴리스에서는 요청 시에 `/dev/pts` 디렉토리의 내용이 전역 `/dev` 네임스페이스에 만들어지며 비전역 영역에서 필요에 따라 `/dev` 인스턴스에도 만들어집니다. 또한, `pty` 링크가 할당된 전역 영역 또는 비전역 영역에만 표시됩니다.

자세한 내용은 `grantpt(3C)`를 참조하십시오.

장치 구성에 대한 자세한 내용은 60 페이지 “Oracle Solaris OS에서 장치 관리”를 참조하십시오.

## PCIe(PCI Express) 지원

**Oracle Solaris 11:** 이 Oracle Solaris 릴리스는 PCIe(PCI Express) 상호 연결을 지원합니다. 이 상호 연결은 SPARC와 x86 시스템 모두에서 데스크탑, 엔터프라이즈, 모바일, 통신 및 내장 응용 프로그램에 주변 장치를 연결하도록 디자인되었습니다.

PCIe 상호 연결은 업계 표준의 고성능 직렬 I/O 버스입니다. PCIe 기술에 대한 자세한 내용은 다음 사이트를 참조하십시오.

<http://www.pcisig.com/home>

PCIe 소프트웨어는 이 Oracle Solaris 릴리스에서 다음과 같은 기능을 제공합니다.

- 확장된 PCIe 구성 공간 지원
- PCIe 기본 오류 처리 및 MSI 인터럽트 지원
- PCIe 장치에 대한 수정된 IEEE-1275 등록 정보
- `cfgadm` 명령의 `cfgadm_pci` 구성 요소를 향상하여 PCIe 핫 플러그 지원(고유 및 ACPI 기반 모두)
- ATTN 버튼 사용 기반 PCIe 주변 장치 자동 구성

PCIe 주변 기기를 핫 플러그할 수 있는 관리 모델은 `cfgadm` 명령을 사용하는 PCI 주변 기기와 동일합니다.

사용자 시스템에서 PCIe 및 PCIe 핫 플러그 기능이 지원되는지 여부는 해당 하드웨어 플랫폼 설명서를 참조하십시오. 또한 해당되는 경우, 시스템에서 어댑터를 물리적으로 삽입 또는 제거하는 방법에 대한 지침과 장치 자동 구성에 대한 의미를 신중하게 검토합니다.

PCIe 주변 장치에서의 `cfgadm` 명령 사용에 대한 자세한 내용은 93 페이지 “`cfgadm` 명령을 사용하여 PCI 또는 PCIe 핫 플러그 기능 사용(작업 맵)”을 참조하십시오.

## 추가 장치 관리 작업 검색 위치

다음 표는 프린터 및 모뎀 등의 직렬 장치를 추가하고 장치의 핫 플러그 기능을 사용하는 단계별 지침을 찾을 수 있는 위치에 대해 설명합니다.

표 5-1 장치 추가 지침 검색 위치

장치 관리 작업	자세한 정보
핫 플러그 기능을 사용할 수 없는 디스크 추가	12 장, “SPARC: 디스크 설정(작업)” 또는 13 장, “x86: 디스크 설정(작업)”
SCSI 또는 PCI 장치의 핫 플러그 기능 사용	84 페이지 “ <code>cfgadm</code> 명령을 사용하여 SCSI 핫 플러그 기능 사용” 또는 94 페이지 “ <code>cfgadm</code> 명령을 사용하여 PCI 또는 PCIe 핫 플러그 기능 사용”
USB 장치의 핫 플러그 기능 사용	128 페이지 “USB 대용량 저장 장치 사용(작업 맵)”
모뎀 추가	<b>Oracle Solaris 관리: 일반 작업</b> 의 16 장, “시스템 콘솔, 터미널 장치 및 전원 서비스 관리(작업)”

장치 관리 작업	자세한 정보
프린터 추가	<b>Oracle Solaris 관리: 일반 작업</b> 의 15 장, “CUPS를 사용하여 프린터 설정 및 관리(작업)”
장치 보안	<b>Oracle Solaris 관리: 보안 서비스</b> 의 5 장, “장치에 대한 액세스 제어(작업)”

## Oracle Solaris OS에서 장치 관리

다음 섹션에서는 Oracle Solaris OS에서 장치를 관리하는 기능에 대한 개요 정보를 제공합니다. 장치 액세스에 대한 자세한 내용은 71 페이지 “장치 액세스”를 참조하십시오.

### 장치 지원 식별

장치 감지 도구를 사용하여 이 Oracle Solaris 릴리스에서 x86 하드웨어가 지원되는지 여부를 식별할 수 있습니다. 자세한 내용은 다음 사이트를 참조하십시오.

[http://www.oracle.com/webfolder/technetwork/hcl/hcts/device\\_detect.jsp](http://www.oracle.com/webfolder/technetwork/hcl/hcts/device_detect.jsp)

### 장치 드라이버 정보

컴퓨터는 일반적으로 다양한 주변 장치와 대용량 저장 장치를 사용합니다. 예를 들어 대부분의 시스템은 디스크 드라이브, 키보드와 마우스 및 자기 백업 매체를 사용합니다. 일반적으로 사용되는 기타 장치에는 다음이 포함됩니다.

- DVD 드라이브
- 프린터 및 플로터
- 라이트 펜
- 터치 스크린
- 디지털타이저
- 태블릿과 스타일러스

Oracle Solaris 소프트웨어는 이러한 모든 장치와 직접 통신하지 않습니다. 각각의 장치 유형에는 서로 다른 데이터 형식, 프로토콜 및 전송 속도가 필요합니다.

**장치 드라이버**는 운영 체제가 특정 하드웨어와 통신할 수 있게 해 주는 저급 프로그램입니다. 드라이버는 해당 하드웨어에 대해 운영 체제의 “인터프리터” 역할을 담당합니다.

## ▼ 드라이버 구성을 사용자 정의하는 방법

Oracle Solaris 11 릴리스에서는 드라이버 사용자 정의를 `/etc/driver/drv` 디렉토리에서 수행하며, 이전 릴리스와 같이 `/kernel` 디렉토리에서 수행하지 않습니다. 이 향상으로 인해 시스템을 업그레이드할 때 드라이버 사용자 정의를 겹쳐 쓰지 않습니다. 업그레이드를 진행하는 동안 `/etc/driver/drv` 디렉토리에 있는 파일은 보존됩니다.

드라이버 구성을 사용자 정의한다는 것은 보통 장치별 매개변수 또는 모든 장치에 영향을 미치는 전역 등록 정보가 추가되었거나 수정되었음을 의미합니다.

- 1 관리자로 전환합니다.
- 2 공급업체에서 제공한 원래의 `driver.conf` 파일을 `/etc/driver/drv` 디렉토리에 복사합니다. 예를 들면 다음과 같습니다.

```
# cp /kernel/drv/sd.conf /etc/driver/drv/sd.conf
```

- 3 매개변수 항목을 수정하고 파일을 저장합니다. 예를 들어 `sd.conf`에는 대상 `0, lun 0`의 `sd` 장치에 대한 다음 항목이 포함되어 있습니다. 이 장치에 `retries` 매개변수를 추가하려면 기존 항목을 다음과 같이 수정합니다.

```
name="sd" class="scsi" target=0 lun=0 retries=4;
```

- 4 사용자 정의 등록 정보 값을 표시합니다. 예를 들면 다음과 같습니다.

```
# prtconf -u
sd, instance #1
    Admin properties:
        name='retries' type=int items=1
        value=00000004
```

## 장치의 자동 구성

커널은 플랫폼별 구성 요소가 있는 작은 일반 코어와 모듈 세트로 구성됩니다. Oracle Solaris 릴리스에서는 커널이 자동으로 구성됩니다.

**커널 모듈**은 시스템에서 특정 작업을 수행하는 데 사용되는 하드웨어 또는 소프트웨어 구성 요소입니다. **로드 가능한 커널 모듈**의 예로는 장치에 액세스할 때 로드되는 장치 드라이버가 있습니다.

플랫폼 독립 커널은 `/kernel/genunix`입니다. 플랫폼 특정 구성 요소는 `/platform/'uname -m'/kernel/unix`입니다.

커널 모듈에 대해서는 아래 표에 설명되어 있습니다.

표 5-2 Solaris 커널 모듈 설명

위치	디렉토리 내용
/platform/‘uname -m’/kernel	플랫폼 특정 커널 구성 요소
/kernel	시스템 부트 시 필요하며 모든 플랫폼에서 공통되는 커널 구성 요소
/usr/kernel	특정 명령 세트 안에서 모든 플랫폼에 공통적인 커널 구성 요소

부트 시에 연결되는 장치는 시스템에서 결정합니다. 그러면 커널이 동적으로 자체 구성되며 필요한 모듈을 메모리에 로드합니다. 이때 디스크 장치 및 테이프 장치와 같은 장치에 액세스하면 장치 드라이버가 로드됩니다. 모든 커널 모듈이 필요할 때 자동으로 로드되기 때문에 이 프로세스를 **자동 구성**이라고 합니다.

/etc/system 파일을 수정하면 커널 모듈의 로드 방법을 사용자 정의할 수 있습니다. 이 파일의 수정에 대한 지침은 [system\(4\)](#)을 참조하십시오.

## 자동 구성의 기능 및 이점

자동 구성의 이점은 다음과 같습니다.

- 모듈이 필요할 때 로드되므로 주 메모리가 더 효율적으로 사용됩니다.
- 새 장치가 시스템에 추가될 때 커널을 재구성할 필요가 없습니다.
- 커널을 재구성하고 시스템을 재부트하지 않아도 드라이버를 로드 및 테스트할 수 있습니다.

자동 구성은 시스템에 새 장치(및 드라이버)를 추가할 때 사용됩니다. 이전 릴리스에서는 종료된 시스템에 장치를 추가할 경우 재구성 부트를 수행해야 했습니다. 이제는 장치 구성의 향상된 기능으로 종료된 시스템에 장치를 추가해도 재구성 부트를 수행할 필요가 없습니다.

시스템 구성 요소가 핫 플러그 기능을 지원하는 경우에는 시스템이 아직 실행 중인 동안 Oracle Solaris OS에서 장치를 추가, 제거 또는 교체할 수 있습니다. 장치의 핫 플러그 기능에 대한 자세한 내용은 [6 장, “동적으로 장치 구성\(작업\)”](#)을 참조하십시오.

## 지원되지 않는 장치에 필요한 사항

Oracle Solaris 릴리스에는 다양한 표준 장치를 지원하는데 필요한 장치 드라이버가 포함되어 있습니다. 이러한 드라이버는 /kernel/drv 및 /platform/‘uname -m’/kernel/drv 디렉토리에서 찾을 수 있습니다.

그러나 지원되지 않는 장치를 구입한 경우에는 장치를 올바르게 설치, 유지 및 관리하는데 필요한 소프트웨어를 제조업체에서 제공해야 합니다.

최소한 이 소프트웨어에는 장치 드라이버와 관련 구성(.conf) 파일이 포함됩니다. .conf 파일은 drv 디렉토리에 상주합니다. 장치가 Oracle Solaris 유틸리티와 호환되지 않을 가능성이 있기 때문에 이 소프트웨어에 사용자 정의 유지 관리 및 관리 유틸리티가 포함될 수도 있습니다.

지원되지 않는 장치에 필요한 사항에 대한 자세한 내용은 장치 제조업체에 문의하십시오.

## 장치 구성 정보 표시

시스템 및 장치 구성 정보를 표시하는 경우에는 세 개의 명령이 사용됩니다.

명령	설명	매뉴얼 페이지
prtconf	시스템의 장치 계층에 설명된 장치 구성과 총 메모리 크기를 포함한 시스템 구성 정보를 표시합니다. 이 명령으로 표시되는 출력은 시스템의 유형에 따라 다릅니다.	<a href="#">prtconf(1M)</a>
sysdef	시스템 하드웨어, 의사 장치, 로드 가능한 모듈 및 선택한 커널 매개변수 등의 장치 구성 정보를 표시합니다.	<a href="#">sysdef(1M)</a>
dmesg	시스템 진단 메시지와 마지막 재부트 후 시스템에 연결된 장치 목록을 표시합니다.	<a href="#">dmesg(1M)</a>

시스템에서 장치 식별에 사용되는 장치 이름에 대한 자세한 내용은 72 페이지 “장치 이름 지정 규약”을 참조하십시오.

## driver not attached 메시지

prtconf 및 sysdef 명령에서 다음과 같은 드라이버 관련 메시지가 표시될 수 있습니다.

*device, instance #number (driver not attached)*

이 메시지가 항상 이 장치에 드라이버를 사용할 수 없다는 것을 의미하는 것은 아닙니다. 이 메시지는 이 노드에 장치가 없거나 장치가 사용되지 않기 때문에 **현재** 장치 인스턴스에 연결된 드라이버가 없다는 것을 의미합니다. 장치에 액세스하면 드라이버가 자동으로 로드됩니다. 장치가 사용되지 않는 동안은 언로드됩니다.

## 사용 중인 장치 오류 검사

다음 유틸리티는 지정된 장치를 사용 중인 경우를 감지합니다.

- dumpadm
- format
- mkfs 및 newfs
- swap

이러한 향상된 기능에 따라 위의 유틸리티에서 다음과 같은 몇 가지 사용 시나리오를 감지할 수 있습니다.

- 장치가 ZFS 저장소 풀의 일부인 경우
- 장치가 덤프 또는 스왑 장치인 경우
- 마운트된 파일 시스템 또는 장치에 대한 항목이 `/etc/vfstab` 파일에 있을 경우

예를 들어, `format` 유틸리티를 사용하여 활성 장치에 액세스하려고 하면 다음과 비슷한 메시지가 표시됩니다.

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
  0. clt0d0 <FUJITSU-MAY2073RCSUN72G-0401 cyl 8921 alt 2 hd 255 sec 63>
    /pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@0,0
    /dev/chassis/SYS/HD0/disk
  1. clt1d0 <FUJITSU-MAY2073RCSUN72G-0401-68.37GB>
    /pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@1,0
    /dev/chassis/SYS/HD1/disk
Specify disk (enter its number): 0
selecting clt0d0
[disk formatted]
/dev/dsk/clt0d0s0 is part of active ZFS pool rpool. Please see zpool(1M).
```

FORMAT MENU:

```
.
```

## ▼ 시스템 구성 정보를 표시하는 방법

`prtconf` 및 `sysdef` 명령의 출력을 사용하여 시스템에 연결된 디스크, 테이프 및 DVD 장치를 식별합니다. 이러한 명령의 출력은 장치 인스턴스 옆에 `driver not attached` 메시지를 표시합니다. 이러한 장치는 항상 시스템 프로세스에 의해 모니터링되므로 `driver not attached` 메시지는 보통 해당 장치 인스턴스에 장비가 없다는 것을 나타냅니다.

`sysdef` 명령을 사용하여 의사 장치, 로드 가능한 모듈 및 선택된 커널 매개변수를 포함하는 시스템 구성 정보를 표시합니다.

### ● 시스템 및 장치 구성 정보를 표시합니다.

- 시스템에 연결된 모든 장치를 표시합니다.

예를 들어, SPARC 시스템의 다음 `prtconf -v` 출력은 시스템에 연결된 디스크 장치를 식별합니다. 자세한 디스크 정보는 **장치 보조 노드** 섹션에서 설명합니다.

```
$ /usr/sbin/prtconf -v | more
Memory size: 32640 Megabytes
System Peripherals (Software Nodes):
```

```
SUNW,SPARC-Enterprise-T5220
```

```
.
```



```

.
.
location: /dev/chassis/SUN-Storage-J4400.09180AKA24/SCSI_Device__2/disk
Device Minor Nodes:
  dev=(27,40)
    dev_path=/pci@0/pci@0/pci@8/pci@0/pci@a/LSILogic,sas@0/sd@2,0:a
    spectype=blk type=minor
    dev_link=/dev/dsk/c4t2d0s0
  dev_path=/pci@0/pci@0/pci@8/pci@0/pci@a/LSILogic,sas@0/sd@2,0:a,raw
    spectype=chr type=minor
    dev_link=/dev/rdisk/c4t2d0s0
Device Minor Layered Under:
  mod=zfs accesstype=blk
    dev_path=/pseudo/zfs@0
Minor properties:
  name='Nblocks' type=int64 items=1 dev=(27,40)
    value=0000000074702c8f
  name='Size' type=int64 items=1 dev=(27,40)
    value=0000000e8e0591e00
.
.
.

```

- 시스템에 연결된 특정 장치 하나의 정보를 표시합니다.

예를 들어, SPARC 시스템의 다음 `prtconf` 출력은 `/dev/dsk/c4t2d0s0`의 `sd` 인스턴스 번호를 표시합니다.

```
# prtconf -v /dev/dsk/c4t2d0s0 | grep instance
sd, instance #5
```

- 시스템에 연결된 장치만 표시합니다.

```
# prtconf | grep -v not
```

- 장치 사용 정보를 표시합니다.

예를 들어 다음 `fuser` 명령은 `/dev/console` 장치에 액세스하는 프로세스를 표시합니다.

```
# fuser -d /dev/console
/dev/console: 5742o 2269o 20322o 858o
```

## 예 5-1 시스템 구성 정보 표시

다음 `prtconf` 출력은 x86 기반 시스템에 표시됩니다.

```
# prtconf
System Configuration: Oracle Corporation i86pc
Memory size: 8192 Megabytes
System Peripherals (Software Nodes):

i86pc
  scsi_vhci, instance #0
  pci, instance #0
    pci108e,4843, instance #0
    pci8086,25e2, instance #0
    pci8086,3500, instance #7
```

```

        pci8086,3510, instance #9
        pci8086,3518, instance #10
            pci108e,4843, instance #0
            pci108e,4843, instance #1
        pci8086,350c, instance #8
pci8086,25e3 (driver not attached)
pci8086,25f8, instance #2
    pci108e,286, instance #0
        disk, instance #0
        disk, instance #2
        disk, instance #3
        disk, instance #1
pci8086,25e5 (driver not attached)
pci8086,25f9 (driver not attached)
pci8086,25e7 (driver not attached)
pci108e,4843, instance #0 (driver not attached)
pci108e,4843, instance #1
pci108e,4843, instance #2 (driver not attached)
pci108e,4843 (driver not attached)
pci108e,4843 (driver not attached)
pci108e,4843 (driver not attached)
pci108e,4843 (driver not attached)
pci8086,2690, instance #6
    pci108e,125e, instance #2
    pci108e,125e, instance #3
pci108e,4843, instance #0
pci108e,4843, instance #1
    device, instance #0
        keyboard, instance #0
        mouse, instance #1
pci108e,4843, instance #2
pci108e,4843, instance #3
pci108e,4843, instance #0
    storage, instance #0
        disk, instance #4
.
.
.

```

다음 sysdef 출력은 x86 기반 시스템에서 표시됩니다.

```

# sysdef
* Hostid
*
  29f10b4d
*
* i86pc Configuration
*
*
* Devices
*
+boot (driver not attached)
memory (driver not attached)
aliases (driver not attached)
chosen (driver not attached)
i86pc-memory (driver not attached)
i86pc-mmu (driver not attached)
openprom (driver not attached)

```

```

options, instance #0
packages (driver not attached)
delayed-writes (driver not attached)
itu-props (driver not attached)
isa, instance #0
  motherboard (driver not attached)
  pnpADP,1542, instance #0
  asy, instance #0
  asy, instance #1
  lp, instance #0 (driver not attached)
  fd, instance #0
    fd, instance #0
    fd, instance #1 (driver not attached)
  kd (driver not attached)
  kdmouse (driver not attached)
.
.
.

```

## 고장난 장치 해결

장치 회수 방식의 경우, FMA(고장 관리 프레임워크)에 따라 장치를 **고장**으로 격리합니다. 이 기능을 사용하면 고장난 장치를 안전하게 자동으로 비활성화하여 데이터 손실, 데이터 손상, 패닉 및 시스템 정지를 방지할 수 있습니다. 장치가 제거된 후 시스템의 안정성을 고려하여 제거 프로세스가 안전하게 완료됩니다.

중요한 장치는 제거되지 않습니다. 제거된 장치를 수동으로 교체해야 하는 경우 수동 교체 단계를 수행하는 것 이외에 시스템에서 장치가 교체된 것을 인식할 수 있도록 장치를 교체한 후 `fmadm repair` 명령을 사용합니다.

자세한 내용은 `fmadm(1M)`을 참조하십시오.

제거된 장치를 인식할 수 있도록 콘솔에 장치 제거에 대한 일반 메시지가 나타나고 `/var/adm/messages` 파일에 기록됩니다. 예를 들면 다음과 같습니다.

```
Aug 9 18:14 starbug genunix: [ID 751201 kern.notice] NOTICE: One or more I/O devices have been retired
```

`prtconf` 명령을 사용하여 제거된 특정 장치를 식별할 수 있습니다. 예를 들면 다음과 같습니다.

```

# prtconf
.
.
.
pci, instance #2
  scsi, instance #0
    disk (driver not attached)
    tape (driver not attached)
    sd, instance #3
    sd, instance #0 (retired)
  scsi, instance #1 (retired)
    disk (retired)

```

```

        tape (retired)
pci, instance #3
    network, instance #2 (driver not attached)
    network, instance #3 (driver not attached)
os-io (driver not attached)
iscsi, instance #0
pseudo, instance #0
.
.
.

```

## ▼ 고장난 장치를 해결하는 방법

다음 단계에 따라 고장난 장치 또는 회수 장치를 해결합니다.

zpool status 또는 fmdump 명령을 사용하여 ZFS 장치 문제나 오류 정보를 검토할 수도 있습니다. ZFS 장치 문제 또는 오류 정보의 경우 [Oracle Solaris 관리: ZFS 파일 시스템의 11 장, “Oracle Solaris ZFS 문제 해결 및 풀 복구”](#)를 참조하십시오.

### 1 fmadm faulty 명령을 사용하여 고장난 장치를 식별합니다.

```
# fmadm faulty
```

TIME	EVENT-ID	MSG-ID	SEVERITY
May 06 03:38:06	0376b4b6-fce7-c0f0-ffd9-a0a685376284	ZFS-8000-D3	Major

```

Host      : neo-2
Platform  : Sun-Fire-X4140   Chassis_id : 0904QAD02C
Product_sn :

Fault class : fault.fs.zfs.device
Affects     : zfs://pool=tank/vdev=c26c72a8ffcff889
              faulted and taken out of service
Problem in  : zfs://pool=tank/vdev=c26c72a8ffcff889
              faulted and taken out of service

Description : A ZFS device failed. Refer to http://sun.com/msg/ZFS-8000-D3 for
              more information.

Response    : No automated response will occur.

Impact      : Fault tolerance of the pool may be compromised.

Action      : Run 'zpool status -x' and replace the bad device.

```

### 2 고장난 장치 또는 회수 장치를 교체하거나 장치 오류를 해결합니다.

ZFS 저장소 풀에서 고장난 장치를 교체한 후에 장치 오류를 해결합니다. 예를 들면 다음과 같습니다.

```
# zpool clear rpool c4t0d0s0
```

간헐적으로 장치 오류가 발생하는데 장치를 교체하지 않은 경우에는 위에 표시된 것과 같이 오류를 해결할 수 있습니다.

**3 FMA 고장을 해결합니다.**

```
# fmadm repair 0376b4b6-fce7-c0f0-ffd9-a0a685376284
```

**4 고장이 해결된 것을 확인합니다.**

```
# fmadm faulty
```

## 시스템에 주변 장치 추가

핫 플러그를 지원하지 않는 새 주변 장치를 추가하는 과정에는 일반적으로 다음이 포함됩니다.

- 시스템 종료
- 장치를 시스템에 연결
- 시스템 재부트

핫 플러그를 지원하지 않는 다음 장치를 시스템에 추가하려면 [69 페이지](#) “주변 장치를 추가하는 방법”을 사용합니다.

- DVD 드라이브
- 보조 디스크 드라이브
- 테이프 드라이브
- SBUS 카드

경우에 따라 새 장치를 지원할 타사 장치 드라이버를 추가해야 할 수도 있습니다.

장치의 핫 플러그 기능에 대한 자세한 내용은 [6 장](#), “동적으로 장치 구성(작업)”을 참조하십시오.

### ▼ 주변 장치를 추가하는 방법

**1 관리자로 전환합니다.**

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

**2 (옵션) 장치를 지원하는 장치 드라이버를 추가해야 하는 경우에는 [70 페이지](#) “장치 드라이버를 추가하는 방법” 절차를 완료합니다.****3 시스템을 종료합니다.**

```
# shutdown -i0 -g30 -y
```

-i0     시스템 전원을 끄고 장치를 추가 및 제거하기에 좋은 0 init 상태로 전환합니다.

-g30    30초 안에 시스템을 종료합니다. 기본값은 60초입니다.

-y 사용자 개입 없이 시스템 종료를 계속합니다. 이 명령을 사용하지 않을 경우 종료 프로세스를 계속할지 묻는 프롬프트가 표시됩니다.

**4 다음 중 하나를 선택하여 시스템이 종료된 후에 전원을 끕니다.**

- SPARC 플랫폼의 경우 ok 프롬프트가 표시되면 안전하게 전원을 끌 수 있습니다.
- x86 플랫폼의 경우에는 type any key to continue 프롬프트가 표시되면 안전하게 전원을 끌 수 있습니다.

**5 모든 주변 장치의 전원을 끕니다.**

주변 장치의 전원 스위치 위치는 주변 장치와 함께 제공되는 하드웨어 설치 설명서를 참조하십시오.

**6 추가하는 장치의 대상 번호가 시스템에 있는 다른 장치와 다른지 확인하면서 주변 장치를 설치합니다.**

디스크 뒤쪽에 대상 번호를 선택하는 작은 스위치가 있는 경우도 많습니다.

장치 설치 및 연결에 대한 내용은 주변 장치와 함께 제공되는 하드웨어 설치 설명서를 참조하십시오.

**7 시스템의 전원을 켭니다.**

시스템이 다중 사용자 모드로 부트되고 로그인 프롬프트가 표시됩니다.

**8 장치에 대한 액세스를 시도하여 주변 장치가 추가되었는지 확인합니다.**

장치 액세스에 대한 자세한 내용은 71 페이지 “장치 액세스”를 참조하십시오.

## ▼ 장치 드라이버를 추가하는 방법

Oracle Solaris 11 릴리스에서는 pkg 명령을 사용하여 시스템에 패키지를 추가합니다. 장치 드라이버 패키지가 아직 SRV4 형식일 수도 있으므로 아래 단계에서는 pkg install 명령 대신 pkgadd 명령을 사용합니다.

이 절차에서는 장치가 이미 시스템에 추가된 경우를 가정합니다. 그렇지 않은 경우에는 62 페이지 “지원되지 않는 장치에 필요한 사항”을 참조하십시오.

**1 관리자로 전환합니다.**

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

**2 매체를 드라이브에 넣습니다.**

**3 드라이버를 설치합니다.**

```
# pkgadd [-d] device package-name
```

**4 패키지가 제대로 추가되었는지 확인합니다.**

```
# pkgchk package-name
#
```

패키지가 제대로 설치된 경우에는 응답 없이 시스템 프롬프트가 돌아옵니다.

**예 5-2 장치 드라이버 추가**

다음 예는 XYZdrv라는 패키지를 설치 및 확인하는 방법을 보여 줍니다.

```
# pkgadd XYZdrv
(licensing messages displayed)
.
.
.
Installing XYZ Company driver as <XYZdrv>
.
.
.
Installation of <XYZdrv> was successful.
# pkgchk XYZdrv
#
```

## 장치 액세스

명령을 사용하여 디스크, 파일 시스템 및 기타 장치를 관리할 때 장치 이름을 지정하는 방법을 알아야 합니다. 대부분의 경우에는 논리적 장치 이름을 사용하여 시스템에 연결된 장치를 나타낼 수 있습니다. 논리적 및 물리적 장치 이름은 모두 시스템에서 논리적 및 물리적 장치 파일로 나타냅니다.

### 장치 정보를 만드는 방법

시스템을 처음으로 부트하면 시스템에 연결된 모든 장치를 나타내는 장치 계층이 만들어집니다. 커널은 장치 계층 정보를 사용하여 드라이버를 적절한 장치와 연결합니다. 커널은 또한 드라이버에 특정 작업을 수행하는 포인터 세트를 제공합니다.

### 장치를 관리하는 방법

devfs 파일 시스템은 시스템에 있는 모든 장치의 네임스페이스인 /devices 디렉토리를 관리합니다. 이 디렉토리는 실제 버스 및 장치 주소로 구성된 물리적 장치를 나타냅니다.

dev 파일 시스템은 논리적 장치 이름의 네임스페이스인 /dev 디렉토리를 관리합니다.

기본적으로 devfsadm 명령은 시스템에 있는 모든 드라이버를 로드하여 가능한 모든 장치 인스턴스에 연결하려고 합니다. 다음으로 devfsadm은 /devices 디렉토리에 장치 파일을 만들고 /dev 디렉토리에 논리적 링크를 만듭니다. devfsadm 명령은 또한 path\_to\_inst 인스턴스 데이터베이스를 유지 관리합니다.

동적 재구성 이벤트나 파일 시스템 액세스에 따른 /dev 및 /devices 디렉토리 업데이트는 devfsadm 명령의 데몬 버전인 devfsadmd에서 처리합니다. 이 데몬은 시스템을 부트할 때 서비스 관리 기능에서 시작됩니다.

devfsadmd 데몬은 재구성 이벤트에서 만들어진 장치 구성 변경을 자동으로 감지하므로 이 명령을 대화식으로 실행할 필요는 없습니다.

자세한 내용은 다음을 참조하십시오.

- [devfsadm\(1M\)](#)
- [dev\(7FS\)](#)
- [devfs\(7FS\)](#)
- [path\\_to\\_inst\(4\)](#)

## 장치 이름 지정 규약

Oracle Solaris OS에서는 장치를 세 가지 방법으로 참조합니다.

- **물리적 장치 이름** - 장치 정보 계층의 전체 장치 경로 이름을 나타냅니다. 물리적 장치 이름은 장치를 시스템에 처음 추가할 때 만들어집니다. 물리적 장치 파일은 /devices 디렉토리에서 찾을 수 있습니다.
- **인스턴스 이름** - 시스템에 있는 가능한 모든 장치의 커널 축약 이름을 나타냅니다. 예를 들어 sd0 및 sd1은 두 디스크 장치의 인스턴스 이름을 나타냅니다. 인스턴스 이름은 /etc/path\_to\_inst 파일에 매핑됩니다.
- **논리적 장치 이름** - 논리적 장치 이름은 장치를 시스템에 처음 추가할 때 만들어집니다. 논리적 장치 이름은 대부분의 파일 시스템 명령에서 장치를 나타내는 데 사용됩니다. 논리적 장치 이름을 사용하는 파일 명령의 목록은 [표 5-3](#)을 참조하십시오. /dev 디렉토리에 있는 논리적 장치 파일은 /devices 디렉토리의 물리적 장치 파일과 심볼릭 링크로 연결됩니다.

이와 같은 장치 이름 정보는 다음 명령을 사용하여 표시합니다.

- `dmesg`
- `format`
- `sysdef`
- `pvtconf`



## 논리적 디스크 장치 이름

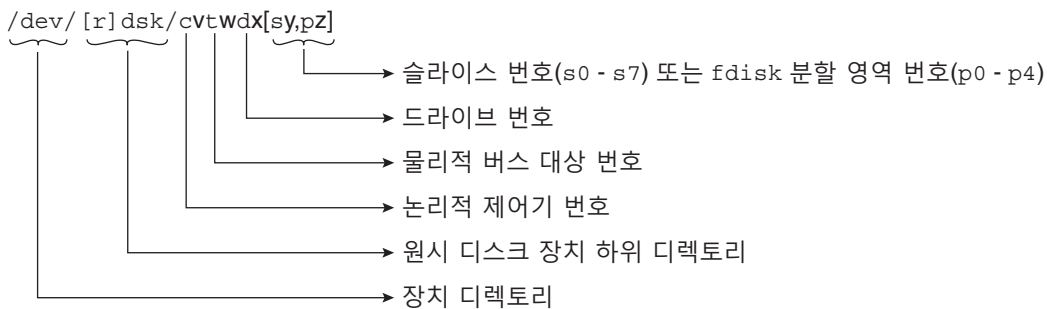
논리적 장치 이름은 다음과 같은 작업을 수행할 때 디스크 장치 액세스에 사용됩니다.

- 시스템에 새 디스크 추가
- 디스크를 한 시스템에서 다른 시스템으로 이동
- 로컬 디스크에 상주하는 파일 시스템 액세스 또는 마운트
- 로컬 파일 시스템 백업

관리 명령 중에는 디스크 슬라이스 또는 파일 시스템을 참조하는 인수를 사용하는 것이 많습니다.

심볼릭 링크로 연결된 하위 디렉토리(/dev/dsk 또는 /dev/rdisk)를 지정하고 특정 제어기, 디스크 또는 슬라이스를 식별하는 문자열을 추가하여 디스크 장치를 참조합니다.

그림 5-1 논리적 장치 이름 설명



### 디스크 하위 디렉토리 지정

일부 디스크 및 파일 관리 명령에는 **원시**(또는 **문자**) 장치 인터페이스나 **블록** 장치 인터페이스를 사용해야 합니다. 둘을 구분하는 기준은 장치에서 데이터를 읽는 방법입니다.

원시 장치 인터페이스는 한 번에 적은 양의 데이터만 전송합니다. 블록 장치 인터페이스에는 한 번에 대량의 데이터 블록을 읽는 버퍼가 포함됩니다.

명령에 따라 다른 인터페이스가 필요합니다.

- 명령에 원시 장치 인터페이스가 필요한 경우에는 /dev/rdisk 하위 디렉토리를 지정합니다. (rdisk의 "r"은 "raw(원시)"를 나타냅니다.)
- 명령에 블록 장치 인터페이스가 필요한 경우에는 /dev/dsk 하위 디렉토리를 지정합니다.

- 명령에 /dev/dsk와 /dev/rdisk 중 어느 것을 사용해야 하는지 확실하지 않은 경우에는 명령의 매뉴얼 페이지를 참조하십시오.

다음 표는 일반적으로 사용되는 일부 디스크 및 파일 시스템 명령에 필요한 인터페이스를 표시합니다.

표 5-3 자주 사용되는 일부 명령에 필요한 장치 인터페이스 유형

명령 참조	인터페이스 유형	사용 예
<code>dumpadm(1M)</code>	블록	<code>dumpadm -d /dev/zvol/dsk/rpool/dump</code>
<code>prtvtoc(1M)</code>	원시	<code>prtvtoc /dev/rdisk/c0t0d0s0</code>
<code>swap(1M)</code>	블록	<code>swap -a /dev/zvol/dsk/rpool/swap</code>

## 직접 및 버스 지향 제어기

디스크 장치가 직접 또는 버스 지향 제어기 중 어느 쪽에 연결되었는지에 따라 디스크 분할 영역이나 슬라이스에 다르게 액세스할 수 있습니다. 일반적으로 직접 제어기는 논리적 장치 이름에 **대상** 식별자가 포함되지 않습니다.

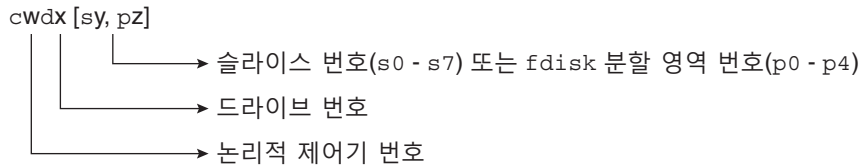
다음 세부 섹션에서는 두 가지 제어기 유형 모두에 대한 규약을 설명합니다.

**주** - 제어기 번호는 시스템을 초기화하는 동안 자동으로 할당됩니다. 번호는 논리적으로만 지정되며 물리적 제어기에 대한 직접 매핑을 의미하지 않습니다.

## x86: 직접 제어기를 사용하는 디스크

x86 기반 시스템에서 IDE 제어기를 사용하는 디스크의 슬라이스를 지정하려면 다음 그림에 표시된 이름 지정 규약에 따릅니다.

그림 5-2 x86: 직접 제어기를 사용하는 디스크



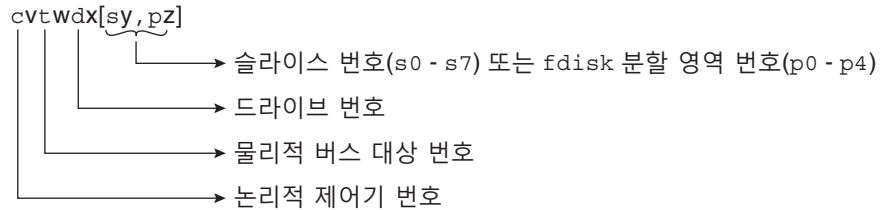
전체 Solaris fdisk 분할 영역을 나타내려면 슬라이스 2(s2)를 지정합니다.

시스템에 제어기가 하나뿐인 경우 *w*는 일반적으로 0입니다.

## 버스 지향 제어기를 사용하는 디스크

SCSI 등의 버스 지향 제어기를 사용하는 디스크의 슬라이스를 지정하려면 다음 그림에 표시된 이름 지정 규약에 따릅니다.

그림 5-3 버스 지향 제어기를 사용하는 디스크



UltraSPARC 시스템에 IDE 디스크 등의 직접 연결된 디스크가 있는 SPARC 기반 시스템의 이름 지정 규약은 버스 지향 제어기가 있는 시스템의 경우와 같습니다.

시스템에 제어기가 하나뿐인 경우  $w$ 는 일반적으로 0입니다.

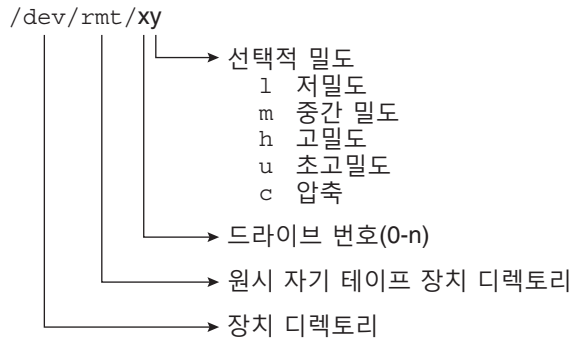
SCSI 제어기의 경우  $x$ 는 장치 뒤쪽에 있는 스위치로 설정한 대상 주소이고  $y$ 는 대상에 연결된 드라이브의 LUN(논리 장치 번호)입니다. 디스크에 내장 제어기가 있는 경우  $y$ 는 보통 0입니다.

전체 디스크를 나타내려면 슬라이스 2( $s2$ )를 지정합니다.

## 논리적 테이프 장치 이름

논리적 테이프 장치 파일은 `/devices` 디렉토리의 심볼링 링크로 `/dev/rmt/*` 디렉토리에 있습니다.

그림 5-4 논리적 테이프 장치 이름



시스템에 연결된 첫번째 테이프 장치는 0(/dev/rmt/0)입니다. 테이프 밀도 값(l, m, h, c 및 u)에 대한 설명은 21 장, “테이프 드라이브 관리(작업)”를 참조하십시오.

## 논리적 이동식 매체 장치 이름

이동식 매체는 이동식 매체 관리 서비스에서 관리하기 때문에 매체를 수동으로 마운트하는 경우가 아니면 논리적 장치 이름이 거의 사용되지 않습니다.

이동식 매체 장치를 나타내는 논리적 장치 이름에 대한 설명은 3 장, “이동식 매체에 액세스(작업)”를 참조하십시오.

## 동적으로 장치 구성(작업)

---

이 장은 Oracle Solaris OS에서 동적으로 장치를 구성하는 작업에 대한 지침을 제공합니다. 시스템 구성 요소가 핫 플러그 기능을 지원하는 경우에는 시스템이 아직 실행 중인 동안 Oracle Solaris OS에서 장치를 추가, 제거 또는 교체할 수 있습니다. 시스템 구성 요소가 핫 플러그 기능을 지원하지 않는 경우에는 시스템을 재부트하여 장치를 재구성할 수 있습니다.

장치의 동적 구성과 관련된 절차에 대한 자세한 내용은 다음을 참조하십시오.

- 83 페이지 “`cfgadm` 명령을 사용하여 SCSI 핫 플러그 기능 사용(작업 맵)”
- 93 페이지 “`cfgadm` 명령을 사용하여 PCI 또는 PCIe 핫 플러그 기능 사용(작업 맵)”
- 99 페이지 “`cfgadm` 명령을 사용하여 SATA 핫 플러그 기능 사용”
- 103 페이지 “응용 프로그램 개발자 RCM 스크립트(작업 맵)”
- 104 페이지 “시스템 관리자 RCM 스크립트(작업 맵)”

`cfgadm` 명령을 이용한 USB 장치의 핫 플러그 기능 사용에 관련된 자세한 내용은 151 페이지 “`cfgadm` 명령을 사용하여 USB 장치 핫 플러그 기능 사용”을 참조하십시오.

`cfgadm` 명령을 이용한 InfiniBand 장치의 핫 플러그 기능 사용에 관련된 자세한 내용은 9 장, “InfiniBand 장치 사용(개요/작업)”을 참조하십시오.

장치 액세스에 대한 자세한 내용은 71 페이지 “장치 액세스”를 참조하십시오.

## 동적 재구성 및 핫 플러그 기능 사용

핫 플러그는 시스템을 실행하는 동안 시스템 구성 요소를 물리적으로 추가, 제거 또는 교체하는 기능입니다. 동적 재구성은 시스템 구성 요소에서 핫 플러그를 사용하는 기능입니다. 이 용어는 시스템에서 시스템 리소스(하드웨어 및 소프트웨어 모두)를 이동하거나 시스템에서 물리적으로 제거하지 않으면서 사용 안함으로 설정하는 일반적인 능력을 나타내기도 합니다.

일반적으로 다음과 같은 버스 유형에 핫 플러그를 사용할 수 있습니다.

- USB
- Fibre Channel
- 1394
- ATA
- SCSI
- Infiniband

또한 `cfgadm` 명령을 사용하여 다음과 같은 장치에서 핫플러그 기능을 사용할 수 있습니다.

- SPARC 및 x86 플랫폼의 USB 장치
- SPARC 및 x86 플랫폼의 SCSI 또는 SAS 장치
- SPARC 및 x86 플랫폼의 PCI 장치
- SPARC 및 x86 플랫폼의 PCIe 장치
- SPARC 및 x86 플랫폼의 SATA 장치
- SPARC 및 x86 플랫폼의 InfiniBand 장치

`cfgadm` 명령의 기능에는 다음이 포함됩니다.

- 시스템 구성 요소 상태 표시
- 시스템 구성 요소 테스트
- 구성 요소의 구성 변경
- 구성 도움말 메시지 표시

`cfgadm` 명령을 사용하여 시스템 구성 요소를 재구성하는 경우의 이점은 시스템을 실행하는 동안 구성 요소를 추가, 제거 또는 교체할 수 있다는 것입니다. 또한 시스템 구성 요소의 추가, 제거 또는 교체에 필요한 단계를 `cfgadm` 명령이 안내한다는 이점도 있습니다.

구성 요소의 핫플러그 기능을 사용하는 단계별 지침은 다음을 참조하십시오.

- 84 페이지 “`cfgadm` 명령을 사용하여 SCSI 핫플러그 기능 사용”
- 94 페이지 “`cfgadm` 명령을 사용하여 PCI 또는 PCIe 핫플러그 기능 사용”
- 99 페이지 “`cfgadm` 명령을 사용하여 SATA 핫플러그 기능 사용”
- `cfgadm(1M)`

---

주 - 일부 SCSI 및 PCI 제어기는 `cfgadm` 명령을 사용한 핫플러그 기능을 지원하지 않습니다.

---

동적 재구성은 Oracle 고가용성 전략의 일부로서 대체 경로나 페일오버 소프트웨어와 같은 추가 계층의 제품과 함께 사용됩니다. 두 제품 모두 장치 오류가 발생한 경우에 고장 허용 능력을 제공합니다.

고가용성 소프트웨어가 없는 경우에는 해당 응용 프로그램을 수동으로 중지하고 중요하지 않은 파일 시스템을 마운트 해제한 다음 추가 또는 제거 작업을 진행할 수 있습니다.

주 - 일부 시스템은 핫플러그 기능을 지원하는 슬롯과 핫플러그 기능을 지원하지 않는 슬롯이 모두 있습니다. 특정 하드웨어 구성(엔터프라이즈 레벨 시스템 등)의 핫플러그 가능 장치에 대한 자세한 내용은 하드웨어 구성 설명서를 참조하십시오.

## 첨부 지점

`cfgadm` 명령은 시스템에서 동적 재구성 작업을 진행할 수 있는 위치인 **첨부 지점**에 대한 정보를 표시합니다.

첨부 지점은 다음으로 구성됩니다.

- **점유자:** 시스템에 구성할 수 있는 하드웨어 구성 요소를 나타냅니다.
- **컨센트:** 점유자를 받는 위치입니다.

첨부 지점은 논리적 및 물리적 첨부 지점 ID(`Ap_Id`)로 표현합니다. 물리적 `Ap_Id`는 첨부 지점의 물리적인 경로 이름입니다. 논리적 `Ap_Id`는 물리적 `Ap_Id`를 대체하는, 사용자에게 친숙한 ID입니다. `Ap_Id`에 대한 자세한 내용은 `cfgadm(1M)`을 참조하십시오.

SCSI HBA(호스트 버스 어댑터) 또는 SCSI 제어기의 논리적 `Ap_Id`는 일반적으로 `c0`과 같은 제어기 번호로 표현됩니다.

SCSI HBA에 제어기 번호를 할당하지 않은 경우에는 내부에서 생성된 고유 식별자가 제공됩니다. SCSI 제어기의 고유 식별자 예는 다음과 같습니다.

```
fas1:scsi
```

SCSI 장치의 논리적 `Ap_Id`는 일반적으로 다음과 같은 형식입니다.

```
HBA-logical-apid::device-identifier
```

다음 예에서 `c0`은 SCSI HBA의 논리적 `Ap_Id`입니다.

```
c0::dsk/c0t3d0
```

장치 식별자는 보통 `/dev` 디렉토리에 있는 해당 장치의 논리적 장치 이름에서 파생됩니다. 예를 들어 논리적 장치 이름이 `/dev/rmt/1`인 테이프 장치의 논리적 `Ap_Id`는 다음과 같습니다.

```
c0::rmt/1
```

`/dev` 디렉토리에 있는 논리적 이름에서 SCSI 장치의 논리적 `Ap_Id`가 파생되지 않는 경우에는 내부에서 생성한 고유 식별자가 제공됩니다. `/dev/rmt/1` 테이프 장치 식별자의 예는 다음과 같습니다.

```
c0::st4
```

SCSI `Ap_Id`에 대한 자세한 내용은 `cfgadm_scsi(1M)`를 참조하십시오.

cfgadm 명령은 공통의 상태(구성됨 및 구성되지 않음 등) 및 작업(연결, 구성, 구성 해제 등) 세트에 관련된 모든 리소스 및 동적 재구성 작업을 나타냅니다. 이러한 공통 상태 및 작업에 대한 자세한 내용은 [cfgadm\(1M\)](#)을 참조하십시오.

다음 표는 SCSI HBA 첨부 지점의 콘센트 및 점유자 상태를 보여 줍니다.

콘센트 상태	설명	점유자 상태	설명
empty	SCSI HBA에는 해당 없음	configured	버스에 하나 이상의 장치가 구성됨
disconnected	버스 비활동 상태	unconfigured	장치가 구성되지 않음
connected	버스 활성 상태		

다음 표는 SCSI 장치 첨부 지점의 콘센트 및 점유자 상태를 보여 줍니다.

콘센트 상태	설명	점유자 상태	설명
empty	SCSI 장치에는 해당 없음	configured	장치가 구성됨
disconnected	버스 비활동 상태	unconfigured	장치가 구성되지 않음
connected	버스 활성 상태		

특수 하드웨어에서 따로 표시하지 않는 한 SCSI 첨부 지점의 상태는 알 수 없습니다. SCSI 구성 요소 정보의 표시에 대한 지침은 [84 페이지](#) “SCSI 장치에 대한 정보를 표시하는 방법”을 참조하십시오.

## PCI 또는 PCIe 어댑터 카드 분리

장치 드라이버가 핫플러그 기능을 지원하는 경우에는 필수적이지 않은 시스템 리소스를 호스팅하는 PCI 어댑터 카드를 제거할 수 있습니다. 필수적인 시스템 리소스인 경우에는 PCI 어댑터 카드를 분리할 수 없습니다. PCI 어댑터 카드를 분리하려면 다음과 같은 조건이 충족되어야 합니다.

- 장치 드라이버가 핫플러그 기능을 지원해야 합니다.
- 대체 경로를 통해 중요한 리소스에 액세스할 수 있어야 합니다.

예를 들어, 시스템에 이더넷 카드가 하나만 설치되어 있는 경우 이더넷 카드를 분리하면 네트워크 연결이 끊어집니다. 이렇게 분리하면서 네트워크 연결을 활성 상태로 유지하려면 추가 계층의 소프트웨어 지원이 필요합니다.



## PCI 또는 PCIe 어댑터 카드 연결

다음과 같은 조건이 충족되면 PCI 어댑터 카드를 시스템에 추가할 수 있습니다.

- 사용 가능한 슬롯이 있습니다.
- 장치 드라이버가 이 어댑터 카드에 대해 핫플러그 기능을 지원합니다.

PCI 어댑터 카드 추가 또는 제거에 대한 단계별 지침은 94 페이지 “[cfgadm 명령을 사용하여 PCI 또는 PCIe 핫플러그 기능 사용](#)”을 참조하십시오.

## hotplug 명령을 사용하여 PCIe 핫플러그 기능 사용

hotplug 명령을 사용하여 핫플러그 가능한 연결을 사용할 수 있습니다. PCIe(PCI Express) 및 PCI SHPC 장치의 커넥터 또는 포트 연결에만 해당됩니다. 핫플러그 커넥터는 시스템에서 구성 요소를 삽입 또는 제거하는 위치의 물리적 표현입니다. 핫플러그 포트는 시스템 장치 트리에서 시스템에 대한 장치의 연결을 관리하는 위치의 논리적 표현입니다.

hotplug 기능을 사용하여 장치를 시스템에서 물리적으로 추가 또는 제거하지 않고도 온라인 또는 오프라인 상태로 전환할 수 있습니다(내장 장치의 경우도 마찬가지).

hotplug 명령으로 장치를 관리하려면 핫플러그 서비스를 사용해야 합니다.

```
# svcadm enable svc:/system/hotplug:default
```

다음 예에서는 hotplug 명령을 사용하는 방법을 설명합니다.

다음과 같이 시스템에 있는 모든 PCI/PCIe 핫플러그 가능 커넥터/포트(가상 및 물리적)를 표시합니다.

```
# hotplug list -lv
```

PCIe 슬롯에 이더넷 카드를 구성합니다. 예를 들면 다음과 같습니다.

```
# hotplug enable /pci0,0 pcie0
```

PCIe 슬롯의 이더넷 카드를 구성 해제합니다. 예를 들면 다음과 같습니다.

```
# hotplug disable /pci0,0 pcie0
```

PCI 장치 노드를 오프라인 상태로 만들어 해당 노드의 장치 드라이버를 분리합니다. 예를 들면 다음과 같습니다.

```
# hotplug offline /pci0,0/pci1 pci.0,2
```

PCI 장치 노드를 온라인 상태로 만들어 해당 노드에 드라이버를 연결합니다. 예를 들면 다음과 같습니다.

```
# hotplug online /pci0,0/pci1 pci.0,2
```

IOV 물리적 기능의 종속 항목 포트를 설치합니다. 예를 들면 다음과 같습니다.

```
# hotplug install /pci@400/pci@1/pci@0/pci@4 pci.0,1
```

그런 다음 탐색된 결과 IOV 가상 기능을 표시합니다. 예를 들면 다음과 같습니다.

```
# hotplug list -v /pci@400/pci@1/pci@0/pci@4 pci.0,1
<pci.0,1> (ONLINE)
  { IOV physical function }
  { IOV virtual function 'pci.0,81' }
  { IOV virtual function 'pci.0,83' }
  { IOV virtual function 'pci.0,85' }
  { IOV virtual function 'pci.0,87' }
<pci.0,81> (OFFLINE)
ethernet@0,81
<pci.0,83> (OFFLINE)
ethernet@0,83
<pci.0,85> (OFFLINE)
ethernet@0,85
<pci.0,87> (OFFLINE)
ethernet@0,87
```

IOV 물리적 기능의 종속 항목 포트를 제거합니다. 예를 들면 다음과 같습니다.

```
# hotplug uninstall /pci@400/pci@1/pci@0/pci@4 pci.0,0
```

종속 항목 IOV 가상 기능이 사용 중이면 이 작업이 실패합니다. 예를 들면 다음과 같습니다.

```
# hotplug uninstall /pci@400/pci@1/pci@0/pci@4 pci.0,0
ERROR: devices or resources are busy.
ethernet@0,81:
  { Network interface igbvf1 }
  { igbvf1: hosts IP addresses: 10.0.0.1 }
  { Plumbed IP Address }
```

## PCI 핫플러그 작업 문제 해결(hotplug)

핫플러그 포트에 연결된 장치에서 다음과 같은 유지 관리 상태가 표시될 수 있습니다.

```
/pci@0,0/pci10de,5d@e <pci.a,1> (MAINTENANCE)
.
.
./pci@0,0/pci108e,534a@d <pci.0,0> (MAINTENANCE-SUSPENDED)
```

위 메시지는 고장 이벤트나 유지 관리 작업이 발생한 것을 나타냅니다. MAINTENANCE 상태는 장치가 사용 중이지만 완전히 작동하지 않는 경우를 의미합니다. MAINTENANCE-SUSPENDED 상태는 장치가 유지 관리 작업 때문에 **라이브 일시 중단된** 경우를 의미합니다. 예를 들면 장치 하드웨어 재구성이 있습니다.

hotplug 명령을 사용하려면 다음 서비스가 실행 중이어야 합니다.

svc:/system/hotplug:default

그렇지 않으면 다음 메시지가 표시됩니다.

ERROR: hotplug service is not available.

지원되는 I/O 버스가 없는 시스템에는 다음과 같은 오류 메시지가 표시됩니다.

ERROR: there are no connections to display.  
(See hotplug(1m) for more information.)

위 메시지는 시스템에 핫 플러그 가능한 다른 I/O 장치가 있을 수도 있지만 그런 장치를 관리하려면 hotplug 명령이 아닌 cfgadm 명령을 사용해야 한다는 것을 의미할 수 있습니다.

## cfgadm 명령을 사용하여 SCSI 핫 플러그 기능 사용(작업 맵)

작업	설명	수행 방법
SCSI 장치에 대한 정보를 표시합니다.	SCSI 제어기 및 장치에 대한 정보를 표시합니다.	84 페이지 “SCSI 장치에 대한 정보를 표시하는 방법”
SCSI 제어기를 구성 해제합니다.	SCSI 제어기를 구성 해제합니다.	85 페이지 “SCSI 제어기를 구성 해제하는 방법”
SCSI 제어기를 구성합니다.	이전에 구성 해제한 SCSI 제어기를 구성합니다.	85 페이지 “SCSI 제어기를 구성하는 방법”
SCSI 장치를 구성합니다.	특정 SCSI 장치를 구성합니다.	86 페이지 “SCSI 장치를 구성하는 방법”
SCSI 제어기의 연결을 해제합니다.	특정 SCSI 제어기의 연결을 해제합니다.	86 페이지 “SCSI 제어기를 연결 해제하는 방법”
SCSI 제어기를 연결합니다.	이전에 연결을 해제한 특정 SCSI 제어기를 연결합니다.	87 페이지 “SPARC: SCSI 제어기를 연결하는 방법”
SCSI 장치를 SCSI 버스에 추가합니다.	특정 SCSI 장치를 SCSI 버스에 추가합니다.	88 페이지 “SPARC: SCSI 장치를 SCSI 버스에 추가하는 방법”
SCSI 제어기에서 동일한 장치를 교체합니다.	SCSI 버스에 있는 장치를 같은 유형의 다른 장치로 교체합니다.	89 페이지 “SPARC: SCSI 제어기에서 동일한 장치를 교체하는 방법”
SCSI 장치를 제거합니다.	SCSI 장치를 시스템에서 제거합니다.	90 페이지 “SPARC: SCSI 장치를 제거하는 방법”
SCSI 구성 문제를 해결합니다.	실패한 SCSI 구성 해제 작업을 해결합니다.	93 페이지 “실패한 SCSI 구성 해제 작업을 해결하는 방법”

## cfgadm 명령을 사용하여 SCSI 핫 플러그 기능 사용

이 섹션에서는 `cfgadm` 명령으로 수행할 수 있는 다양한 SCSI 핫 플러그 절차를 설명합니다.

주 - SCSI 프레임워크는 일반적으로 SCSI 장치의 핫 플러그 기능을 지원합니다. 그러나 특정 SCSI 장치에 대해 핫 플러그 기능이 지원되는지 확인하려면 하드웨어 설명서를 참조해야 합니다.

이러한 절차는 특정 장치를 예로 들며 `cfgadm` 명령을 사용하여 SCSI 구성 요소의 핫 플러그 기능을 사용하는 방법을 설명합니다. 제공하는 장치 정보와 `cfgadm` 명령에서 표시하는 정보는 시스템 구성에 따라 다릅니다.

이 섹션의 모든 절차를 수행하려면 사용자 계정에는 일반적으로 부여되지 않는 관리 권한이 필요합니다. 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

### ▼ SCSI 장치에 대한 정보를 표시하는 방법

다음 절차는 예에 사용된 SCSI 제어기 `c0` 및 `c1`과 여기에 연결된 장치를 사용하여 `cfgadm` 명령으로 표시할 수 있는 장치 구성 정보 유형을 표시합니다.

주 - SCSI 장치가 `cfgadm` 명령에서 지원되지 않는 경우에는 `cfgadm` 명령 출력에 장치가 표시되지 않습니다.

- 1 관리자로 전환합니다.
- 2 시스템의 첨부 지점에 대한 정보를 표시합니다.

```
# cfgadm -l
Ap_Id          Type          Receptacle  Occupant    Condition
c0             scsi-bus     connected   configured  unknown
c1             scsi-bus     connected   configured  unknown
```

이 예에서 `c0`와 `c1`은 두 개의 SCSI 제어기를 나타냅니다.

- 3 시스템의 SCSI 제어기와 연결된 장치에 대한 정보를 표시합니다.

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c0             scsi-bus     connected   configured  unknown
c0::dsk/c0t0d0 disk         connected   configured  unknown
c0::rmt/0      tape         connected   configured  unknown
c1             scsi-bus     connected   configured  unknown
c1::dsk/c1t3d0 disk         connected   configured  unknown
c1::dsk/c1t4d0 unavailable  connected   unconfigured unknown
```

주 - `cfgadm -l` 명령은 SCSI HBA에 대한 정보를 표시하지만 SCSI 장치에 대한 정보는 표시하지 않습니다. 디스크 및 테이프와 같은 SCSI 장치에 대한 정보를 표시하려면 `cfgadm -al` 명령을 사용합니다.

## ▼ SCSI 제어를 구성 해제하는 방법

다음 절차는 SCSI 제어기 `c1`을 사용하여 SCSI 제어를 구성 해제하는 과정의 예를 보여 줍니다.

- 1 관리자로 전환합니다.
- 2 SCSI 제어를 구성 해제합니다.
- 3 SCSI 제어기가 구성 해제되었는지 확인합니다.

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c0             scsi-bus     connected   configured  unknown
c0::disk/c0t0d0 disk         connected   configured  unknown
c0::rmt/0      tape         connected   configured  unknown
c1             scsi-bus     connected   unconfigured unknown
```

`c1`의 `Occupant` 열에 `unconfigured`가 나타나며 SCSI 버스에 구성된 점유자가 없는 것을 표시하는지 확인합니다.

구성 해제 작업이 실패한 경우에는 93 페이지 “실패한 SCSI 구성 해제 작업을 해결하는 방법”을 참조하십시오.

## ▼ SCSI 제어를 구성하는 방법

다음 절차는 SCSI 제어기 `c1`을 사용하여 SCSI 제어를 구성하는 과정의 예를 보여 줍니다.

- 1 관리자로 전환합니다.
- 2 SCSI 제어를 구성합니다.
- 3 SCSI 제어기가 구성되었는지 확인합니다.

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c0             scsi-bus     connected   configured  unknown
c0::disk/c0t0d0 disk         connected   configured  unknown
c0::rmt/0      tape         connected   configured  unknown
```

```
c1                scsi-bus    connected    configured    unknown
c1::dsk/c1t3d0   disk        connected    configured    unknown
c1::dsk/c1t4d0   unavailable connected    unconfigured  unknown
```

이전의 구성 해제 절차에서는 SCSI 버스에 있는 모든 장치를 제거했습니다. 이제 모든 장치가 다시 시스템에 구성되었습니다.

## ▼ SCSI 장치를 구성하는 방법

다음 절차는 SCSI 디스크 c1t4d0을 사용하여 SCSI 제어를 구성하는 과정의 예를 보여줍니다.

- 1 관리자로 전환합니다.
- 2 구성할 장치를 식별합니다.

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c0             scsi-bus     connected   configured  unknown
c0::dsk/c0t0d0 disk         connected   configured  unknown
c0::rmt/0      tape         connected   configured  unknown
c1             scsi-bus     connected   configured  unknown
c1::dsk/c1t3d0 disk         connected   configured  unknown
c1::dsk/c1t4d0 unavailable  connected   unconfigured unknown
```

- 3 SCSI 장치를 구성합니다.
- 4 SCSI 장치가 구성되었는지 확인합니다.

```
# cfgadm -c configure c1::dsk/c1t4d0
```

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c0             scsi-bus     connected   configured  unknown
c0::dsk/c0t0d0 disk         connected   configured  unknown
c0::rmt/0      tape         connected   configured  unknown
c1             scsi-bus     connected   configured  unknown
c1::dsk/c1t3d0 disk         connected   configured  unknown
c1::dsk/c1t4d0 disk         connected   configured  unknown
```

## ▼ SCSI 제어를 연결 해제하는 방법



주의 - SCSI 장치의 연결을 해제하는 경우 주의가 필요하며 루트(/), usr, var 및 swap 분할 영역과 같이 중요한 파일 시스템이 포함된 디스크의 제어기에서 작업하는 경우에는 특히 조심해야 합니다. 동적 재구성 소프트웨어로 시스템이 중단될 수 있는 경우를 모두 감지할 수는 없습니다. 이 절차는 주의해서 사용해야 합니다.

다음 절차는 SCSI 제어기 c1을 사용하여 SCSI 장치를 연결 해제하는 과정의 예를 보여줍니다.

- 1 관리자로 전환합니다.
- 2 장치의 연결을 해제하기 전에 연결되어 있는지 확인합니다.

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c0             scsi-bus     connected   configured  unknown
c0::dsk/c0t0d0 disk         connected   configured  unknown
c0::rmt/0      tape         connected   configured  unknown
c1             scsi-bus     connected   configured  unknown
c1::dsk/c1t3d0 disk         connected   configured  unknown
c1::dsk/c1t4d0 disk         connected   configured  unknown
```

- 3 SCSI 제어기를 연결 해제합니다.

```
# cfgadm -c disconnect c1
WARNING: Disconnecting critical partitions may cause system hang.
Continue (yes/no)? y
```



주의 - 이 명령은 `cfgadm -c connect` 명령을 사용할 때까지 SCSI 버스의 모든 I/O 작업을 일시 중단합니다. `cfgadm` 명령은 중요한 분할 영역의 연결이 끊어지지 않도록 기본적인 검사를 수행하지만 가능한 모든 경우를 감지할 수는 없습니다. 이 명령을 잘못 사용하면 시스템이 중단되고 시스템을 재부트해야 할 수도 있습니다.

- 4 SCSI 버스가 연결 해제되어 있는지 확인합니다.

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c0             scsi-bus     connected   configured  unknown
c0::dsk/c0t0d0 disk         connected   configured  unknown
c0::rmt/0      tape         connected   configured  unknown
c1             unavailable  disconnected  configured  unknown
c1::dsk/c1t3d0 unavailable  disconnected  configured  unknown
c1::dsk/c1t4d0 unavailable  disconnected  configured  unknown
```

제어기와 제어기에 연결된 모든 장치가 시스템에서 연결 해제됩니다.

## ▼ SPARC: SCSI 제어기를 연결하는 방법

다음 절차는 SCSI 제어기 c1을 사용하여 SCSI 제어기를 연결하는 과정의 예를 보여줍니다.

- 1 관리자로 전환합니다.
- 2 장치를 연결하기 전에 연결 해제되어 있는지 확인합니다.

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c0             scsi-bus     connected   configured  unknown
c0::dsk/c0t0d0 disk         connected   configured  unknown
c0::rmt/0      tape         connected   configured  unknown
c1             unavailable  disconnected  configured  unknown
```

```
c1::dsk/c1t3d0      unavailable  disconnected  configured  unknown
c1::dsk/c1t4d0      unavailable  disconnected  configured  unknown
```

**3 SCSI 제어기를 연결합니다.**

```
# cfgadm -c connect c1
```

**4 SCSI 제어기가 연결되었는지 확인합니다.**

```
# cfgadm -al
Ap_Id                Type           Receptacle    Occupant      Condition
c0                   scsi-bus      connected     configured    unknown
c0::dsk/c0t0d0      disk          connected     configured    unknown
c0::rmt/0            tape          connected     configured    unknown
c1                   scsi-bus      connected     configured    unknown
c1::dsk/c1t3d0      disk          connected     configured    unknown
c1::dsk/c1t4d0      disk          connected     configured    unknown
```

## ▼ SPARC: SCSI 장치를 SCSI 버스에 추가하는 방법

이 예에서는 SCSI 제어기 c1을 사용하여 SCSI 장치를 SCSI 버스에 추가하는 방법을 보여 줍니다.

---

주 - 장치를 추가하는 경우, 장치 자체의 Ap\_Id가 아니라 장치를 연결하는 SCSI HBA(제어기)의 Ap\_Id를 사용합니다.

---

**1 관리자로 전환합니다.**

**2 현재 SCSI 구성을 확인합니다.**

```
# cfgadm -al
Ap_Id                Type           Receptacle    Occupant      Condition
c0                   scsi-bus      connected     configured    unknown
c0::dsk/c0t0d0      disk          connected     configured    unknown
c0::rmt/0            tape          connected     configured    unknown
c1                   scsi-bus      connected     configured    unknown
c1::dsk/c1t3d0      disk          connected     configured    unknown
```

**3 SCSI 장치를 SCSI 버스에 추가합니다.**

**a. 다음 cfgadm 명령을 입력합니다.**

예를 들면 다음과 같습니다.

```
# cfgadm -x insert_device c1
Adding device to SCSI HBA: /devices/sbus@1f,0/SUNW,fas@1,8800000
This operation will suspend activity on SCSI bus: c1
```

**b. Continue (yes/no)? 프롬프트에서 y를 입력하여 계속합니다.**

```
Continue (yes/no)? y
SCSI bus quiesced successfully.
It is now safe to proceed with hotplug operation.
```



핫플러그 작업을 진행하는 동안 SCSI 버스의 I/O 활동이 일시 중단됩니다.

c. 장치를 연결한 다음 전원을 켭니다.

d. Enter y if operation is complete or n to abort (yes/no)? 프롬프트에서 y를 입력합니다.

Enter y if operation is complete or n to abort (yes/no)? y

4 장치가 추가되었는지 확인합니다.

```
# cfgadm -al
Ap_Id                Type          Receptacle  Occupant    Condition
c0                   scsi-bus     connected   configured  unknown
c0::dsk/c0t0d0       disk         connected   configured  unknown
c0::rmt/0            tape         connected   configured  unknown
c1                   scsi-bus     connected   configured  unknown
c1::dsk/c1t3d0       disk         connected   configured  unknown
c1::dsk/c1t4d0       disk         connected   configured  unknown
```

c1 제어기에 새 디스크가 추가되었습니다.

## ▼ SPARC: SCSI 제어기에서 동일한 장치를 교체하는 방법

다음 절차는 SCSI 디스크 c1t4d0을 사용하여 SCSI 제어기에서 동일한 장치를 교체하는 과정의 예를 보여 줍니다.

SCSI 제어기에서 동일한 장치를 교체하려고 시도할 때 다음 조건을 검토하십시오.

- ZFS 루트 풀의 일부인 미러링되거나 미러링되지 않는 부트 장치를 교체하는 경우 다음을 참조하십시오.  
<http://www.oracle.com/technetwork/articles/servers-storage-admin/011-091-sol-dis-recovery-489183.html>
- 장치가 레거시 볼륨 관리 소프트웨어로 관리되는 경우 해당 볼륨 관리 설명서에서 장치 교체를 위한 특정 단계를 참조하십시오. 활성 Oracle 지원 계획을 보유하고 있는 경우 다음을 참조하십시오.
  - SVM(Solaris Volume Manager)을 사용하는 경우 MOS 문서 1010753.1을 참조하십시오.
  - VxVM(Veritas Volume Manager)을 사용하는 경우 MOS 문서 1003122.1, 1011782.1 및 1002285.1을 참조하십시오.

1 관리자로 전환합니다.

2 현재 SCSI 구성을 확인합니다.

```
# cfgadm -al
Ap_Id                Type                Receptacle  Occupant  Condition
c0                   scsi-bus           connected   configured unknown
c0::dsk/c0t0d0      disk                connected   configured unknown
c0::rmt/0            tape                connected   configured unknown
c1                   scsi-bus           connected   configured unknown
c1::dsk/c1t3d0      disk                connected   configured unknown
c1::dsk/c1t4d0      disk                connected   configured unknown
```

3 SCSI 버스에 있는 장치를 같은 유형의 다른 장치로 교체합니다.

a. 다음 **cfgadm** 명령을 입력합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -x replace_device c1::dsk/c1t4d0
Replacing SCSI device: /devices/sbus@1f,0/SUNW,fas@1,8800000/sd@4,0
This operation will suspend activity on SCSI bus: c1
```

b. **Continue (yes/no)?** 프롬프트에서 **y**를 입력하여 계속합니다.

핫플러그 작업을 진행하는 동안 SCSI 버스의 I/O 활동이 일시 중단됩니다.

```
Continue (yes/no)? y
SCSI bus quiesced successfully.
It is now safe to proceed with hotplug operation.
```

c. 제거할 장치의 전원을 끈 후 제거합니다.

d. 교체용 장치를 추가합니다. 그리고 전원을 켭니다.

교체용 장치는 제거한 장치와 같은 유형이고 주소가 같아야 합니다(대상 및 LUN).

e. **Enter y if operation is complete or n to abort (yes/no)?** 프롬프트에서 **y**를 입력합니다.

```
Enter y if operation is complete or n to abort (yes/no)? y
```

4 장치가 교체되었는지 확인합니다.

```
# cfgadm -al
Ap_Id                Type                Receptacle  Occupant  Condition
c0                   scsi-bus           connected   configured unknown
c0::dsk/c0t0d0      disk                connected   configured unknown
c0::rmt/0            tape                connected   configured unknown
c1                   scsi-bus           connected   configured unknown
c1::dsk/c1t3d0      disk                connected   configured unknown
c1::dsk/c1t4d0      disk                connected   configured unknown
```

## ▼ SPARC: SCSI 장치를 제거하는 방법

다음 절차는 SCSI 디스크 c1t4d0을 사용하여 SCSI 제어기에서 장치를 제거하는 과정의 예를 보여 줍니다.

1 관리자로 전환합니다.

2 현재 SCSI 구성을 식별합니다.

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c0             scsi-bus     connected   configured  unknown
c0::dsk/c0t0d0 disk         connected   configured  unknown
c0::rmt/0      tape         connected   configured  unknown
c1             scsi-bus     connected   configured  unknown
c1::dsk/c1t3d0 disk         connected   configured  unknown
c1::dsk/c1t4d0 disk         connected   configured  unknown
```

3 SCSI 장치를 시스템에서 제거합니다.

a. 다음 cfgadm 명령을 입력합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -x remove_device c1::dsk/c1t4d0
Removing SCSI device: /devices/sbus@1f,0/SUNW,fas@1,8800000/sd@4,0
This operation will suspend activity on SCSI bus: c1
```

b. Continue (yes/no)? 프롬프트에서 y를 입력하여 계속합니다.

```
Continue (yes/no)? y
SCSI bus quiesced successfully.
It is now safe to proceed with hotplug operation.
```

핫플러그 작업을 진행하는 동안 SCSI 버스의 I/O 활동이 일시 중단됩니다.

c. 제거할 장치의 전원을 끈 후 제거합니다.

d. Enter y if operation is complete or n to abort (yes/no)? 프롬프트에서 y를 입력합니다.

```
Enter y if operation is complete or n to abort (yes/no)? y
```

주 - SCSI RAID 어레이에서 SCSI RAID 장치를 제거하는 경우 이 단계를 수행해야 합니다.

4 장치가 시스템에서 제거되었는지 확인합니다.

```
# cfgadm -al
Ap_Id          Type          Receptacle  Occupant    Condition
c0             scsi-bus     connected   configured  unknown
c0::dsk/c0t0d0 disk         connected   configured  unknown
c0::rmt/0      tape         connected   configured  unknown
c1             scsi-bus     connected   configured  unknown
c1::dsk/c1t3d0 disk         connected   configured  unknown
```

## SCSI 구성 문제 해결

이 섹션에서는 SCSI 구성 문제 해결과 관련된 오류 메시지와 가능한 해결 방법을 제공합니다. SCSI 구성 문제 해결에 대한 자세한 내용은 [cfgadm\(1M\)](#)을 참조하십시오.

오류 메시지

```
cfgadm: Component system is busy, try again: failed to offline:
  device-path
  Resource          Information
  -----
  /dev/dsk/clt0d0s0  mounted filesystem "/file-system"
```

원인

마운트된 파일 시스템이 있는 장치를 제거 또는 교체하려고 했습니다.

해결 방법

오류 메시지에 나열된 파일 시스템을 마운트 해제하고 `cfgadm` 작업을 다시 시도합니다.

오류 메시지

```
cfgadm: Component system is busy, try again: failed to offline:
  device-path
  Resource          Information
  -----
  /dev/dsk/device-name  swap area
```

원인

시스템 리소스가 아직 활성 상태인 채로 `cfgadm` 명령을 사용하여 교체 장치나 전용 덤프 장치 등의 시스템 리소스를 제거해도 비슷한 오류 메시지가 표시됩니다.

해결 방법

지정된 장치에서 교체 영역을 구성 해제하고 `cfgadm` 작업을 다시 시도합니다.

오류 메시지

```
cfgadm: Component system is busy, try again: failed to offline:
  device-path
  Resource          Information
  -----
  /dev/dsk/device-name  dump device (swap)
```

원인

교체 영역에 구성된 덤프 장치를 제거 또는 교체하려고 했습니다.

해결 방법

교체 영역에 구성된 덤프 장치를 구성 해제하고 `cfgadm` 작업을 다시 시도합니다.

오류 메시지

```
cfgadm: Component system is busy, try again: failed to offline:
  device-path
  Resource          Information
  -----
  /dev/dsk/device-name  dump device (dedicated)
```

원인

전용 덤프 장치를 제거 또는 교체하려고 했습니다.

해결 방법

전용 덤프 장치를 구성 해제하고 cfgadm 작업을 다시 시도합니다.

## ▼ 실패한 SCSI 구성 해제 작업을 해결하는 방법

하나 이상의 대상 장치가 사용 중이고 SCSI 구성 작업이 실패한 경우에는 이 절차를 사용합니다. 이 절차를 수행하지 않으면 앞으로 이 제어기와 대상 장치에서 동적 재구성 작업을 수행할 경우 `dr in progress` 메시지가 표시되며 실패합니다.

- 1 관리자로 전환합니다.
- 2 제어기를 재구성합니다.

```
# cfgadm -c configure device-name
```

## cfgadm 명령을 사용하여 PCI 또는 PCIe 핫 플러그 기능 사용(작업 맵)

다음 작업 맵은 시스템에서 PCI 또는 PCIe 장치를 관리하는 작업에 대해 설명합니다.

작업	설명	수행 방법
PCI 슬롯 구성 정보를 표시합니다.	시스템에 있는 PCI 핫 플러그 가능 장치 및 슬롯의 상태를 표시합니다.	94 페이지 “PCI 슬롯 구성 정보를 표시하는 방법”
PCI 어댑터 카드를 제거합니다.	카드를 구성 해제하고 슬롯에서 전원 연결을 해제한 다음 시스템에서 카드를 제거합니다.	96 페이지 “PCI 어댑터 카드를 제거하는 방법”
PCI 어댑터 카드를 추가합니다.	어댑터 카드를 핫 플러그 가능 슬롯에 삽입하고 슬롯에 전원을 연결한 다음 카드를 구성합니다.	97 페이지 “PCI 어댑터 카드를 추가하는 방법”
PCI 구성 문제를 해결합니다.	오류 메시지와 가능한 PCI 구성 문제 해결 방법을 식별합니다.	99 페이지 “PCI 구성 문제 해결”

## cfgadm 명령을 사용하여 PCI 또는 PCIe 핫 플러그 기능 사용

이 섹션은 SPARC 및 x86 시스템에서 PCI 또는 PCIe 어댑터 카드의 핫 플러그 기능 사용에 대한 단계별 지침을 제공합니다.

cfgadm 명령 외에 prtconf 명령도 핫 플러그 작업에 유용합니다. prtconf 명령은 하드웨어에 대한 추가 구성 정보를 표시합니다.

하드웨어를 추가한 후에 prtconf 명령을 사용하여 하드웨어가 올바르게 구성되었는지 확인할 수 있습니다. 예를 들어, 구성 작업을 수행한 후에 prtconf -D 명령을 사용하면 새로 설치된 하드웨어 장치에 드라이버가 연결되었는지 확인할 수 있습니다. 하드웨어를 구성하기 전에 장치 드라이버를 시스템에 추가하지 않은 경우에는 add\_drv 명령을 사용하여 수동으로 추가할 수 있습니다.

자세한 내용은 prtconf(1M) 및 add\_drv(1M)를 참조하십시오.

이 섹션의 모든 절차를 수행하려면 사용자 계정에는 일반적으로 부여되지 않는 관리 권한이 필요합니다. 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

예에서는 간결한 표시를 위해 PCI 첨부 지점만 나열합니다. 시스템에 표시되는 첨부 지점은 시스템 구성에 따라 달라집니다.

### PCIe LED 표시기 동작

시스템에서 LED 표시기를 관찰하면 슬롯의 핫 플러그 작업 상태를 눈으로 확인할 수 있습니다. PCI Express의 LED 동작은 PCI Express 사양에 정의된 것과 일치하거나 플랫폼에 따라 다를 수도 있습니다.

자세한 내용은 플랫폼 설명서를 참조하십시오. PCI Express의 경우 Attention(주의) 버튼을 누르면 전원 표시기가 깜박이며 상태 전환이 시작되는 것을 알립니다. 상태 전환이 완료되면 깜박임이 끝납니다.

### ▼ PCI 슬롯 구성 정보를 표시하는 방법

이 절차는 PCIe 구성 정보를 포함하도록 업데이트되었습니다.

cfgadm 명령은 시스템에 있는 PCI 핫 플러그 가능 장치 및 슬롯의 상태를 표시합니다. 자세한 내용은 cfgadm(1M)을 참조하십시오.

- 1 관리자로 전환합니다.
- 2 PCI 구성 정보를 표시합니다.
  - PCI 슬롯 구성 정보를 표시합니다.

예를 들면 다음과 같습니다.

```
# cfgadm
Ap_Id           Type           Receptacle    Occupant      Condition
pci1:hpc0_slot0 unknown        empty          unconfigured unknown
pci1:hpc0_slot1 unknown        empty          unconfigured unknown
pci1:hpc0_slot2 unknown        empty          unconfigured unknown
pci1:hpc0_slot3 ethernet/hp    connected     configured    ok
pci1:hpc0_slot4 unknown        empty          unconfigured unknown
```

- 특정 PCI 장치의 정보를 표시합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -s "cols=ap_id:type:info" pci
Ap_Id           Type           Information
pci1:hpc0_slot0 unknown        Slot 7
pci1:hpc0_slot1 unknown        Slot 8
pci1:hpc0_slot2 unknown        Slot 9
pci1:hpc0_slot3 ethernet/hp    Slot 10
pci1:hpc0_slot4 unknown        Slot 11
```

논리적 Ap\_Id인 pci1:hpc0\_slot0은 핫 플러그 가능 슬롯인 Slot 7의 논리적 Ap\_Id입니다. 구성 요소 hpc0은 이 슬롯의 핫 플러그 가능 어댑터를 나타내고 pci1은 PCI 버스 인스턴스를 나타냅니다. Type 필드는 슬롯에 있는 PCI 어댑터의 유형을 나타냅니다.

- PCIe 슬롯 구성 정보를 표시합니다.

예를 들면 다음과 같습니다.

```
# cfgadm pci
Ap_Id           Type           Receptacle    Occupant      Condition
pcie1           unknown        empty          unconfigured unknown
pcie2           unknown        empty          unconfigured unknown
pcie3           unknown        empty          unconfigured unknown
pcie4           etherne/hp    connected     configured    ok
pcie5           pci-pci/hp    connected     configured    ok
pcie6           unknown        disconnected   unconfigured unknown
```

- 특정 PCIe 장치의 정보를 표시합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -s "cols=ap_id:busy:o_state" pci
Ap_Id           Busy           Occupant
pcie1           n              unconfigured
pcie2           n              unconfigured
pcie3           n              unconfigured
pcie4           n              configured
pcie5           n              configured
pcie6           n              configured
```

주 - 대부분의 경우 논리적 Ap\_Id는 시스템 밑판에 실크스크린으로 인쇄된 슬롯 레이블과 일치해야 합니다. 핫 플러그 가능 슬롯의 cfgadm 출력은 플랫폼 설명서를 참조하십시오. 핫 플러그 작업을 시도하기 전에 Ap\_Id가 다른 상태로 전환되지 않도록 Busy 필드가 표시될 수 있습니다.

## ▼ PCI 어댑터 카드를 제거하는 방법

PCIe 어댑터 카드 제거에 대해 다음 절차가 업데이트되었습니다. 그러나 어댑터 카드를 제거하는 절차는 PCI를 사용하는 경우와 PCIe를 사용하는 경우 모두 동일합니다.

- 1 관리자로 전환합니다.
- 2 PCI 어댑터 카드가 있는 슬롯을 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm pci
Ap_Id                Type                Receptacle  Occupant    Condition
pcie1                unknown            empty        unconfigured unknown
pcie2                unknown            empty        unconfigured unknown
pcie3                unknown            empty        unconfigured unknown
pcie4                etherne/hp          connected    configured  ok
pcie5                pci-pci/hp          connected    configured  ok
pcie6                unknown            disconnected  unconfigured unknown
```

- 3 장치를 연 응용 프로그램을 중지합니다.

예를 들어 장치가 이더넷 카드인 경우에는 ipadm 명령을 사용하여 인터페이스를 중지하고 분리합니다. 예를 들면 다음과 같습니다.

```
# ipadm delete-ip bge1
```

- 4 다음과 같이 cfgadm 명령을 사용하여 수동으로 장치를 구성 해제합니다. 또는, PCIe 어댑터 카드를 사용하는 경우 플랫폼 설명서에 정의된 대로 Attention(주의) 버튼을 누르는 등의 자동 구성 방법을 사용합니다.

```
# cfgadm -c unconfigure pcie4
```

- 5 장치가 구성 해제되었는지 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm pci
Ap_Id                Type                Receptacle  Occupant    Condition
pcie1                unknown            empty        unconfigured unknown
pcie2                unknown            empty        unconfigured unknown
pcie3                unknown            empty        unconfigured unknown
pcie4                unknown            connected    unconfigured unknown
pcie5                pci-pci/hp          connected    configured  ok
pcie6                unknown            disconnected  unconfigured unknown
```



주 - 장치가 구성 해제되면 Type 및 Condition 정보도 알 수 없게 됩니다.

- 6 슬롯으로 공급되는 전원의 연결을 수동으로 해제합니다. 자동 구성 방법을 사용하는 경우에는 이 단계를 수행할 필요가 없습니다. 자세한 내용은 플랫폼 설명서를 참조하십시오.

```
# cfgadm -c disconnect pcie4
```

- 7 장치의 연결이 해제되었는지 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm pci
Ap_Id                Type      Receptacle  Occupant    Condition
pcie1                unknown  empty       unconfigured unknown
pcie2                unknown  empty       unconfigured unknown
pcie3                unknown  empty       unconfigured unknown
pcie4                unknown  disconnected unconfigured unknown
pcie5                pci-pci/hp connected   configured  ok
pcie6                unknown  disconnected unconfigured unknown
```

- 8 플랫폼 설명서에 있는 적절한 지침을 수행하여 PCI 어댑터 카드를 제거합니다. 이 카드를 제거하고 나면 콘센트 상태가 비어 있음으로 바뀝니다.

예를 들면 다음과 같습니다.

```
# cfgadm pci
Ap_Id                Type      Receptacle  Occupant    Condition
pcie1                unknown  empty       unconfigured unknown
pcie2                unknown  empty       unconfigured unknown
pcie3                unknown  empty       unconfigured unknown
pcie4                unknown  empty       unconfigured unknown
pcie5                pci-pci/hp connected   configured  ok
pcie6                unknown  disconnected unconfigured unknown
```

주 - 플랫폼 구현에 따라 부트 시에 자동 구성 방법을 사용 또는 사용 안함으로 설정할 수 있습니다. 환경에 맞게 자동 구성 방법을 설정하십시오.

## ▼ PCI 어댑터 카드를 추가하는 방법

PCIe 어댑터 카드 추가에 대해 다음 절차가 업데이트되었습니다. 그러나 어댑터 카드를 추가하는 절차는 PCI를 사용하는 경우와 PCIe를 사용하는 경우 모두 동일합니다.

- 1 관리자로 전환합니다.
- 2 핫 플러그 가능 슬롯 및 열린 래치를 식별합니다.

예를 들면 pcie3과 같습니다.

