



Solaris 10 6/06 설치 설명서: 사용자 정의 JumpStart 및 고급 설치



Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

부품 번호: 819-6294-10
2006년 6월

Copyright 2006 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 모든 권리는 저작권자의 소유입니다.

Sun Microsystems, Inc.는 본 설명서에서 설명하는 제품에 구현된 기술과 관련한 지적 재산권을 보유합니다. 특히 이러한 지적 재산권에는 하나 이상의 미국 특허 또는 미국 및 기타 국가에서 특허 출원 중인 응용 프로그램이 포함될 수 있습니다.

U.S. 정부 권한 - 상용. 정부 사용자는 Sun Microsystems, Inc. 표준 사용권 계약과 FAR의 해당 규정 및 추가 사항의 적용을 받습니다.

이 배포에는 타사에서 개발한 자료가 포함되어 있을 수 있습니다.

본 제품의 일부는 Berkeley BSD 시스템일 수 있으며 University of California로부터 라이선스를 취득했습니다. UNIX는 X/Open Company, Ltd.를 통해 독점 라이선스를 취득한 미국 및 기타 국가의 등록 상표입니다.

Sun, Sun Microsystems, Sun 로고, Solaris 로고, Java Coffee Cup 로고, docs.sun.com, Java, Solaris는 미국 및 기타 국가에서 Sun Microsystems, Inc.의 상표 또는 등록 상표입니다. 모든 SPARC 상표는 라이선스 하에 사용되며 미국 및 기타 국가에서 SPARC International, Inc.의 상표 또는 등록 상표입니다. SPARC 상표가 부착된 제품은 Sun Microsystems, Inc.가 개발한 아키텍처를 기반으로 합니다.

OPEN LOOK 및 SunTM Graphical User Interface는 Sun Microsystems, Inc.가 해당 사용자 및 라이선스 소유자를 위해 개발했습니다. Sun은 컴퓨터 업계에서 시각적 또는 그래픽 사용자 인터페이스의 개념을 연구하고 개발하는 데 있어 Xerox의 선구자적 업적을 인정합니다. Sun은 Xerox Graphical User Interface에 대한 Xerox의 비독점 라이선스를 보유하고 있으며 이 라이선스는 OPEN LOOK GUI를 구현하거나 그 외의 경우 Sun의 서면 라이선스 계약을 준수하는 Sun의 라이선스 소유자에게도 적용됩니다.

이 발행물에서 다루는 제품과 수목된 정보는 미국 수출 관리법에 의해 규제되며 다른 국가의 수출 또는 수입 관리법의 적용을 받을 수도 있습니다. 이 제품과 정보를 직간접적으로 핵무기, 미사일 또는 생화학 무기에 사용하거나 핵과 관련하여 해상에서 사용하는 것은 엄격하게 금지됩니다. 거부된 사람과 특별히 지정된 국민 목록을 포함하여 미국의 수출 금지 국가 또는 미국의 수출 제의 목록에 나와 있는 대상으로의 수출이나 재수출은 엄격하게 금지됩니다.

본 설명서는 "있는 그대로" 제공되며 상업성, 특정 목적에 대한 적합성 또는 비침해성에 대한 모든 묵시적 보증을 포함하여 모든 명시적 또는 묵시적 조건, 표현 및 보증에 대해 어떠한 책임도 지지 않습니다. 이러한 보증 부인은 법적으로 허용된 범위 내에서만 적용됩니다.

목차

머리말	11
제1부 Solaris 설치 또는 업그레이드의 총괄적 계획	15
1 Solaris 설치의 새로운 기능	17
Solaris 설치 시 Solaris 10 1/06 릴리스의 새로운 기능	17
비전역 영역이 설치되어 있는 경우 Solaris OS 업그레이드	17
x86: GRUB 기반 부트	18
Solaris 릴리스의 업그레이드 지원 변경 사항	19
Solaris 설치 시 Solaris 10 3/05 릴리스의 새로운 기능	20
Solaris 설치 변경 - 설치 통합 포함	20
사용자 정의 JumpStart 설치 패키지 및 패치 향상	21
설치 중 복수 네트워크 인터페이스 구성	21
SPARC: 64비트 패키지 변경 사항	22
새로운 부트 환경을 만드는 사용자 정의 JumpStart 설치 방법	22
축소 네트워크 소프트웨어 그룹	22
가상 목차를 사용하여 디스크 분할 영역 테이블 수정	23
x86: 기본 부트 디스크 분할 영역 레이아웃의 변경	23
2 Solaris 설치 및 업그레이드(로드맵)	25
작업 맵: Solaris 소프트웨어 설치 또는 업그레이드	25
네트워크, DVD 또는 CD 중 어떤 것을 이용해 설치할 것인가?	27
초기 설치인가 업그레이드인가?	28
초기 설치	28
업그레이드	28
Solaris 설치 방법 선택	29
Sun Java System Application Server Platform Edition 8	31

3 Solaris 설치 및 업그레이드(계획)	33
시스템 요구 사항 및 권장사항	33
디스크 할당 및 스왑 공간	34
일반적 디스크 공간 계획 수립 및 권장 사항	34
소프트웨어 그룹의 권장 디스크 공간	36
업그레이드 계획	38
업그레이드 제한 사항	39
업그레이드 프로그램	39
업그레이드 대신 Solaris Flash 아카이브 설치	40
디스크 공간 재할당으로 업그레이드	40
업그레이드할 때 Patch Analyzer 사용	41
업그레이드하기 전에 시스템 백업	42
시스템이 실행 중인 Solaris OS의 버전을 확인하는 방법	42
로케일 값	42
플랫폼 이름 및 그룹	43
시스템에 Solaris 영역이 설치된 경우 업그레이드	43
비전역 영역으로 업그레이드	44
비전역 영역의 디스크 공간 요구 사항	45
x86: 분할 권장사항	46
기본 부트 디스크 분할 영역 레이아웃이 서비스 분할 영역을 보존	46
4 x86: Solaris 설치를 위한 GRUB 기반 부트	49
x86: GRUB 기반 부트(개요)	49
x86: GRUB 기반 부트 방법	50
x86: GRUB 장치 이름 지정 규약	50
x86: GRUB 기반 설치에 관한 정보의 위치	51
x86: GRUB 기반 부트(계획)	52
x86: 네트워크에서 GRUB 기반 설치 수행	52
GRUB 주 메뉴 설명	53
x86: GRUB 메뉴의 menu.lst 파일 찾기(작업)	56
▼ GRUB 메뉴의 menu.lst 파일 찾기	56
▼ 활성화 menu.lst 파일이 다른 부트 환경에 있는 경우 GRUB 메뉴의 menu.lst 파일 찾기	57
▼ Solaris Live Upgrade 부트 환경이 마운트된 경우 GRUB 메뉴의 menu.lst 파일 찾기	58
▼ 시스템에 x86 부트 분할 영역이 있는 경우 GRUB 메뉴의 menu.lst 파일 찾기	58

제2부 사용자 정의 JumpStart 사용	61
5 사용자 정의 JumpStart(개요)	63
사용자 정의 JumpStart 소개	63
사용자 정의 JumpStart 예제 시나리오	63
JumpStart 프로그램을 사용하여 Solaris 소프트웨어를 설치하는 방법	64
6 사용자 정의 JumpStart 설치 준비(작업)	69
작업 맵: 사용자 정의 JumpStart 설치 준비	69
네트워크 시스템에 대한 프로필 서버 만들기	71
▼ 서버에 JumpStart 디렉토리 만들기	71
프로필 서버에 모든 시스템 액세스 허용	72
▼ 모든 시스템에 프로필 서버 액세스 허용	73
독립형 시스템에 대한 프로필 디스켓 만들기	75
▼ SPARC: 프로필 디스켓 만들기	75
▼ x86: GRUB를 사용하여 프로필 디스켓 만들기	76
rules 파일 만들기	78
rules 파일의 구문	78
▼ rules 파일 만들기	79
rules 파일의 예	80
프로필 작성	81
프로필의 구문	82
▼ 프로필 작성	82
프로필의 예	83
프로필 테스트	93
▼ 프로필을 테스트하는 임시 Solaris 환경 만들기	94
▼ 프로필 테스트	95
프로필 테스트의 예	96
rules 파일 검증	97
▼ rules 파일 검증	98
7 사용자 정의 JumpStart 선택적 기능 사용(작업)	99
시작 스크립트 만들기	99
시작 스크립트에 관한 주요 정보	100
시작 스크립트를 사용하여 파생된 프로필 작성	100
종료 스크립트 만들기	101

종료 스크립트에 관한 주요 정보	101
▼ 종료 스크립트를 사용하여 파일 추가	101
종료 스크립트를 사용하여 패키지 또는 패치 추가	102
종료 스크립트를 사용하여 루트 환경 사용자 정의	104
종료 스크립트를 사용하여 시스템의 루트 비밀번호 설정	105
종료 스크립트를 사용하여 비대화식 설치	107
압축된 구성 파일 만들기	107
▼ 압축된 구성 파일 만들기	107
압축된 구성 파일의 예	108
디스크 구성 파일 만들기	108
▼ SPARC: 디스크 구성 파일을 만들기	108
SPARC: 디스크 구성 파일 예	109
▼ x86: 디스크 구성 파일을 만들기	110
x86: 디스크 구성 파일 예	111
사이트별 설치 프로그램 사용	113
8 사용자 정의 규칙 및 검사 키워드 만들기(작업)	115
검사 키워드	115
custom_probes 파일 만들기	116
custom_probes 파일의 구문	116
custom_probes의 함수 이름 구문	116
▼ custom_probes 파일 만들기	117
custom_probes 파일 및 키워드의 예	117
custom_probes 파일 검증	118
▼ custom_probes 파일 검증	119
9 사용자 정의 JumpStart 설치 수행(작업)	121
SPARC: 작업 맵: 사용자 정의 JumpStart 설치를 위한 시스템 설정	121
SPARC: 사용자 정의 JumpStart 설치 수행	122
▼ 사용자 정의 JumpStart 설치로 Solaris Flash 아카이브 설치 준비하기	122
▼ SPARC: 사용자 정의 JumpStart 프로그램을 사용하여 설치 또는 업그레이드 수행하기	125
SPARC: boot 명령어의 명령어 참조	126
x86: 작업 맵: 사용자 정의 JumpStart 설치를 위한 시스템 설정	127
x86: 사용자 정의 JumpStart 설치 수행	128
▼ x86: 사용자 정의 JumpStart 프로그램 및 GRUB를 사용하여 설치 또는 업그레이드 수행	129

x86: GRUB 부트 명령어를 편집하여 사용자 정의 JumpStart 설치 수행	131
▼ x86: GRUB 부트 명령어를 수정하려면	131
x86: 시스템 부트에 대한 명령어 참조	132
10 사용자 정의 JumpStart로 설치(예)	135
예제 사이트 설정	135
설치 서버 만들기	136
x86: 마케팅 시스템용 부트 서버 만들기	138
JumpStart 디렉토리 만들기	139
JumpStart 디렉토리 공유	139
SPARC: 엔지니어링 그룹의 프로필 작성	139
x86: 마케팅 그룹의 프로필 작성	140
rules 파일 업데이트	140
rules 파일 검증	141
SPARC: 네트워크에서 설치할 엔지니어링 시스템 설정	141
x86: 네트워크에서 설치할 마케팅 시스템 설정	142
SPARC: 엔지니어링 시스템 부트 및 Solaris 소프트웨어 설치	143
x86: 마케팅 시스템 부트 및 Solaris 소프트웨어 설치	143
11 사용자 정의 JumpStart(참조)	145
규칙 키워드 및 값	145
프로필 키워드 및 값	149
프로필 키워드 빠른 참조	149
프로필 키워드에 대한 설명 및 예	151
비전역 영역으로 업그레이드하는 경우 프로필 키워드 제한	186
사용자 정의 JumpStart 환경 변수	187
검사 키워드 및 값	189
제3부 RAID-1 볼륨 사용	191
12 설치 중 RAID-1볼륨(미러) 만들기(개요)	193
RAID-1 볼륨을 사용하는 이유	193
RAID-1 볼륨의 작동 방법	194
Solaris 볼륨 관리자 구성요소의 개요	196
상태 데이터베이스 및 상태 데이터베이스 복제본	196

RAID-0 볼륨(연결)	197
RAID-1 볼륨(미러)	198
RAID-1 볼륨 디스크 레이아웃의 예	198
13 설치 중 RAID-1볼륨(미러) 만들기(계획)	201
시스템 요구 사항	201
상태 데이터베이스 복제본 지침 및 요구 사항	201
상태 데이터베이스 복제본에 대한 슬라이스 선택	202
상태 데이터베이스 복제본 수 선택	202
제어기에 상태 데이터베이스 복제본 분산	203
RAID-1 및 RAID-0 볼륨 요구 사항 및 지침	203
사용자 정의 JumpStart 및 Solaris Live Upgrade 지침	203
디스크 및 제어기 선택 지침	206
슬라이스 선택 지침	206
단일 사용자 모드로 부트가 RAID-1 볼륨에 미치는 영향	207
제4부 부록	209
A 문제 해결(작업)	211
네트워크 설치 설정 문제	211
시스템 부트 관련 문제	211
매체에서 부트할 때 나타나는 오류 메시지	211
매체에서 부트할 때 발생하는 일반적인 문제	213
네트워크에서 부트할 때 나타나는 오류 메시지	213
네트워크에서 부트할 때 발생하는 일반적인 문제	216
Solaris OS의 초기 설치	217
▼ x86: IDE 디스크의 잘못된 블록 확인	217
Solaris OS 업그레이드	219
업그레이드할 때 나타나는 오류 메시지	219
업그레이드할 때 발생하는 일반적인 문제	220
▼ 업그레이드 실패 후에 업그레이드 계속 진행	221
x86: GRUB를 사용하는 경우의 Solaris Live Upgrade 문제	222
▼ Veritas VxVm을 실행하는 Solaris Live Upgrade를 사용하여 업그레이드하는 중에 시스템 장애 발생	223
x86: 기존 Service 분할 영역이 없는 시스템에서 기본적으로 만들지 않는 Service 분할 영역	225

▼ 네트워크 설치 이미지 또는 Solaris Operating System DVD에서 소프트웨어를 설치하는 방법	226
▼ Solaris Software - 1 CD 또는 네트워크 설치 이미지에서 설치하는 방법	226
B 추가 SVR4 패키징 요구 사항(참조)	229
현재 OS의 수정 방지	229
절대 경로 사용	229
pkgadd -R 명령어 사용	230
\$PKG_INSTALL_ROOT와 \$BASEDIR 사이의 차이점 개요	230
스크립트 작성 지침	231
디스크 없는 클라이언트 호환성 유지관리	231
패키지 확인	232
설치 또는 업그레이드 시 사용자 상호 작용 방지	233
영역에 대한 패키지 매개 변수 설정	234
배경 정보	237
용어집	239
색인	253

머리말

이 설명서에서는 네트워크에 연결되거나 연결되지 않은 SPARC® 및 x86 구조 기반 시스템에서 Solaris™ OS를 설치 및 업그레이드하는 방법에 대해 설명합니다. 이 설명서는 JumpStart 설치 방법 사용과 설치 중 RAID-1 볼륨 작성에 대해 다룹니다.

이 책에서는 시스템 하드웨어 또는 기타 주변 기기를 설정하는 방법에 대해서는 설명하지 않습니다.

주 - 본 Solaris 릴리스는 프로세서 아키텍처의 SPARC 및 x86 제품군을 사용하는 UltraSPARC®, SPARC64, AMD64, Pentium 및 Xeon EM64T 시스템을 지원합니다. 지원되는 시스템은 <http://www.sun.com/bigadmin/hcl>의 **Solaris 10 Hardware Compatibility List**를 참조하십시오. 이 문서에서는 플랫폼 유형에 따른 구현 차이가 있는 경우 이에 대하여 설명합니다.

다음은 이 문서에서 사용하는 주요 x86 관련 용어 설명입니다.

- “x86”은 64비트 및 32비트 x86 호환 제품의 큰 제품군을 의미합니다.
- “x64”는 AMD 64 또는 EM64T 시스템에 대한 특정 64비트 정보를 나타냅니다.
- “32비트 x86”은 x86 기반 시스템에 대한 특정 32비트 정보를 나타냅니다.

지원되는 시스템을 보려면 **Solaris 10 Hardware Compatibility List**를 참조하십시오.

본 설명서의 대상

이 책은 Solaris OS 설치를 담당하는 시스템 관리자를 대상으로 합니다. 이 책은 다음 유형의 정보를 모두 제공합니다.

- 네트워크에 연결된 환경에서 복수 Solaris 시스템을 관리하는 기업 시스템 관리자를 위한 고급 Solaris 설치 정보
- Solaris 설치 또는 업그레이드를 수행하는 시스템 관리자를 위한 기본적인 Solaris 설치 정보

관련 설명서

표 P-1은 Solaris 소프트웨어를 설치할 때 필요한 관련 정보를 나열합니다.

표 P-1 관련 정보

정보	설명
Solaris 10 6/06 설치 설명서: 기본 설치	그래픽 사용자 인터페이스(GUI)를 사용하는 기본 OS 설치에 대해 설명합니다.
Solaris 10 6/06 설치 설명서: 네트워크 기반 설치	LAN 또는 WAN을 통해 원격 Solaris 설치를 수행하는 방법에 대해 설명합니다.
Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획	CD 또는 DVD 매체를 사용하여 Solaris OS로 시스템을 업그레이드할 때 계획 정보를 제공합니다. 이 설명서는 Solaris Live Upgrade를 사용하여 새 부트 환경을 만들고 업그레이드하는 방법에 대해서도 설명합니다.
Solaris 10 6/06 설치 설명서: Solaris Flash 아카이브(작성 및 설치)	Solaris Flash 아카이브 작성 및 Solaris Flash 아카이브를 사용하여 여러 시스템에 Solaris OS를 설치하는 방법에 대한 지침을 제공합니다.
System Administration Guide: Devices and File Systems	시스템 파일 및 다른 시스템 관리 작업을 백업하는 방법에 대해 설명합니다.
Solaris 릴리스 노트	모든 버그, 알려진 문제, 중지되어 있는 소프트웨어 및 Solaris 릴리스에 관련된 패치를 설명합니다.
SPARC: http://docs.sun.com 의 Solaris Sun 하드웨어 플랫폼 안내서	지원되는 하드웨어에 대한 정보를 포함합니다.
Solaris 패키지 목록	Solaris OS의 패키지를 나열하고 설명합니다.
x86: Solaris 하드웨어 호환성 목록	지원되는 하드웨어 정보 및 장치 구성을 포함합니다.

설명서, 지원 및 교육

Sun 웹 사이트는 다음의 추가 자원에 대한 정보를 제공합니다.

- 설명서 (<http://www.sun.com/documentation/>)
- 지원 (<http://www.sun.com/support/>)
- 교육 (<http://www.sun.com/training/>)

표기 규약

다음 표는 이 책에서 사용된 표기 규약에 대해 설명합니다.

표 P-2 표기 규약

서체	의미	예
AaBbCc123	명령어 및 파일, 디렉토리 이름; 또는 컴퓨터 화면에 출력되는 내용입니다.	.login 파일을 편집하십시오. 모든 파일 목록을 보려면 <code>ls -a</code> 명령어를 사용하십시오. machine_name% you have mail.
AaBbCc123	사용자가 입력하는 내용으로 컴퓨터 화면의 출력 내용과 반대입니다.	machine_name% su Password:
<i>aabbcc123</i>	자리 표시자: 실제 이름이나 값으로 대체됩니다.	파일을 삭제하려면 <code>rm filename</code> 을 입력하십시오.
<i>AaBbCc123</i>	책 제목, 새로 나오는 단어나 용어, 강조 표시할 단어입니다.	사용자 설명서의 6장을 읽으십시오. 캐시는 로컬로 저장된 복사본입니다. 파일을 저장하면 안 됩니다. 참고: 일부 강조된 항목은 온라인에서 굵은체로 나타납니다.

명령어 예의 셸 프롬프트

다음 표에서는 C 셸, Bourne 셸 및 Korn 셸에 대한 기본 UNIX® 시스템 프롬프트 및 슈퍼유저 프롬프트를 보여 줍니다.

표 P-3 셸 프롬프트

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

1

Solaris 설치 또는 업그레이드의 총괄적 계획

이 부분에서는 설치 프로그램을 사용하는 경우의 Solaris 운영 체제 설치 또는 업그레이드 계획을 안내합니다.

Solaris 설치의 새로운 기능

이 장에서는 Solaris 설치 프로그램의 새로운 기능을 설명합니다. 모든 Solaris OS의 기능을 보려면 **Solaris 10** 새로운 기능을 참조하십시오.

- 17 페이지 “Solaris 설치 시 Solaris 10 1/06 릴리스의 새로운 기능”
- 20 페이지 “Solaris 설치 시 Solaris 10 3/05 릴리스의 새로운 기능”

Solaris 설치 시 Solaris 10 1/06 릴리스의 새로운 기능

이 절에서는 Solaris 10 1/06 릴리스의 다음과 같은 새로운 설치 기능을 설명합니다.

비전역 영역이 설치되어 있는 경우 Solaris OS 업그레이드

Solaris 영역 분할 기술을 사용하여 전역 영역인 Solaris의 단일 인스턴스에서 비전역 영역을 구성할 수 있습니다. 비전역 영역은 다른 모든 영역에서 프로세스가 격리되는 응용 프로그램 실행 환경입니다. **Solaris 10 1/06** 릴리스부터 비전역 영역이 설치된 시스템을 실행하는 경우에는 표준 Solaris 업그레이드 프로그램을 사용하여 업그레이드할 수 있습니다. Solaris 대화식 설치 프로그램 또는 사용자 정의 JumpStart를 사용하여 업그레이드를 수행할 수 있습니다. 비전역 영역이 설치된 경우에는 업그레이드에 몇 가지 제한이 적용됩니다.

- 제한된 수의 사용자 정의 JumpStart 키워드가 지원됩니다. 지원되는 사용자 정의 JumpStart 키워드의 목록은 **Solaris 10 6/06** 설치 설명서: 사용자 정의 **JumpStart** 및 고급 설치를 참조하십시오.
- CD-ROM은 배포되지 않지만 DVD-ROM 또는 네트워크 설치 이미지를 사용하여 업그레이드할 수 있습니다.
- 비전역 영역이 설치된 시스템에서는 Solaris Live Upgrade를 사용하여 시스템을 업그레이드하지 마십시오. lucreate 명령어를 사용하면 부트 환경을 만들 수 있지만 비전역 영역이 설치된 부트 환경은 luupgrade 명령어로 업그레이드할 수 없습니다. 이 경우에는 업그레이드가 실패하고 오류 메시지가 표시됩니다.

Solaris 대화식 설치 프로그램에 대한 자세한 내용은 **Solaris 10 6/06 설치 설명서: 기본 설치를 참조하십시오.**

x86: GRUB 기반 부트

Solaris 10 1/06 릴리스부터 시작하여 Solaris OS for x86 기반 시스템에 오픈 소스 GNU GRUB(Grand Unified Boot Loader)가 사용됩니다. GRUB는 시스템의 메모리에 부트 아카이브를 로드합니다. 부트 아카이브는 루트(/) 파일 시스템을 마운트하기 전에 시스템을 시작하는 동안 필요한 중요 파일의 모음입니다. 부트 아카이브는 Solaris OS 부트에 사용됩니다.

가장 주목할 만한 변경 내용은 Solaris Device Configuration Assistant가 GRUB 메뉴로 바뀐 것입니다. GRUB 메뉴를 사용하면 시스템에 설치된 여러 운영 체제를 부트할 수 있습니다. GRUB 메뉴는 x86 기반 시스템을 부트하면 표시됩니다. GRUB 메뉴에서는 화살표 키를 사용하여 설치할 OS 인스턴스를 선택할 수 있습니다. 선택을 수행하지 않으면 기본 OS 인스턴스가 부트됩니다.

GRUB 기반 부트 기능에서는 다음과 같은 사항이 향상되었습니다.

- 더욱 빨라진 부트 시간
- USB CD 또는 DVD 드라이브에서 설치
- USB 저장 장치에서 부트하는 기능
- 단순화된 PXE 부트용 DHCP 설정(공급업체별 옵션 없음)
- 모든 실제 모드 드라이버 제거
- Solaris Live Upgrade와 GRUB 메뉴를 사용하여 빠르게 활성화하고 부트 환경으로 폴백하는 기능

GRUB에 대한 자세한 내용은 다음 절을 참조하십시오.

작업	GRUB 작업	자세한 내용
설치	GRUB 기반 부트 관련 개요 정보	49 페이지 “x86: GRUB 기반 부트(개요)”
	GRUB 기반 부트의 설치 계획	52 페이지 “x86: GRUB 기반 부트(계획)”
	GRUB 메뉴를 사용하여 네트워크에서 부트 및 설치를 수행하는 방법	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 “DVD 이미지를 사용하여 네트워크에서 시스템 부팅 및 설치”
	GRUB 메뉴와 사용자 정의 JumpStart 설치 방법을 사용하여 부트 및 설치를 수행하는 방법	128 페이지 “x86: 사용자 정의 JumpStart 설치 수행”
시스템 관리	GRUB 메뉴와 Solaris Live Upgrade를 사용하여 활성화하고 부트 환경으로 폴백하는 방법	<ul style="list-style-type: none"> ■ Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 “부트 환경 활성화” ■ Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 10 장, “오류 복구: 원래 부트 환경으로 폴백(작업)”
	GRUB 메뉴의 menu.lst 파일 찾기	56 페이지 “x86: GRUB 메뉴의 menu.lst 파일 찾기(작업)”
	GRUB 메뉴를 사용하여 시스템 관리 작업을 수행하는 방법	<ul style="list-style-type: none"> ■ System Administration Guide: Basic Administration ■ System Administration Guide: Devices and File Systems ■ bootadm(1M) ■ installgrub(1M)

주 - GNU는 “GNU’s Not UNIX”의 재키적 머리글자입니다. 자세한 내용은 <http://www.gnu.org>를 참조하십시오.

Solaris 릴리스의 업그레이드 지원 변경 사항

Solaris 1/06 릴리스부터 Solaris 8, 9 또는 10 릴리스에서 Solaris OS를 업그레이드할 수 있습니다. Solaris 7 릴리스에서 업그레이드는 지원되지 않습니다.

Solaris 설치 시 Solaris 10 3/05 릴리스의 새로운 기능

이 절에서는 Solaris 10 3/05 릴리스의 새로운 설치 기능에 대해 설명합니다.

Solaris 설치 변경 - 설치 통합 포함

Solaris 10 3/05 릴리스부터 Solaris OS의 설치에 여러 가지 변경 내용이 적용되어 더욱 간단하며 통합된 설치 과정을 제공합니다.

변경 사항은 다음과 같습니다.

- 이 릴리스에는 설치 DVD 한 장과 설치 CD 여러 장이 있습니다. Solaris 운영 체제 DVD에는 모든 설치 CD의 내용이 포함됩니다.
 - **Solaris Software 1** - 이 CD만이 부트 가능한 CD입니다. 이 CD에서 Solaris 설치 그래픽 사용자 인터페이스(GUI) 및 콘솔 기반 설치에 액세스할 수 있습니다. 또한 이 CD를 사용하여 GUI와 콘솔 기반 설치에서 선택한 소프트웨어 제품을 설치할 수 있습니다.
 - 기타 **Solaris** 운영 체제 **CD** - 이 CD의 내용은 다음과 같습니다.
 - 필요한 경우 소프트웨어가 설치 요구 프롬프트를 표시하는 Solaris 패키지
 - 지원 및 비지원 소프트웨어를 포함하는 ExtraValue 소프트웨어
 - 설치 프로그램
 - 지역화된 인터페이스 소프트웨어 및 설명서
- Solaris Installation CD는 더 이상 제공되지 않습니다.
- CD 및 DVD 매체의 경우 GUI 설치가 기본(시스템의 메모리가 충분한 경우)입니다. 그러나 **text** 부트 옵션을 사용하여 콘솔 기반 설치를 지정할 수 있습니다.
- 설치 프로세스가 단순화 되었으며, 설치 프로세스는 부트할 때 언어 지원을 선택하고 로케일을 나중에 선택할 수 있게 되었습니다.

주 - (비대화식) Solaris 사용자 정의 JumpStart™ 설치 방법은 변경되지 않았습니다.

OS를 설치하려면 간단히 Solaris Software - 1CD 또는 DVD를 넣고 다음 명령어 중 한 가지를 입력합니다.

- 기본 GUI 설치(시스템 메모리가 충분한 경우)의 경우 **boot cdrom**을 입력하십시오.
- 콘솔 기반 설치를 시작하려면 **boot cdrom - text**를 입력합니다.

새 **text** 부트 옵션으로 CD 또는 DVD 매체를 사용하여 Solaris OS를 설치하는 방법은 다음을 참조하십시오.

Solaris 10 6/06 설치 설명서: 기본 설치

CD 매체를 사용하여 설치 서버를 설정하는 데 대한 변경 사항은 다음을 참조하십시오.

Solaris 10 6/06 설치 설명서: 네트워크 기반 설치

GUI 또는 콘솔 기반 설치 액세스

Solaris 10 3/05 릴리스부터 GUI를 사용하거나 윈도우화 환경을 사용 혹은 사용하지 않고 소프트웨어를 설치하도록 선택할 수 있습니다. 충분한 메모리가 있을 경우 GUI가 기본적으로 표시됩니다. GUI용 메모리가 부족한 경우 다른 환경이 기본으로 표시됩니다. `nowin` 또는 `text` 부트 옵션으로 기본값을 대체할 수 있습니다. 하지만 시스템의 메모리 양 또는 원격 설치에 의해 제한을 받습니다. 또한, Solaris 설치 프로그램이 비디오 어댑터를 감지하지 않으면 프로그램은 자동으로 콘솔 기반 환경으로 표시됩니다.

특정 메모리 요구 사항은 [33 페이지 “시스템 요구 사항 및 권장사항”](#)을 참조하십시오.

사용자 정의 JumpStart 설치 패키지 및 패치 향상

Solaris 10 3/05 릴리스부터 사용자 정의 JumpStart 설치 방법을 사용하여 Solaris OS를 설치 및 업그레이드하면 새 사용자 정의로 다음 작업을 수행할 수 있습니다.

- 추가 패키지를 포함하는 Solaris Flash 설치
 - 사용자 정의 JumpStart 프로필 `package` 키워드가 추가 패키지를 포함하여 Solaris Flash 아카이브를 설치할 수 있도록 개선되었습니다. 예를 들어 동일한 기본 아카이브를 두 대의 컴퓨터에 설치할 수 있으나 각 컴퓨터에서 서로 다른 패키지 집합을 추가할 수 있습니다. 이들 패키지가 Solaris OS 배포의 일부분일 필요는 없습니다.
- Solaris 배포의 일부가 아닐 수 있는 추가 패키지 포함 설치
 - `package` 키워드 또한 Solaris 배포의 일부가 아닌 패키지를 포함하여 설치할 수 있도록 개선되었습니다. 더 이상 여러분의 패키지를 추가하기 위하여 설치 후 스크립트를 작성하지 않아도 됩니다.
- Solaris OS 패치 설치 기능이 있는 설치
 - 새 사용자 정의 JumpStart 프로필 `patch` 키워드를 사용하면 Solaris OS 패치를 설치할 수 있습니다. 이 기능을 사용하면 패치 파일에 지정된 목록의 패치를 설치할 수 있습니다.

자세한 내용은 **Solaris 10 6/06** 설치 설명서: 사용자 정의 JumpStart 및 고급 설치를 참조하십시오.

설치 중복수 네트워크 인터페이스 구성

Solaris 10 3/05 릴리스부터 Solaris 설치 프로그램을 사용하면 설치 중에 여러 인터페이스를 구성할 수 있습니다. 이들 인터페이스는 시스템에 맞추어 `sysidcfg` 파일에서 미리 구성할 수 있습니다. 또는 설치 중에 여러 개의 인터페이스를 구성할 수 있습니다. 자세한 내용은 다음 설명서를 참조하십시오.

- **Solaris 10 6/06** 설치 설명서: 네트워크 기반 설치
- `sysidtool(1M)`
- `sysidcfg(4)`

SPARC: 64비트 패키지 변경 사항

이전의 Solaris 릴리스에서 Solaris 소프트웨어는 32비트 구성 요소용과 64비트 구성 요소용으로 별도로 제공되었습니다. **Solaris 10 3/05** 릴리스부터 시작하여 대부분의 32비트 구성 요소와 64비트 구성 요소를 한 패키지에 넣어 패키지를 단순화했습니다. 통합 패키지는 원래 32비트 패키지 이름을 계속 사용하고 64비트 패키지 이름은 더 이상 사용하지 않습니다.

64비트 패키지를 제거하여 다음과 같이 설치를 단순화하고 성능을 향상시켰습니다.

- 패키지의 수를 감소시켜 패키지 목록을 포함하는 사용자 정의 JumpStart 스크립트 단순화
- 소프트웨어 기능을 그룹화하는 하나의 패키지로 시스템 패키징 단순화
- 설치되는 패키지의 수가 적으므로 설치 시간 감소

64비트 패키지는 다음 규약으로 이름 변경됩니다.

- 64비트 패키지에 32비트 대응 항목이 있으면 64비트 패키지는 32비트 패키지 이름으로 이름 지정됩니다. 예를 들어 이전에는 `/usr/lib/sparcv9/libc.so.1` 등의 64비트 라이브러리가 `SUNWcslx`에 들어 있었지만 지금은 `SUNWcsl`에 들어 있습니다. 64비트 `SUNWcslx` 패키지는 더 이상 제공되지 않습니다.
- 패키지에 32비트 대응 항목이 없으면 이름에서 "x" 접미어가 제거됩니다. 예를 들어 `SUNW1394x`는 `SUNW1394`로 변경되었습니다.

이 변경은 64비트 패키지에 대한 참조를 제거하기 위해 사용자 정의 JumpStart 스크립트 또는 다른 패키지 설치 스크립트를 수정해야 할 수 있다는 뜻입니다.

새로운 부트 환경을 만드는 사용자 정의 JumpStart 설치 방법

Solaris 10 3/05 릴리스부터 이제 Solaris 운영 체제를 설치할 때 JumpStart 설치 방법을 사용하여 빈 부트 환경을 만들 수 있습니다. 그런 다음 이후 사용을 위해 빈 부트 환경을 Solaris Flash 아카이브로 채울 수 있습니다.

자세한 내용은 [11 장](#)을 참조하십시오.

축소 네트워크 소프트웨어 그룹

Solaris 10 3/05 릴리스부터 이제 설치하는 동안 축소 네트워크 소프트웨어 그룹(SUNWCrnet)을 선택 또는 지정하여 사용 설정된 네트워크 서비스의 수를 줄임으로써 더욱 안전한 시스템을 만들 수 있습니다. 시스템 관리자는 축소 네트워크 소프트웨어 그룹을 사용하여 유틸리티와 복수 사용자 텍스트 기반 콘솔을 사용할 수 있습니다. 시스템은 SUNWCrnet을 사용하여 네트워크 인터페이스를 인식합니다. 설치 중에 소프트웨어 패키지를 추가하고 필요한 네트워크 서비스를 활성화하여 시스템의 구성을 사용자 정의할 수 있습니다.

자세한 내용은 **Solaris 10 6/06** 설치 설명서: 사용자 정의 JumpStart 및 고급 설치를 참조하십시오.

가상 목차를 사용하여 디스크 분할 영역 테이블 수정

Solaris 10 3/05 릴리스부터 이제 Solaris 설치 프로그램을 사용하여 가상 목차(VTOC)에서 기존 슬라이스를 로드할 수 있습니다. 이제 설치 프로그램의 기본 디스크 레이아웃을 사용하지 않고 설치 중에 시스템의 기존 디스크 슬라이스 테이블을 보존하고 사용할 수 있습니다.

x86: 기본 부트 디스크 분할 영역 레이아웃의 변경

Solaris 10 3/05 릴리스부터 Solaris 설치 프로그램의 새 기능은 부트 디스크 분할 영역 레이아웃입니다. 기본적으로 이 레이아웃은 Sun x86 기반 시스템에서 서비스 분할 영역을 수용합니다. 이 설치 프로그램을 사용하여 기존 서비스 분할 영역을 보존할 수 있습니다.

새로운 기본값은 다음 분할 영역을 포함합니다.

- 첫 번째 분할 영역 - 서비스 분할 영역(시스템 기본 크기)
- 두 번째 분할 영역 - x86 부트 분할 영역(약 11MB)
- 세 번째 분할 영역 - Solaris 운영 체제 분할 영역(부트 디스크의 나머지 공간)

이 기본 레이아웃을 사용하려면 Solaris 설치 프로그램에서 부트 디스크 레이아웃을 선택하라는 메시지를 표시할 때 기본값을 선택합니다.

주 - 현재 서비스 분할 영역이 없는 시스템에 x86 기반 시스템의 Solaris OS를 설치하는 경우 Solaris 설치 프로그램은 새 서비스 분할 영역을 만들지 않습니다. 시스템에 서비스 분할 영역을 만들려는 경우 우선 사용하시는 시스템중의 진단 CD를 사용하여 서비스 분할 영역을 만듭니다. 서비스 분할 영역을 만든 후, Solaris 운영 체제를 설치합니다.

서비스 분할 영역을 만드는 방법은 하드웨어 설명서를 참조하십시오.

자세한 내용은 **Solaris 10 6/06** 설치 설명서: 사용자 정의 **JumpStart** 및 고급 설치를 참조하십시오.

Solaris 설치 및 업그레이드(로드맵)

이 장에서는 Solaris 운영 체제를 설치 또는 업그레이드하기 전에 필요한 의사 결정에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

- 25 페이지 “작업 맵: Solaris 소프트웨어 설치 또는 업그레이드”
- 27 페이지 “네트워크, DVD 또는 CD 중 어떤 것을 이용해 설치할 것인가?”
- 28 페이지 “초기 설치인가 업그레이드인가?”
- 29 페이지 “Solaris 설치 방법 선택”
- 31 페이지 “Sun Java System Application Server Platform Edition 8”

주 - 이 책에서는 슬라이스라는 용어를 사용하지만 일부 Solaris 설명서와 프로그램에서는 슬라이스를 분할 영역이라고 부르기도 합니다.

x86: 혼란을 피하기 위해 이 책은 x86 fdisk 분할 영역과 Solaris fdisk 분할 영역 내의 디비전을 구분합니다. x86 fdisk 디비전은 분할 영역이라고 합니다. Solaris fdisk 분할 영역 내의 디비전은 슬라이스라고 합니다.

작업 맵: Solaris 소프트웨어 설치 또는 업그레이드

다음 작업 맵은 모든 설치 프로그램을 사용하여 Solaris OS를 설치 또는 업그레이드하는 데 필요한 단계의 개요입니다. 이 작업 맵을 사용하여 운영 환경을 가장 효율적으로 설치하기 위해 필요한 모든 의사 결정을 확인합니다.

표 2-1 작업 맵: Solaris 소프트웨어 설치 또는 업그레이드

작업	설명	지침
초기 설치 또는 업그레이드를 선택합니다.	초기 설치를 수행할지 업그레이드를 수행할지 결정합니다.	28 페이지 “초기 설치인가 업그레이드인가?”

표 2-1 작업 맵: Solaris 소프트웨어 설치 또는 업그레이드 (계속)

작업	설명	지침
설치 프로그램을 선택합니다.	Solaris OS가 설치 또는 업데이트를 위해 몇 가지 프로그램을 제공합니다. 사용자 환경에 가장 적합한 설치 방법을 선택합니다.	29 페이지 “Solaris 설치 방법 선택”
(Solaris 대화식 설치 프로그램) 기본 또는 사용자 정의 설치를 선택합니다.	<p>사용자 환경에 적합한 설치 유형을 결정합니다.</p> <ul style="list-style-type: none"> ■ 그래픽 사용자 인터페이스(GUI)를 사용 중이면 기본 또는 사용자 정의 설치를 선택할 수 있습니다. <ul style="list-style-type: none"> ■ 기본 설치 는 하드 디스크를 포맷하고 미리 선택된 일부 소프트웨어를 설치합니다. ■ 사용자 정의를 통해서 는 하드 디스크 레이아웃을 수정하고 설치하려는 소프트웨어를 선택할 수 있습니다. ■ 텍스트 프로그램(비그래픽 인터페이스)을 사용할 경우 기본값을 선택하거나 값을 편집하여 설치하려는 소프트웨어를 선택할 수 있습니다. 	Solaris 설치 프로그램의 선택 사항에 대한 자세한 내용은 Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 5 장, “설치 또는 업그레이드 전 정보 수집(계획)”을 참조하십시오.
시스템 요구 사항을 검토합니다. 디스크 공간과 스왑 공간을 계획 및 할당합니다.	시스템이 설치나 업그레이드를 위한 최소 요구 사항을 충족하는지 확인합니다. 설치할 Solaris OS 구성 요소의 디스크 공간을 시스템에서 할당합니다. 시스템에 적합한 스왑 공간 레이아웃을 결정합니다.	3 장.
시스템을 로컬 매체에서 설치할지 네트워크에서 설치할지 선택합니다.	사용자 환경에 가장 적합한 설치 매체를 결정합니다.	27 페이지 “네트워크, DVD 또는 CD 중 어떤 것을 이용해 설치할 것인가?”
시스템 관련 정보를 수집합니다.	<ul style="list-style-type: none"> ■ Solaris 설치 프로그램의 경우 워크시트를 작성하여 설치 또는 업그레이드하는 데 필요한 모든 정보를 수집합니다. ■ 사용자 정의 JumpStart 설치 방법의 경우 프로필에서 사용할 프로필 키워드를 결정합니다. 그런 다음 키워드 설명을 검토하여 필요한 시스템에 대한 정보를 찾습니다. 	<ul style="list-style-type: none"> ■ Solaris 설치 프로그램의 경우 다음 문서를 참조하십시오. <ul style="list-style-type: none"> ■ 초기 설치의 경우: Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 “설치를 위한 검사 목록” ■ 업그레이드의 경우: Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 4 장, “업그레이드 전 정보 수집(계획)” ■ 사용자 정의 JumpStart 설치 방법은 11 장을 참조하십시오.

표 2-1 작업 맵: Solaris 소프트웨어 설치 또는 업그레이드 (계속)

작업	설명	지침
(선택 사항) 시스템 매개변수를 설정합니다.	시스템 정보를 사전 구성함으로써 설치나 업그레이드 중에 정보 입력을 요청받게 되는 일을 방지할 수 있습니다.	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 6 장, “시스템 구성 정보를 미리 구성(작업)”
(선택 사항) 네트워크를 통한 Solaris 소프트웨어 설치를 준비합니다.	네트워크에서 Solaris 소프트웨어를 설치하는 경우 다음 작업을 완료합니다. <ul style="list-style-type: none"> ■ (x86 기반 시스템) 시스템에서 PXE를 지원하는지 확인합니다. ■ 설치 서버를 만듭니다. ■ 부트 서버를 만듭니다(필요한 경우). ■ DHCP 서버를 구성합니다(필요한 경우). ■ 네트워크에서 설치할 시스템을 설정합니다. 	LAN을 통해 설치하려면 Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 9 장, “CD 매체를 사용하여 네트워크 설치 준비(작업)”를 참조하십시오. WAN을 통해 설치하려면 Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 13 장, “WAN 부트 사용 설치 준비(작업)”를 참조하십시오.
(업그레이드에만 해당) 업그레이드 전에 작업을 수행합니다.	시스템을 백업하고 디스크 공간 재할당으로 업그레이드할 수 있는지 결정합니다.	38 페이지 “업그레이드 계획”.
설치 또는 업그레이드를 수행합니다.	선택한 Solaris 설치 방법을 사용하여 Solaris 소프트웨어를 설치 또는 업그레이드합니다.	설치 프로그램에 대한 자세한 지침을 제공하는 장.
설치 문제 해결	설치에 문제가 발생하면 문제 해결 정보를 검토합니다.	부록 A.

네트워크, DVD 또는 CD 중 어떤 것을 이용해 설치할 것인가?

Solaris 소프트웨어는 DVD나 CD 매체를 통해 유통되므로 DVD-ROM이나 CD-ROM 드라이브에 액세스할 수 있는 시스템에 설치하거나 업그레이드할 수 있습니다.

원격 DVD 또는 CD 이미지로 네트워크에서 설치하도록 시스템을 설정할 수 있습니다. 다음 이유로 인해 이런 식으로 시스템을 설정하고자 할 수 있습니다.

- 로컬 DVD-ROM 또는 CD-ROM 장치가 없는 시스템의 경우
- 여러 시스템을 설치할 예정이고 Solaris 소프트웨어를 설치하기 위해 모든 로컬 드라이브에 디스크를 삽입 안 하려는 경우

모든 Solaris 설치 방법을 이용해 네트워크에서 시스템을 설치할 수 있습니다. 하지만 Solaris Flash 설치 기능이나 사용자 정의 JumpStart 설치를 이용하여 네트워크에서 시스템을 설치하여 대기업의 설치 프로세스를 중앙 집중화 및 자동화할 수 있습니다. 서로 다른 설치 방법에 대한 자세한 내용은 29 페이지 “Solaris 설치 방법 선택”을 참조합니다.

Solaris 소프트웨어를 네트워크를 통해 설치하려면 초기 설정이 필요합니다. 네트워크를 통한 설치 준비에 대한 자세한 내용은 다음 옵션 중 하나를 선택하십시오.

LAN을 통한 설치 준비에 대한 자세한 내용은	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 9 장, “CD 매체를 사용하여 네트워크 설치 준비(작업)”
WAN을 통한 설치 준비에 대한 자세한 내용은	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 13 장, “WAN 부트 사용 설치 준비(작업)”
PXE를 사용하여 네트워크를 통해 x86 기반 클라이언트를 설치하는 방법에 대한 지침	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 “PXE를 사용하여 네트워크를 통해 부팅 및 설치 개요”

초기 설치인가 업그레이드인가?

초기 설치 수행을 선택하거나 시스템에 이미 Solaris OS가 실행되고 있다면 시스템 업그레이드를 선택할 수 있습니다.

초기 설치

초기 설치하는 새로운 버전의 Solaris OS로 시스템의 디스크를 덮어씁니다. 시스템에 Solaris OS가 실행되고 있지 않은 경우 초기 설치를 수행해야 합니다.

시스템에 이미 Solaris OS가 실행되고 있는 경우에도 초기 설치를 수행할 수 있습니다. 로컬 수정 사항을 보존하려면 설치하기 전에 로컬 수정 사항을 백업합니다. 설치를 완료한 후 로컬 수정 사항을 복원할 수 있습니다.

Solaris 설치 방법 중 하나를 사용하여 초기 설치를 수행할 수 있습니다. 다른 Solaris 설치 방법에 대한 자세한 내용은 29 페이지 “Solaris 설치 방법 선택”을 참조하십시오.

업그레이드

표준 및 Solaris Live Upgrade의 두 가지 업그레이드 방법을 사용하여 Solaris OS를 업그레이드할 수 있습니다. 표준 업그레이드는 현재 Solaris OS의 가능한 한 많은 기존 구성 매개변수를 유지합니다. Solaris Live Upgrade는 현재 시스템의 복사본을 만듭니다. 이 복사본이 표준 업그레이드로 업그레이드될 수 있습니다. 업그레이드된 Solaris OS는 그 다음 단일 재부트로 현재 시스템이 되도록 전환될 수 있습니다. 오류가 발생하면 재부트로 원래 Solaris OS로 다시 전환할 수 있습니다. Solaris Live Upgrade를 사용하여 업그레이드하는 동안 시스템을 계속 실행할 수 있고 Solaris OS 릴리스 사이클을 전환할 수 있습니다.

업그레이드 및 업그레이드 방법의 목록에 대해서는 38 페이지 “업그레이드 계획”을 참조하십시오.

Solaris 설치 방법 선택

Solaris OS는 설치 또는 업그레이드를 위해 몇 가지 프로그램을 제공합니다. 각각의 설치 기술은 특정 설치 요구 사항 및 환경을 위해 설계된 다양한 기능을 제공합니다. 다음 표가 사용할 설치 방법을 결정하는 데 도움이 됩니다.

표 2-2 설치 방법 선택

작업	설치 방법	프로그램 선택 이유	지침
대화식 프로그램으로 CD-ROM 또는 DVD-ROM 매체에서 시스템을 설치합니다.	Solaris 설치 프로그램	<ul style="list-style-type: none"> 이 프로그램은 작업을 패넬로 나누고, 정보 입력 프롬프트를 표시하고, 기본값을 제공합니다. 여러 시스템에서 설치 또는 업그레이드를 수행하는 경우 이 프로그램은 효율적인 방법이 아닙니다. 많은 시스템을 일괄 설치하려면 사용자 정의 JumpStart나 Solaris Flash 설치 기능을 사용합니다. 	Solaris 10 6/06 설치 설명서: 기본 설치
LAN을 통해 시스템을 설치합니다.	네트워크를 통한 Solaris 설치 프로그램	이 프로그램을 사용하면 서버에 설치할 소프트웨어의 이미지를 설정하고 원격 시스템에 이 이미지를 설치할 수 있습니다. 여러 시스템을 설치해야 하는 경우에는 사용자 정의 JumpStart 및 Solaris Flash 설치 방법으로 네트워크 설치 이미지를 사용하여 네트워크에 시스템을 효율적으로 설치 또는 업그레이드할 수 있습니다.	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 제II부, “LAN(Local Area Network)을 통한 설치”
작성한 프로필을 기준으로 여러 시스템의 설치 또는 업그레이드를 자동화합니다.	사용자 정의 JumpStart	이 프로그램은 여러 시스템을 효율적으로 설치합니다. 하지만 시스템이 몇 개 뿐인 경우에는 사용자 정의 JumpStart 환경을 만드는 데 시간이 많이 걸릴 수 있습니다. 시스템이 몇 개 정도인 경우에는 Solaris 대화식 설치 프로그램을 사용합니다.	6장

표 2-2 설치 방법 선택 (계속)

작업	설치 방법	프로그램 선택 이유	지침
여러 시스템에 동일한 소프트웨어와 구성을 복제합니다.	Solaris Flash 아카이브	<ul style="list-style-type: none"> 이 프로그램은 모든 Solaris 패키지를 시스템에 한꺼번에 설치하여 설치 시간을 절약합니다. 다른 프로그램은 각 개별 Solaris 패키지를 설치하고 각 패키지의 패키지 맵을 업데이트합니다. Solaris Flash 아카이브는 대형 파일로서 상당한 양의 디스크 공간을 필요로 합니다. 여러 설치 구성을 관리하거나 설치 구성을 변경하려면 사용자 정의 JumpStart 설치 방법을 고려할 수 있습니다. 또는 JumpStart 종료 스크립트나 내장된 Solaris Flash 배치 후 스크립트를 사용하여 시스템별로 사용자 정의할 수 있습니다. 	Solaris 10 6/06 설치 설명서: Solaris Flash 아카이브(작성 및 설치)의 1 장, “Solaris Flash(개요)”
WAN 또는 네트워크를 통해 시스템을 설치합니다.	WAN 부트	이 프로그램에서는 보안 설치를 사용하여 네트워크를 통해 Solaris Flash 아카이브를 설치할 수 있습니다.	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 11 장, “WAN 부트(개요)”
시스템이 실행 중인 동안 업그레이드합니다.	Solaris Live Upgrade	<ul style="list-style-type: none"> 이 프로그램을 사용하면 업그레이드 또는 패치 추가를 사용하여 표준 업그레이드에 비해 시스템 정지 시간을 줄일 수 있습니다. 이 프로그램을 사용하면 현재 OS에 영향을 주지 않고도 업그레이드 또는 새 패치를 테스트할 수 있습니다. 	Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 6 장, “Solaris Live Upgrade(개요)”
Solaris OS 설치 후 고립된 응용 프로그램 환경을 만듭니다.	Solaris 영역 분할 기술	이 프로그램은 보안 응용 프로그램 환경을 제공하는 고립된 비전역 영역을 만듭니다. 이 고립으로 한 영역에서 실행 중인 프로세스가 다른 영역에서 실행 중인 프로세스를 모니터링하거나 영향을 미칠 수 없게 됩니다.	System Administration Guide: Solaris Containers-Resource Management and Solaris Zones 의 16 장, “Introduction to Solaris Zones”

Sun Java System Application Server Platform Edition 8

Sun Java System Application Server Platform Edition 8은 응용 프로그램 서비스와 웹 서비스의 광범위한 배포를 제공합니다. 이 소프트웨어는 Solaris OS로 자동 설치됩니다. 다음 영역에서 서버에 대한 설명서를 찾을 수 있습니다.

서버 시작에 대한 설명서	/docs/QuickStart.html 에서 설치 디렉토리에 있는 Sun Java System Application Server Platform Edition 8 빠른 시작 설명서를 참조하십시오.
전체 애플리케이션 서버 설명서 집합	http://docs.sun.com/db/coll/ApplicationServer8_04q2
자습서	http://java.sun.com/j2ee/1.4/docs/tutorial/doc/index.html

Solaris 설치 및 업그레이드(계획)

이 장에서는 Solaris OS를 설치하거나 업그레이드하기 위한 시스템 요구 사항에 대해 설명합니다. 디스크 공간과 기본 스왑 공간 할당을 위한 일반적인 지침도 제공됩니다. 이 장은 다음 내용으로 구성되어 있습니다.