```
# cfgadm pci
Ap_Id                Type      Receptacle  Occupant    Condition
pcie1                unknown  empty       unconfigured unknown
```

pcie2	unknown	empty	unconfigured	unknown
pcie3	unknown	empty	unconfigured	unknown
pcie4	unknown	empty	unconfigured	unknown
pcie5	pci-pci/hp	connected	configured	ok
pcie6	unknown	disconnected	unconfigured	unknown

3 플랫폼 설명서에 있는 적절한 지침에 따라 PCI 어댑터 카드를 슬롯에 삽입합니다.

4 카드를 삽입한 후 PCI 어댑터 카드가 있는 슬롯을 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm pci
```

Ap_Id	Type	Receptacle	Occupant	Condition
pcie1	unknown	empty	unconfigured	unknown
pcie2	unknown	empty	unconfigured	unknown
pcie3	unknown	disconnected	unconfigured	unknown
pcie4	unknown	empty	unconfigured	unknown
pcie5	pci-pci/hp	connected	configured	ok
pcie6	unknown	disconnected	unconfigured	unknown

5 **cfgadm** 명령을 사용하여 수동으로 슬롯에 전원을 연결합니다. 또는, PCIe 어댑터 카드를 사용하는 경우 플랫폼 설명서에 정의된 대로 **Attention(주의)** 버튼을 누르는 등의 자동 구성 방법을 사용합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -c connect pcie3
```

6 첨부 지점이 연결되어 있는지 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm pci
```

Ap_Id	Type	Receptacle	Occupant	Condition
pcie1	unknown	empty	unconfigured	unknown
pcie2	unknown	empty	unconfigured	unknown
pcie3	unknown	connected	unconfigured	unknown
pcie4	unknown	empty	unconfigured	unknown
pcie5	pci-pci/hp	connected	configured	ok
pcie6	unknown	disconnected	unconfigured	unknown

7 다음과 같이 **cfgadm** 명령을 사용하여 수동으로 PCI 어댑터 카드를 구성합니다. 자동 구성 방법을 사용하는 경우에는 이 단계가 필요 없습니다. 자세한 내용은 플랫폼 설명서를 참조하십시오.

예를 들면 다음과 같습니다.

```
# cfgadm -c configure pcie3
```

8 슬롯에 있는 PCI 어댑터 카드의 구성을 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm pci
```

Ap_Id	Type	Receptacle	Occupant	Condition
pcie1	unknown	empty	unconfigured	unknown
pcie2	unknown	empty	unconfigured	unknown

```
pcie3          etherne/hp  connected  configured unknown
pcie5          pci-pci/hp  connected  configured  ok
pcie6          unknown    disconnected unconfigured unknown
```

9 장치가 새 장치인 경우에는 지원 소프트웨어를 구성합니다.

예를 들어 이 장치가 이더넷 카드인 경우에는 ipadm 명령을 사용하여 인터페이스를 설정합니다. 예를 들면 다음과 같습니다.

```
# ipadm create-addr bge1
```

주 - 플랫폼 구현에 따라 부트 시에 자동 구성 방법을 사용 또는 사용 안함으로 설정할 수 있습니다. 환경에 맞게 자동 구성 방법을 설정하십시오.

## PCI 구성 문제 해결

### 오류 메시지

```
cfgadm: Configuration operation invalid: invalid transition
```

#### 원인

잘못된 전환을 시도했습니다.

#### 해결 방법

cfgadm -c 명령을 적절하게 실행했는지 확인합니다. cfgadm 명령을 사용하여 현재 콘센트 및 점유자 상태를 확인하고 Ap\_Id가 올바른지 확인합니다.

### 오류 메시지

```
cfgadm: Attachment point not found
```

#### 원인

지정한 첨부 지점을 찾을 수 없습니다.

#### 해결 방법

첨부 지점이 올바른지 확인합니다. cfgadm 명령을 사용하여 사용 가능한 첨부 지점 목록을 표시합니다. 물리적 경로도 검사하여 첨부 지점이 아직 있는지 확인합니다.

## cfgadm 명령을 사용하여 SATA 핫 플러그 기능 사용

SATA 제어기 및 포트 멀티플라이어 장치 포트는 장치 트리에서 첨부 지점으로 표현됩니다. 시스템에 연결 및 구성된 SATA 장치는 첨부 지점 이름 확장자로 표시됩니다. 첨부 지점과 SATA 포트라는 말을 같은 의미로 사용할 수 있습니다.

SATA 장치에 사용되는 cfgadm 구문은 SCSI 또는 PCI 장치의 cfgadm 구문과 약간 다릅니다.

다음과 같이 SATA 장치 정보를 표시할 수 있습니다.

```
% cfgadm -al
Ap_Id                                     Type      Receptacle  Occupant    Condition
sata0/0::dsk/c7t0d0                      disk     connected   configured  ok
sata0/1::dsk/c7t1d0                      disk     connected   configured  ok
sata0/2::dsk/c7t2d0                      disk     connected   configured  ok
sata0/3::dsk/c7t3d0                      disk     connected   configured  ok
.
.
.
```

## ▼ SATA 장치를 구성 해제하는 방법

일반적으로 SATA 장치를 제거 및 교체하려면 먼저 구성 해제해야 합니다. 활성 ZFS 저장소 풀에 속한 장치를 구성 해제하려고 하면 다음과 비슷한 오류 메시지가 표시됩니다.

```
# cfgadm -c unconfigure sata5/7
Unconfigure the device at: /devices/pci@2,0/pci1022...
This operation will suspend activity on the SATA device
Continue (yes/no)? y
cfgadm: Hardware specific failure: Failed to unconfig device at ap_id: /devices/pci@2,0/pci10...
```

- 1 관리자로 전환합니다.
- 2 구성을 해제할 장치를 식별합니다.

```
# cfgadm -al | grep c7t0d0
sata0/0::dsk/c7t0d0          disk          connected    configured  ok
```

- 3 장치를 구성 해제합니다.

```
# cfgadm -c unconfigure sata0/0
```

개별 장치를 지정하여 장치를 구성 해제하려고 하면 다음과 비슷한 메시지가 표시됩니다.

```
# cfgadm -c unconfigure sata0/0::dsk/c7t0d0
do_control_ioctl: open failed: errno:2
cfgadm: Library error: Cannot open ap_id: /devices/pci@0,0/pci10...
No such file or directory
```

- 4 장치가 구성 해제되었는지 확인합니다.

```
# cfgadm | grep sata0/0
sata0/0          disk          connected    unconfigured ok
```

## ▼ SATA 장치를 구성하는 방법

디스크를 물리적으로 제거 또는 교체한 후에 구성할 수 있습니다.

- 1 관리자로 전환합니다.

**2 장치 구성합니다.**

```
# cfgadm -c configure sata0/0
```

**3 장치가 구성되었는지 확인합니다.**

```
# cfgadm | grep sata0/0
sata0/0::disk/c7t0d0          disk          connected    configured    ok
```

## RCM(Reconfiguration Coordination Manager) 스크립트 개요

RCM(Reconfiguration Coordination Manager)은 시스템 구성 요소의 동적인 제거를 관리하는 프레임워크입니다. RCM을 사용하면 순차적으로 시스템 리소스를 등록 및 해제할 수 있습니다.

새 RCM 스크립트 기능을 사용하여 응용 프로그램을 종료하는 스크립트를 쓰거나 동적 재구성 중에 응용 프로그램에서 장치를 간단하게 해제할 수 있습니다. 재구성 요청이 스크립트에 의해 등록된 리소스에 영향을 주는 경우 RCM 프레임워크는 요청에 응답하여 자동으로 스크립트를 시작합니다.

리소스를 동적으로 해제하기 전에 수동으로 응용 프로그램에서 리소스를 해제할 수도 있습니다. 또는 `cfgadm` 명령을 `-f` 옵션과 함께 사용하여 재구성 작업을 강제로 실행할 수도 있습니다. 그러나 이 옵션 때문에 응용 프로그램이 알 수 없는 상태가 될 수도 있습니다. 또한 응용 프로그램에서 리소스를 수동으로 해제하면 오류가 발생하는 경우도 많습니다.

RCM 스크립트 기능을 사용하면 동적 재구성 프로세스를 더 간단하게 수행하고 효율적으로 제어할 수 있습니다. RCM 스크립트를 만들면 다음을 수행할 수 있습니다.

- 장치를 동적으로 제거할 경우 장치를 자동으로 해제합니다. 응용 프로그램에서 장치를 연 경우 이 프로세스를 통해 장치가 닫힙니다.
- 시스템에서 장치를 동적으로 제거할 때 사이트별 작업을 실행합니다.

### RCM 스크립트란 무엇입니까?

- RCM 데몬에서 실행하는 실행 가능 셸 스크립트(Perl, sh, csh 또는 ksh) 또는 이진 프로그램입니다. 권장 언어는 Perl입니다.
- 스크립트 파일 소유자의 사용자 ID를 사용하여 자체 주소 공간에서 실행되는 스크립트입니다.
- `cfgadm` 명령을 사용하여 시스템 리소스를 동적으로 재구성할 때 RCM 데몬에서 실행하는 스크립트입니다.

## RCM 스크립트로 어떤 작업을 수행할 수 있습니까?

장치를 동적으로 제거할 때 RCM 스크립트를 사용하여 응용 프로그램에서 장치를 해제할 수 있습니다. 또한 장치가 현재 열려 있는 경우 RCM 스크립트는 장치를 닫습니다.

예를 들어, 테이프 백업 응용 프로그램의 RCM 스크립트가 테이프 드라이브를 닫거나 테이프 백업 응용 프로그램을 종료하도록 테이프 백업 응용 프로그램에 알림을 보낼 수 있습니다.

## RCM 스크립트 프로세스는 어떻게 작동합니까?

다음과 같이 RCM 스크립트를 호출할 수 있습니다.

```
$ script-name command [args ...]
```

RCM 스크립트는 다음과 같은 기본 단계를 수행합니다.

1. 명령줄 인수에서 RCM 명령을 받습니다.
2. 명령을 실행합니다.
3. 결과를 stdout에 이름-값 쌍으로 씁니다.
4. 적절한 종료 상태를 사용하여 종료합니다.

RCM 데몬은 한 번에 하나의 스크립트 인스턴스를 실행합니다. 예를 들어, 스크립트를 실행 중인 경우 RCM 데몬은 첫 번째 스크립트가 종료될 때까지 같은 스크립트를 실행하지 않습니다.

## RCM 스크립트 명령

RCM 스크립트에 다음 RCM 명령을 포함해야 합니다.

- `scriptinfo` - 스크립트 정보를 수집합니다.
- `register` - 관심 리소스를 등록합니다.
- `resourceinfo` - 리소스 정보를 수집합니다.

다음 RCM 명령 중 일부 또는 전부를 포함할 수 있습니다.

- `queryremove` - 리소스를 해제할 수 있는지 여부를 질의합니다.
- `preremove` - 리소스를 해제합니다.
- `postremove` - 리소스 제거 후의 알림을 제거합니다.
- `undoremove` - `preremove`로 수행한 작업을 실행 취소합니다.

이러한 RCM 명령에 대한 전체 설명은 [rcmscript\(4\)](#)를 참조하십시오.

## RCM 스크립트 처리 환경

장치를 동적으로 제거하면 RCM 데몬이 다음을 실행합니다.

- 스크립트의 `register` 명령으로 스크립트에서 식별된 리소스(장치 이름)의 목록을 수집합니다.
- 스크립트의 등록된 리소스가 동적 제거 작업으로 영향을 받는 경우 리소스를 제거하기 전에 스크립트의 `queryremove` 및 `preremove` 명령을 실행합니다.
- 제거 작업이 성공한 경우 스크립트의 `postremove` 명령을 실행합니다. 그러나 제거 작업이 실패하면 RCM 데몬은 스크립트의 `undoremove` 명령을 실행합니다.

## RCM 스크립트 작업

다음 섹션에서는 응용 프로그램 개발자 및 시스템 관리자가 사용하는 RCM 스크립트 작업에 대해 설명합니다.

### 응용 프로그램 개발자 RCM 스크립트(작업 맵)

다음 작업 맵은 RCM 스크립트를 만드는 응용 프로그램 개발자의 작업에 대해 설명합니다.

작업	설명	수행 방법
1. 응용 프로그램이 사용하는 리소스를 식별합니다.	응용 프로그램이 사용하는 리소스(장치 이름) 중에 동적으로 제거될 수 있는 리소스를 식별합니다.	<a href="#">cfgadm(1M)</a>
2. 리소스를 해제하는 명령을 식별합니다.	응용 프로그램에서 리소스를 간단하게 해제하도록 알림을 보내는 명령을 식별합니다.	응용 프로그램 설명서
3. 리소스를 제거한 후에 사용하는 명령을 식별합니다.	응용 프로그램에 리소스 제거에 대해 알리는 명령을 포함합니다.	<a href="#">rcmscript(4)</a>
4. 리소스 제거가 실패할 경우와 관련된 명령을 식별합니다.	응용 프로그램에 사용 가능한 리소스에 대해 알리는 명령을 포함합니다.	<a href="#">rcmscript(4)</a>
5. RCM 스크립트를 씁니다.	작업 1-4에서 식별된 정보를 기반으로 RCM 스크립트를 씁니다.	106 페이지 “테이프 백업 RCM 스크립트의 예”
6. RCM 스크립트를 설치합니다.	스크립트를 적절한 스크립트 디렉토리에 추가합니다.	105 페이지 “RCM 스크립트를 설치하는 방법”
7. RCM 스크립트를 테스트합니다.	수동으로 스크립트 명령을 실행하고 동적 재구성 작업을 시작하여 스크립트를 테스트합니다.	106 페이지 “RCM 스크립트를 테스트하는 방법”

## 시스템 관리자 RCM 스크립트(작업 맵)

다음 작업 맵에서는 RCM 스크립트를 만들어 사이트를 사용자 정의하는 시스템 관리자의 작업에 대해 설명합니다.

작업	설명	수행 방법
1. 동적으로 제거할 리소스를 식별합니다.	<code>cfgadm -l</code> 명령을 사용하여 제거될 가능성이 있는 리소스(장치 이름)를 식별합니다.	<code>cfgadm(1M)</code>
2. 중지할 응용 프로그램을 식별합니다.	응용 프로그램을 간단하게 중지하는 명령을 식별합니다.	응용 프로그램 설명서
3. 리소스를 제거하기 전과 제거 후에 사용하는 명령을 식별합니다.	리소스를 제거하기 전과 제거 후에 수행할 작업을 식별합니다.	<code>rcmscript(4)</code>
4. RCM 스크립트를 씁니다.	작업 1-3에서 식별된 정보를 기반으로 RCM 스크립트를 씁니다.	106 페이지 “테이프 백업 RCM 스크립트의 예”
5. RCM 스크립트를 설치합니다.	스크립트를 적절한 스크립트 디렉토리에 추가합니다.	105 페이지 “RCM 스크립트를 설치하는 방법”
6. RCM 스크립트를 테스트합니다.	수동으로 스크립트 명령을 실행하고 동적 재구성 작업을 시작하여 스크립트를 테스트합니다.	106 페이지 “RCM 스크립트를 테스트하는 방법”

## RCM 스크립트 이름 지정

다음 사항이 적용되는 경우 스크립트의 이름을 *vendor, service*로 지정해야 합니다.

*vendor* 스크립트를 제공하는 공급업체의 스톡 기호 또는 공급업체를 식별하는 이름입니다.

*service* 스크립트가 나타내는 서비스의 이름입니다.

## RCM 스크립트 설치 또는 제거

RCM 스크립트를 설치 또는 제거하려면 관리자여야 합니다. 다음 표를 사용하여 RCM 스크립트를 설치해야 하는 위치를 확인합니다.



표 6-1 RCM 스크립트 디렉토리

디렉토리 위치	스크립트 유형
/etc/rcm/scripts	특정 시스템에 사용되는 스크립트
/usr/platform/'uname -i'/lib/rcm/scripts	특정 하드웨어 구현에 해당되는 스크립트
/usr/platform/'uname -m'/lib/rcm/scripts	특정 하드웨어 클래스에 해당되는 스크립트
/usr/lib/rcm/scripts	모든 하드웨어에 해당되는 스크립트

## ▼ RCM 스크립트를 설치하는 방법

- 1 관리자로 전환합니다.
- 2 적절한 디렉토리에 스크립트를 복사합니다.  
표 6-1을 참조하십시오.  
예를 들면 다음과 같습니다.  

```
# cp ABC,sample.pl /usr/lib/rcm/scripts
```
- 3 스크립트의 사용자 ID와 그룹 ID를 원하는 값으로 변경합니다.  

```
# chown user:group /usr/lib/rcm/scripts/ABC,sample.pl
```
- 4 RCM 데몬에 SIGHUP을 보냅니다.  

```
# pkill -HUP -x -u root rcm_daemon
```

## ▼ RCM 스크립트를 제거하는 방법

- 1 관리자로 전환합니다.
- 2 RCM 스크립트 디렉토리에서 스크립트를 제거합니다.  
예를 들면 다음과 같습니다.  

```
# rm /usr/lib/rcm/scripts/ABC,sample.pl
```
- 3 RCM 데몬에 SIGHUP을 보냅니다.  

```
# pkill -HUP -x -u root rcm_daemon
```

## ▼ RCM 스크립트를 테스트하는 방법

- 1 스크립트를 실행하기 전에 명령줄 셸에서 RCM\_ENV\_FORCE 등의 환경 변수를 설정합니다. 예를 들어, Korn 셸에서는 다음 명령을 사용합니다.

```
$ export RCM_ENV_FORCE=TRUE
```

- 2 명령줄에서 스크립트 명령을 수동으로 실행하여 스크립트를 테스트합니다. 예를 들면 다음과 같습니다.

```
$ script-name scriptinfo
$ script-name register
$ script-name preremove resource-name
$ script-name postremove resource-name
```

- 3 스크립트에 있는 각 RCM 스크립트 명령이 stdout으로 적절한 출력을 인쇄하는지 확인합니다.
- 4 스크립트를 적절한 스크립트 디렉토리에 설치합니다.

자세한 내용은 105 페이지 “RCM 스크립트를 설치하는 방법”을 참조하십시오.

- 5 동적 제거 작업을 실행하여 스크립트를 테스트합니다.

예를 들어, 스크립트가 /dev/dsk/c1t0d0s0 장치를 등록하는 경우를 가정할 수 있습니다. 다음 명령을 사용해 봅니다.

```
$ cfgadm -c unconfigure c1::dsk/c1t0d0
$ cfgadm -f -c unconfigure c1::dsk/c1t0d0
$ cfgadm -c configure c1::dsk/c1t0d0
```



주의 - 이러한 명령은 시스템의 상태를 바꾸고 시스템 오류를 일으킬 수 있으므로 충분히 숙지한 후 사용해야 합니다.

## 테이프 백업 RCM 스크립트의 예

이 예에서는 테이프 백업에 RCM 스크립트를 사용하는 방법을 보여 줍니다.

### 테이프 백업 RCM 스크립트가 수행하는 작업

테이프 백업 RCM 스크립트는 다음 단계를 수행합니다.

1. RCM 명령의 전달 테이블을 설정합니다.
2. 지정한 RCM 명령에 해당되며 구현되지 않은 RCM 명령의 경우 상태 2로 종료되는 전달 루틴을 호출합니다.
3. scriptinfo 섹션을 설정합니다.

```
rcm_script_func_info=Tape backup appl script for DR
```

4. 모든 테이프 드라이브 장치 이름을 `stdout`으로 인쇄하여 시스템에 있는 모든 테이프 드라이브를 등록합니다.

```
rcm_resource_name=/dev/rmt/$f
```

오류가 발생하면 스크립트에서 `stdout`으로 오류 정보를 인쇄합니다.

```
rcm_failure_reason=$errmsg
```

5. 테이프 장치의 리소스 정보를 설정합니다.

```
rcm_resource_usage_info=Backup Tape Unit Number $unit
```

6. 백업 응용 프로그램이 장치를 사용하는지 검사하여 `preremove` 정보를 설정합니다. 백업 응용 프로그램이 장치를 사용하고 있지 않으면 동적 재구성 작업이 계속됩니다. 백업 응용 프로그램이 장치를 사용하고 있는 경우에는 스크립트에서 `RCM_ENV_FORCE`를 검사합니다. `RCM_ENV_FORCE`가 `FALSE`로 설정된 경우 스크립트는 동적 재구성 작업을 거부하고 다음과 같은 메시지를 인쇄합니다.

```
rcm_failure_reason=tape backup in progress pid=...
```

`RCM_ENV_FORCE`가 `TRUE`로 설정되어 있으면 백업 응용 프로그램이 중지되고 재구성 작업이 진행됩니다.

## 테이프 백업 재구성 시나리오의 결과

다음은 RCM 스크립트를 사용하지 않고 `cfgadm` 명령을 사용하여 테이프 장치를 제거한 경우에 나올 수 있는 다양한 결과입니다.

- `cfgadm` 명령을 사용하고 백업 응용 프로그램에서 테이프 장치를 사용하고 있지 않는 경우에는 작업이 성공합니다.
- `cfgadm` 명령을 사용하고 백업 응용 프로그램에서 테이프 장치를 사용하고 있는 경우에는 작업이 실패합니다.

다음은 RCM 스크립트와 `cfgadm` 명령을 사용하여 테이프 장치를 제거한 경우에 나올 수 있는 다양한 결과입니다.

- `cfgadm` 명령을 사용하고 백업 응용 프로그램에서 테이프 장치를 사용하고 있지 않는 경우에는 작업이 성공합니다.
- `cfgadm` 명령을 `-f` 옵션 없이 사용하고 백업 응용 프로그램에서 테이프 장치를 사용하고 있는 경우에는 작업이 실패하고 다음과 비슷한 오류 메시지가 표시됩니다.

```
tape backup in progress pid=...
```

- `cfgadm -f` 명령을 사용하고 백업 응용 프로그램에서 테이프 장치를 사용하고 있는 경우에는 스크립트가 백업 응용 프로그램을 중지하고 `cfgadm` 작업이 성공합니다.

## 예—테이프 백업 RCM 스크립트

```
#!/usr/bin/perl -w
#
# A sample site customization RCM script.
```

```

#
# When RCM_ENV_FORCE is FALSE this script indicates to RCM that it cannot
# release the tape drive when the tape drive is being used for backup.
#
# When RCM_ENV_FORCE is TRUE this script allows DR removing a tape drive
# when the tape drive is being used for backup by killing the tape
# backup application.
#

use strict;

my ($cmd, %dispatch);
$cmd = shift(@ARGV);
# dispatch table for RCM commands
%dispatch = (
    "scriptinfo"    =>    \&do_scriptinfo,
    "register"      =>    \&do_register,
    "resourceinfo" =>    \&do_resourceinfo,
    "queryremove"  =>    \&do_preremove,
    "preremove"    =>    \&do_preremove
);

if (defined(%dispatch{$cmd})) {
    &{%dispatch{$cmd}};
} else {
    exit (2);
}

sub do_scriptinfo
{
    print "rcm_script_version=1\n";
    print "rcm_script_func_info=Tape backup appl script for DR\n";
    exit (0);
}

sub do_register
{
    my ($dir, $f, $errmsg);

    $dir = opendir(RMT, "/dev/rmt");
    if (!$dir) {
        $errmsg = "Unable to open /dev/rmt directory: $!";
        print "rcm_failure_reason=$errmsg\n";
        exit (1);
    }

    while ($f = readdir(RMT)) {
        # ignore hidden files and multiple names for the same device
        if (($f !~ /^\.\/) && ($f =~ /^[0-9]+$/)) {
            print "rcm_resource_name=/dev/rmt/$f\n";
        }
    }

    closedir(RMT);
    exit (0);
}
sub do_resourceinfo

```

```

{
    my ($rsrc, $unit);

    $rsrc = shift(@ARGV);
    if ($rsrc =~ /^\/dev\/rmt\/([0-9]+)$/) {
        $unit = $1;
        print "rcm_resource_usage_info=Backup Tape Unit Number $unit\n";
        exit (0);
    } else {
        print "rcm_failure_reason=Unknown tape device!\n";
        exit (1);
    }
}

sub do_preremove
{
    my ($rsrc);

    $rsrc = shift(@ARGV);

    # check if backup application is using this resource
    #if (the backup application is not running on $rsrc) {
        # allow the DR to continue
    #
        exit (0);
    #}
    #
    # If RCM_ENV_FORCE is FALSE deny the operation.
    # If RCM_ENV_FORCE is TRUE kill the backup application in order
    # to allow the DR operation to proceed
    #
    if ($ENV{RCM_ENV_FORCE} eq 'TRUE') {
        if ($cmd eq 'preremove') {
            # kill the tape backup application
        }
        exit (0);
    } else {
        #
        # indicate that the tape drive can not be released
        # since the device is being used for backup by the
        # tape backup application
        #
        print "rcm_failure_reason=tape backup in progress pid=...\n"
;
        exit (3);
    }
}

```



## USB 장치 사용(개요)

---

이 장에서는 Oracle Solaris OS의 USB(Universal Serial Bus) 장치에 대한 개요를 제공합니다.

이 장에서 다루는 개요 정보는 다음과 같습니다.

- 112 페이지 “USB 장치의 새로운 기능은 무엇입니까?”
- 115 페이지 “USB 장치 개요”
- 120 페이지 “Oracle Solaris OS의 USB 정보”

USB 장치에 대한 최신 정보는 다음 사이트를 참조하십시오.

<http://www.oracle.com/us/sun/index.html>

USB 장치에 대한 일반 정보는 다음 사이트를 참조하십시오.

<http://developers.sun.com/solaris/developer/support/driver/usb.html>

Oracle Solaris OS에서의 USB 장치 사용에 대한 단계별 지침은 8 장, “USB 장치 사용(작업)”을 참조하십시오.

동적 재구성 및 핫 플러그 기능 사용에 대한 자세한 내용은 6 장, “동적으로 장치 구성(작업)”을 참조하십시오.

USB 프린터 구성에 대한 자세한 내용은 **Oracle Solaris 관리: 일반 작업**의 15 장, “CUPS를 사용하여 프린터 설정 및 관리(작업)”를 참조하십시오.

## USB 장치의 새로운 기능은 무엇입니까?

다음 섹션에서는 Oracle Solaris 릴리스의 새로운 USB 기능에 대해 설명합니다.

- 112 페이지 “USB IAD(Interface Association Descriptor) 지원”
- 112 페이지 “EHCI 동시 전송 지원”
- 112 페이지 “USB 장치 Hotpluggable 동작”
- 113 페이지 “x86: GRUB 기반 부트 시 USB CD 및 DVD 지원”
- 113 페이지 “USB 가상 키보드 및 마우스 지원”

### USB IAD(Interface Association Descriptor) 지원

**Oracle Solaris 11:** IAD(Interface Association Descriptor) 지원을 위해 USB 장치 노드 유형 IA 노드가 만들어집니다. 이 기능은 웹캠의 비디오 및 오디오 인터페이스처럼 드라이버가 같은 장치에 대해 여러 인터페이스를 지원할 수도 있는 것을 의미합니다. IA 노드에서 발견되는 드라이버가 없는 경우에는 연결 드라이버 `usb_ia`가 IA 노드에 바인딩되어 인터페이스 노드를 만듭니다. 자세한 내용은 `usb_ia(7D)`를 참조하십시오.

### EHCI 동시 전송 지원

**Oracle Solaris 11:** USB EHCI 호스트 제어기 드라이버는 USB 2.0 이상의 고속 동시 장치에 대해 동시 전송 지원을 제공합니다. 자세한 내용은 `usb_isoc_request(9S)`를 참조하십시오.

### USB 장치 Hotpluggable 동작

**Oracle Solaris 11:** 이 Solaris 릴리스에는 시스템을 재부트하지 않고 연결 또는 연결 해제하거나 사용자 개입 없이 자동으로 구성 또는 구성 해제할 수 있는 장치를 식별하는 새로운 *hotpluggable* 장치 속성이 추가되었습니다. 모든 USB 및 1394 장치를 핫 플러그 가능 장치로 식별하여 129 페이지 “USB 대용량 저장 장치 사용”에 설명된 이점을 얻을 수 있습니다. 또한 제거할 수 없는 매체 USB 및 1394 장치의 경우 더 이상 이동식 매체로 식별되지 않고 더 이상 *removable-media* 속성이 지정되지 않습니다.

변경 사항은 제거할 수 없는 매체 USB 및 1394 지원을 향상하고 장치 성능을 개선하기 위해 주로 커널 레벨에서 적용되었습니다. 그러나 이러한 변경 사항은 장치 사용에 영향을 주지 않습니다. 예를 들어, 이러한 장치를 마운트 또는 마운트 해제하는 작업은 `rmvolmgr`에서 제어합니다. 사용자의 관점에서 눈에 띄는 변경 사항은 장치의 *hotpluggable* 및 *removable-media* 속성뿐입니다.



## USB 및 1394(FireWire) 지원

제거할 수 없는 USB 저장 장치와 1394 대용량 저장 장치는 드라이버 레벨에서 핫 플러그 가능 장치로 식별됩니다. 이 동작은 시스템을 재부트하지 않고도 이러한 장치를 연결 또는 연결 해제할 수 있고 사용자 개입 없이 구성 또는 구성 해제할 수 있는 것을 의미합니다. 이러한 변경 사항은 커널 레벨에서 적용되며 장치의 사용에 영향을 주지 않습니다. 예를 들어 이러한 장치의 마운트 및 마운트 해제는 이동식 매체 관리 서비스에서 제어합니다.

또한 `format` 유틸리티를 사용하여 제거할 수 없는 USB 장치 및 1394 대용량 저장 장치에 액세스하고 레이블을 지정할 수 있습니다. 그러나 `/etc/driver/drv/scsa2usb.conf` 파일에서 `remvalue`를 `true`로 설정하면 이러한 장치의 새로운 핫 플러그 가능 동작을 대체할 수 있습니다. 이 매개변수를 `true`로 설정하는 것은 기본 설정과 다른 경우에도 드라이버 레벨에서 장치를 이동식 매체 장치로 취급하는 것을 의미합니다.

이러한 장치의 사용에 대한 자세한 내용은 [scsa1394\(7D\)](#) 및 [128 페이지 “USB 대용량 저장 장치 사용\(작업 맵\)”](#)을 참조하십시오.

## x86: GRUB 기반 부트 시 USB CD 및 DVD 지원

**Oracle Solaris 11:** GRUB 기반 부트 환경에서 다음과 같은 USB 기능을 사용할 수 있습니다.

- USB CD 또는 DVD 드라이브에서 설치
- USB 저장 장치에서 부트합니다. USB 드라이브에서 부트하려면 먼저 USB 드라이브에 Solaris 릴리스를 설치해야 합니다.

부트에 대한 자세한 내용은 [SPARC 플랫폼에서 Oracle Solaris 부트 및 종료](#) 또는 [x86 플랫폼에서 Oracle Solaris 부트 및 종료](#)를 참조하십시오.

## USB 가상 키보드 및 마우스 지원

**Oracle Solaris 11:** USB 가상 키보드 및 마우스 지원을 사용하면 여러 키보드와 여러 마우스를 연결하여 키보드 또는 마우스 세트가 하나의 가상 키보드 또는 마우스로 작동하게 만들 수 있습니다. 그러면 각 물리적 장치의 입력이 단일 입력 스트림으로 통합됩니다. 예를 들어 한 키보드에서 Shift를 누르고 다른 키보드에서 A를 누르면 대문자 A가 표시됩니다.

또한 USB 키보드 또는 마우스를 랩탑에 연결하여 랩탑의 PS/2 키보드 및 패드와 한 장치로 작동하게 만드는 기능도 지원됩니다.

그리고 가상 키보드 및 마우스 기능을 통해 바코드 판독기 기능도 지원됩니다.

자세한 내용은 [virtualkm\(7D\)](#)을 참조하십시오.

## USB 장치용 Oracle Solaris 지원

따로 언급되지 않은 경우 SPARC 및 x86 기반 시스템 모두에서 다음과 같은 USB 1.1 및 USB 2.0 장치가 지원됩니다.

- 오디오 장치 - USB 2.0 허브에서 USB 1.x 장치만 지원됩니다.
  - USB 2.0 오디오 장치는 지원되지 않습니다.
  - USB 2.0 포트의 USB 2.0 허브에 연결된 USB 1.x 오디오 장치는 Oracle Solaris 11 릴리스에서만 사용할 수 있습니다. 자세한 내용은 `usb_ac(7D)` 및 `usb_as(7D)`를 참조하십시오.
  - USB 드라이버에서 지원되지 않는 장치에는 `gphoto2`, `gtkam` 및 `pilotlink` 등의 `libusb` 응용 프로그램이 있을 수 있습니다. 자세한 내용은 `/usr/sfw/share/doc/libusb/libusb.txt`를 참조하십시오.
- 일반 USB 드라이버(`ugen(7D)`)
- HID 장치(키보드 및 마우스 장치, `hid(7D)`)
- 허브(`hubd(7D)`)
- 프린터
- 직렬 장치(Edgeport(`usbser_edge(7D)`), Prolific(`usbprl(7D)`), Keyspan(`usbksp(7D)`))
- 저장 장치(`scca2usb(7D)`)
- 사용자 공간 USB 장치 관리 라이브러리(`libusb(3LIB)`)

다음 표를 사용하여 특정 USB 1.1 및 USB 2.0 장치에 대한 Solaris 지원 정보를 확인할 수 있습니다.

### 참고:

- USB 1.x 오디오 장치만 지원됩니다. USB 2.0 오디오 장치는 지원되지 않습니다.
- USB 2.0 포트의 USB 2.0 허브에 연결된 USB 1.x 오디오 장치는 사용할 수 있습니다. 자세한 내용은 `usb_ac(7D)` 및 `usb_as(7D)`를 참조하십시오.
- USB 드라이버에서 지원되지 않는 장치에는 `gphoto2`, `gtkam` 및 `pilotlink` 등의 `libusb` 응용 프로그램이 있을 수 있습니다. 자세한 내용은 `/usr/sfw/share/doc/libusb/libusb.txt`를 참조하십시오.

대용량 저장 장치와 관련된 작업 정보는 8 장, “USB 장치 사용(작업)”을 참조하십시오.

`ugen`에 대한 자세한 내용은 117 페이지 “USB 드라이버 향상”을 참조하십시오.

## USB 장치 개요

USB(Universal Serial Bus)는 PC 업계에서 키보드, 마우스 및 프린터 등의 주변 장치를 시스템에 연결할 수 있는 저가 솔루션을 제공하기 위해 개발했습니다.

USB 커넥터는 한 가지 유형의 케이블에 한 방향으로만 맞도록 디자인되었습니다. USB의 주된 디자인 동기는 장치 종류에 따라 여러 가지 커넥터 유형을 사용해야 할 필요성을 줄이자는 것이었습니다. 이 디자인을 사용하면 시스템 뒤쪽 패널을 덜 복잡하게 만들 수 있습니다.

장치는 외부 USB 허브 또는 컴퓨터 본체에 있는 루트 허브의 USB 포트에 연결됩니다. 허브에는 여러 포트가 있기 때문에 허브에서 여러 개의 장치 트리를 뺀어 연결할 수 있습니다.

자세한 내용은 [usb\(7D\)](http://www.usb.org/7D) 또는 다음 사이트를 참조하십시오.

<http://www.usb.org/home>

## 일반적으로 사용되는 USB 머리글자어

다음 표는 Oracle Solaris OS에서 사용되는 USB 머리글자어에 대해 설명합니다. USB 구성 요소 및 머리글자어에 대한 전체 설명은 다음 위치를 참조하십시오.

<http://www.usb.org/home>

머리글자어	정의	자세한 정보
UGEN	USB 일반 드라이버	<a href="#">ugen(7D)</a>
USB	Universal Serial Bus	<a href="#">usb(7D)</a>
USBA	Universal Serial Bus 아키텍처(Solaris)	<a href="#">usba(7D)</a>
USBAI	USBA 클라이언트 드라이버 인터페이스(Solaris)	해당 없음
HCD	USB 호스트 제어기 드라이버	해당 없음
EHCI	확장 호스트 제어기 인터페이스	<a href="#">ehci(7D)</a>
OHCI	개방형 호스트 제어기 인터페이스	<a href="#">ohci(7D)</a>
UHCI	범용 호스트 제어기 인터페이스	<a href="#">uhci(7D)</a>

## USB 버스 설명

USB 사양은 로열티 없이 누구나 사용할 수 있습니다. 이 사양은 버스와 커넥터의 전자 및 기계 인터페이스를 정의합니다.

USB는 허브가 USB 장치의 첨부 지점을 제공하는 토폴로지를 사용합니다. 호스트 제어기에는 시스템에 있는 모든 USB 포트의 근원인 루트 허브가 있습니다. 허브에 대한 자세한 내용은 123 페이지 “USB 호스트 제어기 및 허브”를 참조하십시오.

그림 7-1 USB 물리적 장치 계층

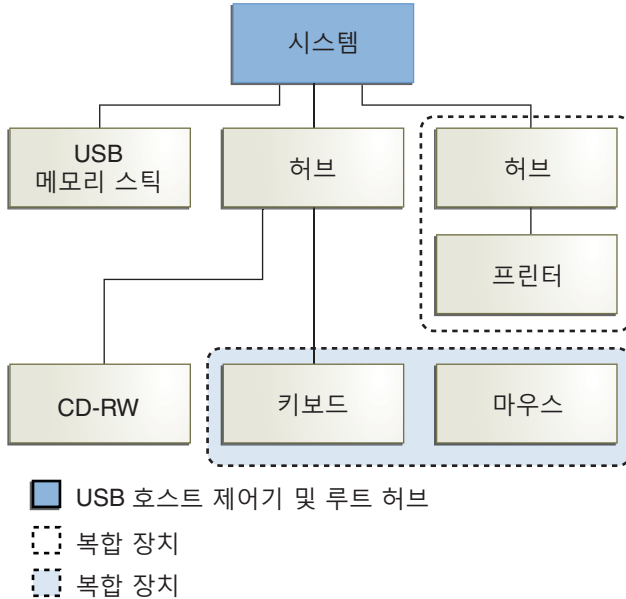


그림 7-1에는 세 개의 활성 USB 포트가 있는 시스템이 표시되어 있습니다. 첫 번째 USB 포트는 USB 메모리 스틱에 연결됩니다. 두 번째 USB 포트는 외부 허브에 연결되고 외부 허브는 다시 cdrw 장치와 조합 키보드/마우스 장치에 연결됩니다. 조합 장치인 이 키보드에는 연결된 마우스와 키보드 모두를 작동하는 USB 제어기가 있습니다. 키보드와 마우스는 같은 USB 제어기로 제어되기 때문에 공통의 USB 버스 주소를 공유합니다.

그림 7-1에는 복합 장치인 허브와 프린터의 예도 표시되어 있습니다. 허브는 프린터와 같은 케이스에 들어 있는 외부 허브입니다. 프린터는 허브에 영구적으로 연결되어 있습니다. 허브와 프린터는 별도의 USB 버스 주소를 사용합니다.

다음은 그림 7-1에 표시된 일부 장치의 장치 트리 경로 이름입니다.

```

메모리 스틱    /pci@1f,4000/usb@5/storage@1
키보드        /pci@1f,4000/usb@5/hub@2/device@1/keyboard@0
마우스       /pci@1f,4000/usb@5/hub@2/device@1/mouse@1
cdrw 장치    /pci@1f,4000/usb@5/hub@2/storage@3
    
```

프린터                    /pci@1f,4000/usb@5/hub@3/printer@1

## USB 장치 및 드라이버

속성과 서비스가 비슷한 USB 장치는 장치 클래스로 그룹화됩니다. 각 장치 클래스에는 해당되는 드라이버가 있습니다. 클래스에 있는 장치는 같은 장치 드라이버 쌍으로 관리됩니다. 그러나 USB 사양에서는 특정 클래스에 속하지 않은 공급업체 특정 장치도 허용됩니다.

HID(Human Interface Device) 클래스에는 다음과 같이 사용자가 제어하는 장치가 포함됩니다.

- 키보드
- 마우스 장치
- 조이스틱

통신 장치 클래스에는 다음 장치가 포함됩니다.

- 모뎀
- 이더넷 어댑터

기타 장치 클래스에는 다음 클래스가 포함됩니다.

- 오디오
- 모니터
- 프린터
- 저장 장치

각 USB 장치에는 장치의 클래스를 반영하는 설명자가 포함됩니다. 장치 클래스는 구성 및 데이터 전송 시에 구성된 장치가 작동하는 방식을 지정합니다. 자세한 클래스 정보는 다음을 참조하십시오.

<http://www.usb.org/home>

Oracle Solaris 릴리스에서 지원되는 USB 장치에 대한 자세한 내용은 [usb\(7D\)](#)를 참조하십시오.

## USB 드라이버 향상

다음과 같은 USB 드라이버 향상된 기능이 포함되어 있습니다.

- **USB CDC ACM 장치 지원** - acm 드라이버는 USB 통신 클래스 장치 사양의 추상 제어 모델을 준수하는 장치 및 모뎀 기능이 있는 일부 PCMCIA 장치와 함께 사용할 수 있습니다.

pppd 데몬은 /dev/term/[0-9]\* 항목을 통해 이러한 장치에 액세스할 수 있습니다. 자세한 내용은 [pppd\(1M\)](#)를 참조하십시오.

자세한 내용은 [usbsacm\(7D\)](#)을 참조하십시오.

- **일반 USB 드라이버** - 특별한 커널 드라이버를 작성하지 않고도 표준 UNIX read(2) 및 write(2) 시스템 호출을 사용하는 응용 프로그램에서 USB 장치를 액세스하고 조작할 수 있습니다. 다음과 같은 추가 기능이 있습니다.
  - 응용 프로그램이 원시 장치 데이터 및 장치 상태에 액세스할 수 있습니다.
  - 드라이버가 제어, 벌크 전송 및 인터럽트(송수신) 전송을 지원합니다.

ugen 드라이버를 더 이상 장치에 명시적으로 바인딩할 필요가 없습니다. 기본적으로 클래스 드라이버가 없는 장치에는 usb\_mid가 바인딩되고 libusb에서 작동하는 ugen 인터페이스를 내보냅니다. 예를 들어 대용량 저장 장치가 아닌 USB 카메라를 연결하고 libusb 응용 프로그램을 사용하여 액세스할 수 있습니다. 또한 scsa2usb 및 usbprn 드라이버는 ugen 인터페이스를 내보내며, 이러한 장치 클래스에서 직접 libusb 응용 프로그램을 사용할 수 있습니다.

자세한 내용은 [ugen\(7D\)](#)을 참조하십시오.

- **USB 직렬 드라이버 지원**
  - **Digi Edgeport USB 지원** - Edgeport USB 드라이버는 Edgeport 장치에서만 작동하며 다른 USB 직렬 장치에서 작동하지 않습니다.
    - 새 장치는 /dev/term/[0-9]\* 및 /dev/cua/[0-9]\*로 액세스됩니다.
    - USB 직렬 포트는 로컬 직렬 콘솔의 역할을 할 수 없다는 점을 제외하면 다른 직렬 포트와 동일하게 사용할 수 있습니다. 데이터가 USB 포트를 통과한다는 것은 사용자에게 알려진 사실입니다.

자세한 내용은 [usbser\\_edge\(7D\)](#) 또는 다음 사이트를 참조하십시오.

- <http://www.digi.com>
- <http://www.oracle.com/us/sun/index.html>

- **Keyspan** - Keyspan USB 직렬 드라이버는 USA-19HS 및 USA-49WLC 모델을 지원하는 Keyspan 장치에서만 작동합니다.

자세한 내용은 [usbkskp\(7D\)](#)를 참조하십시오.

- **Prolific** - Prolific USB 직렬 드라이버는 PL2303 칩셋을 기반으로 한 장치에서만 작동합니다.

자세한 내용은 [usbprl\(7D\)](#)을 참조하십시오.

USB 직렬 장치 지원에 대한 자세한 내용은 다음 사이트를 참조하십시오.

<http://www.oracle.com/us/sun/index.html>

- **사용자가 작성한 커널 및 userland 드라이버의 설명서 및 이진 지원** - USB 드라이버 개발에 대한 최신 정보는 다음을 참조하십시오.
  - <http://www.oracle.com/us/sun/index.html>
  - **Writing Device Drivers**의 20 장, "USB Drivers"
  - **Writing Device Drivers**의 부록 C, "Making a Device Driver 64-Bit Ready"
  - **Device Driver Tutorial**

- [Intro\(7\)](#), [Intro\(9F\)](#) 및 [Intro\(9S\)](#)

## EHCI, OHCI 및 UHCI 드라이버

EHCI 드라이버의 기능에는 다음이 포함됩니다.

- USB 2.0을 지원하는 향상된 호스트 컨트롤러 인터페이스와 호환
- 고속 제어, 벌크, 인터럽트 및 동시 전송 지원
- USB 2.0 칩에는 하나의 EHCI 제어기와 하나 이상의 OHCI 또는 UHCI 제어기가 있습니다.
- USB 1.1 장치를 연결하면 OHCI 또는 UHCI 제어기에 동적으로 할당됩니다. USB 2.0 장치를 연결하면 EHCI 제어기에 동적으로 할당됩니다.

시스템이 USB 1.1 또는 USB 2.0 장치를 지원하는지 여부를 확인하려면 `prtconf` 명령 출력을 사용합니다. 예를 들면 다음과 같습니다.

```
# prtconf -D | egrep "ehci|ohci|uhci"
```

`prtconf` 출력으로 EHCI 제어기가 식별되는 경우에는 시스템에서 USB 2.0 장치를 지원합니다.

`prtconf` 출력으로 OHCI 또는 UHCI 제어기가 식별되는 경우에는 시스템에서 USB 1.1 장치를 지원합니다.

## Oracle Solaris USBA(USB 아키텍처)

USB 장치는 두 개의 장치 트리 노드 레벨로 표현할 수 있습니다. 장치 노드는 전체 USB 장치를 나타냅니다. 하나 이상의 자식 인터페이스 노드는 장치에 있는 개별 USB 인터페이스를 나타냅니다.

드라이버 바인딩에는 호환되는 이름 등록 정보를 사용합니다. 자세한 내용은 IEEE 1275 USB 바인딩의 3.2.2.1과 [Writing Device Drivers](#)를 참조하십시오. 드라이버는 전체 장치에 바인딩되어 모든 인터페이스를 제어할 수도 있고 한 인터페이스에만 바인딩될 수도 있습니다. 전체 장치를 차지하는 공급업체 또는 클래스 드라이버가 없으면 일반 USB 다중 인터페이스 드라이버가 장치 레벨 노드에 바인딩됩니다. 이 드라이버는 IEEE 1275 바인딩 사양의 3.3.2.1절에 정의된 것과 같이 호환되는 이름 등록 정보를 사용하여 각 인터페이스에 드라이버를 바인딩하려 합니다.

Oracle Solaris USBA(USB 아키텍처)는 USB 1.1 및 USB 2.0 사양을 준수하며 Oracle Solaris DDI(장치 드라이버 인터페이스)의 일부입니다. USBA 모델은 Oracle SCSCA(Solaris 일반 SCSI 아키텍처)와 비슷합니다. 다음 그림과 같이 USBA는 클라이언트에 일반 USB 전송 계층 추상화를 제공하여 핵심적인 일반 USB 기능을 구현하는 서비스를 제공하는, 서버의 시스템 기능에 의존하는 계층입니다.

그림 7-2 Oracle Solaris USBA(USB 아키텍처)



## Oracle Solaris OS의 USB 정보

이 섹션에서는 Oracle Solaris OS의 USB에 대해 알아야 할 사항을 설명합니다.

### USB 2.0 기능

다음과 같은 USB 2.0 기능이 포함되어 있습니다.

- **더 나은 성능** - USB 2.0 제어기에 연결된 장치의 데이터 처리량이 USB 1.1 장치보다 최대 40배까지 늘었습니다.  
DVD 및 하드 디스크와 같은 고속 USB 장치에 액세스할 때 고속 USB 프로토콜의 이점을 이용할 수 있습니다.
- **역방향 호환성** - 1.0 및 1.1 장치와 호환되므로 동일한 케이블, 커넥터 및 소프트웨어 인터페이스를 사용할 수 있습니다.

USB 장치에 대한 설명과 용어는 [115 페이지 “USB 장치 개요”](#)를 참조하십시오.

### USB 2.0 장치 기능 및 호환성 문제

USB 2.0 장치는 USB 2.0 사양을 따르는 고속 장치로 정의됩니다. USB 2.0 사양은 <http://www.usb.org/home>에서 참조할 수 있습니다.

USB 장치의 속도를 식별하려면 `/var/adm/messages` 파일에서 다음과 비슷한 메시지를 확인하십시오.

```
Dec 13 17:05:57 mysystem usba: [ID 912658 kern.info] USB 2.0 device
(usb50d,249) operating at hi speed (USB 2.x) on USB 2.0 external hub:
storage@4, scsa2usb0 at bus address 4
```

다음은 이 Oracle Solaris 릴리스에서 지원되는 몇 가지 USB 장치입니다.



- CD-RW, 하드 디스크, DVD, 디지털 카메라, 디스켓, 테이프 드라이브, 메모리 스틱 및 다중 형식 카드 판독기와 같은 대용량 저장 장치
- 키보드 및 마우스 장치
- 스피커 및 마이크 등의 오디오 장치

Oracle Solaris 릴리스에서 확인된 USB 장치의 전체 목록을 보려면 다음을 참조하십시오.

<http://www.oracle.com/us/sun/index.html>

추가 저장 장치는 `scsa2usb.conf` 파일을 수정하여 작동할 수 있습니다. 자세한 내용은 [scsa2usb\(7D\)](#)를 참조하십시오.

Oracle Solaris USB 2.0 장치 지원에는 다음 기능이 포함됩니다.

- 12MB/초에서 480MB/초로 향상된 USB 버스 속도. 따라서 USB 2.0 사양을 지원하는 장치를 USB 2.0 포트에 연결하면 해당 USB 1.1 장치보다 훨씬 빠르게 작동합니다.

USB 2.0 포트는 다음 중 한 가지로 사용할 수 있습니다.

- USB 2.0 PCI 카드의 포트
- USB 2.0 포트에 연결된 USB 2.0 허브의 포트
- SPARC 또는 x86 컴퓨터 주보드에 있는 포트

오래된 SPARC 플랫폼의 경우에는 USB 2.0 PCI 카드가 필요할 수도 있습니다.

- Oracle Solaris 릴리스에 대해 검증된 USB 2.0 PCI 카드 목록은 다음을 참조하십시오.

<http://www.oracle.com/us/sun/index.html>

- 동일한 시스템에 USB 1.1과 USB 2.0 장치가 모두 있는 경우에도 USB 1.1 장치는 이전과 마찬가지로 작동합니다.
- USB 2.0 장치는 USB 1.x 포트에서 작동하기는 하지만 USB 2.0 포트에 연결했을 때 더 뛰어난 성능을 보입니다.
- USB 2.0 호스트 제어기에는 하나의 고속 EHCI(Enhanced Host Controller Interface)와 하나 이상의 OHCI(OpenHCI Host Controller Interface) 또는 UHCI(Universal Host Controller Interface) 내장 제어기가 있습니다. USB 2.0 포트에 연결된 장치는 USB 2.0의 지원 여부에 따라 EHCI 또는 OHCI 컨트롤러 중 하나에 동적으로 지정됩니다.

주 - USB 2.0 PCI 카드의 포트에 연결된 USB 2.0 저장 장치와 동일한 하드웨어 구성에서 이전 Solaris 릴리스에서 사용되었던 장치는 이 릴리스로 업그레이드 한 후 장치 이름을 변경할 수 있습니다. 이들 장치는 USB 2.0 장치로 인식되며 EHCI 제어가 담당하게 되므로 이러한 변경이 발생합니다. `/dev/[r]dsk/c wtxdysz`의 제어기 번호 `w`가 이러한 장치에 맞게 변경되었습니다.

USB 장치의 속도는 상위 포트에서 지원할 수 있는 속도로도 제한된다는 점에 유의하십시오. 예를 들어, USB 2.0 외부 허브에 USB 1.x 허브 및 USB 2.0 장치가 연결되어 있는 경우 USB 2.0 외부 허브에 연결된 장치는 고속이 아닌 표준 속도로 실행됩니다.

USB 2.0 장치 지원에 대한 자세한 내용은 `ehci(7D)` 및 `usba(7D)`를 참조하십시오.

## 버스 전원 사용 장치

버스 전원 사용 허브는 연결된 USB 버스의 전원을 사용하여 연결된 장치에 전원을 공급합니다. 이러한 허브에서 연결된 장치에 제공하는 전원이 제한되어 있으므로 이러한 허브가 과부하가 걸리지 않도록 주의해야 합니다.

USB 장치에 대해 전원 배정이 구현됩니다. 이 기능에는 다음과 같은 제한이 있습니다.

- 두 개의 버스 전원 사용 허브를 중첩해서 사용할 수 없습니다.
- 각각의 버스 전원 사용 허브에서는 각 포트에 대해 최대 100mA의 전원만 허용됩니다.
- 버스 전원 사용 허브에는 자체 전원이 공급되는 장치나 버스 전원을 적게 사용하는 장치만 연결할 수 있습니다. 버스 전원을 많이 사용하는 장치는 연결이 거부됩니다. 일부 허브 또는 장치에서 전원을 잘못 보고할 수도 있으므로 연결 결과를 보장하기는 어렵습니다.

## USB 키보드 및 마우스 장치

USB 키보드 및 마우스 장치를 사용할 때에는 다음 문제에 주의하십시오.

- 재부트하는 동안이나 SPARC 시스템에서 ok 프롬프트가 표시되어 있을 때 키보드와 마우스를 이동하지 마십시오. 시스템 재부트가 끝난 후에는 언제든지 키보드와 마우스를 다른 허브로 이동할 수 있습니다. 키보드와 마우스를 연결하고 나면 다시 제대로 작동합니다.
- 일부 타사 USB 키보드에서는 키패드의 왼쪽에 있는 키가 작동하지 않을 수도 있습니다.
- SPARC - SPARC 시스템에서 USB 키보드 및 마우스 장치를 사용할 때에는 다음 문제에 주의하십시오.

- USB 키보드에 있는 전원 키는 type-5 키보드에 있는 전원 키와 다른 방식으로 작동합니다. USB 키보드에서는 SUSPEND/SHUTDOWN 키를 사용하여 시스템을 일시 중단하거나 종료할 수 있습니다. 그러나 이 키를 사용하여 시스템 전원을 켤 수는 없습니다.
- 부트 프로세스가 완료되기 전에는 OBP(OpenBoot PROM)에서 키보드 및 마우스 장치가 주보드 루트 허브 포트에만 제한됩니다.
- 기존 SPARC 시스템에서는 Type 3, 4 또는 5 키보드와 USB 키보드 및 마우스 장치를 동시에 사용할 수 없습니다.
- 다중 키보드 및 마우스 장치 지원에 대한 자세한 내용은 [virtualkm\(7D\)](#)을 참조하십시오.

## USB 휠 마우스 지원

다음과 같은 휠 마우스 기능이 지원됩니다.

- USB 또는 PS/2 마우스 장치에서 버튼을 세 개 이상 사용할 수 있습니다.
- USB 또는 PS/2 마우스 장치에서 휠 마우스 스크롤을 사용할 수 있습니다. 따라서 USB 또는 PS/2 마우스의 휠을 돌리면 마우스가 가리키는 응용 프로그램 또는 창에서 스크롤할 수 있습니다. StarOffice, Firefox 및 GNOME 응용 프로그램은 휠 마우스 스크롤을 지원합니다. 그러나 다른 응용 프로그램은 휠 마우스 스크롤을 지원하지 않을 수 있습니다.

## USB 호스트 제어기 및 허브

USB 허브는 다음 기능을 담당합니다.

- 포트의 장치 삽입 또는 제거 모니터링
- 포트에 있는 개별 장치의 전원 관리
- 포트에 공급되는 전원 제어

USB 호스트 제어기에는 **루트 허브**라는 내장 허브가 있습니다. 시스템의 뒤쪽 패널에 보이는 포트는 루트 허브의 포트입니다. USB 호스트 제어기는 다음을 담당합니다.

- USB 버스 감지. 개별 장치는 버스에 대한 중재를 수행할 수 없습니다.
- 장치에서 결정된 폴링 간격을 사용하여 장치 폴링. 장치는 폴링 사이의 시간을 고려하기에 충분한 버퍼가 있는 것으로 가정합니다.
- USB 제어기와 연결된 장치 사이에서 데이터 보내기. P2P 통신은 지원되지 않습니다.

## USB 허브 장치

- SPARC 기반 시스템이나 x86 기반 시스템에서 4개를 넘는 수준으로 허브를 중첩하지 마십시오. SPARC 시스템의 OpenBoot PROM은 4개를 넘는 수준의 장치를 안정적으로 탐지할 수 없습니다.
- 버스 전원 사용 허브를 다른 버스 전원 사용 허브에 중첩하여 연결하지 마십시오. 버스 전원 사용 허브에는 자체 전원 공급 장치가 없습니다.
- 전원이 많이 필요한 장치를 버스 전원 사용 허브에 연결하지 마십시오. 이러한 장치는 버스 전원 사용 허브에서 연결이 거부되거나 허브에서 다른 장치에 공급할 전원을 소진할 수 있습니다. 그런 장치의 예에는 USB 디스켓 장치가 있습니다.

## SPARC: USB 전원 관리

SPARC 시스템에서는 USB 장치의 일시 중단 및 재개가 완전히 지원됩니다. 그러나 사용 중인 장치를 일시 중단하거나 일시 중단 종료 상태로 전원이 꺼진 시스템에서 장치를 제거하지 마십시오.

USB 프레임워크는 전원 관리 기능을 사용하는 SPARC 기반 시스템에서 모든 장치의 전원을 관리하기 위해 최대한으로 노력합니다. USB 장치의 전원 관리란 장치가 연결된 포트를 허브 드라이버가 일시 중단하는 것을 의미합니다. **원격 웨이크업**을 지원하는 장치는 시스템에서 장치 경로에 있는 모든 요소를 깨우도록 하여 장치를 사용할 수 있습니다. 응용 프로그램이 장치로 I/O를 보내는 경우 호스트 시스템에서 장치를 깨울 수도 있습니다.

모든 HID 장치(키보드, 마우스, 허브 및 저장 장치), 허브 장치 및 저장 장치는 원격 웨이크업 기능을 지원하는 경우 기본적으로 전원 관리가 사용됩니다. USB 프린터는 두 인쇄 작업 사이에만 전원 관리를 사용합니다. UGEN(일반 USB 드라이버)에서 관리하는 장치는 닫힌 경우에만 전원 관리를 사용합니다.

전원 소비를 줄이기 위해 전원 관리를 실행하는 경우에는 가장 먼저 USB 최하위 장치 전원이 꺼집니다. 허브 포트에 연결된 모든 장치의 전원이 꺼지고 나면 잠시 후에 허브 전원이 꺼집니다. 전원 관리 효율을 높이려면 여러 허브를 중첩하지 마십시오.

SPARC 시스템의 SUSPEND/SHUTDOWN 키 사용에 대한 자세한 내용은 [122 페이지 “USB 키보드 및 마우스 장치”](#)를 참조하십시오.

## USB 케이블 지침

USB 케이블을 연결할 때 다음 지침에 주의하십시오.

- USB 2.0 장치의 연결에는 항상 USB 2.0 호환 정격(480Mbit/초) 20/28 AWG 케이블을 사용하십시오.
- 지원되는 최대 케이블 길이는 5미터입니다.

- 케이블 연장 장치를 사용하지 마십시오. 최상의 결과를 얻으려면 자체 전원 공급 장치가 있는 허브를 사용하여 케이블 길이를 연장하십시오.

자세한 내용은 다음을 참조하십시오.

<http://www.usb.org/about/faq/ans5>



## USB 장치 사용(작업)

---

이 장에서는 Oracle Solaris OS에서 USB 장치를 사용하는 방법에 대한 단계별 지침을 제공합니다. 이 장에서는 특히 Oracle Solaris 11 릴리스의 HAL 서비스에서 USB 장치를 사용하는 방법에 대해 다룹니다.

USB 장치의 사용과 관련된 절차에 대한 자세한 내용은 다음을 참조하십시오.

- 127 페이지 “Oracle Solaris OS에서 USB 장치 관리(로드맵)”
- 128 페이지 “USB 대용량 저장 장치 사용(작업 맵)”
- 147 페이지 “USB 오디오 장치 사용(작업 맵)”
- 151 페이지 “cfgadm 명령을 사용하여 USB 장치 핫 플러그 기능 사용(작업 맵)”

USB 장치에 대한 최신 정보는 다음 사이트를 참조하십시오.

<http://www.oracle.com/us/sun/index.html>

USB 장치에 대한 개요 정보는 7 장, “USB 장치 사용(개요)”을 참조하십시오.

## Oracle Solaris OS에서 USB 장치 관리(로드맵)

이 로드맵을 사용하여 Oracle Solaris OS에 있는 USB 장치의 모든 관리 작업을 식별합니다. 각 작업은 USB 장치 사용, USB 장치의 핫 플러그 기능 사용, USB 오디오 장치 추가 등의 추가 작업과 연관됩니다.

Oracle Solaris OS의 USB 구성 요소에 대한 자세한 내용은 120 페이지 “Oracle Solaris OS의 USB 정보”를 참조하십시오.

작업	설명	수행 방법
USB 대용량 저장 장치를 사용합니다.	USB 대용량 저장 장치에 데이터를 추가하기 전에 파일 시스템이 포함되어 있어야 합니다. 또한 파일 시스템을 만들어 마운트하기 전에 USB 디스켓을 포맷해야 합니다.  이 섹션에서는 시스템에서 물리적으로 USB 장치를 추가 또는 제거하는 방법도 설명합니다.	128 페이지 “USB 대용량 저장 장치 사용(작업 맵)”
USB 오디오 장치를 추가합니다.	이 작업 맵을 사용하여 USB 오디오 장치 추가와 관련된 작업을 식별합니다.	147 페이지 “USB 오디오 장치 사용(작업 맵)”
cfgadm 명령을 사용하여 시스템에서 USB 장치를 추가 또는 제거합니다.	cfgadm 명령을 사용하여 시스템에서 USB 장치를 논리적으로 추가 또는 제거할 수 있습니다.	151 페이지 “cfgadm 명령을 사용하여 USB 장치 핫플러그 기능 사용(작업 맵)”

## USB 대용량 저장 장치 사용(작업 맵)

작업	설명	수행 방법
USB 대용량 저장 장치를 추가 또는 제거합니다.	다음 중 하나를 선택하여 USB 대용량 저장 장치를 추가합니다.  USB 대용량 저장 장치를 추가합니다.  USB 카메라를 추가하여 디지털 이미지에 액세스합니다.  USB 대용량 저장 장치를 제거합니다.	133 페이지 “USB 대용량 저장 장치를 추가하는 방법”  133 페이지 “USB 카메라를 추가하는 방법”  135 페이지 “USB 대용량 저장 장치를 제거하는 방법”
비호환 USB 대용량 저장 장치를 추가합니다.	scsa2usb.conf에 항목을 추가하여 비호환 USB 대용량 저장 장치를 추가합니다.	131 페이지 “비호환 USB 대용량 저장 장치를 사용하는 방법”
USB 대용량 저장 장치 사용을 준비합니다.	USB 대용량 저장 장치 사용을 준비합니다.	135 페이지 “USB 대용량 저장 장치 사용 준비”
USB 장치 정보를 표시합니다.	USB 장치에 대한 정보를 표시합니다.	136 페이지 “USB 장치 정보를 표시하는 방법”



작업	설명	수행 방법
USB 대용량 저장 장치에 파일 시스템을 만듭니다.	장치에 데이터를 넣으려면 먼저 파일 시스템을 만들어야 합니다.	136 페이지 “USB 대용량 저장 장치에 파일 시스템을 만드는 방법”
USB 대용량 저장 장치에서 분할 영역을 수정하고 파일 시스템을 만듭니다.	USB 대용량 저장 장치에 파일 시스템을 만들기 전에 기존 분할 영역을 수정해야 할 수도 있습니다.	139 페이지 “USB 대용량 저장 장치에서 분할 영역을 수정하고 PCFS 파일 시스템을 만드는 방법”
Solaris 분할 영역을 만들고 USB 대용량 저장 장치의 슬라이스를 수정합니다.	rmformat을 사용하여 슬라이스 정보를 수정하려면 먼저 슬라이스 정보가 있는 파일을 만들어야 합니다.	142 페이지 “USB 대용량 저장 장치에서 Solaris 분할 영역을 만들고 슬라이스를 수정하는 방법”
USB 대용량 저장 장치를 마운트합니다.	USB 대용량 저장 장치를 마운트합니다.	144 페이지 “USB 대용량 저장 장치를 마운트 또는 마운트 해제하는 방법”
(옵션) USB 장치 드라이버를 사용 안함으로 설정합니다.	시스템에서 USB 지원을 사용하지 않으려면 USB 장치 드라이버를 사용 안함으로 설정합니다.	146 페이지 “특정 USB 드라이버를 사용 안함으로 설정하는 방법”
(옵션) 사용되지 않은 USB 장치 링크를 제거합니다.	devfsadm 명령을 사용하여 USB 장치 링크를 제거합니다.	146 페이지 “사용되지 않은 USB 장치 링크를 제거하는 방법”

## USB 대용량 저장 장치 사용

다음과 같은 USB 이동식 대용량 저장 장치가 지원됩니다.

- CD-RW
- 하드 디스크
- DVD
- 디지털 카메라
- 디스켓 장치
- SmartMedia 및 CompactFlash 장치

Oracle Solaris ZFS 파일 시스템에서의 USB 대용량 저장 장치 사용에 대한 자세한 내용은 112 페이지 “USB 장치의 새로운 기능은 무엇입니까?”을 참조하십시오.

Oracle Solaris OS에서 지원되는 USB 장치의 전체 목록은 다음을 참조하십시오.

<http://www.oracle.com/us/sun/index.html>

이전의 Solaris 릴리스에서는 모든 USB 저장 장치가 이동식 매체 장치로 식별되어 자동 마운트를 비롯한 다음의 이점을 활용할 수 있었습니다. Oracle Solaris 11 릴리스에서는 USB 대용량 저장 장치가 핫 플러그 가능 장치로 식별되지만 다음과 같은 USB 이동식

장치의 이점도 활용할 수 있습니다. 핫 플러그 가능 동작에 대한 자세한 내용은 113 페이지 “USB 및 1394(FireWire) 지원”을 참조하십시오.

- 핫 플러그 가능 장치는 자동으로 마운트됩니다.
- 표준 MS-DOS 또는 Windows(FAT) 파일 시스템을 사용하는 USB 저장 장치가 지원됩니다.
- 사용자에게 친숙한 `rmformat` 명령을 사용하여 슬라임을 만들 수 있습니다. `fdisk` 명령을 사용하여 USB 장치를 분할할 수도 있지만 `format` 유틸리티나 `rmformat -F` 명령을 사용하여 USB 드라이브를 물리적으로 포맷해서는 안 됩니다.
- `rmformat` 명령을 사용하여 매체가 삽입된 모든 USB 장치를 표시합니다. 자세한 예는 136 페이지 “USB 장치 정보를 표시하는 방법”을 참조하십시오.
- `mount` 명령이 더 이상 필요하지 않으므로 이제 루트가 아닌 사용자가 USB 저장 장치에 액세스할 수 있습니다. 이 장치는 자동으로 마운트되어 `/media` 디렉토리 아래에서 사용할 수 있습니다.
- 이러한 장치는 이동식 매체 서비스를 사용하거나 사용하지 않고 관리할 수 있습니다.
- FAT 파일 시스템을 사용한 디스크는 마운트 및 액세스할 수 있습니다. 예를 들면 다음과 같습니다.

```
mount -F pcfs /dev/dsk/c2t0d0s0:c /mnt
```

- LOG SENSE 페이지를 지원하는 장치를 제외한 모든 USB 저장 장치의 전원을 관리할 수 있습니다. LOG SENSE 페이지가 있는 장치는 보통 SCSI 드라이브로, USB-to-SCSI 브리지 장치를 통해 연결됩니다.
- USB 대용량 저장 장치를 사용할 때는 응용 프로그램이 다르게 작동할 수 있습니다. 응용 프로그램에서 USB 저장 장치를 사용하는 경우 다음 문제에 유의하십시오.
  - 기존에는 디스켓과 같은 소용량 장치를 이동식 장치로 인식했기 때문에 응용 프로그램에서 매체 크기를 잘못 인식할 수 있습니다.
  - 하드 드라이브 등의 배출이 불가능한 장치에 대해 응용 프로그램이 매체 배출을 요청하는 경우 성공하지만 아무 것도 수행되지 않습니다.
  - 모든 USB 대용량 저장 장치를 이동식 매체 장치로 취급하는 이전 Solaris 릴리스의 동작을 선호하는 경우에는 `/etc/driver/drv/scsa2usb.conf` 파일을 업데이트하면 이전 동작을 강제로 적용할 수 있습니다.

USB 대용량 저장 장치의 사용에 대한 자세한 내용은 `scsa2usb(7D)`를 참조하십시오.

## USB 디스켓 장치 사용

USB 디스켓 장치는 이동식 매체 장치로 나타납니다. USB 디스켓 장치는 `fd`(플로피) 드라이버로 관리하지 않습니다. `fd`(기본 플로피) 드라이버용으로 `ioctl(2)` 호출을 전송하는 응용 프로그램은 실패합니다. `read(2)` 및 `write(2)` 호출만 전송하는 응용 프로그램은 성공합니다. SunPCI 및 `rmformat` 등의 기타 응용 프로그램도 성공합니다.

USB 디스켓 장치는 SCSI 이동식 매체 장치로 식별됩니다. 이 장치는 `/media` 디렉토리에서 액세스할 수 있습니다.

USB 디스켓 장치를 사용하는 방법에 대한 자세한 내용은 1 장, “이동식 매체 관리(개요)”를 참조하십시오.

## 비호환 USB 대용량 저장 장치 사용

일부 장치는 USB 대용량 저장 장치 클래스와 호환되는 것으로 식별되지 않거나, 잘못 식별되는 경우에도 USB 대용량 저장 장치 드라이버에서 지원됩니다. `scsa2usb.conf` 파일에는 공급업체 ID, 제품 ID 및 일치하는 대용량 저장 장치에 대한 개정, 기본 장치 속성을 대체하는 필드를 나열하는 속성 대체 목록이 포함되어 있습니다. 이 목록에 있는 항목은 기본적으로 주석 처리됩니다. 이러한 항목을 복사하여 주석 처리를 제거하면 특정 장치에 대한 지원을 사용할 수 있습니다.

이 Solaris 릴리스를 사용하는 시스템에 USB 대용량 저장 장치를 연결했는데 시스템에서 장치를 사용할 수 없는 경우에는 `/kernel/drv/scsa2usb.conf` 파일을 검사하여 이 장치와 일치하는 주석 처리된 항목이 있는지 확인할 수 있습니다. 대체 정보를 사용하여 특정 장치를 지원할 수 있는지 확인하려면 `scsa2usb.conf` 파일에 있는 정보에 따르십시오.

권장 USB 대용량 저장 장치의 목록을 보려면 다음 위치로 이동합니다.

<http://www.oracle.com/us/sun/index.html>

자세한 내용은 `scsa2usb(7D)`를 참조하십시오.

### ▼ 비호환 USB 대용량 저장 장치를 사용하는 방법

- 1 관리자로 전환합니다.
- 2 `/etc/driver/drv/scsa2usb.conf` 파일에 항목을 추가합니다.  
다음 항목은 USB 메모리 스틱에 해당됩니다.  
`attribute-override-list = "vid=* reduced-cmd-support=true";`
- 3 시스템을 재부트하거나 다음을 수행합니다.
  - a. 장치를 제거합니다.
  - b. 드라이버를 수동으로 업데이트합니다.  
`# update_drv -f scsa2usb`
  - c. 장치를 추가합니다.

## USB 대용량 저장 장치의 핫 플러그 기능 사용

장치의 핫 플러그 기능을 사용하면 운영 체제를 종료하거나 시스템 전원을 끄지 않고 장치를 추가 또는 제거할 수 있습니다. 모든 USB 장치는 핫 플러그 가능 장치입니다.

이제 이동식 매체 관리자가 핫 플러그 가능 장치를 인식할 수 있습니다. 장치를 플러그인하기만 하면 몇 초 안에 마운트됩니다. 아무 작업도 수행되지 않는 경우는 장치가 마운트되었는지 확인하십시오.

이동식 매체 서비스가 실행 중인지 확인합니다.

```
# svcs hal dbus rvmolmgr
STATE          STIME    FMRI
online         May_03   svc:/system/dbus:default
online         May_03   svc:/system/hal:default
online         May_03   svc:/system/filesystem/rvmolmgr:default
```

파일 시스템이 유효하며 인식되는 경우 장치에서 마운트할 수 있습니다.

장치에 있는 파일 시스템이 자동으로 마운트되지 않는 경우는 수동 마운트를 시도하십시오.

장치의 핫 플러그 기능을 사용하려면 먼저 `eject -l` 명령의 별칭에서 장치 이름을 찾습니다. 그리고 장치의 매체를 꺼냅니다. 이 단계를 수행하지 않으면 장치가 해제되어 포트를 다시 사용할 수 있게 되지만 장치에 있는 시스템이 손상될 수 있습니다.

USB 장치의 핫 플러그 기능을 사용하면 장치가 시스템의 장치 계층에 즉시 나타나고 `prtconf` 명령 출력으로 표시됩니다. 장치를 사용하고 있지 않은 경우 USB 장치를 제거하면 시스템의 장치 계층에서 바로 제거됩니다.

장치의 연결이 해제되어 있는 동안 장치를 사용하려 하면 장치 노드는 남지만 이 장치를 제어하는 드라이버가 장치의 모든 작업을 중지합니다. 이 장치로 전송되는 새 I/O 작업은 모두 오류를 반환합니다.

이 상황에서는 시스템이 원래 장치를 플러그인하라는 내용의 메시지를 표시합니다. 장치를 더 이상 사용할 수 없으면 응용 프로그램을 중지하십시오. 몇 초가 지나면 포트를 다시 사용할 수 있게 됩니다.

---

**주 -** 활성 상태이거나 열려 있는 장치를 제거하면 데이터 무결성이 손상될 수 있습니다. 장치를 제거하기 전에 항상 닫으십시오. 단, 키보드와 마우스는 활성 상태인 동안 이동이 가능합니다.

---

## ▼ USB 대용량 저장 장치를 추가하는 방법

- 1 관리자로 전환합니다.
- 2 USB 대용량 저장 장치를 연결합니다.
- 3 USB 장치가 추가되었는지 확인합니다.

예를 들면 다음과 같습니다.

```
$ rmformat
Looking for devices...
  1. Logical Node: /dev/rdisk/c3t0d0p0
     Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
     Connected Device: SanDisk  Cruzer Micro    0.3
     Device Type: Removable
     Bus: USB
     Size: 245.0 MB
     Label: <None>
     Access permissions: Medium is not write protected.
```

- 4 장치가 /media 디렉토리에 자동으로 마운트되었는지 확인합니다.

예를 들면 다음과 같습니다.

```
$ ls /media/NONAME
aa  bb
```

rmmount -l 명령을 사용하여 마운트 가능한 장치의 경로와 명령을 나열할 수도 있습니다. 장치가 /media 디렉토리에 마운트된 경우에는 다음과 비슷한 출력이 표시됩니다.

```
$ rmmount -l
/dev/dsk/c3t0d0p0:1  rmdisk0,NONAME,/media/NONAME
```

## ▼ USB 카메라를 추가하는 방법

카메라의 매체가 PCFS 파일 시스템을 사용하는 경우에는 자동으로 마운트됩니다. 장치가 scsa2usb 드라이버에 바인딩되지 않는 경우에는 libusb 응용 프로그램을 사용하여 사진을 전송합니다. 자세한 내용은 /usr/sfw/share/doc/libusb/libusb.txt 를 참조하십시오.

- 1 관리자로 전환합니다.
- 2 USB 카메라를 플러그인하고 켵니다.
 

시스템에서 카메라에 대해 논리적 장치를 만듭니다. 카메라가 플러그인되고 나면 /var/adm/messages 파일에 출력이 기록되며 장치 연결을 알립니다. 시스템에서는 카메라를 저장 장치로 취급합니다.
- 3 /var/adm/messages 파일에 기록된 출력을 검사하십시오.

```
# more /var/adm/messages
```

이 출력을 검사하면 만들어진 논리적 장치를 확인하고 그 장치를 사용하여 이미지에 액세스할 수 있습니다. 출력은 다음과 유사합니다.

```
Jul 15 09:53:35 buffy usba: [ID 349649 kern.info] OLYMPUS, C-3040ZOOM,
000153719068
Jul 15 09:53:35 buffy genunix: [ID 936769 kern.info] scsa2usb1 is
/pci@0,0/pci925,1234@7,2/storage@2
Jul 15 09:53:36 buffy scsi: [ID 193665 kern.info] sd3 at scsa2usb1:
target 0 lun 0
```

다음은 수행하여 마운트 가능한 /dev/dsk 링크 항목이 있는 장치와 대조합니다.

```
# ls -l /dev/dsk/c*0 | grep /pci@0,0/pci925,1234@7,2/storage@2
lrwxrwxrwx 1 root root 58 Jun 14 2010 c3t0d0p0 ->
.././devices/pci@0,0/pci925,1234@7,2/storage@2/disk@0,0:a
```

#### 4 USB 카메라 파일 시스템을 마운트합니다.

카메라의 파일 시스템은 일반적으로 PCFS 파일 시스템입니다. 파일 시스템이 PCFS이면 자동으로 마운트됩니다.

x86 시스템에 파일 시스템을 수동으로 마운트하려면 다음과 비슷한 구문을 사용합니다.

```
# mount -F pcfs /dev/dsk/c3t0d0p0:c /mnt
```

SPARC 시스템에 파일 시스템을 수동으로 마운트하려면 다음과 비슷한 구문을 사용합니다.

```
# mount -F pcfs /dev/dsk/c3t0d0s0:c /mnt
```

파일 시스템의 마운트에 대한 자세한 내용은 328 페이지 “Oracle Solaris 파일 시스템 마운트 및 마운트 해제”를 참조하십시오.

다양한 PCFS 파일 시스템의 마운트에 대한 자세한 내용은 mount\_pcfs(1M)를 참조하십시오.

#### 5 이미지 파일을 사용할 수 있는지 확인합니다.

예를 들면 다음과 같습니다.

```
# ls /mnt/DCIM/100OLYMP/
P7220001.JPG* P7220003.JPG* P7220005.JPG*
P7220002.JPG* P7220004.JPG* P7220006.JPG*
```

#### 6 USB 카메라에서 만든 이미지 파일을 보고 조작합니다.

예를 들면 다음과 같습니다.

```
# /usr/dt/bin/sdtimage P7220001.JPG &
```

#### 7 카메라를 연결 해제하기 전에 시스템을 마운트 해제합니다.

예를 들면 다음과 같습니다.

```
# umount /mnt
```

#### 8 (옵션) 카메라를 끄고 연결 해제합니다.

## ▼ USB 대용량 저장 장치를 제거하는 방법

- 1 관리자로 로그인합니다.
- 2 장치를 사용하는 활성 응용 프로그램을 모두 중지합니다.
- 3 장치를 마운트 해제합니다.

```
$ rmount NONAME
```

또는 관리자로 `umount` 명령을 사용합니다. 예를 들면 다음과 같습니다.

```
# umount /media/NONAME
```

USB 장치의 마운트 해제에 대한 자세한 내용은 144 페이지 “USB 대용량 저장 장치를 마운트 또는 마운트 해제하는 방법”을 참조하십시오.

- 4 장치를 제거합니다.

## USB 대용량 저장 장치 사용 준비

이동식 매체 서비스를 사용하거나 사용하지 않고 이동식 매체의 정보에 액세스할 수 있습니다. GNOME의 File Manager를 사용하여 이동식 매체 관련 정보에 액세스하는 방법에 대한 자세한 내용은 GNOME 데스크탑 설명서를 참조하십시오.

USB 장치가 포맷된 후에는 일반적으로 `/media/label` 디렉토리 아래에 마운트됩니다.

문자 장치의 경우에는 `/dev/rdisk` 디렉토리에 장치 노드가 만들어지고, 블록 장치의 경우에는 `/dev/dsk` 디렉토리 아래에 장치 노드가 만들어집니다. 장치가 핫 플러그 기능으로 사용되면 장치 링크가 만들어집니다. 자세한 내용은 [scsa2usb\(7D\)](#)를 참조하십시오.

이동식 매체 서비스에서 장치를 식별할 수 없는 경우에는 수동으로 장치를 마운트해 봅니다. 먼저 `rmformat` 명령을 사용하여 장치 경로를 식별한 다음 `mount` 명령을 사용하여 관리자로서 장치를 수동으로 마운트합니다.

장치가 이동식 매체 서비스에서 마운트된 경우에는 `rmount` 명령을 사용하여 마운트 해제할 수 있습니다. 장치가 수동으로 마운트된 경우에는 관리자로서 `umount` 명령을 사용하여 마운트 해제할 수 있습니다.

USB 장치의 마운트 및 마운트 해제에 대한 자세한 내용은 144 페이지 “USB 대용량 저장 장치를 마운트 또는 마운트 해제하는 방법”을 참조하십시오.

## ▼ USB 장치 정보를 표시하는 방법

### ● USB 장치에 대한 정보를 표시합니다.

예를 들면 `prtconf` 명령을 사용하여 USB 장치 정보를 표시합니다. 이 예에서 `prtconf` 출력은 USB 장치 정보만 표시하도록 잘렸습니다.

```
$ prtconf
usb, instance #0
  hub, instance #2
    device, instance #8
      interface (driver not attached)
    printer (driver not attached)
  mouse, instance #14
    device, instance #9
      keyboard, instance #15
      mouse, instance #16
  storage, instance #7
    disk (driver not attached)
  communications, instance #10
    modem (driver not attached)
    data (driver not attached)
  storage, instance #0
    disk (driver not attached)
  storage, instance #1
    disk (driver not attached)
```

예를 들면 `rmformat` 명령을 사용하여 USB 장치 정보를 표시합니다.

```
$ rmformat
Looking for devices...
1. Logical Node: /dev/rdisk/c3t0d0p0
   Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
   Connected Device: SanDisk  Cruzer Micro    0.3
   Device Type: Removable
   Bus: USB
   Size: 245.0 MB
   Label: <None>
   Access permissions: Medium is not write protected.
```

## ▼ USB 대용량 저장 장치에 파일 시스템을 만드는 방법

USB 디스켓에 파일 시스템을 추가하려면 먼저 해당 디스켓을 포맷해야 합니다. 다른 모든 USB 대용량 저장 장치는 파일 시스템만 있으면 사용할 수 있습니다. USB 장치를 포맷할 때 다음 요점에 주의하십시오.

- USB 디스켓이 아닌 경우에는 `rmformat -F`를 사용하지 마십시오.
- 기본 슬라이스가 적합하지 않은 경우에는 `rmformat -s` 명령을 사용하여 슬라이스를 만듭니다. 필요한 경우 `fdisk` 유틸리티를 사용하여 USB 장치를 분할합니다. 단계별 지침을 보려면 다음을 참조하십시오.
  - 139 페이지 “USB 대용량 저장 장치에서 분할 영역을 수정하고 PCFS 파일 시스템을 만드는 방법”



- 142 페이지 “USB 대용량 저장 장치에서 Solaris 분할 영역을 만들고 슬라이스를 수정하는 방법”
- USB 장치가 자동으로 마운트된 경우 USB 장치에 파일 시스템을 만들려면 먼저 마운트 해제해야 합니다. `rmmount -l` 명령을 사용하여 장치 별명을 확인한 다음 `rmumount` 명령을 사용하여 USB 장치를 마운트 해제합니다.

---

주 - USB 디스켓을 포맷해야 하는 경우에만 4-5 단계를 수행하십시오.

---

- 1 관리자로 전환합니다.
- 2 USB 장치를 시스템에 추가합니다. USB 장치의 핫 플러그 기능 사용에 대한 자세한 내용은 다음을 참조하십시오.
  - 132 페이지 “USB 대용량 저장 장치의 핫 플러그 기능 사용”
  - 151 페이지 “`cfgadm` 명령을 사용하여 USB 장치 핫 플러그 기능 사용(작업 맵)”

3 (옵션) USB 장치를 식별합니다.

예를 들면 다음과 같습니다.

```
# rmformat
Looking for devices...
  1. Logical Node: /dev/rdsk/c2t0d0p0
     Physical Node: /pci@0,0/pci108e,534a@2,1/hub7/floppy@1/disk@0,0
     Connected Device: MITSUMI USB FDD 1039
     Device Type: Floppy drive
     Bus: USB
     Size: 1.4 MB
     Label: <None>
     Access permissions: Medium is not write protected.
```

이 예에서 디스켓 장치는 `c2t0d0p0`입니다.

- 4 필요한 경우 디스켓을 디스켓 드라이브에 삽입합니다.
- 5 필요한 경우 디스켓을 포맷합니다.

```
% rmformat -F long raw-device
```

예를 들어, SPARC 시스템에서는 다음을 수행합니다.

```
% rmformat -F long /dev/rdsk/c2t0d0s2
```

예를 들어, x86 시스템에서는 다음을 수행합니다.

```
% rmformat -F long /dev/rdsk/c3t0d0p0
```

- 6 파일 시스템 유형을 확인하고 장치가 마운트 해제되었는지 확인합니다. 그런 다음, 다음 중 하나를 선택합니다.

USB 장치의 마운트 해제에 대한 자세한 내용은 144 페이지 “USB 대용량 저장 장치를 마운트 또는 마운트 해제하는 방법”을 참조하십시오.

- PCFS 파일 시스템을 만듭니다.

```
# mkfs -F pcfs -o nofdisk,size=size raw-device
```

-size 옵션을 512바이트 블록으로 지정합니다.

다음 예는 SPARC 시스템에서 1.4MB 디스켓에 PCFS 파일 시스템을 만드는 방법을 보여 줍니다.

```
# mkfs -F pcfs /dev/rdisk/c2t0d0p0
```

```
Construct a new FAT file system on /dev/rdisk/c2t0d0p0: (y/n)? y
```

다음 예는 x86 시스템에서 1.4MB 디스켓에 PCFS 파일 시스템을 만드는 방법을 보여 줍니다.

```
# mkfs -F pcfs /dev/rdisk/c2t0d0s2
```

```
Construct a new FAT file system on /dev/rdisk/c2t0d0s2: (y/n)? y
```

다음 예는 SPARC 시스템에서 100MB USB 메모리 스틱에 PCFS 파일 시스템을 만드는 방법을 보여 줍니다.

```
# mkfs -F pcfs /dev/rdisk/c5t0d0s2:c
```

다음 예는 x86 시스템에서 100MB USB 메모리 스틱에 PCFS 파일 시스템을 만드는 방법을 보여 줍니다.

```
# mkfs -F pcfs /dev/rdisk/c5t0d0p0:c
```

이 명령을 완료하는 데 몇 분 정도 걸릴 수 있습니다.

- 레거시 UFS 파일 시스템을 만듭니다.

```
# newfs raw-device
```

예를 들면 다음과 같습니다.

```
# newfs /dev/rdisk/c4t0d0s7
```

대용량 USB 하드 디스크에는 newfs -f 4096 옵션이나 newfs -T 옵션의 사용을 고려합니다.

---

주 - 디스켓의 저장소 용량이 매우 제한적이기 때문에 UFS 파일 시스템 오버헤드는 디스켓 공간에서 큰 부분을 차지합니다.

---

USB 대용량 저장 장치에서 PCFS 파일 시스템을 만들고 슬라이스를 수정하는 작업에 대한 자세한 예는 다음 두 절차를 참조하십시오.

## ▼ USB 대용량 저장 장치에서 분할 영역을 수정하고 PCFS 파일 시스템을 만드는 방법

다음 단계는 기존 분할 영역을 삭제하고 새 분할 영역을 만든 다음 USB 장치에 PCFS 파일 시스템을 만드는 방법을 설명합니다. 이 작업을 수행하기 전에 모든 데이터를 백업하십시오.

1 관리자로 전환합니다.

2 **fdisk** 유틸리티를 시작합니다.

예를 들면 다음과 같습니다.

```
# fdisk /dev/rdisk/c3t0d0p0
```

3 옵션 3을 선택하여 분할 영역을 삭제합니다.

예를 들면 다음과 같습니다.

```
Total disk size is 29 cylinders
      Cylinder size is 2048 (512 byte) blocks
```

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	=====	=====	=====
1	Active	Solaris2	1	28	28	97

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection: 3

4 삭제할 분할 영역 번호를 선택합니다.

예를 들면 다음과 같습니다.

```
Total disk size is 29 cylinders
      Cylinder size is 2048 (512 byte) blocks
```

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	=====	=====	=====
1	Active	Solaris2	1	28	28	97

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Specify the partition number to delete (or enter 0 to exit): 1

Partition deleted.

**5 분할 영역을 만듭니다.**

예를 들면 다음과 같습니다.

Total disk size is 29 cylinders  
 Cylinder size is 2048 (512 byte) blocks

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	=====	=====	=====

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection: 1

**6 FAT32 분할 영역 유형을 선택합니다.**

Total disk size is 29 cylinders  
 Cylinder size is 2048 (512 byte) blocks

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	=====	=====	=====

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Select the partition type to create:

- |            |             |              |           |
|------------|-------------|--------------|-----------|
| 1=SOLARIS2 | 2=UNIX      | 3=PCIXOS     | 4=Other   |
| 5=DOS12    | 6=DOS16     | 7=DOSEXT     | 8=DOSBIG  |
| 9=DOS16LBA | A=x86 Boot  | B=Diagnostic | C=FAT32   |
| D=FAT32LBA | E=DOSEXTLBA | F=EFI        | 0=Exit? c |

**7 이 분할 영역에 사용할 디스크 비율을 지정합니다.**

Total disk size is 29 cylinders  
 Cylinder size is 2048 (512 byte) blocks

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	=====	=====	=====

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Select the partition type to create:

Specify the percentage of disk to use for this partition (or type "c" to specify the size in cylinders). **100**

## 8 새 분할 영역을 활성 또는 비활성 분할 영역으로 선택합니다.

Total disk size is 29 cylinders  
Cylinder size is 2048 (512 byte) blocks

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	====	=====	====

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Select the partition type to create:

Should this become the active partition? If yes, it will be activated each time the computer is reset or turned on.

Please type "y" or "n". **n**

## 9 디스크 구성을 업데이트하고 종료합니다.

Total disk size is 29 cylinders  
Cylinder size is 2048 (512 byte) blocks

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	====	=====	====
1		Win95 FAT32	1	28	28	97

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection: **5**

## 10 이 분할 영역에 PCFS 파일 시스템을 만듭니다.

새 파일 시스템을 만들기 전에 장치가 마운트 해제되었는지 확인합니다. USB 장치의 마운트 해제에 대한 자세한 내용은 144 페이지 “USB 대용량 저장 장치를 마운트 또는 마운트 해제하는 방법”을 참조하십시오.

예를 들면 다음과 같습니다.

```
# mkfs -F pcfs -o fat=32 /dev/rdisk/c3t0d0p0:c
Construct a new FAT file system on /dev/rdisk/c3t0d0p0:c: (y/n)? y
```

## ▼ USB 대용량 저장 장치에서 Solaris 분할 영역을 만들고 슬라이스를 수정하는 방법

다음 단계는 Solaris 분할 영역을 만들고 슬라이스를 수정하는 방법을 보여 줍니다.

이 작업을 수행하기 전에 데이터를 모두 백업해야 합니다.

1 슈퍼유저로 로그인합니다.

2 **fdisk** 유틸리티를 시작합니다.

예를 들면 다음과 같습니다.

```
# fdisk /dev/rdisk/c5t0d0s2
No fdisk table exists. The default partition for the disk is:
```

```
  a 100% "SOLARIS System" partition
```

```
Type "y" to accept the default partition, otherwise type "n" to edit the
partition table.
```

```
y
```

3 현재 슬라이스를 표시합니다.

예를 들면 다음과 같습니다.

```
# prtvtoc /dev/rdisk/c5t0d0s2
* /dev/rdisk/c5t0d0s2 partition map
*
* Dimensions:
*   512 bytes/sector
*   63 sectors/track
*   255 tracks/cylinder
*  16065 sectors/cylinder
*   5836 cylinders
*   5836 accessible cylinders
*
* Flags:
*   1: unmountable
*  10: read-only
*
* Partition  Tag  Flags      First      Sector      Last
* Partition  Tag  Flags      Sector     Count       Sector  Mount Directory
*   0         0    00         0  93755340  93755339
*   2         0    00         0  93755340  93755339
```

#### 4 슬라이스 정보를 사용하여 텍스트 파일을 만듭니다.

예를 들면 다음과 같습니다.

```
slices: 0 = 0, 5GB, "wm", "home" :
        1 = 8225280000, 6GB :
        2 = 0, 44GB, "wm", "backup" :
        6 = 16450560000, 15GB
```

각 슬라이스가 실린더 경계에서 시작되는지 확인합니다. 예를 들어, 슬라이스 1은 실린더 크기를 바이트 수로 나타낸 값에 1000을 곱한 822280000에서 시작합니다.

자세한 내용은 `rmformat(1)`의 `-s` 옵션 설명을 참조하십시오.

#### 5 위에서 만든 슬라이스 파일을 포함하여 슬라이스를 만듭니다.

예를 들면 다음과 같습니다.

```
# rmformat -s slice_file /dev/rdisk/c5t0d0s2
```

#### 6 새 슬라이스 정보를 봅니다.

예를 들면 다음과 같습니다.

```
# prtvtoc /dev/rdisk/c5t0d0s2
* /dev/rdisk/c5t0d0s2 partition map
*
* Dimensions:
*   512 bytes/sector
*   63 sectors/track
*   255 tracks/cylinder
* 16065 sectors/cylinder
*   5836 cylinders
*   5836 accessible cylinders
*
* Flags:
*   1: unmountable
*  10: read-only
*
* Unallocated space:
*   First Sector Last
*   Sector Count Sector
*   10485760 5579240 16064999
*   28647912 3482088 32129999
*   63587280 30168060 93755339
*
*
* Partition Tag Flags First Sector Last Mount Directory
*            Sector Count Sector
*   0         8    00      0 10485760 10485759
*   1         3    01 16065000 12582912 28647911
*   2         5    00      0 92274688 92274687
*   6         4    00 32130000 31457280 63587279
```

## ▼ USB 대용량 저장 장치를 마운트 또는 마운트 해제하는 방법

다음 단계는 USB 대용량 저장 장치를 마운트 및 마운트 해제하는 방법을 보여 줍니다.

- 1 슈퍼유저 또는 콘솔 사용자로 로그인합니다.

- 2 (옵션) 장치를 식별합니다.

예를 들면 다음과 같습니다.

```
$ rmformat
Looking for devices...
  1. Logical Node: /dev/rdisk/c3t0d0p0
     Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
     Connected Device: SanDisk  Cruzer Micro    0.3
     Device Type: Removable
     Bus: USB
     Size: 245.0 MB
     Label: <None>
     Access permissions: Medium is not write protected.
```

이 예에서 물리적 디스크 장치의 이름은 `c2t0d0p0`입니다.

- 3 다음 중 하나를 선택하여 USB 대용량 저장 장치를 마운트 또는 마운트 해제합니다.

- 콘솔 사용자로 USB 대용량 저장 장치를 마운트합니다.

다음과 비슷하게 장치 별명, 마운트 지점 또는 장치 경로를 사용하여 `rmmount` 명령을 사용할 수 있습니다.

```
$ rmmount rmdisk0
$ rmmount NONAME
$ rmmount /dev/dsk/c3t0d0p0:1
```

예를 들면 다음과 같습니다.

```
$ rmmount NONAME
NONAME /dev/dsk/c2t0d0p0 mounted
$ ls /media/NONAME
AA.TXT
```

- 콘솔 사용자로 USB 대용량 저장 장치를 마운트 해제합니다.

예를 들면 다음과 같습니다.

```
$ rmumount NONAME
NONAME /dev/dsk/c2t0d0p0 unmounted
```

- 슈퍼유저로 USB 대용량 저장 장치를 마운트합니다.

이 예는 UFS 파일 시스템이 있는 장치를 마운트하는 방법을 보여 줍니다.

```
# mount /dev/dsk/c1t0d0s2 /mnt
```

이 예는 SPARC 시스템에 PCFS 파일 시스템이 있는 장치를 마운트하는 방법을 보여 줍니다.



```
# mount -F pcfs /dev/dsk/c3t0d0s2:c /mnt
```

이 예는 x86 시스템에 PCFS 파일 시스템이 있는 장치를 마운트하는 방법을 보여 줍니다.

```
# mount -F pcfs /dev/dsk/c3t0d0p0:c /mnt
```

이 예는 읽기 전용의 HSFS 파일 시스템이 있는 CD를 마운트하는 방법을 보여 줍니다.

```
# mount -F hsfs -o ro /dev/dsk/c1t0d0s2 /mnt
```

- 수퍼유저로 USB 대용량 저장 장치를 마운트 해제합니다.

먼저 장치에서 파일 시스템을 사용하는 사람이 없는지 확인합니다.

예를 들면 다음과 같습니다.

```
# fuser -c -u /mnt
# umount /mnt
```

- 4 장치를 꺼냅니다. DVD, CD 또는 디스켓 장치의 경우에는 선택 사항입니다.

예를 들면 다음과 같습니다.

```
# eject /dev/rdisk/c1t0d0s2
```

## USB 대용량 저장 장치 문제 해결 팁

USB 대용량 저장 장치를 추가 또는 제거하는 데 문제가 있는 경우에는 다음 사항에 주의하십시오.

/var/adm/messages 파일에서 장치 열거에 오류가 없는지 확인합니다. 열거 오류가 발생한 경우에는 USB 허브를 삽입하거나 허브를 제거한 후 루트 USB 허브에 직접 연결하십시오.

- 시스템을 실행하는 동안 연결한 장치에 액세스하는 데 문제가 있는 경우에는 다음 명령을 시도해 보십시오.

```
# devfsadm
```

- 일시 중단 작업으로 시스템 전원이 꺼진 경우 장치를 이동하지 마십시오. 자세한 내용은 124 페이지 “SPARC: USB 전원 관리”를 참조하십시오.
- 응용 프로그램에서 장치를 사용하는 동안 장치를 제거하여 더 이상 사용할 수 없는 경우에는 응용 프로그램을 중지합니다. prtconf 명령을 사용하여 장치 노드가 제거되었는지 확인합니다.

## 특정 USB 드라이버를 사용 안함으로 설정

클라이언트 드라이버를 사용 안함으로 설정하면 특정 유형의 USB 장치를 사용하지 않을 수 있습니다. 예를 들어, USB 프린터에 지시를 내리는 usbprn 드라이버를 사용 안함으로 설정하면 USB 프린터를 사용하지 않을 수 있습니다. usbprn을 사용 안함으로 설정해도 USB 저장 장치 등의 다른 장치 종류에는 영향을 주지 않습니다.

다음 표는 몇 가지 USB 장치 유형과 해당 드라이버를 나타냅니다.

장치 유형	사용 안함으로 설정할 드라이버
오디오	usb_ac 및 usb_as
HID(일반적으로 키보드와 마우스)	hid
저장소	scsa2usb
프린터	usbprn
직렬	usbser_edge

아직 시스템에 연결되어 있는 USB 장치의 드라이버를 사용 안함으로 설정하면 다음과 비슷한 콘솔 메시지가 표시됩니다.

```
usba10: WARNING: usba: no driver found for device name
```

## ▼ 특정 USB 드라이버를 사용 안함으로 설정하는 방법

- 1 슈퍼유저로 로그인합니다.
- 2 `/etc/system` 파일에서 드라이버 별칭 항목을 제외합니다.  
예를 들어, 다음과 같은 `exclude` 문을 포함하여 `usbprn` 드라이버를 제외합니다.  
`exclude: usbprn`
- 3 시스템을 다시 부트합니다.  
`# init 6`

## ▼ 사용되지 않은 USB 장치 링크를 제거하는 방법

시스템 전원이 꺼져 있는 동안 USB 장치가 제거된 경우에 이 절차를 사용합니다. 시스템 전원이 꺼져 있는 동안 USB 장치를 제거하면 존재하지 않는 장치의 링크가 남을 수 있습니다.

- 1 슈퍼유저로 로그인합니다.
- 2 장치에 액세스할 만한 응용 프로그램을 모두 닫습니다.
- 3 특정 USB 클래스의 사용되지 않은 링크를 제거합니다.  
예를 들면 다음과 같습니다.  
`# devfsadm -C -c audio`

또는 그냥 남은 링크를 제거합니다.

```
# devfsadm -C
```

## USB 오디오 장치 사용(작업 맵)

작업	설명	수행 방법
USB 오디오 장치를 추가합니다.	USB 마이크와 스피커를 추가합니다.	148 페이지 “USB 오디오 장치를 추가하는 방법”
시스템의 기본 오디오 장치를 확인합니다.	어떤 오디오 장치가 기본 오디오 장치인지 확인합니다.	149 페이지 “시스템의 기본 오디오 장치를 식별하는 방법”
기본 USB 오디오 장치를 변경합니다.	USB 오디오 장치를 제거 또는 변경한 경우 한 오디오 장치를 기본 오디오 장치로 만들고 싶을 수도 있습니다.	150 페이지 “기본 USB 오디오 장치를 변경하는 방법”
사용되지 않은 USB 장치 링크를 제거합니다.	시스템 전원이 꺼져 있는 동안 USB 오디오 장치를 제거하면 <code>/dev/audio</code> 장치가 존재하지 않는 <code>/dev/sound/*</code> 장치를 가리킬 수 있습니다.	146 페이지 “사용되지 않은 USB 장치 링크를 제거하는 방법”
USB 오디오 문제를 해결합니다.	USB 스피커에서 소리가 나지 않는 경우에는 이 섹션을 활용합니다.	150 페이지 “USB 오디오 장치 문제 해결”

## USB 오디오 장치 사용

특정 Oracle Solaris 릴리스의 USB 오디오 지원에 대한 자세한 내용은 114 페이지 “USB 장치용 Oracle Solaris 지원”을 참조하십시오.

이 Oracle Solaris 릴리스는 함께 작동하는 두 드라이버, `usb_ac` 및 `usb_as`의 쌍으로 구현되는 USB 오디오 지원을 제공합니다. 오디오 제어 드라이버인 `usb_ac`는 사용자 응용 프로그램에 제어 인터페이스를 제공하는 Solaris USB 아키텍처 호환 클라이언트 드라이버입니다. 오디오 스트리밍 드라이버인 `usb_as`는 재생 및 녹음 시에 오디오 데이터 메시지를 처리합니다. 여기서는 샘플 주파수와 정밀도를 설정하고 `usb_ac` 드라이버의 요청을 인코딩합니다. 두 드라이버 모두 USB 오디오 클래스 1.0 사양을 준수합니다.

일부 오디오 장치는 소프트웨어 제어를 통해 볼륨을 설정할 수 있습니다. STREAMS 모듈인 `usb_ah`는 이 기능의 관리를 위해 HID 드라이버의 위로 푸시됩니다.

Oracle Solaris는 재생 전용, 녹음 전용 또는 녹음 및 재생용 USB 오디오 장치를 지원합니다. USB 오디오 장치의 핫 플러그 기능 사용이 지원됩니다.

- USB 오디오 장치는 USB 커넥터가 있는 SPARC Ultra 및 x86 플랫폼에서 지원됩니다.
- 완전히 지원되는 오디오 데이터 형식 정보는 [usb\\_ac\(7D\)](#)를 참조하십시오.

기본 오디오 장치는 `/dev/audio`입니다. 다음 명령을 사용하면 `/dev/audio`가 USB 오디오를 가리키는 지 확인할 수 있습니다.

```
% mixerctl
Device /dev/audiocctl:
  Name    = USB Audio
  Version = 1.0
  Config  = external
```

Audio mixer for `/dev/audiocctl` is enabled

USB 오디오 장치를 연결하고 나면 `audioplay` 및 `audiorecord` 명령을 사용하여 `/dev/sound/N` 장치 링크를 통해 액세스합니다.

`/dev/audio` 및 `/dev/sound/N` 장치는 스피커, 마이크 또는 조합된 장치를 나타낼 수 있습니다. 잘못된 장치 유형을 참조하면 명령이 실패합니다. 예를 들어 `audioplay` 명령을 마이크에 사용하려고 하면 실패합니다.

`audioplay` 및 `audiorecord`와 같은 대부분의 Oracle 오디오 응용 프로그램에 대해 `AUDIODEV` 셸 변수를 설정하거나 이러한 명령에 `-d` 옵션을 지정하면 특정 기본 오디오 장치를 선택할 수 있습니다. 그러나 `/dev/audio`가 오디오 파일로 하드코딩된 타사 응용 프로그램의 경우에는 `AUDIODEV`를 설정해도 소용이 없습니다.

`/dev/audio`가 사용 중인 경우가 아니면 USB 오디오 장치를 플러그인했을 때 자동으로 그 장치가 기본 오디오 장치 `/dev/audio`가 됩니다. `/dev/audio`를 내장 오디오에서 USB 오디오로 변경하거나 그 반대로 변경하는 방법에 대한 지침은 [150 페이지 “기본 USB 오디오 장치를 변경하는 방법”](#) 및 [usb\\_ac\(7D\)](#)를 참조하십시오.

## 여러 개의 USB 오디오 장치에서 핫 플러그 기능 사용

USB 오디오 장치를 시스템에 플러그인하면 그 장치가 기본 오디오 장치 `/dev/audio`가 됩니다. 이 장치는 시스템을 재부트한 후에도 기본 오디오 장치로 유지됩니다. 추가 USB 장치를 플러그인하면 마지막 장치가 기본 오디오 장치가 됩니다.

USB 오디오 장치 문제 해결에 대한 자세한 내용은 [usb\\_ac\(7D\)](#)를 참조하십시오.

### ▼ USB 오디오 장치를 추가하는 방법

#### 1 USB 스피커를 플러그인합니다.

기본 오디오 장치 `/dev/audio`가 USB 스피커를 가리킵니다.

```
% ls -l /dev/audio
lrwxrwxrwx  1 root    root    10 Feb 13 08:46 /dev/audio -> usb/audio0
```

## 2 (옵션)스피커를 제거합니다.그리고 다시 플러그인합니다.

스피커를 제거하면 /dev/audio 장치가 다시 내장 오디오로 돌아갑니다.

```
% ls -l /dev/audio
lrwxrwxrwx 1 root root 7 Feb 13 08:47 /dev/audio -> sound/0
```

## 3 USB 마이크를 추가합니다.

```
% ls -l /dev/audio
lrwxrwxrwx 1 root root 10 Feb 13 08:54 /dev/audio -> usb/audio1
```

# ▼ 시스템의 기본 오디오 장치를 식별하는 방법

이 절차에서는 USB 오디오 장치를 이미 연결한 경우를 가정합니다.

### ● 시스템의 새 오디오 링크를 검사합니다.

- ls 명령을 사용하여 시스템의 새 오디오 링크를 표시합니다.

예를 들면 다음과 같습니다.

```
% ls -lt /dev/audio*
lrwxrwxrwx 1 root root 7 Jul 23 15:46 /dev/audio -> usb/audio0
lrwxrwxrwx 1 root root 10 Jul 23 15:46 /dev/audiocctl ->
usb/audiocctl0/
% ls -lt /dev/sound/*
lrwxrwxrwx 1 root root 74 Jul 23 15:46 /dev/sound/1 ->
../devices/pci@1f,4000/usb@5/hub@1/device@3/sound-control@0:...
lrwxrwxrwx 1 root root 77 Jul 23 15:46 /dev/sound/1ctl ->
../devices/pci@1f,4000/usb@5/hub@1/device@3/sound-control@0:...
lrwxrwxrwx 1 root other 66 Jul 23 14:21 /dev/sound/0 ->
../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audio
lrwxrwxrwx 1 root other 69 Jul 23 14:21 /dev/sound/0ctl ->
../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audiocctl
%
```

기본 오디오 장치 /dev/audio가 새로 플러그인된 USB 오디오 장치 /dev/usb/audio0을 가리키는 것을 확인합니다.

- prtconf 명령을 사용하여 시스템의 USB 오디오 장치를 검사하고 USB 장치 정보를 찾을 수도 있습니다.

```
% prtconf
.
.
.
usb, instance #0
  hub, instance #0
  mouse, instance #0
  keyboard, instance #1
  device, instance #0
    sound-control, instance #0
    sound, instance #0
    input, instance #0
.
```

## ▼ 기본 USB 오디오 장치를 변경하는 방법

- 다음 중 하나를 선택하여 기본 USB 오디오 장치를 변경합니다.
  - 내장 오디오 장치를 기본 오디오 장치로 사용하려면 USB 오디오 장치를 제거합니다. 그러면 `/dev/audio` 링크가 `/dev/sound/0` 항목을 가리킵니다. `/dev/sound/0` 항목이 기본 오디오 장치가 아닌 경우에는 시스템을 종료하고 `boot -r` 명령을 사용하거나 `devfsadm -i` 명령을 루트로 실행합니다.
  - USB 오디오 장치를 기본 오디오 장치로 만들려는 경우에는 장치를 연결하고 장치 링크를 확인하면 됩니다.

## USB 오디오 장치 문제 해결

드라이버가 연결되어 있고 볼륨이 크게 설정되어 있어도 USB 스피커에서 소리가 나지 않는 경우가 있습니다. 장치의 핫 플러그 기능을 사용해도 이 동작이 변경되지는 않습니다.

해결 방법은 USB 스피커 전원을 껐다가 켜는 것입니다.

### 오디오 장치 소유권 요점

오디오 장치로 작업할 경우 오디오 장치 소유권에 대한 다음 요점에 주의하십시오.

- USB 오디오 장치를 플러그인하고 콘솔에서 로그인한 경우에는 콘솔이 `/dev/*` 항목의 소유자입니다. 이 상황은 콘솔에 로그인해 있는 한 오디오 장치를 사용할 수 있음을 의미합니다.
- USB 오디오 장치를 플러그인했을 때 콘솔로 로그인하지 않은 경우에는 루트가 장치의 소유자가 됩니다. 그러나 콘솔에 로그인하여 USB 오디오 장치에 대한 액세스를 시도하면 장치 소유권이 콘솔로 변경됩니다. 자세한 내용은 `logindevperm(4)`을 참조하십시오.
- `rlogin` 명령을 사용하여 원격으로 로그인하고 USB 오디오 장치에 액세스하려고 하는 경우는 소유권이 변경되지 않습니다. 즉, 이를테면 권한이 없는 사용자가 다른 사람 소유의 마이크를 통해 대화를 들을 수 없습니다.

# cfgadm 명령을 사용하여 USB 장치 핫플러그 기능 사용(작업 맵)

작업	설명	수행 방법
USB 버스 정보를 표시합니다.	USB 장치 및 버스에 대한 정보를 표시합니다.	152 페이지 “USB 버스 정보를 표시하는 방법(cfgadm)”
USB 장치 구성을 해제합니다.	아직 시스템에 물리적으로 연결되어 있는 USB 장치를 논리적으로 구성 해제합니다.	153 페이지 “USB 장치를 구성 해제하는 방법”
USB 장치를 구성합니다.	이전에 구성 해제한 USB 장치를 구성합니다.	154 페이지 “USB 장치를 구성하는 방법”
USB 장치를 논리적으로 연결 해제합니다.	시스템에 물리적으로 가까운 위치에 있지 않은 경우 USB 장치를 논리적으로 연결 해제할 수 있습니다.	154 페이지 “논리적으로 USB 장치를 연결 해제하는 방법”
USB 장치를 논리적으로 연결합니다.	이전에 논리적으로 연결 해제 또는 구성 해제했던 USB 장치를 논리적으로 연결할 수 있습니다.	154 페이지 “논리적으로 USB 장치를 연결하는 방법”
USB 장치의 하위 트리를 연결 해제합니다.	허브 아래의 장치 계층(또는 트리)에 해당되는 USB 장치 하위 트리의 연결을 해제합니다.	155 페이지 “USB 장치 하위 트리를 논리적으로 연결 해제하는 방법”
USB 장치를 재설정합니다.	USB 장치를 재설정하여 장치를 논리적으로 제거했다가 다시 만듭니다.	155 페이지 “USB 장치를 재설정하는 방법”
다중 구성 USB 장치의 기본 구성을 변경합니다.	다중 구성 USB 장치의 기본 구성을 변경합니다.	156 페이지 “다중 구성 USB 장치의 기본 구성을 변경하는 방법”

## cfgadm 명령을 사용하여 USB 장치 핫플러그 기능 사용

cfgadm 명령을 사용하지 않고 실행 중인 시스템에서 USB 장치를 추가 및 제거할 수 있습니다. 또한 USB 장치를 물리적으로 제거하지 않고 **논리적으로** 핫플러그 기능을 사용할 수도 있습니다. 이 시나리오는 원격에서 작업하는 동안 작동하지 않는 USB 장치를 사용 안함으로 설정하거나 재설정하는 경우에 유용합니다. cfgadm 명령은 또한 제조업체 및 제품 정보를 포함하여 USB 장치 트리를 표시하는 방법도 제공합니다.

cfgadm 명령은 시스템에서 동적 재구성 작업을 진행할 수 있는 위치인 **첨부 지점**에 대한 정보를 표시합니다. 첨부 지점은 다음으로 구성됩니다.

- 점유자는 USB 장치와 같이 시스템에 구성할 수 있는 하드웨어 리소스를 나타냅니다.
- 콘센트는 USB 포트와 같이 점유자와 연결되는 위치입니다.

첨부 지점은 논리적 및 물리적 첨부 지점 ID(Ap\_Id)로 표현합니다. 물리적 Ap\_Id는 첨부 지점의 물리적인 경로 이름입니다. 논리적 Ap\_Id는 물리적 Ap\_Id를 대체하는, 사용자에게 친숙한 ID입니다. Ap\_Id에 대한 자세한 내용은 `cfgadm_usb(1M)`를 참조하십시오.

cfgadm 명령은 다음과 같은 USB 장치 상태 정보를 제공합니다.

콘센트 상태	설명
empty/unconfigured	장치가 물리적으로 연결되어 있지 않습니다.
disconnected/unconfigured	장치가 물리적으로는 아직 연결되어 있더라도 논리적으로 연결 해제되어 사용할 수 없습니다.
connected/unconfigured	장치가 논리적으로 연결되어 있지만 사용할 수 없습니다. 장치가 prtconf 출력에 표시됩니다.
connected/configured	장치가 연결되어 있고 사용할 수 있습니다.

다음 섹션에서는 `cfgadm` 명령으로 소프트웨어를 통해 USB 장치의 핫 플러그 기능을 사용하는 방법을 설명합니다. 관련 정보에 집중할 수 있도록 이 섹션의 모든 샘플 USB 장치 정보가 잘려 있습니다.

## ▼ USB 버스 정보를 표시하는 방법(cfgadm)

`prtconf` 명령을 사용하여 USB 구성 정보를 표시한 경우의 예는 136 페이지 “USB 장치 정보를 표시하는 방법”을 참조하십시오.

### 1 USB 버스 정보를 표시합니다.

예를 들면 다음과 같습니다.

```
% cfgadm
Ap_Id          Type          Receptacle  Occupant    Condition
usb0/4.5       usb-hub       connected   configured  ok
usb0/4.5.1     usb-device    connected   configured  ok
usb0/4.5.2     usb-printer   connected   configured  ok
usb0/4.5.3     usb-mouse     connected   configured  ok
usb0/4.5.4     usb-device    connected   configured  ok
usb0/4.5.5     usb-storage   connected   configured  ok
usb0/4.5.6     usb-communi   connected   configured  ok
usb0/4.5.7     unknown      empty       unconfigured ok
```

앞의 예에서 `usb0/4.5.1`은 첫번째 USB 제어기의 루트 허브 포트 4에 연결된 첫번째 레벨 외부 허브의 포트 5에 연결되어 있는 두번째 레벨 외부 허브의 포트 1에 연결된 장치를 식별합니다.



## 2 특정 USB 장치 정보를 표시합니다.

예를 들면 다음과 같습니다.

```
% cfgadm -l -s "cols=ap_id:info"
Ap_Id      Information
usb0/4.5.1 Mfg: Inside Out Networks Product: Edgeport/421 NConfigs: 1
Config: 0 : ...
usb0/4.5.2 Mfg: <undef> Product: <undef> NConfigs: 1 Config: 0 ...
usb0/4.5.3 Mfg: Mitsumi Product: Apple USB Mouse NConfigs: 1
Config: 0 ...
usb0/4.5.4 Mfg: NMB Product: NMB USB KB/PS2 M NConfigs: 1 Config: 0
usb0/4.5.5 Mfg: Hagiwara Sys-Com Product: SmartMedia R/W NConfigs: 1
Config: 0 : ...
usb0/4.5.6 Mfg: 3Com Inc. Product: U.S.Robotics 56000 Voice USB Modem
NConfigs: 2 ...
usb0/4.5.7
```

## ▼ USB 장치를 구성 해제하는 방법

아직 시스템에 물리적으로 연결되어 있는 USB 장치를 구성 해제할 수 있습니다. 이 경우 드라이버가 장치에 연결되지 않습니다. 장치를 구성 해제한 후에도 USB 장치는 prtconf 출력에 남습니다.

### 1 슈퍼유저로 로그인합니다.

### 2 USB 장치를 구성 해제합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -c unconfigure usb0/4.7
Unconfigure the device: /devices/pci@8,700000/usb@5,3/hub@4:4.7
This operation will suspend activity on the USB device
Continue (yes/no)? y
```

### 3 장치가 구성 해제되었는지 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm
Ap_Id      Type      Receptacle  Occupant    Condition
usb0/4.5   usb-hub   connected   configured  ok
usb0/4.5.1 usb-device connected   configured  ok
usb0/4.5.2 usb-printer connected   configured  ok
usb0/4.5.3 usb-mouse  connected   configured  ok
usb0/4.5.4 usb-device connected   configured  ok
usb0/4.5.5 usb-storage connected   configured  ok
usb0/4.5.6 usb-communi connected   configured  ok
usb0/4.5.7 unknown   empty       unconfigured ok
usb0/4.6   usb-storage connected   configured  ok
usb0/4.7   usb-storage connected   unconfigured ok
```

## ▼ USB 장치를 구성하는 방법

1 슈퍼유저로 로그인합니다.

2 USB 장치를 구성합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -c configure usb0/4.7
```

3 USB 장치가 구성되었는지 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm usb0/4.7
Ap_Id          Type          Receptacle  Occupant    Condition
usb0/4.7      usb-storage  connected   configured  ok
```

## ▼ 논리적으로 USB 장치를 연결 해제하는 방법

USB 장치를 시스템과 prtconf 출력에서 제거하고 싶지만, 물리적으로 시스템과 가까운 위치에 있지 않은 경우에는 논리적으로 USB 장치의 연결을 해제합니다. 장치는 아직 물리적으로 연결되어 있습니다. 그러나 논리적으로는 연결이 해제되어 사용할 수 없으며 시스템에 표시되지 않습니다.

1 슈퍼유저로 로그인합니다.

2 USB 장치를 연결 해제합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -c disconnect -y usb0/4.7
```

3 장치의 연결이 해제되었는지 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm usb0/4.7
Ap_Id          Type          Receptacle  Occupant    Condition
usb0/4.7      unknown      disconnected  unconfigured ok
```

## ▼ 논리적으로 USB 장치를 연결하는 방법

이 절차를 사용하여 이전에 논리적으로 연결 해제되거나 구성 해제된 USB 장치를 논리적으로 연결합니다.

1 슈퍼유저로 로그인합니다.

**2 USB 장치를 연결합니다.**

예를 들면 다음과 같습니다.

```
# cfgadm -c configure usb0/4.7
```

**3 장치가 연결되었는지 확인합니다.**

예를 들면 다음과 같습니다.

```
# cfgadm usb0/4.7
```

Ap_Id	Type	Receptacle	Occupant	Condition
usb0/4.7	usb-storage	connected	configured	ok

이제 장치를 사용할 수 있으며 시스템에도 표시됩니다.

## ▼ USB 장치 하위 트리를 논리적으로 연결 해제하는 방법

이 절차를 사용하여 허브 아래의 장치 계층(또는 트리)에 해당되는 USB 장치 하위 트리의 연결을 해제합니다.

**1 슈퍼유저로 로그인합니다.****2 USB 장치 하위 트리를 제거합니다.**

예를 들면 다음과 같습니다.

```
# cfgadm -c disconnect -y usb0/4
```

**3 USB 장치 하위 트리가 연결 해제되었는지 확인합니다.**

예를 들면 다음과 같습니다.

```
# cfgadm usb0/4
```

Ap_Id	Type	Receptacle	Occupant	Condition
usb0/4	unknown	disconnected	unconfigured	ok

## ▼ USB 장치를 재설정하는 방법

USB 장치가 오작동하는 경우에는 cfgadm 명령을 사용하여 장치를 재설정하면 장치를 제거했다가 다시 만들 수 있습니다.

**1 슈퍼유저로 로그인합니다.****2 장치를 사용하고 있지 않은지 확인합니다.****3 장치를 재설정합니다.**

예를 들면 다음과 같습니다.

```
# cfgadm -x usb_reset -y usb0/4.7
```

**4 장치 연결되었는지 확인합니다.**

예를 들면 다음과 같습니다.

```
# cfgadm usb0/4.7
Ap_Id          Type          Receptacle  Occupant    Condition
usb0/4.7      usb-storage  connected   configured  ok
```

**▼ 다중 구성 USB 장치의 기본 구성을 변경하는 방법**

다중 구성 USB 장치에서 작업하는 경우 다음 사항에 주의하십시오.

- USB 장치 구성은 장치가 운영 체제에 표시되는 방식을 정의합니다. 이 방법은 다른 cfgadm 섹션에서 설명한 시스템 장치 구성과 다릅니다.
- 일부 USB 장치는 여러 개의 구성을 지원하지만 한 번에는 한 구성만 활성화할 수 있습니다.
- 다중 구성 장치는 `cfgadm -lv` 출력을 검사하여 식별할 수 있습니다. `Nconfigs`가 1보다 큼니다.
- 기본 USB 구성이 구성 1입니다. 현재 구성이 `cfgadm -lv` 출력에 `Config`로 반영됩니다.
- 재부트, 핫 플러그 기능으로 제거, 장치 재구성 등을 수행해도 같은 포트에 다시 연결하기만 하면 기본 구성의 변경 사항이 유지됩니다.

**1 장치를 사용하고 있지 않은지 확인합니다.**

**2 기본 USB 구성을 변경합니다.**

예를 들면 다음과 같습니다.

```
# cfgadm -x usb_config -o config=2 usb0/4
Setting the device: /devices/pci@1f,0/usb@c,3:4
to USB configuration 2
This operation will suspend activity on the USB device
Continue (yes/no)? yes
```

**3 변경된 장치를 확인합니다.**

예를 들면 다음과 같습니다.

```
# cfgadm -lv usb0/4
Ap_Id Receptacle Occupant Condition Information When Type
      Busy Phys_Id
usb0/4 connected unconfigured ok          Mfg: Sun 2000
Product: USB-B0B0 aka Robotech
With 6 EPPS High Clk Mode NConfigs: 7 Config: 2 : EVAL Board Setup
unavailable
usb-device n /devices/pci@1f,0/usb@c,3:4
```

이제 `Config`:에 2가 표시되는 것을 확인합니다.

## InfiniBand 장치 사용(개요/작업)

---

이 장에서는 네트워크의 IB(InfiniBand) 장치에 대한 일반적인 개요와 이러한 장치를 사용하는 데 필요한 단계별 지침을 제공합니다.

이 장에서 다루는 개요 정보는 다음과 같습니다.

- 157 페이지 “InfiniBand 장치 개요”
- 161 페이지 “동적으로 IB 장치 재구성(cfgadm)”

IB 장치 사용과 관련된 절차에 대한 자세한 내용은 다음을 참조하십시오.

- 159 페이지 “동적으로 IB 장치 재구성(작업 맵)”
- 170 페이지 “InfiniBand 장치에서 uDAPL 응용 프로그램 인터페이스 사용”
- 172 페이지 “IPoIB 장치 관리(dladm)”
- 175 페이지 “IB 장치 모니터 및 문제 해결”

동적 재구성 및 핫플러그 기능 사용에 대한 자세한 내용은 6 장, “동적으로 장치 구성(작업)”을 참조하십시오.

### InfiniBand 장치 개요

IB(InfiniBand)는 교환 패브릭 기반의 새로운 I/O 기술입니다. I/O 장치를 호스트에 연결하고 호스트 간에 통신할 수 있도록 대역폭이 높고, 대기 시간이 짧은 상호 연결을 제공합니다. IB 장치는 Solaris IB 연결 드라이버를 통해 관리합니다.

다음 장치에 대한 지원은 Oracle Solaris 11에서 제공됩니다.

- IPoIB(IP over IB) 장치 - IB 연결을 통해 IP 패킷을 전송할 수 있는 기능으로 ibp(7D) 드라이버로 구현됩니다.
- SDP(Socket Direct Protocol) - IB를 통해 소켓을 지원합니다.
- RDSv3(Reliable Datagram Service version 3) 및 RDS(Reliable Datagram Service)
- NFSoRDMA(NFS over RDMA) - RDMA를 사용하여 IB를 통해 NFS 서비스를 제공합니다.

- iSER(iSCSI Extension for RDMA) – iSCSI 프로토콜로 RDMA 데이터 전송 기능을 제공합니다.
- uDAPL(User Direct Access Programming Language)
- OFUV(Open Fabric User Verb)
- EoIB(Ethernet over IB)

IB 연결 드라이버는 서비스용(이 설명서에서는 **통신 서비스**라고 함) Solaris IBDM(IB Device Manager)을 질의하여 IB 포트, HCA\_SVC 및 IB VPPA 장치를 열거합니다.

IB 분할 영역 링크는 데이터 링크의 새로운 **부분** 클래스를 나타내고 이러한 객체는 새로운 하위 명령인 `dladm`을 사용하여 관리됩니다. IB 물리적 링크의 최상위에서, 포트에서 각 `P_Key`당 하나씩 IB 분할 영역 링크를 만들 수 있습니다. 분할 영역 링크는 데이터 전송에 사용됩니다.

포트 장치는 HCA(Host Channel Adapter)의 지정된 `port#`에 통신 서비스를 바인딩합니다. VPPA 장치는 통신 서비스를 대신 `port#`, `P_key#` 결합으로 바인딩합니다. HCA\_SVC 장치는 지정된 HCA에 통신 서비스를 바인딩합니다. 포트 장치 및 HCA\_SVC 장치는 값이 0인 `P_key`(분할 영역 키)를 항상 사용합니다. 포트, HCA\_SVC 및 VPPA 장치는 HCA의 자식으로 `ib.conf` 파일을 통해 열거됩니다. 자세한 내용은 [ib\(7D\)](#)를 참조하십시오.

IOC 장치는 IB 연결 드라이버의 자식이며 I/O 장치의 일부입니다. 의사 장치는 IB 연결 드라이버의 자식으로, 열거할 고유한 구성 파일을 제공하는 다른 모든 장치를 참조합니다. 자세한 내용은 [ib\(4\)](#)를 참조하십시오.

다음 표에는 가능한 IB 장치 트리 경로 이름이 나열되어 있습니다.

IOC 장치	<code>/ib/ioc@1730000007F510C,173000007F50</code>
IB 의사 장치	<code>/ib/&lt;driver&gt;@&lt;unit-address&gt;</code>
IB VPPA 장치	<code>/pci@1f,2000/pci@1/pci15b3,5a44@0/ibport@&lt;port#&gt;,&lt;P_key&gt;,&lt;service&gt;</code>
IB HCA_SVC 장치	<code>/pci@1f,2000/pci@1/pci15bc,5a44@0/ibport@0,0,&lt;service&gt;</code>
IB 포트 장치	<code>/pci@1f,2000/pci@1/pci15b3,5a44@0/ibport@&lt;port#&gt;,0,&lt;service&gt;</code>
HCA	<code>/pci@1f,2000/pci@1/pci15b3,5a44@0</code>

IB HCA\_SVC 장치는 `port#` 및 `P_key`의 값이 0입니다.

앞의 표에서 IB 구성 요소는 다음과 같이 설명됩니다.

`<services>`            통신 서비스입니다. 예를 들어 `ipib`는 `ibd` 커널 클라이언트 드라이버에서 사용하는 통신 서비스입니다.

`<P_key>`                분할 영역 링크 키 값이 사용 중입니다.

<port> 포트 번호입니다.

<unit-address> driver.conf 파일에 지정된 이름을 기준으로 IB 커널 클라이언트 드라이버의 등록 정보를 참조합니다. 자세한 내용은 driver.conf(4)를 참조하십시오.

IB 진단 명령 및 유틸리티에 대한 자세한 내용은 175 페이지 “IB 장치 모니터 및 문제 해결”을 참조하십시오.

## InfiniBand 소프트웨어 패키지

IB 관련 소프트웨어 패키지는 다음과 같습니다.

- driver/infiniband/connectx – Mellanox ConnectX Family InfiniBand HCA 및 10GbE NIC 드라이버
- driver/infiniband/tavor – Tavor HCA 드라이버
- system/io/infiniband/ethernet-over-ib – Ethernet over InfiniBand를 구현하는 InfiniBand 장치 드라이버
- system/io/infiniband/ib-device-mgt-agent – InfiniBand Device Manager Agent
- system/io/infiniband/ib-sockets-direct – InfiniBand layered Sockets Direct Protocol
- system/io/infiniband/ip-over-ib – IPoIB(IP over InfiniBand) 프로토콜을 지원하는 네트워크 장치 드라이버
- system/io/infiniband/open-fabrics – Open Fabrics 커널 구성 요소
- system/io/infiniband/reliable-datagram-sockets-v3 – Reliable Datagram Socket(RDSv3)
- system/io/infiniband/reliable-datagram-sockets – Reliable Datagram Socket
- system/io/infiniband/rpc-over-rdma – InfiniBand RPC over RDMA 드라이버
- system/io/infiniband/udapl – UDAPL 라이브러리 및 명령
- system/io/infiniband – InfiniBand 프레임워크

## 동적으로 IB 장치 재구성(작업 맵)

작업	설명	수행 방법
IB 장치 정보를 표시합니다.	시스템의 IB 장치에 대한 정보를 표시합니다.	162 페이지 “IB 장치 정보 표시 방법”

작업	설명	수행 방법
포트 또는 VPPA 장치를 구성하거나 구성을 해제합니다.	다음 중 하나를 선택합니다.  포트 또는 VPPA 장치를 구성 해제합니다.  포트 또는 VPPA 장치를 구성합니다.	164 페이지 “IB 포트, HCA_SVC 또는 VPPA 장치를 구성 해제하는 방법”  164 페이지 “IB 포트, HCA_SVC 또는 VPPA 장치 구성 방법”
IB 의사 장치를 구성하거나 구성 해제합니다.	다음 중 하나를 선택합니다.  IB 의사 장치를 구성 해제합니다.  IB 의사 장치를 구성합니다.	165 페이지 “IB 의사 장치의 구성 해제 방법”  165 페이지 “IB 의사 장치 구성 방법”
HCA의 커널 IB 클라이언트를 표시합니다.	특히 HCA의 구성을 해제하려는 경우 HCA의 커널 IB 클라이언트에 대한 정보를 표시해야 할 수 있습니다.	166 페이지 “HCA의 커널 IB 클라이언트 표시 방법”
IB HCA 또는 EoIB 인터페이스를 구성 또는 구성 해제합니다.	다음 중 하나를 선택합니다.  HCA에 연결된 IB 장치를 구성 해제합니다.  EoIB 장치가 활성 상태인 경우 HCA를 동적으로 재구성합니다.  EoIB 인터페이스가 활성 상태이며 마지막이 아닌 IB HCA를 구성 해제합니다.  HCA에 연결된 IB 장치를 구성합니다.	166 페이지 “EoIB 장치가 활성화된 상태로 HCA를 동적으로 재구성하는 방법”  167 페이지 “작업 중 제거 후 EoIB 인터페이스를 재구성 및 복원하는 방법”  168 페이지 “IB HCA 구성”
IB P_key 테이블을 업데이트합니다.	HCA 포트의 P_key 테이블 정보가 업데이트되면 내부 P_key 데이터베이스가 업데이트되도록 IBTF 및 IBDM에 알려야 합니다.	168 페이지 “IB P_key 테이블을 업데이트하는 방법”
IB 통신 서비스를 표시합니다.	현재 IBTF에서 사용 중인 IB 통신 서비스를 표시합니다.	169 페이지 “IB 통신 서비스 표시 방법”



작업	설명	수행 방법
VPPA 통신 서비스를 추가하거나 제거합니다.	다음 중 하나를 선택합니다.  VPPA 통신 서비스를 추가합니다.  VPPA 통신 서비스를 제거합니다.	169 페이지 “VPPA 통신 서비스 추가 방법”  170 페이지 “기존 IB 포트, HCA_SVC 또는 VPPA 통신 서비스 제거 방법”

## 동적으로 IB 장치 재구성(cfgadm)

cfgadm CLI만 사용하여 실행 중인 시스템에서 IB 장치를 구성하거나 구성을 해제할 수 있습니다. 또한 이 명령은 IB 패브릭 표시, 통신 서비스 관리 및 P\_key 테이블 데이터베이스 업데이트 방법을 제공합니다. 자세한 내용은 `cfgadm_ib(1M)`를 참조하십시오.

cfgadm CLI는 호스트에 표시되는 전체 IB 패브릭의 동적 재구성(이 설명서에서는 DR이라고 함)을 관리합니다. cfgadm 작업은 포트, VPPA, HCA\_SVC, IOC 및 의사 장치와 같은 모든 IB 장치에서 지원됩니다.

cfgadm 명령은 DR 작업이 발생할 수 있는 시스템상의 위치인 첨부 지점(Ap\_Id)에 대한 정보를 표시합니다. cfgadm이 지원하는 Ap\_Id에 대한 자세한 내용은 `cfgadm_ib.1M`를 참조하십시오. 모든 IB Ap\_Id는 `connected`로 표시됩니다.

cfgadm 명령은 다음 IB 장치 상태 정보를 제공합니다.

콘센트 상태	설명
connected/configured/ok	장치가 연결되어 있고 사용할 수 있습니다. devinfo 노드가 있습니다.
connected/unconfigured/unknown	장치를 사용할 수 없으며 이 장치에 대한 devinfo 노드 또는 장치 드라이버가 있습니다. 또는 장치가 ib 연결 드라이버에 사용되도록 구성되지 않았습니. 장치가 IB 장치 관리자에 알려져 있을 수도 있습니다.

이 섹션의 모든 절차를 수행하려면 사용자 계정에는 일반적으로 부여되지 않는 관리 권한이 필요합니다. 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.

다음 섹션에서는 cfgadm 명령을 사용하여 IB 장치를 동적으로 재구성(DR)하는 방법에 대해 설명합니다. 이 섹션의 모든 샘플 IB 장치 정보는 관련 정보에 중점을 두기 위해 잘렸습니다.

## IB 장치 구성 해제 고려 사항

HCA의 실제 DR(동적 재구성)은 IB cfgadm 플러그인의 범위를 벗어납니다. 그러나 기본 버스의 플러그인을 사용하여 HCA DR을 수행할 수 있습니다. 예를 들어 PCI 기반 HCA는 `cfgadm_pci` 명령을 사용할 수 있습니다. 자세한 내용은 `cfgadm_pci(1M)`를 참조하십시오.

주 - RDSv3은 HCA 구성 해제를 지원하지 않습니다. 시스템에 DR 시 설치된 RDSv3 드라이버가 있으면 아래 표시된 것처럼 HCA 구성 해제에 실패합니다.

```
# cfgadm -c unconfigure ib::rdsv3,0
This operation will suspend activity on the IB device
Continue (yes/no)? yes
cfgadm: Hardware specific failure: unconfigure operation failed ap_id: /devices/ib:fabric::rdsv3,0

# cfgadm -c unconfigure PCI-EM0
cfgadm: Component system is busy, try again: unconfigure failed
```

### 해결 방법:

생산 시스템에서는 HCA DR 작동 전에 RDSv3을 제거하고 시스템을 재부트해야 합니다.

```
# rem_drv rdsv3
Device busy
Cannot unload module: rdsv3
Will be unloaded upon reboot.

# init 6
```

## ▼ IB 장치 정보 표시 방법

`prtconf` 명령을 사용하여 IB 장치에 대한 일반 정보를 표시할 수 있습니다. 예:

```
$ prtconf
.
.
.
  ib, instance #0
    rpcib, instance #0
    rdsib, instance #0
    daplt, instance #0
    rdsv3, instance #0
    sdplib, instance #0
    eibnx, instance #0
    sol_umad, instance #0
    sol_uverbs, instance #0
    iser, instance #0
.
.
.
  pci15b3,673c, instance #0
    ibport, instance #0
    ibport, instance #1
```

위 예에서 pci15b3,673c는 IB HCA입니다.

특정 IB 장치 정보를 표시하려면 다음 단계를 사용합니다.

### 1 관리자로 전환합니다.

### 2 IB 패브릭 정보를 표시합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -al
Ap_Id          Type          Receptacle    Occupant    Condition
ib             IB-Fabric     connected     configured  ok
hca:21280001A0A478  IB-HCA       connected     configured  ok
ib::21280001A0A47A,0,ipib  IB-PORT     connected     configured  ok
ib::21280001A0A479,0,ipib  IB-PORT     connected     configured  ok
ib::1730000008070,0,hnfs  IB-HCA_SVC  connected     configured  ok
ib::daplt,0       IB-PSEUDO    connected     configured  ok
ib::eibnx,0       IB-PSEUDO    connected     configured  ok
ib::iser,0        IB-PSEUDO    connected     configured  ok
ib::rdsib,0       IB-PSEUDO    connected     configured  ok
ib::rdsv3,0       IB-PSEUDO    connected     configured  ok
ib::rpcib,0       IB-PSEUDO    connected     configured  ok
ib::sdpib,0       IB-PSEUDO    connected     configured  ok
ib::sol_umad,0    IB-PSEUDO    connected     configured  ok
ib::sol_uverbs,0  IB-PSEUDO    connected     configured  ok
```

위의 출력 예에서 구성 요소는 다음과 같습니다.

Ap_Id ib::21280001A0A47A,0,ipib	포트 GUID에 연결되어 있고 ipib 서비스로 바인딩되는 IB 포트 장치를 식별합니다.
Ap_ID ib::sdpib,0	의사 장치를 식별합니다.
Ap_ID hca:21280001A0A478	HCA 장치를 식별합니다.
Ap_ID ib::1730000008070,0,hnfs	hnfs 서비스에 바인딩된 IB HCA_SVC 장치를 식별합니다.
Ap_Id ib::ibgen,0	의사 장치를 식별합니다.

### 3 특정 IB 장치 정보를 표시합니다.

예를 들어 IB 포트 장치의 경우 다음과 같이 표시됩니다.

```
# cfgadm -al -s "cols=ap_id:info" ib::21280001A0A47A,0,ipib
Ap_Id          Information
ib::21280001A0A47A,0,ipib  ipib
```

예를 들어, IB HCA 장치의 경우 다음과 같이 표시됩니다.

```
# cfgadm -al -s "cols=ap_id:info" hca::1730000008070
Ap_Id          Information
hca::1730000008070  VID: 0x15b3, PID: 0x5a44, #ports: 0x2,
port1 GUID: 0x1730000008071, port2 GUID: 0x1730000008072
```

이전 출력에서 포트 및 GUID 수를 표시합니다.

## ▼ IB 포트, HCA\_SVC 또는 VPPA 장치를 구성 해제하는 방법

시스템에서 IB 포트, HCA\_SVC 또는 VPPA 장치를 제거하려는 경우 다음 단계를 사용합니다.

아래 예는 IB 포트 장치를 구성 해제하는 방법에 대해 설명합니다. VPPA 및 HCA\_SVC 장치를 구성하는 경우에도 같은 절차를 사용할 수 있습니다.

- 1 관리자로 전환합니다.
- 2 가상 IB 포트 장치를 구성 해제합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -c unconfigure ib::1730000007F51,*0*,ipib
Unconfigure the device: /devices/ib:fabric::1730000007F51,*0*,ipib
This operation will suspend activity on the IB device
Continue (yes/no)? Y
```

- 3 장치의 연결이 해제되었는지 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -a ib::1730000007F51,*0*,ipib
Ap_Id                Type      Receptacle Occupant    Condition
ib::1730000007F51,*0*,ipib IB-VPPA   connected  unconfigured unknown
```

## ▼ IB 포트, HCA\_SVC 또는 VPPA 장치 구성 방법

시스템에 IB 포트, HCA\_SVC 또는 VPPA 장치를 구성하려는 경우 다음 단계를 사용합니다.

아래 예는 VPPA 장치를 구성하는 방법에 대해 설명합니다. IB 포트 및 HCA\_SVC 장치를 구성하는 경우에도 유사한 단계를 사용할 수 있습니다.

- 1 관리자로 전환합니다.
- 2 가상 IB 포트 장치를 구성합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -c configure ib::1730000007F51,*0*,ipib
```

- 3 장치가 연결되었는지 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -a ib::1730000007F51,*0*,ipib
Ap_Id                Type      Receptacle Occupant    Condition
ib::1730000007F51,*0*,ipib IB-VPPA   connected  configured  ok
```

주 - IB 포트 및 HCA\_SVC 장치의 cfgadm 기반 구성 및 구성 해제 작업은 IB VPPA 장치에 대한 이전 예와 유사합니다.

## ▼ IB 의사 장치의 구성 해제 방법

시스템에서 IB 의사 장치를 제거하려면 다음 단계를 사용합니다.

- 1 관리자로 전환합니다.

- 2 IB 의사 장치를 구성 해제합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -c unconfigure ib::ibgen,0
Unconfigure the device: /devices/ib:fabric::ibgen,0
This operation will suspend activity on the IB device
Continue (yes/no)? Y
```

- 3 장치의 연결이 해제되었는지 확인합니다.

```
# cfgadm -a ib::ibgen,0
Ap_Id                Type      Receptacle Occupant  Condition
ib::ibgen,0         IB-PSEUDO connected  unconfigured unknown
```

## ▼ IB 의사 장치 구성 방법

IB 의사 장치를 구성하려면 다음 단계를 사용합니다.

- 1 관리자로 전환합니다.

- 2 IB 의사 장치를 구성합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -yc configure ib::ibgen,0
```

- 3 장치가 연결되었는지 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -a ib::ibgen,0
Ap_Id                Type      Receptacle Occupant  Condition
ib::ibgen,0         IB-PSEUDO connected  configured ok
```

## ▼ HCA의 커널 IB 클라이언트 표시 방법

다음 IB cfgadm 플러그인 명령을 호출하여 이 HCA를 사용하는 커널 IB 클라이언트를 나열할 수 있습니다. 커널 IB 클라이언트가 다른 HCA를 사용하는 경우 마지막 열에 "yes"가 표시됩니다. HCA를 사용하지 않는 IB 관리자 및 커널 클라이언트는 Ap\_Id가 "-"로 표시됩니다.

### ● HCA의 커널 IB 클라이언트를 표시합니다.

예를 들면 다음과 같습니다.

```
$ cfgadm -x list_clients hca:173000007F50
Ap_Id          IB Client          Alternate HCA
ib::173000007F51D0    ibgen              no
ib::173000007F51D1    ibgen              no
ib::173000007F51,0,ipib  ibd                no
ib::ibgen,0          ibgen              no
-                  ibdm               no
-                  ibmf               no
-                  nfs/ib             no
```

## ▼ EoIB 장치가 활성화된 상태로 HCA를 동적으로 재구성하는 방법

시스템에서 DR(동적 재구성) 작업을 시도했으며 활성화 EoIB 인터페이스에서 Connect-X 계열 IB HCA가 사용 중인 경우에는(예: EoIB 데이터 링크가 연결되었거나 HCA에 있는 데이터 링크를 통해 VNIC를 만든 경우) 다음 조건 중 해당되는 것이 있으면 DR 작업이 성공합니다.

- 시스템에 있는 유일한 Connect-X 계열 HCA인 경우
- 시스템에 여러 개의 Connect-X HCA가 있지만 이 어댑터가 마지막으로 구성 해제할 어댑터인 경우

여기에 해당되지 않는 경우에는 활성화 EoIB 인터페이스가 있는 IB HCA에서의 DR(동적 재구성) 작업이 실패하고 cfgadm 오류 메시지가 표시됩니다.

DR 작업이 실패한 경우에는 활성화 EoIB 인터페이스의 연결을 해제하고 데이터 링크에서 VNIC를 모두 삭제한 다음 DR 작업을 재시도해야 합니다. 또한 그런 상황에서 DR 작업 후에 Connect-X 계열 HCA를 같은 슬롯에서 교체하고 cfgadm 또는 hotplug 명령을 사용하여 다시 구성한 경우에는, 이전에 연결 해제한 EoIB 데이터 링크를 다시 연결하고 이전에 삭제한 VNIC를 모두 다시 만들어야 합니다.

EoIB 인터페이스가 작동하려면 Sun Network QDR Infiniband Gateway Switch의 해당 구성 파일에 HCA 포트 GUID가 나열되어 있어야 합니다. 따라서 작업 중 제거 후 다시 삽입하기 전에 Gateway Switch 구성 파일에서 새로운 HCA 포트 GUID 정보를

업데이트해야 합니다. 구성을 업데이트하는 방법에 대한 자세한 내용은 **Sun Network QDR Infiniband Gateway Switch Administration Guide**를 참조하십시오.

1 여러 IB HCA가 있는 시스템에 관리자로 로그인합니다.

2 EoIB 데이터 링크와 연관된 첨부 지점의 구성 해제를 시도합니다.

```
# cfgadm -c unconfigure PCI-EM0
cfgadm: Component system is busy, try again: unconfigure failed
```

3 EoIB 인터페이스가 구성 해제 작업을 차단하는지 확인합니다.

```
# dmesg | tail | grep 'failing HCA detach'
Aug 23 12:37:20 eoib: [ID 530795 kern.warning] WARNING: eoib0 still in use,
failing HCA detach
```

4 IP 인터페이스가 있는 eoib0이 사용 중인지 확인합니다.

```
# ipadm show-if eoib0
ipadm: cannot get information for interface(s): No such interface
```

5 eoib0에 IP 인터페이스가 없는 경우에는 eoib0을 사용하게 만드는 VNIC가 있는지 확인합니다.

```
# dladm show-vnic
LINK          OVER          SPEED  MACADDRESS    MACADDRTYPE  VID
evnic0        eoib0         10000  2:8:20:e5:56:99  random        0
```

6 eoib0을 통해 VNIC를 삭제합니다.

```
# dladm delete-vnic evnic0
```

7 구성 해제 작업을 다시 시도합니다.

```
# cfgadm -c unconfigure PCI-EM0
```

## ▼ 작업 중 제거 후 EoIB 인터페이스를 재구성 및 복원하는 방법

1 Sun Network QDR Gateway Switch에서 관리자로 로그인합니다.

2 Sun Network QDR Gateway Switch의 BXM 구성 파일을 편집하여 새 HCA의 포트 GUID로 이전 HCA 포트 GUID의 모든 항목을 바꿉니다.

3 BXM을 다시 시작합니다.

4 Oracle Solaris 시스템에서 관리자로 로그인합니다.

5 Solaris 시스템에 새 HCA를 작업 중 삽입하고 `cfgadm` 명령을 사용하여 EoIB 데이터 링크와 원래 연관된 첨부 지점을 재구성합니다.

6 원래 EoIB 데이터 링크가 복원되었는지 확인합니다.

```
# dladm show-link | grep eoib0
eoib0          phys      1500    unknown    --
```

7 이전 구성 해제 작업 중에 명시적으로 삭제해야 했던 VNIC가 있으면 지금 다시 만듭니다.

```
# dladm create-vnic -l eoib0 evnic0
```

8 `eoib0`을 통해 VNIC가 있는지 확인합니다.

```
# dladm show-vnic
LINK      OVER      SPEED  MACADDRESS      MACADDRTYPE      VID
evnic0    eoib0    10000  2:8:20:e5:56:99  random           0
```

9 이전 구성 해제 작업 중에 `eoib0`에서 명시적으로 삭제해야 했던 IP 주소가 있으면 `ipadm` 명령을 사용하여 다시 만듭니다.

## IB HCA 구성

버스 관련 `cfgadm` 플러그인을 호출하여 HCA를 구성합니다. 정확한 세부 정보는 이 장에서 제공되지 않습니다.

### ▼ IBP\_key 테이블을 업데이트하는 방법

HCA 포트의 `P_key` 테이블 정보가 변경되면 예를 들어, 추가 `P_key`가 사용으로 설정 또는 사용 안함으로 설정된 경우 내부 `P_key` 데이터베이스가 업데이트되도록 IBTF(InfiniBand Transport Framework) 및 IBDM에 알려야 합니다. `cfgadm` 명령을 사용하여 IBTF 및 IBDM의 `P_key` 데이터베이스를 업데이트할 수 있습니다. 자세한 내용은 [ibtl\(7D\)](#) 및 [ibdm\(7D\)](#)을 참조하십시오.

1 관리자로 전환합니다.

2 `P_key` 테이블을 업데이트합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -x update_pkey_tbls -y ib
```



## ▼ IB 통신 서비스 표시 방법

현재 IBTF에서 사용 중인 통신 서비스를 표시하려면 다음 단계를 사용합니다.

- 1 관리자로 전환합니다.
- 2 IB 통신 서비스를 표시합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -x list_services ib
Port communication services:
    srp
VPPA communication services:
    ibd
HCA_SVC communication services:
    hnfs
```

## ▼ VPPA 통신 서비스 추가 방법

새 VPPA 통신 서비스를 추가하려면 다음 단계를 사용합니다.

유사한 단계를 사용하여 새 HCA\_SVC 또는 포트 통신 서비스를 추가할 수 있습니다.

- 1 관리자로 전환합니다.
- 2 새 VPPA 통신 서비스를 추가합니다.
- 3 새 서비스가 추가되었는지 확인합니다.

예를 들면 다음과 같습니다.

```
# cfgadm -x list_services ib
Port communication services:
    srp
VPPA communication services:
    ibd
    new
HCA_SVC communication services:
    nfs_service
```

## ▼ 기존 IB 포트, HCA\_SVC 또는 VPPA 통신 서비스 제거 방법

기존 IB 포트, HCA\_SVC 또는 VPPA 통신 서비스를 삭제하려면 다음 단계를 사용합니다.

- 1 관리자로 전환합니다.
- 2 VPPA 통신 서비스를 제거합니다.  
예를 들면 다음과 같습니다.
- 3 통신 서비스가 제거되었는지 확인합니다.  
예를 들면 다음과 같습니다.

```
# cfgadm -o comm=vppa,service=new -x delete_service ib
```

```
# cfgadm -x list_services ib
Port communication services:
    srp
VPPA communication services:
    ibd
HCA_SVC communication services:
    hnfs
```

## InfiniBand 장치에서 uDAPL 응용 프로그램 인터페이스 사용

uDAPL(User Direct Access Programming Library)은 InfiniBand와 같은 RDMA(Remote Direct Memory Access) 가능 상호 연결을 통해 데이터 센터 응용 프로그램 데이터 메시징 성능, 확장성 및 안정성을 촉진하는 표준 API입니다. uDAPL 인터페이스는 DAT 공동 커뮤니티에 의해 정의됩니다. DAT 공동 커뮤니티에 대한 자세한 내용을 보려면 다음 사이트로 이동하십시오.

<http://www.datcollaborative.org>

Oracle Solaris 릴리스에서는 다음 uDAPL 기능을 제공합니다.

- 표준 DAT 레지스트리 라이브러리인 libdat. 자세한 내용은 libdat(3LIB)를 참조하십시오.
- 표준 서비스 공급자 등록 파일인 dat.conf. 자세한 내용은 dat.conf(4)를 참조하십시오.
- 각 공급자가 service\_provider.conf 파일에서 고유한 uDAPL 라이브러리 경로, 버전 번호 등을 지정하도록 여러 서비스 공급자 지원. 자세한 내용은 service\_provider.conf(4)를 참조하십시오.
- dat.conf를 구성하는 관리 도구인 datadm 명령. 자세한 내용은 datadm(1M)을 참조하십시오.

- 잠긴 물리적 메모리 크기를 규정하는 새 리소스 제어 등록 정보인 `project.max-device-locked-memory`
- IPv4의 ARP 및 IPv6의 인접 라우터 검색과 같은 IP 기반구조를 주소 확인에 이용하는 IPv4 또는 IPv6 주소를 사용하는 명명 체계. Solaris uDAPL 인터페이스 어댑터는 IPoIB 장치 인스턴스에 직접 매핑됩니다.
- DAT 공동 커뮤니티에서 사용하는 표준 주소 변환 체계를 지원
- `dat.conf` 등록 파일에 자동으로 등록되는 `tavor` 또는 `hermon Host Channel Adapter`를 지원할 uDAPL 서비스 공급자 라이브러리 자세한 내용은 `tavor(7D)` 또는 `hermon(7D)`을 참조하십시오.
- SPARC 플랫폼과 x86 플랫폼을 모두 지원

## ▼ uDAPL을 사용으로 설정하는 방법

- 1 관리자로 전환합니다.
- 2 다음 패키지가 설치되었는지 확인합니다. 또는 필요한 경우 다음 패키지를 설치합니다.
  - `driver/infiniband/connectx` - InfiniBand 프레임워크
  - `driver/infiniband/` - HCA 드라이버
  - `system/io/infiniband/ip-over-ib` - IP over InfiniBand
  - `system/io/infiniband/udapl` - 서비스 공급자 패키지
- 3 IPoIB 인터페이스를 만듭니다.

예를 들면 다음과 같습니다.

```
# ipadm create-ip ibd1
# ipadm create-addr -T static -a 192.168.0.1/24 ibd1/ipv4
# datadm -a /usr/share/dat/ABCudaplt.conf
```

## DAT 정적 레지스트리 업데이트

`datadm` 명령을 사용하여 DAT 정적 레지스트리인 `dat.conf` 파일을 유지 관리할 수 있습니다. 이 파일에 대한 자세한 내용은 `dat.conf(4)`를 참조하십시오.

또한 `datadm` 명령을 사용하여 `dat.conf` 파일에 서비스 공급자를 등록하거나 등록 해제할 수 있습니다. 자세한 내용은 `datadm(1M)`을 참조하십시오.

IPoIB 인터페이스 어댑터를 추가 또는 제거한 경우 `datadm` 명령을 실행하여 현재 시스템 상태가 적용되도록 `dat.conf` 파일을 업데이트합니다. 현재 설치된 모든 서비스 공급자의 새로운 인터페이스 어댑터 세트가 재생성됩니다.

## ▼ DAT 정적 레지스트리 업데이트 방법

- 1 관리자로 전환합니다.
- 2 시스템에서 IBoIP 인터페이스 어댑터를 추가 또는 제거한 후에 DAT 정적 레지스트리를 업데이트합니다.  
# `datadm -u`
- 3 업데이트된 DAT 정적 레지스트리를 표시합니다.  
# `datadm`

## ▼ DAT 정적 레지스트리에 서비스 공급자를 등록하는 방법

- 1 관리자로 전환합니다.
- 2 Mellanox Host Channel Adapter에 대한 공급업체의 서비스 공급자를 추가한 후 DAT 정적 레지스트리를 업데이트합니다.  
# `datadm -a /usr/share/dat/ABCudaplt.conf`
- 3 업데이트된 DAT 정적 레지스트리를 표시합니다.  
# `datadm -v`

## ▼ DAT 정적 레지스트리에서 서비스 공급자의 등록을 해제하는 방법

- 1 관리자로 전환합니다.
- 2 시스템에서 Mellanox Host Channel Adapter에 대한 공급업체의 서비스 공급자를 제거한 다음 DAT 정적 레지스트리를 업데이트합니다.  
# `datadm -r /usr/share/dat/ABCudaplt.conf`
- 3 업데이트된 DAT 정적 레지스트리를 표시합니다.  
# `datadm -v`

# IPoIB 장치 관리(dladm)

기본적으로 HCA당 포트마다 물리적 데이터 링크가 한 개 만들어집니다. 이러한 물리적 링크는 관리 및 관찰 가능 데이터 포인터로 사용할 수 있습니다. NIC를 통해 VNIC를 만드는 것과 유사하게 물리적 데이터 링크를 통해 IB 분할 영역 링크를 만들 수 있습니다. 물리적 데이터 링크는 데이터 전송에는 사용되지 않으므로 이러한 링크에서 IB 주소를 측정 및 할당할 수 없습니다. 데이터는 분할 영역 데이터 링크에서 전송됩니다.

네트워크 구성 요소에 대한 자세한 내용은 **Oracle Solaris 관리: IP 서비스를 참조하십시오.**

## ▼ 물리적 데이터 링크 정보를 표시하는 방법

dladm show-phys 명령을 사용하여 시스템에 대한 물리적 데이터 링크 정보를 표시합니다. 물리적 링크 상태는 IB HCA 포트 상태에 해당합니다.

- 1 관리자로 전환합니다.
- 2 물리적 데이터 링크 정보를 표시합니다.

예를 들어 `ibp0` — `ibp3` 데이터 링크에 대한 정보를 표시합니다.

```
# dladm show-phys
LINK      MEDIA      STATE      SPEED      DUPLEX      DEVICE
ibp0      Infiniband up          8000      unknown    ibp0
ibp1      Infiniband down        8000      unknown    ibp1
ibp2      Infiniband down        8000      unknown    ibp2
ibp3      Infiniband up          8000      unknown    ibp3
```

예를 들어 `show-ib` 하위 명령을 사용하여 물리적 링크, 포트 GUID, port# HCA GUID 및 명령 실행 시 포트에 있는 P\_Key만 표시합니다.

```
# dladm show-ib
LINK      HCAGUID      PORTGUID      PORT      STATE      P_Key
ibp0      2C9020040041C 2C9020040041D 1          up         FFFF, 8001
ibp1      2C9020040041C 2C9020040041E 2          down      FFFF
ibp2      3BA0001004E14 3BA0001004E15 1          down      FFFF
ibp3      3BA0001004E14 3BA0001004E16 2          up         FFFF, 8001
```

## ▼ IB 분할 영역 링크를 만드는 방법

IB 물리적 링크의 최상위에서, 포트의 P\_Key당 하나씩 IB 분할 영역 데이터 링크를 만들 수 있습니다. 분할 영역 데이터 링크는 데이터 전송에 사용됩니다.

- 1 관리자로 전환합니다.
- 2 새 IB 분할 영역 링크를 만듭니다.

예를 들어 P\_Key `0x8001`의 IB 분할 영역 링크는 `ibp0` 물리적 데이터 링크의 맨 위에 만들어집니다.

```
# dladm create-part -l ibp0 -P 0x8001 p8001.ibp0
```

해당 포트가 실행 중이고, 이 포트에 P\_Key가 있으며, IPoIB가 성공적으로 초기화되면 위의 명령이 문제 없이 실행됩니다.

예를 들어 P\_Key 0x9000의 IB 분할 영역 링크가 ibp2의 맨 위에 만들어집니다.

```
# dladm create-part -f -l ibp2 -P 0x9000 p9000.ibp2
```

주 -P\_Key 0x9000이 포트에 대해 구성되지 않았습니다. 강제 옵션(-f)을 사용하여 P\_Key가 없거나 포트가 down 상태이더라도 IB 분할 영역 링크를 만들 수 있습니다. 링크 상태가 down으로 표시됩니다. P\_Key가 포트에 추가되고 포트가 활성화되면 링크 상태가 up으로 전환됩니다.

### 3 IB 분할 영역 링크 정보가 표시됩니다.

예를 들면 다음과 같습니다.

```
# dladm show-part
LINK          P_Key      OVER      STATE      FLAGS
p8000.ibp0    8001      ibp0      unknown    ----
p9000.ibp2    9000      ibp2      unknown    f---
```

링크가 연결된 후 링크를 사용할 준비가 되면 IB 분할 영역 링크 상태가 up이 됩니다. 다음 조건에서 분할 영역 링크 상태는 down입니다.

- HCA 포트가 종료된 경우
- P\_Key가 없는 경우
- 브로드캐스트 그룹이 없는 경우

### 4 IP 주소를 연결하여 IB 분할 영역 링크에 할당합니다.

예를 들면 다음과 같습니다.

```
# ipadm create-ip p9000.ibp2
# ipadm create-addr -T static -a 1.1.1.1 p9000.ibp2/ipv4
# ipadm show-addr
```

## ▼ IB 분할 영역 링크 정보를 표시하는 방법

### 1 관리자로 전환합니다.

### 2 IB 분할 영역 링크 정보가 표시됩니다.

예를 들면 다음과 같습니다.

```
# dladm show-part
LINK          P_Key      OVER      STATE      FLAGS
p8000.ibp0    8001      ibp0      unknown    ----
p9000.ibp2    9000      ibp2      unknown    f---
```

## ▼ IB 분할 영역 링크를 제거하는 방법

- 1 관리자로 전환합니다.
- 2 IB 분할 영역 링크를 제거합니다.  
이 예에서는 분할 영역 링크 p8001.ibp0이 제거됩니다.  
`# dladm delete-part p8001.ibp0`
- 3 분할 영역 링크 정보를 검토합니다.

```
# dldam show-part
LINK          P_Key      OVER      STATE      FLAGS
p9000.ibp2    9000       ibp2      down       f---
```

## IB 장치 모니터 및 문제 해결

Oracle Solaris 11 릴리스에서 새로운 명령 및 유틸리티를 사용하여 IB 패브릭을 더욱 효과적으로 관리할 수 있습니다. 이러한 명령은 `system/io/infiniband/open-fabrics` 패키지에 포함되어 있으며 `open-fabrics` 패키지를 설치하면 매뉴얼 페이지가 자동으로 설치됩니다. 예를 들면 다음과 같습니다.

```
% man rping
Reformatting page. Please Wait... done

librdmacm                                RPING(1)

NAME
    rping - RDMA CM connection and RDMA ping-pong test.

SYNOPSIS
    rping -s [-v] [-V] [-d] [-P] [-a address] [-p port]
           [-C message_count] [-S message_size]
    rping -c [-v] [-V] [-d] -a address [-p port]
           [-C message_count] [-S message_size]
.
.
.
```

다음 새 명령 및 유틸리티를 통해 IB 장치 나열 및 질의, IB 패브릭 문제 진단 및 해결 그리고 IB 성능 측정을 수행할 수 있습니다.

표 9-1 일반적인 IB 모니터링 명령

명령	설명
<code>ibv_asyncwatch</code>	Monitors InfiniBand 비동기 이벤트를 모니터합니다.
<code>ibv_devices</code> or <code>ibv_devinfo</code>	InfiniBand 장치 또는 장치 정보를 나열합니다.

표 9-1 일반적인 IB 모니터링 명령 (계속)	
명령	설명
ibv_rc_pingpong, ibv_srq_pingpong 또는 ibv_ud_pingpong	RC 연결, SRQ 또는 UD 연결을 사용하여 노드 간 연결을 테스트합니다.
mckey	RDMA CM 멀티캐스트 설정 및 간단한 데이터 전송을 테스트합니다.
rping	RDMA CM 연결을 테스트하고 RDMA ping-pong을 시도합니다.
ucmatose	RDMA CM 연결을 테스트하고 간단한 ping-pong을 시도합니다.
udaddy	RDMA CM 데이터그램 설정을 테스트하고 간단한 ping-pong을 시도합니다.

표 9-2 일반적인 IB 성능 테스트 명령

명령	설명
rdma_bw 또는 rdma_lat	스트리밍 대역폭 및 대기 시간에 대한 RDMA 쓰기 트랜잭션을 테스트합니다.
ib_read_bw 또는 ib_read_lat	대역폭 및 대기 시간에 대한 RDMA 읽기 트랜잭션을 테스트합니다.
ib_send_bw 또는 ib_send_lat	대역폭 및 대기 시간에 대한 RDMA 보내기 트랜잭션을 테스트합니다.
ib_write_bw 또는 ib_write_bw_postlist	한 번에 한 I/O 요청을 표시하는 대역폭 또는 I/O 요청 목록을 표시하는 게시 목록 대역폭에 대한 RDMA 쓰기 트랜잭션을 테스트합니다.
ib_write_lat	대기 시간에 대한 RDMA 쓰기 트랜잭션을 테스트합니다.
ib_clock_test	시스템 클록의 정확도를 테스트합니다.
qperf	소켓 및 RDMA 성능을 측정합니다.

표 9-3 RDS 모니터 및 테스트 도구

명령	설명
rds-info	RDS 커널 모듈 정보를 표시합니다.
rds-ping	RDS를 통한 원격 노드에 도달할 수 있는지 확인합니다
rds-stress	RDS 소켓을 통해 프로세스 간에 메시지를 보냅니다.



표 9-4 패브릭 진단 도구

명령	설명
ibdiagnet	전체 패브릭에 대해 진단 검사를 수행합니다.
ibaddr	InfiniBand 주소를 질의합니다.
ibnetdiscover	원격 InfiniBand 토폴로지를 검색합니다.
ibping	IB 노드 간의 연결에 대한 유효성을 검사합니다.
ibportstate	IB 포트의 물리적 포트 상태 및 링크 속도를 질의합니다.
ibroute	InfiniBand 스위치 전달 테이블을 표시합니다.
ibstat 또는 ibsysstat	InfiniBand 장치의 상태 또는 IB 주소에서 시스템 상태를 질의합니다.
ibtracert	IB 경로를 추적합니다.
perfquery 또는 saquery	IB 포트 카운터 또는 sIB 서브넷 관리 속성을 질의합니다.
sminfo	IB SMIInfo 속성을 질의합니다.
smpquery 또는 smpdump	IB 서브넷 관리 속성을 질의 또는 덤프합니다.
ibcheckerrors 또는 ibcheckerrs	IB 포트(또는 노드) 또는 IB 서브넷의 유효성을 검사하고 오류를 보고합니다.
ibchecknet, ibchecknode 또는 ibcheckport	IB 서브넷, 노드 또는 포트의 유효성을 검사하고 오류를 보고합니다.
ibcheckportstate, ibcheckportwidth, ibcheckstate 또는 ibcheckwidth	링크가 살아 있으나 활성화되지 않은 IB 포트, 1x(2.0Gbps) 링크 쪽에 해당하는 포트, 링크가 살아 있으나 활성화되지 않은 IB 서브넷의 포트 또는 IB 서브넷의 1x 링크에 대한 유효성을 검사합니다.
ibclearcounters 또는 ibclearerrors	IB 서브넷에서 포트 카운터 또는 오류 카운터를 지웁니다.
ibdatacounters 또는 ibdatacounts	IB 서브넷의 데이터 카운트 또는 IB 포트 데이터 카운터를 질의합니다.
ibdiscover.pl	IB 토폴로지에 주석을 달고 IB 토폴로지를 비교합니다.
ibhosts	토폴로지의 IB 호스트 노드를 표시합니다.
iblinkinfo.pl 또는 iblinkinfo	패브릭의 모든 링크에 대한 링크 정보를 표시합니다.
ibnodes	토폴로지의 IB 노드를 표시합니다.
ibprintca.pl	ibnetdiscover 출력에서 지정된 CA 및 CA 목록을 표시합니다.
ibprintrt.pl	ibnetdiscover 출력에서 지정된 라우터 또는 라우터 목록만 표시합니다.

표 9-4 패브릭 진단 도구 (계속)

명령	설명
ibprintswitch.pl	ibnetdiscover 출력에서 지정된 스위치 또는 스위치 목록을 표시합니다.
ibqueryerrors.pl	0이 아닌 IB 포트 카운터를 질의 및 보고합니다.
ibrouters	토폴로지의 IB 라우터 노드를 표시합니다.
ibstatus	IB 장치의 기본 상태를 질의합니다.
ibswitches	토폴로지의 IB 스위치 노드를 표시합니다.
ibswportwatch.pl	지정한 스위치 또는 포트의 카운터와 변경 정보 보고 속도를 폴합니다.
set_nodedesc.sh	IB HCA(호스트 제어기 어댑터)에 대한 노드 설명 문자열을 설정 또는 표시합니다.
dump2psl.pl	크레딧 루프 검사에 사용되는 opensm 출력 파일을 기반으로 PSL 파일을 덤프합니다.
dump2slvl.pl	크레딧 루프 검사에 사용되는 opensm 출력 파일을 기반으로 SLVL 파일을 덤프합니다.
ibis	IB 관리 대역 내 서비스에 대한 확장된 TCL 셸입니다.

## 디스크 관리(개요)

---

이 장에서는 Oracle Solaris 디스크 슬라이스에 대한 개요 정보를 제공하고 `format` 유틸리티를 소개합니다.

이 장에서 다루는 개요 정보는 다음과 같습니다.

- 179 페이지 “디스크 관리의 새로운 기능”
- 185 페이지 “디스크 관리 작업 검색 위치”
- 185 페이지 “디스크 관리 개요”
- 186 페이지 “디스크 용어”
- 186 페이지 “디스크 레이블 정보”
- 190 페이지 “디스크 슬라이스 정보”
- 191 페이지 “`format` 유틸리티”
- 194 페이지 “디스크 분할”

시스템에 디스크를 추가하는 방법에 대한 지침은 12 장, “SPARC: 디스크 설정(작업)” 또는 13 장, “x86: 디스크 설정(작업)”을 참조하십시오.

## 디스크 관리의 새로운 기능

이 섹션에서는 Oracle Solaris 릴리스의 새로운 디스크 관리 기능에 대해 설명합니다.

- 180 페이지 “물리적 위치로 장치 식별”
- 183 페이지 “Oracle Solaris OS 설치 및 부트 시 2TB 디스크 지원”
- 184 페이지 “Solaris iSCSI 대상 및 시작 프로그램에서 iSNS 지원”
- 184 페이지 “Solaris COMSTAR iSCSI 지원”
- 184 페이지 “x86: GRUB 부트 환경의 디스크 관리”
- 185 페이지 “2TB보다 큰 SCSI 디스크에 대한 지원”

## 물리적 위치로 장치 식별

**Oracle Solaris 11:** 이 릴리스에서 `/dev/chassis` 디렉토리는 물리적 위치를 포함하는 장치 이름을 제공합니다. 장치를 교체하거나 변경해야 할 경우 이 정보를 사용하여 장치의 물리적 위치를 식별할 수 있습니다. 다음 명령을 사용하여 시스템에 있는 장치의 새시, 콘센트 및 보유자 값별로 정보를 표시할 수 있습니다.

- `diskinfo` - 물리적 디스크 위치에 대한 일반 정보를 표시하려면 이 명령을 사용합니다.
- `format` - 분할 영역 테이블을 검토하거나 레이블을 재지정할 때 디스크에 대한 물리적 디스크 위치 정보를 표시하려면 이 명령을 사용합니다.
- `prtconf -l` - 물리적 디스크 위치 정보를 포함하는 시스템 구성 정보를 표시하려면 이 명령을 사용합니다.
- `zpool status -l` - 풀 장치에 대한 물리적 디스크 위치 정보를 표시하려면 이 명령을 사용합니다.

또한 `fmadm add-alias` 명령을 사용하여 환경에 있는 디스크의 물리적 위치를 식별하는 데 도움이 되는 디스크 별칭을 포함시킬 수 있습니다. 예를 들면 다음과 같습니다.

```
# fmadm add-alias SUN-Storage-J4200.0912QAJ001 J4200@RACK10:U26-27
# fmadm add-alias SUN-Storage-J4200.0905QAJ00E J4200@RACK10:U24-25
```

그런 다음 앞의 명령을 사용하여 물리적 디스크 위치 정보를 표시하면 이러한 별칭을 표시할 수 있습니다. 예를 들면 다음과 같습니다.

```
$ diskinfo
D:devchassis-path                               c:occupant-compdev
-----
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__0/disk c1t13d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__1/disk c1t14d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__2/disk c1t2d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__3/disk c1t3d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__4/disk c1t15d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__5/disk c1t16d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__6/disk c1t6d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__7/disk c1t7d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__8/disk c1t17d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__9/disk c1t18d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__10/disk c1t10d0
```

특정 디스크의 위치를 확인합니다.

```
% diskinfo -c c6t11d0
D:devchassis-path                               c:occupant-compdev
-----
/dev/chassis/SUN-Storage-J4400/SCSI_Device__11/disk c6t11d0
```

이 예에서 `/dev/chassis` 디스크 이름에는 환경에 있는 장치를 찾는 데 도움이 되는 별칭이 포함되어 있습니다.

다음 `diskinfo` 예는 특정 디스크의 물리적 위치를 표시하는 방법을 보여 줍니다.

```
$ diskinfo -c c6t11d0 -o cp
c:occupant-compdev p:occupant-paths
-----
c6t11d0                /devices/pci@0,0/pci8086,3604@1/pci1000,3150@0/sd@b,0
```

시스템에 포함된 특정 유형의 디스크 수를 확인하려는 경우 다음과 유사한 `diskinfo` 구문을 사용합니다.

```
$ diskinfo -n SEAGATE ST31000N-SU0B-931.51GB -o Dcf
D:devchassis-path                t:occupant-type  c:occupant-compdev
-----
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__0/disk  disk            c0t13d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__1/disk  disk            c0t14d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__2/disk  disk            c0t2d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__3/disk  disk            c0t1d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__4/disk  disk            c0t15d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__5/disk  disk            c0t16d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__6/disk  disk            c0t6d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__7/disk  disk            c0t7d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__8/disk  disk            c0t17d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__9/disk  disk            c0t18d0
/dev/chassis/colab5@RACK10_24-25/SCSI_Device__10/disk disk            c0t10d0
```

주-diskinfo 명령을 사용하려면 새시가 SES 진단 페이지 0xa(추가 요소 상태)를 지원해야 하며, EIP(Element Index Present) 비트를 1로 설정해야 합니다. 이 조건을 충족하지 않는 엔클로저는 완전히 열거되지 않으므로 제대로 표시되지 않습니다.

`format` 명령이 물리적 장치 위치 정보를 제공하도록 업데이트되었습니다. 예를 들면 다음과 같습니다.