- 33 페이지 “시스템 요구 사항 및 권장사항”
- 34 페이지 “디스크 할당 및 스왑 공간”
- 38 페이지 “업그레이드 계획”
- 42 페이지 “시스템이 실행 중인 Solaris OS의 버전을 확인하는 방법”
- 42 페이지 “로컬 값”
- 43 페이지 “플랫폼 이름 및 그룹”
- 43 페이지 “시스템에 Solaris 영역이 설치된 경우 업그레이드”
- 46 페이지 “x86: 분할 권장사항”

시스템 요구 사항 및 권장사항

표 3-1 메모리, 스왑 및 프로세서 권장사항

요구 사항 유형	크기
설치 또는 업그레이드할 메모리	<ul style="list-style-type: none"> ■ SPARC: 256MB는 권장 크기입니다. 128MB는 최소 크기입니다. ■ x86: 512MB는 권장 크기입니다. 256MB는 최소 크기입니다. <p>주-선택적인 일부 설치 기능은 메모리가 충분한 경우에만 사용할 수 있습니다. 예를 들어 메모리가 부족한 상태에서 DVD에서 설치하는 경우 GUI가 아닌 Solaris 설치 프로그램의 텍스트 설치 프로그램으로 설치하게 됩니다. 메모리 요구 사항에 대한 자세한 내용은 표 3-2를 참조하십시오.</p>
스왑 공간	<p>기본 크기는 512MB입니다.</p> <p>주-스왑 공간을 사용자 정의해야 할 수 있습니다. 스왑 공간은 시스템의 하드 디스크 크기를 기준으로 합니다.</p>

표 3-1 메모리, 스왑 및 프로세서 권장사항 (계속)

요구 사항 유형	크기
프로세서 요구 사항	<ul style="list-style-type: none"> ▪ SPARC: 200-MHz 이상 프로세서가 필수입니다. ▪ x86: 120-MHz 이상 프로세서가 권장됩니다. 하드웨어 부동 소수점 지원이 필수입니다.

GUI가 있는 또는 윈도우화 환경이 있거나 없는 소프트웨어를 설치할 것을 선택할 수 있습니다. 충분한 메모리가 있을 경우 GUI가 기본적으로 표시됩니다. 다른 환경은 GUI에 대해 메모리가 충분하지 않을 경우에 기본적으로 표시됩니다. `nowin` 또는 `text` 부트 옵션으로 기본값을 대체할 수 있습니다. 그러나, 시스템의 메모리 양 또는 원격 설치에 의해 제한을 받습니다. 또한, Solaris 설치 프로그램이 비디오 어댑터를 감지하지 않으면 자동으로 콘솔 기반 환경으로 표시됩니다. 표 3-2에는 해당 환경에 대한 설명과 함께 환경 표시를 위한 최소 메모리 요구 사항이 나열되어 있습니다.

표 3-2 옵션 표시에 대한 메모리 요구 사항

메모리	설치 유형	설명
<ul style="list-style-type: none"> ▪ SPARC: 128-383MB ▪ x86: 256-511MB 	텍스트 기반	<p>그래픽이 없지만 윈도우 및 다른 윈도우를 열 수 있는 기능을 제공합니다.</p> <p><code>text</code> 부트 옵션을 사용하여 설치하고 시스템에 메모리가 충분한 경우 윈도우화 환경에서 설치합니다. <code>tip</code> 라인 또는 <code>nowin</code> 부트 옵션을 통해 원격으로 설치하는 경우 콘솔 기반 설치에 제한됩니다.</p>
<ul style="list-style-type: none"> ▪ SPARC: 384MB 이상 ▪ x86: 512MB 	GUI 기반	윈도우, 폴다운 메뉴, 버튼, 스크롤 막대, 아이콘 이미지를 제공합니다.

디스크 할당 및 스왑 공간

Solaris 10 6/06 소프트웨어를 설치하기 전에 몇 가지 높은 수준의 계획을 수립하여 시스템에 충분한 디스크 공간이 있는지 확인할 수 있습니다.

일반적 디스크 공간 계획 수립 및 권장 사항

디스크 공간 계획은 개인에 따라 다릅니다. 사용자의 요구에 따라 다음 조건을 위한 공간 할당을 고려합니다.

표 3-3 일반 디스크 공간 및 스왑 공간 계획

공간 할당을 위한 조건	설명
파일 시스템	<p>만드는 각각의 파일 시스템에 대해서, 추후 Solaris 버전으로의 업그레이드를 위해 필요한 것보다 30퍼센트 많은 디스크 공간을 할당합니다.</p> <p>기본적으로 Solaris 설치 방법은 루트(/) 및 /swap만을 만듭니다. OS 서비스용 공간이 할당되면 /export 디렉토리도 생성됩니다. 주요 Solaris 릴리스로 업그레이드하는 경우 시스템을 다시 분할하거나 설치 시간에 필요한 공간의 두 배를 할당해야 할 수 있습니다. 업데이트로 업그레이드하는 경우 추후 업그레이드용으로 여분의 디스크 공간을 할당하여 시스템을 재분할하지 않도록 할 수 있습니다. Solaris 업데이트 릴리스는 이전 릴리스보다 약 10퍼센트 많은 디스크 공간을 필요로 합니다. 각 파일 시스템에 대하여 추가로 30퍼센트 디스크 공간을 할당하여 여러 Solaris 업데이트에 대한 공간을 허용할 수 있습니다.</p>
/var 파일 시스템	<p>충돌 덤프 기능인 savecore(1M)을 사용하려면 /var 파일 시스템에 있는 실제 메모리의 두 배를 할당하십시오.</p>
스왑	<p>Solaris 설치 프로그램은 다음 조건에서 512MB의 기본 스왑 공간을 할당합니다.</p> <ul style="list-style-type: none"> ■ 설치 프로그램의 디스크 슬라이스 자동 레이아웃을 사용할 경우 ■ 스왑 공간 크기의 수동 변경을 방지할 경우 <p>기본적으로 Solaris 설치 프로그램은 첫 번째 사용 가능 디스크 실린더(일반적으로 SPARC 기반 시스템에서 실린더 0)에서 시작하도록 스왑을 배치하여 스왑 공간을 할당합니다. 이렇게 배치하면 기본 디스크 레이아웃 동안 루트(/) 파일 시스템에 최대한의 공간이 제공되고 업그레이드하는 동안 루트(/) 파일 시스템이 증가될 수 있습니다.</p> <p>향후 스왑 영역의 확장이 필요할 경우에 다음 방법 중 하나를 통해 스왑 슬라이스가 다른 디스크 실린더에서 시작하도록 스왑 슬라이스를 배치할 수 있습니다.</p> <ul style="list-style-type: none"> ■ Solaris 설치 프로그램의 경우 실린더 모드에서 디스크 레이아웃을 사용자 정의하고 스왑 슬라이스를 원하는 위치로 수동 할당할 수 있습니다. ■ 사용자 정의 JumpStart 설치 프로그램의 경우 프로필 파일에 스왑 슬라이스를 할당할 수 있습니다. JumpStart 프로필 파일에 대한 자세한 내용은 81 페이지 “프로필 작성”을 참조하십시오. <p>스왑 공간에 대한 개요는 System Administration Guide: Devices and File Systems의 21 장, “Configuring Additional Swap Space (Tasks)”를 참조하십시오.</p>
홈 디렉토리 파일 시스템을 제공하는 서버	<p>기본적으로 홈 디렉토리는 /export 파일 시스템에 있습니다.</p>
설치하는 Solaris 소프트웨어 그룹	<p>소프트웨어 그룹은 소프트웨어 패키지의 그룹화입니다. 디스크 공간 계획을 세우는 동안 선택한 소프트웨어 그룹에서 개별 소프트웨어 패키지를 추가 또는 제거할 수 있습니다. 소프트웨어 그룹에 대한 자세한 내용은 36 페이지 “소프트웨어 그룹의 권장 디스크 공간”을 참조하십시오.</p>

표 3-3 일반 디스크 공간 및 스왑 공간 계획 (계속)

공간 할당을 위한 조건	설명
업그레이드	<ul style="list-style-type: none"> ■ Solaris Live Upgrade를 사용하여 비활성 부트 환경을 업그레이드하고 디스크 공간 계획에 대한 정보를 보려면, Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 “Solaris Live Upgrade 디스크 공간 요구 사항”을 참조하십시오. ■ Solaris 설치 프로그램 또는 사용자 정의 JumpStart를 사용하여 디스크 공간을 계획하는 경우 40 페이지 “디스크 공간 재할당으로 업그레이드”를 참조하십시오. ■ 시스템에 비전역 영역이 설치된 경우 45 페이지 “비전역 영역의 디스크 공간 요구 사항”을 참조하십시오.
언어 지원	예를 들어, 중국어, 일본어 또는 한국어. 단일 언어를 설치할 예정이면 대략 0.7GB의 추가 디스크 공간을 해당 언어를 위해 할당합니다. 전체 언어 지원을 설치할 예정이면 설치하는 소프트웨어 그룹에 따라 다르지만 대략 2.5GB의 추가 디스크 공간을 언어 지원을 위해 할당해야 합니다.
인쇄 또는 메일 지원	추가 공간을 할당합니다.
추가 소프트웨어 또는 타업체 소프트웨어	추가 공간을 할당합니다.

소프트웨어 그룹의 권장 디스크 공간

Solaris 소프트웨어 그룹은 Solaris 패키지들을 모은 것입니다. 각각의 소프트웨어 그룹은 서로 다른 기능과 하드웨어 드라이버를 지원합니다.

- 초기 설치의 경우 시스템에서 수행하고자 하는 기능을 기준으로 설치할 소프트웨어 그룹을 선택합니다.
- 업그레이드의 경우 시스템에 설치된 소프트웨어 그룹으로 업그레이드해야 합니다. 예를 들어, 이전에 시스템에 최종 사용자 Solaris 소프트웨어 그룹을 설치한 경우, 업그레이드 옵션을 사용하여 개발자 Solaris 소프트웨어 그룹으로 업그레이드할 수 없습니다. 하지만 업그레이드 중에 현재 설치되어 있는 소프트웨어 그룹의 일부가 아닌 소프트웨어를 추가할 수 있습니다.

Solaris 소프트웨어를 설치할 때, 선택한 Solaris 소프트웨어 그룹에서 패키지를 추가하거나 제거할 수 있습니다. 추가 또는 삭제할 패키지를 선택하려면 소프트웨어 종속성과 Solaris 소프트웨어의 패키지를 만드는 방법에 대해 알아야 합니다.

다음 그림은 소프트웨어 패키지의 그룹화를 보여줍니다. 축소 네트워크 지원은 최소 수의 패키지를 포함하고 전체 Solaris 소프트웨어 그룹 및 OEM 지원은 모든 패키지를 포함합니다.

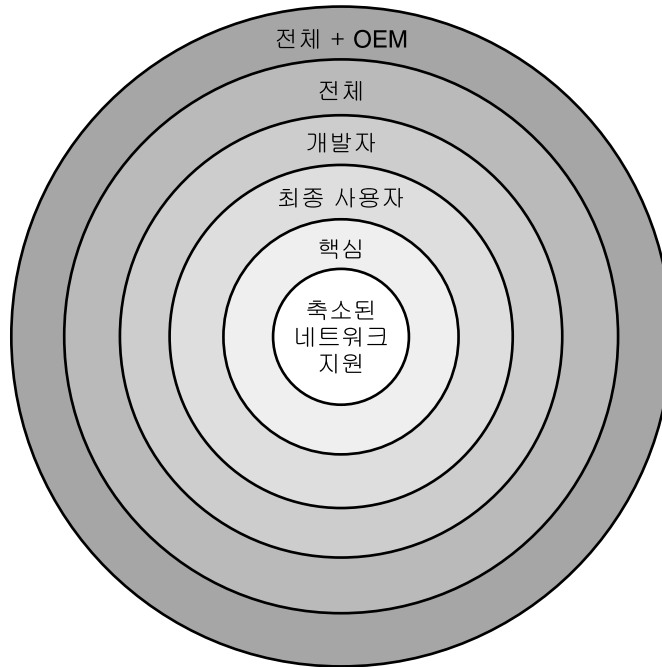


그림 3-1 Solaris 소프트웨어 그룹

표 3-4에는 Solaris 소프트웨어 그룹과 각각의 그룹을 설치하는 데 필요한 권장 디스크 공간이 나열되어 있습니다.

주 - 표 3-4의 권장 디스크 공간에는 다음 항목을 위한 공간이 포함됩니다.

- 스왑 공간
- 패치
- 추가 소프트웨어 패키지

소프트웨어 그룹은 다음 표에 나열된 공간보다 적은 디스크 공간을 요구합니다.

표 3-4 소프트웨어 그룹의 권장 디스크 공간

소프트웨어 그룹	설명	권장 디스크 공간
전체 Solaris 소프트웨어 그룹 및 OEM 지원	전체 Solaris 소프트웨어 그룹과 설치시 시스템에 없는 하드웨어용 드라이버를 포함한 추가 하드웨어 드라이버가 포함되어 있습니다.	6.8GB
전체 Solaris 소프트웨어 그룹	개발자 Solaris 소프트웨어 그룹용 패키지와 서버용으로 필요한 추가 소프트웨어가 포함되어 있습니다.	6.7GB

표 3-4 소프트웨어 그룹의 권장 디스크 공간 (계속)

소프트웨어 그룹	설명	권장 디스크 공간
개발자 Solaris 소프트웨어 그룹	최종 사용자 Solaris 소프트웨어 그룹용 패키지와 소프트웨어 개발용 추가 지원이 들어 있습니다. 추가 소프트웨어 개발 지원에는 라이브러리, 파일, 매뉴얼 페이지, 프로그래밍 도구가 포함되어 있습니다. 컴파일러는 포함되어 있지 않습니다.	6.6GB
최종 사용자 Solaris 소프트웨어 그룹	네트워크에 연결된 Solaris 시스템 및 공통 데스크탑 환경을 부트 및 실행하는 데 필요한 최소 코드를 제공하는 패키지가 포함되어 있습니다.	5.3GB
코어 시스템 지원 소프트웨어 그룹	네트워크에 연결된 Solaris 시스템을 부트 및 실행하는 데 필요한 최소 코드를 제공하는 패키지가 포함되어 있습니다.	2.0GB
축소 네트워크 지원 소프트웨어 그룹	제한된 네트워크 서비스 지원을 가진 Solaris 시스템을 부트 및 실행하는 데 필요한 최소 코드를 제공하는 패키지가 포함되어 있습니다. 축소 네트워크 지원 소프트웨어 그룹은 다중 사용자 텍스트 기반 콘솔과 시스템 관리 유틸리티를 제공합니다. 이 소프트웨어 그룹은 또한 시스템이 네트워크 인터페이스를 인식하도록 하지만 네트워크 서비스를 활성화하지는 않습니다.	2.0GB

업그레이드 계획

Solaris Live Upgrade, Solaris 설치 프로그램, 사용자 정의 JumpStart의 세 가지 업그레이드 방법 중 하나를 사용하여 시스템을 업그레이드할 수 있습니다.

표 3-5 Solaris 업그레이드 방법

현재 Solaris OS	Solaris 업그레이드 방법
Solaris 8, Solaris 9, Solaris 10	<ul style="list-style-type: none"> ■ Solaris Live Upgrade - 실행 중인 시스템의 복사본을 작성 및 업그레이드하여 시스템을 업그레이드합니다. ■ Solaris 설치 프로그램 - 그래픽 사용자 인터페이스 또는 명령줄 인터페이스를 사용하는 대화식 업그레이드를 제공합니다. ■ 사용자 정의 JumpStart 방법 - 자동화된 업그레이드를 제공합니다.

업그레이드 제한 사항

문제	설명
다른 소프트웨어 그룹으로 업그레이드	시스템에 설치되어 있지 않은 소프트웨어 그룹으로는 시스템을 업그레이드할 수 없습니다. 예를 들어, 이전에 시스템에 최종 사용자 Solaris 소프트웨어 그룹을 설치한 경우, 업그레이드 옵션을 사용하여 개발자 Solaris 소프트웨어 그룹으로 업그레이드할 수 없습니다. 하지만 업그레이드 중에 현재 설치되어 있는 소프트웨어 그룹의 일부가 아닌 소프트웨어를 추가할 수 있습니다.
비전역 영역이 설치된 경우 업그레이드	Solaris OS를 업그레이드할 경우 비전역 영역이 설치된 시스템을 업그레이드할 수 있습니다. Solaris 대화식 설치 프로그램과 사용자 정의 JumpStart 프로그램을 사용하여 업그레이드를 수행할 수 있습니다. 업그레이드 시의 제한 사항에 대해서는 43 페이지 “시스템에 Solaris 영역이 설치된 경우 업그레이드”를 참조하십시오.
Veritas 파일 시스템을 사용하여 업그레이드	Solaris 대화식 설치 및 사용자 정의 JumpStart 프로그램은 다음 조건에서 Veritas VxVM 파일 시스템을 사용하는 경우 시스템을 업그레이드할 기회를 제공하지 않습니다. <ul style="list-style-type: none"> ■ 업그레이드할 루트 파일 시스템이 Veritas 제어 상태에 있는 경우 예를 들어 루트(/) 파일 시스템이 /dev/vx/... 장치에 마운트된 경우 ■ Veritas 제어 상태에 있는 파일 시스템에 Solaris 소프트웨어가 설치된 경우 예를 들어 /usr 파일 시스템이 /dev/vx/... 장치에 마운트된 경우 Veritas VxVM가 구성될 때 업그레이드하려면 다음 방법 중 하나를 사용합니다. <ul style="list-style-type: none"> ■ Solaris Live Upgrade 사용. 245 페이지의 223 페이지 “Veritas VxVm을 실행하는 Solaris Live Upgrade를 사용하여 업그레이드하는 중에 시스템 장애 발생” ■ 비전역 영역이 설치되어 있는 경우 영향 받는 파일 시스템을 VxVM 파일 시스템에서 UFS 파일 시스템으로 이전해야 합니다.

업그레이드 프로그램

Solaris 설치 프로그램으로 표준 대화식 업그레이드를 수행하거나 사용자 정의 JumpStart 설치 방법을 사용하여 무인 업그레이드를 수행할 수 있습니다. Solaris Live Upgrade를 사용하여 실행 중 시스템을 업그레이드할 수 있습니다.

업그레이드 프로그램	설명	자세한 내용
Solaris Live Upgrade	현재 실행 중인 시스템의 복사본을 작성할 수 있도록 합니다. 복사본이 업그레이드된 후 재부트가 업그레이드된 복사본을 현재 실행 중 시스템이 되도록 전환합니다. Solaris Live Upgrade를 사용하면 Solaris OS를 업그레이드하는 데 필요한 작동 중지 시간이 줄어듭니다. 또한, Solaris Live Upgrade는 업그레이드로 문제를 방지할 수 있습니다. 그 한 가지 예로는 전원 이상이 발생할 경우 업그레이드에서 복구하는 기능으로, 이는 업그레이드되는 사본이 현재 실행 중인 시스템이 아니기 때문입니다.	Solaris Live Upgrade를 사용할 때 디스크 공간 할당을 계획하려면 Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 “Solaris Live Upgrade 요구 사항”을 참조하십시오.
Solaris 설치 프로그램	대화식 GUI를 사용한 업그레이드 과정을 안내합니다.	Solaris 10 6/06 설치 설명서: 기본 설치의 2장, “Solaris 설치 프로그램을 사용하여 설치(작업)”
사용자 정의 JumpStart 프로그램	자동화된 업그레이드를 제공합니다. 프로필 파일과 옵션 사전 설치 및 사후 설치 스크립트가 필요한 정보를 제공합니다. 업그레이드를 위해 사용자 정의 JumpStart 프로필을 만드는 경우 <code>install_type upgrade</code> 를 지정하십시오. 업그레이드를 하기 전에 먼저 사용자 정의 JumpStart 프로필에서 시스템의 디스크 구성과 현재 설치된 소프트웨어를 테스트해야 합니다. 업그레이드하는 시스템에서 <code>pfinstall -D</code> 명령어를 사용하여 프로필을 테스트합니다. 디스크 구성 파일을 사용하여 업그레이드 프로필을 테스트할 수는 없습니다.	<ul style="list-style-type: none"> ■ 업그레이드 옵션 테스트에 대한 자세한 내용은 93 페이지 “프로필 테스트”를 참조하십시오. ■ 업그레이드 프로필 만들기에 대한 자세한 내용은 83 페이지 “프로필의 예”를 참조하십시오. ■ 업그레이드 수행에 대한 자세한 내용은 122 페이지 “SPARC: 사용자 정의 JumpStart 설치 수행”을 참조하십시오.

업그레이드 대신 Solaris Flash 아카이브 설치

Solaris Flash 설치 기능은 여러 복제 시스템에서 복제될 수 있는 마스터 시스템에서 전체 설치의 복사본을 작성하는 방법을 제공합니다. 이 복사본을 Solaris Flash 아카이브라고 합니다. 설치 프로그램을 사용하여 아카이브를 설치할 수 있습니다.



주의 - 비전역 영역이 설치되어 있는 경우 Solaris Flash 아카이브를 제대로 만들 수 없습니다. Solaris Flash 기능은 Solaris 영역 분할 기술과 호환되지 않습니다. Solaris Flash 아카이브를 만들 때 다음 조건 하에서 아카이브를 배포하면 결과 아카이브가 제대로 설치되지 않습니다.

- 아카이브를 비전역 영역에 만든 경우
- 아카이브를 비전역 영역이 설치된 전역 영역에 만든 경우

디스크 공간 재할당으로 업그레이드

Solaris 설치 프로그램의 업그레이드 옵션과 사용자 정의 JumpStart 프로그램의 `upgrade` 키워드는 디스크 공간을 재할당하는 기능을 제공합니다. 이 재할당은 자동으로 디스크 슬라이스의 크기를 변경합니다. 기존 파일 시스템에 업그레이드를 위한 충분한 공간이

없으면 디스크 공간을 재할당할 수 없습니다. 예를 들어, 다음과 같은 이유로 파일 시스템이 더 많은 업그레이드 공간을 필요로 할 수 있습니다.

- 시스템에 현재 설치되어 있는 Solaris 소프트웨어 그룹에 새 릴리스의 새로운 소프트웨어가 포함되어 있습니다. 소프트웨어 그룹에 포함되어 있는 새로운 소프트웨어는 모두 업그레이드 중에 설치되도록 자동 선택됩니다.
- 시스템의 기존 소프트웨어 크기가 새 릴리스에서 커졌습니다.

자동 레이아웃 기능이 디스크 공간을 재할당하여 파일 시스템의 새로운 크기 요구 사항을 충족시키려 시도합니다. 처음에 자동 레이아웃이 일련의 기본 제약 조건을 바탕으로 공간을 재할당하려 시도합니다. 자동 레이아웃이 공간을 재할당할 수 없다면 파일 시스템에 대한 제약 조건을 변경해야 합니다.

주 - 자동 레이아웃은 파일 시스템을 "증대하는" 기능이 없습니다. 자동 레이아웃은 다음 프로세스로 공간을 재할당합니다.

1. 백업을 위해서는 변경해야 하는 파일 시스템의 파일이 필요합니다.
2. 파일 시스템 기준 디스크 재할당이 변경됩니다.
3. 업그레이드가 시작하기 전에 백업 파일을 복원합니다.

- Solaris 설치 프로그램을 사용하고 있고 자동 레이아웃이 디스크 공간 재할당 방법을 결정할 수 없는 경우, 사용자 정의 JumpStart 프로그램을 사용하여 업그레이드해야 합니다.
- 업그레이드를 위해 사용자 정의 JumpStart 방법을 사용하고 있고 업그레이드 프로필을 만드는 경우 디스크 공간이 문제가 될 수 있습니다. 현재 파일 시스템에 업그레이드를 위한 디스크 공간이 충분하지 않은 경우 `backup_media` 및 `layout_constraint` 키워드를 사용하여 디스크 공간을 재할당할 수 있습니다. 프로필의 `backup_media` 및 `layout_constraint` 키워드를 사용하는 방법의 예는 83 페이지 “프로필의 예”를 참조하십시오.

업그레이드할 때 Patch Analyzer 사용

Patch Analyzer는 다음과 같은 초기 Solaris 10 3/05 릴리스 이후의 릴리스 중 하나로 업그레이드하려면 시스템에서 분석을 수행합니다.

- Solaris 10 1/06 릴리스
- Solaris 10 6/06 릴리스

Solaris OS를 이미 실행 중이고 개별 패치를 설치한 경우 이후 Solaris 10 릴리스로 업그레이드하면 다음이 발생합니다.

- 위에서 언급한 릴리스 중 하나의 일부로 제공된 패치가 시스템에 재적용됩니다. 이 패치들은 제거할 수 없습니다.
- 이전에 시스템에 설치된 패치 중에 위에서 언급된 릴리스에 포함되어 있지 않은 패치는 모두 제거됩니다.

Patch Analyzer를 사용하여 제거하려는 패치(있는 경우)를 결정할 수 있습니다. Patch Analyzer 사용에 대한 자세한 지침은 **Solaris 10 6/06 설치 설명서: Solaris Live Upgrade** 및 업그레이드 계획의 부록 C, “업그레이드할 때 Patch Analyzer 사용(작업)”을 참조하십시오.

업그레이드하기 전에 시스템 백업

Solaris OS를 업그레이드하기 전에 기존 파일 시스템을 백업할 것을 강력하게 권장합니다. 파일 시스템을 테이프와 같은 이동식 매체에 복사하면 데이터 손실 또는 손상을 방지할 수 있습니다.

- 시스템 백업 지침에 대한 자세한 내용은 **System Administration Guide: Devices and File Systems**의 24 장, “Backing Up and Restoring File Systems (Overview)”를 참조하십시오.
- 비전역 영역이 설치된 경우 시스템을 백업하려면 **System Administration Guide: Solaris Containers-Resource Management and Solaris Zones**의 25 장, “Solaris Zones Administration (Overview)”를 참조하십시오.

시스템이 실행 중인 Solaris OS의 버전을 확인하는 방법

시스템에서 실행 중인 Solaris 소프트웨어의 버전을 확인하려면 다음 명령어를 입력하십시오.

```
$ uname -a
```

cat 명령어는 자세한 정보를 제공합니다.

```
$ cat /etc/release
```

로케일 값

설치의 일부로 시스템이 사용할 로케일을 미리 구성할 수 있습니다. 로케일은 온라인 정보를 특정 언어 및 특정 지역에 맞게 표시하는 방법을 결정합니다. 언어에는 날짜 및 시간 형식, 숫자 및 통화 체계, 맞춤법 등의 차이와 같은 지역적 편차를 반영하여 둘 이상의 로케일을 포함할 수 있습니다.

사용자 정의 JumpStart 프로파일 또는 sysidcfg 파일에서 시스템 로케일을 미리 구성할 수 있습니다.

프로파일의 로케일 설정	81 페이지 “프로파일 작성”
sysidcfg 파일의 로케일 설정	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 “sysidcfg 파일로 미리 구성”

플랫폼 이름 및 그룹

네트워크 설치에 대하여 클라이언트를 추가하는 경우 시스템 아키텍처(플랫폼 그룹)를 알아야 합니다. 사용자 정의 JumpStart 설치 규칙 파일을 쓰는 경우 플랫폼 이름을 알아야 합니다.

다음은 플랫폼 이름과 그룹의 일부 예입니다. SPARC 기반 시스템의 전체 목록은 <http://docs.sun.com/>의 **Solaris Sun** 하드웨어 플랫폼 안내서를 참조하십시오.

표 3-6 플랫폼 이름 및 그룹의 예

시스템	플랫폼 이름	플랫폼 그룹
Sun Fire	T2000	sun4v
Sun Blade™	SUNW,Sun-Blade-100	sun4u
x86 기반	i86pc	i86pc

주 - 또한 실행 중인 시스템에서 `uname -i` 명령어를 사용하여 시스템의 플랫폼 이름을 결정하거나 `uname -m` 명령어를 사용하여 시스템의 플랫폼 그룹을 결정할 수도 있습니다.

시스템에 Solaris 영역이 설치된 경우 업그레이드

이 절에서는 Solaris 영역 분할 기술에 대한 간략한 개요, 비전역 영역으로 업그레이드 개요 및 디스크 공간 계획 지침을 제공합니다.

영역 개요, 계획, 작성 및 구성에 대한 자세한 내용은 **System Administration Guide: Solaris Containers-Resource Management and Solaris Zones**의 16 장, “Introduction to Solaris Zones”를 참조하십시오.

Solaris 영역 분할 기술은 운영 체제 서비스를 가상화하고 실행 중인 응용 프로그램을 위한 격리되고 안전한 환경을 제공하는 데 사용됩니다. 비전역 영역은 Solaris OS의 단일 인스턴스 내에 작성되는 가상화된 운영 체제 환경입니다. 비전역 영역을 작성할 경우 프로세스가 시스템의 나머지에서 고립되는 응용 프로그램 실행 환경을 만듭니다. 이러한 격리는 한 비전역 영역에서 실행 중인 프로세스가 다른 비전역 영역에서 실행 중인 프로세스를 모니터링하거나 영향을 줄 수 없도록 합니다. 슈퍼유저 인증서로 실행 중인 프로세스라도 다른 영역의 활동을 보거나 영향을 미칠 수 없습니다. 또한 비전역 영역은 응용 프로그램과 해당 응용 프로그램이 배포된 시스템의 물리적 속성을 서로 분리하는 추상적 계층을 제공합니다. 이러한 속성의 예로 물리적 장치 경로가 있습니다.

모든 Solaris 시스템에는 전역 영역이 포함되어 있습니다. 전역 영역에는 두 가지 기능이 있습니다. 전역 영역은 시스템의 기본 영역이자 시스템 전체의 관리 제어에 사용되는

영역입니다. 전역 관리자가 비전역 영역을 만들지 않으면 모든 프로세스가 전역 영역에서 실행됩니다. 전역 영역은 비전역 영역이 구성, 설치, 관리 또는 설치 해제될 수 있는 유일한 영역입니다. 전역 영역만 시스템 하드웨어에서 부트 가능합니다. 물리적 장치, 라우팅 또는 동적 재구성(DR)과 같은 시스템 인프라의 관리는 전역 영역에서만 가능합니다. 전역 영역에서 실행 중인 적절한 권한의 프로세스는 비전역 영역과 연관된 객체에 액세스할 수 있습니다.

비전역 영역으로 업그레이드

Solaris OS가 설치된 후 비전역 영역을 설치하고 구성할 수 있습니다. Solaris OS를 업그레이드할 준비가 되었으면 비전역 영역이 설치된 시스템을 업그레이드할 수 있습니다. Solaris 대화식 설치 프로그램과 사용자 정의 JumpStart 프로그램을 사용하여 업그레이드를 수행할 수 있습니다.

- Solaris 대화식 설치 프로그램을 사용하면 업그레이드 또는 초기 설치 선택 패널에서 업그레이드 설치를 선택하여 비전역 영역이 있는 시스템을 업그레이드할 수 있습니다. 그러면 설치 프로그램에서 시스템을 분석하여 시스템의 업그레이드 가능 여부를 확인한 다음 분석 요약을 표시합니다. 그런 다음 설치 프로그램에서 업그레이드를 계속하라는 메시지를 표시합니다. 이 프로그램 사용에는 다음과 같은 제한이 적용됩니다.
 - 업그레이드를 사용자 정의할 수 없습니다. 예를 들어, 추가 소프트웨어 제품 설치, 추가 로컬 패키지 설치 또는 디스크 레이아웃 수정 등을 수행할 수 없습니다.
 - Solaris Operating System DVD 또는 DVD로 만든 네트워크 설치 이미지를 사용해야 합니다. Solaris Software CD를 사용하여 시스템을 업그레이드할 수 없습니다. 이 프로그램을 사용한 설치에 대한 자세한 내용은 **Solaris 10 6/06 설치 설명서: 기본 설치**의 2 장, “Solaris 설치 프로그램을 사용하여 설치(작업)”를 참조하십시오.
- 사용자 정의 JumpStart 설치 프로그램에서는 `install_type` 및 `root_device` 키워드를 사용해야만 업그레이드를 수행할 수 있습니다.

일부 키워드는 비전역 영역에 영향을 미치지 때문에 키워드 중 일부는 프로필에 포함할 수 없습니다. 예를 들어, 패키지 추가, 디스크 공간 재할당 또는 로컬 추가를 수행하는 키워드를 사용하면 비전역 영역에 영향을 줍니다. 이런 키워드를 사용하면 키워드가 무시되거나 JumpStart 업그레이드가 실패합니다. 이런 키워드의 목록은 **186 페이지 “비전역 영역으로 업그레이드하는 경우 프로필 키워드 제한”**을 참조하십시오.

표 3-7 비전역 영역으로 업그레이드할 때 제한 사항

프로그램 또는 조건	설명
Solaris Live Upgrade	비전역 영역이 설치되어 있으면 Solaris Live Upgrade를 사용하여 시스템을 업그레이드할 수 없습니다. <code>lucreate</code> 명령어를 사용하면 부트 환경을 만들 수 있지만 <code>luupgrade</code> 명령어를 사용하면 업그레이드가 실패합니다. 오류 메시지가 표시됩니다.

표 3-7 비전역 영역으로 업그레이드할 때 제한 사항 (계속)

프로그램 또는 조건	설명
Solaris Flash 아카이브	비전역 영역이 설치되어 있으면 Solaris Flash 아카이브를 제대로 만들 수 없습니다. Solaris Flash 기능은 Solaris 영역 분할 기술과 호환되지 않습니다. Solaris Flash 아카이브를 만들 때 다음 조건 하에서 아카이브를 배포하면 결과 아카이브가 제대로 설치되지 않습니다. <ul style="list-style-type: none"> ■ 아카이브를 비전역 영역에 만든 경우 ■ 아카이브를 비전역 영역이 설치된 전역 영역에 만든 경우
-R 옵션이나 이와 동일한 기능을 사용하는 명령어는 일부 상황에서는 사용할 수 없습니다.	다음과 같은 경우에는 -R 옵션이나 이와 동일한 옵션을 사용하여 대체 루트(/) 파일 시스템을 적용하는 명령어를 사용할 수 없습니다. <ul style="list-style-type: none"> ■ 명령어는 전역 영역에서 실행되는 경우 ■ 대체 루트(/) 파일 시스템이 비전역 영역 내의 경로를 참조하는 경우 <p>예로는 비전역 영역 내의 루트(/) 파일 시스템 경로를 사용하여 전역 영역에서 실행되는 pkgadd 유틸리티의 -R <i>root_path</i> 옵션이 있습니다.</p> <p>대체 루트(/) 파일 시스템을 허용하는 유틸리티 목록과 영역에 대한 자세한 내용은 System Administration Guide: Solaris Containers-Resource Management and Solaris Zones의 “Restriction on Accessing A Non-Global Zone From the Global Zone”을 참조하십시오.</p>

영역으로 업그레이드를 수행하기 전에 시스템 백업

업그레이드를 수행하기 전에 Solaris 시스템의 전역 및 비전역 영역을 백업해야 합니다. 전역이 설치된 시스템을 백업하는 방법에 대한 자세한 내용은 **System Administration Guide: Solaris Containers-Resource Management and Solaris Zones**의 25 장, “Solaris Zones Administration (Overview)”을 참조하십시오.

비전역 영역의 디스크 공간 요구 사항

전역 영역을 설치하는 경우 만드는 모든 영역을 수용할 수 있는 충분한 디스크 공간이 확보되어야 합니다. 각 비전역 영역은 고유한 디스크 공간 요구 사항을 가질 수 있습니다.

영역이 소모할 수 있는 디스크 공간의 양에는 제한이 없습니다. 전역 영역 관리자가 공간 제약을 책임집니다. 소형 유니프로세서 시스템도 동시에 실행하는 여러 영역을 지원할 수 있습니다. 전역 영역에 설치된 패키지의 특성은 만들어진 비전역 영역의 공간 요구 사항에 영향을 줍니다. 패키지 수 및 공간 요구 사항은 인수입니다.

전체 계획 요구 사항과 권장 사항은 **System Administration Guide: Solaris Containers-Resource Management and Solaris Zones**의 18 장, “Planning and Configuring Non-Global Zones (Tasks)”을 참조하십시오.

x86: 분할 권장사항

x86 기반 시스템에서 Solaris OS를 사용하는 경우 시스템 분할에 대하여 다음 지침을 따르십시오.

Solaris 설치 프로그램은 기본 디스크 분할 영역 레이아웃을 사용합니다. 이러한 분할 영역을 **fdisk** 분할 영역이라고 합니다. **fdisk** 분할 영역은 x86 기반 시스템의 특정 운영 체제 전용인 디스크 드라이브의 논리적 분할 영역입니다. Solaris 소프트웨어를 설치하려면 x86 기반 시스템에서 Solaris **fdisk** 분할 영역을 적어도 하나 설정해야 합니다. x86 기반 시스템은 디스크에서 최고 4개의 **fdisk** 분할 영역을 허용합니다. 개별 운영 체제를 보유하기 위해 이러한 분할 영역을 사용할 수 있습니다. 각 운영 체제는 고유한 **fdisk** 분할 영역에 위치해야 합니다. 시스템은 디스크당 하나의 Solaris **fdisk** 분할 영역만 가질 수 있습니다.

표 3-8 x86: 기본 분할 영역

분할 영역	분할 영역 이름	분할 영역 크기
첫 번째 분할 영역(일부 시스템에서)	진단 또는 서비스 분할 영역	시스템의 기존 크기
두 번째 분할 영역(일부 시스템에서)	x86 부트 분할 영역	<ul style="list-style-type: none"> ■ 초기 설치를 수행하는 경우 이 분할 영역은 만들어지지 않습니다. ■ 업그레이드를 수행하는 경우 시스템에 기존 x86 부트 분할 영역이 없으면 이 분할 영역이 만들어지지 않습니다. ■ 업그레이드를 수행하는 경우 시스템에 x86 부트 분할 영역이 있으면: <ul style="list-style-type: none"> ■ 한 부트 장치에서 다른 장치로 부트스트랩을 수행하는 데 분할 영역이 필요한 경우에는 시스템에 x86 부트 분할 영역이 보존됩니다. ■ 추가 부트 장치를 부트하는 데 분할 영역이 필요하지 않은 경우에는 x86 부트 분할 영역이 제거됩니다. 분할 영역의 내용이 루트 분할 영역으로 이동됩니다.
세 번째 분할 영역	Solaris OS 분할 영역	부트 디스크의 나머지 공간

기본 부트 디스크 분할 영역 레이아웃이 서비스 분할 영역을 보존

Solaris 설치 프로그램은 기본 부트 디스크 분할 영역 레이아웃을 사용하여 진단 및 서비스 분할 영역을 수용합니다. 시스템에 현재 진단 또는 서비스 분할 영역이 포함되어 있는 경우 새 기본 부트 디스크 분할 영역 레이아웃에서 해당 분할 영역을 보존할 수 있습니다.

주 - 현재 진단 또는 서비스 분할 영역을 포함하지 않는 x86 기반 시스템에 Solaris OS를 설치하는 경우 설치 프로그램은 기본적으로 새로운 진단 또는 서비스 분할 영역을 작성하지 않습니다. 시스템에 진단 또는 서비스 분할 영역을 만들려면 하드웨어 설명서를 참조하십시오.

x86: Solaris 설치를 위한 GRUB 기반 부트

이 장에서는 Solaris 설치와 관련된 x86 기반 시스템의 GRUB 기반 부트에 대해 설명합니다. 이 장은 다음 내용으로 구성되어 있습니다.

- 49 페이지 “x86: GRUB 기반 부트(개요)”
- 52 페이지 “x86: GRUB 기반 부트(계획)”
- 56 페이지 “x86: GRUB 메뉴의 menu.lst 파일 찾기(작업)”

x86: GRUB 기반 부트(개요)

오픈 소스 부트로더인 GRUB가 Solaris OS의 기본 부트로더로 채택되었습니다.

주 - GRUB 기반 부트는 SPARC 기반 시스템에서 사용할 수 없습니다.

부트로더는 시스템 전원을 켜고 후에 처음으로 실행되는 소프트웨어 프로그램입니다. x86 기반 시스템의 전원을 켜면 BIOS(Basic Input/Output System)가 CPU, 메모리, 플랫폼 하드웨어를 초기화합니다. 초기화 단계가 완료되면 BIOS는 구성된 부트 장치에서 부트로더를 로드한 다음 시스템 제어를 부트로더로 전송합니다.

GRUB는 구성 파일에 미리 지정된 부트 옵션을 포함한 단순 메뉴 인터페이스가 있는 오픈 소스 부트로더입니다. GRUB에는 메뉴 인터페이스에서 액세스하여 다양한 부트 명령어를 수행할 수 있는 명령줄 인터페이스도 있습니다. Solaris OS에서 GRUB 구현은 멀티부트 사양과 호환됩니다. 사양에 대한 자세한 설명은 <http://www.gnu.org/software/grub/grub.html>에 있습니다.

Solaris 커널은 멀티부트 사양과 완전히 호환되기 때문에 GRUB를 사용하여 Solaris x86 기반 시스템을 부트할 수 있습니다. GRUB를 사용하면 다양한 운영 체제를 좀더 쉽게 부트 및 설치할 수 있습니다. 예를 들어 한 시스템에서 다음 운영 체제를 개별적으로 부트할 수 있습니다.

- Solaris OS
- Microsoft Windows

주 - GRUB에서는 Microsoft Windows 분할 영역을 감지하지만 그 OS를 부트할 수 있는지는 확인하지는 않습니다.

GRUB의 주요 장점 중 하나는 파일 시스템과 커널 실행 파일 형식이 직관적이기 때문에 디스크 상의 물리적 커널 위치를 기록하지 않고도 운영 체제를 로드할 수 있다는 것입니다. GRUB 기반 부트에서는 커널의 파일 이름, 드라이브, 커널이 있는 분할 영역을 지정하여 커널을 로드합니다. GRUB 기반 부트는 Solaris Device Configuration Assistant를 대체하며 GRUB 메뉴로 부트 프로세스를 단순화합니다.

x86: GRUB 기반 부트 방법

GRUB에서 시스템을 제어하기 시작하면 콘솔에 메뉴가 표시됩니다. GRUB 메뉴에서는 다음을 수행할 수 있습니다.

- 항목을 선택하여 시스템 부트
- 내장 GRUB 편집 메뉴를 사용하여 부트 항목 수정
- 명령줄에서 수동으로 OS 커널 로드

구성 가능 시간 초과를 사용하여 기본 OS 항목을 부트할 수 있습니다. 임의의 키를 누르면 기본 OS 항목 부트가 중단됩니다.

GRUB 메뉴의 예를 보려면 53 페이지 “GRUB 주 메뉴 설명”을 참조하십시오.

x86: GRUB 장치 이름 지정 규약

GRUB에서 사용하는 장치 이름 지정 규약은 이전 Solaris OS 버전과 약간 다릅니다. GRUB 장치 이름 지정 규약을 이해하면 시스템에서 GRUB를 구성할 때 드라이브 및 분할 영역을 제대로 지정하는 데 도움이 됩니다.

다음 표는 GRUB 장치 이름 지정 규약을 설명합니다.

표 4-1 GRUB 장치의 이름 지정 규약

장치 이름	설명
(fd0), (fd1)	첫 번째 디스켓, 두 번째 디스켓
(nd)	네트워크 장치
(hd0, 0), (hd0, 1)	첫 번째 bios 디스크의 첫 번째 및 두 번째 fdisk 분할 영역
(hd0, 0, a), (hd0, 0, b)	첫 번째 bios 디스크의 첫 번째 fdisk 분할 영역에 있는 Solaris/BSD 슬라이스 0 및 1

주 - 모든 GRUB 장치 이름은 괄호 안에 넣어야 합니다. 분할 영역 번호는 1이 아닌 0(영)부터 계산합니다.

fdisk 분할 영역에 대한 자세한 내용은 **System Administration Guide: Devices and File Systems**의 “Guidelines for Creating an fdisk Partition”을 참조하십시오.

x86: GRUB 기반 설치에 관한 정보의 위치

변경 사항에 대한 자세한 내용은 다음 참조를 참조하십시오.

표 4-2 GRUB 기반 설치에 관한 정보의 위치

주제	GRUB 메뉴 작업	자세한 내용
설치	Solaris OS CD 또는 DVD 매체에서 설치	Solaris 10 6/06 설치 설명서: 기본 설치.
	네트워크 설치 이미지에서 설치	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 제II부, “LAN(Local Area Network)을 통한 설치”
	네트워크 설치를 위한 DHCP 서버 구성	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 “DHCP 서비스를 사용하여 시스템 구성 정보를 미리 구성(작업)”
	사용자 정의 JumpStart 프로그램을 사용하여 설치	128 페이지 “x86: 사용자 정의 JumpStart 설치 수행”
	Solaris Live Upgrade를 사용하여 활성화 또는 부트 환경으로 폴백	<ul style="list-style-type: none"> ■ Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 “부트 환경 활성화” ■ Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 10 장, “오류 복구: 원래 부트 환경으로 폴백(작업)”
시스템 관리	GRUB 및 관리 작업에 대한 자세한 정보	System Administration Guide: Basic Administration 의 11 장, “GRUB Based Booting (Tasks)”

x86: GRUB 기반 부트(계획)

이 절에서는 GRUB 기반 부트의 기본 사항과 GRUB 메뉴에 대해 설명합니다.

Solaris OS를 설치할 때 시스템에는 기본적으로 두 개의 GRUB 메뉴 항목이 설치됩니다. 첫 번째 항목은 Solaris OS 항목입니다. 두 번째 항목은 시스템 복구에 사용되는 비상 안전 부트 아카이브입니다. Solaris GRUB 메뉴 항목은 Solaris 소프트웨어 설치 및 업그레이드 프로세스의 일부로서 자동으로 설치 및 업데이트됩니다. 이 항목은 OS에서 직접 관리되며 수동으로 편집하면 안 됩니다.

표준 Solaris OS 설치 중에, GRUB는 시스템 BIOS 설정을 수정하지 않고 Solaris fdisk 분할 영역에 설치됩니다. OS가 BIOS 부트 디스크에 있지 않은 경우에는 다음 중 하나를 수행해야 합니다.

- BIOS 설정 수정.
- 부트 관리자를 사용하여 Solaris 분할 영역으로 부트스트랩. 자세한 내용은 부트 관리자를 참조하십시오.

선호되는 방법은 부트 디스크에 Solaris OS를 설치하는 것입니다. 시스템에 여러 운영 체제가 설치되어 있는 경우에는 menu.lst 파일에 항목을 추가할 수 있습니다. 그러면 이 항목은 다음에 시스템을 부트할 때 GRUB 메뉴에 표시됩니다.

여러 운영 체제에 대한 추가 정보는 **System Administration Guide: Basic Administration**의 “How Multiple Operating Systems Are Supported in the GRUB Boot Environment”를 참조하십시오.

x86: 네트워크에서 GRUB 기반 설치 수행

GRUB 기반 네트워크 부트를 수행하려면 PXE 클라이언트용으로 구성된 DHCP 서버와 tftp 서비스를 제공하는 설치 서버가 필요합니다. DHCP 서버는 DHCP 클래스 PXEClient 및 GRUBClient에 응답할 수 있어야 합니다. DHCP 응답에는 다음 정보가 포함되어 있어야 합니다.

- 파일 서버의 IP 주소
- 부트 파일의 이름(pxegrub)

주-서버측에서 네트워크 부트를 수행할 때 보통 요구 사항에 포함되는 rpc.bootparamd는 GRUB 기반 네트워크 부트에 필요하지 않습니다.

PXE 또는 DHCP 서버를 사용할 수 없으면 CD-ROM 또는 로컬 디스크에서 GRUB를 로드할 수 있습니다. 그런 다음 GRUB에서 네트워크를 수동으로 구성하고 파일 서버에서 멀티부트 프로그램과 부트 아카이브를 다운로드할 수 있습니다.

자세한 내용은 **Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 “PXE를 사용하여 네트워크를 통해 부팅 및 설치 개요”**를 참조하십시오.

GRUB 주 메뉴 설명

x86 기반 시스템을 부트하면 GRUB 메뉴가 표시됩니다. 이 메뉴에는 선택할 수 있는 부트 항목 목록이 있습니다. 부트 항목은 시스템에 설치된 OS 인스턴스입니다. GRUB 메뉴는 구성 파일인 `menu.lst` 파일을 기반으로 합니다. `menu.lst` 파일은 Solaris 설치 프로그램에서 만들어지며 설치 후에 수정할 수 있습니다. `menu.lst` 파일은 GRUB 메뉴에 표시되는 OS 인스턴스의 목록을 결정합니다.

- Solaris OS를 설치 또는 업그레이드하면 GRUB 메뉴가 자동으로 업데이트됩니다. 그런 다음 Solaris OS가 새 부트 항목으로 표시됩니다.
- Solaris OS가 아닌 OS를 설치한 경우에는 새 OS 인스턴스가 포함되도록 `menu.lst` 구성 파일을 수정해야 합니다. 새 OS 인스턴스를 추가하면 다음에 시스템을 부트할 때 GRUB 메뉴에 새 부트 항목이 나타납니다.

예 4-1 GRUB 주 메뉴

다음 예에서 GRUB 주 메뉴에는 Solaris 및 Microsoft Windows 운영 체제가 표시됩니다. 이름이 `second_disk`인 Solaris Live Upgrade 부트 환경도 목록에 표시됩니다. 각 메뉴 항목에 대한 설명은 다음을 참조하십시오.

```
GNU GRUB version 0.95 (616K lower / 4127168K upper memory)
```

```
+-----+
|Solaris                                     |
|Solaris failsafe                           |
|second_disk                                 |
|second_disk failsafe                       |
|Windows                                     |
+-----+
```

Use the ^ and v keys to select which entry is highlighted. Press enter to boot the selected OS, 'e' to edit the commands before booting, or 'c' for a command-line.

Solaris	Solaris OS를 지정합니다.
Solaris failsafe	Solaris OS가 손상된 경우 복구에 사용할 수 있는 부트 아카이브를 지정합니다.
second_disk	Solaris Live Upgrade 부트 환경을 지정합니다. <code>second_disk</code> 부트 환경은 Solaris OS의 복사본으로 만들어집니다. 이 환경은 <code>luactivate</code> 명령어를 사용하여 업그레이드 및 활성화할 수 있습니다. 부트 환경은 부트에 사용할 수 있습니다.
Windows	Microsoft Windows OS를 지정합니다. GRUB에서는 이 분할 영역을 감지하지만 OS를 부트할 수 있는지 확인하지는 않습니다.

GRUB menu.lst 파일 설명

GRUB menu.lst 파일에는 GRUB 주 메뉴의 내용이 나열됩니다. GRUB 주 메뉴는 Solaris Live Upgrade 부트 환경을 포함하여 시스템에 설치된 모든 OS 인스턴스의 부트 항목이 나열됩니다. Solaris 소프트웨어 업그레이드 프로세스는 이 파일에서 변경하는 내용을 모두 보존합니다.

menu.lst 파일의 수정 내용은 Solaris Live Upgrade 항목과 함께 모두 GRUB 주 메뉴에 표시됩니다. 파일의 변경 내용은 다음에 시스템을 다시 부트할 때 적용됩니다. 다음과 같은 목적으로 이 파일을 수정할 수 있습니다.

- Solaris가 아닌 운영 체제의 GRUB 메뉴 항목에 추가
- GRUB 메뉴에서 기본 OS 지정과 같은 부트 동작 사용자 정의



주의 - GRUB menu.lst 파일을 사용하여 Solaris Live Upgrade 항목을 수정하지 마십시오. 수정하면 Solaris Live Upgrade가 실패할 수 있습니다.

menu.lst 파일을 사용하여 커널 디버거를 이용한 부트 등의 부트 동작을 사용자 정의할 수 있지만, 사용자 정의에 선호되는 방법은 eeprom 명령어를 사용하는 것입니다. menu.lst 파일을 사용하여 사용자 정의하는 경우 소프트웨어 업그레이드 중에 Solaris OS 항목을 수정할 수 있습니다. 그러면 파일의 변경 사항이 손실됩니다.

eeprom 명령어의 사용 방법에 대한 자세한 내용은 **System Administration Guide: Basic Administration**의 “How to Set Solaris Boot Parameters by Using the eeprom Command”를 참조하십시오.

예 4-2 Menu.lst 파일

다음은 menu.lst 파일 예제입니다.

```
default 0
timeout 10
title Solaris
  root (hd0,0,a)
  kernel /platform/i86pc/multiboot -B console=ttya
  module /platform/i86pc/boot_archive
title Solaris failsafe
  root (hd0,0,a)
  kernel /boot/multiboot -B console=ttya -s
  module /boot/x86.miniroot.safe
#----- second_disk - ADDED BY LIVE UPGRADE - DO NOT EDIT -----
title second_disk
  root (hd0,1,a)
  kernel /platform/i86pc/multiboot
  module /platform/i86pc/boot_archive
title second_disk failsafe
  root (hd0,1,a)
```

예 4-2 Menu.lst 파일 (계속)

```
kernel /boot/multiboot kernel/unix -s
module /boot/x86.miniroot-safe
#----- second_disk ----- END LIVE UPGRADE -----
title Windows
root (hd0,0)
chainloader -1
```

default	시간 초과 값이 만료된 경우 부트할 항목을 지정합니다. 기본값을 변경하려면 번호를 변경하여 목록의 다른 항목을 지정합니다. 첫 번째 제목은 0(영)에서 시작됩니다. 예를 들어 second_disk 부트 환경에서 자동으로 부트하려면 기본값을 2로 변경합니다.
timeout	기본 항목을 부트하기 전에 사용자 입력을 기다리는 시간(초)을 지정합니다. 시간 초과 값을 지정하지 않은 경우에는 항목을 선택해야 합니다.
title OS name	운영 체제의 이름을 지정합니다. <ul style="list-style-type: none"> ■ Solaris Live Upgrade 부트 환경에서 OS name은 새 부트 환경을 만들 때 지정한 이름입니다. 이전 예에서 Solaris Live Upgrade 부트 환경의 이름은 second_disk입니다. ■ 이 값이 비상 안전 부트 아카이브이면 이 부트 아카이브는 기본 OS가 손상된 경우에 복구에 사용됩니다. 이전 예에서 Solaris 비상 안전과 second_disk 비상 안전은 Solaris 및 second_disk 운영 체제의 복구 부트 아카이브입니다.
root (hd0,0,a)	파일을 로드할 디스크, 분할 영역 및 슬라임을 지정합니다. GRUB는 파일 시스템 유형을 자동으로 감지합니다.
kernel /platform/i86pc/multiboot	멀티부트 프로그램을 지정합니다. 멀티부트 프로그램에는 항상 커널 명령어를 사용해야 합니다. multiboot 뒤의 문자열은 해석되지 않고 Solaris OS에 전달됩니다.