```
# format
.
.
.
18. c0t4d0 <SEAGATE-ST345056SSUN450G-081C-419.19GB>
    /pci@0,600000/pci@0/pci@9/LSILogic,sas@0/sd@4,0
    /dev/chassis/colab5@RACK10_26-27/SCSI_Device__6/disk
19. c0t27d0 <ATA-SEAGATE ST35000N-3AZQ-465.76GB>
    /pci@0,600000/pci@0/pci@9/LSILogic,sas@0/sd@1b,0
    /dev/chassis/colab5@RACK10_26-27/SCSI_Device__7/disk
20. c0t23d0 <ATA-SEAGATE ST31000N-SU0B-931.51GB>
    /pci@0,600000/pci@0/pci@9/LSILogic,sas@0/sd@17,0
    /dev/chassis/colab5@RACK10_26-27/SCSI_Device__8/disk
21. c0t24d0 <ATA-SEAGATE ST31000N-SU0B-931.51GB>
    /pci@0,600000/pci@0/pci@9/LSILogic,sas@0/sd@18,0
    /dev/chassis/colab5@RACK10_26-27/SCSI_Device__9/disk
```

`prtconf -l`를 사용하여 물리적 장치 위치 정보를 표시합니다. 예를 들면 다음과 같습니다.

```
$ prtconf -l | more
System Configuration: Oracle Corporation sun4v
Memory size: 32640 Megabytes
System Peripherals (Software Nodes):
```

```
SUNW,SPARC-Enterprise-T5220 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__0/disk
.
.
pci, instance #15 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__0/disk
LSILogic,sas, instance #1 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__0/disk
  smp, instance #0 (driver not attached)
  sd, instance #2 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__0/disk
  sd, instance #4 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__1/disk
  sd, instance #5 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__2/disk
  sd, instance #6 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__3/disk
  sd, instance #7 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__4/disk
  sd, instance #8 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__5/disk
  sd, instance #9 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__6/disk
  sd, instance #10 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__7/disk
  sd, instance #11 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__8/disk
  sd, instance #12 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__9/disk
  sd, instance #13 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__10/disk
  sd, instance #14 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device__11/disk
```

zpool status -l 옵션을 사용하여 물리적 장치 위치 정보를 표시합니다. 예를 들면 다음과 같습니다.

```
% zpool status -l export
pool: export
state: ONLINE
scan: resilvered 379G in 8h31m with 0 errors on Thu Jan 27 23:10:20 2011
config:
```

NAME	STATE	READ	WRITE	CKSUM
export	ONLINE	0	0	0
mirror-0	ONLINE	0	0	0
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__2/disk	ONLINE	0	0	0
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__3/disk	ONLINE	0	0	0
mirror-1	ONLINE	0	0	0
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__4/disk	ONLINE	0	0	0
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__5/disk	ONLINE	0	0	0
mirror-2	ONLINE	0	0	0
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__6/disk	ONLINE	0	0	0
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__7/disk	ONLINE	0	0	0
mirror-3	ONLINE	0	0	0
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__8/disk	ONLINE	0	0	0
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__9/disk	ONLINE	0	0	0
mirror-4	ONLINE	0	0	0
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__10/disk	ONLINE	0	0	0
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__11/disk	ONLINE	0	0	0
spares				
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__0/disk	AVAIL			
/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__1/disk	AVAIL			

```
errors: No known data errors
```

## 여러 디스크 섹터 크기 지원

**Oracle Solaris 11:** 이전 Solaris 릴리스에서는 512바이트의 디스크 섹터 크기를 지원했습니다. 이 릴리스에서는 섹터 크기가 512바이트, 1024바이트, 2048바이트 또는 4096바이트인 디스크가 지원됩니다. 큰 섹터 디스크는 디스크 용량이 크고 안정성이 높으며 데이터 전송 및 신속한 드라이브 유지 관리에 효율적입니다.

이 Oracle Solaris 릴리스에서 큰 섹터 디스크에 사용할 수 있도록 지원되는 파일 시스템은 루트가 아닌 ZFS 파일 시스템뿐입니다. 큰 섹터 디스크를 COMSTAR 대상으로 사용하는 방법은 14 장, “COMSTAR를 사용하여 저장 장치 구성”을 참조하십시오.

큰 섹터 디스크에서 부트 및 설치하는 기능은 현재 지원되지 않습니다.

## Oracle Solaris OS 설치 및 부트 시 2TB 디스크 지원

**Oracle Solaris 11:** 이전 Solaris 릴리스에서는 크기가 1TB보다 큰 디스크에서 Solaris OS를 설치 및 부트하지 못했습니다. 이 Solaris 릴리스에서는 최대 2TB 크기의 디스크에서 Solaris OS를 설치 및 부트할 수 있습니다. 또한 이전 릴리스에서는 1TB보다 큰 디스크에 EFI 레이블을 사용해야 했습니다. 이 릴리스에서는 모든 크기의 디스크에 VTOC 레이블을 사용할 수 있지만 VTOC에서 주소 지정 가능한 공간이 2TB로 제한됩니다.

Oracle Solaris 디스크 드라이버 및 디스크 유틸리티가 다음 지원을 제공하도록 업데이트되었습니다.

- 2TB 디스크에 Oracle Solaris OS를 설치 및 부트하는 경우 메모리가 최소 1.5GB인 시스템에 연결되어야 합니다.
- `format -e` 유틸리티를 사용하여 임의 크기의 디스크에 VTOC 레이블을 지정할 수 있지만 주소 지정 가능한 공간이 2TB로 제한됩니다.
- `format` 유틸리티 및 설치 소프트웨어가 2TB 크기보다 작은 디스크에 사용하는 기본 레이블은 VTOC 레이블입니다.
- x86 시스템에서는 1TB보다 큰 디스크에 대해 `fdisk` 유틸리티를 사용할 수 있습니다. 비-EFI 분할 영역 유형의 MBR에서 최대 2TB 분할 영역에 대한 지원이 추가되었습니다. 이 지원은 Solaris 분할 영역이 2TB까지 증가할 수 있음을 의미합니다. 다른 비-EFI 분할 영역에는 분할 영역 유형에 따른 제한이 적용될 수 있습니다.

2TB 크기보다 큰 디스크에서 `fdisk` 유틸리티를 실행하면 2TB 크기보다 큰 비-EFI 분할 영역을 만들 수 없다는 경고 메시지가 표시됩니다.

레거시 MBR 또는 레거시 VTOC가 있는 1TB 초과 디스크를 이전 Solaris 릴리스로 이동할 수 없습니다. EFI 레이블 디스크는 계속해서 이전 Solaris 릴리스처럼 작동합니다.

이 릴리스의 EFI 레이블 변경 사항에 대한 자세한 내용은 187 페이지 “EFI 디스크 레이블”을 참조하십시오.

## Solaris iSCSI 대상 및 시작 프로그램에서 iSNS 지원

**Oracle Solaris 11:** 이 Solaris 릴리스는 Solaris iSCSI 대상 및 시작 프로그램 소프트웨어에서 iSNS(Internet Storage Name Service) 프로토콜을 지원합니다. iSNS 프로토콜을 사용하면 TCP/IP 네트워크에서 iSCSI 장치의 자동화된 검색, 관리 및 구성이 가능합니다.

단계별 지침은 다음 리소스를 참조하십시오.

- 타사 iSNS 서버를 사용하도록 Oracle Solaris iSCSI 대상을 구성하는 방법에 대한 자세한 내용은 14 장, “COMSTAR를 사용하여 저장 장치 구성”을 참조하십시오.
- Solaris iSNS 서버를 사용하여 Solaris iSCSI 대상을 구성하는 방법에 대한 자세한 내용은 15 장, “Oracle Solaris iSNS(Internet Storage Name Service) 구성 및 관리” 및 `isnsadm(1M)`을 참조하십시오.

## Solaris COMSTAR iSCSI 지원

**Oracle Solaris 11:** iSCSI는 데이터 저장소 부속 시스템을 연결하기 위한 IP(인터넷 프로토콜) 기반 저장소 네트워킹 표준입니다. SCSI 명령을 IP 네트워크로 전달하는 iSCSI 프로토콜을 사용하면 네트워크상의 디스크 장치를 로컬 시스템으로 마운트할 수 있습니다. 로컬 시스템에서 이 장치를 블록 장치처럼 사용할 수 있습니다.

Common Multiprotocol SCSI TARget, 즉 COMSTAR는 시작 프로그램 호스트가 저장소 네트워크를 통해 액세스할 수 있는 SCSI 대상 장치로 Oracle Solaris 11 호스트를 변환할 수 있게 하는 소프트웨어 프레임워크입니다.

자세한 내용은 14 장, “COMSTAR를 사용하여 저장 장치 구성”을 참조하십시오.

## x86: GRUB 부트 환경의 디스크 관리

**Oracle Solaris 11:** x86 시스템을 부트하는 이전 방법이 GRUB 부트 메뉴로 대체되었습니다. 디스크 관리 영역에서 시스템 디스크를 교체하기 위해 대체 장치에서 부트하거나 부트 블록을 설치하는 경우 GRUB 인터페이스를 사용합니다.

GRUB 부트 환경은 다음 기능을 제공합니다.

- **네트워크 부트** - BIOS 구성 단계에서 F12 키를 눌러 네트워크에서 부트합니다.
- **단일 사용자 부트** - Solaris failsafe boot(Solaris 안정된 부트) 메뉴에서 이 옵션을 선택하여 단일 사용자 모드로 부트합니다.

```
kernel /platform/i86pc/multiboot
```

그런 다음 `e`(편집) 옵션을 사용하여 `-s` 단일 사용자 옵션을 추가합니다. 예를 들면 다음과 같습니다.

```
kernel /platform/i86pc/multiboot -s
```



return 키를 누른 다음 b 키를 눌러 시스템을 부트합니다. 시스템을 다시 다중 사용자 모드로 부트하려면 Ctrl-D를 누릅니다.

GRUB 환경에서는 x86 시스템에서 실행할 경우 `fmthard` 명령을 사용하여 부트 블록을 자동으로 설치할 수 없습니다. 부트 블록을 별도로 설치해야 합니다.

x86 시스템에서 새로운 GRUB 기반 부트를 사용하는 방법에 대한 지침과 자세한 기능 정보는 [x86 플랫폼에서 Oracle Solaris 부트 및 종료](#)를 참조하십시오.

SPARC 시스템에서는 이 기능을 사용할 수 없습니다.

## 2TB보다 큰 SCSI 디스크에 대한 지원

**Oracle Solaris 11:** SCSI 드라이버 `ssd` 또는 `sd`는 2TB 이상을 지원합니다. 이전 Solaris 릴리스에서는 SCSI 드라이버 `ssd` 또는 `sd`가 2TB로 제한되었습니다.

`format` 유틸리티를 사용하여 이러한 큰 디스크에 레이블을 지정하고 구성 및 분할할 수 있습니다. 큰 디스크에 EFI 디스크 레이블을 사용하는 방법 및 `fdisk` 유틸리티를 사용할 경우의 제한 사항에 대한 자세한 내용은 [188 페이지 “EFI 디스크 레이블의 제한 사항”](#)을 참조하십시오.

## 디스크 관리 작업 검색 위치

다음 참조를 사용하여 디스크 관리에 대한 단계별 지침을 찾을 수 있습니다.

디스크 관리 작업	자세한 정보
디스크를 포맷하고 디스크 레이블을 검사합니다.	11 장, “디스크 관리(작업)”
SPARC 시스템에 새 디스크를 추가합니다.	12 장, “SPARC: 디스크 설정(작업)”
x86 시스템에 새 디스크를 추가합니다.	13 장, “x86: 디스크 설정(작업)”
SCSI 또는 PCI 디스크를 핫 플러그합니다.	6 장, “동적으로 장치 구성(작업)”

## 디스크 관리 개요

Oracle Solaris OS에서 디스크를 관리하는 경우 대체로 시스템을 설정하고 Oracle Solaris 설치 프로그램을 실행하여 적절한 디스크 슬라이스 및 파일 시스템을 만들고 Oracle Solaris OS를 설치해야 합니다. 경우에 따라 `format` 유틸리티를 사용하여 새 디스크 드라이브를 추가하거나 결합이 있는 디스크 드라이브를 교체해야 할 수도 있습니다.

## 디스크 용어

이 섹션의 정보를 효율적으로 사용하려면 먼저 기본 디스크 아키텍처를 알고 있어야 합니다. 특히, 다음 용어를 알아두어야 합니다.

디스크 용어	설명
트랙	디스크가 회전할 때 단일 고정 디스크 헤드 아래를 통과하는 디스크의 동심원 링입니다.
실린더	디스크가 회전하는 중심 축에서 동일한 거리만큼 떨어진 트랙 세트입니다.
섹터	각 디스크의 둥근 레코드 섹션입니다. 한 섹터에 512바이트가 저장됩니다.
블록	디스크의 데이터 저장소입니다. 한 디스크 블록이 512바이트입니다.
디스크 제어기	디스크 드라이브를 제어하는 칩 및 연관된 회로입니다.
디스크 레이블	디스크 구조 및 분할 영역 정보를 포함하는 디스크의 첫 번째 섹터입니다.
장치 드라이버	하드웨어 또는 가상 장치를 제어하는 커널 모듈입니다.

자세한 내용은 디스크 제조업체의 제품 정보를 참조하십시오.

## 디스크 레이블 정보

디스크 제어기, 구조 및 슬라이스에 대한 정보를 저장하기 위해 각 디스크의 특수 영역이 할당됩니다. 이 정보를 디스크 **레이블**이라고 합니다. 디스크 레이블을 설명하는 데 사용되는 또 다른 용어는 VTOC 레이블 디스크의 *VTOC(Volume Table of Contents)*입니다. 디스크에 **레이블**을 지정하는 것은 디스크에 슬라이스 정보를 쓰는 것입니다. 일반적으로 슬라이스를 변경한 후 디스크에 레이블을 지정합니다.

Oracle Solaris 릴리스에서는 다음 두 가지 디스크 레이블을 지원합니다.

- SMI - 2TB 크기보다 작은 디스크에 사용되는 기존 VTOC 레이블입니다.
- EFI - 2TB보다 큰 디스크를 지원합니다. 2TB보다 작은 디스크에도 EFI GPT(Extensible Firmware Interface GUID Partition Table) 디스크 레이블을 사용할 수 있습니다.

슬라이스를 만든 후 디스크에 레이블을 지정하지 않으면 OS가 슬라이스를 "확인"할 방법이 없으므로 슬라이스를 사용할 수 없게 됩니다.

## EFI 디스크 레이블

EFI 레이블은 2TB 크기보다 큰 물리적 디스크 및 가상 디스크 볼륨을 지원합니다. 이 릴리스에는 2TB 크기보다 큰 디스크를 관리하기 위한 업데이트된 디스크 유틸리티도 포함되어 있습니다.

다음 파일 시스템 제품은 1TB 크기보다 큰 파일 시스템을 지원합니다.

- Oracle Solaris ZFS 파일 시스템은 1TB 크기보다 큰 파일 시스템을 지원합니다.

2TB 크기보다 작은 디스크에도 VTOC 레이블을 사용할 수 있습니다. 시스템에서 2TB보다 작은 디스크만 사용하는 경우 디스크 관리가 이전 Solaris 릴리스와 동일합니다. 또한 `format -e` 명령을 사용하여 크기가 2TB 이하인 디스크에 EFI 레이블을 지정할 수 있습니다. 자세한 내용은 예 11-5를 참조하십시오.

시스템이 적절한 Oracle Solaris 릴리스를 실행 중인 경우 `format -e` 명령을 사용하여 디스크에 EFI 레이블을 적용할 수 있습니다. 하지만 EFI 레이블을 적용하기 전에 188 페이지 “EFI 디스크 레이블의 제한 사항”에서 중요한 정보를 검토해야 합니다.

EFI 레이블이 더 이상 필요 없는 경우 `format -e` 명령을 사용하여 VTOC 레이블을 재적용할 수도 있습니다. 예를 들면 다음과 같습니다.

```
# format -e
Specify disk (enter its number): 2
selecting c0t5d0
[disk formatted]
.
.
format> label
[0] SMI Label
[1] EFI Label
Specify Label type[1]: 0
Warning: This disk has an EFI label. Changing to SMI label will erase all
current partitions.
Continue? yes
Auto configuration via format.dat[no]?
Auto configuration via generic SCSI-2[no]?
format> quit
```



**주의** - 디스크 레이블을 변경하면 디스크의 모든 데이터가 삭제됩니다.

### EFI 레이블과 VTOC 레이블 비교

EFI 디스크 레이블과 VTOC 디스크 레이블의 차이점은 다음과 같습니다.

- 2TB 크기보다 큰 디스크를 지원합니다.
- 사용 가능한 슬라이스 0-6을 제공합니다. 여기서 슬라이스 2는 단순히 다른 슬라이스입니다.

- 분할 영역(또는 슬라이스)이 기본 레이블 또는 백업 레이블이나 다른 분할 영역과 겹칠 수 없습니다. EFI 레이블의 크기는 대체로 34개 섹터이므로 분할 영역은 섹터 34에서 시작됩니다. 이 기능이 섹터 0에서 분할 영역을 시작할 수 없음을 의미하지는 않습니다.
- 실린더 정보는 EFI 레이블에 저장되지 않습니다. 크기가 블록 단위로 보고됩니다.
- 디스크의 마지막 2개 실린더인 대체 실린더 영역에 저장된 정보가 이제 슬라이스 8에 저장됩니다.
- `format` 유틸리티를 사용하여 분할 영역 크기를 변경하는 경우 크기가 0인 분할 영역에 `unassigned` 분할 영역 태그가 할당됩니다. 기본적으로 `format` 유틸리티는 크기가 0보다 큰 모든 분할 영역에 `usr` 분할 영역 태그를 할당합니다. 분할 영역이 변경된 후 `partition change`(분할 영역 변경) 메뉴를 사용하여 분할 영역 태그를 재할당할 수 있습니다. 하지만 크기가 0이 아닌 분할 영역을 `unassigned` 분할 영역 태그로 변경할 수는 없습니다.

## EFI 디스크 레이블의 제한 사항

1TB보다 큰 디스크를 사용하는 것이 환경에 적합한지 확인할 때 다음 제한 사항을 고려합니다.

- VTOC 레이블 디스크가 있는 시스템을 위한 계층적 소프트웨어 제품은 EFI 디스크 레이블이 있는 디스크에 액세스하지 못할 수도 있습니다.
- EFI 레이블 디스크는 이전 Solaris 릴리스를 실행하는 시스템에서 인식되지 않습니다.
- EFI 디스크 레이블이 있는 디스크에서 부트할 수 없습니다.
- x86 기반 시스템에서는 2TB 크기보다 큰 EFI 레이블 디스크에 대해 `fdisk` 명령을 사용할 수 있습니다.
- `format` 유틸리티를 사용하여 EFI 레이블 디스크를 분할합니다.
- EFI 사양에서는 슬라이스 겹침을 금지합니다. 전체 디스크가 `cxydz`로 표현됩니다.
- EFI 디스크 레이블은 섹터와 블록으로 디스크 또는 분할 영역 크기에 대한 정보를 제공하지만 실린더 및 헤더로는 제공하지 않습니다.
- 다음 `format` 옵션은 지원되지 않거나 EFI 레이블 디스크에 적용할 수 없습니다.
  - EFI 레이블 디스크는 `format.dat` 파일에 항목이 필요 없으므로 `save` 옵션이 지원되지 않습니다.
  - 디스크 드라이버가 기본 레이블을 찾아서 다시 디스크에 쓰기 때문에 `backup` 옵션은 적용할 수 없습니다.

## x86 시스템에서 EFI 레이블 디스크 지원

x86 시스템에서는 EFI 디스크 레이블에 대한 Oracle Solaris 지원을 사용할 수 있습니다. 다음 명령을 사용하여 x86 시스템에서 EFI 레이블을 추가합니다.

```
# format -e
> [0] SMI Label
```

```
> [1] EFI Label
> Specify Label type[0]: 1
> WARNING: converting this device to EFI labels will erase all current
> fdisk partition information. Continue? yes
```

이전 레이블 정보는 EFI 디스크 레이블로 변환되지 않습니다.

format 명령을 사용하여 레이블의 분할 영역 정보를 수동으로 다시 만들어야 합니다. 2TB 크기의 EFI 레이블 디스크에 대해 fdisk 명령을 사용할 수 없습니다. 2TB 크기보다 큰 디스크에서 fdisk 명령을 실행하여 Solaris 분할 영역을 만드는 경우 Solaris 분할 영역이 2TB로 제한됩니다. EFI 디스크 레이블에 대한 자세한 내용은 앞의 섹션을 참조하십시오.

## EFI 레이블 디스크를 사용하여 시스템 설치

Oracle Solaris 루트 풀 디스크에는 SMI 레이블이 있어야 합니다. Oracle Solaris 11 설치 유틸리티는 SMI 레이블을 사용하여 루트 풀 디스크로 선택된 모든 디스크의 레이블을 자동으로 재지정합니다.

## EFI 디스크 레이블을 사용하여 디스크 관리

다음 표를 통해 EFI 디스크 레이블을 사용하여 디스크를 관리하는 방법에 대한 정보를 찾습니다.

작업	자세한 정보
시스템이 이미 설치된 경우 디스크를 시스템에 연결합니다.	217 페이지 “SPARC: ZFS 파일 시스템에 사용할 디스크 설정(작업 맵)” 또는 227 페이지 “x86: ZFS 파일 시스템에 사용할 디스크 설정(작업 맵)”
ZFS 파일 시스템을 만듭니다.	218 페이지 “SPARC: ZFS 파일 시스템에 사용할 디스크 설정” 또는 228 페이지 “x86: ZFS 파일 시스템에 사용할 디스크 설정”

## EFI 디스크 레이블 문제 해결

다음 오류 메시지와 해결 방법을 사용하여 EFI 레이블 디스크 문제를 해결합니다.

### 해결 방법

1TB보다 큰 디스크를 사용하여 SPARC 또는 x86 커널을 실행하는 시스템을 부트합니다.

### 오류 메시지

```
Dec 3 09:12:17 holoship scsi: WARNING: /sbus@a,0/SUNW,socal@d,10000/
sf@1,0/ssd@w50020f23000002a4,0 (ssd1):
Dec 3 09:12:17 holoship corrupt label - wrong magic number
```

### 원인

이전 Solaris 릴리스를 실행하는 시스템에 디스크를 추가하려고 시도했습니다.

### 해결 방법

EFI 디스크 레이블을 지원하는 Solaris 릴리스를 실행하는 시스템에 디스크를 추가하십시오.

## 디스크 슬라이스 정보

디스크에 저장된 파일은 파일 시스템에 포함됩니다. 디스크의 각 파일 시스템은 해당 파일 시스템에 사용하도록 할당된 섹터 그룹인 **슬라이스**에 할당됩니다. 각 디스크 슬라이스는 개별 디스크 드라이브처럼 Oracle Solaris OS 및 시스템 관리자에게 표시됩니다.

파일 시스템에 대한 자세한 내용은 17 장, “파일 시스템 관리(개요)”를 참조하십시오.

주-슬라이스를 *partitions*라고도 합니다. `format` 유틸리티와 같은 특정 인터페이스는 슬라이스를 분할 영역이라고 합니다.

슬라이스를 설정할 때는 다음 규칙을 따릅니다.

- 각 디스크 슬라이스에 파일 시스템 1개만 저장됩니다.
- 파일 시스템은 여러 슬라이스에 걸쳐 있을 수 없습니다.

SPARC 및 x86 플랫폼에서 슬라이스가 약간 다르게 설정됩니다. 다음 표에는 차이점이 요약되어 있습니다.

표 10-1 SPARC 및 x86 플랫폼에서 슬라이스 차이점

SPARC 플랫폼	x86 플랫폼
전체 디스크가 Oracle Solaris OS 전용으로 사용됩니다.	디스크가 운영 체제당 <code>fdisk</code> 분할 영역 1개씩, <code>fdisk</code> 분할 영역으로 나뉩니다.
<b>VTOC</b> - 디스크가 0-7까지 번호가 지정된 8개 슬라이스로 나뉩니다.	<b>VTOC</b> - Solaris <code>fdisk</code> 분할 영역이 0-9까지 번호가 지정된 10개 슬라이스로 나뉩니다.
<b>EFI</b> - 디스크가 0-6까지 번호가 지정된 7개 슬라이스로 나뉩니다.	<b>EFI</b> - 디스크가 0-6까지 번호가 지정된 7개 슬라이스로 나뉩니다.

## 디스크 슬라이스

ZFS 파일 시스템이 있는 시스템에서는 ZFS 루트 파일 시스템이 포함된 ZFS 루트 풀에만 디스크 슬라이스가 필요합니다. 일반적으로 ZFS 루트 풀은 슬라이스 0에 포함됩니다. 오래된 부트 제한 사항 때문에 ZFS 루트 풀은 슬라이스나 미러링된 슬라이스에 있어야 합니다. 루트가 아닌 저장소 풀의 경우 전체 디스크를 사용할 수 있습니다. ZFS 파일 시스템은 특정 디스크 슬라이스에 해당하지 않습니다. 전체 디스크 관리가 슬라이스를 포함하는 디스크 관리보다 더 쉽습니다.

x86 시스템:

- 디스크가 fdisk 분할 영역으로 나뉩니다. fdisk 분할 영역은 Oracle Solaris OS와 같은 특정 운영 체제에 예약된 디스크의 한 섹션입니다.
- Oracle Solaris OS는 Solaris fdisk 분할 영역에 0-9까지 번호가 지정된 10개 슬라이스를 배치합니다.

주 - VTOC 레이블 디스크에서는 슬라이스를 수정하거나 슬라이스 2를 사용하여 파일 시스템을 저장하지 마십시오. 슬라이스 2를 어떤 방식으로든 수정하면 installgrub 명령이 제대로 작동하지 않습니다.

## 원시(raw) 데이터 슬라이스 사용

디스크 레이블은 각 디스크의 블록 0에 저장됩니다. 따라서 원시(raw) 데이터 슬라이스를 만드는 타사 데이터베이스 응용 프로그램은 블록 0에서 시작하면 안 됩니다. 그렇지 않으면 디스크 레이블을 덮어쓰게 되어 디스크의 데이터에 액세스할 수 없습니다.

때때로 타사 데이터베이스 응용 프로그램에서 만들어진 원시(raw) 데이터 슬라이스에 다음 디스크 영역을 사용하지 마십시오.

- 디스크 레이블이 저장되는 블록 0
- VTOC 레이블이 있는 전체 디스크를 나타내는 슬라이스 2

## format 유틸리티

"방법" 또는 참조로 넘어가기 전에 format 유틸리티 및 해당 사용에 대한 다음 개요를 읽어보십시오.

format 유틸리티는 Oracle Solaris 시스템에서 사용할 하드 디스크 드라이브를 준비하는데 사용되는 시스템 관리 도구입니다.

다음 표에서는 format 유틸리티가 제공하는 기능 및 연관된 이점을 보여 줍니다.

표 10-2 format 유틸리티의 기능 및 이점

기능	이점
시스템에서 연결된 모든 디스크 드라이브 검색	다음에 대해 보고합니다. <ul style="list-style-type: none"> <li>■ 대상 위치</li> <li>■ 디스크 구조</li> <li>■ 디스크 포맷 여부</li> <li>■ 디스크에 마운트된 분할 영역이 있는 경우</li> </ul>
디스크 레이블 검색	복구 작업에 편리합니다.

표 10-2 format 유틸리티의 기능 및 이점 (계속)

기능	이점
결함이 있는 섹터 복구	관리자가 드라이브를 다시 제조업체에 보내는 대신 복구 가능한 오류가 있는 디스크 드라이브를 복구할 수 있게 합니다.
디스크 포맷 및 분석	디스크에 섹터를 만들고 각 섹터를 확인합니다.
디스크 분할	디스크를 슬라이스로 나눕니다. ZFS 루트 풀을 제외하고 ZFS 파일 시스템은 디스크 슬라이스에 해당하지 않습니다.
디스크에 레이블 지정	대체로 복구 작업을 위해 나중에 검색할 수 있도록 디스크 이름 및 구성 정보를 디스크에 씁니다.

format 유틸리티 옵션은 16 장, “format 유틸리티(참조)”에서 설명합니다.

## format 유틸리티 사용 시기

Oracle Solaris 릴리스를 설치할 때 Oracle Solaris 설치 유틸리티가 디스크 드라이브를 분할하고 레이블을 지정합니다. format 유틸리티를 사용하여 다음을 수행할 수 있습니다.

- 슬라이스 정보 표시
- 디스크 분할
- 기존 시스템에 디스크 드라이브 추가
- 디스크 드라이브 포맷
- 디스크에 레이블 지정
- 디스크 드라이브 복구
- 디스크 오류 분석

시스템 관리자는 주로 디스크를 분할하기 위해 format 유틸리티를 사용합니다. 이러한 단계는 12 장, “SPARC: 디스크 설정(작업)” 및 13 장, “x86: 디스크 설정(작업)”에서 설명합니다.

format 유틸리티 사용에 대한 지침은 다음 섹션을 참조하십시오.



## format 유틸리티 사용 지침

표 10-3 format 유틸리티 지침

작업	지침	자세한 정보
디스크를 포맷합니다.	<ul style="list-style-type: none"> <li>■ 디스크를 재포맷하면 기존 데이터가 모두 삭제됩니다.</li> <li>■ 디스크 드라이브를 포맷 및 분할한 상태로 제공하는 제조업체가 증가함에 따라 디스크 드라이브 포맷 요구가 감소하고 있습니다. 기존 시스템에 디스크 드라이브를 추가하거나 교체할 때 <b>format</b> 유틸리티를 사용할 필요가 없을 수도 있습니다.</li> <li>■ 디스크를 재배치한 후 많은 디스크 오류가 표시되는 경우 레이블을 재지정해 볼 수 있습니다.</li> </ul>	203 페이지 “디스크 포맷 방법” 또는 206 페이지 “디스크에 레이블을 지정하는 방법”
ZFS 루트 파일 시스템이 포함된 디스크를 설정합니다.	비중복 구성에서는 손상된 디스크의 ZFS 루트 파일 시스템 데이터를 백업 매체에서 복원해야 합니다. 그렇지 않으면 설치 유틸리티를 사용하여 시스템을 재설치해야 합니다.	219 페이지 “SPARC: ZFS 루트 파일 시스템에 사용할 디스크 설정 방법” 또는 229 페이지 “x86: ZFS 루트 파일 시스템에 사용할 디스크 설정 방법”, 또는 시스템을 재설치해야 하는 경우 <b>Oracle Solaris 11 시스템</b>
루트 풀에 사용할 디스크 슬라이스를 만듭니다.	<ul style="list-style-type: none"> <li>■ ZFS 저장소 풀을 사용하는 최상의 방법은 전체 디스크가 포함된 풀을 만드는 것입니다.</li> <li>■ 디스크를 루트 풀에 사용하려는 경우 디스크 슬라이스를 만들어야 합니다. 이것은 오래된 부트 제한 사항입니다.</li> </ul>	220 페이지 “SPARC: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만드는 방법” 또는 230 페이지 “x86: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만드는 방법”
ZFS 파일 시스템이 포함된 디스크를 설정합니다.	루트가 아닌 ZFS 파일 시스템에 사용되는 디스크는 대체로 사용자 또는 데이터 파일용 공간을 포함합니다. 추가 디스크 공간을 위해 루트 풀이나 루트가 아닌 풀에 다른 디스크를 연결 또는 추가할 수 있습니다.	224 페이지 “SPARC: ZFS 파일 시스템에 사용할 디스크 설정 방법” 또는 234 페이지 “x86: ZFS 파일 시스템에 사용할 디스크 설정 방법”

## 디스크 포맷

대부분의 경우 디스크는 제조업체나 판매자가 포맷합니다. 따라서 드라이브를 설치할 때 재포맷할 필요가 없습니다. 디스크가 포맷되었는지 확인하려면 **format** 유틸리티를 사용합니다. 자세한 내용은 202 페이지 “디스크가 포맷되었는지 확인하는 방법”을 참조하십시오.

디스크가 포맷되지 않은 것을 확인하면 `format` 유틸리티를 사용하여 디스크를 포맷합니다.

디스크를 포맷하면 다음 두 단계가 수행됩니다.

- 디스크 매체를 사용할 준비가 완료됩니다.
- 표면 분석을 기반으로 하는 디스크 결함 목록이 컴파일됩니다.



**주의** - 디스크 포맷은 디스크의 데이터를 덮어쓰기 때문에 파괴적 프로세스입니다. 이런 이유 때문에 대체로 제조업체나 재판매자만 디스크를 포맷합니다. 디스크 결함이 반복되는 문제의 원인으로 의심되는 경우 `format` 유틸리티를 사용하여 표면 분석을 수행할 수 있습니다. 하지만 데이터를 삭제하지 않는 명령만 사용하십시오. 자세한 내용은 203 페이지 “디스크 포맷 방법”을 참조하십시오.

데이터에 사용 가능한 총 디스크 공간 중 작은 비율이 결함 및 포맷 정보 저장에 사용됩니다. 이 비율은 디스크 구조에 따라 달라지며, 디스크 사용 기간이 오래되고 더 많은 결함이 생길수록 감소합니다.

디스크 포맷은 디스크 유형과 크기에 따라 몇 분에서 몇 시간까지 걸릴 수 있습니다.

## 디스크 분할

`format` 유틸리티는 주로 시스템 관리자가 디스크를 분할하는 데 사용됩니다. 단계는 다음과 같습니다.

- 필요한 슬라이스 확인
- 각 슬라이스 또는 분할 영역의 크기 확인
- `format` 유틸리티를 사용하여 디스크 분할
- 새 분할 영역 정보를 사용하여 디스크에 레이블 지정
- 각 분할 영역에 대한 파일 시스템 만들기

디스크를 분할하는 가장 쉬운 방법은 `format` 유틸리티의 `partition`(분할 영역) 메뉴에서 `modify` 명령을 사용하는 것입니다. `modify` 명령을 사용하면 시작 실린더 경계를 추적할 필요 없이 각 분할 영역의 크기를 지정하여 분할 영역을 만들 수 있습니다. 또한 `modify` 명령은 "Free Hog" 슬라이스에 남아 있는 디스크 공간을 모두 추적합니다.

## 분할 영역 테이블 용어

**분할 영역 테이블**은 디스크 레이블의 중요한 일부입니다. 분할 영역 테이블은 디스크의 슬라이스, 슬라이스 경계(실린더 단위) 및 총 슬라이스 크기를 식별합니다. `format` 유틸리티를 사용하여 디스크의 분할 영역 테이블을 표시할 수 있습니다. 다음은 분할 영역 테이블 용어에 대한 설명입니다.

표 10-4 분할 영역 테이블 용어

분할 영역 용어	값	설명
숫자	0-7	<b>VTOC</b> - 0-7까지 번호가 지정된 분할 영역 또는 슬라이스입니다.  <b>EFI</b> - 0-6까지 번호가 지정된 분할 영역 또는 슬라이스입니다.
태그	0=UNASSIGNED 1=BOOT 2=ROOT 3=SWAP 4=USR 5=BACKUP 7=VAR 8=HOME 11=RESERVED	대체로 이 분할 영역에 마운트된 파일 시스템을 설명하는 숫자 값입니다.
플래그	wm	분할 영역이 쓰기 및 마운트 가능합니다.
	wu rm	분할 영역이 쓰기 및 마운트 해제 가능합니다. 스왑 영역 전용으로 사용되는 분할 영역의 경우 이 상태가 기본값입니다. 하지만 mount 명령은 "마운트 가능하지 않음" 플래그를 확인하지 않습니다.
	rm	분할 영역이 읽기 전용이고 마운트 가능합니다.

분할 영역 플래그와 태그가 규칙에 의해 할당되며 유지 관리가 필요 없습니다.

분할 영역 테이블 표시에 대한 자세한 내용은 다음을 참조하십시오.

- 195 페이지 “분할 영역 테이블 정보 표시”
- 204 페이지 “디스크 슬라이스 정보 표시 방법”
- 211 페이지 “디스크 레이블 검사 방법”

## 분할 영역 테이블 정보 표시

다음 format 유틸리티 출력에서는 VTOC 레이블이 표시된 74GB 디스크의 분할 영역 테이블 예를 보여 줍니다.

Total disk cylinders available: 38756 + 2 (reserved cylinders)

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	3 - 2083	4.00GB	(2081/0/0) 8390592
1	swap	wu	2084 - 3124	2.00GB	(1041/0/0) 4197312
2	backup	wm	0 - 38755	74.51GB	(38756/0/0) 156264192
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	unassigned	wm	0	0	(0/0/0) 0
7	home	wm	3125 - 38755	68.50GB	(35631/0/0) 143664192
8	boot	wu	0 - 0	1.97MB	(1/0/0) 4032
9	alternates	wu	1 - 2	3.94MB	(2/0/0) 8064

partition>

format 유틸리티에서 표시되는 분할 영역 테이블은 다음 정보를 포함합니다.

열 이름	설명
Part	분할 영역 또는 슬라이스 번호입니다. 이 열에 대한 설명은 표 10-4를 참조하십시오.
Tag	분할 영역 태그입니다. 이 열에 대한 설명은 표 10-4를 참조하십시오.
Flag	분할 영역 플래그입니다. 이 열에 대한 설명은 표 10-4를 참조하십시오.
Cylinders	슬라이스의 시작 및 종료 실린더 번호입니다. EFI 레이블 디스크에는 표시되지 않습니다.
Size	슬라이스 크기(MB)입니다.
Blocks	슬라이스당 총 섹터 수 및 총 실린더 수입니다. EFI 레이블 디스크에는 표시되지 않습니다.
First Sector	<b>EFI</b> - 시작 블록 번호입니다. VTOC 레이블 디스크에는 표시되지 않습니다.
Last Sector	<b>EFI</b> - 종료 블록 번호입니다. VTOC 레이블 디스크에는 표시되지 않습니다.

다음은 prtvtoc 명령을 사용하여 표시되는 EFI 디스크 레이블의 예입니다.

```
# prtvtoc /dev/rdisk/c4t1d0s0
* /dev/rdisk/c4t1d0s0 partition map
*
* Dimensions:
*   512 bytes/sector
* 2576941056 sectors
* 2576940989 accessible sectors
*
* Flags:
*   1: unmountable
*  10: read-only
*
*
*      First      Sector      Last
* Partition Tag  Flags      Sector      Count      Sector      Mount Directory
*   0         2    00           34    629145600    629145633
*   1         4    00    629145634    629145600    1258291233
*   6         4    00   1258291234   1318633404    2576924637
*   8        11    00   2576924638         16384    2576941021
```

prtvtoc 명령의 출력은 다음 세 개 섹션의 정보를 제공합니다.

- 차원
- 플래그
- 분할 영역 테이블

prtvtoc 열 이름	설명
Partition	분할 영역 또는 슬라이스 번호입니다. 이 열에 대한 설명은 표 10-4를 참조하십시오.
Tag	분할 영역 태그입니다. 이 열에 대한 설명은 표 10-4를 참조하십시오.
Flags	분할 영역 플래그입니다. 이 열에 대한 설명은 표 10-4를 참조하십시오.
First Sector	슬라이스의 첫 번째 섹터입니다.
Sector Count	슬라이스의 총 섹터 수입니다.
Last Sector	슬라이스의 마지막 섹터입니다.
Mount Directory	파일 시스템의 마지막 마운트 지점 디렉토리입니다.

## Free Hog 슬라이스 사용

`format` 유틸리티를 사용하여 하나 이상의 디스크 슬라이스 크기를 변경하는 경우 크기 조정 작업을 위해 확장 및 축소될 임시 슬라이스를 지정합니다.

이 임시 슬라이스는 슬라이스를 확장할 때 공간을 기증 또는 "비우고(*free*)", 슬라이스를 축소할 때 삭제된 공간을 수신 또는 "보유(*hog*)"합니다. 이런 이유로 기증자 슬라이스를 *Free Hog*라고도 합니다.

Free Hog 슬라이스는 설치 도중이나 `format` 유틸리티를 실행할 때만 존재합니다. 일상 작업 중에는 영구적인 Free Hog 슬라이스가 없습니다.

Free Hog 슬라이스 사용에 대한 자세한 내용은 220 페이지 “SPARC: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만드는 방법” 또는 230 페이지 “x86: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만드는 방법”을 참조하십시오.



## 디스크 관리(작업)

이 장에는 디스크 관리 절차가 포함되어 있습니다. Oracle Solaris OS를 실행하는 시스템에서 디스크가 관리되는 방식을 이미 알고 있는 경우 이 장에서 설명하는 대부분의 절차는 선택 사항입니다.

디스크 관리와 연관된 절차에 대한 자세한 내용은 199 페이지 “디스크 관리(작업 맵)”를 참조하십시오.

디스크 관리에 대한 개요 정보는 10 장, “디스크 관리(개요)”를 참조하십시오.

### 디스크 관리(작업 맵)

작업	설명	수행 방법
시스템의 디스크를 식별합니다.	시스템의 디스크 유형이 확실하지 않은 경우 <code>format</code> 유틸리티를 사용하여 디스크 유형을 식별합니다.	200 페이지 “시스템의 디스크 식별 방법”
디스크를 포맷합니다.	<code>format</code> 유틸리티를 사용하여 디스크가 이미 포맷되었는지 확인합니다.  대부분의 경우 디스크가 이미 포맷되었습니다. 디스크를 포맷해야 하는 경우 <code>format</code> 유틸리티를 사용합니다.	202 페이지 “디스크가 포맷되었는지 확인하는 방법”  203 페이지 “디스크 포맷 방법”
슬라이스 정보를 표시합니다.	<code>format</code> 유틸리티를 사용하여 슬라이스 정보를 표시합니다.	204 페이지 “디스크 슬라이스 정보 표시 방법”
디스크에 레이블을 지정합니다.	<code>format</code> 유틸리티를 사용하여 디스크 레이블을 만듭니다.	206 페이지 “디스크에 레이블을 지정하는 방법”

작업	설명	수행 방법
디스크 레이블을 검사합니다.	prtvtoc 명령을 사용하여 디스크 레이블을 검사합니다.	211 페이지 “디스크 레이블 검사 방법”
손상된 디스크 레이블을 복구합니다.	시스템 오류나 정전으로 인해 손상된 디스크 레이블의 복구를 시도할 수 있습니다.	213 페이지 “손상된 디스크 레이블 복구 방법”

## 시스템의 디스크 식별

format 유틸리티를 사용하여 시스템에 연결된 디스크 유형을 검색합니다. format 유틸리티를 사용하여 디스크가 시스템에 알려졌는지 확인할 수도 있습니다. format 유틸리티 사용에 대한 자세한 내용은 16 장, “format 유틸리티(참조)”를 참조하십시오.

### ▼ 시스템의 디스크 식별 방법

#### 1 관리자로 전환합니다.

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

#### 2 format 유틸리티를 사용하여 시스템에서 인식되는 디스크를 식별합니다.

# format

format 유틸리티는 인식되는 디스크 목록을 AVAILABLE DISK SELECTIONS 아래에 표시합니다.

#### 예 11-1 시스템의 디스크 식별

다음 예에서는 format 명령 출력을 보여 줍니다.

```
# format
AVAILABLE DISK SELECTIONS:
 0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
   /pci@1c,600000/scsi@2/sd@0,0
 1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
   /pci@1c,600000/scsi@2/sd@1,0
 2. c2t2d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
   /pci@1c,600000/scsi@2/sd@2,0
 3. c2t3d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
   /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number):
```

이 출력에서는 디스크의 물리적 및 논리적 장치 이름이 꺾쇠 괄호(<>) 안에 표시되는 디스크의 마케팅 이름에 연결됩니다. 아래 예를 참조하십시오. 이 방법은 시스템에 연결된 디스크를 나타내는 논리적 장치 이름을 식별하는 편리한 방법입니다. 논리적 및 물리적 장치 이름에 대한 설명은 72 페이지 “장치 이름 지정 규약”을 참조하십시오.



다음 예에서는 와일드카드를 사용하여 제어기 0에 연결된 디스크 4개를 표시합니다.

```
# format /dev/rdisk/c0t6*
AVAILABLE DISK SELECTIONS:
  0. /dev/rdsk/c0t600A0B800022024E000054AC4970A629d0p0 <...>
    /scsi_vhci/disk@g600a0b800022024e000054ac4970a629
  1. /dev/rdsk/c0t600A0B800022024E000054AE4970A711d0p0 <...>
    /scsi_vhci/disk@g600a0b800022024e000054ae4970a711
  2. /dev/rdsk/c0t600A0B800022028A000050444970A834d0p0 <...>
    /scsi_vhci/disk@g600a0b800022028a000050444970a834
  3. /dev/rdsk/c0t600A0B800022028A000050454970A8EAd0p0 <...>
    /scsi_vhci/disk@g600a0b800022028a000050454970a8ea
Specify disk (enter its number):
```

다음 예에서는 디스크 정보를 식별하는 방법을 보여 줍니다.

```
# format
0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
   /pci@1c,600000/scsi@2/sd@0,0
```

이 출력에서 디스크 0(대상 0)이 두번째 SCSI 호스트 어댑터(scsi@2)에 연결되었다고 식별됩니다. 두번째 호스트는 두번째 PCI 인터페이스(/pci@1c,600000/...)에 연결됩니다. 또한 이 출력에서는 물리적 및 논리적 장치 이름이 모두 디스크의 마케팅 이름 SUN36G에 연결됩니다.

일부 디스크에는 마케팅 이름이 없습니다. format 출력에서 디스크가 마케팅 이름으로 식별되지 않는 경우 다음 단계에 설명된 대로 format 유틸리티의 type 및 label 기능을 사용하여 디스크의 마케팅 이름을 포함할 수 있습니다.

다음 단계는 디스크나 시스템이 비활성 상태일 때 수행해야 합니다. 즉, 디스크가 현재 사용되지 않으며 Oracle Solaris 릴리스를 포함하지 않는 경우가 아니면 설치 DVD 또는 네트워크에서 부트해야 합니다. 또한 최종 단계는 디스크 레이블을 재지정하는 것으로, 이 경우 기존 분할 영역 정보나 데이터가 모두 제거될 수 있습니다.

```
# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
  0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
    /pci@1c,600000/scsi@2/sd@0,0
  1. c2t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
    /pci@1c,600000/scsi@2/sd@1,0
  2. c2t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
    /pci@1c,600000/scsi@2/sd@2,0
  3. c2t3d0 <drive type unknown>
    /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
format> type
AVAILABLE DRIVE TYPES:
  0. Auto configure
  1. other
```

```
Specify disk type (enter its number): 0
c2t3d0: configured with capacity of 33.92GB
<SEAGATE-ST336607LSUN36G-0507-33.92GB>
[disk formatted]
format> label
Ready to label disk, continue? yes
format> quit
```

### 자세한 정보 **format** 유틸리티가 디스크를 인식할 수 없는 경우...

- 12 장, “SPARC: 디스크 설정(작업)” 또는 13 장, “x86: 디스크 설정(작업)”으로 이동합니다.
- 206 페이지 “디스크에 레이블을 지정하는 방법”으로 이동합니다.
- 디스크 하드웨어 설명서를 사용하여 디스크를 시스템에 연결합니다.

## 디스크 포맷

일반적으로 디스크는 제조업체나 재판매자가 포맷합니다. 따라서 드라이브를 설치할 때 재포맷할 필요가 없습니다.

다음을 수행하려면 먼저 디스크를 포맷해야 합니다.

- 디스크에 데이터를 씁니다. 하지만 대부분의 디스크가 이미 포맷되었습니다.
- Oracle Solaris 설치 유틸리티를 사용하여 시스템을 설치합니다.



주의 - 디스크 포맷은 디스크의 데이터를 덮어쓰기 때문에 파괴적 프로세스입니다. 이런 이유 때문에 대체로 제조업체나 재판매자만 디스크를 포맷합니다. 디스크 결함이 반복되는 문제의 원인으로 의심되는 경우 **format** 유틸리티를 사용하여 표면 분석을 수행할 수 있습니다. 하지만 데이터를 삭제하지 않는 명령만 사용하십시오.

### ▼ 디스크가 포맷되었는지 확인하는 방법

#### 1 관리자로 전환합니다.

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

#### 2 **format** 유틸리티를 호출합니다.

```
# format
```

디스크의 번호 매기기 목록이 표시됩니다.

#### 3 확인하려는 디스크의 번호를 입력합니다.

```
Specify disk (enter its number): 0
```

- 4 다음 메시지를 검사하여 선택한 디스크가 포맷되었는지 확인합니다.

```
[disk formatted]
```

## 예 11-2 디스크가 포맷되었는지 확인

다음 예에서는 c2t3d0 디스크가 포맷되었음을 보여 줍니다.

```
# format
AVAILABLE DISK SELECTIONS:
  0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
    /pci@1c,600000/scsi@2/sd@0,0
  1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
    /pci@1c,600000/scsi@2/sd@1,0
  2. c2t2d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
    /pci@1c,600000/scsi@2/sd@2,0
  3. c2t3d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
    /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
```

## ▼ 디스크 포맷 방법

디스크는 제조업체가 포맷합니다. 디스크를 재포맷할 때는 주의해야 합니다. 이 프로세스는 오랜 시간이 걸리며 디스크에서 모든 데이터를 제거합니다.

- 1 관리자로 전환합니다.

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

- 2 **format** 유틸리티를 호출합니다.

```
# format
디스크의 번호 매기기 목록이 표시됩니다.
```

- 3 포맷하려는 디스크의 번호를 입력합니다.

```
Specify disk (enter its number): 3
```



주의 - 루트 파일 시스템이 포함된 디스크를 선택하지 마십시오. 루트 풀 디스크를 포맷하는 경우 이 디스크의 OS와 데이터가 모두 삭제됩니다.

- 4 디스크 포맷을 시작하려면 **format>** 프롬프트에서 **format**을 입력합니다. **y**를 입력하여 명령을 확인합니다.

```
format> format
The protection information is not enabled
The disk will be formatted with protection type 0
```

```
Ready to format. Formatting cannot be interrupted
and takes 169 minutes (estimated). Continue? yes
```

**5 다음 메시지를 검사하여 디스크 포맷이 성공했는지 확인합니다.**

```
Beginning format. The current time is Fri Apr 1 ...
```

```
Formatting...
done
```

```
Verifying media...
    pass 0 - pattern = 0xc6dec6de
14086/23/734
```

```
    pass 1 - pattern = 0x6db6db6d
14086/23/734
```

```
Total of 0 defective blocks repaired.
```

**6 format 유틸리티를 종료합니다.**

```
format> quit
```

## 디스크 슬라이스 표시

전체 디스크를 관리하는 것이 더 쉽기 때문에 ZFS 저장소 풀을 만드는 최상의 방법은 디스크 슬라이스 대신 전체 디스크를 사용하는 것입니다. 디스크 슬라이스를 사용해야 하는 유일한 경우는 디스크를 ZFS 루트 풀에 사용하려는 경우입니다. 이것은 오래된 부트 제한 사항입니다. 루트가 아닌 풀의 경우 전체 디스크를 사용합니다. 전체 디스크를 사용하여 풀을 만드는 경우 EFI 레이블이 적용됩니다. 아래 EFI 디스크 레이블 예를 참조하십시오.

루트 풀 디스크로 사용할 디스크를 준비해야 하는 경우 아래 VTOC 디스크 레이블 예와 같이 전체 디스크 용량을 포함하는 슬라이스 0을 만듭니다.

ZFS 저장소 풀에 사용할 디스크 설정에 대한 자세한 내용은 [12 장, “SPARC: 디스크 설정\(작업\)”](#) 또는 [13 장, “x86: 디스크 설정\(작업\)”](#)을 참조하십시오.

---

주 - format 유틸리티는 슬라이스 대신 **분할 영역**이란 용어를 사용합니다.

---

### ▼ 디스크 슬라이스 정보 표시 방법

디스크를 ZFS 루트 풀에 사용하려는 경우 디스크 슬라이스 정보를 표시해야 할 수 있습니다. 또한 SMI 레이블을 포함해야 합니다.

1 관리자로 전환합니다.

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

2 **format** 유틸리티를 호출합니다.

```
# format
```

디스크의 번호 매기기 목록이 표시됩니다.

3 슬라이스 정보를 표시하려는 디스크의 번호를 입력합니다.

```
Specify disk (enter its number):1
```

4 **partition** 메뉴를 선택합니다.

```
format> partition
```

5 선택한 디스크에 대한 슬라이스 정보를 표시합니다.

```
partition> print
```

6 **format** 유틸리티를 종료합니다.

```
partition> q
format> q
```

7 특정 슬라이스 태그와 슬라이스를 식별하여 표시된 슬라이스 정보를 확인합니다.

화면 출력이 슬라이스 크기가 지정되지 않았다고 표시되는 경우 디스크에 슬라이스가 없는 것입니다.

### 예 11-3 디스크 슬라이스 정보 표시

다음 예에서는 VTOC 레이블 디스크에 대한 슬라이스 정보를 표시합니다.

```
# format
Searching for disks...done
Specify disk (enter its number):3
Selecting c2t3d0
format> partition
partition> print
Current partition table (c2t3d0):
Total disk cylinders available: 14087 + 2 (reserved cylinders)

Part    Tag    Flag    Cylinders    Size    Blocks
  0      root   wm      0 - 14086    136.71GB (14087/0/0) 286698624
  1      swap   wu      0            0      (0/0/0) 0
  2      backup wu      0 - 14086    136.71GB (14087/0/0) 286698624
  3  unassigned wm      0            0      (0/0/0) 0
  4  unassigned wm      0            0      (0/0/0) 0
  5  unassigned wm      0            0      (0/0/0) 0
  6      usr    wm      0            0      (0/0/0) 0
  7  unassigned wm      0            0      (0/0/0) 0
partition> q
format> q
```

이러한 예의 슬라이스 정보에 대한 자세한 설명은 10 장, “디스크 관리(개요)”를 참조하십시오.

다음 예에서는 EFI 레이블 디스크에 대한 슬라이스 정보를 보여 줍니다.

```
# format
Searching for disks...done
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
format> partition
partition> print
Current partition table (default):
Total disk sectors available: 286722878 + 16384 (reserved sectors)

Part      Tag      Flag      First Sector      Size      Last Sector
  0        usr      wm         34      136.72GB     286722911
  1 unassigned  wm         0         0           0
  2 unassigned  wm         0         0           0
  3 unassigned  wm         0         0           0
  4 unassigned  wm         0         0           0
  5 unassigned  wm         0         0           0
  6 unassigned  wm         0         0           0
  7 unassigned  wm         0         0           0
  8 reserved   wm      286722912      8.00MB     286739295
partition> q
format> q
```

## 디스크 레이블 만들기 및 검사

디스크 레이블은 일반적으로 시스템을 설치하거나 새 디스크를 사용할 때 지정됩니다. 디스크 레이블이 손상된 경우 디스크 레이블을 재지정해야 할 수도 있습니다. 예를 들어, 정전이 발생한 경우입니다.

format 유틸리티는 레이블이 없는 SCSI 디스크를 자동으로 구성하려고 시도합니다. format 유틸리티가 레이블이 없는 디스크를 자동으로 구성할 수 있는 경우 다음과 유사한 메시지를 표시합니다.

```
c2t3d0: configured with capacity of 136.73GB
```

### ▼ 디스크에 레이블을 지정하는 방법

다음 절차를 사용하여 다음을 수행할 수 있습니다.

- 2TB 크기 디스크의 경우 VTOC 또는 EFI 레이블로 디스크 레이블을 지정합니다.
- 2TB 크기보다 큰 디스크의 경우 EFI 레이블로 디스크 레이블을 지정합니다.

2TB보다 작은 디스크에 EFI 레이블을 지정하려는 경우 예 11-5를 참조하십시오.

**1 관리자로 전환합니다.**

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

**2 format 유틸리티를 호출합니다.**

```
# format
```

디스크의 번호 매기기 목록이 표시됩니다.

**3 레이블을 지정하려는 디스크의 번호를 입력합니다.**

```
Specify disk (enter its number):1
```

format 유틸리티가 디스크 유형을 인식하는 경우 다음 단계는 백업 레이블을 검색하여 디스크에 레이블을 지정하는 것입니다. 백업 레이블로 디스크 레이블을 지정하면 올바른 분할 정보, 디스크 유형 및 디스크 구조로 디스크에 레이블이 지정됩니다.

**4 다음 중 하나를 선택하여 디스크에 레이블을 지정합니다.**

- 디스크에 레이블이 없고 성공적으로 구성된 경우 5단계로 이동하여 디스크에 레이블을 지정합니다.

format 유틸리티가 디스크에 레이블을 지정할 것인지 확인합니다.

- 디스크에 레이블이 있지만 디스크 유형을 변경하려는 경우 또는 format 유틸리티가 디스크를 자동으로 구성할 수 없는 경우 6단계로 이동하여 디스크 유형을 설정하고 디스크에 레이블을 지정합니다.

**5 Label it now? 프롬프트에서 y를 입력하여 디스크에 레이블을 지정합니다.**

```
Disk not labeled. Label it now? y
```

이제 디스크에 레이블이 지정되었습니다. 10단계로 이동하여 format 유틸리티를 종료합니다.

**6 format> 프롬프트에서 type을 입력합니다.**

```
format> type
```

Available Drive Types(사용 가능한 드라이브 유형) 메뉴가 표시됩니다.

**7 가능한 디스크 유형 목록에서 디스크 유형을 선택합니다.**

```
Specify disk type (enter its number)[12]: 12
```

또는 0을 선택하여 SCSI-2 디스크를 자동으로 구성합니다.

**8 디스크에 레이블을 지정합니다. 디스크에 레이블이 지정되지 않은 경우 다음 메시지가 표시됩니다.**

```
Disk not labeled. Label it now? y
```

그렇지 않으면 다음 메시지가 표시됩니다.

```
Ready to label disk, continue? y
```

**9 디스크 레이블을 확인합니다.**

```
format> verify
```

**10 format 유틸리티를 종료합니다.**

```
format> q
#
```

**예 11-4 디스크에 레이블 지정**

다음 예에서는 36GB 디스크를 자동으로 구성하고 레이블을 지정하는 방법을 보여 줍니다.

```
# format
AVAILABLE DISK SELECTIONS:
  0. c0t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
    /pci@lc,600000/scsi@2/sd@0,0
  1. c0t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
    /pci@lc,600000/scsi@2/sd@1,0
  2. c0t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
    /pci@lc,600000/scsi@2/sd@2,0
  3. c0t3d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
    /pci@lc,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
c0t3d0: configured with capacity of 33.92GB
Disk not labeled. Label it now? yes
format> verify
format> q
```

**예 11-5 디스크에 EFI 레이블 지정**

다음 예에서는 `format-e` 명령을 사용하여 EFI 레이블로 디스크 레이블을 지정하는 방법을 보여 줍니다. 계층형 소프트웨어 제품이 EFI 레이블 디스크를 포함하는 시스템에서 계속 작동하는지 확인해야 합니다. EFI 레이블 제한 사항에 대한 일반적인 내용은 188 페이지 “EFI 디스크 레이블의 제한 사항”을 참조하십시오.

```
# format -e
Searching for disks...done
AVAILABLE DISK SELECTIONS:
  0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
    /pci@lc,600000/scsi@2/sd@0,0
  1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
    /pci@lc,600000/scsi@2/sd@1,0
  2. c2t2d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
    /pci@lc,600000/scsi@2/sd@2,0
  3. c2t3d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
    /pci@lc,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
```



```

selecting c2t3d0
[disk formatted]
format> Label
[0] SMI Label
[1] EFI Label
Specify Label type[0]: 1
Ready to label disk, continue? yes
format> quit

```

## 예 11-6 EFI 레이블 디스크를 SMI 레이블 디스크로 변경

다음 예에서는 `format -e` 명령을 사용하여 EFI 레이블 디스크를 ZFS 루트 풀에 사용할 수 있는 SMI 레이블 디스크로 변경하는 방법을 보여 줍니다.

x86 시스템에서는 먼저 EFI fdisk 분할 영역을 Solaris 분할 영역으로 변경해야 합니다. 예를 들면 다음과 같습니다.

```

# format -e
select disk ...
format> fdisk
FORMAT MENU:
    disk      - select a disk
    type      - select (define) a disk type
    Total disk size is 17833 cylinders
    Cylinder size is 16065 (512 byte) blocks

```

Partition	Status	Type	Cylinders		Length	%
			Start	End		
1		EFI	0	17833	17834	100

```

SELECT ONE OF THE FOLLOWING:
  1. Create a partition
  2. Specify the active partition
  3. Delete a partition
  4. Change between Solaris and Solaris2 Partition IDs
  5. Edit/View extended partitions
  6. Exit (update disk configuration and exit)
  7. Cancel (exit without updating disk configuration)
Enter Selection: 3
Specify the partition number to delete (or enter 0 to exit): 1
This will make all files and
programs in this partition inaccessible (type "y" or "n"). y
Enter Selection: 1
Select the partition type to create:
  1=SOLARIS2  2=UNIX      3=PCIXOS      4=Other      5=DOS12
  6=DOS16    7=DOSEXT    8=DOSBIG     9=DOS16LBA  A=x86 Boot
  B=Diagnostic C=FAT32   D=FAT32LBA  E=DOSEXTLBA F=EFI (Protective)
  G=EFI_SYS  0=Exit? 1
Specify the percentage of disk to use for this partition
(or type "c" to specify the size in cylinders). 100
Should this become the active partition? If yes, it will be activated
each time the computer is reset or turned on.

```

```
Please type "y" or "n". y
Enter Selection: 6
Partition 1 is now the active partition.
```

SPARC 시스템에서는 아래 단계를 수행합니다.

```
# format -e
Searching for disks...done
AVAILABLE DISK SELECTIONS:
    0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
       /pci@lc,600000/scsi@2/sd@0,0
    1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
       /pci@lc,600000/scsi@2/sd@1,0
    2. c2t2d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
       /pci@lc,600000/scsi@2/sd@2,0
    3. c2t3d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
       /pci@lc,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
selecting c2t0d0
[disk formatted]
format> label
[0] SMI Label
[1] EFI Label
Specify Label type[1]: 0
Ready to label disk, continue? yes
```

또한 기본 분할 영역 테이블을 검사하여 루트 폴 슬라이스에 최적화되었는지 확인해야 합니다. 즉, 디스크 공간의 대부분이 슬라이스 0에 있는지 확인합니다. 슬라이스 0의 디스크를 늘리려면 아래 단계를 참조하십시오.

```
format> partition
partition> print
Current partition table (default):
Total disk cylinders available: 14085 + 2 (reserved cylinders)
```

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	1 - 13	129.19MB	(13/0/0) 264576
1	swap	wu	14 - 26	129.19MB	(13/0/0) 264576
2	backup	wu	0 - 14086	136.71GB	(14087/0/0) 286698624
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	27 - 14084	136.43GB	(14058/0/0) 286108416
7	unassigned	wm	0	0	(0/0/0) 0
8	boot	wu	0 - 0	9.94MB	(1/0/0) 20352
9	alternates	wm	0	0	(0/0/0) 0

```
partition> modify
Select partitioning base:
    0. Current partition table (default)
    1. All Free Hog
Choose base (enter number) [0]? 1
```

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0	0	(0/0/0) 0
1	swap	wu	0	0	(0/0/0) 0
2	backup	wu	0 - 14084	136.69GB	(14085/0/0) 286657920

```

3 unassigned   wm      0          0          (0/0/0)      0
4 unassigned   wm      0          0          (0/0/0)      0
5 unassigned   wm      0          0          (0/0/0)      0
6      usr      wm      0          0          (0/0/0)      0
7 unassigned   wm      0          0          (0/0/0)      0
8      boot     wu      0 -      0          9.94MB      (1/0/0)      20352
9 alternates   wm      0          0          (0/0/0)      0

```

Do you wish to continue creating a new partition table based on above table[yes]? **yes**

Free Hog partition[6]? **0**

Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:

Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:

Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:

Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:

Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:

Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	1 - 14084	136.68GB	(14084/0/0) 286637568
1	swap	wu	0	0	(0/0/0) 0
2	backup	wu	0 - 14084	136.69GB	(14085/0/0) 286657920
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	0	0	(0/0/0) 0
7	unassigned	wm	0	0	(0/0/0) 0
8	boot	wu	0 - 0	9.94MB	(1/0/0) 20352
9	alternates	wm	0	0	(0/0/0) 0

Okay to make this the current partition table[yes]? **yes**

Enter table name (remember quotes): **"c2t0d0"**

Ready to label disk, continue? **yes**

partition> **quit**

format> **quit**

## ▼ 디스크 레이블 검사 방법

`prtvtoc` 명령을 사용하여 디스크 레이블 정보를 검사합니다. 디스크 레이블 및 `prtvtoc` 명령에 의해 표시되는 정보에 대한 자세한 설명은 10 장, “디스크 관리(개요)”를 참조하십시오.

### 1 관리자로 전환합니다.

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

### 2 디스크 레이블 정보를 표시합니다.

```
# prtvtoc /dev/rdisk/device-name
```

여기서 `device-name`은 검사하려는 원시 디스크 장치입니다.

## 예 11-7 디스크 레이블 검사

다음 예에서는 VTOC 레이블 디스크에 대한 디스크 레이블 정보를 보여 줍니다.

```
# prtvtoc /dev/rdisk/c2t3d0s0
* /dev/rdisk/c2t3d0s0 partition map
*
* Dimensions:
*   512 bytes/sector
*   848 sectors/track
*   24 tracks/cylinder
*  20352 sectors/cylinder
*  14089 cylinders
*  14087 accessible cylinders
*
* Flags:
*   1: unmountable
*  10: read-only
*
*
* Partition Tag  Flags      First Sector    Last Sector  Mount Directory
*         0     2    00          0 286698624 286698623
*         2     5    01          0 286698624 286698623
```

다음 예에서는 EFI 레이블 디스크에 대한 디스크 레이블 정보를 보여 줍니다.

```
# prtvtoc /dev/rdisk/c2t3d0s0
* /dev/rdisk/c2t3d0s0 partition map
*
* Dimensions:
*   512 bytes/sector
*   848 sectors/track
*   24 tracks/cylinder
*  20352 sectors/cylinder
*  14089 cylinders
*  14087 accessible cylinders
*
* Flags:
*   1: unmountable
*  10: read-only
*
*
* Partition Tag  Flags      First Sector    Last Sector  Mount Directory
*         0     2    00          0 286698624 286698623
*         2     5    01          0 286698624 286698623
```

## 손상된 디스크 레이블 복구

때때로 정전이나 시스템 오류로 인해 디스크 레이블을 인식할 수 없게 되는 경우가 있습니다. 디스크 레이블이 손상되었다고 해서 항상 슬라이스 정보나 디스크의 데이터를 다시 만들거나 복원해야 하는 것은 아닙니다.

손상된 디스크 레이블을 복구하는 첫번째 단계는 올바른 구조와 디스크 유형 정보로 디스크에 레이블을 지정하는 것입니다. 자동 구성이나 수동 디스크 유형 지정을 사용하여 일반적인 디스크 레이블 지정 방법을 통해 이 단계를 완료할 수 있습니다.

format 유틸리티가 디스크 유형을 인식하는 경우 다음 단계는 백업 레이블을 검색하여 디스크에 레이블을 지정하는 것입니다. 백업 레이블로 디스크 레이블을 지정하면 올바른 분할 정보, 디스크 유형 및 디스크 구조로 디스크에 레이블이 지정됩니다.

## ▼ 손상된 디스크 레이블 복구 방법

### 1 시스템을 단일 사용자 모드로 부트합니다.

필요한 경우 로컬 Oracle Solaris DVD 또는 단일 사용자 모드의 네트워크에서 시스템을 부트하여 디스크에 액세스합니다.

시스템 부트에 대한 자세한 내용은 [SPARC 플랫폼에서 Oracle Solaris 부트 및 종료](#) 또는 [x86 플랫폼에서 Oracle Solaris 부트 및 종료](#)를 참조하십시오.

### 2 디스크 레이블을 재지정합니다.

```
# format
```

format 유틸리티는 레이블이 없는 SCSI 디스크를 자동으로 구성하려고 시도합니다. format 유틸리티가 레이블이 없고 손상된 디스크를 구성할 수 있는 경우 다음 메시지를 표시합니다.

```
cwtxdy: configured with capacity of abcMB
```

그런 다음 format 유틸리티가 시스템에 있는 디스크의 번호 매기기 목록을 표시합니다.

### 3 복구해야 하는 디스크의 번호를 입력합니다.

```
Specify disk (enter its number): 1
```

### 4 다음 중 하나를 선택하여 디스크에 레이블을 지정하는 방법을 결정합니다.

- 디스크가 성공적으로 구성된 경우 5단계와 6단계를 수행합니다. 그런 다음 12단계로 이동합니다.
- 디스크가 성공적으로 구성되지 않은 경우 7-11 단계를 수행합니다. 그런 다음 12단계로 이동합니다.

### 5 백업 레이블을 검색합니다.

```
format> verify
Warning: Could not read primary label.
Warning: Check the current partitioning and 'label' the disk or
use the 'backup' command.
Backup label contents:
Volume name = < >
ascii name = <FUJITSU-MAP3147N SUN146G-0501>
pcyl      = 14089
ncyl     = 14087
acyl     = 2
nhead    = 24
nsect    = 848
```

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0 - 14086	136.71GB	(14087/0/0) 286698624
1	swap	wu	0	0	(0/0/0) 0
2	backup	wu	0 - 14086	136.71GB	(14087/0/0) 286698624
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	0	0	(0/0/0) 0
7	unassigned	wm	0	0	(0/0/0) 0

- 6 **format** 유틸리티가 백업 레이블을 찾을 수 있고 백업 레이블 내용이 만족스러운 경우 **backup** 명령을 사용하여 백업 레이블로 디스크 레이블을 지정합니다.

```
format> backup
Disk has a primary label, still continue? y
```

```
Searching for backup labels...found.
Restoring primary label
```

디스크 레이블이 복구되었습니다. 12단계로 이동합니다.

- 7 **format** 유틸리티가 디스크를 자동으로 구성하지 못한 경우 **type** 명령을 사용하여 디스크 유형을 지정합니다.

```
format> type
```

Available Drive Types(사용 가능한 드라이브 유형) 메뉴가 표시됩니다.

- 8 0을 선택하여 디스크를 자동으로 구성합니다. 또는 가능한 디스크 유형 목록에서 디스크 유형을 선택합니다.

```
Specify disk type (enter its number)[12]: 12
```

- 9 디스크가 성공적으로 구성된 경우 **format** 유틸리티가 디스크에 레이블을 지정할 것인지 확인할 때 **no**로 응답합니다.

```
Disk not labeled. Label it now? no
```

- 10 **verify** 명령을 사용하여 백업 레이블을 검색합니다.

```
format> verify
Warning: Could not read primary label.
Warning: Check the current partitioning and 'label' the disk
or use the 'backup' command.
```

```
.
.
.
```

- 11 **format** 유틸리티가 백업 레이블을 찾을 수 있고 백업 레이블 내용이 만족스러운 경우 **backup** 명령을 사용하여 백업 레이블로 디스크 레이블을 지정합니다.

```
format> backup
Disk has a primary label, still continue? y
Searching for backup labels...found.
Restoring primary label
```

디스크 레이블이 복구되었습니다.

**12 format 유틸리티를 종료합니다.**

```
format> q
```

**13 복구된 디스크의 파일 시스템을 확인합니다.**

ZFS 파일 시스템에 `zpool scrub` 명령을 사용하는 방법에 대한 자세한 내용은 [zpool\(1M\)](#)을 참조하십시오.

## 타사 디스크 추가

Oracle Solaris OS는 많은 타사 디스크를 지원합니다. 하지만 디스크를 인식하기 위해 장치 드라이버를 제공해야 할 수도 있습니다. 디스크를 추가하는 기타 옵션은 다음과 같습니다.

- SCSI 디스크를 추가하는 경우 `format` 유틸리티의 자동 구성 기능을 시도할 수 있습니다.
- PCI, SCSI 또는 USB 디스크는 핫 플러그를 시도할 수 있습니다. 자세한 내용은 [5 장, “장치 관리\(개요/작업\)”](#)를 참조하십시오.

---

**주** - Oracle은 `format` 유틸리티가 모든 타사 디스크 드라이버에서 제대로 작동할 것을 보장하지 않습니다. 디스크 드라이버가 Solaris `format` 유틸리티와 호환되지 않는 경우 디스크 드라이브 공급업체에서 사용자 정의 디스크 포맷 프로그램을 제공해야 합니다.

---

일반적으로 `format` 유틸리티를 호출할 때 소프트웨어 지원이 없고 디스크 유형을 인식할 수 없는 것을 발견합니다.

시스템 디스크나 보조 디스크를 추가하는 올바른 구성 절차는 [12 장, “SPARC: 디스크 설정\(작업\)”](#) 또는 [13 장, “x86: 디스크 설정\(작업\)”](#)을 참조하십시오.





## SPARC: 디스크 설정(작업)

이 장에서는 SPARC 시스템에서 디스크를 설정하는 방법에 대해 설명합니다.

SPARC 시스템의 디스크 설정과 관련된 절차에 대한 자세한 내용은 217 페이지 “SPARC: ZFS 파일 시스템에 사용할 디스크 설정(작업 맵)”을 참조하십시오.

디스크 관리에 대한 개요 정보는 10 장, “디스크 관리(개요)”를 참조하십시오. x86 기반 시스템의 디스크 설정에 대한 단계별 지침은 13 장, “x86: 디스크 설정(작업)”을 참조하십시오.

### SPARC: ZFS 파일 시스템에 사용할 디스크 설정(작업 맵)

다음 작업 맵은 SPARC 기반 시스템에서 ZFS 루트 파일 시스템에 사용할 ZFS 루트 풀 디스크 또는 루트가 아닌 ZFS 풀 디스크를 설정하는 절차를 식별합니다.

작업	설명	수행 방법
1. ZFS 루트 파일 시스템에 사용할 디스크를 설정합니다.	<b>ZFS 루트 파일 시스템에 사용할 디스크</b>  새 디스크를 연결하거나 기존 루트 풀 디스크를 교체하고 로컬 또는 원격 Oracle Solaris DVD에서 부트합니다.	219 페이지 “SPARC: ZFS 루트 파일 시스템에 사용할 디스크 설정 방법”
2. ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만듭니다.	ZFS 루트 풀에 사용하려는 디스크에 대해 디스크 슬라이스를 만듭니다. 이것은 오래된 부트 제한 사항입니다.	220 페이지 “SPARC: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만드는 방법”

작업	설명	수행 방법
3. 필요한 경우 ZFS 루트 파일 시스템의 부트 블록을 설치합니다.	zpool replace 명령을 사용하여 루트 풀에 사용하려는 디스크를 교체하는 경우 시스템이 교체 디스크에서 부트될 수 있도록 부트 블록을 수동으로 설치해야 합니다.	223 페이지 “SPARC: ZFS 루트 파일 시스템의 부트 블록 설치 방법”
4. ZFS 파일 시스템에 사용할 디스크를 설정합니다.	ZFS 파일 시스템에 사용할 디스크  ZFS 파일 시스템에 사용할 디스크를 설정합니다.	224 페이지 “SPARC: ZFS 파일 시스템에 사용할 디스크 설정 방법”

## SPARC: ZFS 파일 시스템에 사용할 디스크 설정

디스크 설정 방법을 설명하는 절차를 ZFS 파일 시스템에서 사용할 수 있지만 ZFS 파일 시스템은 디스크 또는 디스크 슬라이스에 직접 매핑되지 않습니다. ZFS 파일 시스템을 만들기 전에 ZFS 저장소 풀을 만들어야 합니다. 자세한 내용은 [Oracle Solaris 관리: ZFS 파일 시스템](#)을 참조하십시오.

루트 풀에는 Oracle Solaris OS 부트에 사용되는 루트 파일 시스템이 포함됩니다. 루트 풀 디스크가 손상되고 루트 풀이 미러링되지 않은 경우 시스템이 부트되지 않을 수 있습니다. 루트 풀 디스크가 손상된 경우 다음 두 가지 복구 방법이 있습니다.

- 전체 Oracle Solaris OS를 재설치할 수 있습니다.
- 또는 루트 풀 디스크를 교체하고 스냅샷이나 백업 매체에서 파일 시스템을 복원할 수 있습니다. 중복 루트 풀을 만들면 하드웨어 장애로 인한 시스템 작동 중지 시간을 줄일 수 있습니다. 지원되는 중복 루트 풀 구성은 미러링된 루트 풀뿐입니다.

루트가 아닌 풀에 사용되는 디스크에는 대체로 사용자 또는 데이터 파일을 위한 공간이 있습니다. 추가 디스크 공간을 위해 루트 풀이나 루트가 아닌 풀에 다른 디스크를 연결 또는 추가할 수 있습니다. 또는 다음과 같은 방법으로 풀의 손상된 디스크를 교체할 수 있습니다.

- 모든 장치가 현재 온라인 상태인 경우 비중복 풀의 디스크를 교체할 수 있습니다.
- 다른 장치에 충분한 중복이 있는 경우 중복 풀의 디스크를 교체할 수 있습니다.
- 미러링된 루트 풀에서는 디스크를 교체하거나, 디스크를 연결하고 실패한 디스크나 작은 디스크를 분리하여 풀의 크기를 늘릴 수 있습니다.

일반적으로 시스템의 디스크 설정은 하드웨어에 따라 달라지므로 시스템에 디스크를 추가하거나 교체하는 경우 하드웨어 설명서를 참조하십시오. 기존 제어기에 디스크를 추가해야 하는 경우 시스템이 핫 플러그를 지원하면 디스크를 빈 슬롯에 넣기만 하면 됩니다. 새 제어기를 구성해야 하는 경우 [77 페이지 “동적 재구성 및 핫 플러그 기능 사용”](#)을 참조하십시오.

## ▼ SPARC: ZFS 루트 파일 시스템에 사용할 디스크 설정 방법

디스크 교체에 대한 자세한 내용은 하드웨어 설치 설명서를 참조하십시오.

- 1 필요한 경우 시스템에서 손상된 디스크의 연결을 해제합니다.
- 2 필요한 경우 교체 디스크를 시스템에 연결하고 디스크의 물리적 연결을 확인합니다.
- 3 로컬 Oracle Solaris DVD에서 부트하는지 또는 네트워크의 원격 Oracle Solaris DVD에서 부트하는지에 따라 다음 표의 지침을 따르십시오.

부트 유형	작업
로컬 드라이브의 Oracle Solaris DVD에서 부트	1. Oracle Solaris DVD가 드라이브에 들어 있는지 확인합니다. 2. 매체에서 단일 사용자 모드로 부트합니다. <b>ok boot cdrom -s</b>
네트워크에서 부트	네트워크에서 단일 사용자 모드로 부트합니다. <b>ok boot net -s</b>

몇 분 후에 루트 프롬프트(#)가 표시됩니다.

### 자세한 정보 SPARC 루트 파일 시스템에 사용할 디스크를 설정한 후...

디스크를 연결하거나 교체한 후 슬라이스를 만들고 디스크 레이블을 업데이트할 수 있습니다. 220 페이지 “SPARC: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만드는 방법”으로 이동합니다.

## SPARC: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스 만들기

ZFS 루트 풀에 사용하려는 디스크에 대해 디스크 슬라이스를 만들어야 합니다. 이것은 오래된 부트 제한 사항입니다. 다음과 같은 루트 풀 디스크 요구 사항을 검토합니다.

- 디스크 슬라이스와 SMI(VTOC) 레이블을 포함해야 합니다.
- EFI 레이블은 루트 풀 디스크에 사용할 수 없습니다.
- 단일 디스크이거나 미러링된 구성의 일부여야 합니다. 비중복 구성과 RAIDZ 구성은 루트 풀에 사용할 수 없습니다.
- /var을 제외하고 OS 이미지에 속한 루트 파일 시스템의 모든 하위 디렉토리는 루트 파일 시스템과 동일한 데이터 집합에 있어야 합니다.

- 스왑 및 덤프 장치를 제외한 모든 Solaris OS 구성 요소는 루트 풀에 상주해야 합니다.

일반적으로 대부분의 디스크 공간을 슬라이스 0에 포함하여 디스크 슬라이스를 만들어야 합니다. 디스크에서 여러 슬라이스를 사용하고 해당 디스크를 여러 운영 체제에서 공유하거나 다른 ZFS 저장소 풀 또는 저장소 풀 구성 요소와 공유하는 것은 권장되지 않습니다.

## ▼ SPARC: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만드는 방법

일반적으로 루트 풀 디스크는 시스템을 설치할 때 자동으로 설치됩니다. 루트 풀 디스크를 교체하거나 새 디스크를 미러링된 루트 풀 디스크로 연결해야 하는 경우 아래 단계를 참조하십시오.

### 1 관리자로 전환합니다.

### 2 필요한 경우 실패한 디스크를 오프라인으로 전환하고 구성을 해제합니다.

일부 하드웨어의 경우 실패한 디스크를 교체하기 위해 `zpool replace` 작업을 시도하기 전에 디스크를 오프라인 상태로 만들고 구성을 해제해야 합니다. 예를 들면 다음과 같습니다.

```
# zpool offline rpool c2t1d0s0
# cfgadm -c unconfigure c2::disk/c2t1d0
```

### 3 필요한 경우 새 디스크나 교체 디스크를 시스템에 물리적으로 연결합니다.

a. 실패한 디스크를 물리적으로 제거합니다.

b. 교체 디스크를 물리적으로 넣습니다.

c. 필요한 경우 교체 디스크를 구성합니다. 예를 들면 다음과 같습니다.

```
# cfgadm -c configure c2::disk/c2t1d0
```

교체 디스크를 넣은 후 해당 디스크를 재구성할 필요가 없는 하드웨어도 있습니다.

### 4 `format` 출력을 검토하여 디스크에 액세스할 수 있는지 확인합니다.

예를 들어, `format` 명령이 이 시스템에 4개 디스크가 연결되어 있다고 표시합니다.

```
# format -e
AVAILABLE DISK SELECTIONS:
  0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
    /pci@1c,600000/scsi@2/sd@0,0
  1. c2t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
    /pci@1c,600000/scsi@2/sd@1,0
  2. c2t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
    /pci@1c,600000/scsi@2/sd@2,0
  3. c2t3d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
    /pci@1c,600000/scsi@2/sd@3,0
```

## 5 ZFS 루트 풀에 사용할 디스크를 선택합니다.

## 6 분할 영역(슬라이스) 정보를 표시하여 디스크에 SMI 레이블이 있는지 확인합니다.

예를 들어, c2t1d0에 대한 분할 영역(슬라이스) 출력에 이 디스크는 첫번째 섹터와 마지막 섹터를 식별하므로 EFI 레이블이 있다고 표시됩니다.

```
Specify disk (enter its number): 1
selecting c2t1d0
[disk formatted]
format> p
PARTITION MENU:
    0      - change '0' partition
    1      - change '1' partition
    2      - change '2' partition
    3      - change '3' partition
    4      - change '4' partition
    5      - change '5' partition
    6      - change '6' partition
expand - expand label to use whole disk
select - select a predefined table
modify - modify a predefined partition table
name    - name the current table
print   - display the current table
label   - write partition map and label to the disk
!<cmd> - execute <cmd>, then return
quit
partition> p
Current partition table (original):
Total disk sectors available: 71116508 + 16384 (reserved sectors)

Part      Tag      Flag      First Sector      Size      Last Sector
  0        usr      wm          256      33.91GB      71116541
  1 unassigned  wm           0           0           0
  2 unassigned  wm           0           0           0
  3 unassigned  wm           0           0           0
  4 unassigned  wm           0           0           0
  5 unassigned  wm           0           0           0
  6 unassigned  wm           0           0           0
  8 reserved   wm      71116542      8.00MB      71132925

partition>
```

## 7 디스크에 EFI 레이블이 있는 경우 SMI 레이블로 디스크 레이블을 재지정합니다.

예를 들어, c2t1d0 디스크 레이블이 SMI 레이블로 재지정되었지만 기본 분할 영역 테이블이 최적 슬라이스 구성을 제공하지 않습니다.

```
partition> label
[0] SMI Label
[1] EFI Label
Specify Label type[1]: 0
Auto configuration via format.dat[no]?
Auto configuration via generic SCSI-2[no]?
partition> p
Current partition table (default):
Total disk cylinders available: 24620 + 2 (reserved cylinders)
```

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0 - 90	128.37MB	(91/0/0) 262899
1	swap	wu	91 - 181	128.37MB	(91/0/0) 262899
2	backup	wu	0 - 24619	33.92GB	(24620/0/0) 71127180
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	182 - 24619	33.67GB	(24438/0/0) 70601382
7	unassigned	wm	0	0	(0/0/0) 0

```
partition>
```

## 8 ZFS 루트 풀 디스크에 대해 최적 슬라이스 구성을 만듭니다.

할당되지 않은 모든 디스크 공간이 슬라이스 0에 수집되도록 Free Hog 분할 영역을 설정합니다. 그런 다음 각 슬라이스 크기 필드에서 return 키를 눌러 하나의 큰 슬라이스 0을 만듭니다.

```
partition> modify
Select partitioning base:
  0. Current partition table (default)
  1. All Free Hog
Choose base (enter number) [0]? 1
```

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0	0	(0/0/0) 0
1	swap	wu	0	0	(0/0/0) 0
2	backup	wu	0 - 24619	33.92GB	(24620/0/0) 71127180
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	0	0	(0/0/0) 0
7	unassigned	wm	0	0	(0/0/0) 0

```
Do you wish to continue creating a new partition
table based on above table[yes]?
```

```
Free Hog partition[6]? 0
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:
```

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0 - 24619	33.92GB	(24620/0/0) 71127180
1	swap	wu	0	0	(0/0/0) 0
2	backup	wu	0 - 24619	33.92GB	(24620/0/0) 71127180
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	0	0	(0/0/0) 0
7	unassigned	wm	0	0	(0/0/0) 0

```
Okay to make this the current partition table[yes]?
Enter table name (remember quotes): "c2t1d0"
```

```
Ready to label disk, continue? yes
partition> quit
format> quit
```

**9 실패한 디스크가 교체되었음을 ZFS에 알립니다.**

```
# zpool replace rpool c2t1d0s0
# zpool online rpool c2t1d0s0
```

교체 디스크를 넣은 후 해당 디스크를 온라인 상태로 전환할 필요가 없는 하드웨어도 있습니다.

새 디스크를 연결하여 미러링된 루트 풀을 만들거나 큰 디스크를 연결하여 작은 디스크를 교체하는 경우 다음과 유사한 구문을 사용합니다.

```
# zpool attach rpool c0t0d0s0 c1t0d0s0
```

**10 루트 풀 디스크를 새 디스크로 교체하는 경우 새 디스크나 교체 디스크가 리실버링된 후 부트블록을 적용합니다.**

예를 들면 다음과 같습니다.

```
# zpool status rpool
installboot -F zfs /usr/platform/'uname -i'/lib/fs/zfs/bootblk /dev/rdisk/c2t1d0s0
```

**11 새 디스크에서 부트할 수 있는지 확인합니다.****12 시스템이 새 디스크에서 부트되면 이전 디스크를 분리합니다.**

이 단계는 새 디스크를 연결하여 실패한 디스크나 작은 디스크를 교체하는 경우에만 필요합니다.

```
# zpool detach rpool c0t0d0s0
```

**13 eeprom 명령 또는 SPARC 부트 PROM의 setenv 명령을 사용하여 새 디스크로부터 자동으로 부트하도록 시스템을 설정합니다.****자세한 정보 ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만든 후...**

ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만든 후 루트 풀 스냅샷을 복원하여 루트 풀을 복구해야 하는 경우 [Oracle Solaris 관리: ZFS 파일 시스템의 “ZFS 루트 풀의 디스크 교체 방법”](#)을 참조하십시오.

**▼ SPARC: ZFS 루트 파일 시스템의 부트블록 설치 방법****1 관리자로 전환합니다.****2 ZFS 루트 파일 시스템의 부트블록을 설치합니다.**

```
# installboot -F zfs /usr/platform/'uname -i'/lib/fs/zfs/bootblk
/dev/rdisk/cwtxdys0
```

자세한 내용은 [installboot\(1M\)](#)를 참조하십시오.

- 레벨 3을 실행하려면 시스템을 재부트하여 부트 블록이 설치되었는지 확인합니다.

```
# init 6
```

### 예 12-1 SPARC: ZFS 루트 파일 시스템의 부트 블록 설치

루트 풀에 사용하려는 디스크를 물리적으로 교체한 후 Oracle Solaris OS를 재설치하거나 루트 풀에 사용할 새 디스크를 연결하면 부트 블록이 자동으로 설치됩니다. `zpool replace` 명령을 사용하여 루트 풀에 사용하려는 디스크를 교체하는 경우 시스템이 교체 디스크에서 부트될 수 있도록 부트 블록을 수동으로 설치해야 합니다.

다음 예에서는 ZFS 루트 파일 시스템의 부트 블록을 설치하는 방법을 보여 줍니다.

```
# installboot -F zfs /usr/platform/'uname -i'/lib/fs/zfs/bootblk /dev/rdsk/c0t1d0s0
```

## ▼ SPARC: ZFS 파일 시스템에 사용할 디스크 설정 방법

루트가 아닌 ZFS 파일 시스템에 사용할 디스크를 설정하는 경우 풀을 만들거나 디스크를 풀에 추가하면 자동으로 디스크 레이블이 재지정됩니다. 전체 디스크를 사용하여 풀을 만드는 경우 또는 전체 디스크를 ZFS 저장소 풀에 추가하면 EFI 레이블이 적용됩니다. EFI 디스크 레이블에 대한 자세한 내용은 187 페이지 “EFI 디스크 레이블”을 참조하십시오.

일반적으로 대부분의 현대적 버스 유형은 핫 플러그를 지원합니다. 즉, 빈 슬롯에 디스크를 삽입할 수 있고 시스템에서 이러한 디스크를 인식합니다. 핫 플러그 장치에 대한 자세한 내용은 6장, “동적으로 장치 구성(작업)”을 참조하십시오.

- 관리자로 전환합니다.
- 디스크를 시스템에 연결하고 디스크의 물리적 연결을 확인합니다.  
자세한 내용은 디스크의 하드웨어 설치 설명서를 참조하십시오.
- 필요한 경우 실패한 디스크를 오프라인으로 전환하고 구성을 해제합니다.

일부 하드웨어의 경우 실패한 디스크를 교체하기 위해 `zpool replace` 작업을 시도하기 전에 디스크를 오프라인 상태로 만들고 구성을 해제해야 합니다. 예를 들면 다음과 같습니다.

```
# zpool offline tank c1t1d0
# cfgadm -c unconfigure c1::dsk/c1t1d0
<Physically remove failed disk c1t1d0>
<Physically insert replacement disk c1t1d0>
# cfgadm -c configure c1::dsk/c1t1d0
```

교체 디스크를 넣은 후 해당 디스크를 재구성할 필요가 없는 하드웨어도 있습니다.



**4 새 디스크가 인식되는지 확인합니다.**

`format` 유틸리티의 출력을 검토하여 디스크가 AVAILABLE DISK SELECTIONS 아래에 나열되는지 확인합니다. 그런 다음 `format` 유틸리티를 종료합니다.

```
# format
```

**5 필요한 경우 실패한 디스크가 교체 되었음을 ZFS에 알립니다.**

```
# zpool replace tank c1t1d0
```

```
# zpool online tank c1t1d0
```

새 디스크가 리실버링되는지 확인합니다.

```
# zpool status tank
```

**6 필요한 경우 기존 ZFS 저장소 풀에 새 디스크를 연결합니다.**

예를 들면 다음과 같습니다.

```
# zpool attach tank mirror c1t0d0 c2t0d0
```

새 디스크가 리실버링되는지 확인합니다.

```
# zpool status tank
```

자세한 내용은 [Oracle Solaris 관리: ZFS 파일 시스템의 4 장](#), “Oracle Solaris ZFS 저장소 풀 관리”를 참조하십시오.



## x86: 디스크 설정(작업)

이 장에서는 x86 기반 시스템에서 디스크를 설정하는 방법에 대해 설명합니다.

x86 기반 시스템의 디스크 설정과 관련된 절차에 대한 자세한 내용은 다음 섹션을 참조하십시오.

- 217 페이지 “SPARC: ZFS 파일 시스템에 사용할 디스크 설정(작업 맵)”
- 235 페이지 “Solaris fdisk 분할 영역 만들기 및 변경”

디스크 관리에 대한 개요 정보는 10 장, “디스크 관리(개요)”를 참조하십시오. SPARC 기반 시스템의 디스크 설정에 대한 단계별 지침은 12 장, “SPARC: 디스크 설정(작업)”을 참조하십시오.

### x86: ZFS 파일 시스템에 사용할 디스크 설정(작업 맵)

다음 작업 맵은 x86 기반 시스템에서 ZFS 루트 파일 시스템에 사용할 ZFS 루트 풀 디스크를 설정하는 절차를 식별합니다.

작업	설명	수행 방법
1. ZFS 루트 파일 시스템에 사용할 디스크를 설정합니다.	<b>ZFS 루트 파일 시스템에 사용할 디스크</b> 새 디스크를 연결하거나 기존 루트 풀 디스크를 교체하고 로컬 또는 원격 Oracle Solaris DVD에서 부트합니다.	229 페이지 “x86: ZFS 루트 파일 시스템에 사용할 디스크 설정 방법”
2. 필요한 경우 fdisk 분할 영역을 만들거나 변경합니다.	디스크에 유효한 Solaris fdisk 분할 영역이 포함되어야 합니다.	235 페이지 “Solaris fdisk 분할 영역 만들기 및 변경”

작업	설명	수행 방법
3. ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만듭니다.	ZFS 루트 풀에 사용하려는 디스크에 대해 디스크 슬라이스를 만듭니다. 이것은 오래된 부트 제한 사항입니다.	236 페이지 “x86: Solaris fdisk 분할 영역을 만드는 방법” 및 230 페이지 “x86: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만드는 방법”
4. ZFS 루트 파일 시스템의 부트 블록을 설치합니다.	zpool replace 명령을 사용하여 루트 풀에 사용하려는 디스크를 교체하는 경우 시스템이 교체 디스크에서 부트될 수 있도록 부트 블록을 수동으로 설치해야 합니다.	234 페이지 “x86: ZFS 루트 파일 시스템의 부트 블록 설치 방법”
5. ZFS 파일 시스템에 사용할 디스크를 설정합니다.	ZFS 파일 시스템에 사용할 디스크 디스크를 연결합니다.	234 페이지 “x86: ZFS 파일 시스템에 사용할 디스크 설정 방법”

## x86: ZFS 파일 시스템에 사용할 디스크 설정

디스크를 설정하고 fdisk 분할 영역을 만드는 방법을 설명하는 절차를 ZFS 파일 시스템에서 사용할 수 있지만 ZFS 파일 시스템은 디스크 또는 디스크 슬라이스에 직접 매핑되지 않습니다. ZFS 파일 시스템을 만들기 전에 ZFS 저장소 풀을 만들어야 합니다. 자세한 내용은 [Oracle Solaris 관리: ZFS 파일 시스템](#)을 참조하십시오.

루트 풀에는 Oracle Solaris OS 부트에 사용되는 루트 파일 시스템이 포함됩니다. 루트 풀 디스크가 손상되고 루트 풀이 미러링되지 않은 경우 시스템이 부트되지 않을 수 있습니다. 루트 풀 디스크가 손상된 경우 다음 두 가지 복구 방법이 있습니다.

- 전체 Oracle Solaris OS를 재설치할 수 있습니다.
- 또는 루트 풀 디스크를 교체하고 스냅샷이나 백업 매체에서 파일 시스템을 복원할 수 있습니다. 중복 루트 풀을 만들면 하드웨어 장애로 인한 시스템 작동 중지 시간을 줄일 수 있습니다. 지원되는 중복 루트 풀 구성은 미러링된 루트 풀뿐입니다.

루트가 아닌 풀에 사용되는 디스크는 대체로 사용자 또는 데이터 파일용 공간을 포함합니다. 추가 디스크 공간을 위해 루트 풀이나 루트가 아닌 풀에 다른 디스크를 연결 또는 추가할 수 있습니다. 또는 다음과 같은 방법으로 풀의 손상된 디스크를 교체할 수 있습니다.

- 모든 장치가 현재 온라인 상태인 경우 비중복 풀의 디스크를 교체할 수 있습니다.
- 다른 장치에 충분한 중복이 있는 경우 중복 풀의 디스크를 교체할 수 있습니다.
- 미러링된 루트 풀에서는 디스크를 교체하거나, 디스크를 연결하고 실패한 디스크나 작은 디스크를 분리하여 풀의 크기를 늘릴 수 있습니다.

일반적으로 시스템의 디스크 설정은 하드웨어에 따라 달라지므로 시스템에 디스크를 추가하거나 교체하는 경우 하드웨어 설명서를 참조하십시오. 기존 제어기에 디스크를 추가해야 하는 경우 시스템이 핫 플러그를 지원하면 디스크를 빈 슬롯에 넣기만 하면 됩니다. 새 제어기를 구성해야 하는 경우 77 페이지 “동적 재구성 및 핫 플러그 기능 사용”을 참조하십시오.

## ▼ x86: ZFS 루트 파일 시스템에 사용할 디스크 설정 방법

디스크 교체에 대한 자세한 내용은 하드웨어 설치 설명서를 참조하십시오.

- 1 필요한 경우 시스템에서 손상된 디스크의 연결을 해제합니다.
- 2 교체 디스크를 시스템에 연결하고 디스크의 물리적 연결을 확인합니다.
- 3 로컬 Oracle Solaris DVD에서 부트하는지 또는 네트워크의 원격 Oracle Solaris DVD에서 부트하는지에 따라 다음 표의 지침을 따르십시오.

부트 유형	작업
로컬 드라이브의 Oracle Solaris DVD에서 부트	1. Oracle Solaris DVD가 드라이브에 들어 있는지 확인합니다. 2. 매체에서 부트하는 옵션을 선택합니다.
네트워크에서 부트	3. 네트워크에서 부트하는 옵션을 선택합니다.

### 자세한 정보 ZFS 루트 파일 시스템에 사용할 디스크를 설정한 후...

디스크를 연결하거나 교체한 후 fdisk 분할 영역을 만듭니다. 236 페이지 “x86: Solaris fdisk 분할 영역을 만드는 방법”으로 이동합니다.

## x86: ZFS 루트 파일 시스템에 사용할 디스크 슬라이스 만들기

ZFS 루트 풀에 사용하려는 디스크에 대해 디스크 슬라이스를 만들어야 합니다. 이것은 오래된 부트 제한 사항입니다. 다음과 같은 루트 풀 디스크 요구 사항을 검토합니다.

- 디스크 슬라이스와 SMI(VTOC) 레이블을 포함해야 합니다.
- EFI 레이블은 루트 풀 디스크에 사용할 수 없습니다.
- x86 시스템의 루트 풀 디스크에는 fdisk 분할 영역이 포함되어야 합니다.
- 단일 디스크이거나 미러링된 구성의 일부여야 합니다. 비중복 구성과 RAIDZ 구성은 루트 풀에 사용할 수 없습니다.
- /var을 제외하고 OS 이미지에 속한 루트 파일 시스템의 모든 하위 디렉토리는 루트 파일 시스템과 동일한 데이터 집합에 있어야 합니다.
- 스왑 및 덤프 장치를 제외한 모든 Solaris OS 구성 요소는 루트 풀에 상주해야 합니다.

x86 기반 시스템에서는 먼저 fdisk 분할 영역을 만들어야 합니다. 그런 다음 대부분의 디스크 공간을 슬라이스 0에 포함하여 디스크 슬라이스를 만듭니다.

디스크에서 여러 슬라이스를 사용하고 해당 디스크를 여러 운영 체제에서 공유하거나 다른 ZFS 저장소 풀 또는 저장소 풀 구성 요소와 공유하는 것은 권장되지 않습니다.

## ▼ x86:ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만드는 방법

일반적으로 루트 풀 디스크는 시스템을 설치할 때 자동으로 설치됩니다. 루트 풀 디스크를 교체하거나 새 디스크를 미러링된 루트 풀 디스크로 연결해야 하는 경우 아래 단계를 참조하십시오.

fdisk 분할 영역에 대한 전체 설명은 235 페이지 “x86: fdisk 분할 영역을 만드는 방법에 대한 지침”을 참조하십시오.

### 1 관리자로 전환합니다.

### 2 필요한 경우 실패한 디스크를 오프라인으로 전환하고 구성을 해제합니다.

일부 하드웨어의 경우 실패한 디스크를 교체하기 위해 zpool replace 작업을 시도하기 전에 디스크를 오프라인 상태로 만들고 구성을 해제해야 합니다. 예를 들면 다음과 같습니다.

```
# zpool offline rpool c2t1d0s0
# cfgadm -c unconfigure c2::disk/c2t1d0
```

### 3 필요한 경우 새 디스크나 교체 디스크를 시스템에 물리적으로 연결합니다.

a. 실패한 디스크를 물리적으로 제거합니다.

b. 교체 디스크를 물리적으로 넣습니다.

c. 필요한 경우 교체 디스크를 구성합니다. 예를 들면 다음과 같습니다.

```
# cfgadm -c configure c2::disk/c2t1d0
```

교체 디스크를 넣은 후 해당 디스크를 재구성할 필요가 없는 하드웨어도 있습니다.

### 4 format 출력을 검토하여 디스크에 액세스할 수 있는지 확인합니다.

예를 들어, format 명령이 이 시스템에 4개 디스크가 연결되어 있다고 표시합니다.

```
# format -e
AVAILABLE DISK SELECTIONS:
  1. c8t0d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
    /pci@0,0/pci10de,375ef/pci108e,286@0/disk@0,0
  2. c8t1d0 <Sun-STK RAID INT-V1.0-136.61GB>
    /pci@0,0/pci10de,375ef/pci108e,286@0/disk@1,0
  3. c8t2d0 <Sun-STK RAID INT-V1.0-136.61GB>
    /pci@0,0/pci10de,375ef/pci108e,286@0/disk@2,0
  4. c8t3d0 <Sun-STK RAID INT-V1.0-136.61GB>
    /pci@0,0/pci10de,375ef/pci108e,286@0/disk@3,0
```

## 5 ZFS 루트 풀에 사용할 디스크를 선택합니다.

```
Specify disk (enter its number): 1
selecting c8t0d0
[disk formatted]
.
.
.
format>
```

## 6 fdisk 분할 영역의 상태를 검토합니다.

- 디스크에 fdisk 분할 영역이 없는 경우 다음과 유사한 메시지가 표시됩니다.

```
format> fdisk
No Solaris fdisk partition found.
```

이 경우 4단계로 이동하여 fdisk 분할 영역을 만듭니다.

- 디스크에 EFI fdisk 또는 다른 분할 영역 유형이 있는 경우 5단계로 이동하여 Solaris fdisk 분할 영역을 만듭니다.
- 디스크에 Solaris fdisk 분할 영역이 있는 경우 6단계로 이동하여 루트 풀에 사용할 디스크 슬라이스를 만듭니다.

## 7 필요한 경우 fdisk 옵션을 선택하여 Solaris fdisk 분할 영역을 만듭니다.

```
format> fdisk
No fdisk table exists. The default partition for the disk is:
```

```
a 100% "SOLARIS System" partition
```

```
Type "y" to accept the default partition, otherwise type "n" to edit the
partition table. y
```

그런 다음 6단계로 이동하여 루트 풀에 사용할 디스크 슬라이스를 만듭니다.

## 8 디스크에 EFI fdisk 분할 영역이 있는 경우 Solaris fdisk 분할 영역을 만들어야 합니다.

format 유틸리티를 사용하여 디스크의 분할 영역 테이블을 인쇄하고 분할 영역 테이블이 첫번째 섹터와 크기를 참조하는 경우 EFI 분할 영역입니다. 다음과 같이 Solaris fdisk 분할 영역을 만들어야 합니다.

- 포맷 옵션에서 fdisk를 선택합니다.

```
# format -e c8t0d0
selecting c8t0d0
[disk formatted]
format> fdisk
```

- 옵션 3, Delete a partition을 선택하여 기존 EFI 분할 영역을 삭제합니다.

```
Enter Selection: 3
Specify the partition number to delete (or enter 0 to exit): 1
Are you sure you want to delete partition 1? This will make all files and
programs in this partition inaccessible (type "y" or "n"). y
```

```
Partition 1 has been deleted.
```

- 옵션 1, Create a partition을 선택하여 새 Solaris 분할 영역을 만듭니다.

```
Enter Selection: 1
Select the partition type to create: 1
Specify the percentage of disk to use for this partition
(or type "c" to specify the size in cylinders). 100
Should this become the active partition? If yes, it will be activated
each time the computer is reset or turned on.
Please type "y" or "n". y
Partition 1 is now the active partition.
```

- 디스크 구성을 업데이트하고 종료합니다.

```
Enter Selection: 6
format>
```

- SMI 분할 영역 테이블을 표시합니다. 기본 분할 영역 테이블을 적용하면 슬라이스 0의 크기가 0이거나 너무 작을 수 있습니다. 다음 단계를 참조하십시오.

```
format> partition
partition> print
```

## 9 분할 영역(슬라이스) 정보를 표시하여 디스크에 SMI 레이블이 있는지 확인하고 슬라이스 0 크기 정보를 검토합니다.

할당되지 않은 모든 디스크 공간이 슬라이스 0에 수집되도록 Free Hog 분할 영역을 설정합니다. 그런 다음 각 슬라이스 크기 필드에서 return 키를 눌러 하나의 큰 슬라이스 0을 만듭니다.

```
partition> modify
Select partitioning base:
  0. Current partition table (default)
  1. All Free Hog
Choose base (enter number) [0]? 1
Part      Tag      Flag      Cylinders      Size      Blocks
  0       root      wm         0              0      (0/0/0)      0
  1       swap      wu         0              0      (0/0/0)      0
  2    backup      wu    0 - 17829      136.58GB  (17830/0/0) 286438950
  3 unassigned      wm         0              0      (0/0/0)      0
  4 unassigned      wm         0              0      (0/0/0)      0
  5 unassigned      wm         0              0      (0/0/0)      0
  6       usr      wm         0              0      (0/0/0)      0
  7 unassigned      wm         0              0      (0/0/0)      0
  8       boot      wu    0 - 0          7.84MB   (1/0/0)      16065
  9 alternates      wm         0              0      (0/0/0)      0
```

```
Do you wish to continue creating a new partition
table based on above table[yes]?
Free Hog partition[6]? 0
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:
```

```
Part      Tag      Flag      Cylinders      Size      Blocks
  0       root      wm    1 - 17829      136.58GB  (17829/0/0) 286422885
  1       swap      wu         0              0      (0/0/0)      0
```



```

2   backup   wu      0 - 17829    136.58GB   (17830/0/0) 286438950
3   unassigned  wm      0           0           (0/0/0)      0
4   unassigned  wm      0           0           (0/0/0)      0
5   unassigned  wm      0           0           (0/0/0)      0
6   usr        wm      0           0           (0/0/0)      0
7   unassigned  wm      0           0           (0/0/0)      0
8   boot       wu      0 - 0       7.84MB     (1/0/0)      16065
9   alternates wm      0           0           (0/0/0)      0

```

```

Do you wish to continue creating a new partition
table based on above table[yes]? yes
Enter table name (remember quotes): "c8t0d0"

```

```
Ready to label disk, continue? yes
```

## 10 실패한 디스크가 교체되었음을 ZFS에 알립니다.

```
# zpool replace rpool c2t1d0s0
# zpool online rpool c2t1d0s0
```

교체 디스크를 넣은 후 해당 디스크를 온라인 상태로 전환할 필요가 없는 하드웨어도 있습니다.

새 디스크를 연결하여 미러링된 루트 풀을 만들거나 큰 디스크를 연결하여 작은 디스크를 교체하는 경우 다음과 유사한 구문을 사용합니다.

```
# zpool attach rpool c0t0d0s0 c1t0d0s0
```

## 11 루트 풀 디스크를 새 디스크로 교체하는 경우 부트 블록을 적용합니다.

예를 들면 다음과 같습니다.

```
# installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdisk/c8t0d0s0
```

## 12 새 디스크에서 부트할 수 있는지 확인합니다.

## 13 시스템이 새 디스크에서 부트되면 이전 디스크를 분리합니다.

이 단계는 새 디스크를 연결하여 실패한 디스크나 작은 디스크를 교체하는 경우에만 필요합니다.

```
# zpool detach rpool c0t0d0s0
```

## 14 시스템의 BIOS를 재구성하여 새 디스크에서 자동으로 부트하도록 시스템을 설정합니다.

### 자세한 정보 ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만든 후...

ZFS 루트 파일 시스템에 사용할 디스크 슬라이스를 만든 후 루트 풀 스냅샷을 복원하여 루트 풀을 복구해야 하는 경우 [Oracle Solaris 관리: ZFS 파일 시스템의 “ZFS 루트 풀의 디스크 교체 방법”](#)을 참조하십시오.

## ▼ x86:ZFS 루트 파일 시스템의 부트 블록 설치 방법

- 1 관리자로 전환합니다.
- 2 시스템 디스크의 부트 블록을 설치합니다.  

```
# installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdisk/cwtxdysz
```

 자세한 내용은 [installgrub\(1M\)](#)를 참조하십시오.
- 3 레벨 3을 실행하려면 시스템을 재부트하여 부트 블록이 설치되었는지 확인합니다.  

```
# init 6
```

### 예 13-1 x86:ZFS 루트 파일 시스템의 부트 블록 설치

루트 풀에 사용하려는 디스크를 물리적으로 교체한 후 Oracle Solaris OS를 재설치하거나 루트 풀에 사용할 새 디스크를 연결하면 부트 블록이 자동으로 설치됩니다. `zpool replace` 명령을 사용하여 루트 풀에 사용하려는 디스크를 교체하는 경우 시스템이 교체 디스크에서 부트될 수 있도록 부트 블록을 수동으로 설치해야 합니다.

다음 예에서는 ZFS 루트 파일 시스템의 부트 블록을 설치하는 방법을 보여 줍니다.

```
# installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdisk/c1d0s0
stage2 written to partition 0, 277 sectors starting at 50 (abs 16115)
stage1 written to partition 0 sector 0 (abs 16065)
```

## ▼ x86:ZFS 파일 시스템에 사용할 디스크 설정 방법

루트가 아닌 ZFS 파일 시스템에 사용할 디스크를 설정하는 경우 풀을 만들거나 디스크를 풀에 추가하면 자동으로 디스크 레이블이 재지정됩니다. 전체 디스크를 사용하여 풀을 만드는 경우 또는 전체 디스크를 ZFS 저장소 풀에 추가하면 EFI 레이블이 적용됩니다. EFI 디스크 레이블에 대한 자세한 내용은 [187 페이지 “EFI 디스크 레이블”](#)을 참조하십시오.

일반적으로 대부분의 현대적 버스 유형은 핫 플러그를 지원합니다. 즉, 빈 슬롯에 디스크를 삽입할 수 있고 시스템에서 이러한 디스크를 인식합니다. 핫 플러그 장치에 대한 자세한 내용은 [6 장, “동적으로 장치 구성\(작업\)”](#)을 참조하십시오.

- 1 관리자로 전환합니다.  
 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “관리 권한을 얻는 방법”](#)을 참조하십시오.
- 2 디스크를 시스템에 연결하고 디스크의 물리적 연결을 확인합니다.  
 자세한 내용은 디스크의 하드웨어 설치 설명서를 참조하십시오.

**3 필요한 경우 실패한 디스크를 오프라인으로 전환하고 구성을 해제합니다.**

일부 하드웨어의 경우 실패한 디스크를 교체하기 위해 `zpool replace` 작업을 시도하기 전에 디스크를 오프라인 상태로 만들고 구성을 해제해야 합니다. 예를 들면 다음과 같습니다.

```
# zpool offline tank c1t1d0
# cfgadm -c unconfigure c1::dsk/c1t1d0
<Physically remove failed disk c1t1d0>
<Physically insert replacement disk c1t1d0>
# cfgadm -c configure c1::dsk/c1t1d0
```

교체 디스크를 넣은 후 해당 디스크를 재구성할 필요가 없는 하드웨어도 있습니다.

**4 새 디스크가 인식되는지 확인합니다.**

`format` 유틸리티의 출력을 검토하여 디스크가 AVAILABLE DISK SELECTIONS 아래에 나열되는지 확인합니다. 그런 다음 `format` 유틸리티를 종료합니다.

```
# format
```

**5 필요한 경우 실패한 디스크가 교체되었음을 ZFS에 알립니다.**

```
# zpool replace tank c1t1d0
# zpool online tank c1t1d0
```

새 디스크가 리실버링되는지 확인합니다.

```
# zpool status tank
```

**6 필요한 경우 기존 ZFS 저장소 풀에 새 디스크를 연결합니다.**

예를 들면 다음과 같습니다.

```
# zpool attach tank mirror c1t0d0 c2t0d0
```

새 디스크가 리실버링되는지 확인합니다.

```
# zpool status tank
```

자세한 내용은 [Oracle Solaris 관리: ZFS 파일 시스템의 4 장](#), “Oracle Solaris ZFS 저장소 풀 관리”를 참조하십시오.

## Solaris fdisk 분할 영역 만들기 및 변경

Solaris fdisk 분할 영역을 만들거나 변경하는 방법에 대한 지침과 예는 다음 섹션을 참조하십시오.

### x86: fdisk 분할 영역을 만드는 방법에 대한 지침

fdisk 분할 영역을 하나 이상 설정하는 경우 다음 지침을 따르십시오.

- 디스크를 최대 4개의 fdisk 분할 영역으로 나눌 수 있습니다. 분할 영역 중 하나는 Solaris 분할 영역이어야 합니다.
- 디스크에서 Solaris 분할 영역을 활성 분할 영역으로 설정해야 합니다. 활성 분할 영역은 시스템 시작 시 운영 체제가 기본적으로 부트되는 분할 영역입니다.
- Solaris fdisk 분할 영역은 실린더 경계에서 시작해야 합니다.
- 마스터 부트 레코드를 비롯한 추가 부트 정보가 섹터 0에 기록되기 때문에 Solaris fdisk 분할 영역은 첫번째 디스크의 실린더 0이 아니라 실린더 1에서 시작해야 합니다.
- Solaris fdisk 분할 영역은 전체 디스크일 수 있습니다. 또는 DOS 분할 영역에 사용할 공간을 확보하기 위해 더 작은 크기로 설정할 수 있습니다. 공간이 충분한 경우 새 분할 영역을 만들기 위해 기존 분할 영역을 변경하지 않고 디스크에 새 fdisk 분할 영역을 만들 수도 있습니다.

---

**x86 전용** – Solaris 슬라이스를 분할 영역이라고도 합니다. 슬라이스를 분할 영역이라고 하는 인터페이스도 있습니다.

fdisk 분할 영역은 x86 기반 시스템에서만 지원됩니다. 혼동을 방지하기 위해 Oracle Solaris 설명서에서는 fdisk 분할 영역과 Solaris fdisk 분할 영역 내의 엔티티를 구분합니다. 이러한 엔티티를 슬라이스나 분할 영역이라고 부를 수 있습니다.

---

## ▼ x86: Solaris fdisk 분할 영역을 만드는 방법

시작하기 전에 fdisk 분할 영역에 대한 정보가 필요한 경우 235 페이지 “x86: fdisk 분할 영역을 만드는 방법에 대한 지침”을 참조하십시오.

### 1 관리자로 전환합니다.

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

### 2 format 유틸리티를 호출합니다.

```
# format
```

디스크의 번호 매기기 목록이 표시됩니다.

자세한 내용은 **format(1M)**을 참조하십시오.

### 3 Solaris fdisk 분할 영역을 만들 디스크의 번호를 입력합니다.

```
Specify disk (enter its number): disk-number
```

여기서 *disk-number*는 Solaris fdisk 분할 영역을 만들려는 디스크의 번호입니다.

### 4 fdisk 메뉴를 선택합니다.

```
format> fdisk
```

표시되는 fdisk 메뉴는 디스크에 기존 fdisk 분할 영역이 있는지 여부에 따라 달라집니다.

- 5 프롬프트에서 **y**를 지정하여 전체 디스크에 걸쳐 있는 Solaris fdisk 분할 영역을 만들고 활성화합니다. 그런 다음 13단계로 이동합니다.

No fdisk table exists. The default partition for the disk is:

a 100% "SOLARIS System" partition

Type "y" to accept the default partition, otherwise type "n" to edit the partition table.

y

- 6 Solaris fdisk 분할 영역이 전체 디스크에 걸쳐 있지 않도록 하려면 프롬프트에서 **n**을 지정합니다.

Type "y" to accept the default partition, otherwise type "n" to edit the partition table.

n

Total disk size is 17848 cylinders  
Cylinder size is 16065 (512 byte) blocks

Partition	Status	Type	Cylinders		Length	%
			Start	End		
=====	=====	=====	=====	===	=====	===

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Edit/View extended partitions
6. Exit (update disk configuration and exit)
7. Cancel (exit without updating disk configuration)

Enter Selection:

- 7 옵션 1, **Create a partition**을 선택하여 fdisk 분할 영역을 만듭니다.

Enter Selection: 1

- 8 **1(=Solaris2)**을 선택하여 Solaris fdisk 분할 영역을 만듭니다.

Select the partition type to create:

1=SOLARIS2	2=UNIX	3=PCIXOS	4=Other	5=DOS12
6=DOS16	7=DOSEXT	8=DOSBIG	9=DOS16LBA	A=x86 Boot
B=Diagnostic	C=FAT32	D=FAT32LBA	E=DOSEXTLBA	F=EFI (Protective)
G=EFI_SYS	0=Exit?			

- 9 Solaris fdisk 분할 영역에 예약할 디스크의 비율을 식별합니다. 이 비율을 계산할 때는 기존 fdisk 분할 영역의 크기를 고려합니다.

Specify the percentage of disk to use for this partition (or type "c" to specify the size in cylinders). *nn*

**10 프롬프트에서 y를 입력하여 Solaris fdisk 분할 영역을 활성화합니다.**

```
Should this become the active partition? If yes, it will be
activated each time the computer is reset or turned on.
Please type "y" or "n". y
```

fdisk 분할 영역이 활성화된 후 Enter Selection 프롬프트가 표시됩니다.

**11 옵션 1, Create a partition을 선택하여 다른 fdisk 분할 영역을 만듭니다.**

fdisk 분할 영역을 만드는 방법에 대한 지침은 8-10 단계를 참조하십시오.

**12 디스크 구성을 업데이트하고 selection 메뉴에서 fdisk 메뉴를 종료합니다.**

```
Selection: 6
```

**13 label 명령을 사용하여 디스크 레이블을 재지정합니다.**

```
format> label
Ready to label disk, continue? yes
format>
```

**14 format 유틸리티를 종료합니다.**

```
format> quit
```

**예 13-2 x86: 전체 드라이브에 걸쳐 있는 Solaris fdisk 분할 영역 만들기**

다음 예에서는 format 유틸리티의 fdisk 옵션을 사용하여 전체 드라이브에 걸쳐 있는 Solaris fdisk 분할 영역을 만듭니다.

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
  0. c0d0 <DEFAULT cyl 2466 alt 2 hd 16 sec 63>
    /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0
  1. c0d1 <DEFAULT cyl 522 alt 2 hd 32 sec 63>
    /pci@0,0/pci-ide@7,1/ide@0/cmdk@1,0
  2. c1d0 <DEFAULT cyl 13102 alt 2 hd 16 sec 63>
    /pci@0,0/pci-ide@7,1/ide@1/cmdk@0,0
Specify disk (enter its number): 0
selecting c0d0
Controller working list found
[disk formatted]
format> fdisk
No fdisk table exists. The default partitioning for your disk is:

  a 100% "SOLARIS System" partition.

Type "y" to accept the default partition, otherwise type "n" to edit the
partition table. y

format> label
Ready to label disk, continue? yes
format> quit
```

## 자세한 정보 Solaris fdisk 분할 영역을 만든 후...

디스크에 Solaris fdisk 분할 영역을 만든 후 루트 폴 디스크나 루트가 아닌 폴 디스크에 사용할 수 있습니다. 자세한 내용은 [228 페이지 “x86: ZFS 파일 시스템에 사용할 디스크 설정”](#)을 참조하십시오.

## fdisk 분할 영역 식별자 변경

x86 시스템의 Solaris fdisk 분할 영역 식별자가 130(0x82)에서 191(0xbf)로 변경되었습니다. 모든 Oracle Solaris 명령, 유틸리티 및 드라이버가 두 fdisk 식별자에서 모두 작동하도록 업데이트되었습니다. fdisk 기능에 대한 변경 사항은 없습니다.

### ▼ Solaris fdisk 식별자 변경 방법

새 fdisk 메뉴 옵션을 사용하면 새 식별자와 이전 식별자를 전환할 수 있습니다. 분할 영역에 포함된 파일 시스템이 마운트된 경우에도 fdisk 식별자를 변경할 수 있습니다.

fdisk 메뉴의 두 type 값은 다음과 같이 이전 식별자와 새 식별자를 나타냅니다.

- Solaris는 0x82를 나타냅니다.
- Solaris2는 0xbf를 나타냅니다.

- 1 관리자로 전환합니다.
- 2 현재 fdisk 식별자를 표시합니다.  
예를 들면 다음과 같습니다.

```
Total disk size is 39890 cylinders
Cylinder size is 4032 (512 byte) blocks
```

Partition	Status	Type	Cylinders		Length	%
			Start	End		
1	Active	x86 Boot	1	6	6	0
2		Solaris2	7	39889	39883	100

- 3 fdisk 메뉴에서 옵션 4를 선택하여 fdisk 분할 영역 식별자를 다시 0x82로 변경합니다.

```
SELECT ONE OF THE FOLLOWING:
```

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

```
Enter Selection: 4
```

- 4 옵션 5를 선택하여 디스크 구성을 업데이트하고 종료합니다.

- 5 필요한 경우 fdisk 메뉴에서 옵션 4를 선택하여 fdisk 분할 영역 식별자를 다시 0xbf로 변경합니다.

예를 들면 다음과 같습니다.

Total disk size is 39890 cylinders

Cylinder size is 4032 (512 byte) blocks

Partition	Status	Type	Cylinders		Length	%
			Start	End		
1	Active	x86 Boot	1	6	6	0
2		Solaris	7	39889	39883	100

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection: 4

- 6 옵션 5를 선택하여 디스크 구성을 업데이트하고 종료합니다.



## COMSTAR를 사용하여 저장 장치 구성

---

이 장에서는 시작 프로그램 호스트가 저장소 네트워크를 통해 액세스할 수 있는 SCSI 대상 장치로 Oracle Solaris 11 호스트를 변환할 수 있는 소프트웨어 프레임워크인 COMSTAR(Common Multiprotocol SCSI TARget)를 구성하는 방법에 대해 설명합니다.

즉, 시스템의 저장 장치를 Linux, Mac OS 또는 Windows 클라이언트 시스템에서 로컬 저장 장치인 것처럼 사용할 수 있습니다. 지원되는 저장소 프로토콜은 iSCSI, FC, iSER 및 SRP입니다.

Oracle Solaris의 iSNS 지원에 대한 자세한 내용은 15 장, “Oracle Solaris iSNS(Internet Storage Name Service) 구성 및 관리”를 참조하십시오.

Oracle Solaris의 iSCSI 구성 문제 해결은 283 페이지 “iSCSI 구성 문제 해결”을 참조하십시오.

### COMSTAR 및 iSCSI 기술(개요)

iSCSI는 Internet SCSI(Small Computer System Interface)의 머리글자어로, 데이터 저장소 부속 시스템을 연결하기 위한 IP(인터넷 프로토콜) 기반 저장소 네트워킹 표준입니다.

IP 네트워크를 통해 SCSI 명령을 전달하는 iSCSI 프로토콜을 사용하면 네트워크상의 블록 장치를 로컬 시스템에 연결된 것처럼 액세스할 수 있습니다. COMSTAR는 이러한 iSCSI 대상 장치를 관리하는 보다 편리한 방법을 제공합니다.

COMSTAR는 STMF(SCSI Target Mode Framework)를 이용하여 다음 구성 요소가 포함된 대상 저장 장치를 관리합니다.

- 포트 공급자(또는 플러그인) - FC(Fibre Channel) 및 iSCSI와 같은 프로토콜을 구현합니다.
- 논리 장치 공급자 - 디스크 및 테이프 장치와 같은 다양한 SCSI 장치를 에뮬레이트합니다.

- libstmf 관리 라이브러리 - COMSTAR 관리 인터페이스를 제공합니다. iSCSI 기능을 구현하는 모듈은 기본 전송과 직접 상호 작용하지 않습니다. 유사한 방식으로, 전송 프로토콜을 구현하는 모듈은 전송 중인 패킷에 고유한 SCSI 레벨 기능을 인식할 수 없습니다. 이러한 전송 예로 Fibre Channel과 iSCSI가 있습니다. 프레임워크는 SCSI 명령 및 연관된 리소스의 실행과 정리를 구분합니다. 이 구분은 SCSI 또는 전송 모듈을 작성하는 작업을 단순화합니다.

이러한 기능을 관리하려면 다음 명령을 사용합니다.

- itadm 명령은 SCSI 대상 모드 프레임워크 내에서 iSCSI(Internet SCSI) 노드를 관리합니다.
- stmfadm 명령은 SCSI 대상 모드 프레임워크 내에서 논리 장치를 구성합니다.
- srptadm 명령은 SCSI 대상 모드 프레임워크 내에서 SRP(SCSI RDMA Protocol) 대상 포트를 관리합니다.

다음 솔루션을 통해 기존 TCP/IP 네트워크에서 저장 장치를 사용할 수 있습니다.

- iSCSI 블록 장치 또는 테이프 - 블록 레벨의 SCSI 명령과 데이터를 IP 패킷으로 변환합니다. 네트워크에서 iSCSI를 사용할 경우 한 시스템과 대상 장치(예: 테이프 장치 또는 데이터베이스) 간에 블록 레벨 액세스가 필요한 경우에 도움이 됩니다. 여러 사용자나 시스템이 iSCSI 대상 장치와 같은 블록 레벨 장치에 액세스할 수 있도록 블록 레벨 장치에 대한 액세스는 잠기지 않습니다.
- NFS - IP를 통해 파일 데이터를 전송합니다. 네트워크에서 NFS를 사용할 경우 여러 시스템에서 파일 데이터를 공유할 수 있다는 이점이 있습니다. 많은 사용자가 NFS 환경에서 사용 가능한 데이터에 액세스하는 경우 파일 데이터에 대한 액세스가 적절하게 잠깁니다.

Oracle Solaris의 iSCSI 대상과 시작 프로그램을 사용할 경우의 이점은 다음과 같습니다.

- iSCSI 프로토콜이 기존 이더넷 네트워크에서 실행됩니다.
  - 지원되는 NIC(네트워크 인터페이스 카드), 이더넷 허브 또는 이더넷 스위치를 모두 사용할 수 있습니다.
  - IP 포트 1개가 여러 iSCSI 대상 장치를 처리할 수 있습니다.
  - IP 네트워크의 기존 기반구조 및 관리 도구를 사용할 수 있습니다.
- Fibre-Channel HBA 비용 없이 클라이언트에 연결될 수 있는 기존 Fibre-Channel 장치를 사용할 수 있습니다. 뿐만 아니라 전용 어레이가 있는 시스템에서 이제 Oracle Solaris ZFS 또는 UFS 파일 시스템을 포함하는 복제 저장소를 내보낼 수 있습니다.
- 프로토콜을 사용하여 적절한 하드웨어가 있는 Fibre Channel 또는 iSCSI SAN(Storage Area Network) 환경에 연결할 수 있습니다.

Oracle Solaris의 iSCSI 시작 프로그램 소프트웨어를 사용할 경우의 현재 제한 사항은 다음과 같습니다.

- SLP를 사용하는 iSCSI 장치에 대한 지원은 현재 제공되지 않습니다.
- iSCSI 대상을 덤프 장치로 구성할 수 없습니다.

- 기존 네트워크를 통해 대량의 데이터를 전송할 경우 성능에 영향을 줄 수 있습니다.

## COMSTAR 소프트웨어 및 하드웨어 요구 사항

- Oracle Solaris 저장소 소프트웨어 및 장치
- 저장 장치를 제공하는 시스템용 group/feature/storage-server 소프트웨어 패키지
- 지원되는 NIC

## COMSTAR 구성(작업 맵)

작업	설명	수행 방법
COMSTAR 소프트웨어 및 하드웨어 요구 사항을 식별합니다.	COMSTAR를 사용하여 iSCSI 저장소 네트워크를 설정하기 위한 소프트웨어 및 하드웨어 요구 사항을 식별합니다.	243 페이지 “COMSTAR 소프트웨어 및 하드웨어 요구 사항”
iSCSI 대상 검색 방법을 결정합니다.	환경에 가장 적합한 iSCSI 대상 검색 방법을 결정합니다.	246 페이지 “동적 또는 정적 대상 검색 구성”
STMF 서비스를 사용하여 설정합니다.	영구적 대상 정보를 제공하는 STMF 서비스를 사용하여 설정합니다.	
SCSI 논리 장치를 만들고 사용 가능하게 합니다.	SCSI 논리 장치(LUN)를 만들고 모든 호스트나 특정 호스트가 iSCSI 또는 iSER 구성에 사용할 수 있게 합니다.	248 페이지 “iSCSI LUN을 만드는 방법”
iSCSI 대상을 구성합니다.	iSCSI 저장소 구성 요소의 iSCSI 대상을 구성합니다.	249 페이지 “iSCSI 대상을 만드는 방법”
iSCSI 시작 프로그램을 구성합니다.	iSCSI 대상에 SCSI 요청을 시작하는 시스템을 구성합니다.	251 페이지 “iSCSI 시작 프로그램 구성 방법”
iSCSI 디스크에 액세스합니다.	format 유틸리티를 사용하여 iSCSI 디스크에 액세스할 수 있습니다. 시스템을 재부트한 후 iSCSI 디스크를 자동으로 사용 가능하게 할 수도 있습니다.	255 페이지 “iSCSI 디스크 액세스 방법”
선택한 시스템만 LUN에 액세스하도록 제한합니다.	네트워크의 특정 시스템만 LUN에 액세스하도록 제한할 수 있습니다.	257 페이지 “선택한 시스템만 LUN에 액세스하도록 제한하는 방법”

작업	설명	수행 방법
Fibre Channel 장치를 구성합니다.	환경에 FC 저장소 어레이가 있는 경우 COMSTAR를 사용하여 FC 장치를 구성합니다.	258 페이지 “COMSTAR를 사용하여 Fibre Channel 장치 구성”
FCoE 장치를 구성합니다.	COMSTAR를 사용하여 FCoE(Fibre Channel over Ethernet) 장치를 구성합니다.  FCoE 기능은 이더넷 인터페이스를 통해 제공됩니다. FCoE 포트는 이더넷 인터페이스와 연관된 논리적 엔티티입니다.	263 페이지 “COMSTAR를 사용하여 FCoE 장치 구성”
SRP 장치를 구성합니다.	COMSTAR를 사용하여 SRP 장치를 구성합니다.  SRP(SCSI RDMA Protocol)는 SCSI 데이터 전송 단계를 IB(Infiniband) RDMA(Remote Direct Memory Access) 작업에 매핑하여 SCSI 프로토콜을 가속화합니다.	266 페이지 “COMSTAR를 사용하여 SRP 장치 구성”
(옵션) Oracle Solaris iSCSI 구성에서 인증을 설정합니다.	Oracle Solaris iSCSI 구성에서 인증을 사용할 것인지 여부를 결정합니다.  단방향 CHAP 또는 양방향 CHAP 사용을 고려합니다.  타사 RADIUS 서버를 사용하여 CHAP 관리를 단순화하는 방법을 고려합니다.	268 페이지 “iSCSI 시작 프로그램에 대해 CHAP 인증을 구성하는 방법”  270 페이지 “iSCSI 대상에 대해 CHAP 인증을 구성하는 방법”  271 페이지 “iSCSI 대상에 대해 RADIUS 서버를 구성하는 방법”
iSCSI 구성을 모니터링합니다.	iscsiadm 명령을 사용하여 iSCSI 구성을 모니터링합니다.	276 페이지 “iSCSI 구성 정보 표시 방법”
(옵션) iSCSI 구성을 수정합니다.	헤더 및 데이터 다이제스트 매개변수와 같은 iSCSI 대상 매개변수를 수정할 수도 있습니다.	280 페이지 “iSCSI 시작 프로그램 및 대상 매개변수 수정 방법”

## COMSTAR 구성

COMSTAR를 사용하여 iSCSI 대상과 시작 프로그램을 구성하는 과정에는 다음 작업이 포함됩니다.

- 하드웨어 및 소프트웨어 요구 사항 식별
- IP 네트워크 구성
- iSCSI 대상 장치 연결 및 설정
- 시작 프로그램 구성
- iSCSI 대상 검색 방법 구성
- iSCSI 디스크에 파일 시스템 만들기
- (옵션) iSCSI 시작 프로그램과 iSCSI 대상 간에 iSCSI 인증 구성
- iSCSI 구성 모니터

iSCSI 구성 정보는 `/etc/iscsi` 디렉토리에 저장되지만 수동 관리는 필요 없습니다.

## COMSTAR 용어

iSCSI 대상과 시작 프로그램을 구성하기 전에 다음 용어를 검토합니다.

용어	설명
검색	사용 가능한 대상 목록을 시작 프로그램에 제공하는 프로세스입니다.
검색 방법	iSCSI 대상을 찾을 수 있는 방법입니다. 현재 사용할 수 있는 세 가지 방법은 다음과 같습니다. <ul style="list-style-type: none"> <li>■ iSNS(Internet Storage Name Service) - 하나 이상의 iSNS 서버와 상호 작용하여 잠재적 대상을 검색합니다.</li> <li>■ SendTargets - <i>discovery-address</i>를 사용하여 잠재적 대상을 검색합니다.</li> <li>■ 정적 - 정적 대상 주소 지정을 구성합니다.</li> </ul>
시작 프로그램	iSCSI 대상에 SCSI 요청을 시작하는 드라이버입니다.
시작 프로그램 그룹	시작 프로그램 집합입니다. 시작 프로그램 그룹을 LUN에 연결하면 해당 그룹의 시작 프로그램만 LUN에 액세스할 수 있습니다.
iqn 또는 eui 주소 형식	iqn(iSCSI Qualified Name) 주소는 iSCSI 네트워크에서 <i>iqn.date.authority:uniqueid</i> 형식을 사용하는 장치의 고유 식별자입니다. iSCSI 시작 프로그램 또는 대상을 초기화하면 iSCSI 시작 프로그램 또는 대상에 자동으로 IQN 이름이 할당됩니다.  eui(Extended Unique Identifier) 주소는 16진수 16자리로 구성되며 SCSI 및 InfiniBand 표준에서 모두 사용되는 GUID 클래스를 식별합니다. SRP 장치는 eui 주소 형식을 사용합니다.

용어	설명
논리 장치	저장 시스템에서 고유 번호가 지정된 구성 요소입니다. LUN이라고도 합니다. LUN이 하나 이상의 SCSI 대상에 연결된 경우 하나 이상의 SCSI 시작 프로그램이 대상에 액세스할 수 있습니다.
대상 장치	iSCSI 저장소 구성 요소입니다.
대상 그룹	대상 집합입니다. 한 대상 그룹의 모든 대상이 LUN을 사용하게 할 수 있습니다.
대상 포털 그룹	특정 iSCSI 대상이 수신 대기할 인터페이스를 결정하는 IP 주소 목록입니다. TPG에는 IP 주소와 TCP 포트 번호가 포함됩니다.

## 동적 또는 정적 대상 검색 구성

동적 장치 검색 방법 중 하나를 구성할지 또는 정적 iSCSI 시작 프로그램 대상을 사용하여 장치 검색을 수행할지 결정합니다.

- **동적 장치 검색** - 다음 두 가지 동적 장치 검색 방법을 사용할 수 있습니다.
  - SendTargets - iSCSI 노드가 iSCSI - Fibre-Channel 브릿지와 같은 많은 대상을 노출하는 경우 iSCSI 노드 IP 주소/포트 조합을 제공하고 iSCSI 시작 프로그램이 SendTargets 기능을 사용하여 장치 검색을 수행하게 할 수 있습니다.
  - iSNS - iSNS(Internet Storage Name Service)를 사용하면 iSCSI 시작 프로그램이 가능한 한 적은 구성 정보를 사용하여 액세스 가능한 대상을 검색할 수 있습니다. 또한 iSNS는 저장소 노드의 작동 상태가 변경될 경우 iSCSI 시작 프로그램에 알리는 상태 변경 알림을 제공합니다. iSNS 검색 방법을 사용하기 위해 iSNS 서버 주소/포트 조합을 제공하고 iSCSI 시작 프로그램이 장치 검색을 수행하도록 지정한 iSNS 서버를 질의하게 할 수 있습니다. iSNS 서버의 기본 포트는 3205입니다. iSNS에 대한 자세한 내용은 RFC 4171: <http://www.ietf.org/rfc/rfc4171.txt>를 참조하십시오. iSNS 검색 서비스는 네트워크에서 모든 대상을 검색하는 관리 모델을 제공합니다. Oracle Solaris의 iSNS 지원 설정에 대한 자세한 내용은 15 장, “Oracle Solaris iSNS(Internet Storage Name Service) 구성 및 관리”를 참조하십시오.
- **정적 장치 검색** - iSCSI 노드에 대상이 거의 없거나 시작 프로그램이 액세스를 시도하는 대상을 제한하려는 경우 다음과 같은 정적 대상 주소 명명 규칙을 사용하여 *target-name*을 정적으로 구성할 수 있습니다.
 

```
target,target-address[:port-number]
```

어레이의 관리 도구에서 정적 대상 주소를 결정할 수 있습니다.

주- 정적 및 동적 장치 검색 방법에서 모두 검색되도록 iSCSI 대상을 구성하지 마십시오. 중복 검색 방법을 사용할 경우 시작 프로그램이 iSCSI 대상 장치와 통신 중이면 성능이 느려질 수 있습니다.

## COMSTAR를 사용하여 iSCSI 장치 구성

COMSTAR iSCSI(Internet SCSI) 대상을 설정 및 구성하고 네트워크에서 사용 가능하게 할 수 있습니다. iSCSI 기능은 표준 iSCSI 프로토콜을 사용하여 이더넷과 같은 일반 인터넷 연결을 통해 작동할 수 있습니다. 또한 iSCSI 프로토콜은 명명 및 검색 서비스, CHAP와 RADIUS를 사용한 인증 서비스, iSNS를 통한 중앙 집중식 관리 기능을 제공합니다.

시스템에 IB(InfiniBand) HCA(Host Channel Adapter)가 하나 이상 있고 시작 프로그램과 대상 간에 연결이 있는 경우 iSCSI 연결은 데이터 처리량 향상을 위해 iSER(iSCSI Extensions for RDMA)을 사용합니다. IB HCA와 함께 사용할 경우 iSER은 고대역폭, 낮은 CPU 사용률 및 여러 프로토콜이 공유할 수 있는 단일 네트워크 연결을 제공합니다.

iSER 대상과 시작 프로그램은 모두 iSCSI Data Mover라는 구성 요소를 사용하여 iSCSI 연결 서비스를 제공합니다. 시작 프로그램과 대상이 모두 IB 가능 장치에 해당하는 IP 주소를 사용하도록 구성될 때마다 자동으로 iSER이 사용됩니다.

### ▼ STMF 서비스를 사용으로 설정하는 방법

COMSTAR는 SMF를 사용하여 논리 장치 매핑, 호스트 그룹 정의, 대상 그룹 정의 등의 현재 영구적 구성을 저장합니다. 부트 도중 서비스가 사용으로 설정되었거나 `svcadm` 명령을 사용하는 경우 커널 프레임워크 내의 사용되지 않는 구성 데이터를 지운 다음 SMF 저장소의 구성을 드라이버에 재로드합니다. 구성이 로드된 후 구성 변경 사항이 SMF 저장소는 물론 드라이버 데이터베이스 내에서 자동으로 업데이트됩니다. 예를 들어, `stmfadm` 명령을 통한 변경 사항은 두 영역에서 모두 자동으로 업데이트됩니다.

COMSTAR 대상 모드 프레임워크는 `stmf` 서비스로 실행됩니다. 기본적으로 이 서비스는 사용 안함으로 설정됩니다. COMSTAR 기능을 사용하려면 이 서비스를 사용으로 설정해야 합니다. `svcs` 명령을 사용하여 서비스를 식별할 수 있습니다.

`group/feature/storage-server` 패키지를 설치한 이후 서버를 재부트하지 않은 경우 서비스가 올바르게 사용으로 설정되지 않았을 수 있습니다.

#### 1 COMSTAR 저장소 서버 소프트웨어를 설치합니다.

```
target# pkg install group/feature/storage-server
          Packages to install:    75
          Create boot environment: No
          Services to restart:   7

DOWNLOAD          PKGS          FILES          XFER (MB)
Completed         75/75         9555/9555      105.7/105.7
```

PHASE	ACTIONS
Install Phase	13347/13347
PHASE	ITEMS
Package State Update Phase	75/75
Image State Update Phase	2/2
Loading smf(5) service descriptions: 17/17	
Loading smf(5) service descriptions: 3/3	
PHASE	ITEMS
Reading Existing Index	8/8
Indexing Packages	75/75
Indexing Packages	75/75
Optimizing Index...	
PHASE	ITEMS
Indexing Packages	573/573

## 2 시스템을 재부트하거나 stmf 서비스를 사용으로 설정합니다.

```
target# svcadm enable stmf
# svcs stmf
STATE          STIME          FMRI
online         09:42:32      svc:/system/stmf:default
```

## ▼ COMSTAR 구성 백업 및 복원 방법

COMSTAR 구성을 완료한 후 필요한 경우 복원할 수 있는 복사본을 만듭니다.

- 1 관리자로 전환합니다.
- 2 현재 COMSTAR 구성을 내보냅니다.
 

```
# svccfg export -a stmf > COMSTAR.backup
```
- 3 필요한 경우 내보낸 구성을 복원합니다.
 

```
# svccfg import COMSTAR.backup
```

## ▼ iSCSI LUN을 만드는 방법

디스크 유형 LUN을 만들기 위한 논리 장치 공급자를 `sbd`라고 합니다. 하지만 디스크 유형 LUN을 공유하려면 논리 장치의 저장소를 초기화해야 합니다.

서버에서 제공하는 디스크 볼륨을 **대상**이라고 합니다. LUN이 iSCSI 대상에 연결된 경우 iSCSI 시작 프로그램이 해당 LUN에 액세스할 수 있습니다.

SCSI LUN을 만드는 프로세스는 다음과 같습니다.

- 보조 저장소라고도 하는 LUN의 저장소를 초기화합니다.
- 보조 저장소를 사용하여 SCSI LUN을 만듭니다.



LUN을 만들면 GUID(Global Unique Identifier)가 지정됩니다(예 600144F0B5418B000004DDAC7C10001). 이 GUID는 LUN을 매핑하여 호스트 선택과 같은 이후 작업에서 LUN을 참조하는 데 사용됩니다.

다음 단계는 저장 장치를 제공하는 시스템에서 수행됩니다.

**1 ZFS 저장소 풀을 만듭니다.**

```
target# zpool create sanpool mirror c2t3d0 c2t4d0
```

**2 SCSI LUN으로 사용할 ZFS 볼륨을 만듭니다.**

```
target# zfs create -V 2g sanpool/vol1
```

**3 ZFS 볼륨의 LUN을 만듭니다.**

```
target# stmfadm create-lu /dev/zvol/rdisk/sanpool/vol1
Logical unit created: 600144F0B5418B000004DDAC7C10001
```

/dev/zvol/rdisk/ *pool-name*/ 디렉토리에서 ZFS 볼륨의 장치 경로를 찾을 수 있습니다.

**4 LUN이 만들어졌는지 확인합니다.**

```
target# stmfadm list-lu
LU Name: 600144F0B5418B000004DDAC7C10001
```

**5 LUN 뷰를 추가합니다.**

이 명령을 사용하면 모든 시스템이 LUN에 액세스할 수 있습니다.

```
target# stmfadm add-view 600144F0B5418B000004DDAC7C10001
```

LUN 뷰를 특정 시스템으로 제한하려는 경우 257 페이지 “[선택한 시스템만 LUN에 액세스하도록 제한하는 방법](#)”을 참조하십시오.

**6 LUN 구성을 확인합니다.**

```
target# stmfadm list-view -l 600144F0B5418B000004DDAC7C10001
View Entry: 0
  Host group   : All
  Target group : All
  LUN         : 0
```

## ▼ iSCSI 대상을 만드는 방법

이 절차에서는 iSCSI 대상을 포함하는 로컬 시스템에 로그인했다고 가정합니다.

**1 iSCSI 대상 서비스를 사용으로 설정합니다.**

```
target# svcadm enable -r svc:/network/iscsi/target:default
```

서비스가 사용으로 설정되었는지 확인합니다.

```
target# svcs -l iscsi/target
fmri          svc:/network/iscsi/target:default
```

```

name          iscsi target
enabled       true
state         online
next_state    none
state_time    Mon May 23 14:48:59 2011
logfile       /var/svc/log/network-iscsi-target:default.log
restarter     svc:/system/svc/restarter:default
dependency    require_any/error svc:/milestone/network (online)
dependency    require_all/none svc:/system/stmf:default (online)

```

## 2 iSCSI 대상을 만듭니다.

```

target# itadm create-target
Target iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405 successfully created

```

## 3 iSCSI 대상 정보를 표시합니다.

```

target# itadm list-target -v
TARGET NAME                               STATE   SESSIONS
iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405  online  0
  alias:                                   -
  auth:                                    none (defaults)
  targetchapuser:                          -
  targetchapsecret:                        unset
  tpg-tags:                                default

```

## ▼ iSER에 대해 IBHCA를 구성하는 방법

iSER(iSCSI Extensions for RDMA) 기능을 완전히 활용하려면 IB(InfiniBand) HCA(Host Channel Adapter)가 필요합니다. iSER을 사용하려면 대상과 시작 프로그램에서 모두 HCA를 구성해야 합니다.

### 1 호스트(시작 프로그램) 시스템에서 관리자 로 로그인합니다.

### 2 HCA를 IB 스위치에 연결합니다.

자세한 내용은 공급업체 설명서를 참조하십시오.

### 3 대상과 시작 프로그램에서 HCA를 구성합니다.

대상과 시작 프로그램이 동일한 서브넷에 있어야 합니다. 이 예에서는 `ibd0`을 드라이버로 사용합니다.

```
# ipadm create-addr ibd0
```

### 4 HCA에 대한 IP 주소 및 포트 조합을 구성합니다.

```
# ipadm create-addr -T static -a local=10.1.190.141/24 ibd0/v4addr
```

### 5 인터페이스 구성을 확인합니다.

```

# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4       static    ok         127.0.0.1/8

```

```
e1000g0/_b      dhcp      ok          10.1.190.141/24
lo0/v6          static    ok          ::1/128
e1000g0/_a      addrconf ok          fe80::214:4fff:fe27:360c/10
```

- 대상 시스템에서 관리자로 전환하고 네트워크의 다른 모든 HCA 호스트에 대해 3-5 단계를 반복합니다.
- 대상과 시작 프로그램에서 모두 연결을 확인합니다.

```
target# ping initiator-ip
initiator# ping target-ip
```

## ▼ iSCSI 시작 프로그램 구성 방법

시작 프로그램 구성 프로세스에는 시작 프로그램에 사용 가능한 대상 목록을 제공하는 iSCSI 대상 검색 방법을 식별하는 작업이 포함됩니다. iSCSI 대상에서 정적, SendTargets 또는 iSNS 동적 검색을 구성할 수 있습니다. SendTargets 옵션을 사용한 동적 검색은 iSCSI와 Fibre Channel 간 브릿지와 같이 많은 대상에 액세스하는 iSCSI 시작 프로그램에 최적화된 구성입니다. SendTargets 동적 검색에서 iSCSI 시작 프로그램이 대상 검색을 수행하려면 iSCSI 대상의 IP 주소 및 포트 조합이 필요합니다. 가장 일반적인 검색 방법은 SendTargets입니다.

대상 검색 방법을 구성하는 경우 선택한 방법에 따라 다음 정보를 제공해야 합니다.

- SendTargets - 대상 IP 주소
- iSNS - iSNS 서버 주소
- 정적 - 대상 IP 주소 및 대상 이름

대상 검색 방법 구성에 대한 자세한 내용은 [246 페이지 “동적 또는 정적 대상 검색 구성”](#)을 참조하십시오.

- iSCSI 시작 프로그램 서비스를 사용으로 설정합니다.

```
initiator# svcadm enable network/iscsi/initiator
```

- 대상을 제공하는 서버에 로그인한 상태에서 대상의 이름과 IP 주소를 확인합니다.

```
target# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4       static    ok         127.0.0.1/8
e1000g0/_b   dhcp      ok         10.80.227.189/24
lo0/v6       static    ok         ::1/128
e1000g0/_a   addrconf ok         fe80::214:4fff:fe27:360c/10
target# itadm list-target -v
TARGET NAME                                     STATE   SESSIONS
iqn.1986-03.com.sun.02:73d12edc-9bb9-cb44-efc4-c3b36c039405  online  0
  alias:                                         -
  auth:                                         none (defaults)
  targetchapuser:                             -
  targetchapsecret:                           unset
  tpg-tags:                                    default
```

### 3 대상이 정적으로 검색되도록 구성합니다.

```
initiator# iscsiadm add static-config iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405,
10.80.227.189
```

### 4 정적 구성 정보를 검토합니다.

```
initiator# iscsiadm list static-config
Static Configuration Target: iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405,
10.80.227.189:3260
```

검색 방법을 사용으로 설정할 때까지 iSCSI 연결이 시작되지 않습니다. 다음 단계를 참조하십시오.

### 5 다음 대상 검색 방법 중 하나를 구성합니다.

- 동적으로 검색되는 (SendTargets) 대상을 구성한 경우 SendTargets 검색 방법을 구성합니다.

```
initiator# iscsiadm add discovery-address 10.80.227.189
```

- 동적으로 검색되는 (iSNS) 대상을 구성한 경우 iSNS 검색 방법을 구성합니다.

```
initiator# iscsiadm add isns-server 10.80.227.189
```

### 6 다음 대상 검색 방법 중 하나를 사용으로 설정합니다.

- 동적으로 검색되는 (SendTargets) 대상을 구성한 경우 SendTargets 검색 방법을 사용으로 설정합니다.

```
initiator# iscsiadm modify discovery --sendtargets enable
```

- 동적으로 검색되는 (iSNS) 대상을 구성한 경우 iSNS 검색 방법을 사용으로 설정합니다.

```
initiator# iscsiadm modify discovery --iSNS enable
```

- 정적 대상을 구성한 경우 정적 대상 검색 방법을 사용으로 설정합니다.

```
initiator# iscsiadm modify discovery --static enable
```

### 7 필요한 경우 iSCSI 디스크를 인식하도록 /dev 네임스페이스를 재구성합니다.

```
initiator# devfsadm -i iscsi
```

## ▼ 검색된 iSCSI 대상 제거 방법

다음 작업 중 하나를 수행하면 연관된 대상이 로그아웃됩니다.

- 검색 주소 제거
- iSNS 서버 제거
- 정적 구성 제거
- 검색 방법 사용 안함

마운트된 파일 시스템이 있는 경우와 같이 이러한 연관된 대상이 여전히 사용 중이면 해당 장치의 로그아웃이 실패하며 활성 대상 목록에 유지됩니다.

이 선택적 절차에서는 iSCSI 대상 장치에 대한 액세스가 이미 구성되어 있는 로컬 시스템에 로그인했다고 가정합니다.

### 1 관리자로 전환합니다.

### 2 (옵션) 다음 중 하나를 사용하여 iSCSI 대상 검색 방법을 사용 안함으로 설정합니다.

- SendTargets 검색 방법을 사용 안함으로 설정해야 하는 경우 다음 명령을 사용합니다.

```
initiator# iscsiadm modify discovery --sendtargets disable
```

- iSNS 검색 방법을 사용 안함으로 설정해야 하는 경우 다음 명령을 사용합니다.

```
initiator# iscsiadm modify discovery --iSNS disable
```

- 정적 대상 검색 방법을 사용 안함으로 설정해야 하는 경우 다음 명령을 사용합니다.

```
initiator# iscsiadm modify discovery --static disable
```

### 3 다음 중 하나를 사용하여 iSCSI 장치 검색 항목을 제거합니다.

- iSCSI SendTargets 검색 항목을 제거합니다.

예를 들면 다음과 같습니다.

```
initiator# iscsiadm remove discovery-address 10.0.0.1:3260
```

- iSCSI iSNS 검색 항목을 제거합니다.

예를 들면 다음과 같습니다.

```
# iscsiadm remove isns-server 10.0.0.1:3205
```

- 정적 iSCSI 검색 항목을 제거합니다.

예를 들면 다음과 같습니다.

```
initiator# iscsiadm remove static-config eui.5000ABCD78945E2B,10.0.0.1
```

---

주 - 연관된 논리 장치(LUN)가 사용 중인 검색 항목을 사용 안함으로 설정하거나 제거하려고 하면 사용 안함 또는 제거 작업이 실패하고 다음 메시지가 표시됩니다.

```
logical unit in use
```

이 오류가 발생하면 LUN에서 연관된 I/O를 모두 중지하고 파일 시스템을 마운트 해제합니다. 그런 다음 사용 안함 또는 제거 작업을 반복합니다.

---

### 4 iSCSI 대상 장치를 제거합니다.

예를 들면 다음과 같습니다.

```
target# itadm delete-target target-IQN
```

대상이 여전히 온라인 상태이거나 사용 중이면 이 명령에서 오류가 발생할 수 있습니다.

```
The target is online or busy. Use the -f (force) option, or 'stmfadm offline-target
iqn.1986-03.com.sun:02:99619b8a-a4dc-4cfb-93f0-ee3debe7b0c8'
itadm delete-target failed with error 16
```

## iSCSI 대상 포털 그룹 만들기

TPG(대상 포털 그룹)를 만들어 여러 iSCSI 및 iSER 대상의 검색을 관리할 수 있습니다. TPG는 특정 iSCSI 대상이 수신 대기할 인터페이스를 결정하는 IP 주소 목록입니다.

TPG에는 IP 주소와 TCP 포트 번호가 포함됩니다. 이 기능을 사용하려면 다음을 수행해야 합니다.

- `itadm create-tpg` 명령을 사용하여 TPG를 *ip-address:port* 지정자 목록으로 만듭니다.
- `itadm modify-target -t` 명령을 사용하여 특정 iSCSI 대상을 TPG에 바인딩합니다.
- iSCSI 대상을 활성화하면 해당 대상과 연관된 TPG에 속하는 각 IP 주소와 포트에 대해 iSCSI 리스너가 만들어집니다.

TPG를 사용하면 특정 포트를 통해 검색되는 대상을 효율적으로 제어할 수 있습니다. 예를 들어, 특정 IP 주소를 통해서만 또는 iSER 가능 IP 주소 집합을 통해서만 사용할 수 있도록 iSCSI 대상을 제한할 수 있습니다.

---

주 - 대상 포털 그룹을 대상 그룹과 혼동하지 마십시오. 대상 그룹은 뷰를 만들 때 모두 동일하게 간주되는 SCSI 대상 포트 목록입니다. 뷰를 만들면 신속한 LUN 매핑에 도움이 됩니다. 각 뷰 항목은 대상 그룹(호스트 그룹) 및 LUN을 지정합니다. 대상 그룹 및 LUN 매핑에 대한 자세한 내용은 [256 페이지 “SCSI 논리 장치를 사용 가능하게 설정”](#) 및 [stmfadm\(1M\)](#)을 참조하십시오.

---

정적 및 iSNS 대상 검색에 대한 자세한 내용은 [246 페이지 “동적 또는 정적 대상 검색 구성”](#)을 참조하십시오. iSCSI 시작 프로그램은 `iscsiadm` 명령을 사용하여 TPG를 검색합니다. 자세한 내용은 [iscsiadm\(1M\)](#) 및 [itadm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## iSER과 함께 TPG 사용

SendTargets 검색과 iSER을 동시에 사용하는 경우 일반적인 규칙은 TPG를 사용하여 특정 iSCSI 대상 포트를 iSER 가능 IP 주소에만 연결하는 것입니다. 예를 들어, 대상 시스템에 A, B, C, D라는 네 개의 IP 주소가 있고 주소 B와 C만 iSER 가능인 경우 주소 B와 C를 TPG에 추가하고 대상 T에 할당할 수 있습니다.

이더넷과 IB(InfiniBand) 인터페이스가 모두 있는 iSCSI 시작 프로그램은 SendTargets 검색 방법을 사용하여 가능한 저장소 대상을 검색할 수 있습니다. TPG를 사용하지 않을 경우 시작 프로그램이 항상 IB 인터페이스보다 이더넷 인터페이스 사용을 선호할 수도 있습니다. 대상 T를 IB 인터페이스에 연결하면 시작 프로그램이 대상 T에 연결할 때 올바르게 IB 가능 인터페이스 사용을 선호합니다.

## ▼ iSCSI 대상에 대한 대상 포털 그룹을 만드는 방법

고유한 이름을 제공하여 TPG(대상 포털 그룹)를 만들 수 있으며, 이 경우 TPG 태그(2-65535 범위)가 자동으로 만들어집니다. TPG 태그 1은 대상의 TPG를 명시적으로 설정하지 않을 경우 사용되는 기본 TPG에 예약되어 있습니다. 기본 TPG의 포털은 포트 3260에서 모든 네트워크 인터페이스의 요청을 일치시킵니다.

다음 단계에서는 TPGB의 IP 주소에 대해 포트 8000을 사용하는 두 개의 TPG(TPGA 및 TPGB)를 만드는 방법을 보여 줍니다.

- 1 관리자로 전환합니다.

- 2 TPG 2개를 만듭니다.

```
target# itadm create-tpg TPGA 192.168.0.1 192.168.0.2
target# itadm create-tpg TPGB 192.168.0.2:8000 192.168.0.2:8000
```

주 - IPv4 포털은 점 구분 주소 표기법으로 지정됩니다(예: 192.168.0.1). IPv6 포털 주소는 대괄호로 묶어야 합니다.

- 3 두 TPG인 TPGA와 TPGB를 사용하도록 기존 iSCSI 대상을 구성합니다.

```
# itadm modify-target -t TPGA,TPGB eui.20387ab8943ef7548
```

- 4 만든 TPG를 확인합니다.

```
# itadm list-tpg -v
```

itadm delete-tpg 명령을 사용하여 TPG를 제거할 수 있습니다.

## ▼ iSCSI 디스크 액세스 방법

Oracle Solaris iSCSI 시작 프로그램이 장치를 검색한 후 자동으로 로그인 협상이 발생합니다. Oracle Solaris iSCSI 드라이버는 사용 가능한 LUN 수를 확인하고 장치 노드를 만듭니다. 그런 다음 iSCSI 장치를 다른 SCSI 장치처럼 간주할 수 있습니다.

LUN에 ZFS 저장소 풀을 만든 다음 ZFS 파일 시스템을 만들 수 있습니다.

format 유틸리티를 사용하여 로컬 시스템의 iSCSI 디스크를 볼 수 있습니다.

- 1 format 출력에서 iSCSI LUN 정보를 검토합니다.

```
initiator# format
0. c0t600144F0B5418B0000004DDAC7C10001d0 <Sun-COMSTAR-1.0 cyl 1022 alt 2 hd 128 sec 32>
   /scsi_vhci/disk@g600144f0b5418b0000004ddac7c10001
1. c8t0d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
   /pci@0,0/pci10de,375@f/pci108e,286@0/disk@0,0
2. c8t1d0 <Sun-STK RAID INT-V1.0-136.61GB>
   /pci@0,0/pci10de,375@f/pci108e,286@0/disk@1,0
3. c8t2d0 <Sun-STK RAID INT-V1.0-136.61GB>
```

```

/pci@0,0/pci10de,375@f/pci108e,286@0/disk@2,0
4. c8t3d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
/pci@0,0/pci10de,375@f/pci108e,286@0/disk@3,0
Specify disk (enter its number): 0
selecting c0t600144F0B5418B0000004DDAC7C10001d0
[disk formatted]

```

위 출력에서 디스크 0은 MPxIO에 의해 제어되는 iSCSI LUN입니다. 디스크 1-4는 로컬 디스크입니다.

## 2 iSCSI LUN에 ZFS 저장소 풀과 ZFS 파일 시스템을 만들 수 있습니다.

```

initiator# zpool create pool-name c0t600144F0B5418B0000004DDAC7C10001d0
initiator# zfs create pool-name/fs-name

```

ZFS 파일 시스템은 만들어 질 때 자동으로 마운트되며 부트 시 재마운트됩니다.

## SCSI 논리 장치를 사용 가능하게 설정

- 257 페이지 “모든 시스템이 논리 장치를 사용할 수 있게 하는 방법”
- 257 페이지 “선택한 시스템만 LUN에 액세스하도록 제한하는 방법”

단순히 논리 장치(LUN)를 STMF 프레임워크에 등록할 경우 네트워크의 호스트(시작 프로그램)가 LUN을 사용할 수 없습니다. 이 섹션에서는 다음 구성에 대해 LUN이 시작 프로그램 호스트에 표시되게 하는 방법을 설명합니다.

iSCSI, Fibre Channel 및 FCoE 구성의 경우 LUN에 액세스하려면 먼저 매핑해야 합니다. 다음 방법 중 하나를 선택할 수 있습니다. 두 방법은 모두 `stmfadm` 명령을 사용합니다.

- **단순 매핑** - 한 명령을 사용하여 모든 포트를 통해 LUN을 모든 시작 프로그램에 노출합니다. 모든 호스트가 LUN을 사용할 수 있게 하는 경우 이 방법을 사용합니다.
- **선택적 매핑** - LUN에 액세스할 수 있는 호스트를 지정할 수 있습니다. 선택한 호스트가 LUN을 사용할 수 있게 하는 경우 이 방법을 사용합니다. 이 프로세스에는 다음 단계가 포함됩니다.
  1. 호스트 그룹 정의 - 호스트 그룹은 동일한 LUN에 액세스할 수 있는 호스트(시작 프로그램) 집합에 제공되는 이름입니다. 단순 매핑처럼 모든 호스트에 동일한 LUN 집합이 표시되는 경우에는 이 단계가 필요 없습니다.
  2. 대상 그룹 정의 - 대상 그룹은 동일한 LUN 집합을 동일한 호스트 그룹 집합으로 내보내는 대상 포트 집합에 지정되는 이름입니다. 단순 매핑처럼 모든 호스트에 동일한 LUN 집합이 표시되는 경우에는 이 단계가 필요 없습니다.
  3. 각 논리 장치에 대해 하나 이상의 뷰 추가 - 뷰를 추가하면 LUN과 호스트 시작 프로그램 간에 브릿지가 만들어집니다. 호스트 그룹의 시작 프로그램이 대상 그룹에 포함된 대상 포트에 로그인하면 LUN이 표시됩니다.



주 - 대상 그룹을 TPG(대상 포털 그룹)와 혼동하지 마십시오. TPG는 iSCSI 대상이 수신 대기하는 IP 주소 목록입니다. TPG는 특정 IP 주소를 통해서만 사용하도록 iSCSI 대상을 제한하는 데 유용합니다. 대상 그룹에 대한 자세한 내용은 [stmfadm\(1M\)](#)을 참조하십시오.

뷰 항목은 호스트 그룹, 대상 그룹, 논리 장치(LUN) 및 LUN 식별자의 네 구성 요소로 이루어집니다. 이러한 네 구성 요소 중에서 LUN 식별자만 필수입니다. 다른 구성 요소는 생략할 경우 다음 기본값이 가정됩니다.

- 호스트 그룹을 생략하면 모든 시작 프로그램 값이 가정됩니다.
- 대상 그룹을 생략하면 모든 대상 값이 가정됩니다.
- LUN을 생략하면 시스템이 항목에 적합한 LUN을 선택합니다.

## ▼ 모든 시스템이 논리 장치를 사용할 수 있게 하는 방법

이 절차에서는 저장소 네트워크의 모든 시작 프로그램 호스트가 LUN을 사용할 수 있게 합니다.

- 1 LUN의 GUID(Global Unique Identification) 번호를 가져옵니다.

```
# stmfadm list-lu -v
```

- 2 논리 장치에 대한 뷰를 추가합니다.

```
3 # stmfadm add-view GUID-number
```

## ▼ 선택한 시스템만 LUN에 액세스하도록 제한하는 방법

저장소 네트워크에서 선택한 호스트만 LUN을 사용할 수 있게 제한하려면 이 절차를 사용합니다. Fibre Channel 포트를 사용하는 경우 먼저 WWN(World Wide Name)을 식별합니다. 그런 다음 선택적으로 LUN(논리 장치 번호)을 host-a 등의 포트에 매핑합니다. 또한 동일한 LUN 집합을 동일한 호스트 그룹으로 내보내는 지정된 대상 포트 집합에 대해 대상 그룹(targets-0)이 정의됩니다.

대상 그룹 구성에 대한 자세한 내용은 [stmfadm\(1M\)](#)을 참조하십시오.

- 1 관리자로 전환합니다.
- 2 시작 프로그램에서 FC(Fibre Channel) 포트를 식별합니다.

```
initiator# fcinfo hba-port
HBA Port WWN: 210000e08b195dae
      Port Mode: Initiator
      Port ID: 0
```

```

OS Device Name: /dev/cfg/c8
Manufacturer: QLogic Corp.
Model: 375-3108-xx
Firmware Version: 03.03.28
FCode/BIOS Version: fcode: 1.13;
Serial Number: not available
Driver Name: qlc
Driver Version: 20100408-3.01
Type: unknown
State: offline
Supported Speeds: 1Gb 2Gb
Current Speed: not established
Node WWN: 200000e08b195dae
NPIV Not Supported

```

```

.
.
.

```

### 3 호스트 그룹을 만듭니다.

```
target# stmfadm create-hg host-a 210000e08b195dae 210100e08b395dae
```

### 4 앞의 출력에서 식별된 WWN을 호스트 그룹의 구성원으로 추가합니다.

```
target# stmfadm add-hg-member -g host-a
```

### 5 대상 그룹을 만듭니다.

```
target# stmfadm create-tg targets-0
```

### 6 대상 이름을 추가하여 대상 그룹 구성원을 지정합니다.

각 SCSI 대상이 한 대상 그룹에만 구성원으로 속할 수 있습니다.

```
target# stmfadm add-tg-member -g targets-0 wwn.guid-number
```

### 7 LUN의 GUID 번호를 식별합니다.

```
target# stmfadm list-lu -v
```

### 8 호스트 그룹 이름과 LUN GUID 번호를 지정하여 뷰 항목을 추가하는 방법으로 LUN을 사용할 수 있게 합니다.

```
target# stmfadm add-view -h host-a -t targets-0 -n 1 guid-number
```

## COMSTAR를 사용하여 Fibre Channel 장치 구성

FC 네트워크 환경의 SPARC 시스템 또는 x86 시스템에서 COMSTAR 대상을 설정 및 구성하고 저장소 네트워크에서 액세스 가능하게 할 수 있습니다. 다음 필수 조건 단계를 완료했는지 확인합니다.

- 248 페이지 “iSCSI LUN을 만드는 방법”

## COMSTAR에 대해 Fibre Channel 포트 구성

FC(Fibre Channel) 포트 공급자는 FC에 사용되는 것과 동일한 HBA를 사용할 수 있습니다. 지정된 FC 포트를 시작 프로그램 또는 대상으로 사용할 수 있지만 둘 다로 사용할 수는 없습니다. HBA 포트의 일부는 대상 모드이고 나머지 포트는 시작 프로그램 모드에 있도록 이중 포트 또는 쿼드 포트 FC HBA를 구성할 수도 있습니다.

이 섹션의 절차는 QLogic HBA와 관련된 것입니다. 시작 프로그램 모드에서 HBA에 사용할 드라이버는 Qlogic 드라이버인 `qlc`입니다. `qlc` 드라이버는 시작 프로그램 모드에서만 작동하며 대상 모드에 사용할 수 없습니다. 대상 QLogic 4G HBA의 COMSTAR 드라이버는 `qlt`입니다.

시작 프로그램 모드와 대상 모드는 서로 다른 드라이버를 사용하기 때문에 HBA 포트에 연결하는 드라이버가 해당 기능을 대상이나 시작 프로그램으로 정의합니다. HBA의 PCI 장치 ID를 지정하여 모든 포트에 대해 드라이버를 지정할 수 있습니다. 또는 포트별로 드라이버를 구성할 수 있습니다. 두 방법은 모두 `update_drv` 명령을 사용하며 이 섹션에서 설명합니다. 자세한 내용은 `update_drv(1M)`를 참조하십시오.

### ▼ 기존 FC 포트 바인딩 표시 방법

HBA 포트를 변경하기 전에 먼저 기존 포트 바인딩을 확인합니다.

- 1 관리자로 전환합니다.
- 2 현재 포트 드라이버에 바인딩된 항목을 표시합니다.

이 예에서 현재 바인딩은 `pciex1077,2432`입니다.

```
# mdb -k
Loading modules: [ unix krtld genunix specsfs ...
> ::devbindings -q qlc
30001617a08 pciex1077,2432, instance #0 (driver name: qlc)
300016177e0 pciex1077,2432, instance #1 (driver name: qlc)
> $q
```

### ▼ 모든 FC 포트를 특정 모드로 설정하는 방법

이 절차에서는 특정 PCI 장치 ID를 가진 모든 HBA의 모든 포트를 대상 모드로 변경합니다. PCI 장치 ID 번호는 드라이버를 포트에 바인딩하여 해당 PCI 장치 ID를 가진 모든 HBA 포트(예: 모든 QLogic 4G PCI express HBA)를 대상 모드로 설정합니다.

- 1 관리자로 전환합니다.

**2 현재 바인딩을 제거합니다.**

이 예에서 qlc 드라이버는 이미 pciex1077,2432에 바인딩되어 있습니다. 해당 바인딩을 새 드라이버에 추가하려면 먼저 qlc의 기존 바인딩을 제거해야 합니다. 이 구문에는 작은따옴표 표시가 필요합니다.

```
# update_drv -d -i 'pciex1077,2432' qlc
Cannot unload module: qlc
Will be unloaded upon reboot.
```

이 메시지가 오류를 나타내지는 **않습니다**. 구성 파일은 업데이트되었지만 시스템을 재부트할 때까지 qlc 드라이버가 포트에 바인딩된 상태로 유지됩니다.

**3 새 바인딩을 설정합니다.**

이 예에서는 qlt가 업데이트됩니다. 이 구문에는 작은따옴표가 필요합니다.

```
# update_drv -a -i 'pciex1077,2432' qlt
devfsadm: driver failed to attach: qlt
Warning: Driver (qlt) successfully added to system but failed to attach
```

이 메시지가 오류를 나타내지는 **않습니다**. 시스템을 재부트할 때까지 qlc 드라이버가 포트에 바인딩된 상태로 유지됩니다. 시스템을 재부트하면 qlt 드라이버가 연결됩니다.

**4 시스템을 재부트하여 새 드라이버를 연결합니다. 그런 다음 바인딩을 재확인합니다.**

```
# init 6
.
.
.
# mdb -k

Loading modules: [ unix krtld genunix specfs dtrace ...
> ::devbindings -q qlt
30001615a08 pciex1077,2432, instance #0 (driver name: qlt)
30001615e0 pciex1077,2432, instance #1 (driver name: qlt)
> $q
```

**5 대상 모드 프레임워크가 HBA 포트에 액세스할 수 있는지 확인합니다.**

```
# stmfadm list-target -v
Target: wwn.210100E08BA54E60
Operational Status : Offline
Provider Name : qlt(1)
Alias : -
Sessions : 0
Target: wwn.210100E08BA54E60
Operational Status : Offline
Provider Name : qlt(0)
Alias : -
Sessions : 0
```

**▼ 선택한 FC 포트를 시작 프로그램 또는 대상 모드로 설정하는 방법**

이 절차에서는 경로 기반 바인딩을 사용합니다. 특정 장치 경로를 사용하여 현재 바인딩된 드라이버와는 다른 드라이버에 포트를 바인딩하는 방법을 보여 줍니다.

**1 관리자로 전환합니다.****2 HBA 포트 및 해당 장치 경로 목록을 표시합니다.**

이 예에서는 두 개의 포트가 있는 단일 HBA의 장치 경로를 보여 줍니다.

```
# luxadm -e port
/devices/pci@780/QLGC,qlc@0,1/fp@0,0:devctl CONNECTED
/devices/pci@780/QLGC,qlc@0/fp@0,0:devctl CONNECTED
```

**3 위쪽 포트를 대상 모드로 설정하고 아래쪽 포트는 시작 프로그램 모드로 유지합니다.**

경로에서 초기 /devices 부분을 제거하고 /fp@0...까지 모든 내용을 포함합니다. /devices 부분이 제거된 경로는 시스템이 qlt 드라이버를 바인딩하는 경로입니다.