여러 운영 체제에 대한 자세한 내용은 **System Administration Guide: Basic Administration**의 “How Multiple Operating Systems Are Supported in the GRUB Boot Environment”를 참조하십시오.

menu.lst 파일을 찾아 GRUB 메뉴 변경

항상 `bootadm` 명령어를 사용하여 GRUB 메뉴의 `menu.lst` 파일을 찾아야 합니다. `list-menu` 하위 명령어는 활성 GRUB 메뉴를 찾습니다. `menu.lst` 파일은 시스템에 설치된 모든 운영 체제의 목록을 표시합니다. 이 파일의 내용에 따라 GRUB 메뉴에 표시되는 운영 체제의 목록이 결정됩니다. 이 파일을 변경하려면 [56 페이지 “x86: GRUB 메뉴의 menu.lst 파일 찾기\(작업\)”](#)를 참조하십시오.

x86: GRUB 메뉴의 menu.lst 파일 찾기(작업)

GRUB 메뉴를 업데이트할 수 있습니다. 예를 들어 기본 OS의 부트 속도를 나타내는 기본 시간을 변경할 수 있습니다. 또는 GRUB 메뉴에 다른 OS를 추가할 수 있습니다.

보통 활성 GRUB 메뉴의 `menu.lst` 파일은 `/boot/grub/menu.lst`에 있습니다. 경우에 따라 GRUB `menu.lst` 파일이 다른 위치에 있을 수도 있습니다. 예를 들어 Solaris Live Upgrade를 사용하는 시스템에서는 GRUB `menu.lst` 파일이 현재 부트 환경을 실행하지 않는 부트 환경에 있을 수 있습니다. 또는 x86 부트 플랫폼으로 시스템을 업그레이드한 경우 `menu.lst` 파일이 `/stubboot` 디렉토리에 있을 수도 있습니다. 시스템 부트에는 활성 GRUB `menu.lst` 파일만 사용됩니다. 시스템을 부트할 때 표시되는 GRUB 메뉴를 수정하려면 활성 GRUB `menu.lst` 파일을 수정해야 합니다. 다른 GRUB `menu.lst` 파일을 변경해도 시스템을 부트할 때 표시되는 메뉴에는 영향을 주지 않습니다. 활성 GRUB `menu.lst` 파일의 위치를 확인하려면 `bootadm` 명령어를 사용합니다. `list-menu` 하위 명령어는 활성 GRUB 메뉴의 위치를 표시합니다. 다음 절차는 GRUB 메뉴의 `menu.lst` 파일 위치를 결정합니다.

`bootadm` 명령어에 대한 자세한 내용은 `bootadm(1M)` 매뉴얼 페이지를 참조하십시오.

▼ GRUB 메뉴의 menu.lst 파일 찾기

다음 절차의 시스템에는 Solaris 및 Solaris Live Upgrade 부트 환경(`second_disk`)의 두 운영 체제가 있습니다. Solaris OS는 부트되었으며 GRUB 메뉴를 포함합니다.

1 슈퍼유저가 되거나 동등한 역할을 맡습니다.

역할에는 권한 부여 및 권한이 있는 명령어가 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 menu.lst 파일을 찾으려면 다음을 입력합니다.

```
# /sbin/bootadm list-menu
```

파일의 위치와 내용이 표시됩니다.

```
The location for the active GRUB menu is: /boot/grub/menu.lst
default 0
timeout 10
```



```
0 Solaris
1 Solaris failsafe
2 second_disk
3 second_disk failsafe
```

▼ 활성 menu.lst 파일이 다른 부트 환경에 있는 경우 GRUB 메뉴의 menu.lst 파일 찾기

다음 절차의 시스템에는 Solaris 및 Solaris Live Upgrade 부트 환경(second_disk)이라는 두 운영 체제가 있습니다. 이 예에서 menu.lst 파일은 현재 실행 중인 부트 환경에 없습니다. second_disk 부트 환경이 부트되었습니다. Solaris 부트 환경에 GRUB 메뉴가 있습니다. Solaris 부트 환경이 마운트되지 않았습니다.

1 수퍼유저가 되거나 동등한 역할을 맡습니다.

역할에는 권한 부여 및 권한이 있는 명령어가 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 menu.lst 파일을 찾으려면 다음을 입력합니다.

```
# /sbin/bootadm list-menu
```

파일의 위치와 내용이 표시됩니다.

```
The location for the active GRUB menu is: /dev/dsk/device_name(not mounted)
The filesystem type of the menu device is <ufs>
default 0
timeout 10
0 Solaris
1 Solaris failsafe
2 second_disk
3 second_disk failsafe
```

3 menu.lst 파일이 있는 파일 시스템이 마운트되지 않았기 때문에 파일 시스템을 마운트해야 합니다. UFS 파일 시스템과 장치 이름을 지정합니다.

```
# /usr/sbin/mount -F ufs /dev/dsk/device_name /mnt
```

*device_name*은 마운트할 부트 환경의 디스크 장치에 있는 루트(/) 파일 시스템의 위치를 지정합니다. 장치 이름은 /dev/dsk/cwtXdysz 형식으로 입력합니다. 예를 들면 다음과 같습니다.

```
# /usr/sbin/mount -F ufs /dev/dsk/c0t1d0s0 /mnt
```

GRUB 메뉴는 /mnt/boot/grub/menu.lst에서 액세스할 수 있습니다.

4 파일 시스템 마운트 해제

```
# /usr/sbin/umount /mnt
```

주- 부트 환경이나 부트 환경의 파일 시스템을 마운트한 경우, 사용이 끝나면 파일 시스템을 마운트 해제해야 합니다. 파일 시스템이 마운트 해제되지 않은 경우 앞으로 그 부트 환경에서 Solaris Live Upgrade 작업이 실패할 수 있습니다.

▼ Solaris Live Upgrade 부트 환경이 마운트된 경우 GRUB 메뉴의 menu.lst 파일 찾기

다음 절차의 시스템에는 Solaris 및 Solaris Live Upgrade 부트 환경(second_disk)이라는 두 운영 체제가 있습니다. second_disk 부트 환경이 부트되었습니다. Solaris 부트 환경에 GRUB 메뉴가 있습니다. Solaris 부트 환경이 /.alt.Solaris에 마운트되었습니다.

1 슈퍼유저가 되거나 동등한 역할을 맡습니다.

역할에는 권한 부여 및 권한이 있는 명령어가 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 menu.lst 파일을 찾으려면 다음을 입력합니다.

```
# /sbin/bootadm list-menu
```

파일의 위치와 내용이 표시됩니다.

```
The location for the active GRUB menu is:
```

```
/.alt.Solaris/boot/grub/menu.lst
```

```
default 0
```

```
timeout 10
```

```
0 Solaris
```

```
1 Solaris failsafe
```

```
2 second_disk
```

```
3 second_disk failsafe
```

GRUB 메뉴가 있는 부트 환경이 이미 마운트되어 있기 때문에

/.alt.Solaris/boot/grub/menu.lst 위치에서 menu.lst 파일에 액세스할 수 있습니다.

▼ 시스템에 x86 부트 분할 영역이 있는 경우 GRUB 메뉴의 menu.lst 파일 찾기

다음 절차의 시스템에는 Solaris 및 Solaris Live Upgrade 부트 환경(second_disk)의 두 운영 체제가 있습니다. second_disk 부트 환경이 부트되었습니다. 시스템이 업그레이드되고 x86

부트 분할 영역이 남았습니다. 부트 분할 영역은 /stubboot에 마운트되어 있으며 GRUB 메뉴가 있습니다. x86 부트 분할 영역에 대한 설명은 46 페이지 “x86: 분할 권장사항”을 참조하십시오.

1 슈퍼유저가 되거나 동등한 역할을 말합니다.

역할에는 권한 부여 및 권한이 있는 명령어가 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 menu.lst 파일을 찾으려면 다음을 입력합니다.

```
# /sbin/bootadm list-menu
```

파일의 위치와 내용이 표시됩니다.

```
The location for the active GRUB menu is:
```

```
/stubboot/boot/grub/menu.lst
```

```
default 0
```

```
timeout 10
```

```
0 Solaris
```

```
1 Solaris failsafe
```

```
2 second_disk
```

```
3 second_disk failsafe
```

/stubboot/boot/grub/menu.lst에서 menu.lst 파일에 액세스할 수 있습니다.

2

사용자 정의 JumpStart 사용

이 부분에서는 사용자 정의 JumpStart 설치를 만들고 준비하고 수행하는 지침을 제공합니다.

사용자 정의 JumpStart(개요)

이 장에서는 사용자 정의 JumpStart 설치 프로세스를 소개하고 간략히 설명합니다.

- 63 페이지 “사용자 정의 JumpStart 소개”
- 64 페이지 “JumpStart 프로그램을 사용하여 Solaris 소프트웨어를 설치하는 방법”

사용자 정의 JumpStart 소개

사용자 정의 JumpStart 설치 방법은 사용자가 만든 프로필을 바탕으로 여러 대의 시스템을 자동으로 설치 또는 업그레이드할 수 있게 하는 명령줄 인터페이스입니다. 이 프로필은 구체적인 소프트웨어 설치 요건을 정의합니다. 셸 스크립트를 통합하여 설치 이전 및 이후 작업을 포함시킬 수 있습니다. 설치나 업그레이드를 위해 어떤 프로필과 스크립트를 사용할 것인지 선택합니다. 사용자 정의 JumpStart 설치 방법은 선택한 프로필 및 스크립트를 기준으로 시스템을 설치 또는 업그레이드합니다. 또한 `sysidcfg` 파일을 사용하여 사용자 정의 JumpStart 설치를 완벽하게 자동화할 수 있도록 구성 정보를 지정할 수 있습니다.

사용자 정의 JumpStart 예제 시나리오

사용자 정의 JumpStart 프로세스는 예로 든 시나리오를 사용하여 설명할 수 있습니다. 이 예로 든 시나리오에서 시스템은 다음 매개 변수로 설정될 수 있습니다.

- 100개의 새 시스템에 Solaris를 설치합니다.
- 70개 시스템이 엔지니어링 그룹 소유이고 개발자용 Solaris OS 그룹을 사용하여 독립형 시스템으로 설치되어야 하는 SPARC 기반 시스템입니다.
- 나머지 30개의 시스템은 마케팅 그룹 소유인 x86 기반이며 최종 사용자용 Solaris OS 소프트웨어 그룹을 사용하여 독립 실행형 시스템으로 설치되어야 합니다.

먼저, 시스템 관리자는 각 시스템 그룹용 프로필 및 `rules` 파일을 만들어야 합니다. `rules` 파일은 Solaris 소프트웨어를 설치하려는 각 시스템 그룹 또는 단일 시스템에 대한 규칙을 포함하는 텍스트 파일입니다. 각 규칙은 하나 이상의 시스템 속성에 기반한 시스템 그룹을 구별합니다. 각 규칙은 또한 각 그룹을 프로필과 연결합니다.

프로필은 Solaris 소프트웨어가 그룹의 각 시스템에 설치되는 방법을 정의하는 텍스트 파일입니다. rules 파일과 프로필은 모두 JumpStart 디렉토리에 위치해야 합니다.

예제 시나리오의 경우, 시스템 관리자는 엔지니어링 그룹용과 마케팅 그룹용의 두 가지 다른 규칙을 포함하는 rules 파일을 만듭니다. 각 규칙의 경우, 시스템의 네트워크 번호는 엔지니어링 그룹과 마케팅 그룹을 구별하는 데 사용됩니다.

각 규칙은 또한 해당 프로필에 대한 링크를 포함합니다. 예를 들어, 엔지니어링 그룹용 규칙에서 링크는 엔지니어링 그룹용으로 작성된 프로필, eng_profile에 추가됩니다. 마케팅 그룹용 규칙에서는 링크가 마케팅 그룹용으로 만든 프로필 market_profile에 추가됩니다.

디스켓 또는 서버에 rules 파일과 프로필을 저장할 수 있습니다.

- 프로필 디스켓은 네트워크에 연결되지 않은 독립 실행형 시스템에서 사용자 정의 JumpStart 설치를 수행하려 할 때 필요합니다.
- 프로필 서버는 서버에 액세스하는 네트워크에 연결된 시스템에서 사용자 정의 JumpStart 설치를 수행하려 할 때 사용됩니다.

rules 파일과 프로필을 만든 후 check 스크립트로 파일을 검증하십시오. check 스크립트가 성공적으로 실행되면 rules.ok 파일이 생성됩니다. rules.ok는 JumpStart 프로그램이 Solaris 소프트웨어를 설치하기 위해 사용하는 rules 파일의 생성된 버전입니다.

JumpStart 프로그램을 사용하여 Solaris 소프트웨어를 설치하는 방법

rules 파일과 프로필을 검증한 후 사용자 정의 JumpStart 설치를 시작할 수 있습니다. JumpStart 프로그램이 rules.ok 파일을 읽습니다. 그런 다음 JumpStart 프로그램이 Solaris 소프트웨어를 설치하려 시도하는 시스템과 일치하는 정의된 시스템 속성을 가진 첫 번째 규칙을 검색합니다. 일치할 경우 JumpStart 프로그램은 시스템에 Solaris 소프트웨어를 설치하기 위해 규칙에 지정된 프로필을 사용합니다.

그림 5-1은 사용자 정의 JumpStart 설치가 네트워크에 연결되지 않은 독립형 시스템에서 작동하는 방법을 보여 줍니다. 시스템 관리자는 Pete의 시스템에서 사용자 정의 JumpStart 설치를 시작합니다. JumpStart 프로그램은 시스템의 디스켓 드라이브에서 디스켓의 rules 파일에 액세스합니다. JumpStart 프로그램은 규칙 2를 시스템에 일치시킵니다. 규칙 2는 JumpStart 프로그램이 Pete의 프로필을 사용하여 Solaris 소프트웨어를 설치할 것을 지정합니다. JumpStart 프로그램은 시스템 관리자가 Pete의 프로필에 지정한 지침을 기준으로 Pete의 프로필을 읽고 Solaris 소프트웨어를 설치합니다.

Pete의 시스템

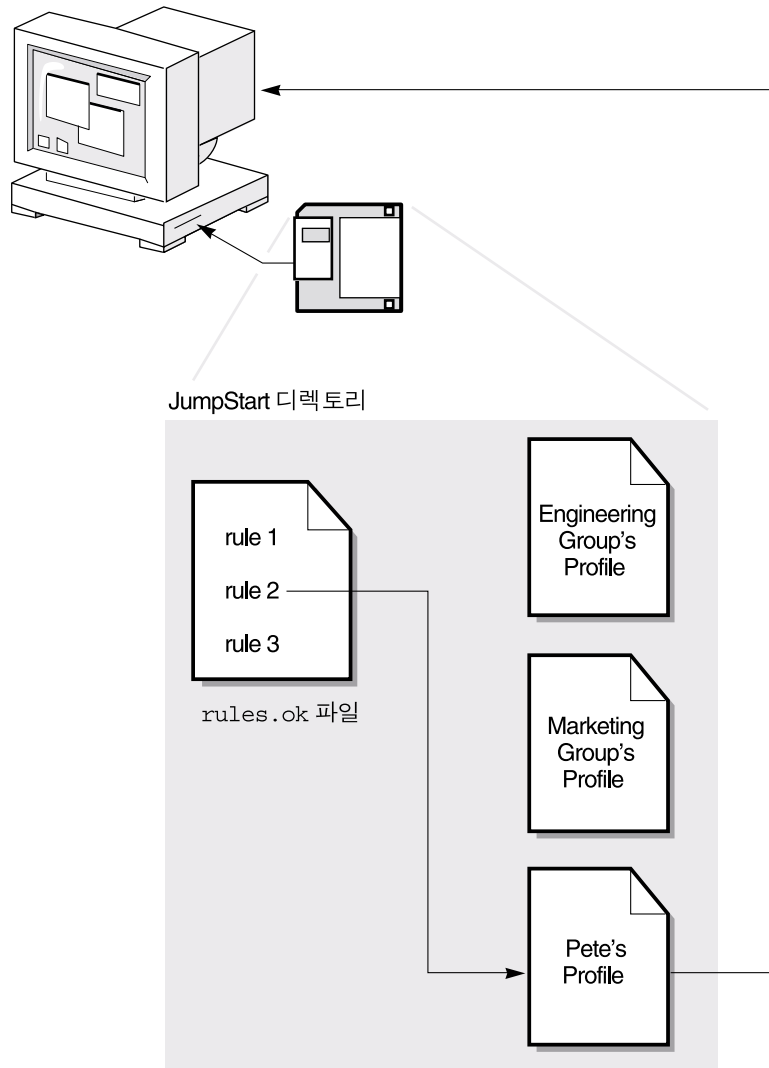


그림 5-1 사용자 정의 JumpStart 설치 작업: 네트워크에 연결되지 않은 예

그림 5-2는 사용자 정의 JumpStart 설치가 네트워크에서 하나 이상의 시스템과 작동하는 방법을 보여 줍니다. 이전에 시스템 관리자는 여러 프로필을 설정하고 단일 서버에 프로필을 저장했습니다. 시스템 관리자는 엔지니어링 시스템 중 하나에서 사용자 정의 JumpStart 설치를 시작합니다. JumpStart 프로그램은 서버의 JumpStart/ 디렉토리에 있는 rules 파일에 액세스합니다. JumpStart 프로그램은 엔지니어링 시스템을 규칙 1에 일치시킵니다. 규칙 1은 JumpStart 프로그램이 Engineering Group's Profile을 사용하여 Solaris 소프트웨어를

설치하도록 지정합니다. JumpStart 프로그램은 시스템 관리자가 Engineering Group's Profile에 지정한 지침을 기준으로 Engineering Group's Profile을 읽고 Solaris 소프트웨어를 설치합니다.

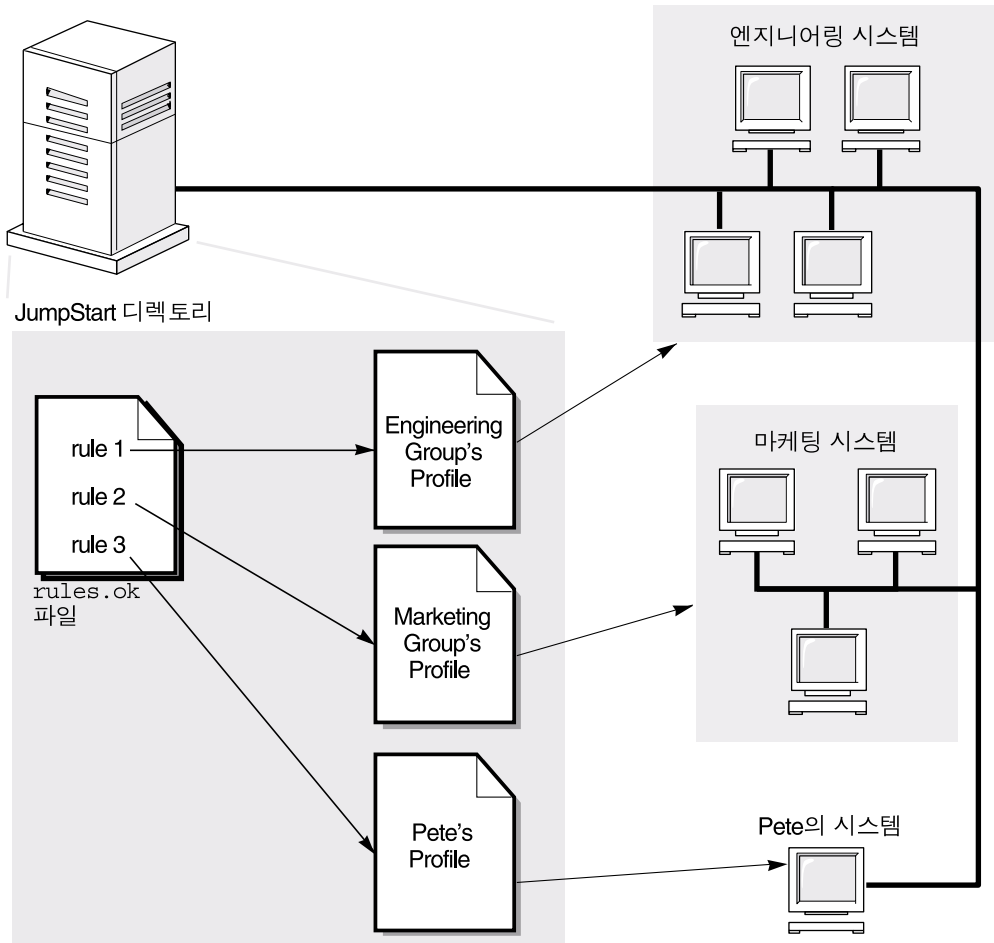


그림 5-2 사용자 정의 JumpStart 설치 작업: 네트워크에 연결된 예

그림 5-3은 JumpStart 프로그램이 사용자 정의 JumpStart 파일을 검색하는 순서를 설명합니다.

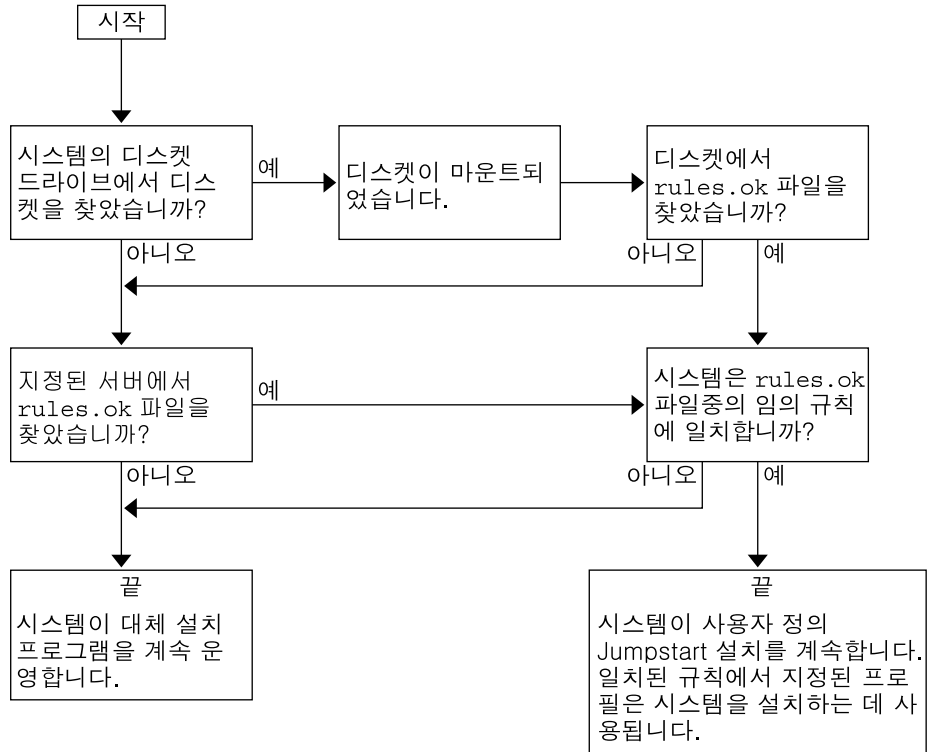


그림 5-3 사용자 정의 JumpStart 설치 중의 작업 진행

사용자 정의 JumpStart 설치 준비(작업)

이 장에서는 사용자 정의 JumpStart 설치 방법을 사용하여 Solaris 소프트웨어를 설치할 사이트에서 시스템을 준비하는 방법에 대한 단계별 지침을 제공합니다.

- 69 페이지 “작업 맵: 사용자 정의 JumpStart 설치 준비”
- 71 페이지 “네트워크 시스템에 대한 프로필 서버 만들기”
- 75 페이지 “독립형 시스템에 대한 프로필 디스켓 만들기”
- 78 페이지 “rules 파일 만들기”
- 81 페이지 “프로필 작성”
- 93 페이지 “프로필 테스트”
- 97 페이지 “rules 파일 검증”

작업 맵: 사용자 정의 JumpStart 설치 준비

표 6-1 작업 맵: 사용자 정의 JumpStart 설치 준비

작업	설명	지침
Solaris 소프트웨어의 이전 버전이 시스템에 설치된 경우 시스템을 업그레이드하는 방법을 결정합니다.	Solaris의 이전 버전이 시스템에 설치된 경우 시스템을 업그레이드하는 방법을 결정해야 합니다. 시스템을 업그레이드하기 전과 업그레이드를 한 후에 수행할 작업을 숙지합니다. 계획을 수립하여 프로필을 만들고 스크립트를 시작하고 종료하도록 합니다.	38 페이지 “업그레이드 계획”

표 6-1 작업 맵: 사용자 정의 JumpStart 설치 준비 (계속)

작업	설명	지침
JumpStart 디렉토리를 만듭니다.	<p>서버에서</p> <p>네트워크에 연결된 시스템에서 사용자 정의 JumpStart 설치를 수행하려면 프로필 서버를 만들어야 합니다. 프로필 서버는 사용자 정의 JumpStart 파일에 대한 JumpStart 디렉토리를 포함합니다.</p> <p>디스켓에서</p> <p>네트워크에 연결되지 않은 시스템에서 사용자 정의 JumpStart 설치를 수행하려면 프로필 디스켓을 만들어야 합니다. 프로필 디스켓은 사용자 정의 JumpStart 파일을 포함합니다.</p>	<p>71 페이지 “네트워크 시스템에 대한 프로필 서버 만들기”</p> <p>75 페이지 “독립형 시스템에 대한 프로필 디스켓 만들기”</p>
rules 파일에 규칙을 추가합니다.	<p>시스템의 각 그룹 또는 단일 시스템을 설치할 방법을 결정 한 후에 설치하려는 각 그룹에 대한 규칙을 만듭니다. 각 규칙은 하나 이상의 시스템 속성을 기준으로 그룹을 구별합니다. 각 규칙은 또한 각 그룹을 프로필과 연결합니다.</p>	78 페이지 “rules 파일 만들기”
모든 규칙에 대한 프로필을 만듭니다.	<p>프로필은 Solaris 소프트웨어, 예를 들어, 소프트웨어 그룹을 시스템에 설치하는 방법을 정의하는 텍스트 파일입니다. 모든 규칙은 규칙이 일치할 때 시스템을 Solaris 소프트웨어와 함께 설치하는 방법을 정의하는 프로필을 지정합니다. 일반적으로 모든 규칙에 대하여 다른 프로필을 만듭니다. 그러나 동일한 프로필이 하나 이상의 규칙에서 사용될 수 있습니다.</p>	81 페이지 “프로필 작성”
(선택 사항) 프로필을 테스트합니다.	<p>프로필을 만든 후 pfinstall(1M) 명령어를 사용하여 시스템의 설치 또는 업그레이드를 위한 프로필을 사용하기 전에 프로필을 테스트합니다.</p>	93 페이지 “프로필 테스트”
rules 파일을 검증합니다.	<p>rules.ok 파일은 JumpStart 프로그램이 프로필과 함께 설치할 시스템을 일치시키는 데 사용하는 rules 파일의 생성된 버전입니다. rules 파일을 검증하는 데 check 스크립트를 사용해야 합니다.</p>	97 페이지 “rules 파일 검증”

네트워크 시스템에 대한 프로필 서버 만들기

네트워크의 시스템에 대한 사용자 정의 JumpStart 설치를 설정할 때 JumpStart 디렉토리라는 서버에 디렉토리를 만들어야 합니다. JumpStart 디렉토리는 모든 핵심 사용자 정의 JumpStart 파일, 예를 들어 rules 파일, rules.ok 파일 및 프로필을 포함합니다. 프로필 서버의 루트(/) 디렉토리에 JumpStart 디렉토리를 저장해야 합니다.

JumpStart 디렉토리를 포함하는 서버를 프로필 서버라고 합니다. 프로필 서버는 설치 서버 또는 부트 서버와 동일한 시스템일 수 있고 아니면 완전히 다른 서버일 수 있습니다. 프로필 서버에서 다른 플랫폼에 대하여 사용자 정의 JumpStart 파일을 제공할 수 있습니다. 예를 들어, x86 서버는 SPARC 기반 시스템 및 x86 기반 시스템에 사용자 정의 JumpStart 파일을 제공할 수 있습니다.

주 - 프로필 서버를 만든 후 시스템이 서버에 액세스하도록 해야 합니다. 자세한 지침은 73 페이지 “모든 시스템에 프로필 서버 액세스 허용”를 참조하십시오.

▼ 서버에 JumpStart 디렉토리 만들기

주 - 이 절차는 시스템이 볼륨 관리자를 실행 중이라고 가정합니다. 볼륨 관리자를 사용하여 디스크를 관리하지 않은 경우 볼륨 관리자 없이 이동식 매체를 관리하는 방법에 대한 자세한 내용은 **System Administration Guide: Devices and File Systems**을 참조하십시오.

1 JumpStart 디렉토리를 만들려는 서버에 수퍼유저로 로그인합니다.

2 서버의 어느 위치에든 JumpStart 디렉토리를 만듭니다.

```
# mkdir -m 755 jumpstart_dir_path
```

명령어에서 *jumpstart_dir_path*는 JumpStart 디렉토리의 절대 경로입니다.

예를 들어, 다음 명령어는 루트 디렉토리(/)에서 jumpstart라고 하는 디렉토리를 작성하고 사용 권한을 755로 설정합니다.

```
# mkdir -m 755 /jumpstart
```

3 다음 항목을 추가하여 /etc/dfs/dfstab 파일을 편집합니다.

```
share -F nfs -o ro,anon=0 jumpstart_dir_path
```

예를 들어, 다음 항목은 /jumpstart 디렉토리를 공유합니다.

```
share -F nfs -o ro,anon=0 /jumpstart
```

4 shareall을 입력하고 Enter 키를 누릅니다.

5 사용자 정의 JumpStart 파일의 예를 JumpStart 디렉토리에 복사할지 여부를 결정합니다.

- 아니오, 단계 8로 이동합니다.
- 복사하려면 다음 결정 테이블을 사용하여 다음 해야 할 작업을 결정합니다.

예 위치	지침
플랫폼에 대한 Solaris Operating System DVD 또는 Solaris Software - 1 CD	Solaris Operating System DVD 또는 Solaris Software - 1 CD를 서버의 CD-ROM 드라이브에 삽입합니다. 볼륨 관리자가 자동으로 CD를 마운트합니다.
로컬 디스크의 플랫폼에 대한 Solaris Operating System DVD 또는 Solaris Software - 1 CD의 이미지	Solaris Operating System DVD 또는 Solaris Software - 1 이미지의 위치로 디렉토리를 변경합니다. 예를 들어, 다음 명령어를 입력합니다. cd /export/install

6 사용자 정의 JumpStart 예제 파일을 프로필 서버의 JumpStart 디렉토리에 복사합니다.

```
# cp -r media_path/Solaris_10_606/Misc/jumpstart_sample/* jumpstart_dir_path
```

media_path 로컬 디스크의 CD, DVD 또는 이미지의 경로

jumpstart_dir_path 사용자 정의 JumpStart 파일을 배치할 프로필 서버의 경로

예를 들어, 다음 명령어는 프로필 서버에서 *jumpstart_sample* 디렉토리를 */jumpstart* 디렉토리로 복사합니다.

- SPARC 기반 시스템의 경우:

```
cp -r /cdrom/cdrom0/s0/Solaris_10_606/Misc/jumpstart_sample/* /jumpstart
```

- x86 기반 시스템의 경우:

```
cp -r /cdrom/cdrom0/Solaris_10_606/Misc/jumpstart_sample/* /jumpstart
```

7 파일이 사용자의 환경에서 작동하도록 예들 든 JumpStart 파일을 업데이트합니다.

8 root가 JumpStart 디렉토리를 소유하고 권한이 755로 설정되도록 합니다.

9 네트워크의 시스템이 프로필 서버에 액세스하도록 합니다.

자세한 지침은 73 페이지 “모든 시스템에 프로필 서버 액세스 허용”를 참조하십시오.

프로필 서버에 모든 시스템 액세스 허용

프로필 서버를 작성할 때 시스템이 사용자 정의 JumpStart 설치 중에 프로필 서버의 JumpStart 디렉토리에 액세스할 수 있도록 해야 합니다. 다음 방법 중 하나를 사용하여 액세스를 확보합니다.

명령어 또는 파일	액세스 제공	지침
add_install_client 명령어	네트워크 설치를 위해 시스템을 추가할 때마다 add_install_client 명령어와 함께 -c 옵션을 사용하여 프로필 서버를 지정합니다. 주-NFS를 사용하지 않는 경우에는 다른 방법으로 액세스를 확보해야 합니다. <ul style="list-style-type: none"> ■ SPARC 기반 시스템의 경우, boot 명령어를 사용합니다. ■ x86 기반 시스템의 경우, GRUB 메뉴를 편집합니다. 	<ul style="list-style-type: none"> ■ DVD 매체의 경우 Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 “DVD 이미지로 네트워크에서 설치할 시스템 추가”를 참조하십시오. ■ CD 매체의 경우 Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 “CD 이미지로 네트워크에서 설치할 시스템 추가”를 참조하십시오.
시스템을 부트할 때 JumpStart 디렉토리 위치 지정	<ul style="list-style-type: none"> ■ SPARC 기반 시스템의 경우, boot 명령어를 사용하여 시스템을 부트합니다. 시스템을 부트할 때 프로필 서버에 JumpStart 디렉토리의 위치를 지정합니다. 사용자 정의 JumpStart 구성 파일을 하나의 파일로 압축해야 합니다. 그런 다음 압축된 구성 파일을 HTTP 또는 HTTPS 서버에 저장합니다. ■ x86 기반 시스템의 경우, 시스템을 부트할 때 GRUB 메뉴의 부트 항목을 편집하여 프로필 서버에 JumpStart 디렉토리 위치를 지정합니다. 사용자 정의 JumpStart 구성 파일을 하나의 파일로 압축해야 합니다. 그런 다음 압축된 구성 파일을 HTTP 또는 HTTPS 서버에 저장합니다. GRUB 메뉴 항목을 편집할 때 압축 파일의 위치를 지정합니다. 	<ul style="list-style-type: none"> ■ 107 페이지 “압축된 구성 파일 만들기” ■ 125 페이지 “SPARC: 사용자 정의 JumpStart 프로그램을 사용하여 설치 또는 업그레이드 수행하기”의 단계 5 ■ 107 페이지 “압축된 구성 파일 만들기” ■ 131 페이지 “x86: GRUB 부트 명령어를 편집하여 사용자 정의 JumpStart 설치 수행”
/etc/bootparams 파일	/etc/bootparams 파일에 와일드카드를 추가합니다.	73 페이지 “모든 시스템에 프로필 서버 액세스 허용”

▼ 모든 시스템에 프로필 서버 액세스 허용

다음 절차는 네트워크 설치 정보를 다음 위치에 저장하는 경우에만 사용합니다.

- /etc/bootparams 파일
- bootparams 데이터베이스의 이름 서비스. bootparams 데이터베이스를 업데이트하려면 단계 3에 표시된 항목을 추가합니다.

다음 절차를 사용하는 경우에는 모두 SPARC 시스템인 경우와 같이 시스템이 같은 유형이어야 합니다.

다음 조건에서는 이 절차를 사용하지 마십시오.

- JumpStart 디렉토리를 디스켓에 저장하는 경우

- 시스템을 부트할 때 프로필 서버의 위치를 지정하는 경우구조가 다른 시스템이 있는 경우에는 시스템을 부트할 때 프로필 서버의 위치를 지정해야 합니다.

위의 조건에서는 SPARC boot 명령어를 사용하거나 x86 GRUB 메뉴를 사용합니다.

주 - 네트워크 설치 정보를 DHCP 서버에 저장할 수도 있습니다.

- **SPARC** 기반 시스템의 경우 `add_install_client` 명령어와 `-d` 옵션을 사용하여 사용자 정의 JumpStart 프로그램이 DHCP 서버를 사용하도록 지정합니다. 또는 `boot` 명령어와 `dhcp` 옵션을 사용하여 사용자 정의 JumpStart 프로그램이 DHCP 서버를 사용하도록 지정합니다. 이 옵션의 사용 지침은 [126 페이지 “SPARC: boot 명령어의 명령어 참조”](#)를 참조하십시오.
- **x86** 기반 시스템의 경우, 다음 중 한 가지 방법으로 `dhcp`를 사용합니다.
 - 설치 서버를 사용하는 경우 `add_install_client` 명령어 및 `-d` 옵션을 사용하여 사용자 정의 JumpStart 프로그램이 PXE가 있는(혹은 PXE를 갖춘) DHCP서버와 PXE를 사용하도록 지정합니다.
 - GRUB 메뉴에서 GRUB 항목을 편집하고 DHCP 옵션을 추가할 수 있습니다. GRUB 항목 편집에 대한 지침은 [131 페이지 “x86: GRUB 부트 명령어를 편집하여 사용자 정의 JumpStart 설치 수행”](#)을 참조하십시오.

- 1 설치 또는 부트 서버에서 슈퍼유저로 로그인합니다.
- 2 텍스트 편집기를 사용하여 `/etc/bootparams`를 엽니다.
- 3 이 항목을 추가합니다.

```
* install_config=server:jumpstart_dir_path
```

* 모든 시스템이 액세스 권한이 있음을 지정하는 와일드카드 문자

<code>server</code>	JumpStart 디렉토리가 위치하는 프로필 서버의 호스트 이름
<code>jumpstart_dir_path</code>	JumpStart 디렉토리의 절대 경로

예를 들어, 다음 항목을 사용하여 모든 시스템이 이름이 `sherlock`인 프로필 서버의 `/jumpstart` 디렉토리에 액세스합니다.

```
* install_config=sherlock:/jumpstart
```



주의 - 이 절차를 사용하면 설치 클라이언트가 부트될 때 다음 오류 메시지를 생성합니다.

```
WARNING:getfile:RPC failed:error 5:(RPC Timed out).
```

[213 페이지 “네트워크에서 부트할 때 나타나는 오류 메시지”](#)에는 이 오류 메시지에 대한 자세한 설명이 있습니다.

모든 시스템이 이제 프로필 서버에 액세스할 수 있습니다.

독립형 시스템에 대한 프로필 디스켓 만들기

JumpStart 디렉토리를 포함하는 디스켓을 프로필 디스켓이라고 합니다. 네트워크에 연결되지 않은 시스템은 프로필 서버에 액세스할 수 없습니다. 결과적으로, 시스템이 네트워크에 연결되지 않은 경우 디스켓에서 JumpStart 디렉토리를 만들어야 합니다. 프로필 디스켓을 만든 시스템은 디스켓 드라이브가 있습니다.

JumpStart 디렉토리는 모든 핵심 사용자 정의 JumpStart 파일(예를 들어, rules 파일, rules.ok 파일) 및 프로필을 포함합니다. 프로필 디스켓의 루트(/) 디렉토리에 JumpStart 디렉토리를 저장해야 합니다.

다음 절차 중 하나를 참조하십시오.

- 75 페이지 “SPARC: 프로필 디스켓 만들기”
- 76 페이지 “x86: GRUB를 사용하여 프로필 디스켓 만들기”

▼ SPARC: 프로필 디스켓 만들기

주 - 이 절차는 시스템이 볼륨 관리자를 실행 중이라고 가정합니다. 디스켓, CD 및 DVD를 관리하는 데 볼륨 관리자를 사용하지 않는 경우 볼륨 관리자 없이 이동식 매체 관리에 대한 자세한 정보는 **System Administration Guide: Devices and File Systems**를 참조하십시오.

1 디스켓 드라이브가 연결된 SPARC 기반 시스템에서 슈퍼유저로 로그인합니다.

2 디스켓 드라이브에 덮어쓸 수 있는 공백 디스켓 또는 디스켓을 삽입합니다.

3 디스켓을 마운트합니다.

```
# volcheck
```

4 디스켓이 UNIX 파일 시스템(UFS)을 포함하는지 결정합니다.

다음과 같은 항목에 대하여 시스템의 /etc/mnttab 파일 내용을 조사합니다.

```
/vol/dev/diskette0/scrap /floppy/scrap ufs suid,rw,largefiles,dev=1740008 927147040
```

- 항목이 존재하면 단계 6로 이동합니다.
- 항목이 존재하지 않으면 다음 단계로 이동합니다.

5 디스켓에 UFS를 만듭니다.

```
# newfs /vol/dev/aliases/floppy0
```

6 사용자 정의 JumpStart 파일의 예를 JumpStart 디렉토리에 복사할지 여부를 결정합니다.

- 복사하지 않을 경우 단계 9로 이동합니다.
- 복사하려면 다음 결정 테이블을 사용하여 다음 해야 할 작업을 결정합니다.

예 위치	지침
Solaris Operating System for SPARC Platforms DVD 또는 Solaris Software for SPARC Platforms - 1 CD	Solaris Operating System for SPARC Platforms DVD 또는 Solaris Software for SPARC Platforms - 1 CD를 서버의 CD-ROM 드라이브에 삽입합니다. 볼륨 관리자가 자동으로 CD를 마운트합니다.
로컬 디스크의 Solaris Operating System for SPARC Platforms DVD 또는 Solaris Software for SPARC Platforms - 1 CD 이미지	디렉토리를 Solaris Operating System for SPARC Platforms DVD 또는 Solaris Software for SPARC Platforms - 1 CD 이미지의 위치로 변경합니다. 예를 들어, 다음 명령어를 입력합니다. <code>cd /export/install</code>

7 사용자 정의 JumpStart 예제 파일을 프로필 디스켓의 JumpStart 디렉토리에 복사합니다.

```
# cp -r media_path/Solaris_10_606/Misc/jumpstart_sample/* jumpstart_dir_path
```

media_path 로컬 디스크의 CD, DVD 또는 이미지의 경로
jumpstart_dir_path 사용자 정의 JumpStart 파일을 배치하려는 프로필 디스켓의 경로

주 - 디스켓의 루트(/) 디렉토리에 사용자 정의 JumpStart 설치 파일을 배치해야 합니다.

예를 들어, 다음 명령어는 Solaris Software for SPARC Platforms - 1 CD의 *jumpstart_sample*의 내용을 이름이 *scrap*인 프로필 디스켓의 루트(/) 디렉토리로 복사합니다.

```
cp -r /cdrom/sol_10_606_sparc/s0/Solaris_10_606/Misc/jumpstart_sample/* /floppy/scrap
```

8 파일이 환경에서 작동하도록 프로필 디스켓의 예로 든 JumpStart 파일을 업데이트합니다.

9 root가 JumpStart 디렉토리를 소유하고 권한을 755로 설정하도록 합니다.

10 디스켓을 꺼냅니다.

```
# eject floppy
```

프로필 디스켓 만들기를 완료하였습니다. 이제 *rules* 파일을 업데이트하고 프로필 디스켓에서 프로필을 만들어 사용자 정의 JumpStart 설치를 수행할 수 있습니다. 계속하려면 78 페이지 “*rules* 파일 만들기”로 이동합니다.

▼ x86: GRUB를 사용하여 프로필 디스켓 만들기

이 절차에 따라 GRUB로 프로필 디스켓을 만듭니다. GRUB 메뉴는 부트 프로세스를 활성화하는 설치 절차 중에 제공됩니다. GRUB 메뉴는 이전 릴리스에서 시스템을 부트하는데 필요했던 Solaris Device Configuration Assistant 을 대신합니다.

주 - 이 절차는 시스템이 볼륨 관리자를 실행 중이라고 가정합니다. 디스켓, CD 및 DVD를 관리하는 데 볼륨 관리자를 사용하지 않는 경우 볼륨 관리자 없이 이동식 매체를 관리하는 방법에 대한 자세한 정보는 **System Administration Guide: Devices and File Systems**를 참조하십시오.

- 1 디스켓 드라이브가 연결된 x86 기반 시스템에서 수퍼유저로 로그인합니다.
- 2 디스켓 드라이브에 덮어쓸 수 있는 공백 디스켓 또는 디스켓을 삽입합니다.
- 3 디스켓을 마운트합니다.

```
# volcheck
```

- 4 사용자 정의 JumpStart 파일의 예를 JumpStart 디렉토리에 복사할지 여부를 결정합니다.

- 복사하지 않을 경우 단계 7로 이동합니다.
- 복사하려면 다음 결정 테이블을 사용하여 다음 해야 할 작업을 결정합니다.

예 위치	지침
Solaris Operating System for x86 Platforms DVD 또는 Solaris Software for x86 Platforms - 1 CD	Solaris Operating System for x86 Platforms DVD 또는 Solaris Software for x86 Platforms - 1 CD를 서버의 CD-ROM 드라이브에 삽입합니다. 볼륨 관리자가 자동으로 CD를 마운트합니다.
로컬 디스크의 Solaris Operating System for x86 Platforms DVD 또는 Solaris Software for x86 Platforms - 1 CD 이미지	Solaris Operating System for x86 Platforms DVD 또는 Solaris Software for x86 Platforms - 1 CD 이미지의 위치로 디렉토리를 변경합니다. 예를 들어, 다음 명령어를 입력합니다. cd /export/install

- 5 사용자 정의 JumpStart 예제 파일을 프로필 디스켓의 JumpStart 디렉토리에 복사합니다.

```
# cp -r media_path/Solaris_10_606/Misc/jumpstart_sample/* jumpstart_dir_path
```

media_path 로컬 디스크의 CD, DVD 또는 이미지의 경로

jumpstart_dir_path 사용자 정의 JumpStart 파일을 배치하려는 프로필 디스켓의 경로

주 - 프로필 디스켓의 루트(/) 디렉토리에 모든 사용자 정의 JumpStart 설치 파일을 배치해야 합니다.

예를 들어, 다음 명령은 Solaris Software for x86 Platforms - 1 CD의 jumpstart_sample의 내용을 이름이 scrap인 프로필 디스켓의 루트(/) 디렉토리로 복사합니다.

```
cp -r /cdrom/sol_10_606_x86/Solaris_10_606/Misc/jumpstart_sample/* /floppy/scrap
```

- 6 파일이 환경에서 작동하도록 프로필 디스켓의 예로 든 JumpStart 파일을 업데이트합니다.
- 7 root가 JumpStart 디렉토리를 소유하고 권한을 755로 설정하도록 합니다.
- 8 파일 관리자 창의 디스크 꺼내기를 누르거나 명령줄에 eject floppy를 입력하여 디스켓을 꺼냅니다.
- 9 이동식 매체 관리자 대화 상자에서 OK를 누릅니다.
- 10 수동으로 디스켓을 꺼냅니다.

참조 프로필 디스켓 만들기를 완료하였습니다. 이제 rules 파일을 업데이트하고 프로필 디스켓에서 프로필을 만들어 사용자 정의 JumpStart 설치를 수행할 수 있습니다. 계속하려면 78 페이지 “rules 파일 만들기”로 이동합니다.

rules 파일 만들기

rules 파일은 Solaris OS를 설치하고자 하는 각 시스템 그룹에 대한 규칙을 포함하는 텍스트 파일입니다. 각 규칙은 하나 이상의 시스템 속성에 기반한 시스템 그룹을 구별합니다. 각 규칙은 또한 각 그룹을 프로필과 연결합니다. 프로필은 Solaris 소프트웨어가 그룹의 각 시스템에 설치되는 방법을 정의하는 텍스트 파일입니다. 예를 들어, 다음 규칙은 JumpStart 프로그램이 basic_prof 프로필의 정보를 사용하여 sun4u 플랫폼 그룹을 가진 모든 시스템을 설치할 것을 지정합니다.

```
karch sun4u - basic_prof -
```

rules 파일은 사용자 정의 JumpStart 설치에 필요한 rules.ok 파일을 만드는 데 사용됩니다.

주 - 75 페이지 “독립형 시스템에 대한 프로필 디스켓 만들기” 또는 71 페이지 “네트워크 시스템에 대한 프로필 서버 만들기”의 절차에 따라 JumpStart 디렉토리를 설정할 경우, rules 예제 파일이 이미 JumpStart 디렉토리에 있습니다. 샘플 rules 파일은 설명서와 예로 든 일부 규칙을 포함합니다. 샘플 rules 파일을 사용하는 경우 사용하지 않을 예로 든 규칙을 주석 처리합니다.

rules 파일의 구분

rules 파일은 다음 속성을 가져야 합니다.

- 파일 이름은 rules로 지정해야 합니다.
- 파일은 적어도 하나의 규칙을 포함해야 합니다.

rules 파일은 다음을 포함할 수 있습니다.

- 주석이 달린 텍스트
JumpStart에서는 행에서 # 기호 뒤에 나오는 모든 텍스트를 주석으로 처리합니다. 행이 # 기호로 시작되면 전체 행은 주석으로 처리됩니다.
- 하나 이상의 공백 행
- 하나 이상의 다중 행 규칙
하나의 규칙을 새 행에 계속하려면 Return 키를 누르기 전에 백슬래시 문자(\)를 넣습니다.

▼ rules 파일 만들기

- 1 텍스트 편집기를 사용하여 `rules`라는 이름의 텍스트 파일을 만듭니다. 또는 만든 JumpStart 디렉토리에서 샘플 `rules` 파일을 엽니다.
- 2 Solaris 소프트웨어를 설치하려는 각 시스템 그룹에 대하여 `rules` 파일에 규칙을 추가합니다. `rules` 파일 키워드 및 값 목록은 145 페이지 “규칙 키워드 및 값”을 참조하십시오. `rules` 파일의 규칙은 다음 구문을 지켜야 합니다.

```
!rule_keyword rule_value && !rule_keyword rule_value ... begin profile finish
```

! 부정을 나타내기 위해 키워드 앞에 사용되는 기호

`rule_keyword` 호스트 이름, `hostname` 또는 메모리 크기, `memsizesize`와 같은 일반 시스템 속성을 설명하는 미리 정의된 어휘 단위 또는 단어 `rule_keyword`는 같은 속성을 가진 시스템을 프로필에 일치시키는 규칙 값과 함께 사용됩니다. 규칙 키워드의 목록은 145 페이지 “규칙 키워드 및 값”을 참조하십시오.

`rule_value` 해당하는 규칙 키워드에 특정 시스템 속성을 제공하는 값. 규칙 값은 145 페이지 “규칙 키워드 및 값”에서 설명합니다.

&& 동일한 규칙에서 규칙 키워드와 규칙 값 쌍을 결합하기 위해 사용해야 하는 기호(논리 AND). 사용자 정의 JumpStart 설치 중에 시스템은 규칙이 일치되기 전에 규칙의 모든 쌍을 일치시켜야 합니다.

`begin` 설치가 시작되기 전에 실행될 수 있는 선택적 Bourne 셸 스크립트의 이름. 시작 스크립트가 존재하지 않으면 이 필드에 마이너스 기호(-)를 입력해야 합니다. 모든 시작 스크립트는 JumpStart 디렉토리에 위치해야 합니다.

시작 스크립트를 작성하는 방법에 대한 정보는 99 페이지 “시작 스크립트 만들기”에서 제시됩니다.

`profile` 시스템이 규칙과 일치할 때 Solaris 소프트웨어가 시스템에 설치되는 방법을 정의하는 텍스트 파일의 이름. 프로필의 정보는 프로필 키워드와 해당하는 프로필 값으로 구성됩니다. 모든 프로필은 JumpStart 디렉토리에 위치해야 합니다.

주-프로필 필드를 사용하는 선택적 방법은 113 페이지 “사이트별 설치 프로그램 사용” 및 100 페이지 “시작 스크립트를 사용하여 파생된 프로필 작성”에서 설명합니다.

finish

설치가 완료된 후 실행될 수 있는 선택적 Bourne 셸 스크립트의 이름. 종료 스크립트가 존재하지 않으면 이 필드에 마이너스 기호(-)를 입력해야 합니다. 모든 종료 스크립트는 JumpStart 디렉토리에 위치해야 합니다.

종료 스크립트를 만드는 방법에 대한 정보는 101 페이지 “종료 스크립트 만들기”에서 설명합니다.

최소한 각 규칙은 다음을 포함해야 합니다.

- 키워드, 값 및 해당 프로필
- 시작 또는 종료 스크립트가 지정되지 않은 경우 *begin* 및 *finish* 필드의 마이너스 기호(-)

3 JumpStart 디렉토리에 rules 파일을 저장합니다.

4 root가 rules 파일을 소유하고 권한이 644로 설정되도록 하십시오.

rules 파일의 예

다음 예는 rules 파일의 예로 든 여러 규칙을 보여 줍니다. 각 행은 규칙 키워드와 해당 키워드에 대한 유효한 값을 가집니다. JumpStart 프로그램이 rules 파일을 맨 위에서 맨 아래로 스캔합니다.