이 구문에는 작은따옴표 표시가 필요합니다.

```
# update_drv -a -i '/pci@780/QLGC,qlc@0,1' qlt
devfsadm: driver failed to attach: qlt
Warning: Driver (qlt) successfully added to system but failed to attach.
```

이 메시지가 오류를 나타내지는 않습니다. 재부트할 때까지 qlc 드라이버가 포트에 바인딩된 상태로 유지됩니다. 재부트 도중 qlt 드라이버가 연결됩니다.

**4 시스템을 재부트하여 새 드라이버를 연결합니다. 그런 다음 바인딩을 재확인합니다.**

포트가 시작 프로그램 모드(qlc)에서 대상 모드(qlt)로 변경되었습니다.

```
# init 6
.
.
.
# mdb -k
.
.
.
> $q
```

**5 대상 모드 프레임워크가 HBA 포트에 액세스할 수 있는지 확인합니다.**

```
# stmfadm list-target -v
```

## 논리 장치를 FC 및 FCoE에 사용할 수 있게 설정

단순히 논리 장치(LUN)를 STMF 프레임워크에 등록할 경우 네트워크의 호스트(시작 프로그램)가 LUN을 사용할 수 없습니다. 논리 장치를 매핑하여 Fibre Channel 및 FCoE 구성의 시작 프로그램 호스트에 논리 장치가 표시되게 해야 합니다. 사용할 방법과 논리 장치 매핑 방법을 확인하려면 257 페이지 “모든 시스템이 논리 장치를 사용할 수 있게 하는 방법”을 참조하십시오. 두 방법은 모두 stmfadm 명령을 사용합니다. 아래 추가 단계는 FC 및 FCoE 구성을 위한 것입니다.

## ▼ 논리 장치를 FC 및 FCoE에 사용할 수 있게 하는 방법

이 절차에서는 저장소 네트워크의 FC 및 FCoE 구성에 대해 모든 호스트나 선택한 호스트가 LUN을 사용할 수 있게 합니다. 이 단계는 호스트에서 실행됩니다.

### 1 관리자로 전환합니다.

### 2 호스트가 LUN을 사용할 수 있게 합니다.

LUN의 GUID(Global Unique Identification) 번호를 가져옵니다.

```
# sbdadm list-lu
# stmfadm list-lu -v
```

호스트의 FC 또는 FCoE 포트에 대한 WWN을 식별합니다.

```
# fcinfo hba-port
HBA Port WWN: *210000e08b83378d*
OS Device Name: /dev/cfg/c4
Manufacturer: Qlogic Corp.
Model: QLA2462
Firmware Version: 4.0.27
Fcode/BIOS Version: N/A
Type: N-port
State: online
Supported Speeds: 1Gb 2Gb 4Gb
Current Speed: 4Gb
Node WWN: 210000e08b83378d
HBA Port WWN: *210100e08ba3378d*
OS Device Name: /dev/cfg/c5
Manufacturer: Qlogic Corp.
Model: QLA2462
Firmware Version: 4.0.27
Fcode/BIOS Version: N/A
Type: N-port
State: online
Supported Speeds: 1Gb 2Gb 4Gb
Current Speed: 4Gb
Node WWN: 210100e08ba3378d
```

### 3 뷰를 추가하고 매핑을 수행합니다.

257 페이지 “모든 시스템이 논리 장치를 사용할 수 있게 하는 방법”의 지침을 따릅니다.

### 4 다음 스크립트를 실행하여 LUN이 Oracle Solaris 시작 프로그램 호스트에 표시되는지 확인합니다.

```
#!/bin/ksh
fcinfo hba-port |grep "^HBA" | awk '{print $4}' | while read ln
do
    fcinfo remote-port -p $ln -s >/dev/null 2>&1
done
```

스크립트에 의해 시작 프로그램이 모든 포트를 통해 모든 LUN에 접근해야 하므로 후속 LUN이 모두 format 출력에 표시됩니다. LUN이 표시되지 않는 경우 format 명령을 다시 실행합니다. 그래도 LUN이 표시되지 않으면 svcs stmf 명령을 사용하여 대상에서

서비스가 사용으로 설정되었는지 확인합니다. 257 페이지 “모든 시스템이 논리 장치를 사용할 수 있게 하는 방법”에 설명된 대로 LUN에 대한 뷰 항목을 추가했는지도 확인합니다.

#### 5 LUN이 다른 시스템에 표시되는지 확인합니다.

- Linux 시작 프로그램 호스트의 경우 HBA 공급업체가 제공한 유틸리티를 실행하여 LUN이 표시되는지 확인합니다. 이 유틸리티는 구성 변경을 검색합니다.
- Windows 시작 프로그램 호스트의 경우 Control Panel(제어판) → Administrative Tools(관리 도구) → Computer Management(컴퓨터 관리) → Disk Management(디스크 관리)를 선택하여 논리 장치가 표시되는지 확인합니다. 그런 다음 Action(작업) 메뉴에서 Rescan Disks(디스크 다시 검사)를 선택합니다.

## COMSTAR를 사용하여 FCoE 장치 구성

FCoE(Fibre Channel over Ethernet) 네트워크 환경에서 COMSTAR 대상을 설정 및 구성하고 저장소 네트워크에서 액세스 가능하게 할 수 있습니다. 시작하기 전에 시스템이 필요한 전제 조건을 충족하는지 확인합니다.

- 248 페이지 “iSCSI LUN을 만드는 방법”
- 249 페이지 “iSCSI 대상을 만드는 방법”
- 251 페이지 “iSCSI 시작 프로그램 구성 방법”

### FCoE 포트 구성

- 264 페이지 “이더넷 인터페이스에서 802.3x PAUSE 및 점보 프레임 사용”
- 264 페이지 “FCoE 대상 포트를 만드는 방법”
- 264 페이지 “FCoE 대상 포트가 작동하는지 확인하는 방법”
- 265 페이지 “FCoE 대상 포트 삭제 방법”

FCoE 기능은 이더넷 인터페이스를 통해 제공됩니다. FCoE(Fibre Channel over Ethernet) 포트는 이더넷 인터페이스와 연관된 논리 엔티티입니다. Oracle Solaris 시스템에는 FCoE 포트와 이더넷 인터페이스의 일대일 매핑이 있습니다. FCoE 포트 한 개만 지정된 이더넷 인터페이스와 연결할 수 있습니다. FCoE와 IP는 동일한 이더넷 인터페이스를 공유할 수 없습니다. 따라서 이더넷 인터페이스에 FCoE 포트를 만들기 전에 인터페이스가 연결 해제(unplumb)되었는지 확인합니다.

FCoE 포트 구성은 재부트 후에도 유지됩니다. 구성된 모든 FCoE 포트가 시스템 재부트 후에 자동으로 만들어지고 온라인 상태로 설정됩니다.

FCoE 대상 포트의 경우 지속성 동작을 얻으려면 FCoE 대상 포트를 만들기 전에 다음 서비스를 사용으로 설정해야 합니다.

```
# svcadm enable svc:/system/fcoe_target:default
```

## 이더넷 인터페이스에서 802.3x PAUSE 및 점보 프레임 사용

이러한 설정은 이더넷 하드웨어와 드라이버마다 다를 수 있습니다. 대부분의 경우 이더넷 인터페이스의 `driver.conf` 파일을 수정한 후 재부트해야 합니다. 이러한 기능을 사용으로 설정하는 방법에 대한 자세한 내용은 이더넷 인터페이스에 대한 `driver.conf` 파일을 참조하십시오.

이더넷 인터페이스에 FCoE 포트를 만들기 전에 다음 필수 조건 단계 중 하나를 수행합니다.

- 이더넷 인터페이스에서 802.3x(PAUSE라고도 함)를 사용으로 설정합니다. 이렇게 하면 일관성 있는 이더넷 전송이 보장됩니다.
- 이더넷 인터페이스에서 점보 프레임(2.5KB 초과)을 사용으로 설정합니다. 광섬유 채널 데이터 프레임은 최대 2136바이트일 수 있습니다.

### ▼ FCoE 대상 포트를 만드는 방법

- 1 지정한 네트워크 인터페이스에 FCoE 대상 포트를 만듭니다.

```
# fcadm create-fcoe-port -t nxge0
```

선택한 이더넷 인터페이스가 여러 유니캐스트 주소를 지원하지 않는 경우(예: VMware 네트워크 인터페이스) 해당 인터페이스에서 명시적으로 무차별 모드를 사용으로 설정하라는 메시지가 표시됩니다.

- 2 메시지가 표시되면 무차별 모드를 사용으로 설정합니다.

```
# fcadm create-fcoe-port -t -f e1000g0
```

오류 메시지가 표시되지 않으면 FCoE 대상 포트가 만들어지고 FCoE 대상이 온라인 상태입니다.

### ▼ FCoE 대상 포트가 작동하는지 확인하는 방법

- 1 만든 FCoE 포트를 표시합니다.

```
# fcadm list-fcoe-ports
HBA Port WWN: 200000144fda7f66
Port Type: Target
MAC Name: nxge0
MTU Size: 9194
MAC Factory Address: 00144fda7f66
MAC Current Address: 0efc009a002a
Promiscuous Mode: On
```



## 2 호스트의 모든 대상 모드 Fibre Channel HBA 포트를 표시합니다.

```
# fcinfo hba-port -t
HBA Port WWN: 200000144fda7f66
  Port Mode: Target
  Port ID: 9a002a
  OS Device Name: Not Applicable
  Manufacturer: Sun Microsystems, Inc.
  Model: FCoE Virtual FC HBA
  Firmware Version: N/A
  FCode/BIOS Version: N/A
  Serial Number: N/A
  Driver Name: COMSTAR FCOET
  Driver Version: 1.0
  Type: F-port
  State: online
  Supported Speeds: 1Gb 10 Gb
  Current Speed: 10Gb
  Node WWN: 100000144fda7f66
```

## 3 FCoE 대상 포트 목록을 봅니다.

-v 옵션은 로그인한 시작 프로그램의 SCSI 세션 정보와 함께 대상에 대한 추가 정보를 표시합니다.

```
# stmfadm list-target -v
Target wwn.200000144FDA7F66
  Operational Status: Online
  Provider Name      : fcoet
  Alias              : fcoet1
  Sessions           : 1
  Initiator: wwn.210000E08B818343
  Alias: #QLA2342 FW:v3.03.25 DVR:v8.02.14.01
  Logged in since: ...
```

## ▼ FCoE 대상 포트 삭제 방법

필요한 경우 FCoE 기능을 사용 안함으로 설정할 수 있습니다.

### 1 FCoE 대상 포트를 오프라인 상태로 전환합니다.

```
# stmfadm offline-target wwn.200000144fda7f66
```

### 2 FCoE 대상 포트를 제거합니다.

```
# fcadm delete-fcoe-port nxge0
```

## COMSTAR를 사용하여 SRP 장치 구성

SCSI RDMA Protocol은 SCSI 데이터 전송 단계를 IB(Infiniband) RDMA(Remote Direct Memory Access) 작업에 매핑하여 SCSI 프로토콜을 가속화합니다. 그 결과, SRP 시작 프로그램은 비교적 낮은 CPU 사용률과 높은 데이터 속도로 COMSTAR SRP 대상에서 데이터를 읽고 쓸 수 있습니다.

COMSTAR SRP 대상을 설정 및 구성하고 IB(Infiniband) 패브릭에서 사용 가능하게 할 수 있습니다. 지원되는 IB HCA(Host Channel Adapter)가 대상 시스템에 설치되어 있을 때마다 SRP 대상을 사용할 수 있습니다.

- **IB HCA당 SCSI 대상 1개** - COMSTAR SRP 대상은 지원되는 각 IB HCA가 SCSI 대상으로 등록되는 단순 모델을 사용합니다. SCSI 대상은 작업 라우터를 포함하며 SCSI 전송(이 경우 SRP)과 SCSI 백엔드(STMF 및 SBD) 간의 연결로 작동하는 가상 객체입니다.

HCA는 물리적 포트를 여러 개 포함할 수 있습니다. 동일한 SCSI 대상이 HCA의 모든 포트 간에 공유됩니다. HCA를 나타내는 SCSI 대상은 해당 HCA의 모든 활성 포트를 통한 수신 연결에 자동으로 사용할 수 있습니다.
- **SRP 대상 eui 식별자** - IB 아키텍처에서는 제조업체가 각 HCA와 각 포트에 64비트 GUID를 할당합니다. 각 HCA에 대해 만들어진 COMSTAR SCSI 대상에는 HCA의 GUID에 해당하는 이름이 지정됩니다. 이 이름은 eui. HCA-GUID 형식을 사용합니다. 예를 들어, 대상 시스템에 HCA GUID가 0003BA0001002E48인 지원되는 IB HCA가 포함된 경우 이름이 eui.0003BA0001002E48인 SCSI 대상이 만들어집니다. 문자열 eui는 *Extended Unique Identifier*의 약자로, SCSI 및 IB 표준에서 모두 사용되는 GUID 클래스를 나타냅니다.
- **SRP 시작 프로그램 eui 식별자** - 이와 유사하게, SRP는 64비트 시작 프로그램 GUID를 사용하여 시작 프로그램 시스템을 식별합니다. 사용할 GUID는 SRP 시작 프로그램 구현에서 선택됩니다. 많은 시작 프로그램은 송신 연결에 사용 중인 HCA의 GUID를 사용합니다. 예를 들어, GUID 0003BA0001002EA5를 사용하는 시작 프로그램은 COMSTAR에 eui.0003BA0001002EA5로 알려집니다.

## SRP와 함께 COMSTAR 뷰 사용

COMSTAR 뷰 기능을 사용하면 256 페이지 “SCSI 논리 장치를 사용 가능하게 설정”에 설명된 대로 각 SCSI 대상 또는 시작 프로그램을 통해 액세스할 수 있는 저장소 논리 장치(LUN)를 제한하고 구성하는 대상 그룹 및 호스트 그룹을 만들 수 있습니다. SRP 시작 프로그램의 eui 식별자는 호스트 그룹에 추가됩니다. SRP SCSI 대상의 eui 식별자는 대상 그룹에 추가됩니다. 그런 다음 각 LUN의 뷰 항목에서 각 시작 프로그램이 액세스할 수 있는 특정 LUN 집합을 결정합니다.

## ▼ SRP 대상 서비스를 사용으로 설정하는 방법

COMSTAR SRP 대상의 COMSTAR 포트 공급자는 SMF(서비스 관리 기능)에서 관리됩니다. 주 SRP 대상 서비스는 `svc:/system/ibsrp/target:default`이며, `ibsrp/target`으로 축약할 수 있습니다.

SRP 패키지는 `storage/scsi-rdma/scsi-rdma-target`입니다.

- 1 관리자로 전환합니다.
- 2 SRP 대상 서비스를 반복적으로 사용으로 설정합니다.  

```
# svcadm enable -r ibsrp/target
```
- 3 SRP 대상 서비스 정보를 표시합니다.  

```
# svcs -l ibsrp/target
```

## ▼ SRP 대상 상태 확인 방법

- 1 관리자로 전환합니다.
- 2 시스템에 필요한 SRP SCSI 대상이 있는지 확인합니다.  

```
# srptadm list-target
Target HCA 21280001A0D0F0:
  Enabled          : true
  SRP Target Name  : eui.0021280001A0D0F0
  Operational Status : online
```

# iSCSI 기반 저장소 네트워크에서 인증 구성

iSCSI 장치에 대한 인증 설정은 선택 사항입니다.

보안 환경에서는 신뢰할 수 있는 시작 프로그램만 대상에 액세스할 수 있기 때문에 인증이 필요 없습니다.

덜 안전한 환경에서는 대상이 지정된 호스트에서 연결 요청을 보낸 것인지 확인할 수 없습니다. 이 경우 대상이 CHAP(Challenge-Handshake Authentication Protocol)를 사용하여 시작 프로그램을 인증할 수 있습니다.

CHAP 인증은 챌린지 및 응답 방식을 사용하며, 이 경우 대상이 ID를 증명하도록 시작 프로그램을 챌린지합니다. 챌린지/응답 방법이 작동하려면 대상이 시작 프로그램의 보안 키를 알고 있어야 하며 챌린지에 응답하도록 시작 프로그램을 설정해야 합니다. 어레이의 보안 키 설정에 대한 지침은 어레이 공급업체의 설명서를 참조하십시오.

iSCSI는 다음과 같이 단방향 및 양방향 인증을 지원합니다.

- **단방향 인증**을 사용하면 대상이 시작 프로그램의 ID를 인증할 수 있습니다. 단방향 인증은 대상 대신 시작 프로그램을 인증하기 위해 수행됩니다.
- **양방향 인증**은 시작 프로그램이 대상의 ID를 인증할 수 있게 하여 두번째 보안 레벨을 추가합니다. 양방향 인증은 양방향 인증의 수행 여부를 제어하는 시작 프로그램에서 구동됩니다. 대상에 필요한 유일한 설정은 CHAP 사용자와 CHAP 보안을 올바르게 정의해야 한다는 것입니다.

## ▼ iSCSI 시작 프로그램에 대해 CHAP 인증을 구성하는 방법

이 절차에서는 구성된 iSCSI 대상 장치에 안전하게 액세스하려는 로컬 시스템에 로그인했다고 가정합니다.

- COMSTAR iSCSI 대상에 대한 CHAP 보안 키의 길이는 최소 12자, 최대 255자여야 합니다. 일부 시작 프로그램에서 지원하는 보안 키의 최대 길이는 이보다 더 짧습니다.
- CHAP를 사용하여 자신을 식별하는 각 노드에는 사용자 이름과 암호가 모두 있어야 합니다. Oracle Solaris 11 환경에서는 CHAP 사용자 이름이 기본적으로 시작 프로그램 또는 대상 노드 이름(즉, iqn 이름)으로 설정됩니다. CHAP 사용자 이름은 512바이트보다 작은 임의 길이의 텍스트로 설정할 수 있습니다. 512바이트 길이 제한은 Oracle Solaris 11 제한입니다. 하지만 CHAP 사용자 이름을 설정하지 않을 경우 초기화 시 노드 이름으로 설정됩니다.

중앙 집중식 인증 서비스로 작동하는 타사 RADIUS 서버를 사용하여 CHAP 보안 키 관리를 단순화할 수 있습니다. RADIUS를 사용하는 경우 RADIUS 서버가 노드 이름 집합 및 일치하는 CHAP 보안 키를 저장합니다. 인증을 수행하는 시스템이 요청자의 노드 이름 및 요청자가 제공한 보안을 RADIUS 서버로 전달합니다. RADIUS 서버는 보안 키가 지정된 노드 이름을 인증하는 데 적합한 키인지 확인합니다. iSCSI와 iSER은 모두 RADIUS 서버 사용을 지원합니다.

타사 RADIUS 서버 사용에 대한 자세한 내용은 [270 페이지 “타사 RADIUS 서버를 사용하여 iSCSI 구성에서 CHAP 관리 단순화”](#)를 참조하십시오.

### 1 관리자로 전환합니다.

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

### 2 단방향 또는 양방향 CHAP를 구성할 것인지 결정합니다.

- 기본 방법인 단방향 인증을 사용하면 대상이 시작 프로그램을 검증할 수 있습니다. 3-5 단계만 완료합니다.
- 양방향 인증은 시작 프로그램이 대상을 인증할 수 있게 하여 두번째 보안 레벨을 추가합니다. 3-9 단계를 완료합니다.

**3 단방향 CHAP: 시작 프로그램의 보안 키를 설정합니다.**

다음 명령은 CHAP 보안 키를 정의하는 대화 상자를 시작합니다.

```
initiator# iscsiadm modify initiator-node --CHAP-secret
Enter CHAP secret: *****
Re-enter secret: *****
```

**4 (옵션) 단방향 CHAP: 시작 프로그램의 CHAP 사용자 이름을 설정합니다.**

기본적으로 시작 프로그램의 CHAP 사용자 이름은 시작 프로그램 노드 이름으로 설정됩니다.

고유한 시작 프로그램 CHAP 사용자 이름을 사용하려면 다음 명령을 사용합니다.

```
initiator# iscsiadm modify initiator-node --CHAP-name new-CHAP-name
```

**5 단방향 CHAP- 시작 프로그램에서 CHAP 인증을 사용으로 설정합니다.**

```
initiator# iscsiadm modify initiator-node --authentication CHAP
```

CHAP를 사용하려면 시작 프로그램 노드에 사용자 이름과 암호가 모두 있어야 합니다. 일반적으로 사용자 이름은 대상이 지정된 사용자 이름의 보안 키를 조회하는 데 사용됩니다.

**6 다음 중 하나를 선택하여 양방향 CHAP를 사용 또는 사용 안함으로 설정합니다.**

- 대상과의 연결에 대해 양방향 CHAP를 사용으로 설정합니다.

```
initiator# iscsiadm modify target-param -B enable target-iqn
```

- 양방향 CHAP를 사용 안함으로 설정합니다.

```
initiator# iscsiadm modify target-param -B disable target-iqn
```

**7 양방향 CHAP: 대상의 인증 방법을 CHAP로 설정합니다.**

```
initiator# iscsiadm modify target-param --authentication CHAP target-iqn
```

**8 양방향 CHAP: 대상을 식별하는 대상 장치 보안 키를 설정합니다.**

다음 명령은 CHAP 보안 키를 정의하는 대화 상자를 시작합니다.

```
initiator# iscsiadm modify target-param --CHAP-secret target-iqn
```

**9 양방향 CHAP: 대상이 대체 CHAP 사용자 이름을 사용하는 경우 대상을 식별하는 CHAP 이름을 설정합니다.**

기본적으로 대상의 CHAP 이름은 대상 이름으로 설정됩니다.

다음 명령을 사용하여 대상의 CHAP 이름을 변경할 수 있습니다.

```
initiator# iscsiadm modify target-param --CHAP-name target-CHAP-name
```

## ▼ iSCSI 대상에 대해 CHAP 인증을 구성하는 방법

이 절차에서는 iSCSI 대상을 포함하는 로컬 시스템에 로그인했다고 가정합니다.

- 1 관리자로 전환합니다.
- 2 단방향 또는 양방향 CHAP를 구성할 것인지 결정합니다.
  - 단방향 인증이 기본 방법입니다. 3-5 단계만 완료합니다.
  - 양방향 인증의 경우 3-7 단계를 완료합니다.
- 3 단방향/양방향 CHAP: 시작 프로그램이 CHAP를 사용하여 자신을 식별해야 하도록 대상을 구성합니다.

```
target# itadm modify-target -a chap target-iqn
```

- 4 단방향/양방향 CHAP: 시작 프로그램을 설명하는 시작 프로그램 컨텍스트를 만듭니다. 시작 프로그램의 전체 노드 이름과 시작 프로그램의 CHAP 보안 키를 사용하여 시작 프로그램 컨텍스트를 만듭니다.

```
target# itadm create-initiator -s initiator-iqn
Enter CHAP secret: *****
Re-enter secret: *****
```

- 5 단방향/양방향 CHAP: 시작 프로그램이 대체 CHAP 이름을 사용하는 경우 대체 이름으로 시작 프로그램 컨텍스트를 구성합니다.

```
target# itadm modify-initiator -u initiator-CHAP-name initiator-iqn
```

- 6 양방향 CHAP: 이 대상을 식별하는 대상 장치 보안 키를 설정합니다.

```
target# itadm modify-target -s target-iqn
Enter CHAP secret: *****
Re-enter secret: *****
```

- 7 (옵션) 양방향 CHAP: 대상이 대상 노드 이름(iqn) 이외의 대체 CHAP 사용자 이름을 사용하는 경우 대상을 수정합니다.

```
target# itadm modify-target -u target-CHAP-name target-iqn
```

## 타사 RADIUS 서버를 사용하여 iSCSI 구성에서 CHAP 관리 단순화

중앙 집중식 인증 서비스로 작동하는 타사 RADIUS 서버를 사용하여 CHAP 키 보안 관리를 단순화할 수 있습니다. 이 방법을 사용할 경우 각 시작 프로그램 노드에 기본 CHAP 이름을 사용하는 것이 좋습니다. 모든 시작 프로그램이 기본 CHAP 이름을 사용하는 일반적인 경우에는 대상에 시작 프로그램 컨텍스트를 만들 필요가 없습니다.

## ▼ iSCSI 대상에 대해 RADIUS 서버를 구성하는 방법

이 절차에서는 구성된 iSCSI 대상 장치에 안전하게 액세스하려는 로컬 시스템에 로그인했다고 가정합니다.

### 1 관리자로 전환합니다.

### 2 RADIUS 서버의 IP 주소와 포트를 사용하여 시작 프로그램 노드를 구성합니다.

기본 포트는 1812입니다. 이 구성은 대상 시스템의 모든 iSCSI 대상에 대해 한 번 완료됩니다.

```
initiator# itadm modify-defaults -r RADIUS-server-IP-address
Enter RADIUS secret: *****
Re-enter secret: *****
```

### 3 대상 시스템과 RADIUS 서버 간의 통신에 사용되는 공유 보안 키를 구성합니다.

```
initiator# itadm modify-defaults -d
Enter RADIUS secret: *****
Re-enter secret: *****
```

### 4 RADIUS 인증이 필요하도록 대상 시스템을 구성합니다.

개별 대상에 대해 또는 모든 대상의 기본값으로 이 구성을 수행할 수 있습니다.

```
initiator# itadm modify-target -a radius target-iqn
```

### 5 다음 구성 요소를 사용하여 RADIUS 서버를 구성합니다.

- 대상 노드의 ID(예: IP 주소)
- 대상 노드가 RADIUS 서버와 통신하는 데 사용하는 공유 보안 키
- 시작 프로그램의 CHAP 이름(예: iqn 이름) 및 인증해야 하는 각 시작 프로그램의 보안 키

## ▼ iSCSI 시작 프로그램에 대해 RADIUS 서버를 구성하는 방법

중앙 집중식 인증 서비스로 작동하는 타사 RADIUS 서버를 사용하여 CHAP 보안 키 관리를 단순화할 수 있습니다. 이 설정은 시작 프로그램이 양방향 CHAP 인증을 요청하는 경우에만 유용합니다. RADIUS 서버에 양방향 인증을 사용하는 경우 시작 프로그램의 CHAP 보안 키를 지정해야 하지만 시작 프로그램의 각 대상에 대한 CHAP 보안 키를 지정할 필요는 없습니다. 시작 프로그램 또는 대상에서 독립적으로 RADIUS를 구성할 수 있습니다. 시작 프로그램과 대상이 RADIUS를 사용하지 않아도 됩니다.

### 1 관리자로 전환합니다.

### 2 RADIUS 서버의 IP 주소와 포트를 사용하여 시작 프로그램 노드를 구성합니다.

기본 포트는 1812입니다.

```
# iscsiadm modify initiator-node --radius-server ip-address:1812
```

- 3 **RADIUS 서버의 공유 보안 키를 사용하여 시작 프로그램 노드를 구성합니다.**  
iSCSI가 서버와 상호 작용하려면 공유 보안을 사용하여 RADIUS 서버를 구성해야 합니다.  

```
# iscsiadm modify initiator-node --radius-shared-secret
Enter secret:
Re-enter secret
```
- 4 **RADIUS 서버를 사용으로 설정합니다.**  

```
# iscsiadm modify initiator-node --radius-access enable
```
- 5 **CHAP 양방향 인증의 다른 측면을 설정합니다.**  

```
# iscsiadm modify initiator-node --authentication CHAP
# iscsiadm modify target-param --bi-directional-authentication enable target-iqn
# iscsiadm modify target-param --authentication CHAP target-iqn
```
- 6 **다음 구성 요소를 사용하여 RADIUS 서버를 구성합니다.**
  - 이 노드의 ID(예: IP 주소)
  - 이 노드가 RADIUS 서버와 통신하는 데 사용하는 공유 보안 키
  - 대상의 CHAP 이름(예: iqn 이름) 및 인증해야 하는 각 대상의 보안 키

## Oracle Solaris iSCSI 및 RADIUS 서버 오류 메시지

이 섹션에서는 Oracle Solaris iSCSI 및 RADIUS 서버 구성과 관련된 오류 메시지에 대해 설명합니다. 복구를 위한 잠재적 해결 방법도 제공됩니다.

### empty RADIUS shared secret (빈 RADIUS 공유 보안)

**원인:** RADIUS 서버가 시작 프로그램에서 사용으로 설정되었지만 RADIUS 공유 보안 키가 설정되지 않았습니다.

**해결책:** RADIUS 공유 보안 키를 사용하여 시작 프로그램을 구성합니다. 자세한 내용은 271 페이지 “iSCSI 대상에 대해 RADIUS 서버를 구성하는 방법”을 참조하십시오.

### WARNING: RADIUS packet authentication failed (경고: RADIUS 패킷 인증 실패)

**원인:** 시작 프로그램이 RADIUS 데이터 패킷을 인증하지 못했습니다. 이 오류는 시작 프로그램 노드에 구성된 공유 보안 키가 RADIUS 서버의 공유 보안 키와 다른 경우에 발생할 수 있습니다.

**해결책:** 올바른 RADIUS 공유 보안을 사용하여 시작 프로그램을 재구성합니다. 자세한 내용은 271 페이지 “iSCSI 대상에 대해 RADIUS 서버를 구성하는 방법”을 참조하십시오.



## Oracle Solaris에서 iSCSI 다중 경로 장치 설정

MPxIO(Multipathed I/O)를 사용하면 I/O 장치의 단일 인스턴스에서 여러 호스트 제어기 인터페이스를 통해 I/O 장치에 액세스할 수 있습니다.

Oracle Solaris에서 iSCSI 다중 경로(MPxIO) 장치를 사용하는 경우 다음 지침을 고려합니다.

- **Oracle Solaris iSCSI 및 MPxIO** – MPxIO는 iSCSI 시작 프로그램에 대상당 여러 세션(MS/T)을 구성하는 Oracle Solaris iSCSI 구성에서 대상 포트 통합과 가용성을 지원합니다.
  - 두 개 이상의 NIC에 대한 통합과 페일오버를 위해 IPMP(IP Network Multipathing)를 사용합니다.
  - iSCSI 호스트의 기본 구성은 iSCSI 트래픽 전용으로 사용되는 NIC 2개가 있는 서버입니다. NIC는 IPMP를 사용하여 구성됩니다. 성능을 최적화하기 위해 비-iSCSI 트래픽에 대해 추가 NIC가 제공됩니다.
  - 활성 다중 경로를 구현하려면 Oracle Solaris의 iSCSI MS/T 기능과 IPMP 구성의 페일오버 및 중복을 사용해야 합니다.
    - IPMP 구성에서 NIC 1개가 실패하면 IPMP가 페일오버를 처리합니다. MPxIO 드라이버는 실패를 감지하지 못합니다. 비-IPMP 구성에서는 MPxIO 드라이버가 실패하고 경로를 오프라인으로 전환합니다.
    - IPMP 구성에서 대상 포트 1개가 실패하면 MPxIO 드라이버가 실패를 감지하고 페일오버를 제공합니다. 비-IPMP 구성에서도 MPxIO 드라이버가 실패를 감지하고 페일오버를 제공합니다.

대상당 여러 세션 구성에 대한 자세한 내용은 274 페이지 “대상에 대해 여러 iSCSI 세션을 사용으로 설정하는 방법”을 참조하십시오. IPMP 구성에 대한 자세한 내용은 **Oracle Solaris 관리: 네트워크 인터페이스 및 네트워크 가상화**의 15 장, “IPMP 관리”를 참조하십시오.

- **Oracle Solaris iSCSI, FC(Fibre Channel) 및 MPxIO** – MPxIO 드라이버는 더 복잡한 iSCSI/FC 구성에서 다음 동작을 제공합니다.
  - FC SAN에 이중 iSCSI - FC 브릿지가 있는 경우 iSCSI가 MPxIO에 대상 경로를 제공합니다. MPxIO는 LUN당 고유한 SCSI 식별자를 일치시키고, 동일한 경우 iSCSI 드라이버에 한 경로를 제공합니다.
  - iSCSI 및 FC를 모두 사용하여 대상을 연결하는 구성이 있는 경우 MPxIO 드라이버가 동일한 장치에 여러 전송을 제공할 수 있습니다. 이 구성에서는 MPxIO가 두 경로를 모두 사용합니다.
  - MPxIO에 iSCSI 및 FC를 사용하는 경우 `/etc/driver/drv/fp.conf` 및 `/driver/drv/iscsi.conf` 파일의 MPxIO 매개변수가 지원하려는 MPxIO 구성과 일치하는지 확인합니다. 예를 들어, `fp.conf`에서 MPxIO가 HBA에서 전역적으로 사용으로 설정되었는지, 아니면 포트별로 사용으로 설정되었는지 확인할 수 있습니다.

- **타사 하드웨어 고려 사항** - 타사 HBA가 Oracle Solaris iSCSI 및 MPxIO에서 작동하는지 확인합니다.

타사 HBA를 사용하는 경우 `/driver/driv/scsi_vhci.conf` 파일에 대한 대칭 옵션 정보를 공급업체에 문의해야 할 수도 있습니다.

## ▼ 대상에 대해 여러 iSCSI 세션을 사용으로 설정하는 방법

이 절차를 사용하여 단일 대상에 연결하는 iSCSI 세션을 여러 개 만들 수 있습니다. 이 시나리오는 로그인 재지정을 지원하거나 동일한 대상 포털 그룹에 여러 대상 포털이 있는 iSCSI 대상 장치에 유용합니다. Oracle Solaris의 SCSI Multipathing(MPxIO) 기능을 통해 대상당 iSCSI 세션을 여러 개 사용합니다. 호스트 쪽에서 여러 NIC를 사용하여 동일한 대상의 여러 포털에 연결하는 경우 더 높은 대역폭을 얻을 수도 있습니다.

MS/T 기능은 ISID(시작 프로그램의 세션 ID)를 변경하여 대상에 세션을 두 개 이상 만듭니다. 이 기능을 사용으로 설정하면 iSCSI 계층을 통해 여러 대상이 Oracle Solaris I/O 계층에 노출되도록 네트워크에 SCSI 계층 경로가 두 개 만들어집니다. MPxIO 드라이버는 이러한 경로의 예약을 처리합니다.

iSCSI가 MPxIO 경로와 상호 작용하는 방식에 대한 자세한 내용은 [273 페이지 “Oracle Solaris에서 iSCSI 다중 경로 장치 설정”](#)을 참조하십시오.

iSCSI 대상에 대해 여러 세션을 구성하기 전에 다음 항목을 검토합니다.

- 일반적인 MS/T 구성에는 구성된 세션이 두 개 이상 있습니다.  
하지만 저장소에서 여러 TPGT를 지원하고 호스트 시스템에서 SendTarget 검색 방법을 사용하는 경우 구성된 세션 수를 1로 설정할 수 있습니다. SendTarget 검색은 여러 경로의 존재를 자동으로 감지하며 여러 대상 세션이 만들어집니다.
- `/etc/driver/driv/iscsi.conf` 파일에서 `mpxio` 구성 매개변수가 사용으로 설정되었는지 확인합니다.  

```
# cd /etc/driver/driv
# grep mpxio iscsi.conf iscsi.conf
iscsi.conf:mpxio-disable="no";
```
- IPMP를 사용하여 여러 네트워크 연결이 구성되었는지 확인합니다.
- 여러 네트워크 연결을 사용할 수 있는지 확인합니다.

```
# ipadm show-addr
```

### 1 관리자로 전환합니다.

## 2 iSCSI 시작 프로그램과 대상의 현재 매개변수 값을 나열합니다.

### a. iSCSI 시작 프로그램의 현재 매개변수 값을 나열합니다.

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: z zr1200
.
.
Configured Sessions: 1
```

### b. iSCSI 대상 장치의 현재 매개변수 값을 나열합니다.

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
Alias: -
.
.
Configured Sessions: 1
```

구성된 세션 값은 대상 포털 그룹의 각 대상 이름에 대해 만들어지도록 구성된 iSCSI 세션 수입니다.

## 3 다음 중 하나를 선택하여 구성된 세션 수를 시작 프로그램 노드에서 수정하여 모든 대상에 적용하거나 대상 레벨에서 수정하여 특정 대상에 적용합니다.

대상의 세션 수는 1에서 4 사이여야 합니다.

- iSCSI 시작 프로그램 노드에 원하는 매개변수 값을 적용합니다.

```
initiator# iscsiadm modify initiator-node -c 2
```

- iSCSI 대상에 원하는 매개변수 값을 적용합니다.

```
initiator# iscsiadm modify target-param -c 2 iqn.1992-08.com.abcstorage:sn.84186266
```

- 구성된 세션을 하나 이상의 로컬 IP 주소에 바인딩합니다.

구성된 세션을 로컬 IP 주소에 바인딩할 수도 있습니다. 이 방법을 사용할 경우 하나 이상의 로컬 IP 주소가 쉼표로 구분된 목록에 제공됩니다. 각 IP 주소는 iSCSI 세션을 나타냅니다. initiator-node 또는 target-param 레벨에서 이 방법을 사용할 수도 있습니다. 예를 들면 다음과 같습니다.

```
initiator# iscsiadm modify initiator-node -c 10.0.0.1,10.0.0.2
```

---

주 - 지정된 IP 주소의 경로를 지정할 수 없는 경우 해당 주소가 무시되며 기본 Oracle Solaris 경로 및 IP 주소가 이 세션에 사용됩니다.

---

## 4 매개변수가 수정되었는지 확인합니다.

### a. 시작 프로그램 노드에 대한 업데이트된 정보를 표시합니다.

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
```

```
Initiator node alias: zvr1200
.
.
Configured Sessions: 2
```

**b. 대상 노드에 대한 업데이트된 정보를 표시합니다.**

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
Alias: -
.
.
Configured Sessions: 2
```

- 5 여러 경로를 나열하여 OS 장치 이름이 `iscsiadm list` 출력과 일치하는지, 경로 수가 2개 이상인지 확인합니다.

```
initiator# mpathadm list lu
```

## iSCSI 구성 모니터

`iscsiadm list` 명령을 사용하여 iSCSI 시작 프로그램 및 대상 장치에 대한 정보를 표시할 수 있습니다.

### ▼ iSCSI 구성 정보 표시 방법

- 1 관리자로 전환합니다.
- 2 iSCSI 시작 프로그램에 대한 정보를 표시합니다.

예를 들면 다음과 같습니다.

```
# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zvr1200
Login Parameters (Default/Configured):
    Header Digest: NONE/-
    Data Digest: NONE/-
Authentication Type: NONE
RADIUS Server: NONE
RADIUS access: unknown
Configured Sessions: 1
```

- 3 사용 중인 검색 방법에 대한 정보를 표시합니다.
- 예를 들면 다음과 같습니다.

```
# iscsiadm list discovery
Discovery:
    Static: enabled
    Send Targets: enabled
    iSNS: enabled
```

## 예 14-1 iSCSI 대상 정보 표시

다음 예에서는 특정 iSCSI 대상의 매개변수 값을 표시하는 방법을 보여 줍니다.

```
# iscsiadm list target-param iqn.1992-08.com.abcstorage:sn.33592219
Target: iqn.1992-08.com.abcstorage:sn.33592219
```

iscsiadm list target-param -v 명령은 다음 정보를 표시합니다.

- 대상의 인증 값
- 대상 로그인 매개변수의 기본값
- 각 로그인 매개변수에 대해 구성된 값

iscsiadm list target-param -v 명령은 / 지정자 앞에 *default* 매개변수 값을 표시하고 / 지정자 뒤에 *configured* 매개변수 값을 표시합니다. 매개변수를 구성하지 않은 경우 해당 값이 하이픈(-)으로 표시됩니다. 자세한 내용은 다음 예를 참조하십시오.

```
# iscsiadm list target-param -v eui.50060e8004275511 Target: eui.50060e8004275511
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
Login Parameters (Default/Configured):
  Data Sequence In Order: yes/-
  Data PDU In Order: yes/-
  Default Time To Retain: 20/-
  Default Time To Wait: 2/-
  Error Recovery Level: 0/-
  First Burst Length: 65536/-
  Immediate Data: yes/-
  Initial Ready To Transfer (R2T): yes/-
  Max Burst Length: 262144/-
  Max Outstanding R2T: 1/-
  Max Receive Data Segment Length: 65536/-
  Max Connections: 1/-
  Header Digest: NONE/-
  Data Digest: NONE/-
Configured Sessions: 1
```

다음 출력 예는 대상과 시작 프로그램 간에 협상된 매개변수를 표시합니다.

```
# iscsiadm list target -v eui.50060e8004275511
Target: eui.50060e8004275511
TPGT: 1
ISID: 4000002a0000
Connections: 1
  CID: 0
    IP address (Local): 172.90.101.71:32813
    IP address (Peer): 172.90.101.40:3260
    Discovery Method: Static
    Login Parameters (Negotiated):
      Data Sequence In Order: yes
      Data PDU In Order: yes
      Default Time To Retain: 0
      Default Time To Wait: 3
      Error Recovery Level: 0
```

```

First Burst Length: 65536
Immediate Data: yes
Initial Ready To Transfer (R2T): yes
Max Burst Length: 262144
Max Outstanding R2T: 1
Max Receive Data Segment Length: 65536
Max Connections: 1
Header Digest: NONE
Data Digest: NONE

```

## iSCSI 시작 프로그램 및 대상 매개변수 수정

iSCSI 시작 프로그램과 iSCSI 대상 장치에서 모두 매개변수를 수정할 수 있습니다. 하지만 iSCSI 시작 프로그램에서 수정할 수 있는 매개변수는 다음 한 개뿐입니다.

- iSCSI 시작 프로그램 노드 이름 - 시작 프로그램 노드 이름을 다른 이름으로 변경할 수 있습니다. 시작 프로그램 노드 이름을 변경하는 경우 이름 변경 시 iSNS 서버의 검색 도메인 구성에 따라 iSNS에 의해 검색된 대상이 시작 프로그램의 대상 목록에서 제거될 수 있습니다. 자세한 내용은 [280 페이지 “iSCSI 시작 프로그램 및 대상 매개변수 수정 방법”](#)을 참조하십시오.
- 헤더 다이제스트 - NONE(기본값) 또는 CRC32
- 데이터 다이제스트 - NONE(기본값) 또는 CRC32
- 인증 및 CHAP 보안 키 - 인증 설정에 대한 자세한 내용은 [268 페이지 “iSCSI 시작 프로그램에 대해 CHAP 인증을 구성하는 방법”](#)을 참조하십시오.

iSCSI 드라이버는 iSCSI 시작 프로그램 및 iSCSI 대상 장치 매개변수의 기본값을 제공합니다. iSCSI 시작 프로그램의 매개변수를 수정하면 iSCSI 대상 장치에 다른 값이 없는 경우 수정된 매개변수가 iSCSI 대상 장치에 상속됩니다.



**주의** - 대상 소프트웨어가 수정할 매개변수를 지원하는지 확인합니다. 지원하지 않을 경우 iSCSI 대상 장치에 로그인할 수 없습니다. 지원되는 매개변수 목록은 어레이 설명서를 참조하십시오.

시작 프로그램과 대상 간의 I/O가 완료된 후에만 iSCSI 매개변수를 수정할 수 있습니다. `iscsiadm modify` 명령을 사용하여 변경한 후 iSCSI 드라이버가 세션을 재연결합니다.

## iSCSI 매개변수 조정

iSCSI 매개변수를 조정하여 iSCSI 시작 프로그램의 다양한 응답 또는 연결 시간 값을 조정할 수 있습니다. 시작 프로그램 시스템이 연결된 모든 대상에 대해 시작 프로그램의 매개변수를 조정할 것인지 또는 특정 대상에 대해 시작 프로그램을 조정할 것인지에 따라 iSCSI 매개변수를 조정할 수 있습니다.

특정 iSCSI 대상의 매개변수 값을 변경하려면 다음 명령을 사용합니다.

```
iscsiadm modify target-param -T tunable-prop=value target-name
```

모든 대상에 대해 매개변수 값을 조정하려면 다음 명령을 사용합니다.

```
iscsiadm modify initiator-node -T tunable-prop=value
```

아래 조정 가능 매개변수는 활성 연결에 적용되며, iSCSI 시작 프로그램 및 시작 프로그램에 연결하는 대상의 동작을 변경합니다. 매개변수를 동적으로 조정하는 기능은 iSCSI 시작 프로그램을 구성할 때 유연성을 제공합니다.

표 14-1 iSCSI 조정 가능 매개변수

매개변수 이름	설명	유효한 값(초)	기본값(초)
recv-login-rsp-timeout	세션 로그인 응답 시간 - iSCSI 시작 프로그램이 지정된 iSCSI 대상에서 iSCSI 세션 로그인 요청의 응답을 기다리는 시간을 지정합니다.	0 - 3600	60
conn-login-max	최대 연결 재시도 시간 - iSCSI 시작 프로그램과 대상 간의 I/O 시간이 초과되거나 연결이 실패한 후 iSCSI 시작 프로그램이 대상에 연결을 시도하는 최대 횟수를 결정합니다.	0 - 3600	180
polling-login-delay	로그인 재시도 시간 간격 - iSCSI 시작 프로그램과 대상 간의 I/O 시간이 초과되거나 연결이 실패한 후 각 iSCSI 세션 로그인을 재시도하는 시간 간격을 결정합니다.	0 - 3600	60

## ▼ iSCSI 매개변수 조정 방법

### 1 조정 가능한 모든 iSCSI 매개변수를 표시합니다.

모든 대상에 대한 iSCSI 매개변수 정보를 표시합니다.

```
# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:e00000000000.4e36d278
Initiator node alias: unknown
  Login Parameters (Default/Configured):
    Header Digest: NONE/-
    Data Digest: NONE/-
    Max Connections: 65535/-
  Authentication Type: NONE
  RADIUS Server: NONE
  RADIUS Access: disabled
  Tunable Parameters (Default/Configured):
    Session Login Response Time: 60/-
    Maximum Connection Retry Time: 180/-
    Login Retry Time Interval: 60/-
  Configured Sessions: 1
```

특정 대상에 대한 iSCSI 매개변수 정보를 표시합니다.

```
# iscsiadm list target-param [target-name]
```

예를 들면 다음과 같습니다.

```
# iscsiadm list target-param -v iqn.1986-03.com.sun:02:47ac0
506-cd48-67f5-fc0d-ab7544d37538
Target: iqn.1986-03.com.sun:02:47ac0506-cd48-67f5-fc0d-ab7544d37538
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
Login Parameters (Default/Configured):
  Data Sequence In Order: yes/-
  Data PDU In Order: yes/-
  Default Time To Retain: 20/-
  Default Time To Wait: 2/-
  Error Recovery Level: 0/-
  First Burst Length: 65536/-
  Immediate Data: yes/-
  Initial Ready To Transfer (R2T): yes/-
  Max Burst Length: 262144/-
  Max Outstanding R2T: 1/-
  Max Receive Data Segment Length: 8192/-
  Max Connections: 65535/-
  Header Digest: NONE/-
  Data Digest: NONE/-
Tunable Parameters (Default/Configured):
  Session Login Response Time: 60/-
  Maximum Connection Retry Time: 180/-
  Login Retry Time Interval: 60/-
Configured Sessions: 1
```

## 2 iSCSI 매개변수를 조정합니다.

예를 들어, 한 대상에 대해 최대 연결 재시도 시간을 90초로 설정하려면 다음 구문을 사용합니다.

```
# iscsiadm modify target-param -T conn-login-max=90 iqn.1986-03.com.sun:
02:47ac0506-cd48-67f5-fc0d-ab7544d37538
```

예를 들어, 모든 대상에 대해 최대 연결 재시도 시간을 90초로 설정하려면 다음 구문을 사용합니다.

```
# iscsiadm modify initiator-node -T conn-login-max=90
```

## ▼ iSCSI 시작 프로그램 및 대상 매개변수 수정 방법

이 절차의 첫번째 부분에서는 iSCSI 시작 프로그램의 수정된 매개변수가 iSCSI 대상 장치에 상속되는 방식을 보여 줍니다. 이 절차의 두번째 부분에서는 iSCSI 대상 장치의 매개변수를 실제로 수정하는 방법을 보여 줍니다.

이 선택적 절차에서는 iSCSI 대상 장치에 대한 액세스가 이미 구성되어 있는 로컬 시스템에 로그인했다고 가정합니다.



- 1 관리자로 전환합니다.
- 2 iSCSI 시작 프로그램과 대상 장치의 현재 매개변수 값을 나열합니다.
  - a. iSCSI 시작 프로그램의 현재 매개변수 값을 나열합니다.

```

initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
  Login Parameters (Default/Configured):
    Header Digest: NONE/-
    Data Digest: NONE/-
  Authentication Type: NONE
  RADIUS Server: NONE
  RADIUS access: unknown
  Configured Sessions: 1

```

- b. iSCSI 대상 장치의 현재 매개변수 값을 나열합니다.

```

initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
  Login Parameters (Default/Configured):
    Data Sequence In Order: yes/-
    Data PDU In Order: yes/-
    Default Time To Retain: 20/-
    Default Time To Wait: 2/-
    Error Recovery Level: 0/-
    First Burst Length: 65536/-
    Immediate Data: yes/-
    Initial Ready To Transfer (R2T): yes/-
    Max Burst Length: 262144/-
    Max Outstanding R2T: 1/-
    Max Receive Data Segment Length: 65536/-
    Max Connections: 1/-
    Header Digest: NONE/-
    Data Digest: NONE/-
  Configured Sessions: 1

```

헤더 다이제스트 및 데이터 다이제스트 매개변수가 현재 iSCSI 시작 프로그램과 iSCSI 대상 장치에 대해 모두 NONE으로 설정되어 있습니다.

iSCSI 대상 장치의 기본 매개변수 값을 검토하려면 예 14-1의 `iscsiadm list target-param` 출력을 참조하십시오.

- 3 iSCSI 시작 프로그램의 매개변수를 수정합니다.

예를 들어, 헤더 다이제스트를 CRC32로 설정합니다.

```

initiator# iscsiadm modify initiator-node -h CRC32

```

시작 프로그램 노드 이름을 변경하는 경우 새 이름이 대상과 동일한 검색 도메인에 속하지 않으면 iSNS에 의해 검색된 대상이 로그아웃되고 시작 프로그램의 대상 목록에서

제거될 수 있습니다. 하지만 대상이 사용 중인 경우에는 제거되지 않습니다. 예를 들어, 이러한 대상에서 파일이 열려 있거나 파일 시스템이 마운트된 경우 대상이 제거되지 않습니다.

대상과 새 시작 프로그램 노드 이름이 동일한 검색 도메인에 속하는 경우 이름 변경 후에 새 대상이 표시될 수도 있습니다.

#### 4 매개변수가 수정되었는지 확인합니다.

##### a. iSCSI 시작 프로그램에 대한 업데이트된 매개변수 정보를 표시합니다.

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: z zr1200
  Login Parameters (Default/Configured):
    Header Digest: NONE/CRC32
    Data Digest: NONE/-
  Authentication Type: NONE
  RADIUS Server: NONE
  RADIUS access: unknown
  Configured Sessions: 1
```

이제 헤더 다이제스트가 CRC32로 설정되었습니다.

##### b. iSCSI 대상 장치에 대한 업데이트된 매개변수 정보를 표시합니다. 예를 들면 다음과 같습니다.

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
  Login Parameters (Default/Configured):
    Data Sequence In Order: yes/-
    Data PDU In Order: yes/-
    Default Time To Retain: 20/-
    Default Time To Wait: 2/-
    Error Recovery Level: 0/-
    First Burst Length: 65536/-
    Immediate Data: yes/-
    Initial Ready To Transfer (R2T): yes/-
    Max Burst Length: 262144/-
    Max Outstanding R2T: 1/-
    Max Receive Data Segment Length: 65536/-
    Max Connections: 1/-
    Header Digest: CRC32/-
    Data Digest: NONE/-
  Configured Sessions: 1
```

이제 헤더 다이제스트가 CRC32로 설정되었습니다.

#### 5 iSCSI 시작 프로그램이 iSCSI 대상에 재연결되었는지 확인합니다.

```
initiator# iscsiadm list target -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
TPGT: 2
```

```

ISID: 4000002a0000
Connections: 1
CID: 0
IP address (Local): nnn.nn.nn.nnn:64369
IP address (Peer): nnn.nn.nn.nnn:3260
Discovery Method: SendTargets
Login Parameters (Negotiated):
.
.
Header Digest: CRC32
Data Digest: NONE

```

#### 6 (옵션) iSCSI 시작 프로그램 매개변수나 iSCSI 대상 장치 매개변수를 재설정합니다.

`iscsiadm modify` 명령을 사용하여 매개변수를 기본값으로 재설정할 수 있습니다. 또는 `iscsiadm remove` 명령을 사용하여 모든 매개변수를 기본값으로 재설정할 수 있습니다.

`iscsiadm modify target-param` 명령은 명령줄에서 지정된 매개변수만 변경합니다.

다음 예에서는 헤더 다이제스트를 NONE으로 재설정하는 방법을 보여 줍니다.

```
initiator# iscsiadm modify target-param -h none iqn.1992-08.com.abcstorage:sn...
```

`iscsiadm remove target-param` 명령에 대한 자세한 내용은 [iscsiadm\(1M\)](#)을 참조하십시오.

## iSCSI 구성 문제 해결

다음 도구를 사용하여 일반적인 iSCSI 구성 문제를 해결할 수 있습니다.

- `snoop` - 이 도구는 iSCSI 패킷을 지원하도록 업데이트되었습니다.
- `wireshark` - 이 제품은 <http://www.wireshark.org/>에서 사용할 수 있습니다.

두 도구는 모두 포트 3260에서 iSCSI 패킷을 필터링할 수 있습니다.

다음 섹션에서는 다양한 iSCSI 문제 및 오류 메시지를 해결하는 방법에 대해 설명합니다.

## 로컬 시스템이 iSCSI 대상에 연결되어 있지 않음

### ▼ iSCSI 연결 문제 해결 방법

1 관리자로 전환합니다.

2 iSCSI 대상 정보를 나열합니다.

예를 들면 다음과 같습니다.

```
initiator# iscsiadm list target
Target: iqn.2001-05.com.abcstorage:6-8a0900-37ad70401-bcfff02df8a421df-zzr1200-01
```

```
TPGT: default
ISID: 400002a0000
Connections: 0
```

- 3 **iscsiadm list target** 출력이 연결이 나열되지 않는 경우 `/var/adm/messages` 파일에서 연결이 실패한 가능한 이유를 확인합니다.

ping 명령을 사용하거나, telnet 명령으로 저장 장치의 iSCSI 포트에 연결하여 iSCSI 서비스를 사용할 수 있는지 확인하는 방법으로 연결에 액세스할 수 있는지 확인할 수도 있습니다. 기본 포트는 3260입니다.

또한 저장 장치의 로그 파일에서 오류를 확인합니다.

- 4 **iscsiadm list target** 출력이 대상이 나열되지 않는 경우 `/var/adm/messages` 파일에서 가능한 원인을 확인합니다.

SendTargets를 검색 방법으로 사용하는 경우 `-v` 옵션을 통해 `discovery-address`를 나열하여 필요한 대상이 호스트에 표시되는지 확인합니다. 예를 들면 다음과 같습니다.

```
initiator# iscsiadm list discovery-address -v 10.0.0.1
Discovery Address: 10.0.0.1:3260
  Target name: eui.210000203787dfc0
    Target address:      10.0.0.1:11824
  Target name: eui.210000203787e07b
    Target address:      10.0.0.1:11824
```

iSNS를 검색 방법으로 사용하는 경우 iSNS 검색 방법을 사용으로 설정하고 `-v` 옵션을 통해 `isns-server`를 나열하여 필요한 대상이 호스트에 표시되는지 확인합니다. 예를 들면 다음과 같습니다.

```
initiator# iscsiadm list isns-server -v
iSNS Server IP Address: 10.20.56.56:3205
  Target name: iqn.1992-08.com.xyz:sn.1234566
    Target address:      10.20.57.161:3260, 1
  Target name: iqn.2003-10.com.abc:group-0:154:abc-65-01
    Target address:      10.20.56.206:3260, 1
  Target name: iqn.2003-10.com.abc:group-0:154:abc-65-02
    Target address:      10.20.56.206:3260, 1
.
.
.
```

## 로컬 시스템에서 iSCSI 장치 또는 디스크를 사용할 수 없음

### ▼ iSCSI 장치 또는 디스크를 사용할 수 없는 문제를 해결하는 방법

- 1 관리자로 전환합니다.
- 2 열거 도중 이 대상에서 검색된 LUN을 식별합니다.  
-s 옵션은 열거 도중 이 대상에서 검색된 LUN을 보여 줍니다.  
예를 들면 다음과 같습니다.  

```
# iscsiadm list target -S
Target: iqn.2001-05.com.abcstorage:6-8a0900-37ad70401-bcfff02df8a421df-zzr1200-01
TPGT: default
ISID: 4000002a0000
Connections: 1
LUN: 0
    Vendor: ABCSTOR
    Product: 0010
    OS Device Name: /dev/rdisk/c3t34d0s2
,

```
- 3 `/var/adm/messages` 파일을 검토하여 오류가 보고되었는지 확인합니다.  
나열되어야 하는 LUN이 표시되지 않는 경우 이 로그 파일을 확인합니다.
- 4 저장 장치의 로그 파일에서 오류를 확인합니다.
- 5 저장 장치 LUN 마스킹이 올바르게 구성되었는지 확인합니다.

## iSNS 검색 방법을 사용할 때 LUN 마스킹 사용

특정 시작 프로그램에 대한 저장소 인증을 제어하는 수단으로 iSNS 검색 도메인을 사용하지 마십시오. 대신 LUN 마스킹을 사용하여 권한이 부여된 시작 프로그램만 LUN에 액세스할 수 있게 합니다.

대상이 사용되는 동안 검색 도메인에서 대상을 제거하면 iSCSI 시작 프로그램이 이 대상에서 로그아웃되지 않습니다. 이 시작 프로그램이 이 대상(및 연관된 LUN)에 액세스할 수 없게 하려면 LUN 마스킹을 사용해야 합니다. 검색 도메인에서 대상을 제거하는 것만으로는 충분하지 않습니다.

## 일반적인 iSCSI 오류 메시지

이 섹션에서는 `/var/adm/messages` 파일에서 발견될 수 있는 iSCSI 메시지 및 복구를 위한 잠재적 해결 방법에 대해 설명합니다.

메시지 형식은 다음과 같습니다.

<code>iscsi TYPE (OID) STRING (STATUS-CLASS#/STATUS-DETAIL#)</code>	
<code>TYPE</code>	연결 또는 세션입니다.
<code>OID</code>	연결 또는 세션의 객체 ID입니다. 이 ID는 OS 인스턴스에 고유합니다.
<code>STRING</code>	조건에 대한 설명입니다.
<code>STATUS-CLASS#/STATUS-DETAIL#</code>	이 값은 RFC 3720에 정의된 대로 iSCSI 로그인 응답에 반환됩니다.

`iscsi connection(OID) login failed - Miscellaneous iSCSI initiator errors.`

**원인:** 특정 형식의 시작 프로그램 오류로 인해 장치 로그인이 실패했습니다.

`iscsi connection(OID) login failed - Initiator could not be successfully authenticated.`

**원인:** 장치가 시작 프로그램을 성공적으로 인증하지 못했습니다.

**해결책:** 해당하는 경우 CHAP 이름, CHAP 암호 또는 RADIUS 서버에 대한 설정이 올바른지 확인합니다.

`iscsi connection(OID) login failed - Initiator is not allowed access to the given target.`

**원인:** 장치에서 시작 프로그램이 iSCSI 대상 장치에 액세스하도록 허용할 수 없습니다.

**해결책:** 시작 프로그램 이름을 확인하고, 저장 장치에서 올바르게 마스킹 또는 프로비전되었는지 확인합니다.

`iscsi connection(OID) login failed - Requested ITN does not exist at this address.`

**원인:** 장치가 요청되는 ITN(iSCSI 대상 이름)에 대한 액세스를 제공하지 않습니다.

**해결책:** 시작 프로그램 검색 정보가 올바르게 지정되었는지, 그리고 저장 장치가 올바르게 구성되었는지 확인합니다.

`iscsi connection(OID) login failed - Requested ITN has been removed and no forwarding address is provided.`

**원인:** 장치가 요청되는 ITN(iSCSI 대상 이름)에 대한 액세스를 더 이상 제공할 수 없습니다.

**해결책:** 시작 프로그램 검색 정보를 올바르게 지정했는지, 그리고 저장 장치를 올바르게 구성했는지 확인합니다.

`iscsi connection(OID) login failed - Requested iSCSI version range is not supported by the target.`

**원인:** 저장 장치가 시작 프로그램의 iSCSI 버전을 지원하지 않습니다.

`iscsi connection(OID) login failed - No more connections can be accepted on this Session ID (SSID).`

**원인:** 저장 장치가 iSCSI 대상 장치에 대한 이 시작 프로그램 노드의 다른 연결을 허용할 수 없습니다.

`iscsi connection(OID) login failed - Missing parameters (e.g., iSCSI initiator and/or target name).`

**원인:** 저장 장치가 시작 프로그램 또는 대상 이름을 올바르게 지정하지 않았다고 보고합니다.

**해결책:** iSCSI 시작 프로그램 또는 대상 이름을 올바르게 지정하십시오.

`iscsi connection(OID) login failed - Target hardware or software error.`

**원인:** 저장 장치에서 하드웨어 또는 소프트웨어 오류가 발생했습니다.

**해결책:** 추가 지원을 받으려면 저장소 설명서를 참조하거나 저장소 공급업체에 문의하십시오.

`iscsi connection(OID) login failed - iSCSI service or target is not currently operational.`

**원인:** 저장 장치가 현재 작동하지 않습니다.

**해결책:** 추가 지원을 받으려면 저장소 설명서를 참조하거나 저장소 공급업체에 문의하십시오.

`iscsi connection(OID) login failed - Target has insufficient session, connection or other resources.`

**원인:** 저장 장치의 리소스가 부족합니다.

**해결책:** 추가 지원을 받으려면 저장소 설명서를 참조하거나 저장소 공급업체에 문의하십시오.

iscsi connection(OID) login failed - unable to initialize authentication  
 iscsi connection( OID) login failed - unable to set authentication  
 iscsi connection(OID) login failed - unable to set username  
 iscsi connection(OID) login failed - unable to set password  
 iscsi connection(OID) login failed - unable to set ipsec  
 iscsi connection(OID) login failed - unable to set remote authentication  
**원인:** 시작 프로그램이 인증을 올바르게 초기화하거나 설정하지 못했습니다.

**해결책:** 인증에 대한 시작 프로그램 설정을 올바르게 구성했는지 확인합니다.

iscsi connection(OID) login failed - unable to make login pdu  
**원인:** 시작 프로그램이 시작 프로그램 또는 저장 장치 설정에 따라 PDU(로그인 페이로드 데이터)를 만들지 못했습니다.

**해결책:** 대상 로그인 매개변수나 기본값이 아닌 기타 설정을 재설정해 보십시오.

iscsi connection(OID) login failed - failed to transfer login  
 iscsi connection( OID) login failed - failed to receive login response  
**원인:** 시작 프로그램이 네트워크 연결을 통해 PDU(로그인 페이로드 데이터)를 전송하거나 수신하지 못했습니다.

**해결책:** 네트워크에 연결할 수 있는지 확인합니다.

iscsi connection(OID) login failed - received invalid login response (OP CODE)  
**원인:** 저장 장치가 예기치 못한 응답으로 로그인에 응답했습니다.

iscsi connection(OID) login failed - login failed to authenticate with target  
**원인:** 시작 프로그램이 저장 장치를 인증하지 못했습니다.

**해결책:** 인증에 대한 시작 프로그램 설정을 올바르게 구성했는지 확인합니다.

iscsi connection(OID) login failed - initiator name is required  
**원인:** 모든 작업을 수행하려면 시작 프로그램 이름을 구성해야 합니다.

**해결책:** 시작 프로그램 이름이 구성되었는지 확인합니다.

iscsi connection(OID) login failed - authentication receive failed  
 iscsi connection( OID) login failed - authentication transmit failed  
**원인:** 시작 프로그램이 인증 정보를 전송하거나 수신하지 못했습니다.

**해결책:** 해당하는 경우 저장 장치 또는 RADIUS 서버와의 네트워크 연결을 확인합니다.



iscsi connection(OID) login failed - login redirection invalid

**원인:** 저장 장치가 시작 프로그램을 잘못된 대상으로 재지정하려고 했습니다.

**해결책:** 추가 지원을 받으려면 저장소 설명서를 참조하거나 저장소 공급업체에 문의하십시오.

iscsi connection(OID) login failed - target protocol group tag mismatch, expected <TPGT>, received <TPGT>

**원인:** 시작 프로그램과 대상의 TPGT(대상 포털 그룹 태그)가 일치하지 않습니다.

**해결책:** 시작 프로그램 또는 저장 장치의 TPGT 검색 설정을 확인합니다.

iscsi connection(OID) login failed - can't accept PARAMETER in security stage

**원인:** 로그인 보안 단계에서 장치가 지원되지 않는 로그인 매개변수로 응답했습니다.

**해결책:** 참조를 위해 매개변수 이름이 표시됩니다. 추가 지원을 받으려면 저장소 설명서를 참조하거나 저장소 공급업체에 문의하십시오.

iscsi connection(OID) login failed - HeaderDigest=CRC32 is required, can't accept VALUE

iscsi connection(OID) login failed - DataDigest=CRC32 is required, can't accept VALUE

**원인:** 시작 프로그램이 이 대상에 대해 CRC32로 설정된 HeaderDigest 또는 DataDigest만 허용하도록 구성되었습니다. 장치가 VALUE 값을 반환했습니다.

**해결책:** 시작 프로그램 및 장치 다이제스트 설정이 호환되는지 확인합니다.

iscsi connection(OID) login failed - HeaderDigest=None is required, can't accept VALUE

iscsi connection(OID) login failed - DataDigest=None is required, can't accept VALUE

**원인:** 시작 프로그램이 이 대상에 대해 NONE으로 설정된 HeaderDigest 또는 DataDigest만 허용하도록 구성되었습니다. 장치가 VALUE 값을 반환했습니다.

**해결책:** 시작 프로그램 및 장치 다이제스트 설정이 호환되는지 확인합니다.

iscsi connection(OID) login failed - can't accept PARAMETER

**원인:** 시작 프로그램이 이 매개변수를 지원하지 않습니다.

iscsi connection(OID) login failed - can't accept MaxOutstandingR2T VALUE

**원인:** 시작 프로그램이 표시된 VALUE의 MaxOutstandingR2T를 허용하지 않습니다.

`iscsi connection(OID) login failed - can't accept MaxConnections VALUE`

**원인:** 시작 프로그램이 표시된 *VALUE*의 최대 연결 횟수를 허용하지 않습니다.

`iscsi connection(OID) login failed - can't accept ErrorRecoveryLevel VALUE`

**원인:** 시작 프로그램이 표시된 *VALUE*의 오류 복구 레벨을 허용하지 않습니다.

`iscsi session(OID) NAME offline`

**원인:** 이 대상 *NAME*에 대한 모든 연결이 제거되었거나 실패했습니다.

`iscsi connection(OID) failure - unable to schedule enumeration`

**원인:** 시작 프로그램이 이 대상의 LUN을 열거하지 못했습니다.

**해결책:** `devfsadm -i iscsi` 명령을 실행하여 LUN 열거를 강제할 수 있습니다. 자세한 내용은 `devfsadm(1M)`을 참조하십시오.

`iscsi connection(OID) unable to connect to target NAME (errno:ERRNO )`

**원인:** 시작 프로그램이 네트워크 연결을 설정하지 못했습니다.

**해결책:** 연결 실패의 특정 *ERRNO*에 대한 자세한 내용은 `/usr/include/sys/errno.h` 파일을 참조하십시오.

## Oracle Solaris iSNS(Internet Storage Name Service) 구성 및 관리

---

이 장에서는 iSNS(Internet Storage Name Service) 개요를 제공하고 Oracle Solaris iSNS 서버 구성, iSNS 서버 관리 및 iSNS 클라이언트 관리 방법에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

- 291 페이지 “iSNS 기술(개요)”
- 293 페이지 “iSNS 서버 구성”
- 299 페이지 “iSNS 서버 및 클라이언트 관리”

### iSNS 기술(개요)

iSNS(Internet Storage Name Service)는 IP SAN(Storage Area Network) 내에서 iSCSI 시작 프로그램과 대상을 동적으로 검색할 수 있게 하는 프로토콜입니다. iSNS 프로토콜은 다음 서비스를 제공하여 iSCSI 장치를 식별, 연결 및 관리할 수 있게 합니다.

- **이름 등록 및 검색:** 저장할 데이터의 소스(시작 프로그램이라고 함) 및 저장소 객체(대상이라고 함)는 속성과 주소를 등록한 다음 액세스 가능한 저장 장치에 대한 정보를 동적으로 얻을 수 있습니다.
- **검색 도메인 및 로그온 제어:** 일반 저장소 네트워크의 리소스는 네트워크 관리 응용 프로그램을 통해 관리할 수 있는 **검색 도메인**이라는 그룹으로 나뉩니다. 검색 도메인은 고유한 액세스 제어를 통해 사용으로 설정되지 않은 대상에 대한 액세스 제어를 제공하는 동시에 각 시작 프로그램의 로그온 프로세스를 네트워크에서 사용할 수 있는 대상의 해당 일부로 제한하여 보안을 향상시킵니다.
- **상태 변경 알림:** iSNS 서버는 새로 만든 디스크 LUN(논리 장치 번호), 저장소 리소스의 오프라인 전환, 검색 도메인 구성원 변경, 네트워크의 링크 오류와 같은 네트워크 이벤트를 iSNS 클라이언트에 알립니다. 이러한 알림을 통해 네트워크가 토폴로지 변경에 신속하게 대응할 수 있으며, 이것이 확장성과 가용성의 핵심 요소입니다. 이것은 선택적 서비스입니다.
- **엔티티 상태 조회:** iSNS 서버가 iSNS 클라이언트를 사용할 수 있는지 확인합니다. 그 결과, 상태 변경 알림이 실행될 수도 있습니다. 이것은 선택적 서비스입니다.

단순 구성에서는 저장할 데이터의 소스(시작 프로그램)가 저장소 객체(대상)와 데이터를 교환합니다. 시작 프로그램이 대상을 찾을 수 있고 대상이 항상 시작 프로그램을 인식합니다. 예를 들어, Oracle StorageTek 5320 NAS(Network Attached Storage) 제품은 데이터를 저장하기 때문에 iSCSI 대상입니다. 데이터 관리 응용 프로그램이나 시작 프로그램으로 작동하는 네트워크 인터페이스 카드와 같은 다양한 iSCSI 클라이언트에서 데이터가 제공됩니다. 하지만 크고 복잡한 구성에서는 각 대상에 대해 각 시작 프로그램을 구성하고 각 대상이 각 시작 프로그램을 인식하는 것이 어렵고 시간이 오래 걸립니다. iSNS 서버는 이 문제를 해결하기 위해 검색 및 보안 방식을 사용하여 시작 프로그램과 대상을 동적 및 자동으로 식별하고 권한이 부여된 리소스에 대한 연결을 관리합니다.

Oracle Solaris 시스템이 iSNS 서버로 구성된 후 모든 대상과 시작 프로그램이 서버에 등록할 수 있습니다. 대상과 시작 프로그램은 iSNS 서버의 iSCSI 클라이언트 또는 노드가 됩니다. 이러한 클라이언트는 기본 검색 도메인 세트의 유일한 도메인인 기본 검색 도메인의 구성원입니다. 기본 검색 도메인 세트를 사용으로 설정하면 iSNS 서버가 간단한 방식으로 클라이언트에 대해 iSNS(iSCSI Name Service)를 제공할 수 있습니다.

iSCSI Name Service의 기능을 활용하려면 검색 도메인 세트와 검색 도메인을 여러 개 만듭니다. 그런 다음 클라이언트를 여러 도메인에 할당하여 구성원이 겹치게 합니다. iSNS 서버는 하나 이상의 검색 도메인에 속한 구성원으로 클라이언트 상태를 추적합니다. 예를 들어, 저장소 네트워크에 새 저장 장치를 추가하고 iSNS 서버에 등록하면 기본 검색 도메인 세트의 기본 검색 도메인에 있습니다. 그런 다음 시작 프로그램이 리소스로 사용할 검색 도메인에 이 대상을 할당합니다. iSNS 서버가 기본 검색 도메인 세트의 기본 검색 도메인 구성원으로 속한 이 대상을 제거합니다.

모든 시작 프로그램과 대상이 하나 이상의 검색 도메인에 할당됩니다. 시작 프로그램을 특정 검색 도메인에 할당하면 동일한 검색 도메인 세트의 해당 대상에만 액세스하도록 제한됩니다. 시작 프로그램을 여러 검색 도메인에 할당하면 시작 프로그램의 검색 도메인을 포함하는 모든 검색 도메인 세트에서 대상을 찾고 사용할 수 있습니다. 다른 검색 도메인 세트의 클라이언트에는 영향을 주지 않고 해당 검색 도메인 세트를 사용 및 사용 안함으로 설정하여 클라이언트에 대한 액세스를 관리할 수 있습니다.

예를 들어, 한 사이트에 기본 검색 도메인 외에도 Production과 Research라는 두 개의 검색 도메인이 있습니다. 두 검색 도메인 세트 내에는 기본 도메인 외에도 Development, Operations 및 Finance라는 세 개의 도메인이 있습니다. Development 검색 도메인은 Research 검색 도메인 세트에 있고 Operations는 Production 도메인 세트에 있고 Finance는 두 검색 도메인 세트에 모두 속하는 구성원입니다. 각 클라이언트가 가장 많이 사용되는 검색 도메인 세트에 할당되었습니다. Operations 검색 도메인의 데이터 응용 프로그램은 Production 검색 도메인 세트의 구성원이기 때문에 이 검색 도메인 세트에서 저장 장치를 찾고 액세스할 수 있지만 Research 검색 도메인 세트의 저장 장치에는 액세스할 수 없습니다. Finance 검색 도메인의 데이터 응용 프로그램은 Production 및 Research 검색 도메인 세트의 구성원이기 때문에 두 세트에서 모두 저장 장치를 찾을 수 있습니다. Research 검색 도메인 세트가 사용 안함으로 설정된 경우 Finance 검색 도메인의 시작 프로그램이 Research 저장 장치에 액세스할 수 없지만 Production 검색 도메인 세트의 저장 장치는 계속 액세스할 수 있습니다.

# iSNS 서버 구성

다음 작업 맵과 섹션에 설명된 대로 iSNS 서버를 구성할 수 있습니다.

작업	수행 방법
1. iSNS 서버의 기본 등록 정보를 적용하거나 변경합니다.	
A. 서버의 상태 변경 알림	294 페이지 “서버 상태 변경에 대한 알림 설정 방법”
B. 클라이언트 가용성 확인 시도 횟수	295 페이지 “클라이언트 조회 재시도 횟수 설정 방법”
C. 클라이언트 데이터를 저장하는 파일 위치	295 페이지 “데이터 저장소 위치 지정 방법”
2. iSNS 서버를 사용으로 설정하고 설정을 표시합니다.	294 페이지 “iSNS 서버 패키지 설치 방법” 296 페이지 “현재 서버 구성 표시 방법”
3. iSNS 서버에 모든 클라이언트를 등록합니다.	클라이언트 관리 인터페이스의 iSCSI 구성 기능을 사용하여 iSNS 서버의 IP 주소를 지정하고 검색을 허용합니다.
4. 기본 검색 도메인 세트를 사용으로 설정합니다.	296 페이지 “기본 검색 도메인 세트를 사용으로 설정하는 방법”

이러한 작업 후에는 iSNS 서버가 최소 방식으로 작동합니다. 모든 클라이언트가 기본 검색 도메인에 있고 할당되지 않은 상태입니다. 각 클라이언트가 다른 모든 클라이언트를 식별하고 액세스할 수 있습니다.

5. 사이트에 대한 검색 도메인 세트를 만듭니다.	296 페이지 “검색 도메인 세트를 만드는 방법”
6. 사이트에 대한 검색 도메인을 만듭니다.	297 페이지 “검색 도메인을 만드는 방법”
7. 각 검색 도메인을 하나 이상의 검색 도메인 세트에 추가합니다.	297 페이지 “검색 도메인 세트에 검색 도메인을 추가하는 방법”
8. 클라이언트를 하나 이상의 검색 도메인에 할당합니다.	298 페이지 “클라이언트를 검색 도메인에 할당하는 방법”
9. 검색 도메인의 클라이언트 구성원과 검색 도메인 세트에 있는 검색 도메인의 구성원을 확인합니다.	299 페이지 “검색 도메인 세트의 상태 표시 방법” 299 페이지 “검색 도메인의 상태 표시 방법” 300 페이지 “클라이언트 상태 표시 방법”

다음 섹션에서는 iSNS 환경을 설정하는 지침을 제공합니다. 다음 내용으로 구성되어 있습니다.

- 294 페이지 “iSNS 관리 설정 지정”

- 296 페이지 “명령줄 인터페이스를 사용하여 iSNS 구성”

## iSNS 관리 설정 지정

이 섹션에서는 iSNS 서비스의 기본 관리 설정을 변경하고 iSNS 데몬을 시작하는 절차를 제공합니다. iSNS 서버가 시작된 후 설정을 변경할 경우 iSNS 서버를 새로 고쳐야 합니다. 데이터 저장소 위치를 변경할 경우 iSNS 서버를 다시 시작해야 합니다.

- 294 페이지 “iSNS 서버 패키지 설치 방법”
- 294 페이지 “서버 상태 변경에 대한 알림 설정 방법”
- 295 페이지 “클라이언트 조회 재시도 횟수 설정 방법”
- 295 페이지 “데이터 저장소 위치 지정 방법”

이러한 작업에 대한 자세한 내용은 `isns(1M)` 명령의 매뉴얼 페이지를 참조하십시오.

### ▼ iSNS 서버 패키지 설치 방법

iSNS 서버 패키지를 설치하고 iSNS 서비스를 시작합니다.

- 1 관리자로 전환합니다.

- 2 iSNS 서버 패키지를 설치합니다.

```
# pkg install service/storage/isns
```

- 3 iSNS 서비스를 사용으로 설정합니다.

```
# svcadm enable isns_server
```

- 4 서비스가 실행 중인지 확인합니다.

```
# svcs svc:/network/isns_server:default
STATE          STIME          FMRI
online         16:10:49      svc:/network/isns_server:default
```

### ▼ 서버 상태 변경에 대한 알림 설정 방법

기본적으로 iSNS 서버를 사용할 수 없는 경우 모든 클라이언트가 알림을 받습니다. 이러한 알림을 사용 안함으로 설정하려면 `Management_SCNs_Enabled` 등록 정보를 변경합니다.

- 1 “iSNS 서버 관리” RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “RBAC 초기 구성\(작업 맵\)”](#)을 참조하십시오.

- 2 `svccfg` 명령을 사용하여 등록 정보를 사용 안함으로 설정합니다.

```
# svccfg -s svc:/network/isns_server setprop config/Management_SCN_Enabled=false
```

### 3 서버 구성을 재로드합니다.

```
# svcadm refresh svc:/network/isns_server
```

## ▼ 클라이언트 조회 재시도 횟수 설정 방법

기본 재시도 횟수는 3회입니다. 서버가 3회의 조회에 대해 응답을 받지 못하면 클라이언트를 사용할 수 없다고 등록합니다. 재시도 횟수를 변경하려면 *ESI Retry Threshold* 등록 정보의 값을 변경합니다.

### 1 “iSNS 서버 관리” RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “RBAC 초기 구성\(작업 맵\)”](#)을 참조하십시오.

### 2 svccfg 명령을 사용하여 등록 정보를 6회 재시도 등으로 변경합니다.

```
# svccfg -s svc:/network/isns_server setprop config/ESI_retry_threshold_count=6
```

### 3 서버 구성을 재로드합니다.

```
# svcadm refresh svc:/network/isns_server
```

## ▼ 데이터 저장소 위치 지정 방법

클라이언트 데이터가 포함된 파일의 기본 위치와 이름은 */etc/isns/isnsdata.xml*입니다. 백업 iSNS 서버가 하나 이상 포함된 복합 네트워크 환경에서는 모든 서버가 사용할 수 있도록 데이터 저장소가 공용 위치에 있어야 합니다. *data\_store\_location* 등록 정보를 사용하여 새 위치를 지정합니다. 파일 이름을 변경할 수도 있습니다.

### 1 “iSNS 서버 관리” RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “RBAC 초기 구성\(작업 맵\)”](#)을 참조하십시오.

### 2 svccfg 명령을 사용하여 위치를 */etc/isns2/isns\_data.xml* 등으로 변경합니다.

```
# svccfg -s svc:/network/isns_server setprop config/data_store_location="/etc/isns2/isns_data.xml"
```

### 3 서버가 사용으로 설정된 후 데이터 저장소 위치를 변경할 경우 서버를 다시 시작해야 합니다.

```
# svcadm restart svc:/network/isns_server
```

## 명령줄 인터페이스를 사용하여 iSNS 구성

이 섹션에서는 명령줄 인터페이스를 사용하여 iSNS 서버를 구성하는 절차를 제공합니다. 다음 내용으로 구성되어 있습니다.

- 296 페이지 “현재 서버 구성 표시 방법”
- 296 페이지 “기본 검색 도메인 세트를 사용으로 설정하는 방법”
- 296 페이지 “검색 도메인 세트를 만드는 방법”
- 297 페이지 “검색 도메인을 만드는 방법”
- 297 페이지 “검색 도메인 세트에 검색 도메인을 추가하는 방법”
- 298 페이지 “클라이언트를 검색 도메인에 할당하는 방법”

이 절차에서는 `isnsadm(1M)` 명령을 사용합니다. 모든 명령 옵션에 대한 자세한 설명은 매뉴얼 페이지를 참조하십시오.

### ▼ 현재 서버 구성 표시 방법

- 다음 명령은 iSNS 서버의 등록 정보를 보여 줍니다.

```
# isnsadm show-config
Data Store Location: /etc/isns/isnsdata.xml
Entity Status Inquiry Non-Response Threshold: 3
Management SCN Enabled: yes
Authorized Control Node Names: -
```

### ▼ 기본 검색 도메인 세트를 사용으로 설정하는 방법

- 1 “iSNS 서버 관리” RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “RBAC 초기 구성\(작업 맵\)”](#)을 참조하십시오.

- 2 기본 검색 도메인 세트를 활성화합니다.

```
# isnsadm enable-dd-set Default
```

### ▼ 검색 도메인 세트를 만드는 방법

- 1 “iSNS 서버 관리” RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “RBAC 초기 구성\(작업 맵\)”](#)을 참조하십시오.

- 2 검색 도메인 세트를 만듭니다.

```
# isnsadm create-dd-set set_name
```



- 3 검색 도메인 세트를 사용으로 설정합니다.

```
# isnsadm enable-dd-set set_name
```

- 4 새 검색 도메인 세트를 포함하여 모든 검색 도메인 세트를 봅니다.

```
# isnsadm list-dd-set -v
  DD Set name: Default
    State: Enabled
  DD Set name:set_name
    State: Enabled
```

검색 도메인 세트 목록에는 기본 검색 도메인 세트와 새 검색 도메인 세트가 모두 포함됩니다.

### ▼ 검색 도메인을 만드는 방법

새 검색 도메인은 기본 검색 도메인 세트의 구성원입니다. 검색 도메인을 만든 후 새 검색 도메인 세트에 추가합니다.

- 1 “iSNS 서버 관리” RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “RBAC 초기 구성\(작업 맵\)”](#)을 참조하십시오.

- 2 검색 도메인을 만듭니다.

```
# isnsadm create-dd domain_name
```

- 3 기본 검색 도메인 세트의 새 검색 도메인을 봅니다.

```
# isnsadm list-dd-set
  DD name: name
    DD set(s): Default
```

- 4 다른 검색 도메인을 만듭니다.

### ▼ 검색 도메인 세트에 검색 도메인을 추가하는 방법

이 작업은 기본 검색 도메인 세트에서 검색 도메인을 제거하고 지정한 검색 도메인 세트에 추가합니다. 새 검색 도메인 세트가 사용으로 설정되었으므로 iSNS 서버가 검색 도메인의 모든 클라이언트를 검색할 수 있습니다.

검색 도메인 및 검색 도메인 세트의 구성원을 나열할 수 있는 권한은 필요 없습니다.

- 1 “iSNS 서버 관리” RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “RBAC 초기 구성\(작업 맵\)”](#)을 참조하십시오.

- 2 검색 도메인을 나열하여 추가할 검색 도메인을 식별합니다.

```
# isnsadm list-dd -v
```

- 3 검색 도메인 세트를 나열하여 새 검색 도메인의 컨테이너로 사용할 검색 도메인 세트를 식별합니다.

```
# isnsadm list-dd-set
```

- 4 검색 도메인을 원하는 검색 도메인 세트로 이동합니다.

```
# isnsadm add-dd domain_name -s set_name
```

- 5 검색 도메인 세트에 새로 추가된 항목을 봅니다.

```
# isnsadm list-dd-set -v domain_name
```

## ▼ 클라이언트를 검색 도메인에 할당하는 방법

시작하기 전에 클라이언트의 관리 인터페이스를 사용하여 클라이언트를 등록합니다. iSCSI 구성 기능을 사용하여 iSNS 서버의 IP 주소를 지정하고 iSNS 서버에 의한 클라이언트 검색을 허용합니다.

- 1 "iSNS 서버 관리" RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 "RBAC 초기 구성\(작업 맵\)"](#)을 참조하십시오.

- 2 클라이언트가 iSNS 서버에 등록되었는지 확인합니다.

```
# isnsadm list-node
    iSCSI Name: iqn.1986-03.com.sun:01:000e0c9f10da.45173FEA.engr
    Alias: STK5320_NAS
    Type: Target
    .
    iSCSI Name: iqn.1986-03.com.sun:01:000e0c9f10da.454F00A2.acct
    Alias:
    Type: Initiator
```

출력에 클라이언트의 iSCSI 이름이 표시됩니다.

- 3 검색 도메인의 이름을 확인합니다.

```
# isnsadm list-dd
```

- 4 클라이언트를 검색 도메인에 추가합니다.

```
# isnsadm add-node -d domain_name iSCSI_Name
```

예를 들어, "STK5320\_NAS"라는 대상을 Eng-dd 검색 도메인에 추가하려면 다음 구문을 사용합니다.

```
# isnsadm add-node -d Eng-dd iqn.1986-03.com.sun:01:000e0c9f10da.454F00A2.engr
```

- 5 검색 도메인의 모든 클라이언트를 나열하여 클라이언트가 추가되었는지 확인합니다.

```
# isnsadm list-dd -v domain_name
```

예를 들어, Eng-dd 검색 도메인을 확인하려면 다음 구문을 사용합니다.

```
# isnsadm list-dd -v Eng-dd
DD name: Eng-dd
DD set: Development-dds
iSCSI Name: iqn.1986-03.com.sun:01:000e0c9f10da.45173FEA.engr
iSCSI Name: iqn.1986-03.com.sun:01:000e0c9f10da.454F00A2.acct
iSCSI name: iqn.1986-03.com.sun:01:e00000000000.46fd8e2b
```

## iSNS 서버 및 클라이언트 관리

이 섹션에서는 iSNS 검색 도메인 세트와 해당 구성원, 시작 프로그램 및 대상을 유지 관리하는 방법에 대해 설명합니다. 사이트가 확장됨에 따라 다음 섹션에 설명된 대로 클라이언트, 검색 도메인 및 검색 도메인 세트를 계속 추가합니다.