JumpStart 프로그램이 알려진 시스템과 규칙 키워드 및 값을 일치시키면 JumpStart 프로그램은 프로필 필드에 나열된 프로필이 지정하는 Solaris 소프트웨어를 설치합니다.

예 6-1 rule 파일

```
# rule keywords and rule values      begin script      profile      finish script
# -----
hostname eng-1                       -                basic_prof   -
network 192.168.255.255 && !model \
'SUNW,Sun-Blade-100'                  -                net_prof     -
model SUNW,SPARCstation-LX           -                lx_prof      complete
network 192.168.2.0 && karch i86pc setup  x86_prof      done
memsize 64-128 && arch i386           -                prog_prof    -
any -                                  -                generic_prof -
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

hostname 시스템의 호스트 이름이 *eng-1*이면 규칙과 일치합니다. *basic_prof* 프로필은 규칙과 일치하는 시스템에 Solaris 소프트웨어를 설치하는 데 사용됩니다.

예 6-1 rule 파일 (계속)

network	시스템이 서브넷 192.168.255.255에 있고 시스템이 Sun Blade™ 100(SUNW,Sun-Blade-100)이 아니면 규칙이 일치합니다.net_prof 프로파일은 이 규칙과 일치하는 시스템에 Solaris 소프트웨어를 설치하는 데 사용됩니다. 규칙은 78 페이지 “rules 파일의 구문”에 정의된 규칙 줄바꿈의 예도 제공합니다.
model	시스템이 SPARCstation LX이면 규칙과 일치합니다.lx_prof 프로파일 및 complete 종료 스크립트는 이 규칙과 일치하는 시스템에 Solaris 소프트웨어를 설치하는데 사용됩니다.
network	규칙은 시스템이 서브넷 193.168.2.0에 있고 x86 기반 sun4u 시스템인 경우 일치합니다.setup 시작 스크립트, x864u_prof 프로파일 및 done 종료 스크립트는 규칙이 일치하는 시스템에 Solaris 소프트웨어를 설치하는 데 사용됩니다.
memsize	시스템이 64MB - 128MB 메모리를 보유하고 x86 기반 시스템이면 규칙이 일치합니다.prog_prof 프로파일은 규칙과 일치하는 시스템에 Solaris 소프트웨어를 설치하는 데 사용됩니다.
any	규칙은 이전 규칙과 일치하지 않은 시스템과 일치합니다.generic_prof 프로파일은 규칙과 일치하는 시스템에 Solaris 소프트웨어를 설치하는 데 사용됩니다.any가 사용되면 항상 rules 파일의 마지막이어야 합니다.

프로필 작성

프로파일은 시스템에 Solaris 소프트웨어를 설치하는 방법을 정의하는 텍스트 파일입니다. 프로파일은 설치의 요소, 예를 들어, 설치할 소프트웨어 그룹을 정의합니다. 모든 규칙은 시스템이 설치될 방법을 정의하는 프로파일을 지정합니다. 모든 규칙에 대하여 다른 프로파일을 만들거나 동일한 프로파일 하나 이상의 규칙에서 사용될 수 있습니다.

프로파일은 하나 이상의 프로파일 키워드와 해당 값으로 구성됩니다. 각 프로파일 키워드는 JumpStart 프로그램이 시스템에 Solaris 소프트웨어를 설치하는 방법의 한 측면을 제어하는 명령입니다. 예를 들어, 다음 프로파일 키워드와 값은 JumpStart 프로그램이 시스템을 서버로 설치함을 지정합니다.

```
system_type server
```

주 - 샘플 프로파일은 다음 절차 중 하나를 사용하여 JumpStart 디렉토리를 만든 경우 이미 JumpStart 디렉토리에 있습니다.

- 71 페이지 “네트워크 시스템에 대한 프로파일 서버 만들기”
- 75 페이지 “독립형 시스템에 대한 프로파일 디스켓 만들기”

프로필의 구문

프로필은 다음을 포함합니다.

- 첫 번째 항목으로서 `install_type` 프로필 키워드
- 해당 하나의 키워드
- 프로필에 의해 업그레이드 중인 시스템이 업그레이드할 수 있는 하나 이상의 루트(/) 파일 시스템을 포함하는 경우 `root_device` 키워드

프로필은 다음을 포함할 수 있습니다.

- 주석이 달린 텍스트
 - 행에서 # 기호 다음에 포함되는 모든 텍스트는 JumpStart 프로그램에 의해 주석이 달린 텍스트로 처리됩니다. 행이 # 기호로 시작되면 전체 행은 주석으로 처리됩니다.
- 하나 이상의 공백 행

▼ 프로필 작성

- 1 텍스트 편집기를 사용하여 텍스트 파일을 만듭니다. 파일의 이름을 자세하게 지정합니다. 또는 만든 JumpStart 디렉토리에서 샘플 프로필을 엽니다.

주 - 프로필의 이름이 시스템에 Solaris 소프트웨어를 설치하기 위해 프로필을 사용하는 방법을 반영하도록 합니다. 예를 들어 프로필을 `basic_install`, `eng_profile` 또는 `user_profile`로 이름 지정할 수 있습니다.

- 2 프로필 키워드와 값을 프로필에 추가합니다.
 - 프로필 키워드 및 값의 목록은 149 페이지 “프로필 키워드 및 값”을 참조하십시오.

주 - 프로필 키워드와 해당 값은 대소문자를 구분합니다.

- 3 JumpStart 디렉토리에 프로필을 저장합니다.
- 4 `root`가 프로필을 소유하고 권한이 644로 설정되도록 합니다.
- 5 (선택 사항) 프로필을 테스트합니다.
 - 93 페이지 “프로필 테스트”는 프로필 테스트에 대한 정보를 포함합니다.

프로필의 예

다음 프로필 예는 서로 다른 여러 프로필 키워드 및 프로필 값을 사용하여 시스템의 Solaris 소프트웨어 설치를 제어하는 방법을 보여 줍니다. 149 페이지 “프로필 키워드 및 값”은 프로필 키워드 및 값에 대한 설명을 포함합니다.

예 6-2 원격 파일 시스템 마운트와 패키지 추가 및 삭제

```
# profile keywords      profile values
# -----
install_type           initial_install
system_type            standalone
partitioning           default
fileys                 any 512 swap # specify size of /swap
cluster                SUNWCprog
package                SUNWman delete
cluster                SUNWCacc
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

<code>install_type</code>	<code>install_type</code> 키워드는 모든 프로필에서 필수입니다.
<code>system_type</code>	<code>system_type</code> 키워드는 시스템이 독립형 시스템으로 설치될 것을 정의합니다.
<code>partitioning</code>	파일 시스템 슬라이스는 값 <code>default</code> 로 설치될 소프트웨어에 의해 결정됩니다. <code>swap</code> 의 크기는 512MB로 설정되고 모든 디스크에서 값 <code>any</code> 로 설치됩니다.
<code>cluster</code>	개발자 Solaris 소프트웨어 그룹 <code>SUNWCprog</code> 는 시스템에 설치됩니다.
<code>package</code>	표준 매뉴얼 페이지가 네트워크 상의 <code>s_ref</code> 파일 시스템 서버에서 마운트될 경우 매뉴얼 페이지 패키지는 시스템에 설치되지 않습니다. 시스템 계정 유틸리티를 포함하는 패키지는 시스템에 설치될 것으로 선택됩니다.

예 6-3 원격 파일 시스템 마운트 및 타사 패키지 추가

```
# profile keywords      profile values
# -----
install_type           initial_install
system_type            standalone
partitioning           default
fileys                 any 512 swap # specify size of /swap
cluster                SUNWCprog
cluster                SUNWCacc
package                apache_server \
```

예 6-3 원격 파일 시스템 마운트 및 타사 패키지 추가 (계속)

```
http://package.central/packages/apache timeout 5
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

<code>install_type</code>	<code>install_type</code> 키워드는 모든 프로필에서 필수입니다.
<code>system_type</code>	<code>system_type</code> 키워드는 시스템이 독립형 시스템으로 설치될 것을 정의합니다.
<code>partitioning</code>	파일 시스템 슬라이스는 값 <code>default</code> 로 설치될 소프트웨어에 의해 결정됩니다. <code>swap</code> 의 크기는 512MB로 설정되고 모든 디스크에서 값 <code>any</code> 로 설치됩니다.
<code>cluster</code>	개발자 Solaris 소프트웨어 그룹 <code>SUNWCprog</code> 는 시스템에 설치됩니다.
<code>package</code>	타사 패키지는 HTTP 서버에 있는 시스템에 설치됩니다.

예 6-4 파일 시스템을 설치할 위치 지정

```
# profile keywords      profile values
# -----
install_type           initial_install
system_type            standalone
partitioning           explicit
fileys                 c0t0d0s0 auto /
fileys                 c0t3d0s1 auto swap
fileys                 any auto usr
cluster                SUNWCall
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

<code>partitioning</code>	파일 시스템 슬라이스는 <code>fileys</code> 키워드, 값 <code>explicit</code> 에 의해 결정됩니다. 루트(/)의 크기는 선택된 소프트웨어, 값 <code>auto</code> 에 기반하며 <code>c0t0d0s0</code> 에 설치됩니다. <code>swap</code> 의 크기는 필요한 크기로 설정되고 <code>c0t3d0s1</code> 에 설치됩니다. <code>usr</code> 은 선택된 소프트웨어에 기반하고 설치 프로그램이 값 <code>any</code> 를 기준으로 <code>usr</code> 이 설치되는 위치를 결정합니다.
<code>cluster</code>	전체 Solaris 소프트웨어 그룹 <code>SUNWCall</code> 이 시스템에 설치됩니다.

예 6-5 패치 업그레이드 및 설치

```
# profile keywords      profile values
# -----
install_type           upgrade
root_device            c0t3d0s2
backup_media           remote_filesystem timber:/export/scratch
package                SUNWbcp delete
```

예 6-5 패치 업그레이드 및 설치 (계속)

```

package          SUNWxwman add
cluster          SUNWCacc add
patch            patch_list nfs://patch_master/Solaris_10_606/patches \
                retry 5
locale           de
    
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

- install_type** 프로필이 디스크 공간을 재할당하여 시스템을 업그레이드합니다. 이 예에서 디스크 공간은 시스템의 일부 파일 시스템이 업그레이드에 충분한 공간을 갖고 있지 않았기 때문에 재할당되어야 합니다.
- root_device** c0t3d0s2의 루트 파일 시스템이 업그레이드됩니다.
- backup_media** 이름이 timber인 원격 시스템이 디스크 공간 재할당 동안 데이터를 백업하는 데 사용됩니다. 백업 매체 키워드 값은 156 페이지 “[backup_media](#) 프로필 키워드”를 참조하십시오.
- package** 바이너리 호환성 패키지 SUNWbcp는 업그레이드 후 시스템에 설치되지 않습니다.
- package** 코드는 X 윈도우 시스템 매뉴얼 페이지 및 시스템 계산 유틸리티가 시스템에 이미 설치되지 않은 경우 설치되게 합니다. 시스템에 이미 있는 모든 패키지가 자동으로 업그레이드됩니다.
- patch** 업그레이드로 설치되는 패치 목록입니다. 패치 목록은 Solaris_10_606/patches 디렉토리 아래 이름이 patch_master인 NFS 서버에 있습니다. 마운트가 실패하면 NFS 마운트는 다섯 번 시도됩니다.
- locale** 한국어 현지화 패키지가 시스템에 설치됩니다.

예 6-6 업그레이드를 위한 디스크 공간 재할당

```

# profile keywords      profile values
# -----
install_type           upgrade
root_device            c0t3d0s2
backup_media           remote_filesystem timber:/export/scratch
layout_constraint      c0t3d0s2 changeable 100
layout_constraint      c0t3d0s4 changeable
layout_constraint      c0t3d0s5 movable
package                SUNWbcp delete
package                SUNWxwman add
cluster                SUNWCacc add
locale                 ko
    
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

예 6-6 업그레이드를 위한 디스크 공간 재할당 (계속)

install_type	프로필이 디스크 공간을 재할당하여 시스템을 업그레이드합니다. 이 예에서 디스크 공간은 시스템의 일부 파일 시스템이 업그레이드에 충분한 공간을 갖고 있지 않았기 때문에 재할당되어야 합니다.
root_device	c0t3d0s2의 루트 파일 시스템이 업그레이드됩니다.
backup_media	이름이 timber인 원격 시스템이 디스크 공간 재할당 동안 데이터를 백업하는 데 사용됩니다. 백업 매체 키워드 값은 156 페이지 “backup_media 프로필 키워드”를 참조하십시오.
layout_constraint	layout_constraint 키워드는 자동 레이아웃이 업그레이드를 위해 디스크 공간을 재할당하려 할 때 다음을 수행할 수 있음을 지정합니다. <ul style="list-style-type: none"> ■ 슬라이스 2 및 4를 변경합니다. 슬라이스를 다른 위치로 이동할 수 있고 크기를 변경할 수 있습니다. ■ 슬라이스 5를 이동합니다. 슬라이스는 크기를 변경할 수 없지만 다른 위치로 이동할 수 있습니다.
package	바이너리 호환성 패키지 SUNWbcp는 업그레이드 후 시스템에 설치되지 않습니다.
package	코드는 X 윈도우 시스템 매뉴얼 페이지 및 시스템 계산 유틸리티가 시스템에 이미 설치되지 않은 경우 설치되게 합니다. 시스템에 이미 있는 모든 패키지가 자동으로 업그레이드됩니다.
locale	한국어 현지화 패키지가 시스템에 설치됩니다.

예 6-7 HTTP 서버에서 Solaris Flash 아카이브 검색

다음 예에서 프로필은 사용자 정의 JumpStart 프로그램이 HTTP 서버에서 Solaris Flash 아카이브를 검색함을 나타냅니다.

```
# profile keywords      profile values
# -----
install_type           flash_install
archive_location       http://192.168.255.255/flasharchive/solarisarchive
partitioning           explicit
filesystems            c0t1d0s0 4000 /
filesystems            c0t1d0s1 512 swap
filesystems            c0t1d0s7 free /export/home
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

install_type	프로필이 복제 시스템에서 Solaris Flash 아카이브를 설치합니다. 초기 설치와 마찬가지로 모든 파일을 덮어씁니다.
--------------	---

예 6-7 HTTP 서버에서 Solaris Flash 아카이브 검색 (계속)

archive_location	Solaris Flash 아카이브가 HTTP 서버에서 검색됩니다.
partitioning	파일 시스템 슬라이스는 filesys 키워드, 값 explicit에 의해 결정됩니다. 루트(/)의 크기는 Solaris Flash 아카이브의 크기를 기준으로 합니다. 루트 파일 시스템은 c0t1d0s0에 설치됩니다. swap의 크기는 필요한 크기로 설정되고 c0t1d0s1에 설치됩니다. /export/home은 남은 디스크 공간을 기준으로 합니다. /export/home은 c0t1d0s7에 설치됩니다.

예 6-8 보안 HTTP 서버에서 Solaris Flash 아카이브 검색

다음 예에서 프로필은 사용자 정의 JumpStart 프로그램이 HTTP 서버에서 Solaris Flash 아카이브를 검색함을 나타냅니다.

```
# profile keywords      profile values
# -----
install_type           flash_install
archive_location       https://192.168.255.255/solarisupdate.flar
partitioning           explicit
filesys                c0t1d0s0 4000 /
filesys                c0t1d0s1 512 swap
filesys                c0t1d0s7 free /export/home
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

install_type	프로필이 복제 시스템에서 Solaris Flash 아카이브를 설치합니다. 초기 설치와 마찬가지로 모든 파일을 덮어씁니다.
archive_location	압축된 Solaris Flash 아카이브는 보안 HTTP 서버에서 검색됩니다.
partitioning	파일 시스템 슬라이스는 filesys 키워드, 값 explicit에 의해 결정됩니다. 루트(/)의 크기는 Solaris Flash 아카이브의 크기를 기준으로 합니다. swap의 크기는 필요한 크기로 설정되고 c0t1d0s1에 설치됩니다. /export/home은 남은 디스크 공간을 기준으로 합니다. /export/home은 c0t1d0s7에 설치됩니다.

예 6-9 Solaris Flash 아카이브 검색 및 타사 패키지 설치

다음 예에서 프로필은 사용자 정의 JumpStart 프로그램이 HTTP 서버에서 Solaris Flash 아카이브를 검색함을 나타냅니다.

```
# profile keywords      profile values
# -----
install_type           flash_install
archive_location       http://192.168.255.255/flasharchive/solarisarchive
```

예 6-9 Solaris Flash 아카이브 검색 및 타사 패키지 설치 (계속)

```
partitioning      explicit
fileys            c0t1d0s0 4000 /
fileys            c0t1d0s1 512 swap
fileys            c0t1d0s7 free /export/home
package           SUNWnew http://192.168.254.255/Solaris_10_606 timeout 5
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

install_type	프로필이 복제 시스템에서 Solaris Flash 아카이브를 설치합니다. 초기 설치와 마찬가지로 모든 파일을 덮어씁니다.
archive_location	Solaris Flash 아카이브가 HTTP 서버에서 검색됩니다.
partitioning	파일 시스템 슬라이스는 fileys 키워드, 값 explicit에 의해 결정됩니다. 루트(/)의 크기는 Solaris Flash 아카이브의 크기를 기준으로 합니다. 루트 파일 시스템은 c0t1d0s0에 설치됩니다. swap의 크기는 필요한 크기로 설정되고 c0t1d0s1에 설치됩니다. /export/home은 남은 디스크 공간을 기준으로 합니다. /export/home은 c0t1d0s7에 설치됩니다.
package	SUNWnew 패키지는 HTTP 서버 192.168.254.255의 Solaris_10_606 디렉토리에서 추가됩니다.

예 6-10 NTF 서버에서 Solaris Flash 차등 아카이브 검색

다음 예에서 프로필은 사용자 정의 JumpStart 프로그램이 NFS 서버에서 Solaris Flash 아카이브를 검색함을 나타냅니다. flash_update 키워드는 이것이 차등 아카이브임을 나타냅니다. 차등 아카이브는 두 시스템 이미지 간 차이만을 설치합니다.

```
# profile keywords      profile values
# -----
install_type            flash_update
archive_location        nfs installserver:/export/solaris/flasharchive \
                        /solarisdiffarchive
no_master_check
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

install_type	프로필이 복제 시스템에서 Solaris Flash 차등 아카이브를 설치합니다. 아카이브에 의해 지정된 파일만이 설치됩니다.
archive_location	NFS 서버에서 Solaris Flash 아카이브가 검색됩니다.
no_master_check	복제 시스템이 유효한 시스템 이미지를 갖는지 확인되지 않습니다. 유효한 시스템 이미지는 원본 마스터 시스템에서 작성되었을 것입니다.

예 6-11 빈 부트 환경 만들기

다음 예에서 프로파일은 사용자 정의 JumpStart 프로그램이 빈 부트 환경을 만드는 것을 나타냅니다. 빈 부트 환경에는 파일 시스템이 포함되지 않으며 현재 부트 환경으로부터 복사가 이루어지지 않습니다. 부트 환경을 나중에 Solaris Flash 아카이브의 내용으로 채운 다음 활성화할 수 있습니다.

```
# profile keywords      profile values
# -----
install_type           initial_install
system_type            standalone
partitioning           explicit
fileys                 c0t0d0s0 auto /
fileys                 c0t3d0s1 auto swap
fileys                 any auto usr
cluster                SUNWCall
bootenv createbe      bename second_BE \
filesystem /:/dev/dsk/c0t1d0s0:ufs \
filesystem -:/dev/dsk/c0t1d0s0:swap \
filesystem /export:shared:ufs
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

partitioning	파일 시스템 슬라이스는 <code>fileys</code> 키워드, 값 <code>explicit</code> 에 의해 결정됩니다. 루트(/)의 크기는 선택된 소프트웨어, 값 <code>auto</code> 에 기반하며 <code>c0t0d0s0</code> 에 설치됩니다. <code>swap</code> 의 크기는 필요한 크기로 설정되고 <code>c0t3d0s1</code> 에 설치됩니다. <code>usr</code> 은 선택된 소프트웨어에 기반하고 설치 프로그램이 값 <code>any</code> 를 기준으로 <code>usr</code> 이 설치되는 위치를 결정합니다.
cluster	전체 Solaris 소프트웨어 그룹 <code>SUNWCall</code> 이 시스템에 설치됩니다.
bootenv createbe	빈 비활성 부트 환경이 디스크 <code>c0t1d0</code> 에 설정됩니다. 루트(/), 스왑 및 <code>/export</code> 에 대한 파일 시스템이 생성되지만 비어 있습니다. 나중에 Solaris Flash 아카이브를 사용하여 두 번째 부트 환경을 만들 수 있습니다. 그러면 새로운 부트 환경이 현재 부트 환경이 되도록 활성화할 수 있습니다.

이 키워드 사용에 대한 키워드 값 및 배경에 대해서는 다음을 참조하십시오.

- 키워드 값의 설명은 149 페이지 “프로필 키워드 및 값”을 참조하십시오.
- 비활성 부트 환경을 작성, 업그레이드 및 활성화하는 Solaris Live Upgrade의 사용에 대한 배경 지식은 **Solaris 10 6/06** 설치 설명서: **Solaris Live Upgrade** 및 업그레이드 계획의 6 장, “Solaris Live Upgrade(개요)”를 참조하십시오.

예 6-11 빈 부트 환경 만들기 (계속)

- Solaris Flash 아카이브 사용에 대한 배경 지식은 **Solaris 10 6/06** 설치 설명서: **Solaris Flash** 아카이브(작성 및 설치)의 1 장, “Solaris Flash(개요)”를 참조하십시오.

예 6-12 Solaris Flash 아카이브를 설치할 때 RAID-1 볼륨 만들기

다음 예에서 프로필은 사용자 정의 JumpStart 프로그램이 Solaris 볼륨 관리자 기술을 사용하여 루트(/), swap, /usr 및 /export/home 파일 시스템에 대해 RAID-1 볼륨(미러)을 만드는 것을 나타냅니다. Solaris Flash 아카이브는 부트 환경에서 설치됩니다.

```
# profile keywords      profile values
# -----
install_type           flash_install
archive_location       nfs server:/export/home/export/flash.s10.SUNWCall
partitioning           explicit
filesys                mirror:d10 c0t0d0s0 c0t1d0s0 4096 /
filesys                mirror c0t0d0s1 2048 swap
filesys                mirror:d30 c0t0d0s3 c0t1d0s3 4096 /usr
filesys                mirror:d40 c0t0d0s4 c0t1d0s4 4096 /usr
filesys                mirror:d50 c0t0d0s5 c0t1d0s5 free /export/home
metadb                 c0t1d0s7 size 8192 count 3
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

install_type	프로필이 복제 시스템에서 Solaris Flash 아카이브를 설치합니다. 초기 설치와 마찬가지로 모든 파일을 덮어씁니다.
archive_location	NFS 서버에서 Solaris Flash 아카이브가 검색됩니다.
partitioning	파일 시스템 슬라이스는 filesys 키워드, 값 explicit에 의해 결정됩니다.
filesys	루트(/) 파일 시스템이 슬라이스 c0t0d0s0과 c0t1d0s0에 만들어지고 미러됩니다. 루트(/) 파일 시스템의 크기는 4096MB로 설정됩니다. c0t0d0s0과 c0t1d0s0을 미러하는 RAID-1 볼륨의 이름은 d10으로 지정됩니다.
filesys	swap 파일 시스템이 슬라이스 c0t0d0s1에 만들어지고 미러되며 크기는 2048MB가 됩니다. 사용자 정의 JumpStart 프로그램은 미러에 이름을 할당합니다.
filesys	/usr 파일 시스템이 슬라이스 c0t1d0s3과 c0t0d0s3에 만들어지고 미러됩니다. /usr 파일 시스템의 크기는 4096MB로 설정됩니다. RAID-1 볼륨의 이름은 d30으로 지정됩니다.

예 6-12 Solaris Flash 아카이브를 설치할 때 RAID-1 볼륨 만들기 (계속)

filesystem /usr 파일 시스템이 슬라이스 `c0t1d0s4`와 `c0t0d0s4`에 만들어지고 미러됩니다. /usr 파일 시스템의 크기는 4096MB로 설정됩니다. RAID-1 볼륨의 이름은 `d40`으로 지정됩니다.

metadb 세 개의 상태 데이터베이스 복제본(`metadb`)이 슬라이스 `c0t1d0s7`에 설치되며 크기는 8192 블록(4MB)이 됩니다.

- 설치 중에 미러된 파일 시스템을 만드는 방법에 대한 개요 정보는 12 장을 참조하십시오.
- 미러된 파일 시스템 만들기에 대한 지침과 요구 사항은 13 장을 참조하십시오.
- 키워드 값의 설명은 168 페이지 “`filesystem` 프로필 키워드(RAID-1 볼륨 작성)” 및 173 페이지 “`metadb` 프로필 키워드(상태 데이터베이스 복제본 만들기)”를 참조하십시오.

예 6-13 루트 파일 시스템을 미러할 RAID-1 볼륨 만들기

다음 예에서 프로필은 사용자 정의 JumpStart 프로그램이 Solaris 볼륨 관리자 기술을 사용하여 루트(/) 파일 시스템에 대해 RAID-1 볼륨(미러)을 만드는 것을 나타냅니다.

```
# profile keywords      profile values
# -----
install_type           initial_install
cluster                SUNWCXall
filesystem             mirror:d30 c0t1d0s0 c0t0d0s0 /
                       c0t0d0s3 512 swap
filesystem             c0t0d0s4 size 8192 count 4
metadb                 c0t1d0s4 size 8192 count 4
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

cluster 시스템에 전체 Solaris 소프트웨어와 OEM 지원 소프트웨어 그룹, SUNWCXall이 설치됩니다.

filesystem 루트(/) 파일 시스템이 슬라이스 `c0t1d0s0`과 `c0t0d0s0`에 만들어지고 미러됩니다. `c0t1d0s0`과 `c0t0d0s0`을 미러하는 RAID-1 볼륨의 이름은 `d30`으로 지정됩니다. 사용자 정의 JumpStart 프로그램은 두 개의 하위 미러에 이름을 할당합니다.

filesystem swap 파일 시스템이 슬라이스 `c0t0d0s3`에 만들어지고 미러되며 크기는 512MB가 됩니다.

metadb 네 개의 상태 데이터베이스 복제본(`metadb`)이 슬라이스 `c0t0d0s4`에 설치되며 크기는 8192 블록(4MB)이 됩니다.

metadb 네 개의 상태 데이터베이스 복제본(`metadb`)이 슬라이스 `c0t1d0s4`에 설치되며 크기는 8192 블록(4MB)이 됩니다.

- 설치 중에 RAID-1 볼륨을 만드는 방법에 대한 개요 정보는 12 장을 참조하십시오.
- RAID-1 볼륨 만들기에 대한 지침 및 요구 사항은 13 장을 참조하십시오.

예 6-13 루트 파일 시스템을 미리할 RAID-1 볼륨 만들기 (계속)

- 키워드 값의 설명은 168 페이지 “[fileysys](#) 프로필 키워드(RAID-1 볼륨 작성)” 및 173 페이지 “[metadb](#) 프로필 키워드(상태 데이터베이스 복제본 만들기)”를 참조하십시오.

예 6-14 여러 파일 시스템을 미리할 RAID-1 볼륨 만들기

다음 예에서 프로필은 사용자 정의 JumpStart 프로그램이 Solaris 볼륨 관리자 기술을 사용하여 루트(/), swap 및 /usr 파일 시스템에 대해 RAID-1 볼륨(미러)을 만드는 것을 나타냅니다.

```
# profile keywords      profile values
# -----
install_type           initial_install
cluster                SUNWCXall
fileysys               mirror:d100 c0t1d0s0 c0t0d0s0 200 /
fileysys               c0t1d0s5 500 /var
fileysys               c0t0d0s5 500
fileysys               mirror c0t0d0s1 512 swap
metadb                 c0t0d0s3 size 8192 count 5
fileysys               mirror c0t1d0s4 c0t0d0s4 2000 /usr
fileysys               c0t1d0s7 free /export/home
fileysys               c0t0d0s7 free
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

- | | |
|----------|--|
| cluster | 시스템에 전체 Solaris 소프트웨어와 OEM 지원 소프트웨어 그룹, SUNWCXall이 설치됩니다. |
| fileysys | 루트(/) 파일 시스템이 슬라이스 c0t1d0s0과 c0t0d0s0에 만들어지고 미리됩니다. 루트(/) 파일 시스템의 크기는 200MB로 설정됩니다. c0t1d0s0과 c0t0d0s0을 미리하는 RAID-1 볼륨의 이름은 d100으로 지정됩니다. |
| fileysys | /var 파일 시스템은 슬라이스 c0t1d0s5에 설치되고 크기는 500MB입니다. 루트(/) 파일 시스템이 슬라이스 c0t1d0s0과 c0t0d0s0에 만들어지고 미리됩니다. 루트(/) 파일 시스템의 크기는 200MB로 설정됩니다. c0t1d0s0과 c0t0d0s0을 미리하는 RAID-1 볼륨의 이름은 d100으로 지정됩니다. |
| fileysys | swap 파일 시스템이 슬라이스 c0t0d0s1에 만들어지고 미리되며 크기는 512MB가 됩니다. 사용자 정의 JumpStart 프로그램은 미러에 이름을 할당합니다. |
| metadb | 다섯 개의 상태 데이터베이스 복제본(metadb)이 슬라이스 c0t0d0s3에 설치되며 크기는 8192블록(4MB)이 됩니다. |
| fileysys | /usr 파일 시스템이 슬라이스 c0t1d0s4와 c0t0d0s4에 만들어지고 미리됩니다. /usr 파일 시스템의 크기는 2000MB로 설정됩니다. 사용자 정의 JumpStart 프로그램은 미러에 이름을 할당합니다. |
- 설치 중에 미리된 파일 시스템을 만드는 방법에 대한 개요 정보는 [12 장](#)을 참조하십시오.
 - 미리된 파일 시스템 만들기에 대한 지침과 요구 사항은 [13 장](#)을 참조하십시오.

예 6-14 여러 파일 시스템을 미리할 RAID-1 볼륨 만들기 (계속)

- 키워드 값의 설명은 168 페이지 “filesys 프로필 키워드(RAID-1 볼륨 작성)” 및 173 페이지 “metadb 프로필 키워드(상태 데이터베이스 복제본 만들기)”를 참조하십시오.

예 6-15 x86: fdisk 키워드 사용

```
# profile keywords      profile values
# -----
install_type           initial_install
system_type            standalone

fdisk                  c0t0d0 0x04 delete
fdisk                  c0t0d0 solaris maxfree
cluster                SUNWCall
cluster                SUNWCacc delete
```

다음 목록에서는 이 예의 일부 키워드와 값에 대해 설명합니다.

fdisk 유형 DOSOS16(04 16진수)의 모든 fdisk 분할 영역이 c0t0d0 디스크에서 삭제됩니다.

fdisk Solaris fdisk 분할 영역은 c0t0d0 디스크의 최대 인접 사용 가능 공간에 생성됩니다.

cluster 전체 배포 Solaris 소프트웨어 그룹 SUNWCall이 시스템에 설치됩니다.

cluster 시스템 계정 유틸리티 SUNWCacc는 시스템에 설치되지 않습니다.

프로필 테스트

프로필을 만든 후 pfinstall(1M) 명령을 사용하여 프로필을 테스트합니다. 프로필을 사용하여 시스템을 설치 또는 업그레이드하기 전에 프로필을 테스트합니다. 프로필 테스트는 디스크 공간을 재할당하는 업그레이드 프로필을 만드는 중일 때 특히 유용합니다.

pfinstall에 의해 생성된 설치 출력을 조사하여 프로필이 의도한대로 작동하는지 신속하게 확인할 수 있습니다. 예를 들어, 프로필을 사용하여 해당 시스템에서 업그레이드를 수행하기 전에 시스템에 Solaris 소프트웨어의 새 릴리스를 업그레이드할 충분한 디스크 공간이 있는지 확인합니다.

pfinstall을 사용하여 다음에 대한 프로필을 테스트할 수 있습니다.

- pfinstall이 실행을 시작할 시스템의 디스크 구성
- 기타 디스크 구성. 디스크의 구조, 예를 들어, 디스크의 바이트/섹터, 플래그 및 슬라이스를 표현하는 디스크 구성 파일을 사용합니다. 디스크 구성 파일 만들기는 108 페이지 “디스크 구성 파일 만들기” 및 110 페이지 “x86: 디스크 구성 파일을 만들기”에서 설명합니다.

주 - 디스크 구성 파일을 사용하여 시스템을 업그레이드하는 데 사용할 프로필을 테스트할 수 없습니다. 대신에 시스템의 실제 디스크 구성과 해당 시스템에 현재 설치된 소프트웨어에 대해 프로필을 테스트해야 합니다.

▼ 프로필을 테스트하는 임시 Solaris 환경 만들기

특정 Solaris 릴리스에 대한 프로필을 성공적이고 정확하게 테스트하려면 동일한 릴리스의 Solaris 환경 내에서 프로필을 테스트해야 합니다. 예를 들어, Solaris 초기 설치 프로필을 테스트하려면 Solaris OS를 실행 중인 시스템에서 `pfinstall` 명령을 실행합니다.

다음 조건 중 하나에 따라 프로필을 테스트하려면 임시 설치 환경을 만들어야 합니다.

- Solaris 소프트웨어의 이전 버전을 실행 중인 시스템에서 Solaris 10 6/06 업그레이드 프로필을 테스트할 수 있습니다.
- Solaris 10 6/06 초기 설치 프로필을 테스트할 Solaris 10 6/06 시스템이 아직 설치되어 있지 않습니다.

1 다음 중 하나의 이미지에서 시스템을 부트합니다.

SPARC 기반 시스템의 경우:

- Solaris Operating System for SPARC Platforms DVD
- Solaris Software for SPARC Platforms - 1 CD

x86 기반 시스템의 경우:

- Solaris Operating System for x86 Platforms DVD
- Solaris Software for x86 Platforms - 1 CD

주 - 업그레이드 프로필을 테스트하려면 업그레이드 중인 시스템을 부트합니다.

2 시스템 식별 질문에 응답합니다.

3 설치 프로그램에서 종료하려면 다음 프롬프트에서 !를 입력합니다.

The Solaris installation program will assist you in installing software for Solaris.
<Press ENTER to continue> {"!" exits}

4 셸에서 `pfinstall` 명령을 실행합니다. `pfinstall` 명령 사용에 대한 자세한 내용은 95 페이지 “프로필 테스트”의 단계 7를 참조하십시오.

▼ 프로필 테스트

x86 전용 -locale 키워드를 사용하는 경우 `pinstall -D` 명령이 프로필 테스트에 실패합니다. 이에 대한 해결책은 219 페이지 “Solaris OS 업그레이드” 섹션의 오류 메시지 “could not select locale” 을 참조하십시오.

- 1 프로필이 생성된 동일한 플랫폼, SPARC 또는 x86인 프로필을 테스트할 시스템을 찾습니다. 업그레이드 프로필을 테스트 중인 경우 업그레이드하려는 실제 시스템에서 프로필을 테스트해야 합니다.
- 2 다음 결정 테이블을 사용하여 다음 해야 할 작업을 결정합니다.

테스트 시나리오	지침
초기 설치 프로필을 테스트하고 Solaris 10 6/06 소프트웨어를 실행합니다.	시스템에서 슈퍼유저가 되고 단계 5로 이동합니다.
업그레이드 프로필을 테스트합니다. 그렇지 않으면 초기 설치 프로필을 테스트할 Solaris 10 6/06이 실행되지 않습니다.	임시 Solaris 10 6/06 환경을 만들어 프로필을 테스트합니다. 자세한 내용은 94 페이지 “프로필을 테스트하는 임시 Solaris 환경 만들기”를 참조하십시오. 그런 다음 단계 3로 이동합니다.

- 3 임시 마운트 지점을 만듭니다.
`mkdir /tmp/mnt`
- 4 테스트하려는 프로필을 포함하는 디렉토리를 마운트합니다.

마운트 시나리오	입력 지침
네트워크상의 시스템에 대하여 원격 NFS 파일 시스템을 마운트합니다.	<code>mount -F nfs server_name:path /tmp/mnt</code>
SPARC: UFS 포맷된 디스켓을 마운트합니다.	<code>mount -F ufs /dev/diskette /tmp/mnt</code>
PCFS 포맷된 디스켓을 마운트합니다.	<code>mount -F pcfs /dev/diskette /tmp/mnt</code>

- 5 특정한 시스템 메모리 크기로 프로필을 테스트하려면 `SYS_MEMSIZE`를 MB 단위의 특정한 메모리 크기로 설정합니다.
`SYS_MEMSIZE=memory_size`
`export SYS_MEMSIZE`
- 6 단계 4에서 디렉토리를 마운트했습니까?
 - 예, 디렉토리를 /tmp/mnt로 변경합니다.

```
# cd /tmp/mnt
```

- 아니요, 일반적으로 JumpStart 디렉토리인, 프로필이 위치한 디렉토리로 변경합니다.

```
# cd jumpstart_dir_path
```

7 pfinstall(1M) 명령으로 프로필을 테스트합니다.

```
# /usr/sbin/install.d/pfinstall -D:-d disk_config_file -c path profile
```



주의 - -d 또는 -D 옵션을 반드시 포함해야 합니다. 이러한 옵션 중 하나를 포함시키지 않은 경우, pfinstall은 지정한 프로필을 사용하여 Solaris 소프트웨어를 설치합니다. 시스템의 모든 데이터를 덮어씁니다.

-D pfinstall은 현재 시스템의 디스크 구성을 사용하여 프로필을 테스트합니다. -D 옵션을 사용하여 업그레이드 프로필을 테스트해야 합니다.

-d disk_config_file pfinstall은 디스크 구성 파일, *disk_config_file*을 사용하여 프로필을 테스트합니다. *disk_config_file*이 pfinstall이 실행되는 디렉토리에 위치하지 않을 경우 해당 경로를 지정해야 합니다.

디스크 구성 파일을 만드는 방법에 대한 지침은 108 페이지 “디스크 구성 파일 만들기”를 참조하십시오.

주 - -d *disk_config_file* 옵션을 업그레이드 프로필 *install_type upgrade*와 함께 사용할 수 없습니다. 항상 시스템의 디스크 구성에 대해 업그레이드 프로필을 테스트해야 합니다. 즉, -D 옵션을 사용해야 합니다.

-c path Solaris 소프트웨어 이미지의 경로. 예를 들어, 시스템이 볼륨 관리자를 사용하여 플랫폼에 대해 Solaris Software - 1 CD를 마운트하고 있는 경우 이 옵션을 사용합니다.

주 - -c 옵션은 플랫폼에 대해 Solaris Operating System DVD 또는 Solaris Software - 1 CD 이미지에서 부트한 경우 필요하지 않습니다. DVD 또는 CD 이미지는 부트 프로세스의 일부로 /cdrom에서 마운트됩니다.

profile 테스트할 프로필의 이름. *profile*이 pfinstall이 실행 중인 디렉토리가 아닌 경우 경로를 지정해야 합니다.

프로필 테스트의 예

다음은 pfinstall을 사용하여 이름이 *basic_prof*인 프로필을 테스트하는 방법을 보여주는 예입니다. Solaris 10 6/06 소프트웨어가 설치되는 시스템의 디스크 구성에 대해 프로필을

테스트합니다. `basic_prof` 프로파일은 `/jumpstart` 디렉토리에 있으며 볼륨 관리자가 사용되고 있으므로 Solaris Operating System DVD 이미지에 대한 경로가 지정됩니다.

예 6-16 Solaris 10 6/06 시스템을 사용하여 프로파일 테스트

```
# cd /jumpstart
# /usr/sbin/install.d/pfinstall -D -c /cdrom/pathname basic_prof
```

다음 예는 `pfinstall`을 사용하여 Solaris 10 6/06 시스템에서 이름이 `basic_prof`인 프로파일을 테스트하는 방법을 보여 줍니다. 테스트는 `535_test` 디스크 구성 파일에 대하여 수행됩니다. 테스트를 통해 시스템에 64MB의 메모리가 있는지 조사합니다. 이 예는 `/export/install` 디렉토리에 있는 Solaris Software for SPARC Platforms - 1 CD 또는 Solaris Software for x86 Platforms - 1 CD 이미지를 사용합니다.

예 6-17 디스크 구성 파일을 사용하는 프로파일 테스트

```
# SYS_MEMSIZE=64
# export SYS_MEMSIZE
# /usr/sbin/install.d/pfinstall -d 535_test -c /export/install basic_prof
```

rules 파일 검증

프로파일, rules 파일을 사용하기 전에 check 스크립트를 실행하여 파일이 제대로 설정되었는지 검증해야 합니다. 모든 규칙 및 프로파일 이름이 제대로 설정되면 시스템을 프로파일에 일치시키는 사용자 정의 JumpStart 설치 소프트웨어가 필요로 하는 `rules.ok` 파일이 생성됩니다.

표 6-2는 check 스크립트가 수행하는 작업을 설명합니다.

표 6-2 check 스크립트를 사용하는 경우 수행되는 작업

단계	설명
1	rules 파일의 구문이 검사됩니다. check 규칙 키워드가 적합하고 <i>begin</i> , <i>class</i> 및 <i>finish</i> 필드가 각 규칙에 대해 지정되었음을 검증합니다. <i>begin</i> 및 <i>finish</i> 필드는 파일 이름 대신 마이너스 기호(-)로 구성될 수 있습니다.
2	rules 파일에 오류가 없으면 규칙에서 지정된 각 프로파일의 구문이 검사됩니다.
3	오류가 없으면 check가 rules 파일에서 <code>rules.ok</code> 파일을 만들고 모든 주석 및 빈 행을 제거하고 모든 규칙을 보관하고 다음 주석 행을 끝에 추가합니다. # version=2 checksum=num

▼ rules 파일 검증

- 1 check 스크립트가 JumpStart 디렉토리에 있는지 확인합니다.

주 - check 스크립트는 Solaris Operating System DVD 또는 Solaris Software - 1 CD의 Solaris_10_606/Misc/jumpstart_sample 디렉토리에 있습니다.

- 2 디렉토리를 JumpStart 디렉토리로 변경합니다.

- 3 check 스크립트를 실행하여 rules 파일을 검증합니다.

```
$ ./check -p path -r file_name
```

-p path 사용 중인 시스템의 check 스크립트 대신 Solaris 소프트웨어 이미지의 check 스크립트를 사용하여 rules를 검증합니다. *path*는 로컬 디스크 또는 마운트된 Solaris Operating System DVD 또는 Solaris Software - 1 CD의 이미지입니다.

시스템에 이전 버전의 Solaris를 실행하는 경우 이 옵션을 사용하여 최신 버전의 check를 실행합니다.

-r file_name rules가 아닌 파일 이름을 지정합니다. 이 옵션을 사용하여 규칙을 rules 파일로 통합하기 전에 규칙의 유효성을 테스트할 수 있습니다.

check 스크립트가 실행될 때 스크립트는 rules 파일과 각 프로파일의 유효성 확인을 보고합니다. 오류가 없으면 스크립트는 다음 정보를 보고합니다.

```
The custom JumpStart configuration is ok
```

- 4 root가 rules.ok 파일을 소유하고 사용 권한이 644로 설정되었는지 확인합니다.

참조 rules 파일을 검증한 후 7장에서 선택적 사용자 정의 JumpStart 기능에 대하여 더 자세히 익힐 수 있습니다. 9장에서 사용자 정의 JumpStart 설치 수행에 대하여 더 자세히 익힐 수 있습니다.

사용자 정의 JumpStart 선택적 기능 사용(작업)

이 장에서는 별도의 사용자 정의 JumpStart 설치 도구를 만드는 데 사용할 수 있는 선택적 기능에 대해 설명합니다.

- 99 페이지 “시작 스크립트 만들기”
- 101 페이지 “종료 스크립트 만들기”
- 107 페이지 “압축된 구성 파일 만들기”
- 108 페이지 “디스크 구성 파일 만들기”
- 113 페이지 “사이트별 설치 프로그램 사용”

주 - 이 장의 지침은 프로필 서버라고 하는, 사용자 정의 JumpStart 파일을 제공하는 데 사용되는 SPARC 서버 또는 x86 서버에 대하여 유효합니다. 프로필 서버에서 다른 플랫폼 유형에 대하여 사용자 정의 JumpStart 파일을 제공할 수 있습니다. 예를 들어, SPARC 서버는 SPARC 기반 시스템 및 x86 기반 시스템에 사용자 정의 JumpStart 파일을 제공할 수 있습니다.

시작 스크립트 만들기

시작 스크립트는 rules 파일에서 지정하는 사용자 정의 Bourne 셸 스크립트입니다. 시작 스크립트는 Solaris 소프트웨어가 시스템에 설치되기 전에 작업을 수행합니다. 사용자 정의 JumpStart를 사용하여 Solaris 소프트웨어를 설치할 경우에만 시작 스크립트를 사용할 수 있습니다.

다음 작업 중 하나를 수행하려면 시작 스크립트를 사용합니다.

- 파생 프로필 작성
- 업그레이드 전에 백업

시작 스크립트에 관한 주요 정보

- 초기 설치 또는 업그레이드 중 /a로의 파일 시스템 마운트를 방해할 수 있는 사항을 스크립트에 지정하지 마십시오. JumpStart 프로그램이 파일 시스템을 /a에 마운트할 수 없는 경우 오류가 발생해 설치에 실패합니다.
- 설치하는 동안 시작 스크립트의 출력은 /tmp/begin.log에 배치됩니다. 설치가 완료된 후에 로그 파일이 /var/sadm/system/logs/begin.log로 리디렉션됩니다.
- root가 시작 스크립트의 소유자이고 권한이 644로 설정되어 있는지 확인하십시오.
- 사용자 정의 JumpStart 환경 변수를 시작 스크립트에서 사용할 수 있습니다. 환경 변수의 목록은 187 페이지 “사용자 정의 JumpStart 환경 변수”를 참조하십시오.
- 시작 스크립트를 JumpStart 디렉토리에 저장합니다.

시작 스크립트를 사용하여 파생된 프로필 작성

파생된 프로필은 사용자 정의 JumpStart 설치 중 시작 스크립트에서 동적으로 만든 프로필입니다. 파생된 프로필은 rules 파일이 특정 시스템을 프로필과 일치하도록 설정할 수 없을 때 필요합니다. 예를 들어, 서로 다른 프레임 버퍼를 갖고 있는 시스템들과 같이 상이한 하드웨어 구성 요소를 갖고 있는 동일한 시스템 모델을 위한 파생된 프로필을 사용할 필요가 있습니다.

파생된 프로필을 사용하기 위해 규칙을 설정하려면 다음 작업을 수행해야 합니다.

- profile 필드를 프로필 대신 등호(=)로 설정합니다.
- 시작 필드를 Solaris를 설치할 시스템에 따른 파생된 프로필을 만드는 시작 스크립트로 설정합니다.

profile 필드의 값이 등호(=)로 설정된 규칙을 만나면 시작 스크립트는 Solaris 소프트웨어를 시스템에 설치하기 위해 사용되는 파생된 프로필을 작성합니다.

다음은 매번 동일한 파생된 프로필을 만드는 시작 스크립트의 예입니다. 규칙 평가에 따라 달라지는 서로 다른 파생된 프로필을 만들기 위해 시작 스크립트를 만들 수 있습니다.

예 7-1 파생된 프로필을 만드는 시작 스크립트

```
#!/bin/sh
echo "install_type      initial_install"    > ${SI_PROFILE}
echo "system_type      standalone"      >> ${SI_PROFILE}
echo "partitioning     default"         >> ${SI_PROFILE}
echo "cluster          SUNWCprog"       >> ${SI_PROFILE}
echo "package          SUNWman         delete"    >> ${SI_PROFILE}
echo "package          SUNWolman      delete"    >> ${SI_PROFILE}
echo "package          SUNWwxman      delete"    >> ${SI_PROFILE}
```

예에서 시작 스크립트는 기본적으로 /tmp/install.input으로 설정되는 파생된 프로필의 이름에 대하여 SI_PROFILE 환경 변수를 사용해야 합니다.

주- 시작 스크립트를 사용하여 파생된 프로필을 만든 경우 해당 스크립트에 오류가 없는지 확인하십시오. 파생된 프로필은 시작 스크립트 실행 전까지는 생성되지 않기 때문에 파생된 프로필은 check 스크립트에 의해 검증되지 않습니다.

종료 스크립트 만들기

종료 스크립트는 rules 파일에 지정하는 사용자 정의 Bourne 셸 스크립트입니다. 종료 스크립트는 시스템에 Solaris 소프트웨어가 설치된 뒤 시스템을 재부트하기 전에 작업을 수행합니다. 종료 스크립트는 사용자 정의 JumpStart를 사용하여 Solaris를 설치할 때만 사용할 수 있습니다.

종료 스크립트를 사용하여 수행할 수 있는 작업은 다음과 같습니다.

- 파일 추가
- 특정 소프트웨어 그룹에서 설치된 것 이외의 개별 패키지나 패치 추가
- 루트 환경 사용자 정의
- 시스템의 루트 비밀번호 설정
- 추가 소프트웨어 설치

종료 스크립트에 관한 주요 정보

- Solaris 설치 프로그램은 /a에 시스템의 파일 시스템을 마운트합니다. 파일 시스템은 시스템을 재부트할 때까지 /a에 마운트된 채로 남게 됩니다. 새로 설치된 파일 시스템 계층에서 /a와 관련된 파일 시스템을 수정함으로써 파일을 추가, 변경 또는 제거하기 위해 종료 스크립트를 사용할 수 있습니다.
- 설치 중 종료 스크립트의 출력은 /tmp/finish.log에 배치됩니다. 설치가 완료된 후에 로그 파일이 /var/sadm/system/logs/finish.log로 리디렉션됩니다.
- root가 종료 스크립트의 소유자이고 권한이 644로 설정되어 있는지 확인하십시오.
- 사용자 정의 JumpStart 환경 변수를 종료 스크립트에서 사용할 수 있습니다. 환경 변수의 목록은 187 페이지 “사용자 정의 JumpStart 환경 변수”를 참조하십시오.
- 종료 스크립트를 JumpStart 디렉토리에 저장합니다.

▼ 종료 스크립트를 사용하여 파일 추가

종료 스크립트를 통해 JumpStart 디렉토리의 파일을 이미 설치된 시스템에 추가할 수 있습니다. JumpStart 디렉토리는 SI_CONFIG_DIR 변수에 의해 지정된 디렉토리에 마운트되므로 파일을 추가할 수 있습니다. 디렉토리는 기본적으로 /tmp/install_config로 설정됩니다.

주 - 종료 스크립트 JumpStart 디렉토리의 파일을 설치된 시스템에 이미 존재하는 파일로 복사하여 파일을 대체할 수도 있습니다.

- 1 설치된 시스템에 추가할 모든 파일을 JumpStart 디렉토리에 복사합니다.
- 2 새로 설치된 파일 시스템 계층에 복사할 각 파일을 위한 종료 스크립트에 다음 행을 삽입합니다.

```
cp ${SI_CONFIG_DIR}/file_name /a/path_name
```

예 7-2 종료 스크립트를 사용하여 파일 추가

예를 들어, 사이트의 모든 사용자를 위해 개발된 `site_prog`라는 특수 응용 프로그램이 있다고 가정합니다. `site_prog` 사본을 JumpStart 디렉토리에 배치하면 종료 스크립트의 다음 행이 `site_prog`를 JumpStart 디렉토리에서 시스템의 `/usr/bin` 디렉토리로 복사합니다.

```
cp ${SI_CONFIG_DIR}/site_prog /a/usr/bin
```

종료 스크립트를 사용하여 패키지 또는 패치 추가

Solaris 소프트웨어가 시스템에 설치된 뒤 자동으로 패키지나 패치를 추가하는 종료 스크립트를 만들 수 있습니다. 종료 스크립트를 사용하여 패키지를 추가함으로써 시간을 절약하고 사이트의 서로 다른 시스템들에 설치되는 패키지과 패치의 일관성을 보장할 수 있습니다.

종료 스크립트에서 `pkgadd(1M)` 또는 `patchadd(1M)` 명령어를 사용하는 경우 `-R` 옵션을 사용하여 `/a`를 루트 경로로 지정합니다.

- 예 7-3은 패키지를 추가하는 종료 스크립트의 예를 보여 줍니다.
- 예 7-4는 패치를 추가하는 종료 스크립트의 예를 보여 줍니다.

예 7-3 종료 스크립트를 사용하여 패키지 추가

```
#!/bin/sh

BASE=/a
MNT=/a/mnt
ADMIN_FILE=/a/tmp/admin

mkdir ${MNT}
mount -f nfs sherlock:/export/package ${MNT}
cat >${ADMIN_FILE} <<DONT_ASK
mail=root
instance=overwrite
partial=nocheck
```

예 7-3 종료 스크립트를 사용하여 패키지 추가 (계속)

```
runlevel=nocheck
idepend=nocheck
rdepend=nocheck
space=ask
setuid=nocheck
conflict=nocheck
action=nocheck
basedir=default
DONT_ASK

/usr/sbin/pkgadd -a ${ADMIN_FILE} -d ${MNT} -R ${BASE} SUNwxyz
umount ${MNT}
rmdir ${MNT}
```

다음에서는 이 예의 일부 명령어에 대해 설명합니다.

- 다음 명령어는 설치할 패키지가 포함되어 있는 서버에 디렉토리를 마운트합니다.

```
mount -f nfs sherlock:/export/package ${MNT}
```

- 다음 명령어는 임시 패키지 관리 파일인 `admin`을 만들어 `pkgadd(1M)` 명령어가 패키지를 설치할 때 확인을 수행하거나 질문을 하지 않도록 합니다. 패키지를 추가할 때 임시 패키지 관리 파일을 사용하여 자동 설치를 관리합니다.

```
cat >${ADMIN_FILE} <<DONT_ASK
```

- 다음의 `pkgadd` 명령어는 `-a` 옵션(패키지 관리 파일 지정)과 `-R` 옵션(루트 경로 지정)을 사용하여 패키지를 추가합니다.

```
/usr/sbin/pkgadd -a ${ADMIN_FILE} -d ${MNT} -R ${BASE} SUNwxyz
```

예 7-4 종료 스크립트를 사용하여 패치 추가

```
#!/bin/sh

#####
#
# USER-CONFIGURABLE OPTIONS
#
#####

# The location of the patches to add to the system after it's installed.
# The OS rev (5.x) and the architecture ('mach') will be added to the
# root. For example, /foo on a 8 SPARC would turn into /foo/5.8/sparc
LUPATCHHOST=ins3525-svr
LUPATCHPATHROOT=/export/solaris/patchdb
#####
```

예 7-4 종료 스크립트를 사용하여 패치 추가 (계속)

```
#
# NO USER-SERVICEABLE PARTS PAST THIS POINT
#
#####

BASEDIR=/a

# Figure out the source and target OS versions
echo Determining OS revisions...
SRCREV='uname -r'
echo Source $SRCREV

LUPATCHPATH=$LUPATCHPATHROOT/$SRCREV/'mach'

#
# Add the patches needed
#
echo Adding OS patches
mount $LUPATCHHOST:$LUPATCHPATH /mnt >/dev/null 2>&1
if [ $? = 0 ] ; then
    for patch in `cat /mnt/*Recommended/patch_order` ; do
        (cd /mnt/*Recommended/$patch ; echo yes | patchadd -u -d -R $BASEDIR .)
    done
    cd /tmp
    umount /mnt
else
    echo "No patches found"
if
```

주 - 과거에는 chroot(1M) 종료 스크립트 환경에서 pkgadd 및 patchadd 명령어와 함께 사용되었습니다. 드물지만 일부 패키지나 패치는 -R 옵션과 함께 작동하지 않습니다. chroot 명령을 내리기 전에 /a 루트 경로에서 임시 /etc/mnttab 파일을 작성해야 합니다.

가상 /etc/mnttab 파일을 만들려면 종료 스크립트에 다음 행을 추가합니다.

```
cp /etc/mnttab /a/etc/mnttab
```

종료 스크립트를 사용하여 루트 환경 사용자 정의

종료 스크립트를 사용하여 이미 시스템에 설치된 파일을 사용자 정의할 수도 있습니다. 예를 들어 예 7-5의 종료 스크립트는 루트(/) 디렉토리의 .cshrc 파일에 정보를 추가하여 루트 환경을 사용자 정의합니다.

예 7-5 종료 스크립트를 사용하여 루트 환경 사용자 정의

```
#!/bin/sh
#
# Customize root's environment
#
echo "****adding customizations in /.cshrc"
test -f a/.cshrc || {
cat >> a/.cshrc <<EOF
set history=100 savehist=200 filec ignoreeof prompt="\$user@'uname -n'> "
alias cp cp -i
alias mv mv -i
alias rm rm -i
alias ls ls -FC
alias h history
alias c clear
unset autologout
EOF
}
```

종료 스크립트를 사용하여 시스템의 루트 비밀번호 설정

Solaris 소프트웨어가 시스템에 설치된 후 시스템이 재부트됩니다. 부트 프로세스가 끝나기 전에 시스템은 루트 비밀번호 입력을 요청합니다. 비밀번호를 입력하기 전에는 시스템은 부트를 마칠 수 없습니다.

set_root_pw라는 이름의 종료 스크립트는 auto_install_sample 디렉토리에 저장됩니다. 종료 스크립트는 루트 비밀번호 입력 요청을 받지 않고 자동으로 설정하는 방법을 보여 줍니다. set_root_pw는 예 7-6에 나와 있습니다.

주 - 종료 스크립트를 사용하여 시스템의 루트 비밀번호를 설정한 경우 사용자가 종료 스크립트에 있는 암호화된 비밀번호에서 루트 비밀번호를 알아내려 할 수 있습니다. 루트 비밀번호를 파악하려고 시도하는 사용자에 대비한 보호 조치를 취해야 합니다.

예 7-6 종료 스크립트를 사용하여 시스템의 루트 비밀번호 설정

```
#!/bin/sh
#
#      @(#)set_root_pw 1.4 93/12/23 SMI
#
# This is an example Bourne shell script to be run after installation.
# It sets the system's root password to the entry defined in PASSWD.
# The encrypted password is obtained from an existing root password entry
# in /etc/shadow from an installed machine.
```

예 7-6 종료 스크립트를 사용하여 시스템의 루트 비밀번호 설정 (계속)

```

echo "setting password for root"

# set the root password
PASSWD=dK05IBkSF42lw
#create a temporary input file
cp /a/etc/shadow /a/etc/shadow.orig

mv /a/etc/shadow /a/etc/shadow.orig
nawk -F: '{
    if ( $1 == "root" )
        printf"%s:%s:%s:%s:%s:%s:%s:%s:%s\n", $1,passwd,$3,$4,$5,$6,$7,$8,$9
    else
        printf"%s:%s:%s:%s:%s:%s:%s:%s:%s\n", $1,$2,$3,$4,$5,$6,$7,$8,$9
}' passwd="$PASSWD" /a/etc/shadow.orig > /a/etc/shadow
#remove the temporary file
rm -f /a/etc/shadow.orig
# set the flag so sysidroot won't prompt for the root password
sed -e 's/0 # root/1 # root/' ${SI_SYS_STATE} > /tmp/state.$$
mv /tmp/state.$$ ${SI_SYS_STATE}

```

다음은 이 예의 일부 명령어에 대해 설명합니다.

- 다음 명령어는 변수 `PASSWD`를 시스템의 `/etc/shadow` 파일에 있는 기존 항목에서 얻은 암호화된 루트 비밀번호로 설정합니다.

```
#create a temporary input file
```

- 다음 명령어는 `/a/etc/shadow`의 임시 입력 파일을 만듭니다.

```
cp /a/etc/shadow /a/etc/shadow.orig
```

- 다음 명령어는 `$PASSWD`를 암호 필드로 사용하여 새로 설치된 시스템의 `/etc/shadow` 파일에서 루트 항목을 변경합니다.

```
if ( $1 == "root" )
```

- 다음 명령어는 임시 `/a/etc/shadow` 파일을 제거합니다.

```
rm -f /a/etc/shadow.orig
```

- 다음 명령어는 상태 파일의 항목을 0에서 1로 변경하여 사용자에게 루트 비밀번호 입력 요청을 하지 않도록 합니다. 상태 파일은 현재 값이 `/a/etc/.sysIDtool.state`인 변수 `SI_SYS_STATE`를 사용하여 액세스합니다. 스크립트 문제를 방지하려면 값이 변경되는 경우 항상 `$SI_SYS_STATE`를 사용하여 이 파일을 참조하십시오. 여기에 보이는 `sed` 명령어는 0 뒤와 1 뒤에 탭 문자가 포함되어 있습니다.

```
sed -e 's/0 # root/1 # root/' ${SI_SYS_STATE} > /tmp/state.$$
```

종료 스크립트를 사용하여 비대화식 설치

종료 스크립트를 사용하여 Solaris OS가 설치된 뒤 추가 소프트웨어를 설치할 수 있습니다. Solaris 설치 프로그램은 설치하는 동안 정보를 입력하라는 메시지를 표시합니다. 자동 설치를 유지하려면 Solaris 설치 프로그램을 `-nodisplay` 또는 `-noconsole` 옵션과 함께 실행하면 됩니다.

표 7-1 Solaris 설치 옵션

옵션	설명
<code>-nodisplay</code>	그래픽 사용자 인터페이스 없이 설치 프로그램을 실행합니다. 설치가 <code>-locales</code> 옵션에 의해 수정되지 않으면 기본 제품 설치를 사용합니다.
<code>-noconsole</code>	대화식 텍스트 콘솔 장치 없이 설치를 수행합니다. UNIX 스크립트 사용을 위해 <code>-nodisplay</code> 와 함께 실행할 때 유용합니다.

추가 정보는 `installer(1M)` 매뉴얼 페이지를 참조하십시오.

압축된 구성 파일 만들기

`add_install_client` 명령어를 사용하여 사용자 정의 JumpStart 구성 파일 위치를 지정하는 대신 시스템을 부트할 때 파일 위치를 지정할 수 있습니다. 하지만 이 경우에는 한 파일의 이름만 지정할 수 있습니다. 결과적으로 사용자 정의 JumpStart 구성 파일 모두를 하나의 파일로 압축해야 합니다.

- SPARC 기반 시스템의 경우, `boot` 명령어에 파일 위치를 지정합니다.
- x86 기반 시스템의 경우, GRUB 메뉴의 GRUB 항목을 편집하여 파일 위치를 지정합니다.

압축된 구성 파일은 다음 유형 중 하나일 수 있습니다.

- tar
- 압축된 tar
- zip
- bzip tar

▼ 압축된 구성 파일 만들기

- 1 디렉토리를 프로필 서버의 JumpStart 디렉토리로 변경합니다.

```
# cd jumpstart_dir_path
```
- 2 사용자 정의 JumpStart 구성 파일들을 하나의 파일로 압축하려면 압축 도구를 사용합니다.

주- 압축된 구성 파일은 관련 경로를 포함할 수 없습니다. 사용자 정의 JumpStart 구성 파일은 압축된 파일로서 동일한 디렉토리에 있어야 합니다.

압축된 구성 파일은 다음 파일을 포함해야 합니다.

- 프로필
- rules
- rules.ok

sysidcfg 파일도 압축된 구성 파일에 포함시킬 수 있습니다.

- 3 압축된 구성 파일을 NFS 서버, HTTP 서버 또는 로컬 하드 디스크에 저장합니다.

압축된 구성 파일의 예

다음은 tar 명령어를 사용하여 config.tar라는 이름의 압축된 구성 파일을 만드는 방법을 보여주는 예입니다. 사용자 정의 JumpStart 구성 파일은 /jumpstart 디렉토리에 있습니다.

예 7-7 압축된 구성 파일 만들기

```
# cd /jumpstart
# tar -cvf config.tar *
a profile 1K
a rules 1K
a rules.ok 1K
a sysidcfg 1K
```

디스크 구성 파일 만들기

이 절은 단일 디스크 및 다중 디스크 구성 파일을 만드는 방법에 대해 설명합니다. 디스크 구성 파일을 사용하여 단일 시스템에서 pfinstall(1M)을 사용하여 다른 디스크 구성에 대하여 프로필을 테스트할 수 있습니다.

▼ SPARC: 디스크 구성 파일을 만들기

- 1 테스트하려는 디스크를 가진 SPARC 기반 시스템을 찾습니다.
- 2 슈퍼유저가 되거나 동등한 역할을 맡습니다.
역할에는 권한 부여 및 권한이 있는 명령어가 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 3 prtvtoc(1M) 명령어의 출력을 파일로 재지정하여 단일 디스크 구성 파일을 만듭니다.

```
# prtvtoc /dev/rdsk/device_name >disk_config_file
```

`/dev/rdisk/device_name` 시스템 디스크의 장치 이름. `device_name`은 `cwt xdys2` 또는 `cxdys2` 형식이어야 합니다.

`disk_config_file` 디스크 구성 파일 이름

4 Solaris 소프트웨어를 여러 개의 디스크에 설치하는 것에 대한 테스트 여부를 결정합니다.

- 테스트를 원하지 않을 경우 중지합니다. 그렇게 하면 필요한 모든 과정이 완료됩니다.
- 테스트를 원할 경우 단일 디스크 구성 파일을 연결하고 새 파일에 출력을 저장합니다.

```
# cat disk_file1 disk_file2 >multi_disk_config
```

새 파일은 다음 예에서와 같이 다중 디스크 구성 파일이 됩니다.

```
# cat 104_disk2 104_disk3 104_disk5 >multi_disk_test
```

5 디스크 장치 이름의 대상 번호가 이전 단계에서 만든 다중 디스크 구성 파일에서 고유한 이름인지 확인합니다.

- 고유한 이름인 경우 중지합니다. 그렇게 하면 필요한 모든 과정이 완료됩니다.
- 고유한 이름이 아닌 경우 텍스트 편집기에서 파일을 열어 대상 번호를 고유한 디스크 장치 이름으로 만듭니다.

예를 들어, 다음과 같이 서로 다른 디스크 장치 이름에 대해 동일한 대상 번호인 `t0`이 파일에 포함되어 있다고 가정합니다.

```
* /dev/rdisk/c0t0d0s2 partition map
...
* /dev/rdisk/c0t0d0s2 partition map
```

두 번째 대상 번호를 다음과 같이 `t2`로 변경합니다.

```
* /dev/rdisk/c0t0d0s2 partition map
...
* /dev/rdisk/c0t2d0s2 partition map
```

SPARC: 디스크 구성 파일 예

다음은 단일 디스크 구성 파일, `104_test`를 104MB 디스크를 포함하는 SPARC 시스템에 만드는 방법을 보여주는 예입니다.

예 7-8 SPARC: 디스크 구성 파일 만들기

`prtvtoc` 명령어의 출력을 `104_test`라는 이름의 단일-디스크 구성 파일로 리디렉션합니다.

```
# prtvtoc /dev/rdisk/c0t3d0s2 >104_test
```

예 7-8 SPARC: 디스크 구성 파일 만들기 (계속)

104_test 파일의 내용은 다음과 유사합니다.

```
* /dev/rdisk/c0t3d0s2 partition map
*
* Dimensions:
*   512 bytes/sector
*   72 sectors/track
*   14 tracks/cylinder
* 1008 sectors/cylinder
* 2038 cylinders*   2036 accessible cylinders
* Flags:
*  1: unmountable
* 10: read-only
*
*
* Partition  Tag  Flags      First   Sector   Last
* Partition  Tag  Flags      Sector  Count    Sector  Mount Directory
*   1         2    00         0      164304   164303  /
*   2         5    00         0      2052288 2052287
*   3         0    00      164304   823536   987839  /disk2/b298
*   5         0    00      987840   614880  1602719 /install/298/sparc/work
*   7         0    00     1602720  449568  2052287  /space
```

SPARC 기반 시스템의 디스크 구성 파일이 작성되었습니다. 93 페이지 “프로필 테스트”는 디스크 구성 파일을 사용하여 프로필을 테스트하는 데 대한 정보를 포함합니다.

▼ x86: 디스크 구성 파일을 만들기

- 1 테스트하려는 디스크를 포함하는 x86 기반 시스템을 찾습니다.
- 2 슈퍼유저가 되거나 동등한 역할을 맡습니다.
역할에는 권한 부여 및 권한이 있는 명령어가 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 3 `fdisk(1M)` 명령어의 출력을 파일에 저장하여 단일 디스크 구성 파일의 일부를 만듭니다.
`fdisk -R -W disk_config_file -h /dev/rdsk/device_name`
`disk_config_file` 디스크 구성 파일의 이름
`/dev/rdsk/device_name` 전체 디스크의 fdisk 레이아웃의 장치 이름 `device_name`은 `c wtxdys0` 또는 `cx dys0` 형식이어야 합니다.
- 4 `prtvtoc(1M)` 명령어의 출력을 디스크 구성 파일에 추가합니다.
`prtvtoc /dev/rdsk/device_name >>disk_config`

`/dev/rdisk/device_name` 시스템 디스크의 장치 이름. `device_name`은 `cwt xdys2` 또는 `cxdys2` 형식이어야 합니다.

`disk_config` 디스크 구성 파일 이름

5 Solaris 소프트웨어를 여러 개의 디스크에 설치하는 것에 대한 테스트 여부를 결정합니다.

- 테스트를 원하지 않을 경우 중지합니다. 그렇게 하면 필요한 모든 과정이 완료됩니다.
- 테스트를 원할 경우 단일 디스크 구성 파일을 연결하고 새 파일에 출력을 저장합니다.

```
# cat disk_file1 disk_file2 >multi_disk_config
```

새 파일은 다음 예에서와 같이 다중 디스크 구성 파일이 됩니다.

```
# cat 104_disk2 104_disk3 104_disk5 >multi_disk_test
```

6 디스크 장치 이름의 대상 번호가 이전 단계에서 만든 다중 디스크 구성 파일에서 고유한 이름인지 확인합니다.

- 고유한 이름인 경우 중지합니다. 그렇게 하면 필요한 모든 과정이 완료됩니다.
- 고유한 이름이 아닌 경우 텍스트 편집기에서 파일을 열어 대상 번호를 고유하게 만듭니다. 예를 들어 파일은 다음과 같이 다른 디스크 장치 이름에 대하여 동일한 대상 번호, `t0`을 포함할 수 있습니다.

```
* /dev/rdisk/c0t0d0s2 partition map
```

```
...
```

```
* /dev/rdisk/c0t0d0s2 partition map
```

두 번째 대상 번호를 다음과 같이 `t2`로 변경합니다.

```
* /dev/rdisk/c0t0d0s2 partition map
```

```
...
```

```
* /dev/rdisk/c0t2d0s2 partition map
```

x86: 디스크 구성 파일 예

다음은 단일 디스크 구성 파일, `500_test`를 500MB 디스크를 포함하는 x86 기반 시스템에 만드는 방법을 보여주는 예입니다.

예 7-9 x86: 디스크 구성 파일 만들기

먼저 다음과 같이 `fdisk` 명령어의 출력을 이름이 `500_test`인 파일에 저장합니다.

```
# fdisk -R -W 500_test -h /dev/rdisk/c0t0d0p0
```

`500_test` 파일이 다음과 같습니다.

예 7-9 x86: 디스크 구성 파일 만들기 (계속)

```
* /dev/rdisk/c0t0d0p0 default fdisk table
* Dimensions:
*   512 bytes/sector
*   94 sectors/track
*   15 tracks/cylinder
* 1455 cylinders
*
* HBA Dimensions:
*   512 bytes/sector
*   94 sectors/track
*   15 tracks/cylinder
* 1455 cylinders
*
* systid:
* 1:  DOS0S12
* 2:  PCIX0S
* 4:  DOS0S16
* 5:  EXTDOS
* 6:  DOSBIG
* 86: DOSDATA
* 98: OTHEROS
* 99: UNIXOS
* 130: SUNIXOS
*
* Id Act Bhead Bsect  Bcyl  Ehead  Esect  Ecyl Rsect  Numsect
130 128 44   3     0    46    30    1001 1410  2050140
```

두 번째로, prtvtoc 명령어의 출력을 500_test 파일에 추가합니다.

```
# prtvtoc /dev/rdisk/c0t0d0s2 >>500_test
```

이제 500_test 파일이 완전한 디스크 구성 파일입니다.

```
* /dev/rdisk/c0t0d0p0 default fdisk table
* Dimensions:
*   512 bytes/sector
*   94 sectors/track
*   15 tracks/cylinder
* 1455 cylinders
*
* HBA Dimensions:
*   512 bytes/sector
*   94 sectors/track
*   15 tracks/cylinder
* 1455 cylinders
```


예 7-9 x86: 디스크 구성 파일 만들기 (계속)

```

*
* systid:
* 1:  DOS0S12
* 2:  PCIX0S
* 4:  DOS0S16
* 5:  EXTDOS
* 6:  DOSBIG
* 86: DOSDATA
* 98: OTHEROS
* 99: UNIXOS
* 130: SUNIXOS
*
* Id  Act  Bhead Bsect Bcyl  Ehead  Esec  Ecyl Rsect  Numsect
130  128  44   3    0   46   30   1001 1410   2050140
* /dev/rdsk/c0t0d0s2 partition map
*
* Dimensions:
*   512 bytes/sector
*   94 sectors/track
*   15 tracks/cylinder
*  1110 sectors/cylinder
*  1454 cylinders
*  1452 accessible cylinders
*
* Flags:
* 1: unmountable
* 10: read-only
*
* Partition  Tag  Flags  First  Sector  Last  Mount Directory
*   2         5   01    1410  2045910 2047319
*   7         6   00    4230  2043090 2047319 /space
*   8         1   01     0    1410    1409
*   9         9   01    1410    2820   422987
    
```

x86 기반 시스템에 대하여 디스크 구성 파일을 만들었습니다. 93 페이지 “프로필 테스트”는 디스크 구성 파일을 사용하여 프로필을 테스트하는 데 대한 정보를 포함합니다.

사이트별 설치 프로그램 사용

시작 및 종료 스크립트를 사용하여 사용자 고유의 Solaris 소프트웨어 설치 프로그램을 만들 수 있습니다.

profile 필드에 음수 기호(-)를 지정하면 시작 및 종료 스크립트가 프로필과 Solaris 설치 프로그램 대신 Solarils 소프트웨어가 시스템에 설치되는 방법을 제어합니다.

예를 들어, 다음 규칙이 시스템에 적용되면 `x_install.beg` 시작 스크립트 및 `x_install.fin` 종료 스크립트는 Solaris 소프트웨어를 `clover`라는 이름의 시스템에 설치합니다.

```
hostname clover x_install.beg - x_install.fin
```

사용자 정의 규칙 및 검사 키워드 만들기(작업)

이 장에서는 사용자 정의 규칙과 검사 키워드를 만드는 정보 및 절차에 대해 설명합니다.

- 115 페이지 “검사 키워드”
- 116 페이지 “custom_probes 파일 만들기”
- 118 페이지 “custom_probes 파일 검증”

검사 키워드

검사 키워드를 이해하려면 먼저 규칙 키워드가 무엇인지 기억해야 합니다. 규칙 키워드는 호스트 이름 `hostname`, 또는 메모리 크기 `memsize`와 같은 일반 시스템 속성을 설명하는 미리 정의된 어휘 단위 또는 단어입니다. 규칙 키워드와 그와 연관된 값을 사용하여 동일한 속성을 가진 시스템을 프로필에 일치시킬 수 있습니다. 이 시스템 속성의 일치는 Solaris 소프트웨어가 그룹의 각 시스템에 설치되는 방법을 정의합니다.

시작 스크립트 및 종료 스크립트에 사용하는 사용자 정의 JumpStart 환경 변수는 요구 시 설정됩니다. 예를 들어, 시스템에 이미 설치된 운영 체제에 대한 정보는 `installed` 규칙 키워드가 사용된 후 `SI_INSTALLED`에서만 사용 가능합니다.

일부 상황에서는 시스템을 일치시키고 프로필을 실행하는 것 외의 목적으로 시작 스크립트 또는 종료 스크립트에서 동일한 정보를 추출해야 할 수 있습니다. 검사 키워드가 솔루션을 제공합니다. 검사 키워드는 속성 정보를 추출하고 일치 조건 설정 및 프로필 실행의 필요성을 제거시켜 줍니다.

검사 키워드 및 값 목록은 189 페이지 “검사 키워드 및 값”을 참조하십시오.

custom_probes 파일 만들기

145 페이지 “규칙 키워드 및 값”과 189 페이지 “검사 키워드 및 값”에서 설명되는 규칙 및 검사 키워드는 사용자의 필요에 맞을 만큼 정확하지 않을 수 있습니다. custom_probes 파일을 만들기하여 사용자 자신의 사용자 정의 규칙 또는 검사 키워드를 정의할 수 있습니다.

custom_probes 파일은 두 가지 유형의 함수를 포함하는 Bourne 셸 스크립트입니다. rules 파일을 저장한 동일한 JumpStart 디렉토리에 custom_probes 파일을 저장해야 합니다. custom_probes 파일에서 정의할 수 있는 두 가지 유형의 함수는 다음과 같습니다.

- 검사 - 원하는 정보를 모으거나 실제 작업을 수행하고 사용자가 정의한 해당 SI_ 환경 변수를 설정합니다. 검사 함수는 검사 키워드가 됩니다.
- 비교 - 해당하는 검사 함수를 호출하고 검사 함수의 출력을 비교하고 키워드가 일치하면 0을, 키워드가 일치하지 않으면 1을 반환합니다. 비교 함수는 규칙 키워드가 됩니다.

custom_probes 파일의 구문

custom_probes 파일은 모든 유효한 Bourne 셸 명령어, 변수 또는 알고리즘을 포함할 수 있습니다.

주 - custom_probes 파일의 단일 인자를 필요로 하는 검사 및 비교 함수를 정의할 수 있습니다. rules 파일에서 해당하는 사용자 정의 검사 키워드를 사용하는 경우 키워드 뒤의 인자는 \$1로 해석됩니다.

rules 파일의 해당하는 사용자 정의 키워드를 사용할 경우 인자는 순차적으로 해석됩니다. 시퀀스는 키워드 뒤에서 시작되고 && 또는 시작 스크립트 앞(둘 중 앞에 오는 것 우선)에서 끝납니다.

custom_probes 파일은 다음 요구 사항을 만족해야 합니다.

- custom_probes 이름을 가짐
- root를 소유자로 가짐
- 실행 가능하며 755로 설정된 권한을 가짐
- 적어도 하나의 검사 함수와 하나의 상응하는 비교 함수를 포함

투명성과 구성을 향상시키기 위해서는 파일 상단에서 모든 검사 함수를 먼저, 모든 비교 함수 앞에서 정의하십시오.

custom_probes의 함수 이름 구문

검사 함수의 이름은 probe_로 시작해야 합니다. 비교 함수의 이름은 cmp_로 시작해야 합니다.

probe_로 시작하는 함수는 새 검사 키워드를 정의합니다. 예를 들어, 함수 probe_tcx는 새 검사 키워드 tcx를 정의합니다. cmp_로 시작하는 함수는 새 규칙 키워드를 정의합니다. 예를 들어, cmp_tcx는 새 규칙 키워드 tcx를 정의합니다.

▼ custom_probes 파일 만들기

- 1 텍스트 편집기를 사용하여 **Bourne** 셸 스크립트 텍스트 파일을 만듭니다. 파일 `custom_probes`의 이름을 지정합니다.
- 2 `custom_probes` 텍스트 파일에서 검사 및 비교 함수를 정의합니다.

주 - `custom_probes` 파일에서 인자를 필요로 하는 검사 및 비교 함수를 정의할 수 있습니다. `rules` 파일에서 해당하는 사용자 정의 검사 키워드를 사용하는 경우 키워드 뒤의 인자는 `$1`, `$2` 등과 같이 시퀀스로 해석됩니다.

`rules` 파일의 해당하는 사용자 정의 키워드를 사용할 경우 인자는 순차적으로 해석됩니다. 시퀀스는 키워드 뒤에서 시작되고 `&&` 또는 시작 스크립트 앞(둘 중 앞에 오는 것 우선)에서 끝납니다.

- 3 `custom_probes` 파일을 `rules` 파일 다음의 **JumpStart** 디렉토리에 저장합니다.
- 4 `root`가 `rules` 파일을 소유하고 권한이 **644**로 설정되도록 하십시오.

custom_probes 파일 및 키워드의 예

다음 디렉토리에서 검사 및 비교 함수의 추가 예를 찾을 수 있습니다.

- Solaris 소프트웨어가 설치된 시스템의 `/usr/sbin/install.d/chkprobe`
- Solaris Operating System DVD 또는 Solaris Software - 1 CD의 `/Solaris_10_606/Tools/Boot/usr/sbin/install.d/chkprobe`

다음 `custom_probes` 파일은 TCX 그래픽 카드의 존재를 테스트하는 검사 및 비교 함수를 포함합니다.

예 8-1 `custom_probes` 파일

```
#!/bin/sh
#
# custom_probe script to test for the presence of a TCX graphics card.
#
#
# PROBE FUNCTIONS
#
probe_tcx() {
    SI_TCX='modinfo | grep tcx | nawk '{print $6}''
    export SI_TCX
}
```

예 8-1 custom_probes 파일 (계속)

```
#
# COMPARISON FUNCTIONS
#
cmp_tcx() {
    probe_tcx

    if [ "$${SI_TCX}" = "$${1}" ]; then
        return 0
    else
        return 1
    fi
}
```

예로 든 다음 rules 파일은 예로 든 이전 tcx에서 정의된 검사 키워드의 사용을 보여 줍니다. TCX 그래픽 카드가 시스템에 설치되어 있고 확인된 경우 profile_tcx가 실행됩니다. 그렇지 않으면, profile이 실행됩니다.

주 - probe 키워드는 rules 파일에 또는 그 가까이 배치하십시오. 이 배치는 키워드가 검사 키워드에 의존할 수 있는 다른 규칙 키워드 앞에서 읽히고 실행되도록 합니다.

예 8-2 rules 파일에 사용되는 사용자 정의 검사 키워드

```
probe tcx
tcx    tcx    -    profile_tcx    -
any    any    -    profile        -
```

custom_probes 파일 검증

프로필, rules 및 custom_probes 파일을 사용하려면 먼저 check 스크립트를 실행하여 파일이 제대로 설정되었는지 검증해야 합니다. 모든 프로필, 규칙, 그리고 검사 및 비교 함수가 제대로 설치되면 rules.ok 및 custom_probes.ok 파일이 생성됩니다. 표 8-1에서는 check 스크립트가 수행하는 일에 대해 설명합니다.

표 8-1 check 스크립트를 사용하는 경우 수행되는 작업

단계	설명
1	check가 custom_probes 파일을 검색합니다.

표 8-1 check 스크립트를 사용하는 경우 수행되는 작업 (계속)

단계	설명
2	파일이 존재하면 check는 custom_probes 파일에서 custom_probes.ok 파일을 만들고, 모든 주석과 공백 줄을 제거하고, 모든 Bourne 셸 명령어, 변수 및 알고리즘을 보관합니다. 그런 다음 check가 끝에 다음 명령줄을 추가합니다. # version=2 checksum=num

▼ custom_probes 파일 검증

- 1 check 스크립트가 JumpStart 디렉토리에 위치하는 것을 검증합니다.

주 - check 스크립트는 Solaris Operating System DVD 또는 Solaris Software - 1 CD의 Solaris_10_606/Misc/jumpstart_sample 디렉토리에 있습니다.

- 2 JumpStart 디렉토리로 변경합니다.

- 3 check 스크립트를 실행하여 rules 및 custom_probes 파일을 검증합니다.

```
$ ./check -p path -r file_name
```

-p path 사용 중인 시스템의 check 스크립트 대신 사용자 플랫폼용 Solaris 소프트웨어 이미지의 check 스크립트를 사용하여 custom_probes 파일을 검증합니다. path는 로컬 디스크 또는 마운트된 Solaris Operating System DVD 또는 Solaris Software - 1 CD의 이미지입니다.

시스템이 이전 Solaris 버전에서 실행되고 있는 경우 이 옵션을 사용하여 가장 최신 check 버전을 실행합니다.

-r file_name custom_probes가 아닌 파일 이름을 지정합니다. -r 옵션을 사용하여 함수를 custom_probes 파일로 통합하기 전에 함수 세트의 유효성을 테스트할 수 있습니다.

check 스크립트가 실행되면, 스크립트는 rules 및 custom_probes 파일과 각 프로파일의 유효성을 보고합니다. 오류가 확인되지 않으면 스크립트는 다음을 보고합니다. “JumpStart 구성이 사용자 정의되었습니다” 를 보고하고 JumpStart 디렉토리에 rules.ok 및 custom_probes.ok 파일을 만듭니다.

- 4 custom_probes.ok 파일이 실행 가능한지 결정합니다.

- 예, 단계 5로 이동합니다.
- 아니요, 다음 명령어를 입력합니다.

```
# chmod +x custom_probes
```

- 5 root가 custom_probes.ok 파일을 소유하고 권한이 755로 설정되도록 합니다.

사용자 정의 JumpStart 설치 수행(작업)

이 장에서는 SPARC 기반 또는 x86 기반 시스템에서 사용자 정의 JumpStart 설치 수행 방법에 대해 설명합니다. Solaris 소프트웨어를 설치할 시스템에서 이 절차를 따라야 합니다.

- 125 페이지 “SPARC: 사용자 정의 JumpStart 프로그램을 사용하여 설치 또는 업그레이드 수행하기”
- 129 페이지 “x86: 사용자 정의 JumpStart 프로그램 및 GRUB를 사용하여 설치 또는 업그레이드 수행”

SPARC: 작업 맵: 사용자 정의 JumpStart 설치를 위한 시스템 설정

표 9-1 작업 맵: 사용자 정의 JumpStart 설치를 위한 시스템 설정

작업	설명	지침
시스템이 지원되는지 검사	Solaris 환경의 시스템 지원은 하드웨어 설명서를 확인합니다.	http://docs.sun.com 의 Solaris 10 6/06 Sun 하드웨어 플랫폼 안내서
시스템에 Solaris 소프트웨어를 위한 충분한 디스크 공간이 있는지 확인합니다.	시스템에 Solaris 소프트웨어를 설치할 충분한 공간이 확보되어 있는지 확인합니다.	3 장
(선택 사항) 시스템 매개변수를 설정합니다.	시스템 정보를 사전 구성함으로써 설치나 업그레이드 중에 정보 입력을 요청받게 되는 일을 방지할 수 있습니다.	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 6 장, “시스템 구성 정보를 미리 구성(작업)”
사용자 정의 JumpStart 설치를 위해 시스템을 준비합니다.	rules 파일과 프로필 파일을 만들어 검증합니다.	6 장

표 9-1 작업 맵: 사용자 정의 JumpStart 설치를 위한 시스템 설정 (계속)

작업	설명	지침
(선택 사항) 선택적 사용자 정의 JumpStart 기능 준비	시작 스크립트, 종료 스크립트 또는 다른 선택적 기능을 사용 중인 경우 해당 스크립트 또는 파일을 준비합니다.	7 장 및 8 장
(선택 사항) 네트워크를 통한 Solaris 소프트웨어 설치를 준비합니다.	원격 Solaris Operating System DVD 또는 Solaris Software for SPARC Platforms CD 이미지에서 시스템을 설치하려면 부트할 시스템을 설정하고 설치 서버 또는 부트 서버에서 설치해야 합니다.	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 8 장, “DVD 매체를 사용한 네트워크에서의 설치 준비(작업)” Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 9 장, “CD 매체를 사용하여 네트워크 설치 준비(작업)”
(선택 사항) Solaris Flash 아카이브 설치를 준비합니다.	Solaris Flash 아카이브 설치를 위한 특정 사항을 설정합니다.	122 페이지 “사용자 정의 JumpStart 설치로 Solaris Flash 아카이브 설치 준비하기”
설치 또는 업그레이드를 수행합니다.	시스템을 부트하여 설치 또는 업그레이드를 시작합니다.	125 페이지 “SPARC: 사용자 정의 JumpStart 프로그램을 사용하여 설치 또는 업그레이드 수행하기”

SPARC: 사용자 정의 JumpStart 설치 수행

사용자 정의 JumpStart 설치 중에 JumpStart 프로그램은 설치되는 시스템을 `rules.ok` 파일의 규칙에 일치시키려고 합니다. JumpStart 프로그램이 첫 번째 규칙에서 마지막 규칙까지 모든 규칙을 읽습니다. 설치되는 시스템이 규칙에 정의된 모든 시스템 속성과 일치하면 일치가 이루어집니다. 시스템이 규칙에 일치하면 JumpStart 프로그램이 `rules.ok` 파일 읽기를 정지하고 일치된 규칙의 프로필을 기준으로 시스템을 설치하기 시작합니다.

▼ 사용자 정의 JumpStart 설치로 Solaris Flash 아카이브 설치 준비하기

초기 설치를 위해, 또는 아카이브, 즉 업데이트용 차등 아카이브를 이미 설치한 경우 전체 아카이브를 설치할 수 있습니다. 사용자 정의 JumpStart 설치 방법을 사용하거나 Solaris Live Upgrade를 사용하여 비활성 부트 환경에 아카이브를 설치할 수 있습니다. 이 절차는 사용자 정의 JumpStart를 사용하는 아카이브 설치의 지침을 제공합니다.

- 전체 또는 차등 아카이브의 개요에 대해서는 **Solaris 10 6/06** 설치 설명서: **Solaris Flash** 아카이브(작성 및 설치)의 1 장, “Solaris Flash(개요)”를 참조하십시오.

- Solaris Live Upgrade를 사용하여 비활성 부트 환경에 아카이브를 설치하는 절차에 대해서는 **Solaris 10 6/06 설치 설명서: Solaris Live Upgrade** 및 업그레이드 계획의 “프로필을 사용하여 Solaris Flash 아카이브 설치(명령줄 인터페이스)”를 참조하십시오.

1 다음 제한을 검토합니다.

설명	예
주의:archive_location 키워드를 사용하여 Solaris Flash 아카이브를 설치할 경우 아카이브와 설치 매체에 동일한 운영 체제 버전이 포함되어 있어야 합니다.	예를 들어, 아카이브의 운영 체제가 Solaris 10이고 DVD 매체를 사용하는 경우에는 Solaris 10 DVD 매체를 사용하여 아카이브를 설치해야 합니다. 운영 체제 버전이 일치하지 않으면 복제 시스템 설치에 실패합니다.
주의 - 비전역 영역이 설치되어 있는 경우 Solaris Flash 아카이브를 제대로 만들 수 없습니다. Solaris Flash 기능은 Solaris 영역 분할 기술과 호환되지 않습니다. Solaris Flash 아카이브를 만들 때 다음 조건 하에서 아카이브를 배포하면 결과 아카이브가 제대로 설치되지 않습니다.	
<ul style="list-style-type: none"> ■ 아카이브를 비전역 영역에 만든 경우 ■ 아카이브를 비전역 영역이 설치된 전역 영역에 만든 경우 	

2 설치 서버에서 사용자 정의 JumpStart rules 파일을 만듭니다.

사용자 정의 JumpStart 파일 만들기에 대한 자세한 지침은 6 장을 참조하십시오.

3 설치 서버에서 사용자 정의 JumpStart 프로필 파일을 만듭니다.

Solaris Flash 아카이브 프로필의 예는 83 페이지 “프로필의 예”를 참조하십시오.

표 11-2의 사용자 정의 JumpStart 키워드 기존 목록에서 Solaris Flash 아카이브를 설치할 때 유효한 키워드는 다음과 같습니다.

키워드	초기 설치	차등 아카이브
(필수) archive_location	X	X
fdisk(x86 전용)	X	X
filesys	X	
주 - auto값에 filesys 키워드를 설정할 수 없습니다.		
forced_deployment		X
(필수) install_type	X	X

키워드	초기 설치	차등 아카이브
local_customization	X	X
no_content_check		X
no_master_check		X
package	X	
root_device	X	X

a. `install_type` 키워드의 값을 다음 중 한 유형으로 설정합니다.

- 전체 아카이브를 설치하려면 해당 값을 `flash_install`에 설정합니다.
- 차등 아카이브를 설치하려면 값을 `flash_update`로 설정합니다.

b. `archive_location` 키워드를 사용하여 Solaris Flash 아카이브에 경로를 추가합니다.

`archive_location` 키워드에 대한 자세한 내용은 151 페이지 “`archive_location` 키워드”를 참조하십시오.

c. 파일 시스템 구성을 지정합니다.

Solaris Flash 아카이브 추출 프로세스는 분할 영역의 자동 레이아웃을 지원하지 않습니다.

d. (선택 사항) 아카이브를 설치하는 동시에 추가 패키지를 설치하려는 경우 `package` 키워드를 사용하십시오. 자세한 내용은 175 페이지 “`package` 프로필 키워드”를 참조하십시오.

e. (선택 사항) 복제 시스템에 추가 Solaris Flash 아카이브를 설치하려면 설치할 각각의 아카이브에 `archive_location` 행을 하나 추가합니다.

4 설치 서버에서 Solaris Flash 아카이브와 함께 설치할 클라이언트를 추가합니다.

자세한 지침은 다음을 참조하십시오.

- Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 “DVD 이미지로 네트워크에서 설치할 시스템 추가”
- Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 “CD 이미지로 네트워크에서 설치할 시스템 추가”

5 복제 시스템에서 사용자 정의 JumpStart 설치를 수행합니다.

자세한 지침은 125 페이지 “SPARC: 사용자 정의 JumpStart 프로그램을 사용하여 설치 또는 업그레이드 수행하기”를 참조하십시오.

▼ SPARC: 사용자 정의 JumpStart 프로그램을 사용하여 설치 또는 업그레이드 수행하기

- 1 시스템이 네트워크의 일부인 경우 이더넷 커넥터 또는 유사한 네트워크 어댑터가 시스템에 연결되어 있는지 확인합니다.
- 2 tip(1) 행을 통해 연결된 시스템을 설치할 경우 윈도우 화면이 최소한 가로 80열, 세로 24행인지 확인하십시오.
tip 창의 현재 크기를 결정하려면 stty(1) 명령어를 사용합니다.
- 3 시스템의 DVD-ROM 또는 CD-ROM 드라이브를 사용하여 Solaris 소프트웨어를 설치 중인 경우 드라이브에 Solaris Operating System for SPARC Platforms DVD 또는 Solaris Software for SPARC Platforms - 1 CD를 삽입합니다.
- 4 플로피 디스켓을 사용 중인 경우 플로피 디스켓을 시스템의 디스켓 드라이브에 삽입합니다.
- 5 시스템을 부트합니다.
 - 새로 구입한 시스템의 경우 시스템을 켭니다.
 - 기존 시스템을 설치 또는 업그레이드하려면 시스템을 종료합니다. ok 프롬프트에서 부트 명령어에 대한 적절한 옵션을 입력합니다. 다음은 boot 명령어 구문입니다.

```
ok boot [cd-dvd|net] - install [url|ask] options
```

예를 들어, 다음 명령어를 입력하면 JumpStart 프로필을 사용하여 네트워크를 통해 OS가 설치됩니다.

```
ok boot net - install http://131.141.2.32/jumpstart/config.tar
```

부트 명령어 옵션에 대한 설명은 다음 표를 참조하십시오.

SPARC 전용 - 시스템이 하드웨어 및 시스템 구성 요소를 검사하고 사용자의 SPARC 시스템이 부트됩니다. 부트하는 데 몇 분 정도 소요됩니다.

- 6 sysidcfg 파일에서 시스템 정보를 미리 구성하지 않은 경우, 메시지가 나타났을 때 시스템 구성에 대한 질문에 답합니다.
- 7 화면에 나타난 지침에 따라 소프트웨어를 설치합니다.
JumpStart 프로그램이 Solaris 소프트웨어 설치를 완료하면 시스템이 자동으로 재부트됩니다. 설치가 끝나면 설치 로그가 파일에 저장됩니다. 설치 로그는 다음 디렉토리에서 찾을 수 있습니다.
 - /var/sadm/system/logs
 - /var/sadm/install/logs

SPARC: boot 명령어의 명령어 참조

다음은 boot 명령어 구문입니다.

```
ok boot [cd-dvd|net] - install [url|ask] options
```

다음 테이블에서는 JumpStart 설치에 대해 적절한 boot 명령어에 대한 명령줄 옵션을 설명합니다.

옵션	설명
[cd-dvd net]	<p>CD 또는 DVD로부터 부트할지 또는 네트워크 상의 설치 서버로부터 부트할지를 지정합니다.</p> <ul style="list-style-type: none"> ■ <i>cd-dvd</i> - CD 또는 DVD로부터 부트하려면 <i>cdrom</i>을 사용합니다. ■ <i>net</i> - 네트워크 상의 설치 서버로부터 부트하도록 지정합니다.
[url ask]	<p>사용자 정의 JumpStart 파일의 위치를 지정하거나 위치를 지정하라는 프롬프트를 표시합니다.</p> <ul style="list-style-type: none"> ■ <i>url</i> - 파일 경로를 지정합니다. HTTP 또는 HTTPS 서버에 있는 파일에 URL을 지정할 수 있습니다. HTTP 서버 <i>http://server_name:IP_address/jumpstart_dir_path/compressed_config_file&proxy_info</i> <ul style="list-style-type: none"> ■ <i>sysidcfg</i> 파일을 압축된 구성 파일에 둔 경우, 다음 예에서와 같이 해당 파일을 포함한 서버의 IP 주소를 지정해야 합니다. <i>http://131.141.2.32/jumpstart/config.tar</i> ■ 방화벽 뒤에 있는 HTTP 서버에 압축된 구성 파일을 저장했으면 부트 동안 프록시 지정자를 사용해야 합니다. 파일을 포함하는 서버에 대하여 IP 주소를 지정할 필요가 없습니다. 다음 예에서와 같이 프록시 서버에 대하여 IP 주소를 지정해야 합니다. <i>http://www.shadow.com/jumpstart/config.tar&proxy=131.141.6.151</i> ■ <i>ask</i> - 압축 구성 파일의 위치를 입력하라는 메시지를 표시하도록 설치 프로그램을 지정합니다. 프롬프트 메시지는 시스템이 부트하고 네트워크에 연결한 후 표시됩니다. 이 옵션을 사용하면 완전한 핸드 오프 JumpStart 설치 방식으로 설치할 수 없습니다. Return 버튼을 눌러 프롬프트를 우회하면 Solaris 설치 프로그램은 네트워크 매개 변수를 대화식으로 구성합니다. 그러면 설치 프로그램이 압축된 구성 파일의 위치를 묻는 메시지를 표시합니다.

옵션	설명
<i>options</i>	<ul style="list-style-type: none"> ■ <code>dhcp</code> - 시스템을 부트하는 데 필요한 네트워크 설치 정보를 얻을 때 DHCP 서버를 사용하도록 지정합니다. 이 옵션은 JumpStart 설치에 필요하지 않습니다. <code>dhcp</code>를 입력하여 DHCP 서버를 사용하도록 지정하지 않은 경우 시스템은 <code>/etc/bootparams</code> 파일이나 <code>bootparams</code> 이름 서비스 데이터베이스를 사용합니다. 예를 들어 정적 IP 주소를 유지하고자 하는 경우 <code>dhcp</code>를 지정하지 않을 것입니다. ■ <code>nowin</code>과 <code>text</code> 옵션은 JumpStart 설치에 적용되지 않습니다. 이러한 옵션은 대화식 설치에 유용합니다. 자세한 내용은 Solaris 10 6/06 설치 설명서: 기본 설치의 “Solaris 설치 프로그램을 사용하여 설치 또는 업그레이드”를 참조하십시오.

x86: 작업 맵: 사용자 정의 JumpStart 설치를 위한 시스템 설정

표 9-2 x86: 작업 맵: 사용자 정의 JumpStart 설치를 위한 시스템 설정

작업	설명	지침
기존 운영 체제 및 사용자 데이터를 보존해야 하는지 결정합니다.	시스템의 기존 운영 체제에서 전체 디스크를 사용하는 경우 Solaris 10 6/06 소프트웨어를 사용할 수 있도록 기존 운영 체제를 보존해야 합니다. 이 결정이 시스템 프로파일의 <code>fdisk(1M)</code> 키워드를 지정하는 방법을 결정합니다.	163 페이지 “x86: <code>fdisk</code> 프로파일 키워드”
시스템이 지원되는지 검사	Solaris 환경의 시스템 지원은 하드웨어 설명서를 확인합니다.	하드웨어 제조업체 설명서
시스템에 Solaris 소프트웨어를 위한 충분한 디스크 공간이 있는지 확인합니다.	시스템에 Solaris 소프트웨어를 설치할 충분한 공간이 확보되어 있는지 확인합니다.	3 장
(선택 사항) 시스템 매개변수를 설정합니다.	시스템 정보를 사전 구성함으로써 설치나 업그레이드 중에 정보 입력을 요청받게 되는 일을 방지할 수 있습니다.	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 6 장, “시스템 구성 정보를 미리 구성(작업)”
사용자 정의 JumpStart 설치를 위해 시스템을 준비합니다.	<code>rules</code> 파일과 프로파일 파일을 만들어 검증합니다.	6 장

표 9-2 x86: 작업 맵: 사용자 정의 JumpStart 설치를 위한 시스템 설정 (계속)

작업	설명	지침
(선택 사항) 선택적 사용자 정의 JumpStart 기능 준비	시작 스크립트, 종료 스크립트 또는 다른 선택적 기능을 사용 중인 경우 해당 스크립트 또는 파일을 준비합니다.	7 장 및 8 장
(선택 사항) 네트워크를 통한 Solaris 소프트웨어 설치를 준비합니다.	원격 Solaris Operating System for x86 Platforms DVD 또는 Solaris Software For x86 Platforms CD 이미지에서 시스템을 설치하려면 시스템이 부트하도록 설정하고 설치 서버 또는 부트 서버에서 설치해야 합니다.	Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 9 장, “CD 매체를 사용하여 네트워크 설치 준비(작업)”
(선택 사항) Solaris Flash 아카이브 설치를 준비합니다.	Solaris Flash 아카이브 설치를 위한 특정 사항을 설정합니다.	122 페이지 “사용자 정의 JumpStart 설치로 Solaris Flash 아카이브 설치 준비하기”
설치 또는 업그레이드를 수행합니다.	시스템을 부트하여 설치 또는 업그레이드를 시작합니다.	129 페이지 “x86: 사용자 정의 JumpStart 프로그램 및 GRUB를 사용하여 설치 또는 업그레이드 수행”

x86: 사용자 정의 JumpStart 설치 수행

사용자 정의 JumpStart 설치 중에 JumpStart 프로그램은 설치되는 시스템을 `rules.ok` 파일의 규칙에 일치시키려고 합니다. JumpStart 프로그램이 첫 번째 규칙에서 마지막 규칙까지 모든 규칙을 읽습니다. 설치되는 시스템이 규칙에 정의된 모든 시스템 속성과 일치하면 일치가 이루어집니다. 시스템이 규칙에 일치하자마자 JumpStart 프로그램이 `rules.ok` 파일 읽기를 정지하고 일치된 규칙의 프로필을 기준으로 시스템을 설치하기 시작합니다.

사용자 정의 JumpStart를 사용하여 Solaris Flash 아카이브를 설치할 수 있습니다. 자세한 내용은 122 페이지 “사용자 정의 JumpStart 설치로 Solaris Flash 아카이브 설치 준비하기”를 참조하십시오.

다음 절차 중 하나를 선택합니다.

- 표준 사용자 정의 JumpStart 절차는 129 페이지 “x86: 사용자 정의 JumpStart 프로그램 및 GRUB를 사용하여 설치 또는 업그레이드 수행”을 참조하십시오.
- GRUB 명령어를 편집하여 사용자 정의 JumpStart 설치를 수행하려면 131 페이지 “x86: GRUB 부트 명령어를 편집하여 사용자 정의 JumpStart 설치 수행”을 참조하십시오.

▼ x86: 사용자 정의 JumpStart 프로그램 및 GRUB를 사용하여 설치 또는 업그레이드 수행

GRUB 메뉴를 사용하여 x86 기반 시스템용 Solaris OS를 설치하려면 이 절차를 사용합니다.

- 1 시스템이 네트워크의 일부인 경우 이더넷 커넥터 또는 유사한 네트워크 어댑터가 시스템에 연결되어 있는지 확인합니다.
- 2 tip(1) 행을 통해 연결된 시스템을 설치할 경우 윈도우 화면이 최소한 가로 80열, 세로 24행인지 확인하십시오.
tip 창의 현재 크기를 결정하려면 stty(1) 명령어를 사용합니다.
- 3 프로필 디스켓을 사용할지 결정합니다.
프로파일 디스켓은 더 이상 시스템 부트에 사용되지 않지만 JumpStart 디렉토리만 포함된 디스켓을 준비할 수 있습니다. 준비된 디스켓은 JumpStart 설치를 수행하거나 CD-ROM이 아닌 위치에서 부트하는 등의 경우에 사용됩니다.
 - 프로필 디스켓을 사용 중인 경우 프로필 디스켓을 시스템의 디스켓 드라이브에 삽입합니다.
 - 프로필 디스켓을 사용하지 않는 경우에는 단계 4로 계속합니다.
- 4 시스템 부트 방법을 결정합니다.
 - Solaris Operating System DVD 또는 Solaris Software - 1 CD에서 부트하려면 디스크를 삽입합니다. 시스템의 BIOS가 DVD 또는 CD에서 부트를 지원해야 합니다.
 - 네트워크에서 부트하는 경우 PXE(Preboot Execution Environment) 네트워크 부트를 사용합니다. 시스템이 PXE를 지원해야 합니다. 시스템이 시스템의 BIOS 설치 도구 또는 네트워크 어댑터의 구성 설치 도구를 사용하여 PXE를 사용할 수 있도록 합니다.
- 5 (선택 사항) DVD 또는 CD에서 부트하는 경우 시스템 BIOS의 부트 설정을 변경하여 DVD 또는 CD 매체에서 부트되도록 설정합니다. 지침은 하드웨어 설명서를 참조하십시오.
- 6 시스템이 꺼져 있는 경우 시스템을 켭니다. 시스템이 켜져 있는 경우 시스템을 재부트합니다. GRUB 메뉴가 표시됩니다. 이 메뉴에는 부트 항목 목록이 표시됩니다. 표시되는 항목은 설치할 Solaris 인스턴스입니다.