- 296 페이지 “검색 도메인 세트를 만드는 방법”
- 297 페이지 “검색 도메인을 만드는 방법”
- 297 페이지 “검색 도메인 세트에 검색 도메인을 추가하는 방법”
- 298 페이지 “클라이언트를 검색 도메인에 할당하는 방법”

이 섹션에서는 명령줄 인터페이스를 사용하여 iSNS 서버를 관리하는 다른 절차를 제공합니다. 다음 내용으로 구성되어 있습니다.

- 299 페이지 “검색 도메인 세트의 상태 표시 방법”
- 299 페이지 “검색 도메인의 상태 표시 방법”
- 300 페이지 “클라이언트 상태 표시 방법”
- 300 페이지 “검색 도메인에서 클라이언트를 제거하는 방법”
- 300 페이지 “검색 도메인 세트에서 검색 도메인을 제거하는 방법”
- 301 페이지 “검색 도메인 세트를 사용 안함으로 설정하는 방법”
- 301 페이지 “검색 도메인 세트 제거 방법”

### ▼ 검색 도메인 세트의 상태 표시 방법

- 검색 도메인 세트의 상태를 표시하고 해당 구성원인 검색 도메인을 나열합니다.

```
# isnsadm list-dd-set -v set_name
```

### ▼ 검색 도메인의 상태 표시 방법

- 검색 도메인의 상태를 표시하고 해당 구성원인 클라이언트를 나열합니다.

```
# isnsadm list-dd -v domain_name
```

## ▼ 클라이언트 상태 표시 방법

- 다음 중 하나를 선택하여 클라이언트 상태를 표시합니다.

- 모든 클라이언트의 상태 표시:

```
# isnsadm list-node -v
```

- 대상, 즉 저장소 객체인 클라이언트의 상태만 표시:

```
# isnsadm list-node -t
```

## ▼ 검색 도메인에서 클라이언트를 제거하는 방법

- 1 “iSNS 서버 관리” RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “RBAC 초기 구성\(작업 맵\)”](#)을 참조하십시오.

- 2 클라이언트를 나열하여 제거할 클라이언트를 식별합니다.

```
# isnsadm list-node -v
iSCSI Name: iqn.1986-03.com.sun:01:000e0c9f10da.45173FEA.engr
Alias: STK5320_NAS
Type: Target
Network Entity: SE5310
Portal: 172.20.57.95:3260
  Portal Group: 1
Portal: 172.20.56.95:3260
  Portal Group: 1
DD Name: Research,Finance
```

출력에 클라이언트의 iSCSI 이름과 이 클라이언트가 구성원으로 속한 검색 도메인 이름이 표시됩니다.

- 3 검색 도메인에서 클라이언트를 제거합니다.

```
# isnsadm remove-node -d domain_name iSCSI_name
```

## ▼ 검색 도메인 세트에서 검색 도메인을 제거하는 방법

- 1 “iSNS 서버 관리” RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “RBAC 초기 구성\(작업 맵\)”](#)을 참조하십시오.

- 2 검색 도메인을 나열하여 제거할 검색 도메인을 식별합니다.

```
# isnsadm list-dd -v
```

- 3 검색 도메인 세트에서 검색 도메인을 제거합니다.

```
# isnsadm remove-dd set_name domain_name
```

## ▼ 검색 도메인 세트를 사용 안함으로 설정하는 방법

- 1 “iSNS 서버 관리” RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “RBAC 초기 구성\(작업 맵\)”](#)을 참조하십시오.

- 2 검색 도메인 세트를 비활성화합니다.

```
# isnsadm disable-dd-set set_name
```

- 3 검색 도메인 세트의 상태가 Disabled(사용 안함)로 변경되었는지 확인합니다.

```
# isnsadm list-dd-set set_name
```

## ▼ 검색 도메인 세트 제거 방법

검색 도메인 세트를 제거해도 해당 검색 도메인은 유지됩니다. 검색 도메인은 적어도 한 개 검색 도메인 세트의 구성원이어야 합니다.

- 1 “iSNS 서버 관리” RBAC 프로파일을 사용하여 iSNS 서비스를 관리하는 데 필요한 권한을 얻습니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [Oracle Solaris 관리: 보안 서비스의 “RBAC 초기 구성\(작업 맵\)”](#)을 참조하십시오.

- 2 검색 도메인 세트를 나열하여 제거할 검색 도메인을 식별합니다.

```
# isnsadm list-dd-set -v
```

- 3 검색 도메인 세트를 제거합니다.

```
# isnsadm remove-dd-set set_name
```



## format 유틸리티(참조)

---

이 장에서는 `format` 유틸리티의 메뉴와 명령에 대해 설명합니다.

이 장에서 다루는 참조 정보는 다음과 같습니다.

- 303 페이지 “`format` 유틸리티 사용을 위한 권장 사항 및 요구 사항”
- 304 페이지 “`format` 메뉴 및 명령 설명”
- 310 페이지 “`format` 명령에 대한 입력 규칙”
- 311 페이지 “`format` 유틸리티에 대한 도움말 보기”

`format` 유틸리티 사용 시기의 개요는 191 페이지 “`format` 유틸리티”를 참조하십시오.

### format 유틸리티 사용을 위한 권장 사항 및 요구 사항

`format` 유틸리티를 사용하려면 관리자로 전환하거나 동등한 역할을 맡아야 합니다. 그렇지 않으면 `format` 유틸리티를 사용할 때 다음 오류 메시지가 표시됩니다.

```
$ format
Searching for disks...done
No permission (or no disks found)!
```

`format` 유틸리티를 사용하고 기존 데이터를 보존하려는 경우 다음 지침을 준수합니다.

- 디스크 드라이브의 모든 파일을 백업합니다.
- `format` 유틸리티의 `dump` 명령을 사용하여 모든 결함 목록을 파일에 저장합니다. 파일 이름에는 드라이브 유형, 모델 번호 및 일련 번호가 포함되어야 합니다.
- 드라이브와 함께 제공된 제조업체 결함 목록의 인쇄본을 보관합니다.

## format 메뉴 및 명령 설명

format 기본 메뉴는 다음과 유사하게 표시됩니다.

```

FORMAT MENU:
  disk      - select a disk
  type      - select (define) a disk type
  partition - select (define) a partition table
  current   - describe the current disk
  format    - format and analyze the disk
  fdisk     - run the fdisk program (x86 only)
  repair    - repair a defective sector
  label     - write label to the disk
  analyze   - surface analysis
  defect    - defect list management
  backup    - search for backup labels
  verify    - read and display labels
  save      - save new disk/partition definitions
  inquiry   - show vendor, product and revision
  volname   - set 8-character volume name
  !<cmd>    - execute <cmd>, then return
  quit
format>

```

다음 표에서는 format 유틸리티의 기본 메뉴 항목에 대해 설명합니다.

표 16-1 format 유틸리티의 기본 메뉴 항목 설명

메뉴 항목	명령 또는 메뉴?	설명
disk	명령	시스템의 드라이브를 모두 나열합니다. 후속 작업에서 사용할 디스크를 선택할 수도 있습니다. 이 디스크를 현재 디스크라고 합니다.
type	명령	현재 디스크의 제조업체 및 모델을 식별합니다. 알려진 드라이브 유형 목록도 표시합니다. 모든 SCSI-2 디스크 드라이브에 대해 <b>Auto configure</b> 옵션을 선택합니다.
partition	메뉴	슬라이스를 만들고 수정합니다. 자세한 내용은 <a href="#">306 페이지 “partition 메뉴”</a> 를 참조하십시오.
current	명령	현재 디스크에 대해 다음 정보를 표시합니다. <ul style="list-style-type: none"> <li>■ 장치 이름 및 장치 유형</li> <li>■ 실린더, 대체 실린더, 헤드 및 섹터 수</li> <li>■ 물리적 장치 이름</li> </ul>



표 16-1 format 유틸리티의 기본 메뉴 항목 설명 (계속)

메뉴 항목	명령 또는 메뉴?	설명
format	명령	다음 정보 소스 중 하나를 이 순서대로 사용하여 현재 디스크를 포맷합니다. <ol style="list-style-type: none"> <li>1. <code>format.dat</code> 파일에 있는 정보</li> <li>2. 자동 구성 프로세스의 정보</li> <li>3. <code>format.dat</code> 항목이 없는 경우 프롬프트에서 입력한 정보</li> </ol> <p>IDE 디스크에는 이 명령이 적용되지 않습니다. IDE 디스크는 제조업체가 미리 포맷합니다.</p>
fdisk	메뉴	x86 플랫폼에만 해당: <code>fdisk</code> 프로그램을 실행하여 Solaris <code>fdisk</code> 분할 영역을 만듭니다. <p>1TB 크기보다 큰 EFI 레이블 디스크에서는 <code>fdisk</code> 명령을 사용할 수 없습니다.</p>
repair	명령	현재 디스크에서 특정 블록을 복구합니다.
label	명령	현재 디스크에 새 레이블을 씁니다.
analyze	메뉴	읽기, 쓰기 및 비교 테스트를 실행합니다. 자세한 내용은 <a href="#">307 페이지 “analyze 메뉴”</a> 를 참조하십시오.
defect	메뉴	결함 목록을 검색하고 표시합니다. 자세한 내용은 <a href="#">309 페이지 “defect 메뉴”</a> 를 참조하십시오. IDE 디스크에는 이 기능이 적용되지 않습니다. IDE 디스크가 자동으로 결함을 관리합니다.
backup	명령	<b>VTOC</b> - 백업 레이블을 검색합니다. <b>EFI</b> - 지원되지 않습니다.
verify	명령	현재 디스크에 대해 다음 정보를 표시합니다. <ul style="list-style-type: none"> <li>■ 장치 이름 및 장치 유형</li> <li>■ 실린더, 대체 실린더, 헤드 및 섹터 수</li> <li>■ 분할 영역 테이블</li> </ul>
save	명령	<b>VTOC</b> - 새 디스크 및 분할 영역 정보를 저장합니다. <b>EFI</b> - 적용할 수 없습니다.
inquiry	명령	<b>SCSI 디스크에만 해당</b> - 현재 드라이브의 공급업체, 제품 이름 및 개정 레벨을 표시합니다.
volname	명령	새로 지정한 8자 볼륨 이름을 사용하여 디스크에 레이블을 지정합니다.
quit	명령	<code>format</code> 메뉴를 종료합니다.

## partition 메뉴

partition 메뉴는 다음과 유사하게 표시됩니다.

```
format> partition
PARTITION MENU:
  0      - change '0' partition
  1      - change '1' partition
  2      - change '2' partition
  3      - change '3' partition
  4      - change '4' partition
  5      - change '5' partition
  6      - change '6' partition
  7      - change '7' partition
select  - select a predefined table
modify  - modify a predefined partition table
name    - name the current table
print   - display the current table
label   - write partition map and label to the disk
quit
```

partition>

다음 표에서는 partition 메뉴 항목에 대해 설명합니다.

표 16-2 partition 메뉴 항목 설명

하위 명령	설명
change 'n' partition	새 분할 영역에 대한 다음 정보를 지정할 수 있습니다. <ul style="list-style-type: none"> <li>■ ID 태그</li> <li>■ 권한 플래그</li> <li>■ 시작 실린더</li> <li>■ 크기</li> </ul>
select	미리 정의된 분할 영역 테이블을 선택할 수 있습니다.
modify	분할 영역 테이블의 모든 슬라이스를 변경할 수 있습니다. 개별 change 'x' partition 명령보다 이 명령이 선호됩니다.
name	현재 분할 영역 테이블의 이름을 지정할 수 있습니다.
print	현재 분할 영역 테이블을 표시합니다.
label	현재 디스크에 분할 영역 맵과 레이블을 씁니다.
quit	partition 메뉴를 종료합니다.

## x86: fdisk 메뉴

fdisk 메뉴는 x86 기반 시스템에만 나타나며 다음과 유사하게 표시됩니다.

```
format> fdisk
Total disk size is 14169 cylinders
Cylinder size is 2510 (512 byte) blocks

      Cylinders
Partition  Status  Type      Start  End    Length  %
=====  =====  =====  =====  =====  =====  =====
      1      Active  x86 Boot   1      9      9        0
      2              Solaris2  10    14168  14159   100
```

SELECT ONE OF THE FOLLOWING:

1. Create a partition
  2. Specify the active partition
  3. Delete a partition
  4. Change between Solaris and Solaris2 Partition IDs
  5. Exit (update disk configuration and exit)
  6. Cancel (exit without updating disk configuration)
- Enter Selection:

다음 표에서는 fdisk 메뉴 항목에 대해 설명합니다.

표 16-3 x86: fdisk 메뉴 항목 설명

메뉴 항목	설명
Create a partition	fdisk 분할 영역을 만듭니다. Oracle Solaris 또는 DOS와 같은 각 OS에 대해 별도의 분할 영역을 만들어야 합니다. 디스크당 최대 4개 분할 영역이 있습니다. 디스크의 비율로 fdisk 분할 영역의 크기를 묻는 메시지가 표시됩니다.
Specify the active partition	부트에 사용할 분할 영역을 지정할 수 있습니다. 이 메뉴 항목은 첫번째 단계 부트 프로그램이 두번째 단계 부트 프로그램을 찾는 위치를 식별합니다.
Delete a partition	이전에 만든 분할 영역을 삭제합니다. 이 명령은 분할 영역의 데이터를 모두 삭제합니다.
Change between Solaris and Solaris2 Partition IDs	분할 영역 ID를 130(0x82)에서 191(0xbf)로 변경하고 되돌립니다.
Exit (update disk configuration and exit)	새 버전의 분할 영역 테이블을 작성하고 fdisk 메뉴를 종료합니다.
Cancel (exit without updating disk configuration)	분할 영역 테이블을 수정하지 않고 fdisk 메뉴를 종료합니다.

## analyze 메뉴

analyze 메뉴는 다음과 유사하게 표시됩니다.

```
format> analyze
```

ANALYZE MENU:

```

read      - read only test   (doesn't harm SunOS)
refresh   - read then write  (doesn't harm data)
test      - pattern testing  (doesn't harm data)
write     - write then read   (corrupts data)
compare   - write, read, compare (corrupts data)
purge     - write, read, write (corrupts data)
verify    - write entire disk, then verify (corrupts data)
print     - display data buffer
setup     - set analysis parameters
config    - show analysis parameters
quit
analyze>

```

다음 표에서는 analyze 메뉴 항목에 대해 설명합니다.

표 16-4 analyze 메뉴 항목 설명

하위 명령	설명
read	현재 디스크에서 각 섹터를 읽습니다. 결함이 있는 블록을 기본값으로 복구합니다.
refresh	데이터를 손상시키지 않고 현재 디스크에서 데이터를 읽은 후 씁니다. 결함이 있는 블록을 기본값으로 복구합니다.
test	데이터를 손상시키지 않고 디스크에 패턴 세트를 씁니다. 결함이 있는 블록을 기본값으로 복구합니다.
write	디스크에 패턴 세트를 쓴 후 디스크의 데이터를 다시 읽습니다. 디스크의 기존 데이터를 삭제합니다. 결함이 있는 블록을 기본값으로 복구합니다.
compare	디스크에 패턴 세트를 쓰고, 데이터를 다시 읽은 후 쓰기 버퍼의 데이터와 비교합니다. 디스크의 기존 데이터를 삭제합니다. 결함이 있는 블록을 기본값으로 복구합니다.
purge	어떤 방법으로든 데이터를 검색할 수 없도록 디스크에서 모든 데이터를 제거합니다. 전체 디스크 또는 디스크의 한 섹션에 고유한 패턴 3개를 쓰면 데이터가 제거됩니다. 확인에 성공하면 전체 디스크 또는 디스크의 한 섹션에 16진수 비트 패턴이 기록됩니다.  결함이 있는 블록을 기본값으로 복구합니다.
verify	첫번째 단계에서 전체 디스크의 각 블록에 고유한 데이터를 씁니다. 다음 단계에서 데이터를 읽고 확인합니다. 디스크의 기존 데이터를 삭제합니다. 결함이 있는 블록을 기본값으로 복구합니다.
print	읽기/쓰기 버퍼의 데이터를 표시합니다.

표 16-4 analyze 메뉴 항목 설명 (계속)

하위 명령	설명
setup	다음과 같은 분석 매개변수를 지정할 수 있습니다.  Analyze entire disk? yes Starting block number: <i>depends on drive</i> Ending block number: <i>depends on drive</i> Loop continuously? no Number of passes: 2 Repair defective blocks? yes Stop after first error? no Use random bit patterns? no Number of blocks per transfer: 126 (0/n/nn) Verify media after formatting? yes Enable extended messages? no Restore defect list? yes Restore disk label? yes
config	현재 분석 매개변수를 표시합니다.
quit	analyze 메뉴를 종료합니다.

## defect 메뉴

defect 메뉴는 다음과 유사하게 표시됩니다.

```
format> defect
DEFECT MENU:
  primary - extract manufacturer's defect list
  grown   - extract manufacturer's and repaired defects lists
  both    - extract both primary and grown defects lists
  print   - display working list
  dump    - dump working list to file
  quit
```

defect>

다음 표에서는 defect 메뉴 항목에 대해 설명합니다.

표 16-5 defect 메뉴 항목 설명

하위 명령	설명
primary	디스크 드라이브에서 제조업체의 결함 목록을 읽고 메모리 내 결함 목록을 업데이트합니다.
grown	발생한 결함 목록을 읽고 메모리 내 결함 목록을 업데이트합니다. 발생한 결함은 분석 도중 감지된 결함입니다.
both	제조업체의 결함 목록과 발생한 결함 목록을 모두 읽습니다. 그런 다음 메모리 내 결함 목록을 업데이트합니다.

표 16-5 defect 메뉴 항목 설명 (계속)

하위 명령	설명
print	메모리 내 결함 목록을 표시합니다.
dump	메모리 내 결함 목록을 파일에 저장합니다.
quit	defect 메뉴를 종료합니다.

## format 명령에 대한 입력 규칙

format 유틸리티를 사용하는 경우 여러 종류의 정보를 제공해야 합니다. 이 섹션에서는 이 정보에 대한 규칙을 설명합니다. 데이터를 지정할 때 format의 도움말 기능을 사용하는 방법에 대한 자세한 내용은 311 페이지 “format 유틸리티에 대한 도움말 보기”를 참조하십시오.

### format 명령에 숫자 지정

format 유틸리티의 여러 위치에 숫자를 입력해야 합니다. 적절한 데이터를 지정하거나 선택 항목 목록에서 숫자를 선택해야 합니다. 두 경우 모두 도움말 기능을 통해 format에 필요한 숫자의 상한과 하한이 표시됩니다. 적절한 숫자를 입력합니다. 밑이 숫자의 일부로 명시적으로 지정되지 않은 경우(예: 16진수의 0x) 숫자는 10진수 형식으로 가정됩니다.

다음은 정수 입력의 예입니다.

```
Enter number of passes [2]: 34
Enter number of passes [34] 0xf
```

### format 명령 이름 지정

format 유틸리티가 메뉴 프롬프트를 표시할 때마다 명령 이름을 입력해야 합니다. 입력한 내용이 원하는 명령을 고유하게 식별하는 경우 명령 이름을 축약할 수 있습니다.

예를 들어, p를 사용하여 format 메뉴의 partition 메뉴에 액세스합니다. 그런 다음 p를 입력하여 현재 슬라이스 테이블을 표시합니다.

```
format> p
PARTITION MENU:
  0   - change '0' partition
  1   - change '1' partition
  2   - change '2' partition
  3   - change '3' partition
  4   - change '4' partition
  5   - change '5' partition
```

```

6      - change '6' partition
7      - change '7' partition
select - select a predefined table
modify - modify a predefined partition table
name   - name the current table
print  - display the current table
label  - write partition map and label to the disk
quit
partition> p

```

## format 명령에 디스크 이름 지정

format 유틸리티의 특정 지점에서는 특정 항목의 이름을 지정해야 합니다. 이 경우 원하는 임의 문자열을 이름에 지정할 수 있습니다. 이름에 공백이 있는 경우 전체 이름을 큰따옴표 표시(")로 묶어야 합니다. 그렇지 않으면 이름의 첫 단어만 사용됩니다.

예를 들어, 디스크의 특정 분할 영역 테이블을 식별하려는 경우 partition 메뉴에서 사용할 가능한 name 하위 명령을 사용할 수 있습니다.

```

partition> name
Enter table name (remember quotes): "new disk3"

```

## format 유틸리티에 대한 도움말 보기

format 유틸리티는 format 유틸리티에 입력이 필요할 때마다 사용할 수 있는 도움말 기능을 제공합니다. 물음표(?)를 입력하여 필요한 입력에 대한 도움말을 요청할 수 있습니다. format 유틸리티가 필요한 입력 유형에 대한 간단한 설명을 표시합니다.

메뉴 프롬프트에서 ?를 입력하면 사용 가능한 명령 목록이 표시됩니다.

format 유틸리티와 연관된 매뉴얼 페이지는 다음과 같습니다.

- `format(1M)` - 기본 format 유틸리티 기능 및 모든 명령줄 변수에 대해 설명합니다.
- `format.dat(4)` - format 유틸리티에 대한 디스크 드라이브 구성 정보를 설명합니다.





## 파일 시스템 관리(개요)

---

파일 시스템 관리는 가장 중요한 시스템 관리 작업 중 하나입니다.

이 장에서 다루는 개요 정보는 다음과 같습니다.

- 313 페이지 “Oracle Solaris 파일 시스템의 새로운 기능”
- 314 페이지 “파일 시스템 관리 작업을 찾을 위치”
- 315 페이지 “파일 시스템 개요”
- 315 페이지 “Oracle Solaris 파일 시스템 유형”
- 320 페이지 “기본 Oracle Solaris 파일 시스템”
- 320 페이지 “스왑 공간”
- 321 페이지 “파일 시스템 마운트 및 마운트 해제에 대한 개요”
- 325 페이지 “파일 시스템 유형 확인”

### Oracle Solaris 파일 시스템의 새로운 기능

이 단원에서는 Oracle Solaris 릴리스의 새로운 파일 시스템 기능에 대해 설명합니다.

- 313 페이지 “파일 시스템 모니터링 도구(fsstat)”
- 314 페이지 “Oracle Solaris ZFS 파일 시스템”

### 파일 시스템 모니터링 도구(fsstat)

**Oracle Solaris 11:** 파일 시스템 작업을 보고하기 위해 새로운 파일 시스템 모니터링 도구인 fsstat가 제공됩니다. 마운트 지점 기준 또는 파일 시스템 유형 기준과 같은 여러 옵션을 사용하여 작업을 보고할 수 있습니다.

예를 들어 다음 fsstat 명령은 ZFS 모듈 로드 후 모든 ZFS 파일 시스템 작업을 표시합니다.

```
$ fsstat zfs
new name name attr attr lookup rddir read read write write
```

```
file remov chng get set ops ops ops bytes ops bytes
268K 145K 93.6K 28.0M 71.1K 186M 2.74M 12.9M 56.2G 1.61M 9.46G zfs
```

예를 들어 다음 `fsstat` 명령은 `/export/ws` 파일 시스템이 마운트된 후 모든 파일 시스템 작업을 표시합니다.

```
$ fsstat /export/ws
new name name attr attr lookup rddir read read write write
file remov chng get set ops ops ops bytes ops bytes
0 0 0 18.1K 0 12.6M 52 0 0 0 0 /export/ws
```

기본 양식은 GB, KB 및 MB와 같은 값을 이해하기 쉽도록 통계 정보를 보고하는 것입니다.

자세한 내용은 `fsstat(1M)`를 참조하십시오.

## Oracle Solaris ZFS 파일 시스템

**Oracle Solaris 11:** 혁신적인 새로운 파일 시스템인 Oracle Solaris ZFS는 간단한 관리 기능, 트랜잭션 의미, 중단간 데이터 무결성 및 놀라운 확장성을 제공합니다. 또한 ZFS는 다음과 같은 관리 기능을 제공합니다.

- 장치 관리 지원
- 지속적 스냅샷 및 복제 기능
- 파일 시스템에 대해 설정 가능한 할당량
- ACL 기반 액세스 제어
- 파일 시스템에 대한 저장소 풀 공간 예약
- 영역이 설치된 Oracle Solaris 시스템 지원

ZFS 사용에 대한 자세한 내용은 [Oracle Solaris 관리: ZFS 파일 시스템](#)을 참조하십시오.

## 파일 시스템 관리 작업을 찾을 위치

파일 시스템 관리를 위한 단계별 지침을 찾으려면 다음 참조를 사용합니다.

파일 시스템 관리 작업	자세한 정보
새 디스크 장치 연결 및 구성	10 장, “디스크 관리(개요)”
새 파일 시스템 만들기 및 마운트	18 장, “파일 시스템 만들기 및 마운트(작업)”
원격 파일을 사용자가 사용할 수 있도록 설정	<a href="#">Oracle Solaris 관리: 네트워크 서비스의 5 장, “네트워크 파일 시스템 관리(작업)”</a>

## 파일 시스템 개요

파일 시스템은 파일을 구성 및 저장하는 데 사용되는 디렉토리 구조입니다. *file system*이라는 용어는 다음을 설명하는 데 사용됩니다.

- 특정 유형의 파일 시스템: 디스크 기반, 네트워크 기반 또는 가상
- 루트(/) 디렉토리로 시작하는 전체 파일 트리
- 디스크 슬라이스 또는 다른 매체 저장 장치의 데이터 구조
- 파일에 액세스할 수 있도록 기본 파일 트리에 있는 마운트 지점에 첨부된 파일 트리 구조의 일부

일반적으로 컨텍스트를 통해 의도된 의미를 파악할 수 있습니다.

Oracle Solaris OS는 다른 파일 시스템 유형에 대한 표준 인터페이스를 제공하는 VFS(가상 파일 시스템)를 사용합니다. VFS 아키텍처를 통해 커널에서 파일 읽기, 쓰기 및 나열과 같은 기본 작업을 처리할 수 있습니다. VFS 아키텍처를 통해 새 파일 시스템을 쉽게 추가할 수 있습니다.

## Oracle Solaris 파일 시스템 유형

Oracle Solaris OS는 다음과 같은 3가지 유형의 파일 시스템을 지원합니다.

- 디스크 기반
- 네트워크 기반
- 가상

파일 시스템 유형을 식별하려면 325 페이지 “파일 시스템 유형 확인”을 참조하십시오.

## Oracle Solaris 디스크 기반 파일 시스템

디스크 기반 파일 시스템은 하드 디스크 및 DVD와 같은 물리적 매체에 저장됩니다. 디스크 기반 파일 시스템은 다른 형식으로 쓸 수 있습니다. 사용 가능한 형식은 다음 표에서 설명합니다.

디스크 기반 파일 시스템	형식 설명
ZFS	ZFS는 Oracle Solaris 11 릴리스의 기본 디스크 기반 루트 파일 시스템입니다. 자세한 내용은 <a href="#">Oracle Solaris 관리: ZFS 파일 시스템</a> 을 참조하십시오.
UFS	4.3 Tahoe 릴리스에 제공된 BSD Fat Fast 파일 시스템을 기반으로 하는 레거시 UNIX 파일 시스템입니다.

디스크 기반 파일 시스템	형식 설명
HSFS	High Sierra, Rock Ridge 및 ISO 9660 파일 시스템입니다. High Sierra는 최초의 CD-ROM 파일 시스템입니다. ISO 9660은 High Sierra 파일 시스템의 공식적인 표준 버전입니다. HSFS 파일 시스템은 CD-ROM에서 사용되는 읽기 전용 파일 시스템입니다. Oracle Solaris HSFS는 Rock Ridge가 ISO 9660으로 확장되도록 지원합니다. CD-ROM에 있으면 이러한 확장은 쓰기 가능성 및 하드 링크를 제외한 모든 파일 시스템 기능 및 파일 유형을 제공합니다.
PCFS	DOS 기반 개인 컴퓨터용으로 작성된 DOS 포맷 디스크에서 데이터 및 프로그램에 대한 읽기 및 쓰기 액세스가 가능한 PC 파일 시스템입니다.
UDFS	UDFS(Universal Disk Format) 파일 시스템. DVD(Digital Versatile Disc 또는 Digital Video Disc)라고 하는 광학 매체 기술에 정보를 저장하기 위한 산업 표준 형식입니다.

각 디스크 기반 파일 시스템 유형은 일반적으로 다음과 같이 특정 매체 장치와 연관되어 있습니다.

- ZFS 또는 UFS - 하드 디스크
- HSFS - CD-ROM
- PCFS - USB 디스켓
- UDF - DVD

그러나 이러한 연관은 제한적이지 않습니다. 예를 들어 DVD에는 해당 DVD에서 만든 ZFS 또는 UFS 파일 시스템이 있을 수 있습니다.

## UDFS(Universal Disk Format) 파일 시스템

이동식 매체에서 UDFS 파일 시스템 만들기에 대한 자세한 내용은 31 페이지 “이동식 매체에서 파일 시스템을 만드는 방법”을 참조하십시오.

UDF 파일 시스템은 DVD(Digital Versatile Disc 또는 Digital Video Disc) 광학 매체에 정보를 저장하기 위한 산업 표준 형식입니다.

UDF 파일 시스템은 SPARC 및 x86 플랫폼 둘 다에서 파일 시스템을 만들고, 마운트하며, 검사하는 데 필요한 시스템 관리 유틸리티와 함께 동적으로 로드 가능한 32비트 및 64비트 모듈로 제공됩니다. Oracle Solaris UDF 파일 시스템은 지원되는 ATAPI 및 SCSI DVD 드라이브, CD-ROM 장치 및 디스크 드라이브와 함께 작동합니다. 또한 Oracle Solaris UDF 파일 시스템은 UDF 1.50 사양과 완전히 호환됩니다.

UDF 파일 시스템은 다음 기능을 제공합니다.

- 산업 표준 CD-ROM 및 DVD-ROM 매체에 UDF 파일 시스템이 포함되어 있는 경우 이러한 매체에 대한 액세스 기능
- 플랫폼 및 운영 체제 전체에서 정보를 교환할 수 있는 유연성

- UDF 형식 기반의 DVD 비디오 사양을 사용하여 방송 품질의 비디오, 고품질 사운드 및 상호 작용성을 갖춘 새로운 응용 프로그램을 구현하기 위한 방식

UDF 파일 시스템에는 다음 기능이 없습니다.

- 순차적 disk-at-once 기록 및 증분 기록과 함께 write-once 매체(CD-RW)에 대한 지원

UDF 파일 시스템에는 다음 항목이 필요합니다.

- 지원되는 SPARC 또는 x86 플랫폼
- 지원되는 CD-ROM 또는 DVD-ROM 장치

Oracle Solaris UDF 파일 시스템을 구현하면 다음 기능이 제공됩니다.

- 산업 표준 읽기/쓰기 UDF 버전 1.50에 대한 지원
- 완전히 국제화된 파일 시스템 유틸리티

## 네트워크 기반 파일 시스템

**네트워크 기반 파일 시스템**은 네트워크에서 액세스할 수 있습니다. 일반적으로 네트워크 기반 파일 시스템은 시스템 하나(주로 서버)에 상주하고 있으며 네트워크를 통해 다른 시스템에서 액세스할 수 있습니다.

NFS 서비스를 통해 서버에서 **리소스**(파일 또는 디렉토리)를 공유한 다음 개별 클라이언트에서 마운트하여 분산된 리소스를 제공할 수 있습니다. 자세한 내용은 [323 페이지 “NFS 환경”](#)을 참조하십시오.

Oracle SMB 서비스를 통해 서버에서 **리소스**(파일 또는 디렉토리)를 공유한 다음 개별 클라이언트에서 마운트하여 Windows 및 Mac OS 시스템에 분산된 리소스를 제공할 수 있습니다. 자세한 내용은 [325 페이지 “Oracle Solaris SMB 서비스”](#)를 참조하십시오.

## 가상 파일 시스템

**가상 파일 시스템**은 특수 커널 정보 및 기능에 대한 액세스를 제공하는 메모리 기반 파일 시스템입니다. 대부분의 가상 파일 시스템은 파일 시스템 디스크 공간을 사용하지 않습니다. TMPFS(임시 파일 시스템)와 같은 일부 가상 파일 시스템에서는 디스크에 있는 스왑 공간을 사용합니다.

## 임시 파일 시스템

TMPFS(임시 파일 시스템)는 파일 시스템 읽기 및 쓰기에 로컬 메모리를 사용합니다. TMPFS를 사용하면 로컬 디스크 또는 네트워크 전체에서 임시 파일을 읽고 쓰는 비용을 절감하여 시스템 성능을 향상할 수 있습니다. 예를 들어 프로그램을 컴파일하는 경우 임시 파일을 만듭니다. 이러한 파일을 조작하는 중 OS에서는 많은 디스크 작업 또는 네트워크 작업을 생성합니다. TMPFS를 사용하여 이러한 임시 파일을 보관하면 임시 파일의 만들기, 조작 및 삭제 속도를 크게 높일 수 있습니다.

TMPFS 파일 시스템의 파일은 영구적이지 않습니다. 파일 시스템이 마운트 해제되는 경우 및 시스템이 종료 또는 재부트되는 경우 이러한 파일이 삭제됩니다.

TMPFS는 Oracle Solaris OS에 있는 /tmp 디렉토리의 기본 파일 시스템입니다. ZFS 또는 UFS 파일 시스템에서 하는 것처럼 /tmp 디렉토리로 파일을 복사하거나 해당 디렉토리 외부로 파일을 이동할 수 있습니다.

TMPFS 파일 시스템은 임시 보조 기억 장치로 스왑 공간을 사용합니다. TMPFS 파일 시스템이 있는 시스템에 적절한 스왑 공간이 없으면 다음과 같은 두 가지 문제가 발생할 수 있습니다.

- 정규 파일 시스템에서처럼 TMPFS 파일 시스템에서 공간이 부족할 수 있습니다.
- 필요한 경우 TMPFS는 스왑 공간을 할당하여 파일 데이터를 저장하므로 스왑 공간이 부족하면 일부 프로그램이 실행되지 않을 수 있습니다.

TMPFS 파일 시스템 만들기에 대한 자세한 내용은 18 장, “파일 시스템 만들기 및 마운트(작업)”를 참조하십시오. 스왑 공간 늘리기에 대한 자세한 내용은 19 장, “추가 스왑 공간 구성(작업)”을 참조하십시오.

## 루프백 파일 시스템

대체 경로 이름을 사용하여 파일에 액세스할 수 있도록 LOFS(루프백 파일 시스템)를 사용하여 새로운 가상 파일 시스템을 만들 수 있습니다. 예를 들어 /tmp/newroot에서 루트(/) 디렉토리의 루프백 마운트를 만들 수 있습니다. 이 루프백 마운트는 NFS 서버에서 마운트된 파일 시스템을 포함하여 전체 파일 시스템 계층을 /tmp/newroot에서 중복된 것처럼 나타나도록 만듭니다. 루트(/)에서 시작하는 경로 이름 또는 /tmp/newroot에서 시작하는 경로 이름을 사용하여 모든 파일에 액세스할 수 있습니다.

LOFS 파일 시스템 만들기에 대한 자세한 내용은 18 장, “파일 시스템 만들기 및 마운트(작업)”를 참조하십시오.

## 프로세스 파일 시스템

PROCFS(프로세스 파일 시스템)는 메모리에 상주하고 /proc 디렉토리에 있는 프로세스 번호 기준 활성 프로세스 목록을 포함하고 있습니다. ps와 같은 명령에서 /proc 디렉토리의 정보를 사용합니다. 또한 디버거 및 기타 개발 도구에서 파일 시스템 호출을 사용하여 프로세스의 주소 공간에 액세스할 수 있습니다.



주의 - /proc 디렉토리의 파일을 삭제하지 마십시오. /proc 디렉토리에서 프로세스를 삭제하더라도 프로세스는 종료되지 않습니다. /proc 파일은 디스크 공간을 사용하지 않으므로 이 디렉토리에서 파일을 삭제할 이유가 없습니다.

/proc 디렉토리는 관리할 필요가 없습니다.

## 추가 가상 파일 시스템

가상 파일 시스템의 다음 추가 유형은 정보 제공용입니다. 이러한 유형은 관리할 필요가 없습니다.

가상 파일 시스템	설명
CTFS	CTFS(계약 파일 시스템)는 계약을 만들고 제어하며 관찰하기 위한 인터페이스입니다. 계약은 더욱 다양한 오류 보고 기능 및 리소스 제거 지연 수단(선택적)을 제공하여 프로세스와 해당 프로세스가 사용 중인 시스템 리소스 간의 관계를 개선합니다.  SMF(서비스 관리 기능)는 프로세스 계약(계약 유형)을 사용하여 서비스를 구성하는 프로세스를 추적합니다. 따라서 다중 프로세스 서비스의 일부에서 발생한 오류가 해당 서비스의 오류로 식별될 수 있습니다.
FIFOFS(선입 선출법)	데이터에 대한 프로세스 공통 액세스를 제공하는 명명된 파이프 파일입니다.
FDFS(파일 설명자)	파일 설명자를 사용하여 열려 있는 파일에 명시적인 이름을 제공합니다.
MNTFS	로컬 시스템에 마운트된 파일 시스템의 테이블에 대한 읽기 전용 액세스를 제공합니다.
NAMEFS	STREAMS에서 최상위 파일의 파일 설명자 동적 마운트에 주로 사용합니다.
OBJFS	OBJFS(객체) 파일 시스템은 커널에서 현재 로드한 모든 모듈 상태를 설명합니다. 이 파일 시스템은 디버거에서 커널에 직접 액세스할 필요 없이 커널 기호에 대한 정보에 액세스하는 데 사용합니다.
SHAREFS	로컬 시스템에 공유 파일 시스템의 테이블에 대한 읽기 전용 액세스를 제공합니다.
SPECFS(특수)	문자 특수 장치 및 블록 장치에 대한 액세스를 제공합니다.
SWAPFS	커널에서 스왑에 사용합니다.

## 확장된 파일 속성

확장된 파일 속성을 포함하도록 ZFS, UFS, NFS 및 TMPFS 파일 시스템이 향상되었습니다. 응용 프로그램 개발자는 확장된 파일 속성을 사용하여 파일에 특정 속성을 연결시킬 수 있습니다. 예를 들어 윈도우와 시스템 관리에 사용되는 응용 프로그램 개발자는 파일과 표시 아이콘을 연결하도록 선택할 수 있습니다. 확장된 파일 속성은 논리적으로는 대상 파일과 연관된 숨겨진 디렉토리 내의 파일로 표시됩니다.

`runat` 명령을 사용하여 확장된 속성 네임스페이스에서 속성을 추가하고 셸 명령을 실행할 수 있습니다. 이러한 네임스페이스는 지정한 파일과 연관된 숨겨진 속성 디렉토리입니다.

`runat` 명령을 사용하여 파일에 속성을 추가하려면 먼저 속성 파일을 만들어야 합니다.

```
$ runat filea cp /tmp/attrdata attr.1
```

그런 다음 `runat` 명령을 사용하여 파일의 속성을 나열합니다.

```
$ runat filea ls -l
```

자세한 내용은 `runat(1)` 매뉴얼 페이지를 참조하십시오.

속성 인식 옵션을 제공하여 파일 시스템 속성을 지원하도록 여러 Oracle Solaris 파일 시스템 명령이 수정되었습니다. 이 옵션을 사용하여 파일 속성을 조회, 복사 또는 찾을 수 있습니다. 자세한 내용은 각 파일 시스템 명령에 대한 특정 매뉴얼 페이지를 참조하십시오.

## 스왑 공간

Oracle Solaris OS에서는 파일 시스템의 디스크 슬라이스가 아닌 임시 저장소의 일부 디스크 슬라이스를 사용합니다. 이러한 슬라이스를 **스왑** 슬라이스 또는 **스왑 공간**이라고 합니다. 현재 프로세스를 처리하는 데 시스템에 충분한 물리적 메모리가 없는 경우, 스왑 공간은 가상 메모리 저장소에 사용됩니다.

여러 응용 프로그램에서 스왑 공간을 사용하므로 필요한 경우 더 많은 스왑 공간을 계획, 모니터 및 추가하는 방법을 알고 있어야 합니다. 스왑 공간에 대한 개요 및 스왑 공간 추가 지침은 19 장, “추가 스왑 공간 구성(작업)”을 참조하십시오.

# 기본 Oracle Solaris 파일 시스템

ZFS 파일 시스템은 계층적으로, 루트 디렉토리(/)에서 시작하여 아래로 여러 디렉토리가 계속 이어집니다. Oracle Solaris 설치 프로세스를 통해 기본 디렉토리 세트를 설치하고 규약 세트를 사용하여 유사한 유형의 파일을 함께 그룹화할 수 있습니다.

Oracle Solaris 파일 시스템 및 디렉토리에 대한 개요는 `filesystem(5)`을 참조하십시오.

다음 표는 기본 Oracle Solaris 파일 시스템에 대해 요약합니다.

표 17-1 기본 Oracle Solaris 파일 시스템

파일 시스템 또는 디렉토리	파일 시스템 유형	설명
루트(/)	ZFS	계층적 파일 트리의 최상위입니다. 루트(/) 디렉토리에는 시스템 작동에 중요한 디렉토리 및 파일이 포함되어 있습니다(예: 커널, 장치 드라이버 및 시스템 부트에 사용되는 프로그램). 또한 루트(/) 디렉토리에는 로컬 및 원격 파일 시스템을 파일 트리에 첨부할 수 있는 마운트 지점 디렉토리가 포함되어 있습니다.



표 17-1 기본 Oracle Solaris 파일 시스템 (계속)

파일 시스템 또는 디렉토리	파일 시스템 유형	설명
/usr	ZFS	다른 사용자와 함께 공유할 수 있는 시스템 파일 및 디렉토리입니다. 특정 유형의 시스템에서만 실행되는 파일은 /usr 파일 시스템 또는 디렉토리에 있습니다(예: SPARC 실행 파일). 모든 유형의 시스템에서 사용할 수 있는 파일(예: 매뉴얼 페이지)은 /usr/share 디렉토리에 있을 수 있습니다.
/export/home 또는 /home	NFS 또는 ZFS	사용자 홈 디렉토리의 마운트 지점으로 사용자 작업 파일을 저장합니다. 기본적으로 /home 디렉토리는 자동으로 마운트된 파일 시스템입니다.
/var	ZFS	로컬 시스템의 수명 중 변경 또는 확장할 수 있는 시스템 파일 및 디렉토리입니다. vi 및 ex 백업 파일과 같은 시스템 로그가 포함되어 있습니다.
/opt	NFS 또는 ZFS	타사 소프트웨어의 선택적 마운트 지점입니다. 일부 시스템에서 /opt 디렉토리는 UFS 파일 시스템 또는 ZFS 파일 시스템일 수 있습니다.
/tmp	TMPFS	시스템 부트 시 또는 /tmp 파일 시스템 마운트 해제 시마다 제거되는 임시 파일입니다.
/proc	PROCFS	프로세스 번호 기준의 활성 프로세스 목록입니다.
/etc/mnttab	MNTFS	로컬 시스템에 대해 마운트된 파일 시스템의 테이블에 대한 읽기 전용 액세스를 제공하는 가상 파일 시스템입니다.
/system/volatile	TMPFS	시스템 부트 후에는 필요 없는 임시 파일을 저장하기 위한 메모리 기반 파일 시스템입니다.
/system/contract	CTFS	계약 정보를 유지 관리하는 가상 파일 시스템입니다.
/system/object	OBJFS	디버거가 커널에 직접 액세스할 필요 없이 커널 심볼에 대한 정보에 액세스하는 데 사용하는 가상 파일 시스템입니다.

## 파일 시스템 마운트 및 마운트 해제에 대한 개요

파일 시스템의 파일에 액세스하려면 파일 시스템을 마운트해야 합니다. 파일 시스템을 마운트하는 경우 해당 파일 시스템을 디렉토리(마운트 지점)에 첨부하여 시스템에서 사용할 수 있도록 만듭니다. 루트(/) 파일 시스템은 항상 마운트됩니다. 다른 모든 파일 시스템은 루트(/) 파일 시스템에서 연결하거나 연결을 해제할 수 있습니다.

시스템 부트 시 대부분의 파일 시스템은 SMF 서비스에서 자동으로 마운트합니다. 일반적으로 파일 시스템을 수동으로 마운트 또는 마운트 해제할 필요가 없습니다. 다른 파일 시스템 유형 마운트에 대한 자세한 내용은 328 페이지 “Oracle Solaris 파일 시스템 마운트 및 마운트 해제”를 참조하십시오.

파일 시스템을 마운트하는 경우 파일 시스템이 마운트된 동안 기본 마운트 지점 디렉토리의 모든 파일 또는 디렉토리는 사용할 수 없습니다. 이러한 파일은 마운트 프로세스의 영향을 영구적으로 받지 않습니다. 이러한 파일은 파일 시스템이 마운트 해제되면 다시 사용할 수 있습니다. 그러나 일반적으로 기존 파일을 숨기려고 하지 않으므로 마운트 디렉토리는 비어 있습니다.

파일 시스템을 마운트하는 방법에 대한 단계적 지침은 [328 페이지 “Oracle Solaris 파일 시스템 마운트 및 마운트 해제”](#)를 참조하십시오.

## 마운트된 파일 시스템 테이블

파일 시스템을 마운트 또는 마운트 해제할 때마다 현재 마운트된 파일 시스템 목록으로 `/etc/mnttab`(마운트 테이블) 파일이 수정됩니다. `cat` 또는 `more` 명령을 사용하여 이 파일의 내용을 표시할 수 있습니다. 그러나 이 파일은 수정할 수 없습니다. 다음은 `/etc/mnttab` 파일의 예입니다.

```
$ more /etc/mnttab
rpool/ROOT/zfsBE      /          zfs      dev=40d0002      0
/devices              /devices  devfs    dev=9500000      1307562777
/dev                  /dev      dev      dev=9580000      1307562777
ctfs                  /system/contract  ctfs     dev=95c0001      1307562777
proc                  /proc     proc     dev=9540000      1307562777
mnttab                /etc/mnttab  mntfs   dev=9600001      1307562777
swap                  /system/volatile  tmpfs   xattr,dev=9640001  1307562777
objfs                 /system/object  objfs   dev=9680001      1307562777
sharefs               /etc/dfs/sharetab  sharefs dev=96c0001      1307562777
fd                    /dev/fd    fd       rw,dev=97c0001  1307562811
swap                  /tmp      tmpfs   xattr,dev=9640002  1307562812
rpool/export          /export   zfs     rw,devices,setuid,nonbmand,exec,rstchown,
xattr,atime,dev=40d0003  1307562818
rpool/export/home    /export/home  zfs     rw,devices,setuid,nonbmand,
exec,rstchown,xattr,atime,dev=40d0004  1307562818
```

## 가상 파일 시스템 테이블

대부분의 파일 시스템은 시스템 부트 시 SMF 서비스에서 자동으로 마운트합니다.

레거시 파일 시스템을 마운트하거나 ZFS 스왑 볼륨을 변경하려면 `/etc/vfstab` 파일을 편집해야 할 수 있습니다. ZFS 스왑 볼륨 변경에 대한 자세한 내용은 [19 장, “추가 스왑 공간 구성\(작업\)”](#)을 참조하십시오.

레거시 파일 시스템을 마운트하는 데 항목을 추가하려면 지정해야 할 정보는 다음과 같습니다.

- 파일 시스템이 상주하고 있는 장치
- 파일 시스템 마운트 지점
- 파일 시스템 유형

- mountall 명령을 사용하여 시스템을 부트하는 경우 파일 시스템을 자동으로 마운트할지 여부
- 모든 마운트 옵션

다음은 시스템에서 ZFS 루트 파일 시스템을 실행하는 경우 `vfstab`의 예입니다.

```
# cat /etc/vfstab
#device          device          mount          FS          fsck          mount          mount
#to mount        to fsck         point          type         pass         at boot      options
#
fd                -              /dev/fd        fd           -            no           -
/proc            -              /proc          proc         -            no           -
/dev/zvol/dsk/rpool/swap -            -              swap         -            no           -
/devices         -              /devices       devfs        -            no           -
sharefs          -              /etc/dfs/sharetabsharefs -            no           -
ctfs             -              /system/contract ctfs         -            no           -
objfs           -              /system/object objfs        -            no           -
swap            -              /tmp           tmpfs        -            yes          -
```

ZFS 파일 시스템은 시스템 부트 시 SMF 서비스에서 자동으로 마운트합니다. 레거시 마운트 기능을 사용하여 `vfstab`에서 ZFS 파일 시스템을 마운트할 수 있습니다. 자세한 내용은 [Oracle Solaris 관리: ZFS 파일 시스템](#)을 참조하십시오.

각 `/etc/vfstab` 필드와 파일 편집 및 사용 방법에 대한 설명은 [vfstab\(4\)](#)를 참조하십시오.

## NFS 환경

NFS는 네트워크의 다른 시스템과 시스템(일반적으로 서버) 하나의 리소스(파일 또는 디렉토리)를 공유하는데 사용할 수 있는 분산된 파일 시스템 서비스입니다. 예를 들어 다른 시스템의 사용자와 타사 응용 프로그램 또는 소스 파일을 공유하려고 할 수 있습니다.

NFS는 리소스의 실제 물리적 위치를 사용자와 관계없이 만듭니다. NFS는 모든 시스템에 공통적으로 사용되는 파일의 복사본을 배치하는 대신 시스템 하나의 디스크에 복사본 하나를 배치하여 다른 모든 시스템이 네트워크에서 해당 복사본에 액세스하도록 허용합니다. NFS에서 원격 파일은 로컬 파일과 실제로 구분할 수 없습니다.

자세한 내용은 [Oracle Solaris 관리: 네트워크 서비스의 4장, “네트워크 파일 시스템 관리\(개요\)”](#)를 참조하십시오.

네트워크에서 공유할 리소스가 있는 시스템이 NFS가 됩니다. 서버는 현재 공유된 리소스 및 액세스 제한(예: 읽기/쓰기 또는 읽기 전용 액세스 권한) 목록을 보관합니다.

리소스를 공유하는 경우 원격 시스템에서 마운트할 수 있도록 해당 리소스를 사용 가능하게 설정합니다.

다음과 같은 방식으로 리소스를 공유할 수 있습니다.

- ZFS 공유 등록 정보를 사용하여 ZFS 공유를 만듭니다.

- share 명령을 사용하여 레거시 공유를 만듭니다.

NFS에 대한 자세한 설명은 **Oracle Solaris 관리: 네트워크 서비스의 4 장, “네트워크 파일 시스템 관리(개요)”**를 참조하십시오.

## NFS 버전 4

Oracle은 Oracle Solaris 릴리스에서 NFS 버전 4 분산 파일 액세스 프로토콜을 구현했습니다.

NFS 버전 4는 파일 액세스, 파일 잠금 및 마운트 프로토콜을 하나의 통합된 프로토콜로 통합하여 방화벽을 통한 흐름을 쉽게 하고 보안을 강화했습니다. Oracle Solaris에서 구현된 NFS 버전 4는 SEAM으로 알려진 Kerberos V5와 완전히 통합되어 인증, 통합 및 프라이버시를 제공합니다. 또한 NFS 버전 4를 사용하여 클라이언트와 서버 사이에서 사용되는 보안 추가 기능을 협상할 수 있습니다. NFS 버전 4를 사용하여 서버는 서로 다른 파일 시스템에 대하여 다양한 보안 추가 기능을 제공할 수 있습니다.

NFS 버전 4 기능에 대한 자세한 내용은 **Oracle Solaris 관리: 네트워크 서비스의 “NFS 서비스의 새로운 기능”**을 참조하십시오.

## 자동 마운트(autofs)

자동 마운트 또는 *autofs*라고 하는 클라이언트측 서비스를 사용하여 NFS 파일 시스템 리소스를 마운트할 수 있습니다. 시스템에서는 *autofs* 서비스를 사용하여 NFS 리소스에 액세스할 때마다 해당 액세스를 자동으로 마운트 및 마운트 해제할 수 있습니다. 디렉토리에 남아 있고 해당 디렉토리내의 파일을 사용 중이면 리소스는 마운트된 상태로 남아 있습니다. 특정 기간에 리소스에 액세스할 수 없는 경우 해당 리소스는 자동으로 마운트 해제됩니다.

*autofs* 서비스는 다음과 같은 기능을 제공합니다.

- NFS 리소스는 시스템 부트 시 마운트할 필요가 없으므로 부트 시간이 줄어듭니다.
- NFS 리소스를 마운트 및 마운트 해제하기 위해 사용자가 루트 암호를 알 필요가 없습니다.
- NFS 리소스는 사용 중인 경우에만 마운트되므로 네트워크 트래픽이 줄어들 수 있습니다.

*autofs* 서비스는 시스템 부트 시 자동으로 실행되는 *automount* 유틸리티에 의해 시작됩니다. *automountd* 데몬은 계속해서 실행되고 필요 시 NFS 파일 시스템의 마운트 및 마운트 해제를 담당합니다. 기본적으로 /home 파일 시스템은 *automount* 데몬에서 마운트합니다.

*autofs*를 사용하면 동일한 파일 시스템을 제공하도록 여러 서버를 지정할 수 있습니다. 이러한 서버 중 하나가 중단된 경우 이러한 방식으로 *autofs*는 다른 시스템에서 파일 시스템을 마운트하려고 합니다.

autofs 설정 및 관리 방법에 대한 자세한 내용은 [Oracle Solaris 관리: 네트워크 서비스의 5 장, “네트워크 파일 시스템 관리\(작업\)”](#)를 참조하십시오.

## Oracle Solaris SMB 서비스

Oracle Solaris OS는 NT LM 0.12 및 CIFS(Common Internet File System)를 비롯하여 여러 SMB 언어에 대한 지원을 포함한 SMB(Server Message Block) 프로토콜 서버 및 클라이언트 구현을 제공합니다. CIFS와 SMB라는 용어는 서로 바꿔 쓸 수 있습니다.

Solaris SMB 서버는 고유의 Oracle Solaris 시스템에서 파일 시스템 공유를 마운트한 SMB 사용 클라이언트에 파일을 SMB *shares*로 제공하도록 허용합니다. Windows 서버와 상호 운영하는 것처럼 Windows, Mac OS 또는 Solaris 클라이언트는 Solaris SMB 서버와 상호 운영할 수 있습니다. Solaris SMB 서버는 작업 그룹 모드 또는 도메인 모드에서 작동할 수 있습니다. 작업 그룹 모드에서 Solaris SMB 서버는 공유된 리소스에 대한 액세스가 요청된 경우 로컬에서 사용자 인증을 담당합니다. 이러한 인증 프로세스를 로컬 로그인이라고 합니다. Solaris SMB 서버는 도메인 모드에서 통과 인증을 사용하므로 사용자 인증이 도메인 제어기로 위임됩니다.

자세한 내용은 [Oracle Solaris Administration: SMB and Windows Interoperability](#)을 확인하십시오.

## 파일 시스템 유형 확인

다음 중 하나를 사용하여 파일 시스템 유형을 확인할 수 있습니다.

- 가상 파일 시스템 테이블(/etc/vfstab 파일)의 FS type 필드
- 로컬 파일 시스템의 /etc/default/fs 파일
- NFS 파일 시스템의 /etc/dfs/fstypes 파일

## 파일 시스템 유형을 확인하는 방법

다음 명령은 파일 시스템의 마운트 여부를 확인합니다.

디스크 슬라이스의 원시 장치 이름이 있는 경우 `fstyp` 명령을 사용하여 파일 시스템 유형을 확인할 수 있습니다(디스크 슬라이스에 파일 시스템이 포함된 경우). 자세한 내용은 [fstyp\(1M\)](#)를 참조하십시오.

예 17-1 파일 시스템 유형 확인

다음 예에서는 `fstyp` 명령을 사용하여 파일 시스템 유형을 확인합니다.

```
# fstyp /dev/rdisk/c0t0d0s0
zfs
```

예 17-1 파일 시스템 유형 확인 (계속)

다음 예에서는 /etc/vfstab 파일을 사용하여 /legacy 파일 시스템의 파일 시스템 유형을 확인합니다.

```
$ grep /legacy /etc/vfstab  
/dev/dsk/c0t3d0s6 /dev/rdisk/c0t3d0s6 /legacy ufs 2 yes -
```

## 파일 시스템 만들기 및 마운트(작업)

---

이 장에서는 ZFS, 임시(TMPFS) 및 루프백(LOFS) 파일 시스템을 만들고 마운트하는 방법에 대해 설명합니다. TMPFS 및 LOFS는 가상 파일 시스템이므로 마운트하여 실제로 "액세스"할 수 있습니다. 또한 이 장에서는 레거시 UFS 파일 시스템 만들기 및 마운트에 대해서도 다룹니다.

이 장에서 다루는 단계별 지침은 다음과 같습니다.

- 332 페이지 “ZFS 파일 시스템을 만드는 방법”
- 333 페이지 “레거시 UFS 파일 시스템을 만들고 마운트하는 방법”
- 334 페이지 “TMPFS 파일 시스템을 만들고 마운트하는 방법”
- 335 페이지 “LOFS 파일 시스템을 만들고 마운트하는 방법”

### Solaris 파일 시스템 만들기

이 단원에서는 Oracle Solaris 파일 시스템 만들기 에 대한 개요를 제공합니다.

#### ZFS 파일 시스템 만들기

ZFS 파일 시스템은 특정 디스크 분할 영역과 직접 연결되어 있지 않습니다. ZFS 파일 시스템은 여러 장치를 포함할 수 있는 ZFS 저장소 풀과 함께 포함되어 있습니다. ZFS 저장소 풀에는 전체 디스크 또는 디스크 슬라이스를 추가할 수 있습니다. 풀 내에서 추가 파일 시스템을 만들 수도 있습니다. 파일 시스템은 동일한 풀에서 여러 데이터 집합을 관리할 수 있도록 관리 지점을 제공합니다.

ZFS 파일 시스템은 `zfs create` 명령을 사용하여 만듭니다. ZFS 파일 시스템은 만들어질 때 자동으로 마운트됩니다. 자세한 내용은 [332 페이지 “ZFS 파일 시스템을 만드는 방법”](#)을 참조하십시오.

ZFS 저장소 풀 및 파일 시스템 만들기 에 대한 자세한 내용은 [Oracle Solaris 관리: ZFS 파일 시스템](#)을 참조하십시오.

## 임시 파일 시스템 만들기

임시 파일 시스템(TMPFS)은 파일 시스템 읽기 및 쓰기에 로컬 메모리를 사용합니다. TMPFS 파일 시스템은 로컬 디스크 또는 네트워크 전체에서 임시 파일을 읽고 쓰는 비용을 절감하여 시스템 성능을 향상시킬 수 있습니다. TMPFS 파일 시스템의 파일은 재부트 또는 마운트 해제 시 삭제됩니다.

여러 TMPFS 파일 시스템을 만드는 경우 해당 파일 시스템은 모두 동일한 시스템 리소스를 사용합니다. `mount` 명령의 `-o size` 옵션을 사용하여 TMPFS 크기를 제한하지 않으면 하나의 TMPFS 파일 시스템에서 만든 파일이 다른 TMPFS 파일 시스템에 사용할 수 있는 공간을 사용합니다.

TMPFS 파일 시스템 만들기 및 마운트에 대한 자세한 내용은 [334 페이지 “TMPFS 파일 시스템을 만들고 마운트하는 방법”](#)을 참조하십시오.

일반적인 정보는 [tmpfs\(7FS\)](#)를 참조하십시오.

## LOFS 파일 시스템 만들기

LOFS 파일 시스템은 기존 파일 시스템에 대한 대체 경로를 제공하는 가상 파일 시스템입니다. LOFS 파일 시스템에 다른 파일 시스템이 마운트된 경우 원래 파일 시스템은 변경되지 않습니다.

LOFS 파일 시스템 만들기 및 마운트에 대한 자세한 내용은 [335 페이지 “LOFS 파일 시스템을 만들고 마운트하는 방법”](#)을 참조하십시오.

일반적인 정보는 [lofs\(7FS\)](#)를 참조하십시오.

---

주 - LOFS 파일 시스템을 만들 때 주의하십시오. LOFS 파일 시스템은 가상 파일 시스템이므로 사용자와 응용 프로그램을 둘 다 혼동할 가능성이 매우 큽니다.

---

## Oracle Solaris 파일 시스템 마운트 및 마운트 해제

ZFS 파일 시스템은 자동으로 마운트 및 마운트 해제됩니다. 레거시 UFS 파일 시스템을 마운트하여 사용 가능하도록 만들 수 있습니다. 그러면 해당 파일 시스템이 지정된 마운트 지점에서 시스템 디렉토리 트리에 첨부됩니다. 루트(/) 파일 시스템은 항상 마운트됩니다.

다음 표는 파일 시스템 사용 방법에 따라 파일 시스템을 마운트하는 지침을 제공합니다.



필요한 마운트 유형	제안된 마운트 방법
자주 마운트할 필요가 없는 로컬 또는 원격 파일 시스템	mount 명령. 명령줄에서 직접 입력합니다.
자주 마운트해야 하는 로컬 레거시 UFS 파일 시스템. 로컬 ZFS 파일 시스템은 SMF 서비스에서 자동으로 마운트합니다.	/etc/vfstab 파일. 시스템이 다중 사용자 상태에서 부트된 경우 파일 시스템을 자동으로 마운트합니다.
자주 마운트해야 하는 원격 레거시 UFS 파일 시스템(예: 홈 디렉토리)	<ul style="list-style-type: none"> <li>■ /etc/vfstab 파일. 시스템이 다중 사용자 상태에서 부트된 경우 파일 시스템을 자동으로 마운트합니다.</li> <li>■ autofs. 파일 시스템에 액세스하는 경우 파일 시스템을 자동으로 마운트하고 다른 디렉토리로 변경하는 경우 파일 시스템을 마운트 해제합니다.</li> </ul>

이동식 매체 마운트에 대한 자세한 내용은 1 장, “이동식 매체 관리(개요)”를 참조하십시오.

mount 명령을 사용하여 이미 마운트된 파일 시스템을 확인할 수 있습니다.

```
$ mount [ -v ]
```

-v는 Verbose 모드에서 마운트된 파일 시스템 목록을 표시합니다.

#### 예 18-1 마운트된 파일 시스템 확인

이 예에서는 mount 명령을 사용하여 현재 마운트된 파일 시스템에 대한 정보를 표시하는 방법을 보여 줍니다.

```
$ mount
/ on rpool/ROOT/zfsBE read/write/setuid/devices/rstchown/dev=40d0002 on Wed ...
/devices on /devices read/write/setuid/devices/rstchown/dev=9500000 on Wed ...
/dev on /dev read/write/setuid/devices/rstchown/dev=9580000 on Wed Jun ...
/system/contract on ctfs read/write/setuid/devices/rstchown/dev=95c0001 ...
/proc on proc read/write/setuid/devices/rstchown/dev=9540000 on Wed Jun ...
/etc/mnttab on mnttab read/write/setuid/devices/rstchown/dev=9600001 on Wed ...
/system/volatile on swap read/write/setuid/devices/rstchown/xattr/dev=9640001 ...
/system/object on objfs read/write/setuid/devices/rstchown/dev=9680001 on Wed ...
/etc/dfs/sharetab on sharefs read/write/setuid/devices/rstchown/dev=96c0001 on ...
/dev/fd on fd read/write/setuid/devices/rstchown/dev=97c0001 on Wed Jun 8 ...
/tmp on swap read/write/setuid/devices/rstchown/xattr/dev=9640002 on Wed Jun ...
/export on rpool/export read/write/setuid/devices/rstchown/nonbmand/exec/xattr/
/export/home on rpool/export/home read/write/setuid/devices/rstchown
/rpool on rpool read/write/setuid/devices/rstchown/nonbmand/exec/
/home/rimmer on pluto:/export/home/rimmer remote/read/write/setuid/xattr/...
```

이 예에서는 zfs mount 명령을 사용하여 현재 마운트된 ZFS 파일 시스템에 대한 정보를 표시하는 방법을 보여 줍니다.

## 예 18-1 마운트된 파일 시스템 확인 (계속)

```
$ zfs mount
rpool/ROOT/solaris          /
rpool/ROOT/solaris/var     /var
rpool                       /rpool
rpool/export                /rpool/export
rpool/export/home          /rpool/export/home
rpool/export/home/admin    /rpool/export/home/admin
```

## /etc/vfstab 파일에 대한 필드 설명

/etc/vfstab 파일의 항목에는 필드가 7개 있습니다. 다음 표에서는 이러한 필드에 대해 설명합니다.

표 18-1 /etc/vfstab 파일에 대한 필드 설명

필드 이름	설명
device to mount	이 필드는 다음 중 하나를 식별합니다. <ul style="list-style-type: none"> <li>■ 로컬 레거시 UFS 파일 시스템의 블록 장치 이름(예: /dev/dsk/c8t1d0s7)</li> <li>■ 원격 파일 시스템의 리소스 이름(예: myserver:/export/home). NFS에 대한 자세한 내용은 <a href="#">Oracle Solaris 관리: IP 서비스</a>를 참조하십시오.</li> <li>■ 스왑할 슬라이스의 블록 장치 이름(예: /dev/dsk/c0t3d0s1)</li> <li>■ 가상 파일 시스템의 디렉토리</li> </ul>
device to fsck	device to mount 필드로 식별되는 레거시 UFS 파일 시스템에 해당하는 원시(문자) 장치 이름입니다(예: /dev/rdisk/c8t1d0s7). 이 필드는 fsck 명령에서 사용한 원시 인터페이스를 확인합니다. 읽기 전용 파일 시스템 또는 원격 파일 시스템과 같이 적용 가능한 장치가 없는 경우 대시(-)를 사용합니다.
mount point	파일 시스템을 마운트할 위치를 식별합니다(예: /usr).
FS type	파일 시스템 유형을 식별합니다.

표 18-1 /etc/vfstab 파일에 대한 필드 설명 (계속)

필드 이름	설명
fsck pass	<p>fsck 명령에서 레거시 UFS 파일 시스템을 검사할지 여부를 결정하는데 사용하는 전달 번호입니다. 필드에 대시(-)가 포함된 경우 해당 파일 시스템은 검사하지 않습니다. 현재 /etc/vfstab 파일의 fsck pass 값은 부트 프로세스 중 무시됩니다.</p> <p>필드에 0이 포함된 경우 레거시 UFS 파일 시스템을 검사하지 않습니다. 필드에 0보다 큰 값이 포함된 경우 해당 UFS 파일 시스템을 항상 검사합니다.</p> <p>이 필드에 값 1이 포함된 모든 레거시 UFS 파일 시스템은 vfstab 파일에 나타나는 순서대로 한 번에 하나씩 검사합니다. fsck pass 값이 1보다 크고 preen 옵션(-op)이 사용되는 여러 UFS 파일 시스템에서 fsck 명령이 실행되는 경우 fsck 명령은 효율성을 극대화하면서 다른 디스크에 있는 파일 시스템을 자동으로 검사합니다. 그렇지 않으면 전달 번호 값이 어떠한 영향도 미치지 않습니다.</p>
mount at boot	<p>시스템 부트 시 mountall 명령이 파일 시스템을 자동으로 마운트할지 여부에 대해 yes 또는 no로 설정합니다. 이 필드는 autofs와는 관계가 없습니다. 루트(/), /usr 및 /var 파일 시스템은 처음에 vfstab 파일에서 마운트되지 않습니다. 이 필드는 이러한 파일 시스템과 가상 파일 시스템(예: /proc 및 /dev/fd)에 대해 항상 no로 설정되어 있어야 합니다.</p>
mount options	<p>파일 시스템을 마운트하는데 사용되는, 공백 없이 쉼표로 구분된 옵션 목록입니다. 옵션이 없음을 나타내려면 대시(-)를 사용합니다. 자세한 내용은 vfstab(4)를 참조하십시오.</p>

주 - /etc/vfstab 파일에 있는 각 필드에는 항목이 있어야 합니다. 필드에 값이 없으면 대시(-)를 지정해야 합니다. 그렇지 않으면 시스템이 제대로 부트되지 않을 수 있습니다. 마찬가지로 공백을 필드 값으로 사용하면 안 됩니다.

## Oracle Solaris 파일 시스템 마운트 해제를 위한 필수 조건

파일 시스템 마운트 해제를 위한 필수 조건은 다음과 같습니다.

- 관리자이거나 상응하는 역할을 가정해야 합니다.
- 사용 중인 파일 시스템은 마운트 해제할 수 없습니다. 사용자가 파일 시스템의 디렉토리에 액세스 중인 경우, 해당 파일 시스템에 파일이 열려 있는 프로그램이 있는 경우 또는 파일 시스템이 공유 중인 경우 파일 시스템은 사용 중인 것으로 간주됩니다. 다음을 수행하여 마운트 해제할 수 있도록 파일 시스템을 설정할 수 있습니다.
  - 다른 파일 시스템에서 디렉토리로 변경합니다.
  - 시스템에서 로그아웃합니다.

- `fuser` 명령을 사용하여 파일 시스템에 액세스 중인 모든 프로세스를 나열한 다음 필요한 경우 중지합니다. 자세한 내용은 340 페이지 “파일 시스템에 액세스 중인 모든 프로세스를 중지하는 방법”을 참조하십시오.  
사용 중인 파일 시스템을 마운트 해제해야 하는 경우 사용자에게 알립니다.
- 파일 시스템의 공유를 해제합니다. 파일 시스템 공유 해제에 대한 자세한 내용은 `unshare(1M)`를 참조하십시오.

파일 시스템 하나 또는 여러 개를 마운트 해제했는지 확인하려면 `mount` 명령의 출력을 확인합니다.

```
$ mount | grep unmounted-file-system
```

## Oracle Solaris 파일 시스템 만들기 및 마운트

이 단원에서는 Oracle Solaris 파일 시스템 만들기 및 마운트에 대한 예를 제공합니다.

### ▼ ZFS 파일 시스템을 만드는 방법

#### 1 관리자로 전환합니다.

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

#### 2 ZFS 저장소 풀을 만듭니다.

다음 예에서는 `tank`라는 이름의 간단한 미러링된 저장소 풀과 `tank`라는 이름의 ZFS 파일 시스템을 하나의 명령으로 만드는 방법을 보여 줍니다. 전체 디스크 `/dev/dsk/c1t0d0` 및 `/dev/dsk/c2t0d0`을 사용할 수 있다고 가정합니다.

```
# zpool create tank mirror c1t0d0 c2t0d0
```

#### 3 ZFS 파일 시스템을 만듭니다.

```
# zfs create tank/fs
```

새 ZFS 파일 시스템인 `tank/fs`는 디스크 공간을 필요한 만큼 사용할 수 있으며 `/tank`에서 자동으로 마운트됩니다.

#### 4 파일 시스템이 만들어 졌는지 확인합니다.

```
# zfs list -r tank
NAME      USED  AVAIL  REFER  MOUNTPOINT
tank      117K  268G   21K    /tank
tank/fs   21K   268G   21K    /tank/fs
```

## ▼ 레거시 UFS 파일 시스템을 만들고 마운트하는 방법

시작하기 전에 다음 필수 조건을 충족하는지 확인합니다.

- 해당 디스크가 포맷되어 있고 슬라이스로 나누어져 있어야 합니다.
- 기존 레거시 UFS 파일 시스템을 다시 만드는 경우 해당 파일 시스템을 마운트 해제합니다.
- 해당 파일 시스템을 포함할 슬라이스의 장치 이름을 알고 있어야 합니다.

디스크 및 디스크 슬라이스 번호 찾기에 대한 자세한 내용은 11 장, “디스크 관리(작업)”를 참조하십시오.

디스크 포맷 및 슬라이스로 디스크 나누기에 대한 자세한 내용은 10 장, “디스크 관리(개요)”를 참조하십시오.

### 1 관리자로 전환합니다.

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

### 2 레거시 UFS 파일 시스템을 만듭니다.

```
# newfs [-N] [-b size] [-i bytes] /dev/rdisk/device-name
```

시스템에서 확인을 요청합니다.



주의 - 이 단계를 수행하기 전에 슬라이스에 올바른 장치 이름을 지정했는지 확인합니다. 잘못된 슬라이스를 지정한 경우 새 파일 시스템이 만들어지면 해당 슬라이스의 내용이 지워집니다. 이러한 오류로 인해 시스템 패닉 상태가 발생할 수 있습니다.

### 3 레거시 UFS 파일 시스템이 만들어졌는지를 확인하려면 새 파일 시스템을 검사합니다.

```
# fsck /dev/rdisk/device-name
```

여기서 *device-name* 인수는 새 파일 시스템이 포함된 디스크 장치의 이름을 지정합니다.

fsck 명령은 새 파일 시스템의 일관성을 검사하고, 모든 문제를 보고하며 문제를 복구하기 전에 알립니다. fsck 명령에 대한 자세한 내용은 fsck(1M)를 참조하십시오.

### 4 레거시 UFS 파일 시스템을 마운트합니다.

```
# mkdir /directory-name
# mount /dev/dsk/device-name /directory-name
```

## 예 18-2 레거시 UFS 파일 시스템 만들기 및 마운트

다음 예에서는 /legacy에서 UFS 파일 시스템 /dev/rdisk/c0t1d0s0을 만들고 마운트하는 방법을 보여 줍니다.

```
# newfs /dev/rdisk/c0t1d0s0
newfs: construct a new file system /dev/rdisk/c0t1d0s0: (y/n)? y
/dev/rdisk/c0t1d0s0: 286722656 sectors in 46668 cylinders of 48 tracks, 128 sectors
140001.3MB in 2917 cyl groups (16 c/g, 48.00MB/g, 5824 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
 32, 98464, 196896, 295328, 393760, 492192, 590624, 689056, 787488, 885920,
Initializing cylinder groups:
.....
super-block backups for last 10 cylinder groups at:
 285773216, 285871648, 285970080, 286068512, 286166944, 286265376, 286363808,
 286462240, 286560672, 286659104
# fsck /dev/rdisk/c0t1d0s0
# mkdir /legacy
# mount /dev/dsk/c0t1d0s0 /legacy
```

### 자세한 정보 레거시 UFS 파일 시스템을 만든 후...

부트 시 레거시 UFS 파일 시스템을 자동으로 마운트하려면 336 페이지 “/etc/vfstab 파일에 항목을 추가하는 방법”으로 이동합니다.

## ▼ TMPFS 파일 시스템을 만들고 마운트하는 방법

### 1 관리자로 전환합니다.

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

### 2 필요한 경우 마운트하려는 디렉토리를 TMPFS 파일 시스템으로 만듭니다.

```
# mkdir /mount-point
```

여기서 *mount-point*는 TMPFS 파일 시스템이 마운트되는 디렉토리입니다.

### 3 TMPFS 파일 시스템을 마운트합니다.

```
# mount -F tmpfs [-o size=number] swap mount-point
```

*-o size=number* TMPFS 파일 시스템의 크기 제한을 MB로 지정합니다.

*mount-point* TMPFS 파일 시스템을 마운트할 디렉토리를 지정합니다.

부트 시 TMPFS 파일 시스템을 자동으로 마운트하도록 시스템을 설정하려면 [예 18-4](#)를 참조하십시오.

### 4 TMPFS 파일 시스템이 만들어졌는지 확인합니다.

```
# mount -v
```

### 예 18-3 TMPFS 파일 시스템 만들기 및 마운트

다음 예에서는 TMPFS 파일 시스템 /export/reports를 만들고 마운트하며, 이 파일 시스템의 크기를 50MB로 제한하는 방법을 보여 줍니다.

```
# mkdir /export/reports
# chmod 777 /export/reports
# mount -F tmpfs -o size=50m swap /export/reports
# mount -v
```

#### 예 18-4 부트 시 TMPFS 파일 시스템 마운트

/etc/vfstab 항목을 추가하여 부트 시 TMPFS 파일 시스템을 자동으로 마운트하도록 시스템을 설정할 수 있습니다. 다음 예는 부트 시 /export/test를 TMPFS 파일 시스템으로 마운트하는 /etc/vfstab 파일의 항목을 보여 줍니다. `size=number` 옵션을 지정하지 않았으므로 /export/test의 TMPFS 파일 시스템 크기는 사용 가능한 시스템 리소스만으로 제한됩니다.

```
swap - /export/test tmpfs - yes -
```

## ▼ LOFS 파일 시스템을 만들고 마운트하는 방법

### 1 관리자로 전환합니다.

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

### 2 필요한 경우 마운트하려는 디렉토리를 LOFS 파일 시스템으로 만듭니다.

```
# mkdir loopback-directory
```

### 3 새로 만든 디렉토리에 적절한 권한 및 소유권을 부여합니다.

### 4 필요한 경우 LOFS 파일 시스템을 마운트하려는 마운트 지점을 만듭니다.

```
# mkdir /mount-point
```

### 5 LOFS 파일 시스템을 마운트합니다.

```
# mount -F lofs loopback-directory /mount-point
```

`loopback-directory` 루프백 마운트 지점에서 마운트하려는 파일 시스템을 지정합니다.

`/mount-point` LOFS 파일 시스템을 마운트할 디렉토리를 지정합니다.

### 6 LOFS 파일 시스템이 마운트되었는지 확인합니다.

```
# mount -v
```

#### 예 18-5 LOFS 파일 시스템 만들기 및 마운트

다음 예에서는 /new/dist 디렉토리에 새 소프트웨어를 만들고 마운트하며, 해당 소프트웨어를 실제로 설치할 필요 없이 루프백 파일 시스템으로 테스트하는 방법을 보여 줍니다.

```
# mkdir /tmp/newroot
# mount -F lofs /new/dist /tmp/newroot
# chroot /tmp/newroot newcommand
```

## 예 18-6 부트 시 LOFS 파일 시스템 마운트

/etc/vfstab 파일 끝에 항목을 추가하여 부트 시 LOFS 파일 시스템을 자동으로 마운트하도록 시스템을 설정할 수 있습니다. 다음 예에서는 /tmp/newroot에서 루트(/) 파일 시스템의 LOFS 파일 시스템을 마운트하는 /etc/vfstab 파일의 항목을 보여 줍니다.

```
/ - /tmp/newroot lofs - yes -
```

루프백 항목이 /etc/vfstab 파일의 마지막 항목인지 확인합니다. 그렇지 않고 루프백 파일 시스템의 /etc/vfstab 항목이 이 파일 시스템에 포함된 파일 시스템보다 우선하는 경우 루프백 파일 시스템을 마운트할 수 없습니다.

## ▼ /etc/vfstab 파일에 항목을 추가하는 방법

일부 ZFS 파일 시스템에 레거시 마운트 동작이 필요하지 않은 경우 부트 시 ZFS 이외의 파일 시스템을 마운트하려면 다음 절차를 수행합니다. ZFS 파일 시스템 마운트에 대한 자세한 내용은 [Oracle Solaris 관리: ZFS 파일 시스템](#)을 참조하십시오.

### 1 관리자로 전환합니다.

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

### 2 필요한 경우 마운트할 파일 시스템의 마운트 지점을 만듭니다.

```
# mkdir /mount-point
```

파일 시스템을 마운트하려면 로컬 시스템에 마운트 지점이 있어야 합니다. **마운트 지점**은 마운트된 파일 시스템이 첨부되는 디렉토리입니다.

### 3 /etc/vfstab 파일을 편집하고 항목을 추가합니다. 다음을 수행합니다.

- a. 공백(공백 또는 탭)으로 각 필드를 구분합니다.
- b. 필드에 내용이 없는 경우 대시(-)를 지정합니다.
- c. 변경 사항을 저장합니다.

---

주 - 부트 프로세스 중 루트(/) 파일 시스템은 커널에 의해 읽기 전용으로 마운트되므로 remount 옵션 및 remount와 함께 사용할 수 있는 옵션만 /etc/vfstab 파일의 루트(/) 항목에 영향을 미칩니다.

---



## 예 18-7 /etc/vfstab 파일에 항목 추가

다음 예에서는 디스크 슬라이스 /dev/dsk/c0t3d0s7을 마운트 지점 /files1에 레거시 UFS 파일 시스템으로 마운트하는 방법을 보여 줍니다. 원시 문자 장치 /dev/rdisk/c0t3d0s7은 device to fsck로 지정됩니다. fsck pass 값 2는 파일 시스템을 검사하지만 순차적으로 검사하지 않음을 나타냅니다.

```
#device          device          mount   FS      fsck  mount  mount
#to mount        to fsck         point   type    pass  at boot options
#
/dev/dsk/c0t3d0s7 /dev/rdisk/c0t3d0s7 /files1 ufs     2     yes    -
```

다음 예에서는 마운트 지점 /usr/man에서 시스템 pluto의 /export/man 디렉토리를 NFS 파일 시스템으로 마운트하는 방법을 보여 줍니다. NFS 파일 시스템이므로 device to fsck 및 fsck pass가 지정되지 않습니다. 이 예에서 mount options의 경우 ro(읽기 전용) 및 soft입니다.

```
#device          device          mount   FS      fsck  mount  mount
#to mount        to fsck         point   type    pass  at boot options
pluto:/export/man -           /usr/man nfs    -     yes    ro,soft
```

## ▼ 파일 시스템(/etc/vfstab 파일)을 마운트하는 방법

### 1 관리자로 전환합니다.

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

### 2 /etc/vfstab 파일에 나열된 파일 시스템을 마운트합니다.

```
# mount /mount-point
```

여기서 */mount-point*는 /etc/vfstab 파일에 있는 mount point 또는 device to mount 필드의 항목을 지정합니다. 일반적으로 마운트 지점은 쉽게 지정할 수 있습니다.

## 예 18-8 파일 시스템(/etc/vfstab 파일) 마운트

다음 예에서는 /etc/vfstab 파일에 나열된 /legacy 파일 시스템을 마운트하는 방법을 보여 줍니다.

```
# mount /legacy
```

## 예 18-9 모든 파일 시스템(/etc/vfstab 파일) 마운트

다음 예에서는 mountall 명령을 사용하고 파일 시스템이 이미 마운트된 경우 표시되는 메시지를 보여 줍니다.

```
# mountall
mount: /tmp is already mounted or swap is busy
```

다음 예에서는 /etc/vfstab 파일에 나열된 모든 로컬 시스템을 마운트하는 방법을 보여 줍니다.

```
# mountall -l
```

다음 예에서는 사용 가능한 모든 ZFS 파일 시스템을 마운트하는 방법을 보여 줍니다.

```
# zfs mount -a
```

다음 예에서는 /etc/vfstab 파일에 나열된 모든 원격 파일 시스템을 마운트하는 방법을 보여 줍니다.

```
# mountall -r
```

## ▼ NFS 파일 시스템을 마운트하는 방법(mount 명령)

### 1 관리자로 전환합니다.

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

### 2 필요한 경우 마운트할 파일 시스템의 마운트 지점을 만듭니다.

```
# mkdir /mount-point
```

로컬 시스템에는 파일 시스템을 마운트할 마운트 지점이 있어야 합니다. 마운트 지점은 마운트된 파일 시스템이 첨부되는 디렉토리입니다.

### 3 해당 리소스(파일 또는 디렉토리)가 서버에서 사용 가능한지 확인합니다.

NFS 파일 시스템을 마운트하려면 share 명령을 사용하여 서버에서 해당 리소스를 사용할 수 있도록 만들어야 합니다. 리소스 공유 방법에 대한 자세한 내용은 [Oracle Solaris 관리: 네트워크 서비스의 “NFS 서비스 정보”](#)를 참조하십시오.

### 4 NFS 파일 시스템을 마운트합니다.

```
# mount -F nfs [-o mount-options] server:/directory /mount-point
```

## 예 18-10 NFS 파일 시스템 마운트(mount 명령)

다음 예에서는 서버 pluto에서 /mnt의 /export/packages 디렉토리를 마운트하는 방법을 보여 줍니다.

```
# mount -F nfs pluto:/export/packages /mnt
```

## ▼ x86: 하드 디스크에서 PCFS(DOS) 파일 시스템을 마운트하는 방법(mount 명령)

하드 디스크에서 PCFS(DOS) 파일 시스템을 마운트하려면 다음 절차를 수행합니다.

### 1 관리자로 전환합니다.

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

### 2 필요한 경우 마운트할 파일 시스템의 마운트 지점을 만듭니다.

```
# mkdir /mount-point
```

로컬 시스템에는 파일 시스템을 마운트할 마운트 지점이 있어야 합니다. 마운트 지점은 마운트된 파일 시스템이 첨부되는 디렉토리입니다.

### 3 PCFS 파일 시스템을 마운트합니다.

```
# mount -F pcfs [-o rw | ro] /dev/dsk/device-name:logical-drive /mount-point
```

`-o rw | ro` PCFS 파일 시스템을 읽기/쓰기(`rw`) 또는 읽기 전용(`ro`)으로 마운트할 수 있도록 지정합니다. 이 옵션을 지정하지 않으면 기본값은 `rw`입니다.

`/dev/dsk/device-name` 전체 디스크의 장치 이름(예: `/dev/dsk/c0t0d0p0`)을 지정합니다.

`logical-drive` DOS 논리 드라이브 문자(`c-z`) 또는 드라이브 번호(`1-24`)를 지정합니다. 드라이브 `c`는 드라이브 `1`과 동일하고 해당 드라이브의 기본 DOS 슬라이스를 나타냅니다. 다른 모든 문자 또는 번호는 확장된 DOS 슬라이스 내의 DOS 논리 드라이브를 나타냅니다.

`/mount-point` 파일 시스템을 마운트할 디렉토리를 지정합니다.

`device-name` 및 `logical-drive`는 콜론으로 구분해야 합니다.

## 예 18-11 x86: 하드 디스크에서 PCFS(DOS) 파일 시스템 마운트(mount 명령)

다음 예에서는 `/pcfs/c` 디렉토리에 있는 기본 DOS 슬라이스에서 논리 드라이브를 마운트하는 방법을 보여 줍니다.

```
# mount -F pcfs /dev/dsk/c0t0d0p0:c /pcfs/c
```

다음 예에서는 `/mnt` 디렉토리에서 확장된 DOS 슬라이스에 있는 첫번째 읽기 전용 논리 드라이브를 마운트하는 방법을 보여 줍니다.

```
# mount -F pcfs -o ro /dev/dsk/c0t0d0p0:2 /mnt
```

## ▼ 파일 시스템에 액세스 중인 모든 프로세스를 중지하는 방법

### 1 관리자로 전환합니다.

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

### 2 파일 시스템에 액세스 중인 모든 프로세스를 나열하므로 중지할 프로세스를 파악할 수 있습니다.

```
# fuser -c [ -u ] /mount-point
```

-c           파일 시스템의 마운트 지점인 파일과 마운트된 파일 시스템 내의 파일을 보고합니다.