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
|Solaris 10 image_directory          |
|                                     |
+-----+
```

Use the ^ and v keys to select which entry is highlighted. Press enter to boot the selected OS, 'e' to edit the commands before booting, or 'c' for a command-line.

`image_directory`는 설치 이미지가 있는 디렉토리의 이름입니다. JumpStart 파일 경로는 `add_install_client` 명령어와 `-c` 옵션으로 정의됩니다.

주 - GRUB 항목에서 지금 부트하는 대신 부트 항목을 편집할 수 있습니다. GRUB 항목을 편집한 다음 JumpStart 설치를 수행합니다. GRUB 항목을 편집하는 방법과 설치 옵션 목록을 보려면 131 페이지 “x86: GRUB 부트 명령어를 편집하여 사용자 정의 JumpStart 설치 수행”을 참조하십시오.

7 Solaris 부트 화면에서 시스템을 부트할 장치를 선택합니다. DVD, CD, 벡 또는 디스크를 선택합니다.

8 프롬프트에서 다음 지침 중 하나를 수행합니다.

Select the type of installation you want to perform:

- 1 Solaris Interactive
- 2 Custom JumpStart
- 3 Solaris Interactive Text (Desktop session)
- 4 Solaris Interactive Text (Console session)
- 5. Apply driver updates
- 6. Single User Shell

Enter the number of your choice.

Please make a selection (1-6).

사용자 정의 JumpStart 방법을 선택하려면 **2**를 입력하고 Enter 키를 누릅니다.

JumpStart 설치가 시작됩니다.

주 -

- 30초 내에 선택하지 않으면 Solaris 대화식 설치 프로그램이 시작됩니다. 명령줄에서 아무 키나 입력하여 타이머를 중지할 수 있습니다.
- 항목 1, 3 또는 4를 선택하면 대화식 설치로 설치합니다. 대화식 설치에 대한 자세한 내용은 **Solaris 10 6/06 설치 설명서: 기본 설치를 참조하십시오.**
- 이 설치에 대한 자세한 내용은 **Solaris 10 6/06 설치 설명서: 기본 설치를 참조하십시오.**
 - 항목 5를 선택하면 설치 드라이버가 업데이트됩니다.
 - 항목 6을 선택하면 관리 작업을 수행할 수 있습니다.

9 sysidcfg 파일에서 시스템 정보를 미리 구성하지 않은 경우, 메시지가 나타났을 때 시스템 구성에 대한 질문에 답합니다.

10 화면에 나타난 지침에 따라 소프트웨어를 설치합니다.

JumpStart 프로그램이 Solaris 소프트웨어 설치를 완료하면 시스템이 자동으로 재부트됩니다. 또한 GRUB menu.lst 파일이 자동으로 업데이트됩니다. 그리고 설치한 Solaris 인스턴스가 다음 GRUB 메뉴를 사용할 때 나타납니다.

설치가 끝나면 설치 로그가 파일에 저장됩니다. 설치 로그는 다음 디렉토리에서 찾을 수 있습니다.

- /var/sadm/system/logs
- /var/sadm/install/logs

x86: GRUB 부트 명령어를 편집하여 사용자 정의 JumpStart 설치 수행

디버깅 용도 등과 같은 일부 상황에서는 GRUB 부트 명령어를 수정할 수 있습니다. 다음 절차에서는 사용자 정의 JumpStart 설치를 수행하기 전에 GRUB 부트 명령어를 편집하는 단계를 설명합니다.

▼ x86: GRUB 부트 명령어를 수정하려면

- 1 설치를 시작하려면 앞의 129 페이지 “x86: 사용자 정의 JumpStart 프로그램 및 GRUB를 사용하여 설치 또는 업그레이드 수행” 절차에 있는 단계 1부터 단계 5를 진행합니다.
- 2 시스템이 꺼져 있는 경우 시스템을 켭니다. 시스템이 켜져 있는 경우 시스템을 재부트합니다. GRUB 메뉴가 표시됩니다. 이 메뉴에는 부트 항목 목록이 표시됩니다. 표시되는 항목은 설치할 Solaris 인스턴스입니다.

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
```

```
+-----+
|Solaris 10 image_directory|
|                           |
+-----+

```

Use the ^ and v keys to select which entry is highlighted. Press enter to boot the selected OS, 'e' to edit the commands before booting, or 'c' for a command-line.

*image_directory*는 설치 이미지가 있는 디렉토리의 이름입니다.

주 -

- NFS를 사용하여 `add_install_client` 명령어와 `-c` 옵션으로 JumpStart 디렉토리 경로를 설정한 경우에는 부트 항목에 경로를 포함할 필요가 없습니다.
- NFS를 사용하지 않는 경우에는 JumpStart 디렉토리가 포함된 압축 구성 파일의 경로를 기록해 두어야 합니다.

- 3 부트 프로세스를 중지하고 메뉴 항목 편집기를 사용하려면 `e`를 입력합니다. GRUB 편집 메뉴가 표시됩니다.

```
kernel /I86PC.Solaris_11-8/multiboot kernel/unix -B console=ttyb,\
install_media=131.141.2.32:/export/mary/v11 \
module /I86PC.Solaris_11-8/x86.new
```

4 화살표 키를 사용하여 부트 항목을 선택합니다.

5 선택된 명령어를 편집하려면 **e**를 입력합니다.

다음 예와 비슷한 명어가 표시됩니다.

```
grub edit>kernel /I86PC.Solaris_11-8/multiboot kernel/unix -B \
console=ttyb,install_media=131.141.2.32:/export/mary/_\
module /I86PC.Solaris_11-8/x86.new
```

6 필요한 옵션을 입력하여 명령어를 편집합니다.

JumpStart 설치 구문은 다음과 같습니다.

```
grub edit>kernel /I86PC.Solaris_11-image_directory/multiboot kernel/unix/ \
install [url|ask] options -B install_media=media_type
```

JumpStart 옵션에 대한 설명은 132 페이지 “x86: 시스템 부트에 대한 명령어 참조”를 참조하십시오.

다음 예에서는 사용자 정의 JumpStart 프로필을 사용하여 네트워크에 OS를 설치합니다.

```
kernel /I86PC.Solaris_11-8/multiboot kernel/unix/ install \
-B install_media=131.141.2.32:/export/mary/v11 \
module /I86PC.Solaris_11-8/x86.new
```

7 편집 결과를 사용하려면 **Enter** 키를 누릅니다.

변경 내용이 저장되고 GRUB 주 메뉴가 표시됩니다.

주 - Esc 키를 누르면 변경 내용을 저장하지 않고 GRUB 주 메뉴로 돌아갑니다.

8 설치를 시작하려면 **b**를 입력합니다.

x86: 시스템 부트에 대한 명령어 참조

다음 표에서는 GRUB 메뉴 부트 명령어에 대한 명령줄 옵션을 설명합니다. 나열된 옵션은 JumpStart 설치에 적합합니다.

부트 명령어의 구문은 다음과 같습니다.

```
kernel /I86PC.Solaris_11-image_directory/multiboot kernel/unix/ install \
[url|ask] options -B install_media=media_type
```

표 9-3 GRUB 메뉴 부트 명령어 참조

옵션	설명
install	<p>사용자 정의 JumpStart 설치를 수행합니다.</p> <p>다음 예에서는 다음 옵션을 사용하여 DVD 매체에서 시스템을 부트합니다.</p> <ul style="list-style-type: none"> ■ <code>install</code>은 사용자 정의 JumpStart 설치를 수행합니다. ■ <code>file://jumpstart/config.tar</code>은 로컬 디스크에서 JumpStart 프로필을 찾습니다. <pre>kernel /I86pc.Solaris_11.8/multiboot install file://jumpstart/config.tar \ -B install_media=dvdrom module /I86Solaris_11.8/x86.new</pre>
[url ask]	<p>사용자 정의 JumpStart 파일의 위치를 지정하거나 위치를 지정하라는 프롬프트를 표시합니다.</p> <ul style="list-style-type: none"> ■ <code>url</code> - 파일 경로를 지정합니다. HTTP 또는 HTTPS 서버에 있는 파일에 URL을 지정할 수 있습니다. HTTP 서버의 구문은 다음과 같습니다. <code>http://server_name:IP_address/jumpstart_dir_path/ compressed_config_file&proxy_info</code> ■ <code>sysidcfg</code> 파일을 압축된 구성 파일에 둔 경우, 다음 예에서와 같이 해당 파일을 포함한 서버의 IP 주소를 지정해야 합니다. <pre>kernel /I86pc.Solaris_11.8/multiboot install \ http://192.168.2.1/jumpstart/config.tar \ -B install_media=192.168.2.1/export/Solaris_11.8/boot \ module /I86PC.Solaris_11.8/x86.new</pre> ■ 방화벽 뒤에 있는 HTTP 서버에 압축된 구성 파일을 저장했다면 부트 동안 프록시 지정자를 사용해야 합니다. 파일을 포함하는 서버에 대하여 IP 주소를 지정할 필요가 없습니다. 다음 예에서와 같이 프록시 서버에 대하여 IP 주소를 지정해야 합니다. <pre>kernel /I86pc.Solaris_11.8/multiboot install \ http://www.shadow.com/jumpstart/config.tar&proxy=131.141.6.151 \ -B install_media=192.168.2.1/export/Solaris_11.8/boot \ module /I86PC.Solaris_11.8/x86.new</pre> ■ <code>ask</code> - 압축 구성 파일의 위치를 입력하라는 메시지를 표시하도록 설치 프로그램을 지정합니다. 시스템이 부트되어 네트워크에 연결되면 프롬프트 메시지가 표시됩니다. 이 옵션을 사용하면 완전한 핸드 오프 JumpStart 설치 방식으로 설치할 수 없습니다. Return 버튼을 눌러 프롬프트를 우회하면 Solaris 설치 프로그램은 네트워크 매개 변수를 대화식으로 구성합니다. 그러면 설치 프로그램이 압축된 구성 파일의 위치를 묻는 메시지를 표시합니다. 다음 예에서는 사용자 정의 JumpStart 설치를 수행하고 DVD 매체에서 부트합니다. 시스템이 네트워크에 연결되면 구성 파일의 위치를 입력하라는 메시지가 표시됩니다. <pre>kernal /boot/multiboot kernel/unix install ask -B \ install_media=192.168.2.1:export/sol_11_x86/boot module \ /I86PC.Solaris_11.8_</pre>

표 9-3 GRUB 메뉴 부트 명령어 참조 (계속)

옵션	설명
<i>options</i>	<ul style="list-style-type: none"> <li data-bbox="313 234 1303 390">■ <code>dhcp</code> - 시스템을 부트하는 데 필요한 네트워크 설치 정보를 얻을 때 DHCP 서버를 사용하도록 지정합니다. 이 옵션은 JumpStart 설치에 필요하지 않습니다. <code>dhcp</code>를 입력하여 DHCP 서버를 사용하도록 지정하지 않은 경우 시스템은 <code>/etc/bootparams</code> 파일이나 <code>bootparams</code> 이름 서비스 데이터베이스를 사용합니다. 예를 들어 정적 IP 주소를 유지하고자 하는 경우 <code>dhcp</code>를 지정하지 않을 것입니다. 예를 들면 다음과 같습니다. <pre data-bbox="356 407 982 494">kernel /I86pc.Solaris_11.8/multiboot install \ dhcp -B install_media=192.168.2.1:/export/Solaris_11.8/ \ boot module /I86PC.Solaris_11.8/x86.new</pre> <li data-bbox="313 503 1303 600">■ <code>nowin</code>과 <code>text</code> 옵션은 JumpStart 설치에 적용되지 않습니다. 이러한 옵션은 대화식 설치에 유용합니다. 자세한 내용은 Solaris 10 6/06 설치 설명서: 기본 설치의 “GRUB를 이용한 Solaris 설치 프로그램을 사용하여 설치 또는 업그레이드”를 참조하십시오.

사용자 정의 JumpStart로 설치(예)

이 장에서는 사용자 정의 JumpStart 설치를 사용하여 SPARC 기반 및 x86 기반의 시스템에서 Solaris 소프트웨어를 설정 및 설치하는 방법에 대해 설명합니다.

- 135 페이지 “예제 사이트 설정”
- 136 페이지 “설치 서버 만들기”
- 138 페이지 “x86: 마케팅 시스템용 부트 서버 만들기”
- 139 페이지 “JumpStart 디렉토리 만들기”
- 139 페이지 “JumpStart 디렉토리 공유”
- 139 페이지 “SPARC: 엔지니어링 그룹의 프로필 작성”
- 140 페이지 “x86: 마케팅 그룹의 프로필 작성”
- 140 페이지 “rules 파일 업데이트”
- 141 페이지 “rules 파일 검증”
- 141 페이지 “SPARC: 네트워크에서 설치할 엔지니어링 시스템 설정”
- 142 페이지 “x86: 네트워크에서 설치할 마케팅 시스템 설정”
- 143 페이지 “SPARC: 엔지니어링 시스템 부트 및 Solaris 소프트웨어 설치”
- 143 페이지 “x86: 마케팅 시스템 부트 및 Solaris 소프트웨어 설치”

예제 사이트 설정

그림 10-1은 이 예의 사이트 설정입니다.

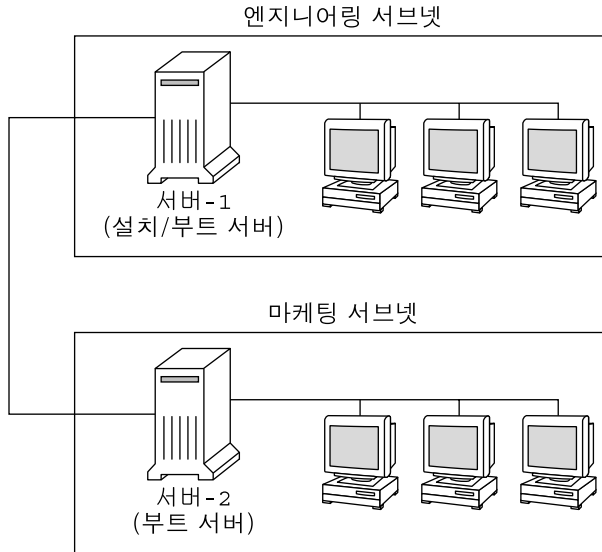


그림 10-1 예제 사이트 설정

이 예제 사이트에서 조건은 다음과 같습니다.

- SPARC: 엔지니어링 그룹은 자체 서브넷에 위치합니다. 이 그룹은 소프트웨어 개발용 SPARCstation™ 시스템을 사용합니다.
- x86: 마케팅 그룹은 자체 서브넷에 위치합니다. 이 그룹은 워드프로세서, 스프레드시트 및 기타 사무용 생산관리 도구 실행을 위해 x86 기반 시스템을 사용합니다.
- 이 사이트에서는 NIS를 사용합니다. 시스템의 이더넷 주소, IP 주소 및 호스트 이름은 NIS 맵에서 미리 구성됩니다. 서브넷 마스크, 날짜 및 시간, 사이트의 지리적 지역도 NIS 맵에서 미리 구성됩니다.

주 - 마케팅 시스템용 주변 기기는 sysidcfg 파일에서 미리 구성됩니다.

- 엔지니어링 시스템과 마케팅 시스템은 모두 네트워크에서 Solaris 10 6/06 소프트웨어로 설치됩니다.

설치 서버 만들기

그룹이 네트워크에서 Solaris 10 6/06 소프트웨어를 설치해야 하기 때문에 server-1을 두 그룹 모두에 대한 설치 서버로 만듭니다. `setup_install_server(1M)` 명령어를 사용하여 이미지를 server-1 로컬 디스크(/export/install 디렉토리)로 복사합니다. 다음 매체 중 하나에서 이미지를 복사합니다.

- Solaris Software CD와 Solaris Languages CD
- Solaris Operating System DVD

디스크에서 빈 디렉토리로 이미지를 복사해야 합니다. 이 예에서는 `sparc_10_606` 디렉토리 및 `x86_10_606` 디렉토리입니다.

예 10-1 SPARC: Solaris 10 6/06 CD 복사

server-1에 연결된 CD-ROM 드라이브에 Solaris Software for SPARC Platforms - 1 CD를 삽입하고 다음 명령어를 입력합니다.

```
server-1# mkdir -p /export/install/sparc_10_606
server-1# cd /CD_mount_point/Solaris_10_606/Tools
server-1# ./setup_install_server /export/install/sparc_10_606
```

server-1에 연결된 CD-ROM 드라이브에 Solaris Software for SPARC Platforms - 2 CD를 삽입하고 다음 명령어를 입력합니다.

```
server-1# cd /CD_mount_point/Solaris_10_606/Tools
server-1# ./add_to_install_server /export/install/sparc_10_606
```

설치하려는 각 Solaris Software에 대하여 이전 명령을 반복합니다.

server-1에 연결된 CD-ROM 드라이브에 SPARC: Solaris Languages for SPARC Platforms CD를 삽입하고 다음 명령어를 입력합니다.

```
server-1# cd /CD_mount_point/Solaris_10_606/Tools
server-1# ./add_to_install_server /export/install/sparc_10_606
```

예 10-2 x86: Solaris 10 6/06 CD 복사

server-1에 연결된 CD-ROM 드라이브에 Solaris Software for x86 Platforms - 1 CD를 삽입하고 다음 명령어를 입력합니다.

```
server-1# mkdir -p /export/install/x86_10_606
server-1# cd /CD_mount_point/Solaris_10_606/Tools
server-1# ./setup_install_server /export/install/x86_10_606
```

server-1에 연결된 CD-ROM 드라이브에 Solaris Software for x86 Platforms - 2 CD를 삽입하고 다음 명령어를 입력합니다.

```
server-1# cd /CD_mount_point/Solaris_10_606/Tools
server-1# ./add_to_install_server /export/install/x86_10_606
```

설치하려는 각 Solaris Software에 대하여 이전 명령어를 반복합니다.

server-1에 연결된 CD-ROM 드라이브에 Solaris Languages for x86 Platforms CD를 삽입하고 다음 명령어를 입력합니다.

```
server-1# cd /CD_mount_point/Solaris_10_606/Tools
server-1# ./add_to_install_server /export/install/x86_10_606
```

예 10-3 SPARC: Solaris 10 6/06 DVD 복사

server-1에 부착된 CD-ROM 드라이브에 Solaris Operating System for SPARC Platforms DVD를 삽입하고 다음 명령을 입력합니다.

```
server-1# mkdir -p /export/install/sparc_10_606
server-1# cd /DVD_mount_point/Solaris_10_606/Tools
server-1# ./setup_install_server /export/install/sparc_10_606
```

예 10-4 x86: Solaris Operating System for x86 Platforms DVD 복사

server-1에 연결된 CD-ROM 드라이브에 Solaris Operating System for x86 Platforms DVD를 삽입하고 다음 명령을 입력합니다.

```
server-1# mkdir -p /export/install/x86_10_606
server-1# cd /DVD_mount_point/Solaris_10_606/Tools
server-1# ./setup_install_server /export/install/x86_10_606
```

x86: 마케팅 시스템용 부트 서버 만들기

시스템이 다른 서브넷의 설치 서버에서 부트할 수 없으므로 마케팅 그룹의 서브넷에 있는 server-2를 부트 서버로 만듭니다. setup_install_server(1M) 명령어를 사용하여 Solaris Operating System for x86 Platforms DVD 또는 Solaris Software for x86 Platforms - 1 CD에서 부트 소프트웨어를 복사합니다. /export/boot 디렉토리의 server-2 로컬 디스크로 부트 소프트웨어를 복사합니다.

매체를 선택하고 부트 소프트웨어를 로컬 디스크에 설치합니다.

- Solaris Software for x86 Platforms - 1 CD를 server-2에 연결된 CD-ROM 드라이브에 삽입한 경우 다음 명령을 입력합니다.

```
server-2# cd /CD_mount_point/Solaris_10_606/Tools
server-2# ./setup_install_server -b /export/boot
```

- server-2에 연결된 DVD-ROM 드라이브에 Solaris Operating System for x86 Platforms DVD를 삽입한 경우 다음 명령을 입력합니다.

```
server-2# cd /DVD_mount_point/Solaris_10_606/Tools
server-2# ./setup_install_server -b /export/boot
```

setup_install_server 명령어에서 -b는 setup_install_server가 이름이 /export/boot인 디렉토리로 부트 정보를 복사할 것을 지정합니다.

JumpStart 디렉토리 만들기

이제 설치 서버와 부트 서버를 설정했으면 server-1에 JumpStart 디렉토리를 만듭니다. 네트워크의 모든 시스템을 사용할 수 있습니다. 이 디렉토리에는 Solaris 소프트웨어의 사용자 정의 JumpStart 설치에 필요한 파일이 들어 있습니다. 다음과 같이 /export/install로 복사된 Solaris Operating System DVD 이미지 또는 Solaris Software - 1 CD 이미지에서 샘플 디렉토리를 복사하여 이 디렉토리를 설정합니다.

```
server-1# mkdir /jumpstart
server-1# cp -r /export/install/sparc_10_606/Solaris_10_606/Misc/jumpstart_sample /jumpstart
```

JumpStart 디렉토리 공유

rules 파일 및 프로필을 네트워크의 시스템에 액세스 가능하게 하려면 /jumpstart 디렉토리를 공유합니다. 디렉토리의 공유를 활성화하려면 다음 행을 /etc/dfs/dfstab 파일에 추가합니다.

```
share -F nfs -o ro,anon=0 /jumpstart
```

그런 다음 명령줄에서 shareall 명령어를 입력합니다.

```
server-1# shareall
```

SPARC: 엔지니어링 그룹의 프로필 작성

엔지니어링 시스템의 경우 /jumpstart 디렉토리에 이름이 eng_prof인 파일을 만듭니다. eng_prof 파일에는 엔지니어링 그룹의 시스템에 설치할 Solaris 10 6/06 소프트웨어를 정의하는 다음 항목이 포함되어 있습니다.

```
install_type  initial_install
system_type   standalone
partitioning  default
cluster       SUNWCprog
filesys       any 512 swap
```

이전 예 프로필은 다음 설치 정보를 지정합니다.

install_type	설치를 업그레이드의 반대인 초기 설치로 처리합니다.
system_type	엔지니어링 시스템은 독립형 시스템입니다.
partitioning	JumpStart 소프트웨어가 엔지니어링 그룹에 Solaris 소프트웨어를 설치하기 위해 기본 디스크 분할을 사용합니다.
cluster	개발자 시스템 지원 소프트웨어 그룹이 설치됩니다.
filesys	엔지니어링 그룹의 각 시스템이 512MB의 스왑 공간을 가집니다.

x86: 마케팅 그룹의 프로필 작성

마케팅 시스템의 경우 /jumpstart 디렉토리에 이름이 marketing_prof인 파일을 만듭니다. marketing_prof 파일에는 마케팅 그룹의 시스템에 설치할 Solaris 10 6/06 소프트웨어를 정의하는 다음 항목이 포함되어 있습니다.

```
install_type  initial_install
system_type   standalone
partitioning  default
cluster       SUNWCuser
package       SUNWaudio
```

이전 예 프로필은 다음 설치 정보를 지정합니다.

```
install_type  설치를 업그레이드의 반대인 초기 설치로 처리합니다.
system_type   마케팅 시스템은 독립형 시스템입니다.
partitioning  JumpStart 소프트웨어가 마케팅 그룹에 Solaris 소프트웨어를 설치하기 위해
              기본 디스크 분할을 사용합니다.
cluster       최종 사용자 Solaris 소프트웨어 그룹이 설치됩니다.
package       오디오 데모 소프트웨어 패키지가 각 시스템에 추가됩니다.
```

rules 파일 업데이트

이제 rules 파일에 규칙을 추가해야 합니다. Solaris 설치 프로그램은 규칙을 사용하여 사용자 정의 JumpStart를 설치하는 동안 각 시스템에 대한 정확한 설치(프로필)를 선택합니다.

이 사이트에서는 각 부서가 자체 서브넷에 위치하며 자체 네트워크 주소를 갖습니다. 엔지니어링 부서는 서브넷 255.222.43.0에 위치합니다. 마케팅 부서는 255.222.44.0에 위치합니다. 이 정보를 통해 Solaris 10 6/06 소프트웨어와 함께 엔지니어링 및 마케팅 시스템을 설치하는 방법을 제어할 수 있습니다. /jumpstart 디렉토리에서 rules 파일을 편집하고 예로 든 모든 규칙을 삭제하고 파일에 다음 행을 추가합니다.

```
network 255.222.43.0 - eng_prof -
network 255.222.44.0 - marketing_prof -
```

기본적으로 이 규칙에서는 255.222.43.0 네트워크에 있는 시스템이 eng_prof 프로필을 사용하여 Solaris 10 6/06 소프트웨어와 함께 설치될 것을 지정합니다. 255.222.44.0 네트워크의 시스템은 marketing_prof 프로필을 사용하여 Solaris 10 6/06 소프트웨어와 함께 설치됩니다.

주- 네트워크 주소를 사용하는 샘플 규칙을 사용하면 각각 `eng_prof` 및 `marketing_prof`를 사용하여 Solaris 10 6/06 소프트웨어와 함께 설치될 시스템을 식별할 수 있습니다. 호스트 이름, 메모리 크기 또는 모델 유형을 규칙 키워드로 사용할 수도 있습니다. 표 11-1에는 `rules` 파일에서 사용할 수 있는 전체 키워드 목록이 포함되어 있습니다.

rules 파일 검증

`rules` 및 프로필이 설정된 후에 `check` 스크립트를 실행하여 파일이 맞는지 검증합니다.

```
server-1# cd /jumpstart
server-1# ./check
```

`check` 스크립트가 오류를 찾아내지 않으면 스크립트는 `rules.ok` 파일을 만듭니다.

SPARC: 네트워크에서 설치할 엔지니어링 시스템 설정

`/jumpstart` 디렉토리 및 파일을 설정한 후 설치 서버 `server-1`에서 `add_install_client` 명령어를 사용하여 설치 서버에서 Solaris 소프트웨어를 설치하도록 엔지니어링 시스템을 설정합니다. `server-1`은 엔지니어링 그룹의 서브넷에 대한 부트 서버이기도 합니다.

```
server-1# cd /export/install/sparc_10_606/Solaris_10_606/Tools
server-1# ./add_install_client -c server-1:/jumpstart host-eng1 sun4u
server-1# ./add_install_client -c server-1:/jumpstart host-eng2 sun4u
```

`add_install_client` 명령어에서 사용되는 옵션은 다음 의미를 갖습니다.

-c JumpStart 디렉토리에 대한 서버(`server-1`)와 경로(`/jumpstart`)를 지정합니다. NFS를 사용하는 경우 이 옵션을 사용합니다.

주-NFS를 사용하지 않는 경우 다음 명령어를 사용하여 JumpStart 디렉토리 경로를 지정합니다.

- SPARC 기반 시스템의 경우 `boot` 명령어에 경로를 지정합니다.
- x86 기반 시스템의 경우, GRUB 메뉴 항목을 편집하여 경로를 지정합니다.

host-eng1	엔지니어링 그룹의 시스템 이름
host-eng2	엔지니어링 그룹의 다른 시스템의 이름
sun4u	<code>server-1</code> 을 설치 서버로 사용하는 시스템의 플랫폼 그룹을 지정합니다. 플랫폼 그룹은 Ultra 5 시스템용입니다.

x86: 네트워크에서 설치할 마케팅 시스템 설정

다음에는 부트 서버(server-2)에서 `add_install_client` 명령어를 사용합니다. 이 명령은 마케팅 시스템이 부트 서버에서 부트되고 설치 서버(server-1)에서 Solaris 소프트웨어가 설치되도록 설정합니다.

```
server-2# cd /marketing/boot-dir/Solaris_10_606/Tools
server-2# ./add_install_client -s server-1:/export/install/x86_10_606 \
-c server-1:/jumpstart host-mkt1 i86pc
server-2# ./add_install_client -s server-1:/export/install/x86_10_606 \
-c server-1:/jumpstart host-mkt2 i86pc
server-2# ./add_install_client -d -s server-1:/export/install/x86_10_606 \
-c server-1:/jumpstart SUNW.i86pc i86pc
server-2# ./add_install_client -c server-1:/jumpstart host-mkt1 sun4u
server-2# ./add_install_client -c server-1:/jumpstart host-mkt2 sun4u
```

`add_install_client` 명령어에서 사용되는 옵션은 다음 의미를 갖습니다.

- d 클라이언트가 DHCP를 사용하여 네트워크 설치 매개 변수를 얻도록 지정합니다. 이 옵션을 사용하려면 네트워크에서 부트하기 위해 PXE 네트워크 부트를 사용하는 클라이언트가 필요합니다. -d는 PXE 네트워크 부트를 사용하지 않는 네트워크 부트 클라이언트에 대한 선택 사항입니다.
- s 설치 서버(server-1) 및 Solaris 소프트웨어의 경로(/export/install/x86_10_606)를 지정합니다.
- c JumpStart 디렉토리에 대한 서버(server-1)와 경로(/jumpstart)를 지정합니다. NFS를 사용하는 경우 이 옵션을 사용합니다.

주 - NFS를 사용하지 않는 경우 다음 명령어를 사용하여 JumpStart 디렉토리 경로를 지정합니다.

- SPARC 기반 시스템의 경우, `boot` 명령어에 경로를 지정합니다.
 - x86 기반 시스템의 경우, GRUB 메뉴 항목을 편집하여 경로를 지정합니다.
-

host-mkt1	마케팅 그룹의 시스템 이름
host-mkt2	마케팅 그룹의 다른 시스템의 이름
sun4u	server-1을 설치 서버로 사용하는 시스템의 플랫폼 그룹을 지정합니다. 플랫폼 그룹은 Ultra 5 시스템용입니다.
SUNW.i86pc	모든 Solaris x86 클라이언트에 대한 DHCP 클래스 이름입니다. 단일 명령어로 Solaris x86 DHCP 클라이언트를 모두 구성하려면 다음 클래스 이름을 사용합니다.
i86pc	이 부트 서버를 사용하는 시스템의 플랫폼 그룹을 지정합니다. 플랫폼 이름은 x86 기반 시스템을 나타냅니다.

SPARC: 엔지니어링 시스템 부트 및 Solaris 소프트웨어 설치

서버와 파일을 설정한 후 각 시스템의 ok(PROM) 프롬프트에서 다음 boot 명령어를 사용하여 엔지니어링 시스템을 부트할 수 있습니다.

```
ok boot net - install
```

Solaris OS는 자동으로 엔지니어링 그룹의 시스템에 설치됩니다.

x86: 마케팅 시스템 부트 및 Solaris 소프트웨어 설치

다음 중 하나에서 시스템을 부트할 수 있습니다.

- Solaris Software for x86 Platforms - 1 CD
- Solaris Operating System for x86 Platforms DVD
- PXE 네트워크 부트를 사용한 네트워크

Solaris 소프트웨어는 자동으로 마케팅 그룹의 시스템에 설치됩니다.

사용자 정의 JumpStart(참조)

이 장에서는 rules 파일, 프로필 및 스크립트의 시작과 끝에서 사용할 수 있는 키워드와 값을 나열합니다.

- 145 페이지 “규칙 키워드 및 값”
- 149 페이지 “프로필 키워드 및 값”
- 187 페이지 “사용자 정의 JumpStart 환경 변수”
- 189 페이지 “검사 키워드 및 값”

규칙 키워드 및 값

표 11-1은 rules 파일에서 사용할 수 있는 키워드와 값에 대해 설명합니다. rules 파일을 만드는 자세한 지침은 78 페이지 “rules 파일 만들기”를 참조하십시오.

표 11-1 규칙 키워드 및 값 설명

키워드	값	의미
any	마이너스 기호(-)	모두. any 키워드가 항상 뒤따릅니다.
arch	processor_type processor_type의 유효한 값은 다음과 같습니다. <ul style="list-style-type: none"> ■ SPARC: sparc ■ x86: i386 	시스템의 프로세서 유형 uname -p 명령어는 시스템의 프로세서 유형을 보고합니다.

표 11-1 규칙 키워드 및 값 설명 (계속)

키워드	값	의미
disksize	<p><i>actual_disk_name size_range</i></p> <p><i>actual_disk_name</i> - c0t3d0이나 c0d0 또는 특수 단어 rootdisk 등 cxytzyz 형식의 디스크 이름. rootdisk를 사용하면 일치하는 디스크는 다음 순서에 따라 결정됩니다.</p> <ul style="list-style-type: none"> ■ SPARC: 출하시 JumpStart가 설치된 새로운 SPARC 기반 시스템인 사전 설치된 부트 이미지를 포함한 디스크. ■ 디스크가 존재하는 경우에는 c0t3d0s0 디스크 ■ 커널 검사 순서에서 검색된 사용 가능한 첫 번째 디스크 <p><i>size_range</i> - MB 범위로 지정해야 하는 디스크의 크기(<i>x-x</i>)</p> <p>주 - <i>size_range</i>를 계산할 때는 1MB가 1,048,576바이트에 해당된다는 점에 주의하십시오. 디스크 용량을 “535Mbyte”로 광고하는 경우에도 이 디스크는 510MB의 디스크 공간만을 포함할 수 있습니다. $535,000,000 / 1,048,576 = 510$이기 때문에 JumpStart 프로그램은 “535-MB” 디스크를 510-MB 디스크로 간주합니다. “535-MB” 디스크는 530-550에 해당하는 <i>size_range</i>에 부합되지 않습니다.</p>	<p>시스템 디스크의 이름 및 크기(MB)</p> <p>예:</p> <p>disksize c0t3d0 250-300</p> <p>이 예에서 JumpStart 프로그램은 c0t3d0이라는 이름의 시스템 디스크와 일치를 시도합니다. 디스크에는 250MB - 300MB의 정보를 담을 수 있습니다.</p> <p>예:</p> <p>disksize rootdisk 750-1000</p> <p>이 예에서, JumpStart 프로그램은 다음 순서로 디스크 일치를 시도합니다.</p> <ol style="list-style-type: none"> 1. 사전 설치된 부트 이미지를 포함하는 시스템 디스크 2. 디스크가 존재하는 경우에는 c0t3d0s0 디스크 3. 750 MB 와 1 GB의 정보 사이에서 유지할 수 있는 첫 번째 사용 가능 디스크
domainname	<i>actual_domain_name</i>	<p>이름 서비스가 정보를 결정하는 방법을 제어하는 시스템의 도메인 이름</p> <p>이미 설치가 되어 있는 시스템이 있는 경우 domainname 명령어가 시스템 도메인 이름을 보고합니다.</p>
hostaddress	<i>actual_IP_address</i>	시스템의 IP 주소
hostname	<i>actual_host_name</i>	<p>시스템의 호스트 이름</p> <p>이미 설치가 되어 있는 시스템이 있다면 uname -n 명령어가 시스템 호스트 이름을 보고합니다.</p>

표 11-1 규칙 키워드 및 값 설명 (계속)

키워드	값	의미
installed	<p><i>slice version</i></p> <p><i>slice</i> - <code>c0t3d0s5</code>와 같은 <code>cwtx dysz</code> 형식의 디스크 슬라이스 이름이나 <i>any</i> 또는 <i>rootdisk</i>와 같은 특수 단어입니다. <i>any</i>를 사용하면 JumpStart 프로그램이 시스템의 모든 디스크를 커널 검사 순서에 따라 일치 시도합니다. <i>rootdisk</i>를 사용하면 다음 순서에 따라 일치하는 디스크가 결정됩니다.</p> <ul style="list-style-type: none"> ■ SPARC: 출하시 JumpStart가 설치된 새로운 SPARC 기반 시스템인 사전 설치된 부트 이미지를 포함한 디스크. ■ 디스크가 존재하는 경우에는 <code>c0t3d0s0</code> 디스크 ■ 커널 검사 순서에서 검색된 사용 가능한 첫 번째 디스크 <p><i>version</i> - 버전 이름 또는 특수 단어 <i>any</i> 또는 <i>upgrade</i>. <i>any</i>를 사용하면 모든 Solaris 또는 SunOS 릴리스가 일치됩니다. <i>upgrade</i>를 사용하면 지원되며 업그레이드할 수 있는 모든 Solaris 릴리스가 일치됩니다.</p> <p>JumpStart 프로그램이 Solaris 릴리스를 찾지만 버전을 파악할 수 없는 경우, 반환되는 버전은 SystemV입니다.</p>	<p>특정 버전의 Solaris 소프트웨어에 해당하는 루트(/) 파일 시스템</p> <p>예:</p> <pre>installed c0t3d0s1 Solaris 10_606</pre> <p>예에서 JumpStart 프로그램은 <code>c0t3d0s1</code>에서 Solaris 루트(/) 파일 시스템이 있는 시스템과 일치 시도를 시도합니다.</p>
karch	<p><i>actual_platform_group</i></p> <p>유효한 값은 <code>sun4u</code>, <code>i86pc</code> 및 <code>prep</code>입니다. 시스템 및 해당하는 플랫폼 그룹의 목록은 http://docs.sun.com의 Solaris Sun 하드웨어 플랫폼 안내서에서 제시합니다.</p>	<p>시스템의 플랫폼 그룹</p> <p>이미 설치된 시스템을 갖고 있다면 <code>arch -k</code> 명령어나 <code>uname -m</code> 명령어가 시스템의 플랫폼 그룹을 보고합니다.</p>
memsize	<p><i>physical_mem</i></p> <p>값은 MB 단위로 <code>x-x</code> 형식의 범위이거나 단일 MB 단위 값이어야 합니다.</p>	<p>시스템의 물리적 메모리 크기(MB)</p> <p>예:</p> <pre>memsize 64-128</pre> <p>이 예에서는 물리적 메모리 크기가 64MB - 128MB 사이인 시스템 일치 시도를 시도합니다.</p> <p>이미 설치된 시스템이 있는 경우 <code>prtconf</code> 명령어, 행 2의 출력은 시스템의 물리적 메모리 크기를 보고합니다.</p>

표 11-1 규칙 키워드 및 값 설명 (계속)

키워드	값	의미
model	<i>actual_platform_name</i>	<p>시스템의 플랫폼 이름. 유효한 플랫폼 이름 목록은 http://docs.sun.com에서 Solaris Sun 하드웨어 플랫폼 안내서를 참조하십시오.</p> <p>설치된 시스템의 플랫폼 이름을 찾으려면 <code>uname -i</code> 명령어나 <code>prtconf</code> 명령어, 행 5의 출력을 사용합니다.</p> <p>주 - <i>actual_platform_name</i>에 공백이 포함되어 있으면 공백을 밑줄(_)로 바꿔야 합니다.</p> <p>예:</p> <p>SUNW,Sun_4_50</p>
network	<i>network_num</i>	<p>시스템의 IP 주소와 서브넷 마스크 사이에 논리적 AND를 수행함으로써 결정되는 시스템의 네트워크 번호</p> <p>예:</p> <p>network 192.168.2.0</p> <p>이 예는 서브넷 마스크가 255.255.255.0인 경우 IP 주소가 192.168.2.8인 시스템과 일치를 시도합니다.</p>
osname	Solaris_x	<p>시스템에 이미 설치되어 있는 Solaris 소프트웨어의 버전</p> <p>예:</p> <p>osname Solaris 10_606</p> <p>이 예에서 JumpStart 프로그램은 이미 Solaris 10 6/06 OS가 설치되어 있는 시스템과 일치를 시도합니다.</p>

표 11-1 규칙 키워드 및 값 설명 (계속)

키워드	값	의미
probe	probe_keyword	올바른 검사 키워드 또는 올바른 사용자 정의 검사 키워드 예: probe disks 이 예는 SPARC 시스템에서 MB 단위 및 커널 검사 순서로 시스템 디스크의 크기(예: c0t3d0s1, c0t4d0s0)를 반환합니다. JumpStart 프로그램은 SI_DISKLIST, SI_DISKSIZE, SI_NUMDISKS 및 SI_TOTALDISK 환경 변수를 설정합니다. 주 - probe 키워드는 속성 일치나 프로필 실행을 시도하지 않는다는 점에서 구별됩니다. probe 키워드는 값을 반환합니다. 결과적으로 probe 규칙 키워드를 사용하여 시작 스크립트, 프로필 및 종료 스크립트를 지정할 수 없습니다. 검사 키워드는 8장에서 설명합니다.
totaldisk	size_range 값은 MB 범위(x-x)로 지정해야 합니다. 주 - size_range를 계산할 때는 1MB가 1,048,576바이트에 해당된다는 점에 주의하십시오. 디스크 용량을 535MB로 광고하는 경우에도 이 디스크는 510MB의 디스크 공간만을 포함할 수 있습니다. 535,000,000 / 1,048,576 = 510이기 때문에 JumpStart 프로그램은 “535-MB” 디스크를 510-MB 디스크로 간주합니다. “535-MB” 디스크는 530-550에 해당하는 size_range에 부합되지 않습니다.	시스템의 전체 디스크 공간(MB). 전체 디스크 공간에는 시스템에 연결된 작동 가능한 디스크 모두가 포함됩니다. 예: totaldisk 300-500 이 예에서 JumpStart 프로그램은 전체 디스크 공간이 300MB - 500MB 사이인 시스템과 일치할 시도합니다.

프로필 키워드 및 값

여기에서는 프로필에서 사용할 수 있는 프로필 키워드 및 값에 대해 설명합니다. 프로필 작성에 대한 자세한 지침은 81 페이지 “프로필 작성”을 참조하십시오.

프로필 키워드 빠른 참조

표 11-2는 설치 시나리오를 기준으로 사용 가능한 키워드를 결정하는 빠른 방법을 제시합니다. 키워드 설명에서 언급되지 않았다면 해당 키워드는 초기 설치 옵션에서만 사용할 수 있습니다.

표 11-2 프로필 키워드 개요

프로필 키워드	설치 상황				
	독립 실행형 시스템(네트워크에 연결되지 않음)	독립 실행형 시스템(네트워크에 연결되어 있음) 또는 서버	OS 서버	업그레이드	디스크 공간 재할당을 통한 업그레이드
archive_location(Solaris Flash 아카이브 설치)	X	X			
backup_media					X
boot_device	X	X	X		
bootenv createbe	X	X	X		
client_arch			X		
client_root			X		
client_swap			X		
cluster(소프트웨어 그룹 추가)	X	X	X		
cluster(클러스터 추가 또는 삭제)	X	X	X	X	X
dontuse	X	X	X		
fdisk(x86 전용)	X	X	X		
filesystem(원격 파일 시스템 마운트)		X	X		
filesystem(로컬 파일 시스템 만들기)	X	X	X		
filesystem(미러된 파일 시스템 만들기)	X	X	X		
forced_deployment(Solaris Flash 차등 아카이브 설치)	X	X			
geo	X	X	X	X	X
install_type	X	X	X	X	X
layout_constraint					X
local_customization(Solaris Flash 아카이브 설치)	X	X			
locale	X	X	X	X	X

표 11-2 프로필 키워드 개요 (계속)

프로필 키워드	설치 상황				
	독립 실행형 시스템(네트워크에 연결되지 않음)	독립 실행형 시스템(네트워크에 연결되어 있음) 또는 서버	OS 서버	업그레이드	디스크 공간 재할당을 통한 업그레이드
metadb(상태 데이터베이스 복제본 만들기)	X	X	X		
no_master_check(Solaris Flash 차등 아카이브 설치)	X	X			
no_content_check(Solaris Flash 차등 아카이브 설치)	X	X			
num_clients			X		
package	X	X	X	X	X
partitioning	X	X	X		
patch	X	X	X	X	X
root_device	X	X	X	X	X
system_type	X	X	X		
usedisk	X	X	X		

프로필 키워드에 대한 설명 및 예

archive_location 키워드

archive_location retrieval_type location

retrieval_type *retrieval_type* 및 *location*의 값은 Solaris Flash 아카이브 저장 위치에 따라 달라집니다. 다음 절에는 *retrieval_type* 및 *location*용으로 사용할 수 있는 값과 archive_location 키워드 사용 방법에 대한 예가 포함되어 있습니다.

- 152 페이지 “NFS 서버에 저장된 아카이브”
- 152 페이지 “HTTP 또는 HTTPS 서버에 저장된 아카이브”
- 154 페이지 “FTP 서버에 저장된 아카이브”
- 155 페이지 “로컬 테이프에 저장된 아카이브”
- 155 페이지 “로컬 장치에 저장된 아카이브”
- 156 페이지 “로컬 파일에 저장된 아카이브”

location 위치에 관한 세부사항은 다음 부문에 있습니다.



주의 - 비전역 영역을 설치하는 경우 Solaris Flash 아카이브를 제대로 만들 수 없습니다. Solaris Flash 기능은 Solaris 영역 분할 기술과 호환되지 않습니다. Solaris Flash 아카이브를 만들 때 다음 조건 하에서 아카이브를 배포하면 결과 아카이브가 제대로 설치되지 않습니다.

- 아카이브를 비전역 영역에 만든 경우
- 아카이브를 비전역 영역이 설치된 전역 영역에 만든 경우

NFS 서버에 저장된 아카이브

아카이브가 NFS 서버에 저장되어 있으면 `archive_location` 키워드용으로 다음 구문을 사용합니다.

```
archive_location nfs server_name:/path/filename retry n
```

server_name 아카이브를 저장한 서버의 이름

path 지정된 서버에서 검색할 아카이브 위치. 경로에 \$HOST가 포함되어 있으면 Solaris Flash 설치 유틸리티는 \$HOST를 설치할 복제 시스템의 이름으로 대체합니다.

filename Solaris Flash 아카이브 파일의 이름

retry n 옵션 키워드. *n*은 Solaris Flash 유틸리티가 아카이브 마운트를 시도하는 최대 횟수입니다.

예 11-1 NFS 서버에 저장된 아카이브

```
archive_location nfs golden:/archives/usrarchive
```

```
archive_location nfs://golden/archives/usrarchive
```

HTTP 또는 HTTPS 서버에 저장된 아카이브

아카이브가 HTTP 서버에 저장되어 있으면 `archive_location` 키워드용으로 다음 구문을 사용합니다.

```
archive_location http://server_name:port/path/filename optional_keyword
```

아카이브가 HTTPS 서버에 저장되어 있으면 `archive_location` 키워드용으로 다음 구문을 사용합니다.

```
archive_location https://server_name:port/path/filename optional_keywords
```

server_name 아카이브를 저장한 서버의 이름

port 옵션 포트. *port*는 포트 번호 또는 런타임에 결정된 포트 번호를 가진 TCP 서비스의 이름일 수 있습니다.

	포트를 지정하지 않으면 Solaris Flash 설치 유틸리티는 기본 HTTP 포트 번호 80을 사용합니다.
<i>path</i>	지정된 서버에서 검색할 아카이브 위치. 경로에 \$HOST가 포함되어 있으면 Solaris Flash 설치 유틸리티는 \$HOST를 설치할 복제 시스템의 이름으로 대체합니다.
<i>filename</i>	Solaris Flash 아카이브 파일의 이름
<i>optional_keywords</i>	HTTP 서버에서 Solaris Flash 아카이브를 검색할 때 지정할 수 있는 옵션 키워드

표 11-3 archive_location ftp와 함께 사용할 수 있는 옵션 키워드

키워드	값 정의
<i>auth basic user_name password</i>	<p>아카이브가 비밀번호를 통해 보호되고 있는 HTTP 서버상에 위치한다면 프로필 파일에서 HTTP 서버에 액세스할 때 필요한 사용자 이름과 비밀번호를 포함시켜야 합니다.</p> <p>주 - 사용자 정의 JumpStart와 함께 사용하도록 만든 프로필에서 이 인증 방법을 사용하는 것은 위험합니다. 인증되지 않은 사용자는 비밀번호를 포함하고 있는 프로필 파일에 액세스하게 됩니다.</p>
<i>timeout min</i>	<p><i>timeout</i> 키워드를 사용하여 HTTP 서버에서 데이터를 받지 않고 전달할 수 있는 최고 시간 길이(분 단위)를 지정할 수 있습니다. 시간 초과가 발생할 경우 연결이 종료되고 다시 열리고 재시작됩니다. <i>timeout</i> 값을 0으로 지정하면 연결이 다시 열리지 않습니다.</p> <ul style="list-style-type: none"> ■ 시간 초과로 인해 다시 연결을 시도하면 Solaris Flash 설치 유틸리티는 가장 최근에 알려진 아카이브 위치에서 설치 재개를 시도합니다. Solaris Flash 설치 유틸리티가 마지막으로 알려진 위치에서 설치를 재개할 수 없다면 아카이브의 시작부터 재검색이 다시 시작되고 시간 초과 이전에 검색된 데이터는 폐기됩니다. ■ 패키지가 설치되는 동안 시간 초과 재연결이 발생하면 시간 초과가 무시되기 전에 검색된 패키지와 데이터의 시작부터 패키지가 재시도됩니다.
<i>proxy host:port</i>	<p><i>proxy</i> 키워드는 프록시 호스트 및 프록시 포트를 지정할 수 있게 합니다. 방화벽의 맞은 편에서 Solaris Flash 아카이브를 검색하기 위해 프록시 호스트를 사용할 수 있습니다. <i>proxy</i> 키워드를 지정할 때는 반드시 프록시 포트를 제공해야 합니다.</p>

예 11-2 HTTP 또는 HTTPS 서버에 저장된 아카이브

```
archive_location http://silver/archives/usrarchive.flar timeout 5
```

auth basic user_name password 키워드의 예는 다음과 같습니다.

```
archive_location http://silver/archives/usrarchive.flar timeout 5 user1 secret
```

FTP 서버에 저장된 아카이브

아카이브가 FTP 서버에 저장되어 있으면 `archive_location` 키워드용으로 다음 구문을 사용합니다.

<code>archive_location</code>	<code>ftp://user_name:password@server_name:port/path/filename optional_keyword</code>
<code>user_name:password</code>	프로필 파일에서 FTP 서버에 액세스하기 위해 필요한 사용자 이름과 비밀번호
<code>server_name</code>	아카이브를 저장한 서버의 이름
<code>port</code>	옵션 포트입니다. <code>port</code> 는 포트 번호 또는 런타임에 결정된 포트 번호를 가진 TCP 서비스의 이름일 수 있습니다. 포트를 지정하지 않으면 Solaris Flash 설치 유틸리티는 기본 FTP 포트 번호 21을 사용합니다.
<code>path</code>	지정된 서버에서 검색할 아카이브 위치. 경로에 <code>\$HOST</code> 가 포함되어 있으면 Solaris Flash 설치 유틸리티는 <code>\$HOST</code> 를 설치할 복제 시스템의 이름으로 대체합니다.
<code>filename</code>	Solaris Flash 아카이브 파일의 이름
<code>optional_keywords</code>	FTP 서버에서 Solaris Flash 아카이브를 검색할 때 지정할 수 있는 옵션 키워드

표 11-4 `archive_location` FTP와 함께 사용할 수 있는 옵션 키워드

키워드	값 정의
<code>timeout min</code>	<p><code>timeout</code> 키워드를 사용하여 HTTP 서버에서 데이터를 받지 않고 전달할 수 있는 최고 시간 길이(분 단위)를 지정할 수 있습니다. 시간 초과가 발생할 경우 연결이 종료되고 다시 열리고 재시작됩니다. <code>timeout</code> 값을 0으로 지정하면 연결이 다시 열리지 않습니다.</p> <ul style="list-style-type: none"> 시간 초과로 인해 다시 연결을 시도하면 Solaris Flash 설치 유틸리티는 가장 최근에 알려진 아카이브 위치에서 설치 재개를 시도합니다. Solaris Flash 설치 유틸리티가 마지막으로 알려진 위치에서 설치를 재개할 수 없다면 아카이브의 시작부터 재검색이 다시 시작되고 시간 초과 이전에 검색된 데이터는 폐기됩니다. 패키지가 설치되는 동안 시간 초과 재연결이 발생하면 시간 초과가 무시되기 전에 검색된 패키지과 데이터의 시작부터 패키지가 재시도됩니다.
<code>proxy host:port</code>	<p><code>proxy</code> 키워드는 프록시 호스트 및 프록시 포트를 지정할 수 있게 합니다. 방화벽의 맞은 편에서 Solaris Flash 아카이브를 검색하기 위해 프록시 호스트를 사용할 수 있습니다. <code>proxy</code> 키워드를 지정할 때는 반드시 프록시 포트를 제공해야 합니다.</p>

예 11-3 FTP 서버에 저장된 아카이브

```
archive_location ftp://user1:secret@silver/archives/usrarchive.flar timeout 5
```

로컬 테이프에 저장된 아카이브

아카이브가 테이프에 저장되어 있으면 `archive_location` 키워드용으로 다음 구문을 사용하십시오.

```
archive_location local_tape device position
```

device Solaris Flash 아카이브를 저장한 테이프 드라이브의 이름. 장치 이름이 정식 경로라면 Solaris Flash 설치 유틸리티는 해당 경로에서 장치 노드로 아카이브를 읽어 들입니다. 정식 경로가 아닌 장치 이름을 제공하면 Solaris Flash 설치 유틸리티가 `/dev/rmt/`를 경로에 추가합니다.

position 아카이브를 저장한 테이프 드라이브상의 위치를 지정합니다. 위치를 제공하지 않으면 Solaris Flash 설치 유틸리티는 테이프 드라이브의 현 위치에서 아카이브를 읽어 들입니다. *position*을 지정함으로써 아카이브 이전에 테이프 드라이브에 시작 스크립트나 `sysidcfg` 파일을 배치할 수 있습니다.

예 11-4 로컬 테이프에 저장된 아카이브

```
archive_location local_tape /dev/rmt/0n 5
```

```
archive_location local_tape 0n 5
```

로컬 장치에 저장된 아카이브

Solaris Flash 아카이브를 디스켓이나 DVD와 같은 파일 시스템 지향 임의 액세스 장치에 저장했다면 로컬 장치에서 Solaris Flash 아카이브를 검색할 수 있습니다. `archive_location` 키워드용으로 다음 구문을 사용하십시오.

주 - 로컬 테이프용 구문을 사용함으로써 테이프와 같은 스트림 지향 장치로부터 아카이브를 검색할 수 있습니다.

```
archive_location local_device device path/filename file_system_type
```

device Solaris Flash 아카이브를 저장한 드라이브의 이름. 장치 이름이 정식 경로라면 해당 장치는 바로 마운트됩니다. 정식 경로가 아닌 장치 이름을 제공하면 Solaris Flash 설치 유틸리티가 `/dev/dsk/`를 경로에 추가합니다.

path 지정한 장치의 파일 시스템 루트에 상대적인 Solaris Flash 아카이브 경로. 경로에 `$HOST`가 포함되어 있으면 Solaris Flash 설치 유틸리티는 `$HOST`를 설치할 복제 시스템의 이름으로 대체합니다.

filename Solaris Flash 아카이브 파일의 이름

file_system_type 장치의 파일 시스템 유형을 지정합니다. 파일 시스템 유형을 제공하지 않으면 Solaris Flash 설치 유틸리티는 UFS 파일 시스템의 마운트를 시도합니다. UFS 마운트에 실패하면 Solaris Flash 설치 유틸리티는 HSFS 파일 시스템의 마운트를 시도합니다.

예 11-5 로컬 장치에 저장된 아카이브

UFS 파일 시스템으로 포맷된 로컬 하드 드라이브에서 아카이브를 검색하려면 다음 명령어를 사용합니다.

```
archive_location local_device c0t0d0s0 /archives/$HOST
```

HSFS 파일 시스템이 있는 로컬 CD-ROM에서 아카이브를 검색하려면 다음 명령어를 사용합니다.

```
archive_location local_device c0t0d0s0 /archives/usrarchive
```

로컬 파일에 저장된 아카이브

복제 시스템을 부트한 미니루트에 저장한 아카이브를 로컬 파일로 읽어들이실 수 있습니다. 사용자 정의 JumpStart 설치를 수행하는 경우 DVD, CD 또는 NFS 기반 미니루트에서 시스템을 부트합니다. 설치 소프트웨어가 로드되고 이 미니루트에서 실행됩니다. 따라서 DVD, CD 또는 NFS 기반 미니루트에 저장한 Solaris Flash 아카이브는 로컬 파일로 액세스할 수 있습니다. `archive_location` 키워드용으로 다음 구문을 사용합니다.

```
archive_location local_file path/filename
```

path 아카이브의 위치. 시스템을 Solaris Software - 1 CD나 Solaris Operating System DVD에서 부트할 때 해당 경로는 로컬 파일로서 시스템에 액세스할 수 있어야 합니다. 시스템은 Solaris Software - 1 CD 또는 Solaris Operating System DVD에서 부트된 경우 `/net` 또는 다른 자동 마운트된 디렉토리에 액세스할 수 없습니다.

filename Solaris Flash 아카이브 파일의 이름

예 11-6 로컬 파일에 저장된 아카이브

```
archive_location local_file /archives/usrarchive
```

backup_media 프로필 키워드

```
backup_media type path
```

`backup_media`는 디스크 공간 재할당이 필요할 때 업그레이드 옵션으로만 사용할 수 있습니다.

비전역 영역이 설치된 경우 이 키워드를 사용하지 마십시오. JumpStart 프로필에서 이 키워드를 사용하면 업그레이드가 중지되고 오류 메시지가 표시됩니다.

`backup_media`는 공간 부족으로 인해 업그레이드 중에 공간을 재할당해야 하는 경우, 파일 시스템 백업을 위해 사용할 매체를 정의합니다. 백업용으로 여러 개의 테이프나 디스켓이 필요한 경우, 업그레이드 중에 테이프나 디스켓을 넣으라는 메시지가 나타납니다.

올바른 <i>type</i> 값	올바른 <i>path</i> 값	설명
<code>local_tape</code>	<code>/dev/rmt/n</code>	업그레이드 중인 시스템의 로컬 테이프 드라이브. <i>path</i> 는 테이프 드라이브용 문자(원시) 장치 경로여야 합니다. <i>n</i> 은 테이프 드라이브의 번호입니다.
<code>local_diskette</code>	<code>/dev/rdisketten</code>	업그레이드 중인 시스템의 로컬 디스켓 드라이브. <i>path</i> 는 디스켓 드라이브용 문자(원시) 장치 경로여야 합니다. <i>n</i> 은 디스켓 드라이브의 번호입니다. 백업용으로 사용할 디스켓은 반드시 포맷해야 합니다.
<code>local_filesystem</code>	<code>/dev/dsk/cwt xdys z</code> <code>/file_system</code>	업그레이드 중인 시스템의 로컬 파일 시스템. 업그레이드로 인해 변경될 로컬 파일 시스템은 지정할 수 없습니다. <i>path</i> 는 디스크 슬라이스용 블록 장치일 수 있습니다. 예를 들어, <code>/dev/dsk/c wtxdys z</code> 의 <i>tx</i> 는 필요하지 않습니다. 또는 <i>path</i> 는 <code>/etc/vfstab</code> 파일에 의해 마운트되는 파일 시스템에 대한 절대 경로일 수 있습니다.
<code>remote_filesystem</code>	<code>host:/file_system</code>	원격 시스템상의 NFS 파일 시스템. <i>path</i> 는 원격 시스템 <i>host</i> 의 이름이나 IP 주소 그리고 NFS 파일 시스템인 <i>file_system</i> 의 절대 경로를 포함해야 합니다. NFS 파일 시스템은 읽기/쓰기 액세스를 가져야 합니다.
<code>remote_system</code>	<code>user@host :/directory</code>	원격 셸, <code>rsh</code> 가 도달할 수 있는 원격 시스템의 디렉토리. 업그레이드 중인 시스템은 원격 시스템의 <code>.rhosts</code> 파일을 통해 원격 시스템에 액세스할 수 있어야 합니다. <i>path</i> 는 원격 시스템 <i>host</i> 의 이름 및 <i>directory</i> 디렉토리의 절대 경로를 포함해야 합니다. 사용자 로그인 ID <i>user</i> 가 지정되지 않았으면 <code>root</code> 를 기본값으로 사용합니다.

예 11-7 backup_media 프로필 키워드

```
backup_media local_tape /dev/rmt/0
```

```
backup_media local_diskette /dev/rdiskette1
```

```
backup_media local_filesystem /dev/dsk/c0t3d0s4
```

```
backup_media local_filesystem /export
```

```
backup_media remote_filesystem system1:/export/temp
```

```
backup_media remote_system user1@system1:/export/temp
```

boot_device 프로필 키워드

`boot_device device eeprom`

`boot_device`는 JumpStart 프로그램이 루트(/) 파일 시스템과 시스템의 부트 장치를 설치할 장소를 장치에 지정합니다. `boot_device`는 루트(/) 파일 시스템과 `root_device` 키워드(지정된 경우)를 지정하는 `filesys` 키워드를 일치시켜야 합니다.

프로필에 `boot_device` 키워드를 지정하지 않으면 설치 중에 다음 `boot_device` 키워드가 기본값으로 지정됩니다.

`boot_device any update`

`device` 다음 값 중 하나를 사용합니다.

SPARC: <code>cw txdy sz</code> 또는 <code>cx dysz</code>	JumpStart 프로그램이 루트(/) 파일 시스템, 예를 들어 <code>c0t0d0s0</code> 을 배치하는 디스크 슬라이스.
x86: <code>cwtxdy</code> 또는 <code>cx dy</code>	JumpStart 프로그램이 루트(/) 파일 시스템, 예를 들어 <code>c0d0</code> 을 배치하는 디스크.
<code>existing</code>	JumpStart 프로그램은 루트(/) 파일 시스템을 시스템의 기존 부트 장치에 배치합니다.
<code>any</code>	JumpStart 프로그램이 루트(/) 파일 시스템을 배치할 위치를 선택합니다. JumpStart 프로그램이 시스템의 기존 부트 장치를 사용하려고 시도합니다. JumpStart 프로그램은 필요한 경우 다른 부트 장치 사용을 선택할 수도 있습니다.

`eeprom` 시스템의 EEPROM을 업데이트할 것인지 아니면 보존할 것인지 선택합니다.

`eeprom` 값을 사용하여 시스템의 EEPROM을 업데이트할 수도 있습니다. 시스템의 EEPROM을 업데이트함으로써 시스템을 새 부트 장치로부터 자동 부트시킬 수 있게 됩니다.

주 -x86: `preserve` 값을 지정해야 합니다.

<code>update</code>	JumpStart 프로그램은 시스템의 EEPROM을 지정된 부트 장치로 업데이트하여 설치된 시스템이 해당 장치로부터 자동 부트될 수 있게 합니다.
<code>preserve</code>	시스템 EEPROM의 부트 장치 값은 변경되지 않습니다. 시스템 EEPROM을 변경하지 않은 채 새 부트 장치를 지정하면 시스템의 EEPROM을 직접 변경하여 시스템이 새 부트 장치로부터 자동 부트되도록 해야 합니다.

예 11-8 boot_device 프로필 키워드

```
boot_device c0t0d0s2 update
```

bootenv createbe 프로필 키워드

```
bootenv createbe bename new_BE_name filesystem mountpoint:device:fs_options
[filesystem...]
```

`bootenv createbe` 키워드를 사용하면 Solaris OS를 설치하면서 비어 있는 비활성 부트 환경을 동시에 만들 수 있습니다. 최소한 루트(/) 파일 시스템은 만들어야 합니다. 해당 슬라이스는 지정된 파일 시스템에 대해 예약되지만 파일 시스템은 복사되지 않습니다. 부트 환경의 이름은 지정되지만 Solaris Flash 아카이브를 사용하여 설치될 때까지 실제로 만들어지지 않습니다. 아카이브를 사용하여 빈 부트 환경이 설치되면 파일 시스템은 예약된 슬라이스에 설치됩니다. 다음은 *bename* 및 *filesystem*에 대한 값의 목록입니다.

bename *new_BE_name*

*bename*은 만들 새 부트 환경의 이름을 지정합니다. *new_BE_name*은 30자 이상일 수 없으며 영숫자 문자만 포함할 수 있고 멀티바이트 문자를 포함할 수 없습니다. 이름은 파일 시스템에서 고유해야 합니다.

filesystem *mountpoint:device:fs_options*

*filesystem*은 새 부트 환경에 만들 파일 시스템의 종류와 수를 결정합니다. 적어도 루트(/) 파일 시스템을 포함하고 있는 하나의 슬라이스를 정의해야 합니다. 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다.

- *mountpoint*는 모든 유효한 마운트 지점일 수도 있고 스왑 슬라이스를 나타내는 -(하이픈)일 수도 있습니다.
- *device*는 설치되는 운영 체제가 처음에 부트될 때 사용할 수 있어야 합니다. 이 장치는 *free*와 같은 JumpStart 특수 저장 장치와는 관계가 없습니다. 이 장치는 Solaris 볼륨 관리자 볼륨 또는 Veritas 볼륨 관리자 볼륨일 수 없습니다. *device*는 디스크 드라이브의 이름이며 /dev/dsk/cwtxd ysz 형식을 사용합니다.
- *fs_options*는 다음 중 하나일 수 있습니다.
 - UFS 파일 시스템을 나타내는 *ufs*
 - 스왑 파일 시스템을 나타내는 *swap*. 스왑 마운트 지점은 -(하이픈)이어야 합니다.

이 키워드 사용에 대한 프로필 예 및 배경에 대해서는 다음을 참조하십시오.

프로필 예에 대해서

예 6-11

비활성 부트 환경을 만들고 업그레이드하고 활성화하는 Solaris Live Upgrade의 사용에 대한 배경에 대해서

Solaris 10 6/06 설치 설명서: **Solaris Live Upgrade** 및 업그레이드 계획의 6 장, “Solaris Live Upgrade(개요)”

Solaris Flash 아카이브 사용 배경에 대해서

Solaris 10 6/06 설치 설명서: **Solaris Flash** 아카이브(작성 및 설치)의 1 장, “Solaris Flash(개요)”

client_arch 프로필 키워드

client_arch *karch_value* ...

client_arch는 운영 체제 서버가 서버가 사용하는 것 이외의 다른 플랫폼 그룹도 지원하도록 지정합니다. 프로필에 client_arch를 지정하지 않는 경우 운영 체제 서버를 사용하는 디스크가 없는 모든 클라이언트는 서버로 동일한 플랫폼 그룹을 포함해야 합니다. 운영 체제 서버의 지원을 원하는 개별 플랫폼 그룹을 지정해야 합니다.

*karch_value*의 유효한 값은 sun4u와 i86pc입니다. 플랫폼 이름과 다양한 시스템에 대한 자세한 목록은 <http://docs.sun.com>의 **Solaris Sun** 하드웨어 플랫폼 안내서를 참조하십시오.

주 - system_type이 server로 지정된 경우에만 client_arch를 사용할 수 있습니다.

client_root 프로필 키워드

client_root *root_size*

client_root는 루트 공간의 양, 즉 MB 단위의 *root_size*를 정의하여 각 클라이언트에 할당합니다. 서버의 프로필에서 client_root를 지정하지 않으면 설치 소프트웨어는 클라이언트 한 대당 15MB의 루트 공간을 할당합니다. 클라이언트 루트 영역의 크기는 num_clients 키워드와 함께 사용하여 /export/root 파일 시스템을 위해 예약해 둘 공간을 결정합니다.

주 - client_root는 system_type이 server로 지정된 경우에만 사용할 수 있습니다.

client_swap 프로필 키워드

client_swap *swap_size*

client_swap은 디스크 없는 클라이언트 각각에 대해 할당할 스왑 공간 크기 *swap_size*를 MB 단위로 정의합니다. 프로필에서 client_swap을 지정하지 않으면 32MB의 스왑 공간이 기본값으로 할당됩니다.

주 - system_type이 server로 지정된 경우에만 client_swap을 사용할 수 있습니다.

예 11-9 client_swap 프로필 키워드

다음 예에서는 각각의 디스크 없는 클라이언트가 64MB의 스왑 공간을 갖도록 지정합니다.

```
client_swap 64
```


swap 크기 결정 방법

프로필에 swap 크기를 지정하지 않으면 JumpStart 프로그램이 시스템의 물리적 메모리를 기반으로 스왑 공간의 크기를 결정합니다. 표 11-5는 사용자 정의 JumpStart 설치 중에 swap 크기가 결정되는 방법을 보여 줍니다.

표 11-5 swap 크기 결정

물리적 메모리(MB)	스왑 공간(MB)
16-64	32
64-128	64
128-512	128
512 초과	256

Jump 프로그램은 swap의 크기를 swap이 있는 디스크의 20퍼센트 미만으로 만듭니다. 다른 파일 시스템을 레이아웃한 후 디스크에 사용 가능 공간이 있을 경우 할당은 달라집니다. 사용 가능 공간이 있으면 JumpStart 프로그램은 swap에 사용 가능 공간을 할당하고 가능하다면 표 11-5에 표시된 양을 할당합니다.

주 - 물리적 메모리와 스왑 공간을 합한 값은 최소 32MB 이상이어야 합니다.

cluster 프로필 키워드(소프트웨어 그룹 추가)

`cluster group_name`

cluster는 시스템에 추가할 소프트웨어 그룹을 지정합니다.

주 - 소프트웨어 그룹은 클러스터 및 패키지 모음이 포함된 메타클러스터입니다. 소프트웨어 그룹은 cluster 키워드와 group_name 변수를 사용하여 설치합니다. 이 클러스터 키워드는 초기 설치에서만 설치할 수 있습니다. 이 cluster 키워드는 clustertoc(4) 파일에 있는 메타클러스터를 참조합니다.

클러스터는 이름이 SUNWname인 패키지의 모음입니다. 클러스터는 cluster 키워드와 cluster_name 변수를 사용하여 설치합니다. 클러스터는 초기 설치 또는 업그레이드 중에 소프트웨어 그룹(메타클러스터)에 추가하거나 여기에서 제거할 수 있습니다.

각 소프트웨어 그룹의 group_name은 다음 표에 나열되어 있습니다.

소프트웨어 그룹	group_name
축소 네트워크 지원 소프트웨어 그룹	SUNWCrnet

소프트웨어 그룹	group_name
코어 시스템 지원 소프트웨어 그룹	SUNWCreq
최종 사용자 Solaris 소프트웨어 그룹	SUNWCuser
개발자 Solaris 소프트웨어 그룹	SUNWCprog
전체 Solaris 소프트웨어 그룹	SUNWCall
전체 Solaris 소프트웨어 그룹 및 OEM 지원	SUNWCXall

다음 제한이 적용됩니다.

- 프로필에는 소프트웨어 그룹을 하나만 지정할 수 있습니다.
- 다른 cluster 및 package 항목 이전에 소프트웨어 그룹을 지정해야 합니다.
- 프로필에서 cluster를 사용하여 소프트웨어 그룹을 지정하지 않으면 최종 사용자 소프트웨어 그룹인 SUNWCuser가 시스템에 설치됩니다.

소프트웨어 그룹에 대한 자세한 내용은 36 페이지 “소프트웨어 그룹의 권장 디스크 공간”을 참조하십시오.

cluster 프로필 키워드(클러스터 추가 또는 삭제)

`cluster cluster_name add_delete_switch`

cluster는 클러스터가 시스템에 설치될 소프트웨어 그룹에 추가 또는 삭제될 것인지 여부를 지정합니다.

cluster_name SUNWC name 형식이여야 하는 클러스터 이름.

add_delete_switch 지정된 클러스터를 추가 또는 삭제할지 나타내는 선택적 키워드. add 또는 delete를 사용합니다. add 또는 delete를 지정하지 않으면 add가 기본값으로 사용됩니다.

업그레이드 중에 cluster를 사용하는 경우 다음 조건이 적용됩니다.

- 시스템에 이미 설치된 모든 클러스터가 자동으로 업그레이드됩니다.
- *cluster_name* add를 지정했는데 *cluster_name*이 시스템에 설치되어 있지 않으면 해당 클러스터가 설치됩니다.
- *cluster_name* delete를 지정했고 *cluster_name*이 시스템에 설치되어 있으면 업그레이드가 시작되기 전에 해당 패키지가 삭제됩니다.

주- 비전역 영역이 설치된 경우 이 키워드를 사용하여 업그레이드하지 마십시오. 이 키워드를 사용하면 업그레이드는 계속되지만 키워드가 무시됩니다.

주- 소프트웨어 그룹은 클러스터 및 패키지 모음이 포함된 메타클러스터입니다. 소프트웨어 그룹은 `cluster` 키워드와 `group_name` 변수를 사용하여 설치합니다. 이 클러스터 키워드는 초기 설치에서만 설치할 수 있습니다. 이 `cluster` 키워드는 `clustertoc(4)` 파일에 있는 메타클러스터를 참조합니다.

클러스터는 패키지의 모음입니다. 클러스터를 그룹으로 묶어 소프트웨어 그룹(메타클러스터)을 만들 수 있습니다. 클러스터 이름은 항상 `SUNW<name>` 형식입니다. 클러스터는 `cluster` 키워드와 `cluster_name` 변수를 사용하여 설치합니다. 클러스터는 초기 설치 또는 업그레이드 중에 소프트웨어 그룹(메타클러스터)에 추가하거나 여기에서 제거할 수 있습니다.

dontuse 프로필 키워드

`dontuse disk_name ...`

기본적으로 JumpStart 프로그램은 분할 기본값이 지정되었을 때 시스템에서 작동 가능한 모든 디스크를 사용합니다. `dontuse`는 JumpStart 프로그램에서 사용하지 않을 한 개 이상의 디스크를 지정합니다. `disk_name`은 `cxydz` 또는 `cydz` 형식으로 지정해야 합니다(예: `c0t0d0`).

주- 동일한 프로필에서 `dontuse` 키워드와 `usedisk` 키워드를 모두 지정할 수는 없습니다.

x86: fdisk 프로필 키워드

`fdisk disk_name type size`

`fdisk`는 `fdisk` 분할 영역을 x86 기반 시스템에서 설정하는 방법을 정의합니다. `fdisk`를 두 번 이상 지정할 수 있습니다. `fdisk`가 x86 기반 시스템을 분할하는 경우 다음과 같은 경우가 발생합니다.

- 디스크의 모든 `fdisk` 분할 영역은 `size`에 `delete` 또는 `0`의 값을 할당하여 `fdisk` 키워드로 분할 영역을 삭제하지 않는 한 보존됩니다. 또한, 모든 존재하는 `fdisk` 분할 영역은 `size`가 `all`로 설정되면 삭제됩니다.
 - 루트(/) 파일 시스템을 포함하는 Solaris `fdisk` 분할 영역은 항상 디스크의 활성 분할 영역으로 지정됩니다.
-

주- 시스템은 기본적으로 활성 분할 영역에서 부트됩니다.

- `fdisk` 키워드가 프로필에 지정되지 않으면 다음 `fdisk` 키워드가 설치 중에 기본적으로 사용됩니다.

`fdisk all solaris maxfree`

- `fdisk` 항목이 프로필에 나열된 순서로 처리됩니다.

`disk_name` 다음 값을 사용하여 `fdisk` 분할 영역이 작성 또는 삭제될 위치를 지정합니다.

- type*
 - `cxytdz` 또는 `cydz` - 특정 디스크(예: `c0t3d0`)
 - `rootdisk` - 시스템의 루트 디스크의 값을 포함하는 변수로 185 페이지 “시스템의 루트 디스크 결정 방법”에 설명된 대로 JumpStart 프로그램에 의해 결정됩니다.
 - `all` - 선택된 모든 디스크
- 다음 값을 사용하여 지정된 디스크에서 생성하거나 삭제할 `fdisk` 분할 영역의 유형을 지정합니다.

- `solaris` - Solaris `fdisk` 분할 영역(SUNIXOS `fdisk` 유형)
- `dosprimary` - 데이터 DOS에 대하여 확장 또는 예약된 `fdisk` 분할 영역이 아닌 기본 DOS `fdisk` 분할 영역. `size`에 `delete`의 값을 할당하여 `fdisk` 분할 영역을 삭제하는 경우 `dosprimary`는 DOSHUGE, DOSOS12 및 DOSOS16 `fdisk` 유형의 별칭입니다. `fdisk` 분할 영역을 만드는 경우 `dosprimary`는 DOSHUGE `fdisk` 분할 영역의 별칭입니다.
- `DDD` - `fdisk` 분할 영역의 정수. `DDD`는 1과 255를 포함해 그 사이에 있는 정수입니다.

주 - `size`가 `delete`인 경우에만 이 값을 지정할 수 있습니다.

- `0xHH` - 16진수 `fdisk` 분할 영역. `HH`는 01 - FF 사이의 16진수입니다.

주 - `size`가 `delete`인 경우에만 이 값을 지정할 수 있습니다.

다음 표는 일부 `fdisk` 유형에 대한 정수와 16진수를 표시합니다.

<code>fdisk</code> 유형	<code>DDD</code>	<code>HH</code>
DOSOS12	1	01
PCIXOS	2	02
DOSOS16	4	04
EXTDOS	5	05
DOSHUGE	6	06
DOSDATA	86	56
OTHEROS	98	62
UNIXOS	99	63

size 다음 값 중 하나를 사용합니다.

- *DDD* - fdisk MB 단위의 크기 *DDD*의 분할 영역은 지정된 디스크에 생성됩니다. *DDD*는 정수이어야 하며 JumpStart 프로그램은 자동으로 숫자를 가장 가까운 실린더 경계로 반올림합니다. 0의 값을 지정하는 것은 delete를 지정하는 것과 동일합니다.
- *all* - fdisk 분할 영역은 전체 디스크에 생성됩니다. 모든 기존 fdisk 분할 영역이 삭제됩니다.

x86 전용 - *all* 값은 *type*이 *solaris*인 경우에만 지정할 수 있습니다.

- *maxfree* - fdisk 분할 영역은 지정된 디스크의 최대의 인접 사용 가능 공간에 생성됩니다. 지정된 *type*의 fdisk 분할 영역이 디스크에 이미 존재하면 기존 fdisk 분할 영역이 사용됩니다. 새로운 fdisk 분할 영역은 디스크에 생성되지 않습니다.

x86 전용 - 디스크는 최소한 하나의 미사용 fdisk 분할 영역을 포함해야 합니다. 또한 디스크는 사용 가능 공간이 있어야 합니다. 그렇지 않으면 설치가 실패합니다. *maxfree* 값은 *type*이 *solaris* 또는 *dosprimary*인 경우에만 지정할 수 있습니다.

- *delete* - 지정된 *type*의 모든 fdisk 분할 영역이 지정된 디스크에서 삭제됩니다.

filesystem 프로필 키워드(원격 파일 시스템 마운트)

filesystem *server:path server_address mount_pt_name mount_options*

filesystem를 나열된 값과 함께 사용함으로써 JumpStart 프로그램은 시스템이 부트될 때 설치된 시스템이 원격 파일 시스템을 자동으로 마운트하도록 설정합니다. filesystem를 두 번 이상 지정할 수 있습니다.

<i>server</i>	원격 파일 시스템이 위치한 서버의 이름으로, 뒤에 콜론이 옵니다.
<i>path</i>	원격 파일 시스템의 마운트 지점 이름. 예를 들어, /usr 또는 /export/home
<i>server_address</i>	<i>server:path</i> 에 지정된 서버의 IP 주소. 네트워크 상에서 이름 서비스가 실행되고 있지 않으면 <i>server_address</i> 값을 사용하여 /etc/hosts 파일에 서버의 호스트 이름과 IP 주소를 채웁니다. 서버의 IP 주소를 지정하지 않으려면 마이너스 기호(-)를 지정해야 합니다. 예를 들어, 네트워크 상에서 실행 중인 이름 서비스가 있으면 서버의 IP 주소를 지정할 필요가 없습니다.
<i>mount_pt_name</i>	원격 파일 시스템을 마운트할 마운트 지점의 이름
<i>mount_options</i>	mount(1M) 명령어의 -o 옵션과 동일한 하나 이상의 마운트 옵션. 마운트 옵션은 지정된 <i>mount_pt_name</i> 을 위해 /etc/vfstab 입력 항목에 추가됩니다.

주 - 두 개 이상의 마운트 옵션을 지정해야 하는 경우 마운트 옵션을 쉼표로 구분하고 공백은 사용하지 말아야 합니다(예: ro, quota).

예 11-10 `filesys` 프로필 키워드

```
filesys sherlock:/export/home/user2 - /home
```

filesys 프로필 키워드(로컬 파일 시스템 만들기)

`filesys slice size file_system optional_parameters`

`filesys`를 나열된 값과 함께 사용하면 JumpStart 프로그램이 설치 도중에 로컬 파일 시스템을 작성합니다. `filesys`를 두 번 이상 지정할 수 있습니다.

`slice` 다음 값 중 하나를 사용합니다.

`any` JumpStart 프로그램이 임의의 디스크에 파일 시스템을 배치합니다.

주 - `size`가 `existing`, `all`, `free` 또는 `start`인 경우에는 `any`를 지정할 수 없습니다. `size` 또는 `ignore`입니다.

`cw txdy sz` 또는 `cx dysz` JumpStart 프로그램이 파일 시스템, 예를 들어 `c0t0d0s0` 또는 `c0d0s0`을 배치하는 디스크 슬라이스.

`rootdisk.sn` 시스템의 루트 디스크에 대한 값을 포함하는 변수로 185 페이지 “시스템의 루트 디스크 결정 방법”에서 설명된 대로 JumpStart 프로그램에서 결정합니다. 접미어 `sn`은 디스크상의 특정 슬라이스를 가리킵니다.

`size` 다음 값 중 하나를 사용합니다.

`num` 파일 시스템의 크기가 `num`MB로 설정됩니다.

`existing` 기존 파일 시스템의 현재 크기가 사용됩니다.

주 - `existing` 값을 사용하면 `file_system`을 별개의 `mount_pt_name`으로 지정함으로써 기존 슬라이스의 이름을 변경할 수 있습니다.

`auto` 선택된 소프트웨어에 따라 파일 시스템의 크기가 자동으로 결정됩니다.

	all	지정된 <i>slice</i> 가 파일 시스템용으로 전체 디스크를 사용합니다. all 값을 지정하면 지정된 디스크에 다른 파일 시스템을 배치할 수 없습니다.
	free	디스크의 나머지 사용되지 않은 공간을 파일 시스템용으로 사용합니다.
		<hr/> 주 - free 를 <i>filesys</i> 의 값으로 사용하면 <i>filesys</i> 항목이 프로필에서 마지막 항목이어야 합니다. <hr/>
	start:size	파일 시스템이 명시적으로 분할되어 있습니다. start 는 슬라이스가 시작되는 실린더입니다. size 는 슬라이스용 실린더 개수입니다.
<i>file_system</i>		<i>file_system</i> 값은 선택 사항이며 <i>slice</i> 가 any 또는 cwtxdysz 로 지정된 경우 사용됩니다. <i>file_system</i> 이 지정되어 있지 않으면 unnamed 가 기본값으로 설정됩니다. unnamed 가 설정되면 <i>optional_parameters</i> 값을 지정할 수 없습니다. 다음 값 중 하나를 사용합니다.
	mount_pt_name	파일 시스템의 마운트 지점 이름, 예를 들어, /var .
	swap	지정된 <i>slice</i> 가 swap 으로 사용됩니다.
	overlap	지정된 <i>slice</i> 는 디스크 범위의 표현으로 정의됩니다. VTOC 값은 V_BACKUP 입니다. 기본적으로 슬라이스 2는 전체 디스크를 나타내는 겹친 슬라이스입니다.
		<hr/> 주 - overlap 은 size 가 existing , all 또는 start:size 인 경우에만 지정할 수 있습니다. <hr/>
	unnamed	지정된 <i>slice</i> 는 <i>slice</i> 가 마운트 지점 이름을 갖지 않는 원시 슬라이스로 정의됩니다. <i>file_system</i> 을 지정하지 않으면 unnamed 가 기본값으로 사용됩니다.
	ignore	지정된 <i>slice</i> 는 JumpStart 프로그램에서는 사용하거나 인식되지 않습니다. 설치 도중에 디스크에서 무시할 파일 시스템을 지정하려면 이 옵션을 사용할 수 있습니다. JumpStart 프로그램은 동일한 이름을 가진 동일 디스크상에 새 파일 시스템을 만듭니다. ignore 는 partitioning existing 이 지정되어 있는 경우에만 사용할 수 있습니다.
<i>optional_parameters</i>		다음 값 중 하나를 사용합니다.
	preserve	지정된 <i>slice</i> 의 파일 시스템이 보존됩니다.

mount_options

주 - preserve는 *size*가 *existing*이고 *slice*가 *cwtxdysz*인 경우에만 지정할 수 있습니다.

mount(1M) 명령어의 *-o* 옵션과 동일한 하나 이상의 마운트 옵션. 마운트 옵션은 지정된 *mount_pt_name*을 위해 */etc/vfstab* 입력 항목에 추가됩니다.

주 - 두 개 이상의 마운트 옵션을 지정해야 하는 경우 마운트 옵션을 쉼표로 구분하고 공백은 사용하지 말아야 합니다(예: *ro,quota*).

filesystem 프로필 키워드(RAID-1 볼륨 작성)

filesystem mirror[:name]slice [slice] size file_system optional_parameters

JumpStart 프로그램은 나열된 값을 가진 *filesystem mirror* 키워드를 사용하여 미러된 파일 시스템을 만드는 데 필요한 RAID-1 및 RAID-0 볼륨을 만듭니다. 다양한 파일 시스템에 RAID-1 볼륨을 만들기 위해 *filesystem mirror*를 두 번 이상 지정할 수 있습니다.

주 - *filesystem mirror* 키워드는 초기 설치에만 지원됩니다.

name

이 옵션 키워드를 사용하여 RAID-1 볼륨(미러)의 이름을 지정할 수 있습니다. 미러 이름은 문자 "d"로 시작하고 그 뒤에 0에서 127 사이의 숫자가 와야 합니다(예: *d100*). 미러 이름을 지정하지 않으면 사용자 정의 JumpStart 프로그램이 사용자의 미러 이름을 할당합니다. 미러 이름을 지정하는 방법에 대한 지침은 204 페이지 "사용자 정의 JumpStart 및 Solaris Live Upgrade에 대한 RAID 볼륨 이름 요구 사항 및 지침"을 참조하십시오.

slice

이 값은 사용자 정의 JumpStart 프로그램에서 사용자가 미러하려는 파일 시스템을 둘 디스크 슬라임을 지정합니다. 슬라임 값은 *cwtxdysz* 형식을 따라야 합니다(예: *c0t0d0s0* 또는 *c0t0d0s5*). 사용자 정의 JumpStart 프로그램은 슬라임에 RAID-0 볼륨(단일 슬라임 연결)을 만들고 연결을 미러할 RAID-1 볼륨을 만듭니다. 2개의 RAID-0 볼륨에 대하여 최고 2개 슬라임을 지정할 수 있습니다.

size

이 값은 파일 시스템의 크기(MB)를 지정합니다.

file_system

이 값은 복제 중인 파일 시스템을 지정합니다. 사용자 정의 JumpStart 프로그램은 지정된 슬라임에서 RAID-1 볼륨을 작성하고 지정된 파일 시스템의 RAID-1 볼륨을 마운트합니다. 루트(/), */usr* 및 */var* 같은 핵심 파일 시스템뿐만 아니라 *swap*을 파일 시스템으로 지정할 수도 있습니다.

optional_parameters mount(1M) 명령어의 -o 옵션과 동일한 하나 이상의 마운트 옵션. 마운트 옵션은 지정된 *file_system*의 /etc/vfstab 항목에 추가됩니다. 두 개 이상의 마운트 옵션을 지정해야 하는 경우 마운트 옵션을 쉼표로 구분하고 공백은 사용하지 말아야 합니다(예: ro,quota).

설치 중에 미러된 파일 시스템을 만드는 것에 대한 자세한 내용은 12 장을 참조하십시오.

forced_deployment 프로필 키워드(Solaris Flash 차등 아카이브 설치)

forced_deployment

*forced_deployment*는 Solaris Flash 차등 아카이브를 강제로 소프트웨어 예상과 다른 복제 시스템으로 설치합니다.



주의 - *forced_deployment*를 사용하는 경우, 예상 상태로 복제 시스템을 가져오기 위해 새로운 모든 파일이 삭제됩니다. 파일을 삭제할지 여부를 결정하지 않은 경우, 설치를 중단하여 새 파일을 보호하는 기본값을 사용합니다.

geo 프로필 키워드

geo region

*geo*는 시스템에 설치하거나 시스템 업그레이드 시 추가할 지역 로케를 지정합니다. *region*은 설치할 로케가 포함된 지역적 영역을 지정합니다. *region*에 지정할 수 있는 값은 다음 표에 나열되어 있습니다.

주 - 비전역 영역이 설치된 경우 이 키워드를 사용하여 업그레이드하지 마십시오. 이 키워드를 사용하면 업그레이드는 계속되지만 키워드가 무시됩니다.

값	설명
N_Africa	이집트를 포함한 북아프리카
C_America	코스타리카, 엘살바도르, 과테말라, 멕시코, 니카라과, 파나마를 포함한 중미 지역
N_America	캐나다, 미국을 포함한 북미 지역
S_America	아르헨티나, 볼리비아, 브라질, 칠레, 콜롬비아, 에콰도르, 파라과이, 페루, 우루과이, 베네수엘라를 포함한 남미 지역
Asia	일본, 대한민국, 중국, 대만, 태국을 포함한 아시아 지역
Ausi	호주, 뉴질랜드를 포함한 오스트랄라시아 지역
C_Europe	오스트리아, 체코 공화국, 독일, 헝가리, 폴란드, 슬로바키아, 스위스를 포함한 중부 유럽 지역

값	설명
E_Europe	알바니아, 보스니아, 불가리아, 크로아티아, 에스토니아, 라트비아, 리투아니아, 마케도니아, 루마니아, 러시아, 세르비아, 슬로베니아, 터키를 포함한 동유럽 지역
N_Europe	덴마크, 핀란드, 아이슬란드, 노르웨이, 스웨덴을 포함한 북유럽 지역
S_Europe	그리스, 이탈리아, 포르투갈, 스페인을 포함한 남부 유럽 지역
W_Europe	벨기에, 프랑스, 영국, 아일랜드, 네덜란드를 포함한 서유럽 지역
M_East	이스라엘을 포함한 중동 지역

이전에 나열된 각 지역 로케를 구성하는 구성 요소 로케 값의 완전한 목록은 국제 언어 환경 설명서에서 제시합니다.

주 - 시스템에 추가할 각각의 로케에 대해 `geo` 키워드를 지정할 수 있습니다.

install_type 프로필 키워드

`install_type initial_upgrade_flash_switch`

`install_type`은 시스템의 운영 환경을 지운 후 새 Solaris OS를 설치할 것인지, 시스템의 기존 Solaris OS를 업그레이드할 것인지, 아니면 시스템에 Solaris Flash 아카이브를 설치할 것인지 정의합니다.

주 - 프로필에서 `install_type`을 지정해야 하며 `install_type`은 모든 프로필의 첫 번째 프로필 키워드여야 합니다.

프로필 키워드 `initial_upgrade_flash_switch`의 다음 옵션 중 하나를 사용해야 합니다.

<code>initial_install</code>	Solaris OS의 초기 설치를 수행하려면 지정합니다.
<code>upgrade</code>	Solaris OS의 업그레이드를 수행하려면 지정합니다.
<code>flash_install</code>	모든 파일을 덮어쓰는 Solaris Flash 아카이브를 설치하려면 지정합니다.
<code>flash_update</code>	지정된 파일만을 덮어쓰는 Solaris Flash 차등 아카이브를 설치하려면 지정합니다.

주 - 일부 프로필 키워드는 `initial_install` 옵션과 함께만 사용할 수 있습니다. 일부 프로필 키워드는 `upgrade` 옵션과 함께만 사용할 수 있습니다. 일부 프로필 키워드는 `flash_install` 옵션과 함께만 사용할 수 있습니다.

layout_constraint 프로필 키워드

`layout_constraint slice_constraint minimum_size`

`layout_constraint`는 공간 문제 때문에 자동 레이아웃이 업그레이드 중에 공간을 재할당해야 할 경우, 자동 레이아웃이 파일 시스템에서 갖는 제약 조건을 지정합니다.

제한	설명
이 키워드는 업그레이드 옵션에만 사용됩니다.	<code>layout_constraint</code> 는 디스크 공간 재할당이 필요할 때 업그레이드 옵션용으로만 사용할 수 있습니다.
비전역 영역이 설치된 경우 이 키워드를 사용하지 마십시오.	이 키워드를 사용하면 업그레이드가 중지되고 오류 메시지가 표시됩니다.
<code>layout_constraint</code> 키워드를 지정하지 않은 경우	JumpStart 프로그램이 다음과 같이 디스크를 레이아웃합니다. <ul style="list-style-type: none"> 업그레이드를 위해 더 많은 공간을 필요로 하는 파일 시스템은 변경 가능으로 표시됩니다. 추가 공간이 필요한 파일 시스템으로서 동일한 디스크에 있고 <code>/etc/vfstab</code> 파일에 의해 마운트되는 파일 시스템은 변경 가능으로 표시됩니다. 나머지 파일 시스템은 자동 레이아웃이 파일 시스템을 변경할 수 없기 때문에 고정된 것으로 표시됩니다.
<code>layout_constraint</code> 키워드를 하나 이상 지정한 경우	JumpStart 프로그램이 다음과 같이 디스크를 레이아웃합니다. <ul style="list-style-type: none"> 업그레이드를 위해 더 많은 공간을 필요로 하는 파일 시스템은 변경 가능으로 표시됩니다. <code>layout_constraint</code> 키워드가 지정되어 있는 파일 시스템은 지정된 제약 조건과 함께 표시됩니다. 나머지 파일 시스템은 고정된 것으로 표시됩니다.
파일 시스템이 변경 가능으로 표시되지 않은 경우	파일 시스템이 변경 가능으로 표시되어야 하기 때문에 업그레이드용 추가 공간을 필요로 하는 파일 시스템의 제약 조건은 변경할 수 없습니다. <code>layout_constraint</code> 키워드를 사용하여 업그레이드를 위한 추가 공간이 필요한 파일 시스템의 <code>minimum_size</code> 값을 변경할 수 있습니다.
시스템이 업그레이드에 대하여 더 많은 공간을 필요로 하는 경우	자동 레이아웃의 공간 재할당을 돕기 위해서는 더 많은 파일 시스템을 변경 가능 또는 이동 가능하도록 선택해야 하며, 특히 업그레이드를 위해 추가 공간이 필요한 파일 시스템과 동일한 디스크상에 있는 파일 시스템들을 변경 가능 또는 이동 가능으로 선택해야 합니다.
<i>slice</i>	파일 시스템의 디스크 슬라이스를 제약을 지정하는 대상으로 지정합니다. 시스템 디스크 슬라이스는 <code>cwtxdysz</code> 또는 <code>cxdysz</code> 형식으로 지정해야 합니다.
<i>constraint</i>	지정된 파일 시스템에 대해 다음 제약 조건 중 하나를 사용합니다. <ul style="list-style-type: none"> <code>changeable</code> 자동 레이아웃은 파일 시스템을 다른 위치로 옮길 수 있으며 파일 시스템의 크기를 변경할 수 있습니다.

changeable 제약 조건은 /etc/vfstab 파일에 의해 마운트된 파일 시스템에서만 지정될 수 있습니다. *minimum_size* 값을 지정함으로써 파일 시스템의 크기를 변경할 수 있습니다.

파일 시스템을 변경 가능으로 표시하고 *minimum_size*가 지정되지 않으면 파일 시스템의 최소 크기는 요청된 최소 크기보다 10퍼센트 크게 설정됩니다. 예를 들어, 파일 시스템용 최소 크기가 100MB라면 변경된 크기는 110MB입니다. *minimum_size*가 지정되면 원래의 크기에서 최소 크기를 뺀 남아있는 여유 공간이 다른 파일 시스템용으로 사용됩니다.

movable 자동 레이아웃은 파일 시스템을 동일한 디스크의 다른 슬라이스나 다른 디스크로 옮길 수 있습니다. 파일 시스템 크기는 동일하게 유지됩니다.

available 자동 레이아웃은 파일 시스템의 모든 공간을 공간 재할당에 사용할 수 있습니다. 파일 시스템의 모든 데이터는 잃게 됩니다. **available** 제약 조건은 /etc/vfstab 파일에 의해 마운트되지 않은 파일 시스템에서만 지정될 수 있습니다.

collapse 자동 레이아웃은 지정된 파일 시스템을 부모 파일 시스템으로 옮겨 축소시킵니다. **collapse** 옵션을 사용하면 업그레이드의 일부로 시스템의 파일 시스템 개수를 줄일 수 있습니다. 예를 들어, 시스템에 /usr 및 /usr/share 파일 시스템이 있으면 /usr/share 파일 시스템을 축소시키는 경우 파일 시스템이 /usr 부모 파일 시스템으로 이동합니다. **collapse** 제약 조건은 /etc/vfstab 파일에 의해 마운트된 파일 시스템에서만 지정할 수 있습니다.

minimum_size 자동 레이아웃이 공간을 재할당한 후의 파일 시스템 크기를 지정합니다. *minimum_size* 옵션은 사용자가 파일 시스템의 크기를 변경할 수 있게 합니다. 파일 시스템에 할당되지 않은 공간이 추가되면 파일 시스템의 크기가 더 커질 수도 있습니다. 하지만 크기는 사용자가 지정한 값보다 작을 수는 없습니다. *minimum_size* 값은 선택 사항입니다. 파일 시스템을 변경 가능으로 표시했을 때만 이 값을 사용해야 하며 최소 크기는 파일 시스템이 기존 파일 시스템 내용을 위해 필요로 하는 것보다 작을 수 없습니다.

예 11-11 layout_constraint 프로필 키워드

```
layout_constraint c0t3d0s1 changeable 200
```

```
layout_constraint c0t3d0s4 movable
```

```
layout_constraint c0t3d1s3 available
```

```
layout_constraint c0t2d0s1 collapse
```

local_customization 프로필 키워드(Solaris Flash 아카이브 설치)

local_customization local_directory

복제 시스템에 Solaris Flash 아카이브를 설치하기 전에 사용자 정의 스크립트를 만들어 복제 시스템에 로컬 구성을 보존할 수 있습니다. local_customization 키워드는 이러한 스크립트를 저장해 놓은 디렉토리를 지정합니다. local_directory는 복제 시스템의 스크립트에 대한 경로입니다.

사전 배치 및 사후 배치 스크립트에 대한 내용은 **Solaris 10 6/06 설치 설명서: Solaris Flash 아카이브(작성 및 설치)**의 “사용자 정의 스크립트 만들기”를 참조하십시오.

locale 프로필 키워드

locale locale_name

주-locale은 초기 설치와 업그레이드 옵션 모두에서 사용할 수 있습니다.

locale은 지정한 locale_name에 대해 업그레이드할 때 설치 또는 추가하려는 로컬 패키지를 지정합니다. locale_name 값은 \$LANG 환경 변수용으로 사용되는 값과 동일합니다. 국제 언어 환경 설명서에는 유효한 로컬 값 목록이 포함되어 있습니다.

locale 키워드를 사용할 때 다음 사항을 고려하십시오.

- 기본 로컬을 사전 구성했다면 로컬은 자동으로 설치됩니다. 영어 패키지는 기본 설치됩니다.
- 시스템에 추가할 필요가 있는 각각의 로컬마다 locale 키워드를 지정할 수 있습니다.
- 비전역 영역이 설치된 경우 이 키워드를 사용하여 업그레이드하지 마십시오. 이 키워드를 사용하면 업그레이드는 계속되지만 키워드가 무시됩니다. 이미 시스템에 설치된 로컬은 자동으로 업그레이드됩니다.

metadb 프로필 키워드(상태 데이터베이스 복제본 만들기)

metadb slice [size size-in-blocks] [count number-of-replicas]

metadb 키워드를 사용하면 사용자 정의 JumpStart 설치 중에 Solaris 볼륨 관리자 상태 데이터베이스 복제본(mediates)을 만들 수 있습니다. 프로필 파일에서 metadb 키워드를 여러 번 사용하여 여러 디스크 슬라이스에 상태 데이터베이스 복제본을 만들 수 있습니다.

slice 사용자 정의 JumpStart 프로그램에서 상태 데이터베이스 복제본을 두려는 디스크 슬라이스를 지정해야 합니다. slice 값은 cwtxdysz 형식을 따라야 합니다.

size size-in-blocks size 옵션 키워드를 사용하면 만들려는 상태 데이터베이스 복제본의 크기를 블록 단위로 지정할 수 있습니다. size 값을 지정하지 않으면 사용자 정의 JumpStart 프로그램은 상태 데이터베이스 복제본에 기본값인 8192 블록 크기를 사용합니다.

count number-of-replicas 프로필에 선택적인 *count* 키워드 값을 설정하여 만들려는 상태 데이터베이스 복제본 수를 지정할 수 있습니다. *count* 값을 지정하지 않으면 사용자 정의 JumpStart 프로그램은 기본적으로 세 개의 상태 데이터베이스 복제본을 만듭니다.

설치 중에 Solaris 볼륨 관리자 상태 데이터베이스 복제본 작성에 대한 자세한 내용은 201 페이지 “상태 데이터베이스 복제본 지침 및 요구 사항”을 참조하십시오.

no_content_check 프로필 키워드(Solaris Flash 아카이브 설치)

no_content_check

복제 시스템을 Solaris Flash 차등 아카이브와 함께 설치할 경우, *no_content_check* 키워드를 사용하여 파일별 검증을 무시할 수 있습니다. 파일별 검증은 복제 시스템이 마스터 시스템의 복제본임을 확인합니다. 복제 시스템이 원본 마스터 시스템의 복제본인지 확실하지 않으면 이 키워드를 사용하지 마십시오.



주의 - *no_content_check*를 사용하는 경우, 예상 상태로 복제 시스템을 가져오기 위해 새로운 모든 파일이 삭제됩니다. 파일을 삭제할지 여부를 결정하지 않은 경우, 설치를 중단하여 새 파일을 보호하는 기본값을 사용합니다.

Solaris Flash 차등 아카이브 설치에 대한 정보는 122 페이지 “사용자 정의 JumpStart 설치로 Solaris Flash 아카이브 설치 준비하기”를 참조하십시오.

no_master_check 프로필 키워드(Solaris Flash 아카이브 설치)

no_master_check

Solaris Flash 차등 아카이브를 사용해 복제 시스템을 설치할 경우 복제 시스템 확인을 무시하는 *no_master_check* 키워드를 사용하여 원본 마스터 시스템에서 구축되었음을 확인할 수 있습니다. 복제 시스템이 원본 마스터 시스템의 복제본인지 확실하지 않으면 이 키워드를 사용하지 마십시오.

Solaris Flash 차등 아카이브 설치에 대한 정보는 122 페이지 “사용자 정의 JumpStart 설치로 Solaris Flash 아카이브 설치 준비하기”를 참조하십시오.

num_clients 프로필 키워드

num_clients client_num

서버가 설치된 경우 공간은 각 디스크 없는 클라이언트의 루트(/) 및 swap 파일 시스템에 대하여 할당됩니다. *num_clients*는 디스크 없는 서버가 지원하는 클라이언트의 개수인 *client_num*을 정의합니다. 프로필에서 *num_clients*를 지정하지 않으면 다섯 개의 디스크 없는 클라이언트가 기본 할당됩니다.

주-`num_clients`는 `system_type`이 `server`로 지정된 경우에만 사용할 수 있습니다.

package 프로필 키워드

`package package_name [add [retrieval_type location] | delete]`

`package`는 초기 설치와 업그레이드 옵션 모두에서 사용할 수 있습니다. `package` 키워드를 사용하여 다음과 같이 할 수 있습니다.

- 설치될 Solaris 배포에서 소프트웨어 그룹에 패키지를 추가합니다.
- 설치될 외부 배포에서 소프트웨어 그룹에 패키지를 추가합니다.
- 설치 또는 업그레이드될 소프트웨어 그룹에서 패키지를 제외 또는 제거합니다.
- Solaris Flash 아카이브에서 설치할 경우 설치되는 배포 외부의 패키지를 추가합니다.

`package_name` SUNW *name* 형식으로 패키지 이름을 지정합니다. 패키지 이름에 대한 자세한 내용은 설치된 시스템에서 `pkginfo -l` 명령어를 사용합니다.

`add | delete` 지정된 패키지를 추가 또는 제거하도록 지정합니다. `add` 또는 `delete`를 지정하지 않으면 `add`가 기본값으로 사용됩니다.

주-프로필에 다른 패키지 항목을 추가하고 위치를 생략하여 하나 이상의 패키지를 추가할 수 있습니다. 이전 패키지의 위치는 위치가 공백으로 남을 경우 모든 이후 패키지에 대하여 사용됩니다.

`[retrieval_type location]` 설치되는 Solaris 배포 외부에 위치한 패키지의 추가를 지정합니다. `retrieval_type` 및 `location`의 값은 패키지가 저장되는 위치에 의존합니다. 다음 절에서는 `retrieval_type` 및 `location`에 대하여 사용할 수 있는 값과 `package_name` 키워드 사용 방법의 예를 설명합니다.

주-비전역 영역이 설치된 경우 이 키워드를 사용하여 업그레이드하지 마십시오. 이 키워드를 사용하면 업그레이드는 계속되지만 키워드가 무시됩니다.

NFS 서버에 저장된 패키지

패키지가 NFS 서버에 저장되면 `package` 키워드에 대하여 다음 구문 중 하나를 사용합니다.

```
package package_name add nfs server_name:/path [retry n]
package package_name add nfs://server_name:/path [retry n]
```

`package_name` SUNW *name* 형식으로 패키지 이름을 지정합니다. 패키지 이름에 대한 자세한 내용은 설치된 시스템에서 `pkginfo -l` 명령어를 사용합니다.

<i>server_name</i>	패키지를 저장한 서버의 이름을 지정합니다.
<i>path</i>	지정된 서버의 패키지 디렉토리의 위치를 지정합니다.경로에 \$HOST가 포함되면 \$HOST는 설치 중인 호스트 시스템의 이름으로 대체됩니다.
<i>retry n</i>	선택적 키워드입니다. <i>n</i> 은 설치 프로세스가 디렉토리 마운트를 시도하는 최대 횟수입니다.

예 11-12 NFS를 사용하여 패키지 추가

이 예에서 package 프로필 키워드는 NFS 위치 `nfs://golden/packages/Solaris_10_606/` 에서 SUNWnew 패키지를 추가합니다. 마운트가 실패하면 NFS 마운트는 다섯 번 시도됩니다.

```
package SUNWnew add nfs golden:/packages/Solaris_10_6/06 retry 5
```

HTTP 서버에 저장된 패키지

패키지가 HTTP 서버에 저장되면 package 키워드에 대하여 다음 구문 중 하나를 사용합니다.

```
package package_name add http://server_name[:port] path optional_keyword
package package_name add http server_name[:port] path optional_keyword
```

<i>package_name</i>	SUNW name 형식으로 패키지 이름을 지정합니다. 패키지와 패키지 이름에 대한 자세한 내용은 설치된 시스템에서 <code>pkginfo -l</code> 명령어를 사용합니다.
<i>server_name</i>	패키지를 저장한 서버의 이름을 지정합니다.
<i>port</i>	선택적 포트를 지정합니다. <i>port</i> 는 포트 번호 또는 런타임에 결정된 포트 번호를 가진 TCP 서비스의 이름일 수 있습니다. 포트를 지정하지 않으면 기본 HTTP 포트 번호 80이 사용됩니다.
<i>path</i>	지정된 서버에서 검색될 패키지의 위치를 지정합니다. HTTP 서버를 사용할 경우 패키지는 패키지 데이터스트림 형식이어야 합니다.
<i>optional_keywords</i>	HTTP 서버에서 패키지를 검색할 경우 사용할 선택적 키워드를 지정합니다.