-u           각 프로세스 ID의 사용자 로그인 이름을 표시합니다.

/mount-point   프로세스를 중지하려는 파일 시스템의 이름을 지정합니다.

### 3 파일 시스템에 액세스 중인 프로세스를 모두 중지합니다.

```
# fuser -c -k /mount-point
```

해당 파일 시스템을 사용 중인 각 프로세스에 SIGKILL이 전송됩니다.

---

주 - 사용자에게 먼저 경고를 표시한 후에 사용자의 프로세스를 중지할 수 있습니다.

---

### 4 해당 파일 시스템에 액세스 중인 프로세스가 없는지 확인합니다.

```
# fuser -c /mount-point
```

## 예 18-12 파일 시스템에 액세스 중인 모든 프로세스 중지

다음 예는 /export/home 파일 시스템을 사용 중인 프로세스 4006c를 중지하는 방법을 보여 줍니다.

```
# fuser -c /export/home
/export/home:   4006c
# fuser -c -k /export/home
/export/home:   4006c
# fuser -c /export/home
/export/home:
```

## ▼ 파일 시스템을 마운트 해제하는 방법

파일 시스템을 마운트 해제하려면 다음 절차를 수행합니다.

- 1 331 페이지 "Oracle Solaris 파일 시스템 마운트 해제를 위한 필수 조건"에 나열된 필수 조건을 충족하는지 확인합니다.
- 2 파일 시스템을 마운트 해제합니다.

```
# umount /mount-point
```

여기서 */mount-point*는 마운트 해제하려는 파일 시스템의 이름입니다. 이 이름은 다음 중 하나일 수 있습니다.

- 파일 시스템이 마운트되는 디렉토리 이름
- 파일 시스템의 장치 이름 경로
- NFS 파일 시스템에 사용할 수 있는 리소스
- LOFS 파일 시스템의 루프백 디렉토리

### 예 18-13 파일 시스템 마운트 해제

다음 예에서는 레거시 UFS 파일 시스템을 마운트 해제하는 방법을 보여 줍니다.

```
# umount /legacy
```

다음 예에서는 UFS /legacy 파일 시스템을 강제로 마운트 해제하는 방법을 보여 줍니다.

```
# umount -f /legacy
#
```

다음 예에서는 모든 ZFS 파일 시스템을 마운트 해제하는 방법을 보여 줍니다.

```
# zfs umount -a
```

사용 중인 파일 시스템을 제외한 모든 파일 시스템이 마운트 해제됩니다.



## 추가 스왑 공간 구성(작업)

---

이 장에서는 Oracle Solaris OS 설치 후 ZFS 루트 파일 시스템에 필요한 추가 스왑 공간을 구성하기 위한 지침 및 단계별 지침을 제공합니다.

다음은 이 장에서 다루는 주제를 나열한 것입니다.

- 343 페이지 “스왑 공간 정보”
- 346 페이지 “추가 스왑 공간의 필요 여부를 확인하는 방법”
- 347 페이지 “스왑 공간을 할당하는 방법”
- 348 페이지 “스왑 공간 계획”
- 349 페이지 “스왑 리소스 모니터”
- 350 페이지 “Oracle Solaris ZFS 루트 환경에서 스왑 공간 추가 및 변경”

### 스왑 공간 정보

다음 사항을 결정하려면 SunOS 스왑 방식의 기능을 이해하고 있어야 합니다.

- 스왑 공간 요구 사항
- 스왑 공간과 TMPFS 파일 시스템 간의 관계
- 스왑 공간과 관련된 오류 메시지에서 복구하는 방법

### 스왑 공간 및 가상 메모리

Oracle Solaris OS 소프트웨어 및 응용 프로그램 소프트웨어는 파일 시스템이 아니라 임시 저장소의 일부 디스크 공간을 사용할 수 있습니다. 디스크의 예약 영역을 **스왑 공간**이라고 합니다. 스왑 공간은 시스템에서 현재 프로세스를 처리하기에 충분한 물리적 메모리가 부족할 경우 가상 메모리 저장소로 사용됩니다. ZFS 루트 파일 시스템에서 스왑을 위해 예약된 디스크 공간이 ZFS 불륨입니다.

가상 메모리 시스템은 디스크에 있는 파일의 물리적 사본을 메모리의 가상 주소에 매핑합니다. 이러한 매핑에 대한 데이터가 포함된 물리적 메모리 페이지는 파일

시스템의 정규 파일 또는 스왑 공간으로 지원할 수 있습니다. 메모리가 스왑 공간으로 지원되면 메모리를 지원하는 디스크 공간에 할당된 ID가 없기 때문에 **익명 메모리**라고 합니다.

Oracle Solaris OS는 이러한 페이지를 실제로 지원하는 익명 메모리 페이지와 물리적 저장소(또는 디스크로 지원되는 스왑 공간) 간의 계층인 **가상 스왑 공간**이라는 개념을 사용합니다. 시스템의 가상 스왑 공간은 시스템의 모든 물리적(디스크 지원) 스왑 공간과 현재 사용 가능한 물리적 메모리 부분의 합과 동일합니다.

가상 스왑 공간에는 다음과 같은 이점이 있습니다.

- 가상 스왑 공간에는 물리적(디스크) 저장소에 상응하는 공간이 필요하지 않기 때문에 많은 물리적 스왑 공간에 대한 필요성이 줄어듭니다.
- SWAPFS라는 의사 파일 시스템은 익명 메모리 페이지에 주소를 제공합니다. 메모리 페이지의 할당을 제어하기 때문에 SWAPFS는 더욱 유연하게 페이지에 대해 발생하는 활동을 결정할 수 있습니다. 예를 들어 SWAPFS는 디스크 지원 스왑 저장소에 대한 페이지의 요구 사항을 변경할 수 있습니다.

## 스왑 공간 및 TMPFS 파일 시스템

TMPFS 파일 시스템은 Oracle Solaris 환경에서 `/etc/vfstab` 파일의 항목에 의해 자동으로 활성화됩니다. TMPFS 파일 시스템은 디스크가 아닌 메모리(`/tmp` 디렉토리)에 파일 및 파일 관련 정보를 저장하므로 이러한 파일에 더욱 빠르게 액세스할 수 있습니다. 이 기능은 `/tmp`를 많이 사용하는 컴파일러 및 DBMS 제품과 같은 응용 프로그램의 성능을 크게 향상시킵니다.

TMPFS 파일 시스템은 시스템 스왑 리소스의 `/tmp` 디렉토리에서 공간을 할당합니다. 이 기능을 사용하면 `/tmp` 디렉토리의 공간을 모두 사용 시 스왑 공간도 모두 사용하게 됩니다. 따라서 응용 프로그램이 `/tmp` 디렉토리를 많이 사용하는 상태에서 스왑 공간의 사용을 모니터하지 않으면 시스템에서 스왑 공간이 부족할 수 있습니다.

스왑 리소스가 제한되어 있더라도 TMPFS를 사용하려면 다음과 같이 사용하십시오.

- TMPFS가 사용할 수 있는 스왑 리소스의 양을 제어하는 크기 옵션(`-o size`)을 사용하여 TMPFS 파일 시스템을 마운트합니다.
- 컴파일러의 `TMPDIR` 환경 변수를 사용하여 다른 큰 디렉토리를 가리킵니다.  
컴파일러의 `TMPDIR` 변수를 사용하는 경우에만 컴파일러의 `/tmp` 디렉토리 사용 여부를 제어할 수 있습니다. 이 변수는 `/tmp` 디렉토리를 사용하는 다른 프로그램에는 영향을 미치지 않습니다.



## 스왑 공간 및 덤프 장치 구성

덤프 장치는 일반적으로 시스템 충돌 덤프 정보를 저장하기 위해 예약된 디스크 공간입니다. 시스템이 설치되면 ZFS 스왑 볼륨 및 덤프 볼륨이 자동으로 만들어집니다. `dumpadm` 명령을 사용하여 시스템의 덤프 볼륨을 변경할 수 있습니다. 자세한 내용은 **Oracle Solaris 관리: 일반 작업의 17 장, “시스템 충돌 정보 관리(작업)”**를 참조하십시오.

ZFS 루트 환경에서 스왑 및 덤프는 별도의 ZFS 볼륨으로 구성됩니다. 이 모델의 장점은 다음과 같습니다.

- 스왑 및 덤프 영역을 포함하기 위해 디스크를 분할할 필요가 없습니다.
- 스왑 및 덤프 장치에서 기본 ZFS I/O 파이프라인 아키텍처의 이점이 활용됩니다.
- 스왑 및 덤프 장치에서 압축과 같은 특성을 설정할 수 있습니다.
- 스왑 및 덤프 장치의 크기를 재설정할 수 있습니다. 예를 들면 다음과 같습니다.

```
# zfs set volsize=2G rpool/dump
# zfs get volsize rpool/dump
NAME          PROPERTY  VALUE          SOURCE
rpool/dump    volsize   2G             -
```

대용량 덤프 장치를 재할당하는 프로세스의 경우 많은 시간이 걸립니다.

- 활성 시스템에서 보조 스왑 볼륨을 추가하여 전체 스왑 크기를 늘릴 수 있습니다.

ZFS 스왑 및 덤프 장치의 사용에 대한 자세한 내용은 **Oracle Solaris 관리: ZFS 파일 시스템의 “ZFS 스왑 및 덤프 장치 관리”**를 참조하십시오.

## 스왑 공간 및 동적 재구성

좋은 실례로 충분한 스왑 공간을 할당하여 동적으로 재구성하는 동안 오류가 발생한 CPU 및 시스템 보드를 지원하는 것을 들 수 있습니다. 그렇지 않으면 CPU 또는 시스템 보드 오류로 인해 메모리가 부족한 상태에서 호스트 또는 도메인이 재부트될 수 있습니다.

이와 같이 추가 스왑 공간을 사용 가능하도록 설정하지 않으면 메모리 부족으로 인해 한 개 이상의 응용 프로그램이 시작하지 못할 수 있습니다. 이러한 문제 발생 시 스왑 공간을 추가하거나 이러한 응용 프로그램의 메모리 사용을 재구성하려면 수동적인 개입이 필요할 수 있습니다.

재부트 시 메모리의 잠재적인 손실을 처리하도록 추가 스왑 공간을 할당한 경우 모든 핵심 응용 프로그램이 정상적으로 실행될 수 있습니다. 이는 사용자가 시스템을 사용할 수 있고 일부 추가 스왑으로 인해 시스템이 느려질 수 있음을 의미합니다.

자세한 내용은 하드웨어 동적 재구성 설명서를 참조하십시오.

## SAN 환경에서 스왑 공간 구성

SAN 환경과 같은 네트워크 연결 디스크에서 스왑 공간을 구성할 수 있는지 확인하려면 다음 항목을 검토합니다.

- 로컬로 연결된 디스크에서 스왑 공간 문제를 진단하는 것은 네트워크 연결 디스크에서 스왑 공간 문제를 진단하는 것보다 쉽습니다.
- SAN을 통한 스왑 공간 성능은 로컬로 연결된 디스크에서 구성된 스왑 공간 성능에 필적해야 합니다.
- 로컬로 연결된 디스크로 스왑을 이동하는 것보다는 성능 데이터 분석 후에 성능 문제가 있는 시스템에 메모리를 추가하는 것이 SAN를 통한 스왑 성능 문제를 해결하는 데 더 낫습니다.

## 추가 스왑 공간의 필요 여부를 확인하는 방법

시스템에 추가 스왑 공간이 필요한지 확인하려면 `swap -l` 명령을 사용합니다.

예를 들어 다음 `swap -l` 출력은 이 시스템의 스왑 공간이 거의 완전히 사용되었거나 100% 할당되었음을 표시합니다.

```
% swap -l
swap -l
swapfile          dev      swaplo  blocks    free
/dev/zvol/dsk/rpool/swap 182,2      16 67108848 67102992
```

시스템의 스왑 공간이 100% 할당되면 응용 프로그램의 메모리 페이지는 임시 잠금 상태가 됩니다. 그러면 응용 프로그램 오류가 발생하지 않을 수 있지만 시스템 성능은 영향을 받을 수 있습니다.

## 스왑 관련 오류 메시지

다음 메시지는 응용 프로그램이 추가 익명 메모리를 얻으려고 함을 나타냅니다. 그러나 이러한 동작을 지원할 스왑 공간이 남아 있지 않습니다.

```
application is out of memory
```

```
malloc error 0
```

```
messages.1:Sep 21 20:52:11 mars genunix: [ID 470503 kern.warning]
WARNING: Sorry, no swap space to grow stack for pid 100295 (myprog)
```

## TMPFS 관련 오류 메시지

파일을 쓸려고 했을 때 페이지를 할당할 수 없으면 다음 메시지가 표시됩니다. TMPFS가 허용되는 것보다 많이 쓰려고 한 경우 또는 현재 실행 중인 프로그램이 많은 메모리를 사용 중인 경우 이러한 문제가 나타날 수 있습니다.

*directory: File system full, swap space limit exceeded*

다음 메시지는 새 파일 또는 디렉토리를 만들려고 시도하는 중 TMPFS에서 물리적 메모리가 부족함을 의미합니다.

*directory: File system full, memory allocation failed*

TMPFS 관련 오류 메시지에서 복구에 대한 자세한 내용은 [tmpfs\(7FS\)](#)를 참조하십시오.

## 스왑 공간을 할당하는 방법

처음에 스왑 공간은 Oracle Solaris 설치 프로세스의 일부로 할당됩니다. ZFS 루트 환경에서 기본 스왑 크기 범위는 일반적으로 512MB-2GB입니다.

스왑 공간 할당에 대한 일반적인 지침은 [348 페이지 “스왑 공간 계획”](#)을 참조하십시오.

## 스왑 영역 및 /etc/vfstab 파일

시스템이 설치된 후에 스왑 영역 및 스왑 파일은 `/etc/vfstab` 파일에 나열됩니다. 스왑 파일은 시스템 부트 시 `/sbin/swapadd` 스크립트로 활성화됩니다.

`/etc/vfstab` 파일에서 스왑 장치에 해당하는 항목에는 다음과 같은 내용이 포함되어 있습니다.

- ZFS 루트 파일 시스템이 있는 시스템의 스왑 볼륨 경로 이름에 대한 전체 경로 이름
- 스왑 슬라이스 또는 스왑 파일의 파일 시스템 유형

스왑 파일이 포함된 파일 시스템은 스왑 파일이 활성화되기 전에 마운트되어야 합니다. 따라서 `/etc/vfstab` 파일에서 스왑 파일을 활성화하는 항목 앞에 파일 시스템을 마운트하는 항목이 왔는지 확인합니다.

## 스왑 공간 계획

스왑 공간 크기를 결정하는 가장 중요한 요소는 시스템의 소프트웨어 응용 프로그램에 대한 요구 사항입니다. 예를 들어 컴퓨터 지원 설계 시뮬레이터, 데이터베이스 관리 제품, 트랜잭션 모니터 및 지질 분석 시스템과 같은 대용량 응용 프로그램은 200-1000MB의 스왑 공간을 사용할 수 있습니다.

응용 프로그램에 대한 스왑 공간 요구 사항은 응용 프로그램 공급업체에 문의하십시오.

응용 프로그램 공급업체로부터 스왑 공간 요구 사항을 확인할 수 없는 경우 시스템 유형을 기반으로 다음 일반 지침을 수행하여 스왑 공간을 할당합니다.

**주** - 충돌 덤프 콘텐츠가 압축되므로 덤프 장치에는 물리적 메모리와 동일한 크기가 있을 필요가 없습니다. 기본적으로 덤프 콘텐츠 값은 커널 페이지로 설정됩니다. 그러나 덤프 콘텐츠 값이 모든 메모리 페이지를 덤프하도록 설정되면 덤프 크기를 물리적 메모리 크기의 절반 이상으로 늘릴 것을 고려해야 합니다.

표 19-1 ZFS 파일 시스템에 대한 스왑 및 덤프 볼륨 크기

시스템 유형	스왑 볼륨 크기	덤프 볼륨 크기
약 4GB의 물리적 메모리가 있는 시스템	1GB	2GB
약 8GB의 물리적 메모리가 있는 중급 서버	2GB	4GB
약 16GB-128GB의 물리적 메모리가 있는 고급 서버	4GB	8-64GB
128GB 이상의 물리적 메모리가 있는 고급 서버	물리적 메모리 크기의 1/4	물리적 메모리 크기의 1/2

**주** - 활성 ZFS 파일 시스템이 많이 있는 사용 중인 시스템은 덤프 장치 크기로 물리적 메모리 크기의 1/2에서 3/4을 사용합니다.

ZFS 루트 파일 시스템이 있는 시스템에서 `dumpadm -d` 명령을 사용하여 시스템 충돌 시스템을 보유하기에 너무 작은 덤프 장치를 지정하려고 하면 다음과 유사한 메시지가 표시됩니다.

```
dumpadm: dump device /dev/zvol/dsk/rpool/dump is too small to hold a
system dump dump size 43467329536 bytes, device size 42949672960 bytes
```

## ZFS 기반 시스템에 스왑 공간 할당

ZFS 루트 파일 시스템의 처음 설치하는 동안 ZFS 루트 풀의 ZFS 볼륨에서 스왑 영역이 자동으로 만들어집니다.

ZFS 루트 풀에서 스왑 장치는 고정된 크기의 슬라이스에 미리 할당되지 않으므로 나중에 매우 쉽게 스왑 크기를 수정할 수 있습니다.

응용 프로그램의 스왑 요구 사항을 평가한 후 기본 스왑 크기를 사용할 수 있고 필요한 경우에는 처음 설치하는 동안 또는 설치 후에 스왑 볼륨 크기를 조정할 수 있습니다.

처음 설치하는 동안 기본 덤프 볼륨 크기는 `dumpadm` 정보 및 물리적 메모리 크기를 기반으로 커널을 통해 계산됩니다.

ZFS 환경에서 파일 시스템은 풀의 공간을 사용합니다. 따라서 `/var/crash` 디렉토리는 충돌 덤프의 저장량에 따라 필요한 공간을 사용합니다.

## 스왑 리소스 모니터

`/usr/sbin/swap` 명령은 스왑 영역 관리에 사용됩니다. `-l` 및 `-s` 옵션은 둘 다 스왑 리소스에 대한 정보를 표시합니다.

`swap -l` 명령을 사용하여 시스템의 스왑 영역을 식별합니다. 활성화된 스왑 장치 또는 파일은 `swapfile` 열 아래에 나열됩니다. 예를 들면 다음과 같습니다.

```
# swap -l
swapfile          dev  swaplo blocks  free
/dev/dsk/c0t0d0s1 136,1    16 1638608 1600528
```

ZFS 루트 파일 시스템이 있는 시스템에서 `swap -l` 명령은 ZFS 볼륨 경로 이름을 식별한다는 점을 제외하고 유사한 출력을 식별합니다. 예를 들면 다음과 같습니다.

```
# swap -l
swapfile          dev  swaplo blocks  free
/dev/zvol/dsk/rpool/swap 256,1    16 1058800 1058800
```

`swap -S` 명령을 사용하여 스왑 리소스를 모니터합니다.

```
# swap -s
total: 57416k bytes allocated + 10480k reserved = 67896k used,
833128k available
```

`used` 값과 `available` 값의 합은 물리적 메모리와 스왑 장치(또는 파일)의 일부를 포함한 시스템의 전체 스왑 공간과 동일합니다.

시간 경과에 따른 스왑 공간 사용을 모니터링하기 위한 방법으로 `swap -s` 출력에서 사용 가능한 스왑 공간과 사용된 스왑 공간의 양을 사용할 수 있습니다. 시스템 성능이 양호하면 `swap -s`를 사용하여 사용 가능한 스왑 공간을 확인합니다. 시스템 성능이

느려지면 사용 가능한 스왑 공간의 양을 검사하여 스왑 공간이 줄어들었는지 확인합니다. 그런 다음 스왑 공간 사용을 증가시켰을 수 있는 시스템에 대한 변경 사항을 파악할 수 있습니다.

이 명령을 사용하는 경우 커널 및 사용자 프로세스가 물리적 메모리를 잠그고 잠금을 해제함에 따라 스왑에 사용할 수 있는 물리적 메모리의 양이 크게 변한다는 사실을 염두에 두십시오.

주 - `swap -l` 명령은 512바이트 블록으로 스왑 공간을 표시합니다. `swap -s` 명령은 1024바이트 블록으로 스왑 공간을 표시합니다. `swap -l`에서 블록을 추가하고 해당 블록을 KB로 변환하면 결과는 `swap -s` 출력에서 `used`와 `available`의 합보다 작습니다. 이유는 `swap -l`이 스왑 공간 계산에 물리적 메모리를 포함하지 않기 때문입니다.

다음 표에서는 `swap -s` 명령의 출력을 요약합니다.

표 19-2 `swap -s` 명령 출력

키워드	설명
bytes allocated	보조 기억 장치(디스크 지원 스왑 공간)로 현재 할당된, 1024바이트 블록으로 표시된 총 스왑 공간입니다.
reserved	현재 할당되지 않지만 추후 사용 가능하도록 메모리에서 요청하는 1024바이트 블록으로 표시된 총 스왑 공간입니다.
used	할당 또는 예약된, 1024바이트 블록으로 표시된 총 스왑 공간입니다.
available	추후 예약 및 할당을 위해 현재 사용 가능할 수 있는, 1024바이트 블록으로 표시된 총 스왑 공간입니다.

## Oracle Solaris ZFS 루트 환경에서 스왑 공간 추가 및 변경

다음 단원에서는 ZFS 루트 환경에서 스왑 공간을 추가하거나 변경하는 방법에 대해 설명합니다. 시스템 또는 응용 프로그램에 추가 스왑 공간이 필요한지 확인하려면 이전 단원을 참조하십시오.

### ▼ Oracle Solaris ZFS 루트 환경에서 스왑 공간을 추가하는 방법

#### 1 관리자로 전환합니다.

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

**2 현재 스왑 볼륨을 파악합니다.**

스왑 볼륨은 사용 중에는 제거할 수 없습니다. `blocks` 열에서 파악된 블록과 `free` 열에서 파악된 블록을 비교하여 현재 스왑 볼륨이 사용 중인지 확인할 수 있습니다. 두 열의 블록이 동일하면 스왑 영역은 사용 중이 아닙니다. 예를 들면 다음과 같습니다.

```
# swap -l
swapfile          dev swaplo  blocks  free
/dev/zvol/dsk/rpool/swap 256,1      16 1058800 1058800
```

**3 다음 중 하나를 선택하여 스왑 볼륨의 크기를 조정합니다.****a. 현재 스왑 영역이 사용 중이 아니면 현재 스왑 볼륨의 크기를 조정할 수 있습니다.**

예를 들면 다음과 같습니다.

```
# zfs get volsize rpool/swap
NAME          PROPERTY VALUE  SOURCE
rpool/swap  volsize  517M  -
# zfs set volsize=2g rpool/swap
# zfs get volsize rpool/swap
NAME          PROPERTY VALUE  SOURCE
rpool/swap  volsize  2G    -
```

**b. 현재 스왑 영역이 사용 중이면 다른 스왑 볼륨을 추가할 수 있습니다.**

예를 들면 다음과 같습니다.

```
# zfs create -V 2G rpool/swap2
```

**c. 두번째 스왑 볼륨을 활성화합니다.**

예를 들면 다음과 같습니다.

```
# swap -a /dev/zvol/dsk/rpool/swap2
# swap -l
swapfile          dev swaplo  blocks  free
/dev/zvol/dsk/rpool/swap 256,1      16 1058800 1058800
/dev/zvol/dsk/rpool/swap2 256,3      16 4194288 4194288
```





## 파일 및 파일 시스템 복사(작업)

---

이 장에서는 다양한 백업 명령을 사용하여 파일 및 파일 시스템을 디스크, 테이프 및 디스켓에 복사하는 방법을 설명합니다.

이 장에서 다루는 단계별 지침은 다음과 같습니다.

- 356 페이지 “파일 시스템 간에 디렉토리를 복사하는 방법(cpio)”
- 357 페이지 “파일을 테이프에 복사하는 방법(tar)”
- 358 페이지 “테이프에 있는 파일을 나열하는 방법(tar)”
- 359 페이지 “테이프에서 파일을 검색하는 방법(tar)”
- 360 페이지 “pax 명령을 사용하여 파일을 테이프에 복사”
- 361 페이지 “디렉토리에 있는 모든 파일을 테이프에 복사하는 방법(cpio)”
- 362 페이지 “테이프에 있는 파일을 나열하는 방법(cpio)”
- 362 페이지 “테이프에서 모든 파일을 검색하는 방법(cpio)”
- 363 페이지 “테이프에서 특정 파일을 검색하는 방법(cpio)”
- 364 페이지 “파일을 원격 테이프 장치에 복사하는 방법(tar 및 dd)”
- 365 페이지 “원격 테이프 장치에서 파일을 추출하는 방법”

### 파일 시스템 복사 명령

개별 파일, 파일 시스템의 일부 또는 전체 파일 시스템을 복사하거나 이동하려는 경우에 이 장에 설명된 절차를 사용할 수 있습니다.

다음 표는 Oracle Solaris 릴리스에서 사용할 수 있는 다양한 백업 및 복원 명령에 대해 설명합니다. 엔터프라이즈 환경에서는 엔터프라이즈 레벨의 백업 제품 사용을 고려해 보시기 바랍니다. 엔터프라이즈 레벨의 백업 제품에 대한 자세한 내용은 Oracle Technical Network를 참조하십시오.

표 20-1 다양한 백업 명령 요약

명령 이름	파일 시스템 경계 인식 여부	다중 볼륨 백업 지원 여부	물리적/논리적 복사
volcopy	예	예	물리적
tar	아니오	아니오	논리적
cpio	아니오	예	논리적
pax	예	예	논리적
dd	예	아니오	물리적
zfs send 및 zfs receive	예	해당 없음	논리적
zfs snapshot	예	해당 없음	논리적

다음 표는 각 명령의 장점과 단점에 대해 설명합니다.

표 20-2 tar, pax 및 cpio 명령의 장점과 단점

명령	기능	장점	단점
tar	파일 및 디렉토리 하위 트리를 테이프 하나에 복사하는 데 사용합니다.	<ul style="list-style-type: none"> <li>■ 대부분의 UNIX 운영 체제에서 사용할 수 있습니다.</li> <li>■ 공용 도메인 버전을 쉽게 사용할 수 있습니다.</li> </ul>	<ul style="list-style-type: none"> <li>■ 파일 시스템 경계를 인식하지 않습니다.</li> <li>■ 전체 경로 이름의 길이가 255자를 초과할 수 없습니다.</li> <li>■ 여러 개의 테이프 볼륨을 만드는 데 사용할 수 없습니다.</li> </ul>
pax	여러 개의 테이프 볼륨이 필요한 파일, 특수 파일 또는 파일 시스템을 복사하는 데 사용합니다. 또는 파일을 POSIX 호환 시스템에서 또는 POSIX 호환 시스템으로 복사하려는 경우에 사용합니다.	<ul style="list-style-type: none"> <li>■ POSIX 호환 시스템에서 tar 또는 cpio 명령보다 이식성이 좋습니다.</li> <li>■ 여러 공급업체를 지원합니다.</li> </ul>	tar 명령과 비교했을 때 비슷한 단점을 갖고 있지만, pax 명령으로는 여러 개의 테이프 볼륨을 만들 수 있습니다.

표 20-2 tar, pax 및 cpio 명령의 장점과 단점		(계속)	
명령	기능	장점	단점
cpio	여러 개의 테이프 볼륨이 필요한 파일, 특수 파일 또는 파일 시스템을 복사하는 데 사용합니다. 또는 최신 Oracle Solaris 릴리스를 실행하는 시스템에서 이전 Solaris 릴리스를 실행하는 시스템으로 파일을 복사하려는 경우에 사용합니다.	<ul style="list-style-type: none"> <li>■ tar 명령보다 효율적으로 데이터를 테이프에 압축합니다.</li> <li>■ 복원할 때 테이프에서 잘못된 부분을 모두 건너뛸니다.</li> <li>■ 서로 다른 시스템 유형 간의 이식성을 위해 tar, ustar, crc, odc, bar 등의 여러 헤더 형식을 사용하여 파일을 기록하는 옵션을 제공합니다.</li> <li>■ 여러 개의 테이프 볼륨을 만듭니다.</li> </ul>	명령 구문은 tar 또는 pax 명령보다 어렵습니다.

다음 섹션에서는 단계별 지침을 설명하고 이러한 명령을 사용하는 방법의 예를 제공합니다.

## 파일 시스템 간에 디렉토리 복사(cpio 명령)

cpio(copy in and out) 명령을 사용하여 개별 파일, 파일 그룹 또는 전체 파일 시스템을 복사할 수 있습니다. 이 섹션에서는 cpio 명령을 사용하여 전체 파일 시스템을 복사하는 방법을 설명합니다.

cpio 명령은 파일 목록을 하나의 큰 출력 파일에 복사하는 아카이브 프로그램입니다. 이 명령은 개별 파일 사이에 헤더를 삽입하여 복구를 지원합니다. cpio 명령을 사용하면 전체 파일 시스템을 다른 슬라이스, 다른 시스템 또는 테이프나 USB 디스켓 등의 다른 매체 장치로 복사할 수 있습니다.

cpio 명령은 매체의 끝 부분을 인식하고 다른 볼륨을 삽입하라는 내용의 프롬프트를 표시하기 때문에 여러 테이프 또는 USB 디스켓을 사용해야 하는 아카이브를 만드는 경우에 가장 효과적인 명령입니다.

cpio 명령을 사용하면 ls 및 find 명령을 사용하여 복사할 파일을 나열 및 선택한 다음 결과를 cpio 명령으로 파이프 출력하는 경우가 많습니다.

## ▼ 파일 시스템 간에 디렉토리를 복사하는 방법(cpio)

1 관리자로 전환합니다.

2 적절한 디렉토리로 변경합니다.

```
# cd filesystem1
```

3 **find** 및 **cpio** 명령을 사용하여 *filesystem1*에서 *filesystem2*로 디렉토리 트리를 복사합니다.

```
# find . -print -depth | cpio -pdm filesystem2
```

. 현재 작업 디렉토리에서 시작합니다.

-print 파일 이름을 인쇄합니다.

-depth 디렉토리 계층을 내려가 파일 이름을 아래쪽부터 인쇄합니다.

-p 파일 목록을 만듭니다.

-d 필요에 따라 디렉토리를 만듭니다.

-m 디렉토리에 올바른 수정 시간을 설정합니다.

자세한 내용은 **cpio(1)**를 참조하십시오.

지정한 디렉토리 이름에 해당되는 파일을 복사합니다. 심볼릭 링크는 보존됩니다.

-u 옵션을 지정할 수도 있습니다. 이 옵션은 무조건 복사를 강제로 실행합니다. 이전 파일로 새 파일을 대체하려면 이 옵션을 지정해야 합니다. 디렉토리의 정확한 복사본이 필요하며 복사할 파일 일부가 이미 대상 디렉토리에 있는 경우에 이 옵션이 유용할 수 있습니다.

4 대상 디렉토리의 내용을 표시하여 복사에 성공했는지 확인합니다.

```
# cd filesystem2
```

```
# ls
```

5 필요한 경우는 소스 디렉토리를 제거합니다.

```
# rm -rf filesystem1
```

### 예 20-1 파일 시스템 간에 디렉토리 복사(cpio)

```
# cd /data1
# find . -print -depth | cpio -pdm /data2
19013 blocks
# cd /data2
# ls
# rm -rf /data1
```

## 파일 및 파일 시스템을 테이프에 복사

tar, pax 및 cpio 명령을 사용하여 파일 및 파일 시스템을 테이프에 복사할 수 있습니다. 복사 시 원하는 유연성과 정확성에 따라 선택하는 명령이 달라집니다. 세 개의 명령이 모두 원시 장치를 사용하므로 테이프를 사용하기 전에 포맷하거나 파일 시스템을 만들 필요는 없습니다.

사용하는 테이프 드라이브 및 장치 이름은 각 시스템의 하드웨어 구성에 따라 달라집니다. 테이프 장치 이름에 대한 자세한 내용은 367 페이지 “사용할 매체 선택”을 참조하십시오.

## 파일을 테이프에 복사(tar 명령)

다음은 tar 명령을 사용하여 파일을 복사하기 전에 알아야 할 사항입니다.

- tar 명령에 -c 옵션을 사용하여 파일을 테이프에 복사하면 현재 테이프 위치 또는 그 이후에 이미 테이프에 있던 파일이 모두 삭제됩니다.
- 파일을 복사할 때 지정하는 파일 이름의 일부로 파일 이름 대체 와일드카드(? 및 \*)를 사용할 수 있습니다. 예를 들어, 접미어가 .doc인 모든 문서를 복사하려면 파일 이름 인수로 \*.doc를 입력합니다.
- tar 아카이브에서 파일을 추출하는 경우에는 파일 이름 대체 와일드카드를 사용할 수 없습니다.

### ▼ 파일을 테이프에 복사하는 방법(tar)

- 1 복사할 파일이 포함된 디렉토리로 변경합니다.
- 2 쓰기 가능한 테이프를 테이프 드라이브에 삽입합니다.
- 3 파일을 테이프에 복사합니다.

```
$ tar cvf /dev/rmt/n filenames
```

c                    아카이브를 만드는 것을 나타냅니다.

v                    아카이브 처리하는 동안 각 파일의 이름을 표시합니다.

f /dev/rmt/ n      지정된 장치 또는 파일에 아카이브를 기록하는 것을 나타냅니다.

filenames        복사할 파일 및 디렉토리를 나타냅니다. 여러 개의 파일은 공백으로 분리합니다.

지정된 파일 이름이 테이프에 복사되며 테이프에 있는 기존 파일을 덮어씁니다.

- 4 드라이브에서 테이프를 제거합니다. 테이프 레이블에 파일의 이름을 씁니다.
- 5 복사한 파일이 테이프에 있는지 확인합니다.

```
$ tar tvf /dev/rmt/n
```

tar 테이프의 파일 나열에 대한 자세한 내용은 358 페이지 “테이프에 있는 파일을 나열하는 방법(tar)”을 참조하십시오.

## 예 20-2 파일을 테이프에 복사(tar)

다음 예는 세 개의 파일을 테이프 드라이브 0에 있는 테이프에 복사하는 방법을 보여줍니다.

```
$ cd /export/home/kryten
$ ls reports
reportA reportB reportC
$ tar cvf /dev/rmt/0 reports
a reports/ 0 tape blocks
a reports/reportA 59 tape blocks
a reports/reportB 61 tape blocks
a reports/reportC 63 tape blocks
$ tar tvf /dev/rmt/0
```

## ▼ 테이프에 있는 파일을 나열하는 방법(tar)

- 1 테이프를 테이프 드라이브에 삽입합니다.
- 2 테이프 내용을 표시합니다.

```
$ tar tvf /dev/rmt/n
```

t                    테이프에 있는 파일의 콘텐츠를 표시합니다.

v                    t 옵션과 함께 사용하며 테이프에 있는 파일에 대해 자세한 정보를 제공합니다.

f /dev/rmt/n        테이프 장치를 나타냅니다.

## 예 20-3 테이프에 있는 파일 나열(tar)

다음 예는 드라이브 0의 테이프에 있는 파일의 목록을 표시합니다.

```
$ tar tvf /dev/rmt/0
drwxr-xr-x  0/0          0 Jul 14 13:50 2010 reports/
-r--r--r--  0/0    206663 Jul 14 13:50 2010 reports/reportC
-r--r--r--  0/0    206663 Jul 14 13:50 2010 reports/reportB
-r--r--r--  0/0    206663 Jul 14 13:50 2010 reports/reportA
```

## ▼ 테이프에서 파일을 검색하는 방법(tar)

- 1 파일을 넣을 디렉토리로 변경합니다.
- 2 테이프를 테이프 드라이브에 삽입합니다.
- 3 테이프에서 파일을 검색합니다.

```
$ tar xvf /dev/rmt/n [filenames]
```

x                    지정한 아카이브 파일에서 파일을 추출해야 하는 것을 나타냅니다.  
지정된 드라이브의 테이프에 있는 모든 파일이 현재 디렉토리에 복사됩니다.

v                    검색하는 동안 각 파일의 이름을 표시합니다.

f /dev/rmt/n        아카이브가 포함된 테이프 장치를 나타냅니다.

filenames         검색할 파일을 지정합니다. 여러 개의 파일은 공백으로 분리합니다.  
자세한 내용은 [tar\(1\)](#) 매뉴얼 페이지를 참조하십시오.

- 4 파일이 복사되었는지 확인합니다.

```
$ ls -l
```

### 예 20-4 테이프에서 파일 검색(tar)

다음 예는 드라이브 0에 있는 테이프에서 파일을 모두 검색하는 방법을 보여 줍니다.

```
$ cd /var/tmp
$ tar xvf /dev/rmt/0
x reports/, 0 bytes, 0 tape blocks
x reports/reportA, 0 bytes, 0 tape blocks
x reports/reportB, 0 bytes, 0 tape blocks
x reports/reportC, 0 bytes, 0 tape blocks
x reports/reportD, 0 bytes, 0 tape blocks
$ ls -l
```

**일반 오류**   테이프에서 추출한 파일의 이름이 아카이브에 저장된 파일의 이름과 정확하게 일치해야 합니다. 파일의 이름이나 경로가 확실하지 않은 경우에는 먼저 테이프에 있는 파일을 나열해 보십시오. 테이프에 있는 파일을 나열하는 방법에 대한 지침은 [358 페이지](#) “테이프에 있는 파일을 나열하는 방법(tar)”을 참조하십시오.

## pax 명령을 사용하여 파일을 테이프에 복사

### ▼ 파일을 테이프에 복사하는 방법(pax)

- 1 복사할 파일이 포함된 디렉토리로 변경합니다.
- 2 쓰기 가능한 테이프를 테이프 드라이브에 삽입합니다.
- 3 파일을 테이프에 복사합니다.

```
$ pax -w -f /dev/rmt/n filenames
```

-w 쓰기 모드를 사용합니다.

-f /dev/rmt/n 테이프 드라이브를 식별합니다.

*filenames* 복사할 파일 및 디렉토리를 나타냅니다. 여러 개의 파일은 공백으로 분리합니다.

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

- 4 파일이 테이프에 복사되었는지 확인합니다.

```
$ pax -f /dev/rmt/n
```

- 5 드라이브에서 테이프를 제거합니다. 테이프 레이블에 파일의 이름을 씁니다.

#### 예 20-5 파일을 테이프에 복사(pax)

다음 예는 pax 명령을 사용하여 현재 디렉토리에 있는 모든 파일을 복사하는 방법을 보여 줍니다.

```
$ pax -w -f /dev/rmt/0 .  
$ pax -f /dev/rmt/0  
filea fileb filec
```



## cpio 명령을 사용하여 파일을 테이프에 복사

### ▼ 디렉토리에 있는 모든 파일을 테이프에 복사하는 방법(cpio)

- 1 복사할 파일이 포함된 디렉토리로 변경합니다.
- 2 쓰기 가능한 테이프를 테이프 드라이브에 삽입합니다.
- 3 파일을 테이프에 복사합니다.

```
$ ls | cpio -oc > /dev/rmt/n
```

ls cpio 명령에 파일 이름 목록을 제공합니다.

cpio -oc cpio 명령이 copy-out 모드로 작동하며(-o) 헤더 정보를 ASCII 문자 형식으로 쓰도록(-c) 지정합니다. 이 옵션을 사용하면 다른 공급업체 시스템에 대한 이식성을 보장할 수 있습니다.

```
> /dev/rmt/n
```

출력 파일을 지정합니다.

디렉토리에 있는 모든 파일이 지정한 드라이브의 테이프로 복사되며 테이프에 있는 기존 파일을 덮어씁니다. 복사한 총 블록 수가 표시됩니다.

- 4 파일이 테이프에 복사되었는지 확인합니다.

```
$ cpio -civt < /dev/rmt/n
```

-c cpio 명령이 파일을 ASCII 문자 형식으로 읽도록 지정합니다.

-i 지금은 파일만 나열하지만, cpio 명령이 copy-in 모드로 작동하도록 지정합니다.

-v ls -l 명령의 출력과 비슷한 형식으로 출력을 표시합니다.

-t 지정한 테이프 드라이브의 테이프에 있는 파일에 대한 콘텐츠를 나열합니다.

```
< /dev/rmt/n
```

기존 cpio 아카이브의 입력 파일을 지정합니다.

- 5 드라이브에서 테이프를 제거합니다. 테이프 레이블에 파일의 이름을 씁니다.

#### 예 20-6 디렉토리에 있는 모든 파일을 테이프에 복사(cpio)

다음 예는 /export/home/kryten 디렉토리에 있는 모든 파일을 테이프 드라이브 0의 테이프에 복사하는 방법을 보여 줍니다.

```
$ cd /export/home/kryten
$ ls | cpio -oc > /dev/rmt/0
1280 blocks
$ cpio -civt < /dev/rmt/0
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filea
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, fileb
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filec
drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, letters
drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, reports
1280 blocks
```

## ▼ 테이프에 있는 파일을 나열하는 방법(cpio)

주 - 테이프의 콘텐츠를 나열하려면 cpio 명령이 아카이브 전체를 처리해야 하기 때문에 시간이 오래 걸립니다.

- 1 아카이브 테이프를 테이프 드라이브에 삽입합니다.
- 2 테이프에 있는 파일을 나열합니다.

```
$ cpio -civt < /dev/rmt/n
```

### 예 20-7 테이프에 있는 파일 나열(cpio)

다음 예는 드라이브 0에 있는 테이프의 파일을 나열하는 방법을 보여 줍니다.

```
$ cpio -civt < /dev/rmt/0
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filea
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, fileb
-r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filec
drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, letters
drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, reports
1280 blocks
```

## ▼ 테이프에서 모든 파일을 검색하는 방법(cpio)

상대 경로 이름을 사용하여 아카이브를 만든 경우에는 파일을 검색할 때 입력 파일이 현재 디렉토리에 있는 디렉토리로 구성됩니다. 그러나 절대 경로 이름을 사용하여 아카이브를 만든 경우에는 시스템에서 동일한 절대 경로를 사용하여 파일을 다시 만듭니다.



주의 - 절대 경로 이름을 사용하면 시스템에 있는 기존 파일을 덮어쓸 위험이 있습니다.

- 1 파일을 넣을 디렉토리로 변경합니다.
- 2 테이프를 테이프 드라이브에 삽입합니다.

- 3 테이프에서 모든 파일을 추출합니다.

```
$ cpio -icvd < /dev/rmt/n
```

-i                   표준 입력에서 파일을 추출합니다.  
 -c                   cpio 명령이 파일을 ASCII 문자 형식으로 읽도록 지정합니다.  
 -v                   ls 명령의 출력과 비슷한 형식으로 파일을 검색하여 표시합니다.  
 -d                   필요에 따라 디렉토리를 만듭니다.  
 < /dev/rmt/n       출력 파일을 지정합니다.

- 4 파일이 복사되었는지 확인합니다.

```
$ ls -l
```

#### 예 20-8 테이프에서 모든 파일 검색(cpio)

다음 예는 드라이브 0에 있는 테이프에서 파일을 모두 검색하는 방법을 보여 줍니다.

```
$ cd /var/tmp
cpio -icvd < /dev/rmt/0
answers
sc.directives
tests
8 blocks
$ ls -l
```

## ▼ 테이프에서 특정 파일을 검색하는 방법(cpio)

- 1 파일을 넣을 디렉토리로 변경합니다.
- 2 테이프를 테이프 드라이브에 삽입합니다.
- 3 테이프에서 일부 파일을 검색합니다.

```
$ cpio -icv "*file" < /dev/rmt/n
```

-i                   표준 입력에서 파일을 추출합니다.  
 -c                   cpio 명령이 헤더를 ASCII 문자 형식으로 읽도록 지정합니다.  
 -v                   ls 명령의 출력과 비슷한 형식으로 파일을 검색하여 표시합니다.

"\*file" 패턴과 일치하는 모든 파일이 현재 디렉토리에 복사되도록 지정합니다. 여러 패턴을 지정할 수도 있지만 그 경우 각 패턴을 큰따옴표로 둘러싸야 합니다.

< /dev/rmt/n 입력 파일을 지정합니다.

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

#### 4 파일이 복사되었는지 확인합니다.

```
$ ls -l
```

### 예 20-9 테이프에서 특정 파일 검색(cpio)

다음 예는 드라이브 0에 있는 테이프에서 접미어가 chapter인 파일을 모두 검색하는 방법을 보여 줍니다.

```
$ cd /home/smith/Book
$ cpio -icv "**chapter" < /dev/rmt/0
Boot.chapter
Directory.chapter
Install.chapter
Intro.chapter
31 blocks
$ ls -l
```

## 파일을 테이프 장치에 복사

### ▼ 파일을 원격 테이프 장치에 복사하는 방법(tar 및 dd)

#### 1 원격 테이프 드라이브를 사용하려면 다음과 같은 필수 조건을 충족해야 합니다.

a. 원격 시스템의 /etc/hosts.equiv 파일에 로컬 호스트 이름이 나타나야 합니다. 선택적으로 복사를 수행하는 사용자의 사용자 이름이 나타날 수도 있습니다. 또는 원격 시스템에서 복사를 수행하는 사용자의 홈 디렉토리에 액세스할 수 있고 \$HOME/.rhosts에 로컬 시스템 이름이 있어야 합니다.

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

b. 로컬 시스템의 /etc/inet/hosts 파일이나 이름 서비스 hosts 파일에 원격 시스템 항목이 있어야 합니다.

#### 2 원격 명령을 실행하기에 적합한 권한이 있는지 테스트하려면 다음을 수행하십시오.

```
$ rsh remotehost echo test
```

test라고 표시되면 원격 명령을 실행할 권한이 있습니다. Permission denied라고 표시되면 1단계에 설명된 대로 설정을 확인하십시오.

- 3 파일을 넣을 디렉토리로 변경합니다.
- 4 테이프를 테이프 드라이브에 삽입합니다.
- 5 파일을 원격 테이프 드라이브에 복사합니다.

```
$ tar cvf - filenames | rsh remote-host dd of=/dev/rmt/n obs=block-size
```

tar cf	테이프 아카이브를 만들고 아카이브 처리되는 파일을 나열하며 테이프 장치를 지정합니다.
v	tar 파일 항목에 대한 추가 정보를 제공합니다.
-(하이픈)	테이프 장치의 자리 표시자를 나타냅니다.
filenames	복사할 파일을 식별합니다. 여러 개의 파일은 공백으로 분리합니다.
rsh remote-host	tar 명령의 결과를 원격 셸로 파이프 출력합니다.
dd of= /dev/rmt/n	출력 장치를 나타냅니다.
obs=block-size	블록화 인자를 나타냅니다.

- 6 드라이브에서 테이프를 제거합니다. 테이프 레이블에 파일의 이름을 씁니다.

#### 예 20-10 원격 테이프 드라이브에 파일 복사(tar 및 dd)

```
# tar cvf - * | rsh mercury dd of=/dev/rmt/0 obs=126b
a answers/ 0 tape blocks
a answers/test129 1 tape blocks
a sc.directives/ 0 tape blocks
a sc.directives/sc.190089 1 tape blocks
a tests/ 0 tape blocks
a tests/test131 1 tape blocks
6+9 records in
0+1 records out
```

### ▼ 원격 테이프 장치에서 파일을 추출하는 방법

- 1 테이프를 테이프 드라이브에 삽입합니다.
- 2 임시 디렉토리로 변경합니다.

```
$ cd /var/tmp
```

**3 원격 테이프 장치에서 파일을 추출합니다.**

```
$ rsh remote-host dd if=/dev/rmt/n | tar xvBpf -
```

rsh remote-host dd 명령을 사용하여 테이프 장치에서 파일을 추출하기 시작한 원격 셸을 나타냅니다.

dd if=/dev/rmt/n 입력 장치를 나타냅니다.

| tar xvBpf - dd 명령의 결과를 파일 복원에 사용되는 tar 명령으로 파이프 출력합니다.

**4 파일이 추출되었는지 확인합니다.**

```
$ ls -l
```

**예 20-11 원격 테이프 드라이브에서 파일 추출**

```
$ cd /var/tmp
$ rsh mercury dd if=/dev/rmt/0 | tar xvBpf -
x answers/, 0 bytes, 0 tape blocks
x answers/test129, 48 bytes, 1 tape blocks
20+0 records in
20+0 records out
x sc.directives/, 0 bytes, 0 tape blocks
x sc.directives/sc.190089, 77 bytes, 1 tape blocks
x tests/, 0 bytes, 0 tape blocks
x tests/test131, 84 bytes, 1 tape blocks
$ ls -l
```

## 테이프 드라이브 관리(작업)

---

이 장에서는 Oracle Solaris OS에서 테이프 드라이브를 관리하는 방법을 설명합니다.

이 장에서 다루는 단계별 지침은 다음과 같습니다.

- 370 페이지 “테이프 드라이브 상태를 표시하는 방법”
- 371 페이지 “자기 테이프 카트리지를 다시 조이기”
- 372 페이지 “자기 테이프 카트리지 되감기”

이 장에서 다루는 개요 정보는 다음과 같습니다.

- 367 페이지 “사용할 매체 선택”
- 368 페이지 “백업 장치 이름”
- 370 페이지 “테이프 드라이브 상태 표시”

### 사용할 매체 선택

일반적으로 다음과 같은 테이프 매체를 사용하여 Oracle Solaris 시스템을 백업합니다.

- 1/2인치 릴 테이프
- 1/4인치 스트리밍 카트리지 테이프
- 8mm 카트리지 테이프
- 4mm 카트리지 테이프(DAT)

선택하는 매체는 지원하는 장비의 가용성과 파일을 저장하는 데 사용하는 매체(보통은 테이프)에 따라 달라집니다. 로컬 시스템에서 백업을 수행해야 하지만 파일을 원격 장치에 쓸 수도 있습니다.

다음 표는 파일 시스템 백업에 사용되는 일반적인 테이프 장치를 보여 줍니다. 각 장치의 저장소 용량은 드라이브 유형과 테이프에 쓰는 데이터에 따라 달라집니다.

표 21-1 매체 저장소 용량

백업 매체	저장소 용량
1/2인치 릴 테이프	140MB(6250bpi)
2.5GB 1/4인치 카트리지(QIC) 테이프	2.5GB
DDS3 4mm 카트리지 테이프(DAT)	12-24GB
14GB 8mm 카트리지 테이프	14GB
DLT 7000 1/2인치 카트리지 테이프	35-70GB

## 백업 장치 이름

백업에 사용할 테이프 또는 장치는 논리적 장치 이름을 제공하여 지정합니다. 이 이름은 "원시" 장치 이름을 포함한 하위 디렉토리를 가리키며 드라이브의 논리 장치 번호를 포함합니다. 테이프 드라이브 이름 지정 규약에서는 물리적 장치 이름이 아닌 논리적 장치 이름을 사용합니다. 다음 표는 이러한 이름 지정 규약을 보여 줍니다.

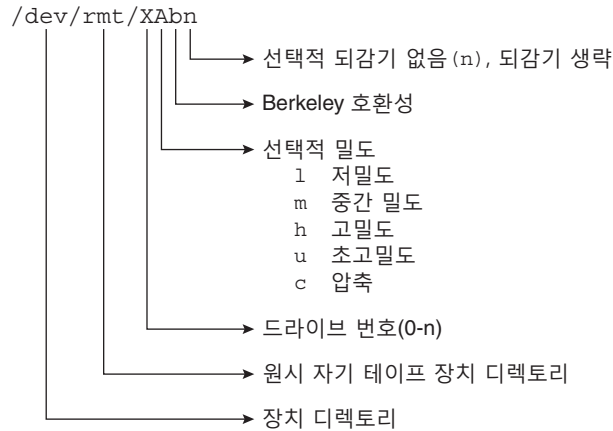
표 21-2 백업 장치의 기본 장치 이름

장치 유형	이름
테이프	/dev/rmt/ <i>n</i>

일반적으로는 다음 그림에 표시된 것과 같이 테이프 장치를 지정합니다.



그림 21-1 테이프 드라이브 장치 이름



밀도를 지정하지 않으면 테이프 드라이브는 일반적으로 "기본 설정" 밀도를 사용합니다. 기본 설정 밀도는 대개 테이프 드라이브가 지원하는 최고의 밀도를 의미합니다. 대부분의 SCSI 드라이브는 테이프의 밀도나 형식을 자동으로 감지하여 읽을 수 있습니다. 드라이브에서 지원되는 여러 밀도를 확인하려면 /dev/rmt 하위 디렉토리를 봅니다. 이 하위 디렉토리에는 각 테이프의 여러 출력 밀도를 지원하는 테이프 장치 파일 세트가 포함되어 있습니다.

또한 SCSI 제어기 하나에는 최대 7개의 SCSI 테이프 드라이브를 지정할 수 있습니다.

## 테이프 드라이브의 되감기 옵션 지정

일반적으로 테이프 드라이브는 0에서  $n$ 까지의 논리 장치 번호로 지정합니다. 다음 표는 되감기 또는 되감기 없음 옵션에 테이프 장치 이름을 지정하는 방법에 대해 설명합니다.

표 21-3 테이프 드라이브에 되감기 또는 되감기 없음 지정

드라이브 및 되감기 값	옵션 사용
첫번째 드라이브, 되감기	/dev/rmt/0
첫번째 드라이브, 되감기 없음	/dev/rmt/0n
두번째 드라이브, 되감기	/dev/rmt/1
두번째 드라이브, 되감기 없음	/dev/rmt/1n

## 테이프 드라이브에 다양한 밀도 지정

기본적으로 드라이브는 "기본 설정" 밀도로 기록하며, 기본 설정 밀도는 대체로 테이프 드라이브가 지원하는 최고의 밀도입니다. 테이프 장치를 지정하지 않으면 이 명령은 장치가 지원하는 기본 밀도로 드라이브 번호 0에 씁니다.

테이프 드라이브가 특정 밀도만 지원하는 시스템으로 테이프를 전송하려면 원하는 밀도로 쓸 수 있는 장치 이름을 지정합니다. 다음 표는 테이프 드라이브에 다양한 밀도를 지정하는 방법에 대해 설명합니다.

표 21-4 테이프 드라이브에 다양한 밀도 지정

드라이브, 밀도 및 퇴감기 값	옵션 사용
첫번째 드라이브, 저밀도, 퇴감기	/dev/rmt/0l
첫번째 드라이브, 저밀도, 퇴감기 없음	/dev/rmt/0ln
두번째 드라이브, 중간 밀도, 퇴감기	/dev/rmt/1m
두번째 드라이브, 중간 밀도, 퇴감기 없음	/dev/rmt/1mn

추가 밀도 값은 368 페이지 “백업 장치 이름”을 참조하십시오.

## 테이프 드라이브 상태 표시

mt 명령에 status 옵션을 사용하여 테이프 드라이브에 대한 상태 정보를 가져올 수 있습니다. mt 명령은 /kernel/drv/st.conf 파일에 설정된 모든 테이프 드라이브에 대한 정보를 보고합니다.

### ▼ 테이프 드라이브 상태를 표시하는 방법

- 1 정보를 보고 싶은 드라이브에 테이프를 로드합니다.
- 2 테이프 드라이브 상태를 표시합니다.  
# mt -f /dev/rmt/n status
- 3 테이프 드라이브 번호 0, 1, 2, 3 등을 바꿔 가며 1-2 단계를 반복하여 사용 가능한 모든 테이프 드라이브에 대해 정보를 표시합니다.

#### 예 21-1 테이프 드라이브 상태 표시

다음 예는 QIC-150 테이프 드라이브(/dev/rmt/0)의 상태를 표시합니다.

```
$ mt -f /dev/rmt/0 status
Archive QIC-150 tape drive:
  sense key(0x0)= No Additional Sense  residual= 0  retries= 0
  file no= 0  block no= 0
```

다음 예는 Exabyte 테이프 드라이브(/dev/rmt/1)의 상태를 표시합니다.

```
$ mt -f /dev/rmt/1 status
Exabyte EXB-8200 8mm tape drive:
sense key(0x0)= NO Additional Sense residual= 0  retries= 0
file no= 0  block no= 0
```

다음 예는 빠르게 시스템을 풀하고 해당 테이프 드라이브를 모두 찾는 방법을 보여줍니다.

```
$ for drive in 0 1 2 3 4 5 6 7
> do
> mt -f /dev/rmt/$drive status
> done
Archive QIC-150 tape drive:
  sense key(0x0)= No Additional Sense  residual= 0  retries= 0
  file no= 0  block no= 0
/dev/rmt/1: No such file or directory
/dev/rmt/2: No such file or directory
/dev/rmt/3: No such file or directory
/dev/rmt/4: No such file or directory
/dev/rmt/5: No such file or directory
/dev/rmt/6: No such file or directory
/dev/rmt/7: No such file or directory
$
```

## 자기 테이프 카트리지 처리

테이프를 읽는 동안 오류가 발생한 경우에는 테이프를 다시 조이거나 테이프 드라이브를 청소한 후 다시 시도할 수 있습니다.

### 자기 테이프 카트리지 다시 조이기

mt 명령으로 자기 테이프 카트리지를 다시 조입니다.

예를 들면 다음과 같습니다.

```
$ mt -f /dev/rmt/1 retension
$
```

---

주 - QIC가 아닌 테이프 드라이브를 다시 조이지 마십시오.

---

## 자기 테이프 카트리지 되감기

자기 테이프 카트리지를 되감으려면 `mt` 명령을 사용합니다.

예를 들면 다음과 같습니다.

```
$ mt -f /dev/rmt/1 rewind
$
```

## 드라이브 유지 관리 및 매체 사용 지침

읽을 수 없는 백업 테이프는 쓸모가 없습니다. 따라서 테이프 드라이브가 올바르게 작동하도록 주기적으로 청소하고 검사하십시오. 테이프 드라이브 청소 절차에 대한 안내는 하드웨어 매뉴얼을 참조하십시오. 일부 파일을 테이프에 복사하고 파일을 다시 읽은 다음, 원래 파일을 복사한 파일과 비교하여 테이프 하드웨어를 검사할 수 있습니다.

시스템이 보고하지 않는 방식으로 하드웨어 오류가 발생할 수 있다는 점에 주의하십시오.

백업한 후에는 항상 테이프에 레이블을 지정하십시오. 이 레이블은 결코 변경하지 말아야 합니다. 백업할 때마다 다음과 같은 정보가 포함된 테이프 레이블을 새로 만드십시오.

- 백업 날짜
- 백업한 파일 시스템 및 시스템의 이름
- 백업 레벨
- 테이프 번호(백업이 여러 볼륨으로 구성된 경우  $1/n$ )
- 사이트 특정 정보

테이프는 먼지가 없고 안전하며 자기장에서 먼 위치에 보관하십시오. 일부 사이트는 아카이브 테이프를 원격 위치의 내화 캐비닛에 보관합니다.

각 작업(백업)을 저장하는 매체(테이프 볼륨)와 각 백업 파일의 위치를 추적하는 로그를 만들고 유지 관리해야 합니다.

# 색인

---

## 번호와 기호

/export/home 디렉토리, 321  
1394(FireWire) 지원, 설명, 113  
4.3 Tahoe 파일 시스템, 315  
9660 DVD 형식, 36

## A

autofs, 324–325

## B

BSD Fat Fast 파일 시스템, 315

## C

cdwr 명령  
  데이터 CD 및 DVD와 오디오 CD에  
  쓰기(개요), 45  
  설명, 43  
  액세스 제한(방법), 46  
cfgadm  
  PCI 핫 플러그 기능 사용(개요), 78  
  SCSI 핫 플러그 기능 사용(개요), 78  
cpio 명령  
  (개요), 355  
  테이프에 있는 파일 나열(방법), 362  
  테이프에서 모든 파일 추출(방법), 363  
  파일 시스템 간에 디렉토리 복사(방법), 356  
CTFS 파일 시스템, 319

## D

datadm 명령, 172  
/dev/dsk 디렉토리, 73  
/dev/rdisk 디렉토리, 73  
/dev 파일 시스템, 설명, 58  
devfsadm 명령, 72  
DOS, 파일 시스템, 316  
driver not attached 메시지, 63  
dumpadm 명령, 74  
DVD, ISO 9660 형식, 36  
DVD-RAM, 파일 시스템 만들기(방법), 32  
DVD-ROM, 317

## E

EFI 레이블  
  (개요), 187  
  VTOC 레이블과 비교, 187  
  문제 해결, 189  
  시스템 설치, 189  
  제한 사항, 188  
eject 명령, 이동식 매체(방법), 39

## F

FDFS 파일 시스템, 319  
FIFOFS 파일 시스템, 319  
format 유틸리티  
  (개요), 191  
  analyze(분석) 메뉴, 307

## format 유틸리티 (계속)

- defect(결함) 메뉴, 309-310
  - fdisk 메뉴, 306
  - partition(분할 영역) 메뉴, 306
  - Solaris fdisk 분할 영역 만들기(방법), 236
  - 기능 및 이점, 191
  - 기본 메뉴, 304
  - 도움말 기능 사용, 311
  - 디스크 슬라이스 정보 표시(예), 205
  - 디스크가 포맷되었는지 확인(방법), 202
  - 디스크에 레이블 지정
    - 예, 208
  - 명령 이름 입력(방법), 310
  - 사용 시기, 192
  - 사용 지침, 193
  - 손상된 디스크 레이블 복구(방법), 213
  - 시스템의 디스크 식별, 200
  - 시스템의 디스크 식별(예), 201
  - 입력, 310, 311
- Free Hog 슬라이스, **참조** 기증자 슬라이스
- fsstat 명령, 설명, 313
- fsstat 명령(예), 313
- fstypes 파일, 325
- fuser 명령
- 이동식 매체가 사용 중인지 확인(방법), 39
  - 이동식 매체에 액세스 중인 프로세스 종료(방법), 39

## G

## GRUB

- GRUB를 사용하여 디스크 관리
  - x86, 184

## H

- hald, 설명, 22
- High Sierra 파일 시스템, 316
- /home(자동 마운트됨), 324
- HSFS, **참조** High Sierra 파일 시스템

## I

## InfiniBand 장치

- HCA의 커널 IB 클라이언트 표시(방법), 166
- IB P\_key 테이블 업데이트(방법), 168
- IB 의사 장치 구성(방법), 165
- IB 의사 장치 구성 해제(방법), 165
- IB 통신 서비스 표시(방법), 169
- IB 포트, HCA\_SVC 또는 VPPA 구성 해제(방법), 164
- IB 포트, HCA\_SVC 또는 VPPA 장치 구성(방법), 164
- VPPA 통신 서비스 추가(방법), 169
- 개요, 157
- 기존 IB 포트, HCA\_SVC 또는 VPPA 통신 서비스 제거(방법), 170
- 동적 재구성(개요), 161
- 표시(방법), 162

Internet Storage Name Service, **참조** iSNS

## iSCSI

- (개요), 241
- iSCSI 구성 모니터(방법), 276
- iSCSI 구성 문제 해결(방법), 283
- iSCSI 디스크 액세스(방법), 255
- iSCSI 시작 프로그램 및 대상 매개변수 수정(방법), 280
- 검색된 iSCSI 대상 제거(방법), 253
- 단방향 또는 양방향 CHAP 인증 구성(방법), 268
- 일반적인 iSCSI 오류 메시지, 286
- 정적 및 동적 대상 검색, 246

## iscsiadm list, iSCSI 구성 정보 표시(예), 276

## iscsiadm modify 명령

- CHAP 사용(예), 269
- 정적 대상이나 동적 대상 사용 또는 사용 안함(예), 253

## iscsiadm remove 명령, 정적 대상이나 동적 대상 제거(예), 253

## iSNS

- 기본 검색 도메인, 292
- 기술 개요, 291
- 노드, 292
- 대상 관리, 299-301
- 대상 상태 표시, 300
- 시작 프로그램 관리, 299-301
- 시작 프로그램 상태 표시, 300

**iSNS (계속)**

- 클라이언트, 292
- 클라이언트 관리, 299-301
- 클라이언트 상태 표시, 300

**iSNS 서버****검색 도메인**

- 만들기, CLI, 297
- 상태 표시, CLI, 299
- 제거, CLI, 300-301
- 추가, CLI, 297-298
- 클라이언트 제거, 300
- 클라이언트 할당, CLI, 298-299

**검색 도메인 세트**

- 만들기, CLI, 296-297
- 사용 안함, CLI, 301
- 상태 표시, CLI, 299
- 제거, 301

**구성, 293****구성, CLI, 296-299****기본 검색 도메인 세트 사용, 296****데이터 저장소 위치 지정, 291, 295****서버 상태 알림 사용, 사용 안함, 294-295****설정 표시, 296****재시도 임계값 설정, 295****ISO 9660 파일 시스템, 316****ISO 표준, 9660 DVD 형식, 36****K****/kernel/drv 디렉토리, 62****M****media was found 메시지, 29****mkisofs 명령, 데이터 CD 또는 DVD 파일 시스템****만들기(방법), 48****MNTFS 파일 시스템, 321****mnttab 파일, 322****mt 명령, 371****N****NAMEFS 파일 시스템, 319****NFS****vfstab 항목, 337****서버 설명, 323****설명, 323****nfsd 데몬****시작, 41****실행 중인지 확인, 40****no media was found 메시지, 30****O****OBJFS 파일 시스템, 319****/opt 디렉토리, 321****P****PCFS 파일 시스템, 316****PCI 장치****PCI 구성 문제 해결, 99****PCI 슬롯 구성 정보 표시(방법), 94****PCI 어댑터 카드 제거(방법), 96****PCI 어댑터 카드 추가(방법), 97****PCIe(PCI Express) 지원, 설명, 58****/proc 디렉토리, 318, 321****PROCFS 파일 시스템, (개요), 318****prtvtoc 명령, 74****(예), 212****R****RCM 스크립트****개요, 102****명령, 102****rmvolmgr, 설명, 22****Rock Ridge 확장(HFSFS 파일 시스템), 316**

**S**

## SATA 장치

- SATA 장치 구성(방법), 100
- SATA 장치 구성 해제(방법), 100
- SATA 장치 정보 표시(방법), 99

## SCSI 장치

- cfgadm 명령을 사용하여 연결 해제(방법), 86
  - SCSI 구성 문제 해결, 92
  - SCSI 장치 구성(방법), 86
  - SCSI 장치 제거(방법), 91
  - SCSI 장치를 SCSI 버스에 추가(방법), 88
  - SCSI 제어기 구성(방법), 85
  - SCSI 제어기 구성 해제(방법), 85
  - SCSI 제어기 연결(방법), 87
  - SCSI 제어기에서 동일한 SCSI 장치 교체(방법), 90
  - 관련 정보 표시(방법), 84
  - 실패한 SCSI 구성 해제 작업 해결(방법), 93
- SCSI 테이프 드라이브, 369
- share 명령, 324
- 다른 시스템에서 사용할 수 있도록 이동식 매체 설정(방법), 40
- shareall 명령, 324
- SHAREFS 파일 시스템, 319
- Solaris fdisk 분할 영역, 지침, 235-236
- SPECFS 파일 시스템, 319
- SunOS 기본 파일 시스템, 320
- svcadm disable, (예), 24
- swap 명령, 74
- swapadd 명령, 347
- SWAPFS 파일 시스템, 319
- sysdef 명령, 64

**T**

## tar 명령

- (개요), 357
  - dd 명령을 사용하여 원격 테이프에서 파일 검색(방법), 366
  - dd 명령을 사용하여 파일을 원격 테이프에 복사(방법), 365
  - 테이프에 있는 파일 나열(방법), 358
  - 테이프에서 파일 검색(방법), 359
- /tmp 디렉토리, 318, 321

TMPFS 파일 시스템, 개요, 317

**U**

## uDAPL

- (개요), 170
  - DAT 정적 레지스트리 업데이트(방법), 172
  - DAT 정적 레지스트리에 서비스 공급자 등록(방법), 172
  - DAT 정적 레지스트리에서 서비스 공급자 등록 해제(방법), 172
  - 사용(방법), 171
- UDF 파일 시스템, 316
- UFS 파일 시스템, 315
- /etc/vfstab로 마운트, 337
  - 마운트, 337
- UNIX 파일 시스템, 315

## USB 장치

- (개요), 115
- 2.0 장치 개요, 120
- EHCI, OHCI 및 UHCI 지원, 119
- hotpluggable 속성, 112
- IAD(Interface Association Descriptor) 지원, 112
- Oracle Solaris USB(USB 아키텍처), 119
- Prolific 및 Keyspan 직렬 어댑터 지원, 118
- USB 2.0 지원, 120
- USB 대용량 저장 장치 제거(방법), 135
- USB 대용량 저장 장치 추가(방법), 133
- USB 장치 구성(방법), 154
- USB 장치 재설정(방법), 155
- USB 장치 정보 표시(방법), 136
- USB 카메라 추가(방법), 133
- 가상 키보드 및 마우스 지원, 113
- 기본 오디오 장치 식별(방법), 149
- 논리적으로 USB 장치 연결(방법), 155
- 논리적으로 USB 장치 연결 해제(방법), 154
- 논리적으로 USB 장치 하위 트리 연결 해제(방법), 155
- 대용량 저장 장치 마운트 또는 마운트 해제(방법), 144
- 대용량 저장 장치 마운트(방법), 144
- 대용량 저장 장치 마운트 해제(방법), 144
- 대용량 저장 장치 문제 해결 팁, 145



**USB 장치 (계속)**

- 대용량 저장 장치에 파일 시스템 만들기(방법), 137
- 대용량 저장 장치에서 Solaris 분할 영역 만들기 및 슬래이스 수정(방법), 142
- 대용량 저장 장치에서 분할 영역 수정 및 PCFS 파일 시스템 만들기(방법), 139
- 드라이버, 117
- 디스켓 장치(개요), 130
- 머리글자어, 115
- 물리적 장치 계층, 116
- 버스 설명, 115
- 버스 전원 사용 장치, 122
- 버스 정보 표시(방법), 152
- 복합 장치, 116
- 비호환 대용량 저장 장치 사용(개요), 131
- 오디오
  - (개요), 147
  - 기본 장치 변경(방법), 150
  - 장치 소유권, 150
- 오디오 장치 문제 해결, 150
- 오디오 장치 추가(방법), 148
- 이동식 대용량 저장 장치(개요), 129
- 이름, 116
- 장치 구성 해제(방법), 153
- 장치 노드, 119
- 장치 클래스, 117
- 전원 관리, 124
- 조합 장치, 116
- 케이블, 124
- 키보드 및 마우스 장치, 122
- 핫 플러그 기능 사용(개요), 132
- 호스트 제어기 및 루트 허브, 123
- 휠 마우스 지원, 123

**V**

- /var 디렉토리, 321
- vfstab 파일, 325, 347
  - LOFS의 항목, 336
  - 스왑 추가, 347
  - 항목 추가(방법), 336
- vol0, 제거, 22
- volfs, 제거, 22

**Z**

- ZFS 루트 파일 시스템에 대한 installboot 명령, 223
- ZFS 루트 파일 시스템에 대한 installgrub 명령, 234
- ZFS 루트 파일 시스템에 대한 디스크 설정 연결(방법)
  - SPARC, 219
  - x86, 229
- ZFS 루트 파일 시스템의 부트 블록 설치(방법), SPARC, 223
- ZFS 파일 시스템, 315
  - 디스크 설정(방법)
    - x86, 234
- ZFS 파일 시스템에 사용할 디스크 설정(방법)
  - SPARC, 224

**가**

- 가상 메모리 저장소, 정의, 343

**개**

- 개요, iSNS 기술, 291

**검**

- 검사, 이동식 매체에서 파일 시스템(방법), 32
- 검색, tar 명령을 사용하여 테이프에서 파일(방법), 359
- 검색 도메인
  - iSNS
    - 만들기, CLI, 297
    - 상태 표시, CLI, 299
    - 추가, CLI, 297-298
    - 클라이언트 할당, CLI, 298-299
  - 제거, 300-301
  - 클라이언트
    - 제거, CLI, 300

검색 도메인 세트

- iSNS
  - 만들기, CLI, 296-297
  - 상태 표시, CLI, 299
- 사용 안함
  - 상태 표시, CLI, 301
- 제거, 301

공

- 공유, 파일, 323-324

관

- 관리
  - GRUB를 사용하여 디스크
    - x86, 184
  - iSNS, 300
    - 클라이언트, CLI, 299-301

교

- 교체, SCSI 제어기에서 동일한 SCSI 장치(방법), 90

구

- 구성
  - IB 의사 장치(방법), 165
  - IB 포트, HCA\_SVC 또는 VPPA 장치(방법), 164
  - iSCSI에 대해 단방향 또는 양방향 CHAP
    - 인증(방법), 268
  - iSNS 서버, 293, 296-299
  - SATA 장치(방법), 100
  - SCSI 장치(방법), 86
  - SCSI 제어기(방법), 85
  - USB 장치(방법), 154
- 구성 해제
  - IB 의사 장치(방법), 165
  - IB 포트, HCA\_SVC 또는 VPPA 장치(방법), 164
  - SATA 장치(방법), 100
  - SCSI 제어기(방법), 85

구성 해제 (계속)

- USB 장치(방법), 153

기

- 기본
  - SunOS 파일 시스템, 320
  - /tmp의 파일 시스템(TMPFS), 318
- 기본 검색 도메인, iSNS, 292
- 기본 검색 도메인 세트, 사용, 296

꺼

- 꺼내기, 이동식 매체(방법), 39

네

- 네트워크 기반 파일 시스템, 317

노

- 노드
  - iSNS, 292
  - iSNS, 관리, CLI, 299-301
  - iSNS, 상태 표시, CLI, 300

논

- 논리적 장치 이름
  - 디스크, 73
  - 이동식 매체, 76
  - 정의, 72
  - 테이프, 75

대

- 대상
  - iSNS, 관리, CLI, 299-301
  - iSNS, 상태 표시, CLI, 300

**동**

- 동적 재구성, InfiniBand 장치, 161
- 동적 재구성(개요), 78

**등**

- 등록, DAT 정적 레지스트리에 서비스 공급자(방법), 172
- 등록 해제, DAT 정적 레지스트리에서 서비스 공급자(방법), 172

**디**

- 디렉토리
  - cpio 명령을 사용하여 파일 시스템 간에 복사(개요), 355
  - /proc, 318
  - /tmp, 318
- 디스켓
  - rmformat을 사용하여 포맷(방법), 30
  - 볼륨 관리를 사용하여 로드(방법), 29
  - 파일 시스템 만들기(방법), 31
- 디스크
  - ZFS 파일 시스템 설정(개요)
    - SPARC, 218
  - ZFS 파일 시스템에 대해 설정(개요)
    - x86, 228-235
  - 손상된 디스크 레이블 복구(개요), 212
  - 손상된 디스크 레이블 복구(방법), 213
  - 포맷(개요), 193
  - 포맷 시기(개요), 202
  - 포맷되었는지 확인(방법), 203
- 디스크 기반 파일 시스템, 315
- 디스크 레이블
  - 만들기(개요), 206
  - 설명, 186
- 디스크 슬라이스
  - ZFS 루트 파일 시스템 만들기
    - SPARC, 220
  - ZFS 루트 파일 시스템(방법)
    - x86, 230
  - 정보 표시(개요), 204
  - 정의, 190

- 디스크 슬라이스 지정, 74
- 디스크 제어기, 74
- 디스크 포맷(개요), 193

**로**

- 로드, 디스켓(방법), 29

**루**

- 루트(/) 파일 시스템 또는 디렉토리, 320
- 루프백 파일 시스템(LOFS), 만들기(개요), 328

**마**

- 마운트
  - /etc/vfstab로 파일 시스템, 337
  - NFS 파일 시스템, 337
  - UFS 파일 시스템, 337
  - USB 대용량 저장 장치(방법), 144
  - 수동으로 원격 이동식 매체(예), 42
  - 이동식 매체
    - 다음과 자동 마운트 비교, 25, 26
    - 자동으로 파일 시스템, 324-325
- 마운트 지점, 정의, 322
- 마운트 테이블, 322
- 마운트 해제
  - USB 대용량 저장 장치(방법), 144

**만**

- 만들기
  - DVD-RAM에서 파일 시스템(방법), 32
  - USB 대용량 저장 장치에 파일 시스템(방법), 137
  - USB 대용량 저장 장치에서 슬라이스 수정 및 Solaris 분할 영역(방법), 142
  - ZFS 루트 파일 시스템에 사용할 디스크 슬라이스
    - SPARC, 220
  - ZFS 루트 파일 시스템에 사용할 디스크 슬라이스(방법)
    - x86, 230

**만들기 (계속)**

- 데이터 CD 또는 DVD 파일 시스템(방법), 48
- 루프백 파일 시스템(개요), 328
- 이동식 매체에서 파일 시스템(방법), 31

**매**

- 매체의 끝 감지  
cpio 명령, 355

**메**

- 메모리 저장소(가상), 정의, 343

**모**

- 모니터, iSCSI 구성(방법), 276

**문**

- 문제 해결
  - EFI 디스크 레이블, 189
  - iSCSI 구성 문제(방법), 283
  - PCI 구성 문제, 99
  - SCSI 구성 문제, 92
  - USB 대용량 저장 장치, 145
  - USB 오디오 장치 문제, 150
  - 실패한 SCSI 구성 해제 작업, 93

**물**

- 물리적 장치 이름  
정의, 72

**백**

- 백업, 장치 이름, 368–370

**버**

- 버스 지향 디스크 제어기, 75

**변**

- 변경, 기본 USB 오디오 장치(방법), 150

**복**

- 복구, 이동식 매체에서 잘못된 블록(방법), 33
- 복사
  - cpio 명령을 사용하여 개별 파일(개요), 355
  - cpio 명령을 사용하여 파일 그룹(개요), 355
  - cpio 명령을 사용하여 파일 시스템 간에 디렉토리(개요), 355

**분**

- 분할 영역(스왑), 정의, 343

**블**

- 블록 디스크 장치 인터페이스  
사용 시기, 74  
정의, 73

**사**

- 사용, uDAPL, 171
- 사용 안함으로 설정
  - 이동식 매체 서비스, 24
  - 이동식 매체 서비스(방법), 37
  - 이동식 매체에 대한 쓰기 보호, 33
- 사용으로 설정
  - 이동식 매체 서비스(방법), 37
  - 이동식 매체에 대한 쓰기 보호(방법), 33

**서**

서비스(SMF), hal 서비스, 설명, 21

**설**

## 설정

ZFS 파일 시스템에 사용할 디스크(개요)

SPARC, 218

x86, 228-235

설정 표시, iSNS 서버, 296

**수**

## 수정

iSCSI 시작 프로그램 및 대상

매개변수(방법), 280

USB 대용량 저장 장치에서 PCFS 파일 시스템  
만들기 및 분할 영역(방법), 139

**스**

스왑 분할 영역, 정의, 343

## 스왑 파일

vfstab에 추가, 347

표시, 349-350

**슬**

슬라이스(정의), 190

**시**

시작, nfsd 데몬, 41

## 시작 프로그램

iSNS, 관리, CLI, 299-301

iSNS, 상태 표시, CLI, 300

**식**

## 식별

기본 USB 오디오 장치(방법), 149

시스템의 디스크(방법), 200

장치, 64

**쓰**

쓰기, 데이터 CD 및 DVD와 오디오 CD(개요), 45

**액**

## 액세스

iSCSI 디스크(방법), 255

디스크 장치, 73

이동식 매체(방법), 38

테이프 장치, 75

**업**

## 업데이트

DAT 정적 레지스트리(방법), 172

IB P\_key 테이블(방법), 168

**연**

## 연결

SCSI 제어기(방법), 87

USB 장치, 논리적(방법), 155

## 연결 해제

SCSI 제어기(방법), 86

USB 장치, 논리적(방법), 154

USB 장치 하위 트리, 논리적(방법), 155

**오**

오류 메시지, iSCSI, 286

## 원

원시 디스크 장치 인터페이스, 73, 74

## 의

의사 파일 시스템, (개요), 317

## 이

이동식 매체

dbus 서비스, 설명, 21

DVD-RAM에서 파일 시스템 만들기(방법), 32

/media

마운트 지점, 21, 36

rmformat을 사용하여 디스켓 포맷(방법), 30

rmvolmgr 서비스, 설명, 21

관리, 이점, 25

관리 향상, 21

꺼내기(방법), 39

다른 시스템에서 매체에 액세스(예), 42

디스켓 로드(방법), 29

마운트

수동을 자동과 비교, 25, 26

매체가 사용 중인지 확인(방법), 39

새 이동식 매체 드라이브 추가(방법), 37

서비스, 사용 안함으로 설정, 24

서비스, 설명, 21

쓰기 보호 사용으로 설정(방법), 33

액세스(방법), 38

액세스 프로세스 종료(방법), 39

역 호환성 (/media), 23

원격 매체 마운트(예), 42

이동식 매체 서비스 사용으로 설정 또는 사용

안함으로 설정(방법), 37

이동식 매체에서 잘못된 블록 복구(방법), 33

파일 시스템 검사(방법), 32

파일 시스템 만들기(방법), 31

## 입

임시 파일 시스템(TMPFS), 개요, 317

## 자

자동 구성 프로세스, 62

자동 마운트, 및 /home, 324

## 장

장치, 액세스, 71

장치 드라이버

정의, 60

추가, 71

장치 이름, 백업, 368-370

## 재

재설정, USB 장치(방법), 155

## 저

저장소(가상 메모리), 정의, 343

저장소 용량(매체), 367

## 제

제거

PCI 어댑터 카드(방법), 96

SCSI 장치(방법), 91

USB 대용량 저장 장치(방법), 135

검색된 iSCSI 대상(방법), 253

기존 IB 포트, HCA\_SVC 또는 VPPA 통신

서비스(방법), 170

제한, 이동식 매체 액세스(방법), 46

## 종

종료

이동식 매체에 액세스 중인 프로세스(방법), 39

파일 시스템에 액세스 중인 모든

프로세스(방법), 340

**중**

## 중지

- 이동식 매체에 액세스 중인 프로세스 종료(방법), 39
- 파일 시스템의 모든 프로세스(방법), 340

**지**

지원되지 않는 장치, 62

**찾**

찾기, 파일 시스템 유형, 325

**추**

## 추가

- /etc/vfstab 파일에 항목(방법), 336
- PCI 어댑터 카드(방법), 97
- SCSI 장치를 SCSI 버스에(방법), 88
- USB 대용량 저장 장치(방법), 133
- USB 오디오 장치(방법), 148
- USB 카메라(방법), 133
- vfstab로 스왑, 347
- VPPA 통신 서비스(방법), 169
- 새 이동식 매체 드라이브 추가(방법), 37

**클**

## 클라이언트

- iSNS, 292
- iSNS, 관리, CLI, 299-301
- iSNS, 상태 표시, CLI, 300

**테**

## 테이프

- tar 명령을 사용하여 파일 검색(방법), 359
- 저장소 용량, 367
- 크기, 367

## 테이프 드라이브

- 되감기, 369-370
- 유지 관리, 372
- 최대 SCSI, 369
- 테이프 드라이브 유지 관리, 372
- 테이프 장치(이름 지정), 76

**파**

## 파일

- /etc/default/fs, 325
- /etc/dfs/fstypes, 325
- /proc 디렉토리, 318
- tar 명령을 사용하여 테이프에서 검색(방법), 359
- 공유, 323-324
- 파일 시스템
  - 4.3 Tahoe, 315
  - BSD Fat Fast, 315
  - CTFS, 319
  - DOS, 316
  - /export/home, 321
  - FDFS, 319
  - FIFOFS, 319
  - High Sierra, 316
  - ISO 9660, 316
  - MNTFS, 321
  - NAMEFS, 319
  - OBJFS, 319
  - /opt, 321
  - PCFS, 316
  - /proc, 321
  - PROCFS, (개요), 318
  - SHAREFS, 319
  - SPECFS, 319
  - SWAPFS, 319
  - TMPFS, 317
  - UFS, 315
  - UNIX, 315
  - /var, 321
  - ZFS, 315
  - 공유, 323-324
  - 기본 SunOS, 320
  - 네트워크 기반, 317

파일 시스템 (계속)

- 디스크 기반, 315
  - 마운트 테이블, 322
  - 만들기(개요)
    - 루프백(LOFS), 328
  - 사용 가능하도록 만들기(개요), 328
  - 액세스 중인 모든 프로세스 중지(방법), 340
  - 유형, 315
  - 유형 찾기, 325
  - 의사, (개요), 317
  - 프로세스, (개요), 318
- 파일 시스템 유형, 315

포

- 포맷, `rmformat`을 사용하여 디스켓(방법), 30

표

표시

- HCA의 커널 IB 클라이언트(방법), 166
- IB 통신 서비스(방법), 169
- InfiniBand 장치 정보(방법), 162
- PCI 슬롯 구성 정보(방법), 94
- SATA 장치(방법), 99
- SCSI 장치에 대한 정보, 84
- USB 버스 정보(방법), 152
- USB 장치 정보(방법), 136
- 디스크 슬라이스 정보(개요), 204
- 스왑 공간, 349-350
- 시스템 구성 정보, 63, 66
- 이동식 매체 사용자(방법), 39

프

- 프로세스 파일 시스템(PROCFS), 318

햇

- 햇 플러그 기능 사용  
(개요), 78

햇 플러그 기능 사용 (계속)

- PCI 어댑터 카드 제거(방법), 96
  - PCI 어댑터 카드 추가(방법), 97
  - PCI 장치(개요), 94
  - SCSI 장치 구성(방법), 86
  - SCSI 장치 구성 해제(방법), 85
  - SCSI 장치 제거(방법), 91
  - SCSI 장치를 SCSI 버스에 추가(방법), 88
  - SCSI 제어기 구성(방법), 85
  - SCSI 제어기 연결(방법), 87
  - SCSI 제어기에서 동일한 SCSI 장치  
교체(방법), 90
  - USB 장치 구성(방법), 154
  - USB 장치 구성 해제(방법), 153
  - 논리적으로 USB 장치 연결(방법), 155
  - 논리적으로 USB 장치 연결 해제(방법), 154
  - 논리적으로 USB 장치 하위 트리 연결  
해제(방법), 155
- 햇 플러스 사용, `cfgadm` 명령을 사용하여 SCSI  
제어기 연결 해제(방법), 86

해

- 해결, 실패한 SCSI 구성 해제 작업(방법), 93

확

확인

- `nfsd` 데몬이 실행 중, 40
- 마운트된 파일 시스템, 329
- 파일 시스템 유형, 325