표 11-6 HTTP와 함께 사용할 선택적 package 키워드

키워드	값 정의
<i>timeout min</i>	<i>timeout</i> 키워드를 사용하여 HTTP 서버에서 데이터를 받지 않고 전달할 수 있는 최고 시간 길이(분 단위)를 지정할 수 있습니다. 시간 초과가 발생할 경우 연결이 종료되고 다시 열리고 재시작됩니다. <i>timeout</i> 값을 0으로 지정하면 연결이 다시 열리지 않습니다. 시간 초과 재연결이 발생하면 시간 초과가 무시되기 전에 검색된 패키지와 데이터의 시작부터 패키지가 재시도됩니다.

표 11-6 HTTP와 함께 사용할 선택적 package 키워드 (계속)

키워드	값 정의
<code>proxy host:port</code>	<code>proxy</code> 키워드는 프록시 호스트 및 프록시 포트를 지정할 수 있게 합니다. 방화벽의 맞은 편에서 Solaris 패키지를 검색하기 위해 프록시 호스트를 사용할 수 있습니다. <code>proxy</code> 키워드를 지정할 때는 반드시 프록시 포트를 제공해야 합니다.

예 11-13 HTTP를 사용하여 패키지 추가

이 예에서 package 프로필 키워드는 HTTP 위치

`http://package.central/Solaris_10_606`에서 Solaris_10_606 디렉토리에 나열된 모든 패키지를 추가합니다. 5분이 경과하고 데이터가 검색되지 않으면 패키지 데이터가 다시 검색됩니다. 이전 패키지 데이터는 무시됩니다. 다음 형식 중 하나를 사용할 수 있습니다.

```
package SUNWnew add http package.central/Solaris_10_606 timeout 5
```

```
package SUNWnew add http://package.central/Solaris_10_606 timeout 5
```

예 11-14 프록시 포트가 있는 HTTP를 사용하여 패키지 추가

이 예에서 package 프로필 키워드는 HTTP 위치

`http://package.central/Solaris_10_606`에서 Solaris_10_606 디렉토리에 나열된 모든 패키지를 추가합니다. 패키지는 `proxy` 키워드를 사용하여 방화벽에 걸쳐 검색됩니다.

```
package SUNWnew add http://package.central/Solaris_10_606 proxy webcache.east:8080
```

로컬 장치에 저장된 패키지

디스켓이나 DVD-ROM과 같은 파일 시스템 지향, 임의 액세스 장치에 패키지를 저장한 경우 로컬 장치에서 Solaris 패키지를 검색할 수 있습니다. package 키워드에 다음 구문을 사용합니다.

```
package package_name add local_device device path file_system_type
```

package_name SUNW name 형식으로 패키지 이름을 지정합니다. 패키지 이름에 대한 자세한 내용은 설치된 시스템에서 `pkginfo -l` 명령어를 사용합니다.

device Solaris 패키지가 상주하는 드라이브의 이름을 지정합니다. 장치 이름이 정식 경로라면 해당 장치는 바로 마운트됩니다. 정식 경로가 아닌 장치 이름을 공급하는 경우 설치 유틸리티는 경로에 `/dev/dsk/`를 추가합니다.

path 지정된 장치의 루트(/) 파일 시스템을 기준으로 Solaris 패키지 경로를 지정합니다.

file_system_type 장치의 파일 시스템 유형을 지정합니다. 파일 시스템 유형을 제공하지 않으면 설치 유틸리티는 UFS 파일 시스템의 마운트를 시도합니다. UFS 마운트에 실패하면 설치 유틸리티는 HFS 파일 시스템의 마운트를 시도합니다.

예 11-15 UFS 파일 시스템을 가진 로컬 장치를 사용하여 패키지 추가

이 예에서 `package` 프로필 키워드는 로컬 장치 `c0t6d0s0`의 디렉토리 `/Solaris_10_606/Product`에서 `SUNWnew` 패키지를 추가합니다. 이는 UFS 파일 시스템입니다.

```
package SUNWnew add local_device c0t6d0s0 /Solaris_10_606/Product ufs
```

예 11-16 HSFS 파일 시스템에서 로컬 장치를 사용하여 패키지 추가

이 예에서 `package` 프로필 키워드는 로컬 장치 `c0t6d0s0`의 디렉토리 `/Solaris_10_606/Product`에서 `SUNWnew` 패키지를 추가합니다. 이는 HSFS 파일 시스템입니다.

```
package SUNWnew add local_device c0t6d0s0 /Solaris_10_606/Product hsfs
```

로컬 파일에 저장된 패키지

패키지는 시스템을 부트한 미니루트에서 설치될 수 있습니다. 사용자 정의 JumpStart 설치를 수행하는 경우 DVD, CD 또는 NFS 기반 미니루트에서 시스템을 부트합니다. 설치 소프트웨어가 로드되고 이 미니루트에서 실행됩니다. 따라서 DVD, CD 또는 NFS 기반 미니루트에 저장한 패키지는 로컬 파일로 액세스할 수 있습니다. `package` 키워드에 다음 구문을 사용합니다.

```
package package_name add local_file path
```

package_name `SUNW name` 형식으로 패키지 이름을 지정합니다. 패키지 이름에 대한 자세한 내용은 설치된 시스템에서 `pkginfo -l` 명령어를 사용합니다.

path 패키지의 위치를 지정합니다. 시스템을 Solaris Software - 1 CD나 Solaris Operating System DVD에서 부트할 때 해당 경로는 로컬 파일로서 시스템에 액세스할 수 있어야 합니다. Solaris Software - 1 CD 또는 Solaris Operating System DVD로부터 부트하는 경우 시스템이 `/net`에 액세스할 수 없습니다.

예 11-17 로컬 파일을 사용하여 패키지 추가

이 예에서 `package` 프로필 키워드는 `/Solaris_10_606/Product` 디렉토리의 `SUNWnew` 패키지를 추가합니다.

```
package SUNWnew add local_file /Solaris_10_606/Product
```

package 키워드를 사용할 경우 제한

`package` 키워드를 사용할 경우 다음 제한을 유의하십시오.

- 일부 패키지들은 필수 패키지이므로 삭제할 수 없습니다.
- `package` 프로필 키워드를 사용하여 현지화 패키지를 개별적으로 추가 또는 삭제할 수 없습니다. 현지화 패키지를 추가하려면 `locale` 프로필 키워드를 사용합니다.

- 패키지는 FTP 서버 위치 또는 테이프와 같은 로컬 백업에서 검색할 수 없습니다.
- 설치되는 Solaris 배포 내부의 패키지는 대체 위치에서 추가될 수 없습니다. Solaris 배포의 패키지가 지정되면 패키지는 결과적으로 설치된 시스템과 일관성을 유지하기 위해 대체 위치 앞에 올 수 없습니다.
- 수동 개입 없이 설치하기 위해서 패키지는 `pkgadd` 명령어를 사용하여 설치 가능해야 합니다. 소프트웨어 그룹 패키지 및 다른 위치에 있는 패키지를 설치하려면 동일한 `admin` 파일을 사용해야 합니다.
 - `retrieval_type`이 HTTP이면 패키지는 스트림 형식이어야 합니다.
 - `retrieval_type`이 NFS 서버, 로컬 장치 또는 로컬 파일이면 패키지는 설치되는 패키지과 동일한 디렉토리 이름을 가진 독립형 패키징 형식을 따라야 합니다.
 - 패키지가 별도의 위치에서 추가되고 패키지가 현재 설치되지 않은 다른 패키지에 의존할 경우 패키지는 설치되지 않습니다. 오류 메시지는 설치 또는 업그레이드 로그 파일에 기록됩니다.
- 패키지가 Solaris Flash 아카이브로 설치되는 경우 다음 지침을 따르십시오.
 - 설치된 모든 패키지는 아카이브와 호환되어야 합니다.
 - 패키지가 아카이브에 있으면 JumpStart는 기존 패키지를 겹쳐씹니다.

package 키워드를 사용할 경우 업그레이드 동작

업그레이드용으로 `package`를 사용하면 JumpStart 프로그램은 다음 활동을 수행합니다.

- 시스템에 이미 있는 모든 패키지가 자동으로 업그레이드됩니다.
- `package_name add`를 지정했는데 `package_name`이 시스템에 설치되어 있지 않으면 해당 패키지가 설치됩니다.
- `package_name delete`를 지정했는데 `package_name`이 시스템에 설치되어 있으면 해당 패키지는 업그레이드 전에 삭제됩니다.
- `package_name delete`를 지정하고 `package_name`이 시스템에 설치되어 있지 않으면, 패키지가 설치되도록 지정된 클러스터의 일부인 경우 해당 패키지는 설치되지 않습니다.

partitioning 프로필 키워드

`partitioning type`

`partitioning`은 설치 중에 디스크가 파일 시스템의 슬라이스로 분할되는 방법을 정의합니다.

프로필에서 `partitioning`을 지정하지 않으면 기본적으로 `default` 분할 유형이 사용됩니다.

`type` 다음 값 중 하나를 사용합니다.

<code>default</code>	JumpStart 프로그램은 디스크를 선택해 <code>filesystem</code> 키워드에 의해 지정된 파일 시스템을 제외하고 지정된 소프트웨어를 설치할 파일 시스템을 작성합니다. <code>rootdisk</code> 가 먼저 선택됩니다. JumpStart 프로그램은 지정된 소프트웨어가 <code>rootdisk</code> 에 적합하지 않은 경우 추가 디스크를 사용합니다.
----------------------	--

existing JumpStart 프로그램은 시스템 디스크의 기존 파일 시스템을 사용합니다. /, /usr, /usr/openwin, /opt 및 /var을 제외한 모든 파일 시스템은 보존됩니다. JumpStart 프로그램은 파일 시스템 수퍼블록에서 마지막 마운트 지점 필드를 사용하여 슬라이스가 나타내는 파일 시스템 마운트 지점이 무엇인지 결정합니다.

주 - **filesys** 및 **partitioning existing** 프로필 키워드를 사용할 때는 **size**를 **existing**으로 설정해야 합니다.

explicit JumpStart 프로그램은 디스크를 사용하고 **filesys** 키워드에 의해 지정된 파일 시스템을 작성합니다. **filesys** 키워드를 통해 루트(/) 파일 시스템을 지정하면 Solaris 소프트웨어 전체는 루트(/) 파일 시스템에 설치됩니다.

주 - **explicit** 프로필 값을 사용하면 **filesys** 키워드를 이용해 사용할 디스크와 만들 파일 시스템을 지정해야 합니다.

patch 프로필 키워드

`patch patch_id_list | patch_file patch_location optional_keywords]`

patch_id_list 설치될 패치 ID 번호를 지정합니다. 목록은 쉼표 구분 Solaris 패치 ID의 목록이어야 합니다. 패치는 목록에 지정된 순서로 설치됩니다. 쉼표 뒤에 공간을 추가하지 마십시오. 예를 들면 다음과 같습니다.
112467-01,112765-02.

patch_file **patch_location**에 있는 패치 목록을 가진 파일입니다. 패치는 파일에 지정된 순서로 설치됩니다.

patch_location 패치가 상주하는 위치를 지정합니다. 허용된 위치는 다음과 같습니다.

- NFS 서버
- HTTP 서버
- 로컬 장치
- 로컬 파일

optional_keywords 선택적 키워드는 패치가 저장된 위치에 의존합니다. 다음 절은 가능한 위치 및 선택적 키워드에 대해 설명합니다.

주 - 비전역 영역이 설치된 경우 이 키워드를 사용하여 업그레이드하지 마십시오. 이 키워드를 사용하면 업그레이드는 계속되지만 키워드가 무시됩니다.

NFS 서버에 저장된 패치

패치가 NFS 서버에 저장되면 **patch** 키워드에 대하여 다음 구문 중 하나를 사용합니다.

<code>patch patch_id_list</code>	<code>patch_file</code> nfs server_name:/patch_directory [retry n]
<code>patch patch_id_list</code>	<code>patch_file</code> nfs://server_name/patch_director [retry n]
<code>patch_id_list</code>	설치될 패치 ID 번호를 지정합니다. 목록은 쉼표 구분 Solaris 패치 ID의 목록이어야 합니다. 패치는 목록에 지정된 순서로 설치됩니다.
<code>patch_file</code>	<code>patch_location</code> 에 있는 패치 목록을 가진 파일입니다. 패치는 파일에 지정된 순서로 설치됩니다.
<code>server_name</code>	패치를 저장한 서버의 이름을 지정합니다.
<code>patch_directory</code>	지정된 서버의 패치 디렉토리의 위치를 지정합니다. 패치는 독립형 패치 형식이어야 합니다.
<code>retry n</code>	선택적 키워드입니다. <i>n</i> 은 설치 유틸리티가 디렉토리 마운트를 시도하는 최대 횟수입니다.

예 11-18 NFS를 사용하여 순서 지정 목록을 가진 패치 추가

이 예에서 `patch` 프로필 키워드는 NFS 패치 디렉토리 `nfs://patch_master/Solaris/v10_606/patches`의 `patch` 파일에 나열된 모든 패치를 추가합니다. 패치는 `patch`에 나열된 순서로 설치됩니다. 마운트가 실패하면 NFS 마운트는 다섯 번 시도됩니다.

```
patch patch_file nfs://patch_master/Solaris/v10_606/patches retry 5
```

예 11-19 NFS를 사용하여 패치 추가

이 예에서 `patch` 프로필 키워드는 서버 `patch_master`의 패치 디렉토리 `/Solaris/v10_606/patches`에서 패치 112467-01 및 112765-02를 추가합니다.

```
patch 112467-01,112765-02 nfs patch_master:/Solaris/v10_606/patches
```

HTTP 서버에 저장된 패치

패치가 HTTP 서버에 저장되면 `patch` 키워드에 대하여 다음 구문 중 하나를 사용합니다.

```
patch patch_id_list | patch_file http://server_name [:port] patch_directory optional_http_keywords
```

<code>patch patch_id_list</code>	<code>patch_file</code> http server_name [:port] patch_directory optional_http_keywords
<code>patch_id_list</code>	설치될 패치 ID 번호를 지정합니다. 목록은 쉼표 구분 Solaris 패치 ID의 목록이어야 합니다. 패치는 목록에 지정된 순서로 설치됩니다. 쉼표 뒤에 공간을 추가하지 마십시오. 예를 들면 다음과 같습니다. 112467-01,112765-02.
<code>patch_file</code>	<code>patch_location</code> 에 있는 패치 목록을 가진 파일입니다. 패치는 파일에 지정된 순서로 설치됩니다.

<i>server_name</i>	패치를 저장한 서버의 이름을 지정합니다.
<i>port</i>	선택적 포트를 지정합니다. <i>port</i> 는 포트 번호 또는 런타임에 결정된 포트 번호를 가진 TCP 서비스의 이름일 수 있습니다. 포트를 지정하지 않으면 기본 HTTP 포트 번호 80이 사용됩니다.
<i>patch_directory</i>	지정된 서버에서 검색될 패치 디렉토리의 위치를 지정합니다. HTTP 서버를 사용할 때 패치는 JAR 형식이어야 합니다.
<i>optional_keywords</i>	HTTP 서버에서 패치를 검색할 경우 사용할 선택적 키워드를 지정합니다.

표 11-7 HTTP와 함께 사용할 선택적 patch 키워드

키워드	값 정의
<i>timeout min</i>	<i>timeout</i> 키워드를 사용하여 HTTP 서버에서 데이터를 받지 않고 전달할 수 있는 최고 시간 길이(분 단위)를 지정할 수 있습니다. 시간 초과가 발생할 경우 연결이 종료되고 다시 열리고 재시작됩니다. <i>timeout</i> 값을 0으로 지정하면 연결이 다시 열리지 않습니다. 시간 초과 재연결이 발생하면 시간 초과가 무시되기 전에 검색된 패키지과 데이터의 시작부터 패키지가 재시도됩니다.
<i>proxy host:port</i>	<i>proxy</i> 키워드는 프록시 호스트 및 프록시 포트를 지정할 수 있게 합니다. 방화벽의 맞은 편에서 Solaris 패키지를 검색하기 위해 프록시 호스트를 사용할 수 있습니다. <i>proxy</i> 키워드를 지정할 때는 반드시 프록시 포트를 제공해야 합니다.

예 11-20 HTTP를 사용하여 순서 지정 목록을 가진 패치 추가

이 예에서 *patch* 프로필 키워드는 HTTP 위치 `http://patch.central/Solaris/v10_606/patches`에서 *patch_file* 파일에 나열된 모든 패치를 추가합니다. 패치는 *patch* 파일에 지정된 순서로 설치됩니다. 5분이 경과하고 데이터가 검색되지 않으면 패치 데이터가 다시 검색됩니다. 이전 패치 데이터는 무시됩니다.

```
patch patch_file http://patch.central/Solaris/v10_606/patches timeout 5
```

예 11-21 HTTP를 사용하여 패치 추가

이 예에서 *patch* 프로필 키워드는 패치 위치 `http://patch_master/Solaris/v10_606/patches`에서 패치 112467-01 및 112765-02를 추가합니다.

```
patch 112467-01,112765-02 http://patch.central/Solaris/v10_606/patches
```

로컬 장치에 저장된 패치

디스켓이나 DVD-ROM과 같은 파일 시스템 지향, 임의 액세스 장치에 패키지를 저장한 경우 로컬 장치에서 Solaris 패키지를 검색할 수 있습니다. *patch* 키워드에 다음 구문을 사용합니다.

```
patch patch_id_list | patch_file local_device \
device path file_system_type
```

<i>patch_id_list</i>	설치될 패치 ID 번호를 지정합니다. 목록은 쉼표 구분 Solaris 패치 ID의 목록이어야 합니다. 패치는 목록에 지정된 순서로 설치됩니다. 쉼표 뒤에 공간을 추가하지 마십시오. 예를 들면 다음과 같습니다. 112467-01,112765-02.
<i>patch_file</i>	<i>patch_location</i> 에 있는 패치 목록을 가진 파일입니다. 패치는 파일에 지정된 순서로 설치됩니다.
<i>device</i>	Solaris 패키지가 상주하는 드라이브의 이름을 지정합니다. 장치 이름이 정식 경로라면 해당 장치는 바로 마운트됩니다. 정식 경로가 아닌 장치 이름을 공급하는 경우 설치 유틸리티는 경로에 /dev/dsk/를 추가합니다.
<i>path</i>	지정한 장치의 루트(/) 파일 시스템을 기준으로 Solaris 패치 경로를 지정합니다.
<i>file_system_type</i>	장치의 파일 시스템 유형을 지정합니다. 파일 시스템 유형을 제공하지 않으면 설치 유틸리티는 UFS 파일 시스템의 마운트를 시도합니다. UFS 마운트에 실패하면 설치 유틸리티는 HSFS 파일 시스템의 마운트를 시도합니다.

예 11-22 로컬 장치를 사용하여 순서 지정 목록을 가진 패치 추가

이 예에서 patch 프로필 키워드는 로컬 장치 c0t6d0s0의 디렉토리 /Solaris_10_606/patches에 있는 patch_file 파일에 나열된 모든 패치를 추가합니다. patch 파일은 설치될 패치의 순서를 결정합니다.

```
patch patch_file c0t6d0s0 /Solaris_10_606/patches
```

예 11-23 로컬 장치를 사용하여 패치 추가

이 예에서 patch 프로필 키워드는 로컬 장치 c0t6d0s0의 패치 디렉토리 /Solaris_10_606/patches에서 패치 112467-01 및 112765-02를 추가합니다.

```
patch 112467-01,112765-02 local_device c0t6d0s0 /Solaris_10_606/patches
```

로컬 파일에 저장된 패치

패치는 시스템을 부트한 미니루트에서 설치될 수 있습니다. 사용자 정의 JumpStart 설치를 수행하는 경우 DVD, CD 또는 NFS 기반 미니루트에서 시스템을 부트합니다. 설치 소프트웨어가 로드되고 이 미니루트에서 실행됩니다. 따라서 DVD, CD 또는 NFS 기반 미니루트에 저장한 패키지에는 로컬 파일로 액세스할 수 있습니다. patch 키워드에 다음 구문을 사용합니다.

```
patch patch_id_list | patch_file local_file patch_directory
```

<i>patch_id_list</i>	설치될 패치 ID 번호를 지정합니다. 목록은 쉘표 구분 Solaris 패치 ID의 목록이어야 합니다. 패치는 목록에 지정된 순서로 설치됩니다. 쉘표 뒤에 공간을 추가하지 마십시오. 예를 들면 다음과 같습니다. 112467-01,112765-02.
<i>patch_file</i>	<i>patch_location</i> 에 있는 패치 목록을 가진 파일입니다. 패치는 파일에 지정된 순서로 설치됩니다.
<i>patch_directory</i>	패치 디렉토리의 위치를 지정합니다. 시스템을 Solaris Software - 1 CD나 Solaris Operating System DVD에서 부트할 때 패치 디렉토리는 로컬 파일로서 시스템에 액세스할 수 있어야 합니다. Solaris Software - 1 CD 또는 Solaris Operating System DVD로부터 부트하는 경우 시스템이 /net에 액세스할 수 없습니다.

예 11-24 로컬 파일을 사용하여 순서 지정 목록을 가진 패치 추가

이 예에서 patch 프로필 키워드는 /Solaris_10_606/patches 디렉토리에서 patch_file 파일에 나열된 모든 패치를 추가합니다. patch 파일은 설치될 패치의 순서를 결정합니다

```
patch patch_cal_file /Solaris_10_606/patches
```

예 11-25 로컬 파일을 사용하여 패치 추가

이 예에서 patch 프로필 키워드는 패치 디렉토리 /Solaris_10_606/patches에서 패치 112467-01 및 112765-02를 추가합니다.

```
patch 112467-01,112765-02 local_file /Solaris_10_606/patches
```

patch 키워드를 사용할 경우 제한

patch 키워드를 사용할 경우 다음 제한을 유의하십시오.

- 패치는 FTP 위치 또는 테이프와 같은 로컬 백업에서 검색할 수 없습니다.
- 부호있는 패치는 추가할 수 없습니다.
- 패치는 patchadd 명령어로 설치 가능해야 합니다.
- 패치가 현재 설치되지 않은 패치에 의존하면 패치는 설치되지 않습니다. 오류 메시지는 설치 또는 업그레이드 로그 파일에 기록됩니다.
- 패치의 올바른 설치를 위해 올바른 패치 순서를 결정해야 합니다.

root_device 프로필 키워드

root_device slice

root_device는 시스템의 루트디스크를 지정합니다. 185 페이지 “시스템의 루트디스크 결정 방법”에서는 추가 정보를 포함합니다.

시스템을 업그레이드할 때 `root_device`는 루트(/) 파일 시스템 및 `/etc/vfstab` 파일에 의해 마운트되는 시스템을 업그레이드하도록 지정합니다. 한 대의 시스템에서 두 대 이상의 루트(/) 파일 시스템을 업그레이드할 수 있다면 `root_device`를 지정해야 합니다. `slice`는 `cwtxdysz` 또는 `cxdysz` 형식으로 지정해야 합니다.

`root_device` 키워드를 사용할 때 다음 사항을 고려하십시오.

- 디스크가 하나 뿐인 시스템에서 `root_device`를 지정하면 `root_device`와 디스크가 일치해야 합니다. 또한, 루트(/) 파일 시스템을 지정하는 `filesys` 키워드는 `root_device`를 일치시켜야 합니다.
- RAID-1 볼륨(미러)을 업그레이드하는 경우에는 `root_device`에 지정된 값이 미러의 한 쪽이어야 합니다. 미러의 다른 쪽은 자동으로 업그레이드됩니다.

예 11-26 `root_device` 프로필 키워드

```
root_device c0t0d0s2
```

시스템의 루트 디스크 결정 방법

시스템의 루트 디스크는 루트(/) 파일 시스템이 포함된 시스템의 디스크입니다. 프로필에서 디스크 이름 대신 JumpStart 프로그램이 시스템의 루트 디스크에 설정하는 `rootdisk` 변수를 사용할 수 있습니다. 표 11-8은 JumpStart 프로그램이 설치를 위해 시스템의 루트 디스크를 결정하는 방법에 대해 설명합니다.

주 - JumpStart 프로그램은 초기 설치 중에 시스템의 루트 디스크 크기만을 결정합니다. 업그레이드 중에는 시스템의 루트 디스크를 변경할 수 없습니다.

표 11-8 JumpStart의 시스템 루트 디스크 결정 방법(초기 설치)

단계	작업
1	<code>root_device</code> 키워드가 프로필에 정의되어 있으면 JumpStart 프로그램은 <code>rootdisk</code> 를 루트 장치에 설정합니다.
2	<code>rootdisk</code> 가 설정되어 있지 않고 <code>boot_device</code> 키워드가 프로필에 지정되어 있으면 JumpStart 프로그램은 <code>rootdisk</code> 를 부트 장치로 설정합니다.
3	If <code>rootdisk</code> 가 설정되어 있지 않고 프로필에 <code>filesys c wtxd ysz size /</code> 항목이 지정되어 있으면 JumpStart 프로그램은 <code>rootdisk</code> 를 항목에 지정되어 있는 디스크로 설정합니다.
4	<code>rootdisk</code> 가 설정되지 않고 <code>rootdisk.sn</code> 항목이 프로필에 지정된 경우 JumpStart 프로그램은 지정된 슬라이스의 기존 루트 파일 시스템에 대한 커널 검사 순서로 시스템의 디스크를 검색합니다. 디스크를 찾으면 JumpStart 프로그램은 <code>rootdisk</code> 를 찾은 디스크로 설정합니다.

표 11-8 JumpStart의 시스템 루트 디스크 결정 방법(초기 설치) (계속)

단계	작업
5	rootdisk가 설정되어 있지 않고 프로필에 <code>partitioning existing</code> 항목이 지정되어 있으면 JumpStart 프로그램이 커널 검사 순서로 시스템의 디스크에서 기존 루트 파일 시스템이 있는지 검색합니다. 루트 파일 시스템이 확인되지 않거나 둘 이상이 확인되면 오류가 발생합니다. 루트 파일 시스템이 확인되면 JumpStart 프로그램은 rootdisk를 확인된 디스크로 설정합니다.
6	rootdisk가 설정되어 있지 않으면 JumpStart 프로그램은 rootdisk를 루트(/) 파일 시스템이 설치되어 있는 디스크로 설정합니다.

system_type 프로필 키워드

system_type *type_switch*

system_type은 Solaris OS가 설치될 시스템 유형을 정의합니다.

*type_switch*는 `standalone` 또는 `server` 옵션을 나타내며 이 옵션들은 사용자가 Solaris 소프트웨어를 설치할 시스템의 유형을 지정하기 위해 사용합니다. 프로필에서 system_type을 지정하지 않으면 `standalone`이 기본값으로 사용됩니다.

usedisk 프로필 키워드

usedisk *disk_name* ...

기본적으로, `partitioning default`를 지정하면 JumpStart 프로그램은 시스템에서 작동 가능한 디스크 모두를 사용합니다. `usedisk` 프로필 키워드는 JumpStart 프로그램이 사용하는 하나 이상의 디스크를 지정합니다. *disk_name*은 `cxydz` 또는 `cydz` 형식으로 지정해야 합니다(예: `c0t0d0` 또는 `c0d0s0`).

프로필에 `usedisk`를 지정하면 JumpStart 프로그램은 `usedisk` 키워드 다음에 지정한 디스크만을 사용합니다.

주 - `usedisk` 키워드와 `dontuse` 키워드는 동일한 프로필에서 지정할 수 없습니다.

비전역 영역으로 업그레이드하는 경우 프로필 키워드 제한

비전역 영역이 설치된 경우 사용자 정의 JumpStart 프로그램을 업그레이드에 사용할 수 있습니다. 프로필에는 `install_type` 및 `root_device` 키워드의 두 가지 프로필 키워드만 사용할 수 있습니다.

일부 키워드는 비전역 영역에 영향을 미치지 때문에 키워드 중 일부는 프로필에 포함할 수 없습니다. 예를 들어, 패키지 추가, 디스크 공간 재할당 또는 로케일 추가를 수행하는 키워드를 사용하면 비전역 영역에 영향을 줍니다. 비전역 영역에 악영향을 주는 키워드를 사용하는 경우 이 키워드는 무시되거나 업그레이드 완료를 방해합니다. 프로필에 사용하면 안 되는 키워드의 목록은 다음 표를 참조하십시오.

표 11-9 비전역 영역에서 업그레이드 시 오류가 발생하는 키워드

프로필 키워드	업그레이드 동작
backup_media	이 키워드를 사용하면 업그레이드가 중지되고 오류 메시지가 표시됩니다.
cluster	이 키워드는 무시되고 업그레이드가 계속됩니다.
geo	이 키워드는 무시되고 업그레이드가 계속됩니다.
layout_constraint	이 키워드를 사용하면 업그레이드가 중지되고 오류 메시지가 표시됩니다.
locale	이 키워드는 무시되고 업그레이드가 계속됩니다.
package	이 키워드는 무시되고 업그레이드가 계속됩니다.
patch	이 키워드는 무시되고 업그레이드가 계속됩니다.

비전역 영역에 대한 자세한 내용은 다음을 참조하십시오.

- 43 페이지 “시스템에 Solaris 영역이 설치된 경우 업그레이드”
- **System Administration Guide: Solaris Containers-Resource Management and Solaris Zones**의 16 장, “Introduction to Solaris Zones”

사용자 정의 JumpStart 환경 변수

시작 및 종료 스크립트에서 환경 변수를 사용할 수 있습니다. 예를 들어, 시작 스크립트는 디스크 크기, `SI_DISK크기S`를 추출할 수 있으며 스크립트가 추출하는 실제 디스크 크기를 기준으로 시스템에 특정 패키지를 설치하거나 설치하지 않습니다.

시스템에 관해 수집된 정보는 `rules` 파일에서 사용하는 규칙 키워드 및 값에 따라 설정되거나 설정되지 않은 이들 환경 변수에 저장됩니다.

예를 들어, 시스템에 설치된 운영 체제에 관한 정보는 `installed` 키워드를 사용한 뒤 `SI_INSTALLED`에서만 사용 가능합니다.

표 11-10에서는 이러한 변수와 값을 설명합니다.

표 11-10 설치 환경 변수

환경 변수	값
SI_ARCH	설치 클라이언트의 하드웨어 구조. SI_ARCH 변수는 <code>rules</code> 파일에서 <code>arch</code> 키워드를 사용할 때 설정됩니다.
SI_BEGIN	시작 스크립트를 사용할 경우의 해당 스크립트 이름

표 11-10 설치 환경 변수 (계속)

환경 변수	값
SI_CLASS	설치 클라이언트를 설치하기 위해 사용되는 프로필 이름
SI_DISKLIST	설치 클라이언트의 씬프로 구분된 디스크 이름 목록. SI_DISKLIST 변수는 disksize 키워드가 사용되고 rules 파일에서 비교될 때 설정됩니다. SI_DISKLIST 및 SI_NUMDISKS 변수는 rootdisk용으로 사용할 물리적 디스크를 결정하는 데 사용됩니다. rootdisk는 185 페이지 “시스템의 루트 디스크 결정 방법”에서 설명합니다.
SI_DISKSIZE	설치 클라이언트의 씬프로 구분된 디스크 크기 목록. SI_DISKSIZE 변수는 disksize 키워드가 사용되고 rules 파일에서 비교될 때 설정됩니다.
SI_DOMAINNAME	도메인 이름. SI_DOMAINNAME 변수는 domainname 키워드가 사용되고 rules 파일에서 비교될 때 설정됩니다.
SI_FINISH	종료 스크립트를 사용할 경우의 해당 스크립트 이름
SI_HOSTADDRESS	설치 클라이언트의 IP 주소
SI_HOSTNAME	설치 클라이언트의 호스트 이름. SI_HOSTNAME 변수는 hostname 키워드가 사용되고 rules 파일에서 비교될 때 설정됩니다.
SI_INSTALLED	Solaris, SunOS 또는 System V와 같은 특정 운영 체제가 설치된 디스크의 장치 이름. SI_INSTALLED 변수는 installed 키워드가 사용되고 rules 파일에서 비교될 때 설정됩니다. SI_INST_OS 및 SI_INST_VER은 SI_INSTALLED의 값을 결정하기 위해 사용됩니다.
SI_INST_OS	운영 체제 이름. SI_INST_OS 및 SI_INST_VER은 SI_INSTALLED의 값을 결정하기 위해 사용됩니다.
SI_INST_VER	운영 체제 버전. SI_INST_OS 및 SI_INST_VER은 SI_INSTALLED의 값을 결정하기 위해 사용됩니다.
SI_KARCH	설치 클라이언트의 커널 구조. SI_KARCH 변수는 karch 키워드가 사용되고 rules 파일에서 비교될 때 설정됩니다.
SI_MEMSIZE	설치 클라이언트의 물리적 메모리 양. SI_MEMSIZE 변수는 memsize 키워드가 사용되고 rules 파일에서 비교될 때 설정됩니다.
SI_MODEL	설치 클라이언트의 모델 이름. SI_MODEL 변수는 model 키워드가 사용되고 rules 파일에서 비교될 때 설정됩니다.
SI_NETWORK	설치 클라이언트의 네트워크 번호. SI_NETWORK 변수는 network 키워드가 사용되고 rules 파일에서 비교될 때 설정됩니다.
SI_NUMDISKS	설치 클라이언트의 디스크 번호. SI_NUMDISKS 변수는 disksize 키워드가 사용되고 rules 파일에서 비교될 때 설정됩니다. SI_NUMDISKS 및 SI_DISKLIST 변수는 rootdisk용으로 사용할 물리적 디스크를 결정하는 데 사용됩니다. rootdisk는 185 페이지 “시스템의 루트 디스크 결정 방법”에서 설명합니다.
SI_OSNAME	Solaris 소프트웨어 이미지의 운영 체제 릴리스. 예를 들어, Solaris 소프트웨어를 Solaris Operating System DVD 또는 Solaris Software - 1 CD 이미지의 운영 체제 버전에 바탕을 둔 시스템에 설치할 경우 스크립트의 SI_OSNAME 변수를 사용할 수 있습니다.

표 11-10 설치 환경 변수 (계속)

환경 변수	값
SI_ROOTDISK	논리 이름 rootdisk로 표현되는 디스크의 장치 이름. SI_ROOTDISK 변수는 rules 파일에서 disksize 또는 installed 키워드가 rootdisk로 설정될 때 설정됩니다.
SI_ROOTDISKSIZE	논리 이름 rootdisk로 표현되는 디스크의 크기. SI_ROOTDISKSIZE 변수는 rules 파일에서 disksize 또는 installed 키워드가 rootdisk로 설정될 때 설정됩니다.
SI_TOTALDISK	설치 클라이언트의 디스크 공간 전체 용량. SI_TOTALDISK 변수는 totaldisk 키워드가 사용되고 rules 파일에서 비교될 때 설정됩니다.

검사 키워드 및 값

표 11-11은 각 규칙 키워드와 동일한 기능을 하는 검사 키워드에 대해 설명합니다.

주 - probe 키워드는 rules 파일에 또는 그 가까이에 배치하십시오.

표 11-11 검사 키워드 설명

규칙 키워드	해당 검사 키워드	검사 키워드 설명
any	없음	
arch	arch	커널 구조, i386 또는 SPARC를 결정하고 SI_ARCH를 설정합니다.
disksize	디스크	커널 검사 순서(c0t3d0s0, c0t3d0s1, c0t4d0s0)에 따라 시스템의 디스크 크기를 MB 단위로 반환합니다. disksize는 SI_DISKLIST, SI_DISKSIZE, SI_NUMDISKS 및 SI_TOTALDISK를 설정합니다.
domainname	domainname	시스템의 NIS 또는 NIS+ 도메인 이름 또는 공백을 반환하고 SI_DOMAINNAME을 설정합니다. domainname 키워드는 domainname(1M)의 출력을 반환합니다.
hostaddress	hostaddress	ifconfig(1M) -a의 출력값에 나열되어 있고 lo0이 아닌 첫 번째 주소인 시스템의 IP 주소를 반환하고 SI_HOSTADDRESS를 설정합니다.
hostname	hostname	시스템의 호스트 이름(uname(1) -n의 출력값)을 반환하고 SI_HOSTNAME을 설정합니다.
installed	installed	시스템에 설치되어 있는 Solaris OS의 버전 이름을 반환하고 SI_ROOTDISK 및 SI_INSTALLED를 설정합니다. JumpStart 프로그램이 Solaris 릴리스를 찾지만 버전을 파악할 수 없는 경우, 반환되는 버전은 SystemV입니다.
karch	karch	시스템의 플랫폼 그룹(예: i86pc 또는 sun4u)을 반환하고 SI_KARCH를 설정합니다. 플랫폼 이름 목록은 http://docs.sun.com 의 Solaris Sun 하드웨어 플랫폼 안내서를 참조하십시오.
memsize	memsize	시스템의 물리적 메모리 크기를 MB 단위로 반환하고 SI_MEMSIZE를 설정합니다.

표 11-11 검사 키워드 설명 (계속)

규칙 키워드	해당 검사 키워드	검사 키워드 설명
model	model	시스템의 플랫폼 이름을 반환하고 SI_MODEL 을 설정합니다. 플랫폼 이름 목록은 http://docs.sun.com 의 Solaris Sun 하드웨어 플랫폼 안내서를 참조하십시오.
network	network	JumpStart 프로그램이 시스템의 IP 주소와 서브넷 마스크 사이에 논리적 AND를 수행함으로써 결정하는 시스템의 네트워크 번호를 반환합니다. 시스템의 IP 주소 및 서브넷 마스크는 <code>ifconfig(1M) -a</code> 의 출력값에 나열되어 있는 <code>lo0</code> 이 아닌 첫 번째 주소에서 추출됩니다. <code>network</code> 키워드는 SI_NETWORK 를 설정합니다.
osname	osname	CD에서 확인되는 Solaris OS의 버전과 운영 체제 이름을 반환하고 SI_OSNAME 을 설정합니다. JumpStart 프로그램이 Solaris 릴리스를 찾지만 버전을 파악할 수 없는 경우, 반환되는 버전은 SystemV 입니다.
	rootdisk	시스템의 루트 디스크의 이름과 MB 단위의 크기를 반환하고 SI_ROOTDISK 를 설정합니다.
totaldisk	totaldisk	시스템의 전체 디스크 공간을 MB 단위로 반환하고 SI_TOTALDISK 를 설정합니다. 전체 디스크 공간에는 시스템에 연결된 모든 작동 가능한 디스크가 포함됩니다.

3

RAID-1 볼륨 사용

이 부분에서는 Solaris 설치 또는 업그레이드에서 사용할 수 있는 Solaris 볼륨 관리자 구성 요소의 개요를 제공합니다. 또한 RAID-1 볼륨을 사용하는 데 필요한 지침 및 요구 사항도 포함되어 있습니다.

설치 중 RAID-1 볼륨(미러) 만들기(개요)

이 절에서는 미러된 파일 시스템 만들기의 장점에 대해 설명합니다. 또한 미러된 파일 시스템을 만드는 데 필요한 Solaris 볼륨 관리자 구성 요소에 대해 설명합니다.

이 장은 다음 내용으로 구성되어 있습니다.

- 193 페이지 “RAID-1 볼륨을 사용하는 이유”
- 194 페이지 “RAID-1 볼륨의 작동 방법”
- 196 페이지 “Solaris 볼륨 관리자 구성요소의 개요”
- 198 페이지 “RAID-1 볼륨 디스크 레이아웃의 예”

Solaris Live Upgrade를 사용하여 미러된 파일 시스템을 만드는 방법에 대한 추가 정보는 **Solaris 10 6/06 설치 설명서: Solaris Live Upgrade** 및 업그레이드 계획의 “RAID-1 볼륨(미러) 파일 시스템 생성 일반 지침”을 참조하십시오.

사용자 정의 JumpStart 설치 방법으로 미러된 파일 시스템을 작성하는 방법에 대한 추가 정보는 168 페이지 “filesys 프로파일 키워드(RAID-1 볼륨 작성)” 및 173 페이지 “metadb 프로파일 키워드(상태 데이터베이스 복제본 만들기)”를 참조하십시오.

RAID-1 볼륨을 사용하는 이유

설치 또는 업그레이드 동안 RAID-1 볼륨을 작성하여 다중 물리적 디스크에 걸쳐 시스템 데이터를 복제할 수 있습니다. 여러 디스크에 데이터를 복제하면 디스크 손상이나 디스크 오류로부터 데이터를 보호할 수 있습니다.

Solaris 사용자 정의 JumpStart 및 Solaris Live Upgrade 설치 방법은 Solaris 볼륨 관리자 기술을 사용하여 파일 시스템을 미러하는 RAID-1 볼륨을 사용합니다. Solaris 볼륨 관리자는 볼륨을 사용하여 디스크를 안정적으로 관리할 수 있는 강력한 방법을 제공합니다. Solaris 볼륨 관리자는 연결, 스트라이프 및 다른 복잡한 구성을 가능하게 합니다. 사용자 정의 JumpStart 및 Solaris Live Upgrade 설치 방법을 통해 루트(/) 파일 시스템에 대한 RAID-1 볼륨 만들기 등의 작업 일부를 사용할 수 있습니다. 설치 또는 업그레이드 동안 RAID-1 볼륨을 작성하여 설치 후 작성할 필요성을 제거할 수 있습니다.

- 지침은 203 페이지 “사용자 정의 JumpStart 및 Solaris Live Upgrade 지침”을 참조하십시오.

- 복잡한 Solaris 볼륨 관리자 소프트웨어 및 구성요소에 대한 자세한 내용은 **Solaris Volume Manager Administration Guide**를 참조하십시오.

RAID-1 볼륨의 작동 방법

Solaris 볼륨 관리자는 가상 디스크를 사용하여 물리적 디스크와 해당하는 연관된 데이터를 관리할 수 있습니다. Solaris 볼륨 관리자에서는 가상 디스크를 볼륨이라고 합니다. 볼륨은 시스템에 단일 논리 장치로 나타나는 물리적 슬라이스 그룹의 이름입니다. 볼륨은 표준 UNIX *용어로 실제로 의사 또는 가상 장치입니다.

볼륨은 응용 프로그램 또는 파일 시스템(예: UFS)의 관점에서 보면 물리적 디스크와 기능적으로 동일합니다. Solaris 볼륨 관리자는 볼륨에 지정된 입출력 요청을 기본 구성 디스크로 변환합니다.

Solaris 볼륨 관리자 볼륨은 슬라이스(디스크 분할 영역)나 다른 Solaris 볼륨 관리자 볼륨에서 구축됩니다.

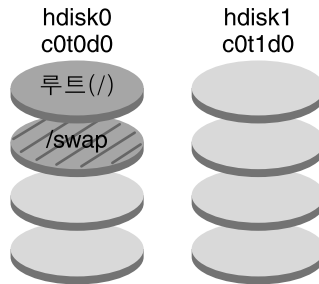
볼륨을 사용하여 성능 및 데이터 가용성을 높입니다. 경우에 따라 볼륨으로 입출력 성능을 높일 수도 있습니다. 볼륨은 기능적으로 슬라이스와 같은 방식으로 작동합니다. 볼륨의 모양은 슬라이스와 유사하므로 최종 사용자, 응용 프로그램 및 파일 시스템에 명확히 나타납니다. 물리적 장치와 마찬가지로 Solaris 볼륨 관리자 소프트웨어를 사용하여 블록 장치 이름이나 원시 장치 이름을 통해 볼륨에 액세스할 수 있습니다. 블록 장치 또는 원시 장치 사용 여부에 따라 볼륨 이름이 변경됩니다.

사용자 정의 JumpStart 설치 방법 및 Solaris Live Upgrade는 미러된 파일 시스템을 만들기 위해 블록 장치 사용을 지원합니다. 볼륨 이름에 대한 자세한 내용은 204 페이지 “[사용자 정의 JumpStart 및 Solaris Live Upgrade에 대한 RAID 볼륨 이름 요구 사항 및 지침](#)”을 참조하십시오.

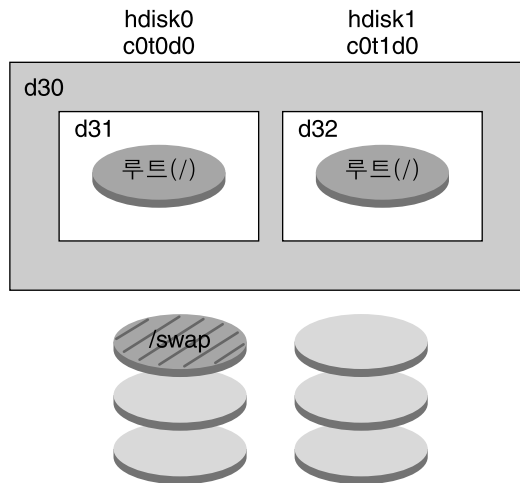
RAID-0 볼륨(단일 슬라이스 연결) 및 RAID-1 볼륨을 작성할 때, Solaris 볼륨 관리자는 연결의 데이터를 복제하고(하위 미러) 하위 미러를 볼륨으로 처리합니다.

[그림 12-1](#)은 2개의 물리적 디스크에 걸쳐 루트(/) 파일 시스템을 복제하는 미러를 보여줍니다.

두 개의 물리적 디스크를 가진 원본 시스템



두 개의 RAID-0 볼륨을 가진 미러된 루트 파일 시스템



d30 - RAID-1 볼륨(미러)

d31 - 단일 슬라이스 연결(하위 미러)

d32 - 단일 슬라이스 연결(하위 미러)

그림 12-1 디스크의 루트(/) 파일 시스템에 RAID-1 볼륨 만들기

그림 12-1은 다음과 같은 구성의 시스템입니다.

- hdisk0의 루트 파일 시스템(/)은 이름이 d31인 단일 슬라이스 연결에 포함됩니다.
- 이름이 d32인 단일 슬라이스 연결은 hdisk1이라는 하드 디스크에 만들어집니다.
- 이름이 d30인 미러는 d31과 d32라는 이름의 하위 미러로 구성됩니다.
- 미러는 양쪽 하위 미러의 루트 파일 시스템에 데이터를 복제합니다.

Solaris 볼륨 관리자 구성요소의 개요

사용자 정의 JumpStart 설치 방법 및 Solaris Live Upgrade를 사용하면 데이터를 복제하는 데 필요한 다음 구성 요소를 만들 수 있습니다.

- 상태 데이터베이스 및 상태 데이터베이스 복제본(metadb)
- 단일 슬라이스 연결(하위 미러)
- RAID-1 볼륨(미러)

이 절에서는 해당 구성 요소 각각에 대해 간단히 설명합니다. 이러한 구성 요소에 대한 자세한 내용은 **Solaris Volume Manager Administration Guide**를 참조하십시오.

상태 데이터베이스 및 상태 데이터베이스 복제본

상태 데이터베이스는 Solaris 볼륨 관리자 구성의 상태에 대해 물리적 디스크에 정보를 저장하는 데이터베이스입니다. 상태 데이터베이스는 사용자 구성 변경 사항을 기록 및 추적합니다. Solaris 볼륨 관리자는 구성이나 상태가 변경되면 자동으로 상태 데이터베이스를 업데이트합니다. 새 볼륨 만들기는 구성 변경의 예입니다. 하위 미러 오류는 상태 변경의 예입니다.

상태 데이터베이스는 실제로 여러 개의 복제된 데이터베이스 복사본 모음입니다. 상태 데이터베이스 복제본으로 불리는 각 복사본은 항상 데이터베이스의 데이터가 유효한지 확인합니다. 상태 데이터베이스의 복사본이 있으면 단일 지점 실패로부터 데이터가 손실되지 않도록 보호합니다. 이 상태 데이터베이스는 모든 알려진 상태 데이터베이스 복제본의 위치 및 상태를 추적합니다.

Solaris 볼륨 관리자는 상태 데이터베이스와 해당 상태 데이터베이스 복제본을 만들어야 작동할 있습니다. Solaris 볼륨 관리자 구성에 운영 상태 데이터베이스가 있어야 합니다.

구성을 설정할 때 다음 중 하나에서 상태 데이터베이스 복제본을 찾을 수 있습니다.

- 전용 슬라이스
- (Solaris Live Upgrade에만 해당) 나중에 볼륨의 일부가 될 슬라이스

두 개 이상의 상태 데이터베이스 복사본을 한 개의 슬라이스에 보관할 수 있습니다. 하지만 상태 데이터베이스 복제본을 단일 슬라이스에 두면 시스템이 단일 지점 실패에 더 취약할 수 있습니다.

상태 데이터베이스 복제본은 항상 상태 데이터베이스의 데이터가 유효한지 확인합니다. 상태 데이터베이스가 업데이트되면 각 상태 데이터베이스 복제본도 업데이트됩니다. 시스템이 충돌하는 경우 모든 업데이트가 손실되는 것을 방지하기 위해 업데이트는 한 번에 하나씩 이루어집니다.

시스템에서 상태 데이터베이스 복제본을 읽는 경우 Solaris 볼륨 관리자는 아직 유효한 데이터를 포함하는 상태 데이터베이스 복제본을 식별해야 합니다. Solaris 볼륨 관리자는 과반수 일치 알고리즘을 사용하여 이 정보를 확인합니다. 이 알고리즘에서 상태 데이터베이스 복제본의 유효성을 결정하려면 상태 데이터베이스 복제본의 과반수(반 + 1)가

사용 가능하며 일치해야 합니다. 이 과반수 일치 알고리즘으로 인해 디스크 구성 설정 시 적어도 세 개의 상태 데이터베이스 복제본을 만들어야 합니다. 세 개의 상태 데이터베이스 복제본 중 적어도 두 개를 사용할 수 있어야 일치에 도달할 수 있습니다.

각 상태 데이터베이스 복제본은 기본적으로 디스크 저장소의 4MB(8192 디스크 섹터)를 차지합니다. 복제본은 다음 장치에 저장될 수 있습니다.

- 전용 로컬 디스크 슬라이스
- (Solaris Live Upgrade에만 해당) 볼륨의 일부가 될 로컬 슬라이스
- (Solaris Live Upgrade에만 해당) UFS 로깅 장치의 일부가 될 로컬 슬라이스

복제본은 루트(/), swap, /usr 슬라이스 또는 기존 파일 시스템이나 데이터를 포함하는 슬라이스에 저장될 수 없습니다. 복제본이 저장된 후에는 볼륨이나 파일 시스템을 같은 슬라이스에 둘 수 없습니다.

<p>사용자 정의 JumpStart 또는 Solaris Live Upgrade를 사용하여 RAID-1 볼륨을 사용할 경우 다음 지침과 요구 사항을 검토하십시오.</p>	<p>201 페이지 “상태 데이터베이스 복제본 지침 및 요구 사항”</p>
---	---

상태 데이터베이스 및 상태 데이터베이스 복제본에 대한 자세한 내용

Solaris Volume Manager Administration Guide

RAID-0 볼륨(연결)

사용자 정의 JumpStart 및 Solaris Live Upgrade 설치 방법을 사용하여 RAID-0 볼륨을 작성할 수 있습니다. RAID-0 볼륨 단일 슬라이스 연결은 데이터가 여러 구성 요소에 연속적으로 그리고 인접하여 구성되어 있는 볼륨으로 한 개의 논리 저장소 단위를 형성합니다. 사용자 정의 JumpStart 설치 방법 및 Solaris Live Upgrade를 사용하여 스트라이프나 다른 복잡한 Solaris 볼륨 관리자 볼륨을 만들 수 없습니다.

설치 또는 업그레이드 동안 RAID-1 볼륨(미러)을 만들고 RAID-0 볼륨을 이러한 미러에 연결할 수 있습니다. 미러된 RAID-0 볼륨을 하위 미러라고 합니다. 미러는 하나 이상의 RAID-0 볼륨으로 구성됩니다. 설치 후 Solaris 볼륨 관리자 소프트웨어를 통해 RAID-1 미러 볼륨을 관리하여 별도의 RAID-0 하위 미러 볼륨에서 데이터를 관리할 수 있습니다.

사용자 정의 JumpStart 설치 방법을 사용하면 최대 두 개의 하위 미러로 구성된 미러를 만들 수 있습니다. Solaris Live Upgrade를 사용하면 최대 세 개의 하위 미러로 구성된 미러를 만들 수 있습니다. 실제로는 대부분 양방향 미러면 충분합니다. 세 번째 하위 미러를 사용하면 한 개의 하위 미러는 백업용으로 오프라인화하여 데이터 중복성을 잃지 않고 온라인 백업을 만들 수 있습니다.

RAID-0 볼륨 계획 정보

203 페이지 “RAID-1 및 RAID-0 볼륨 요구 사항 및 지침”

RAID-1 볼륨(미러)

RAID-1 볼륨 또는 미러는 RAID-0 볼륨(단일 슬라이스 연결)의 데이터의 동일한 복사본을 유지하는 볼륨입니다. RAID-1 볼륨을 사용하여 파일 시스템을 미러하려면 디스크에 투자해야 합니다. 적어도 데이터 양의 2배에 해당하는 디스크 공간이 필요합니다. Solaris 볼륨 관리자 소프트웨어는 모든 RAID-0 볼륨에 쓰기 때문에 데이터를 복제하면 디스크에 쓰는 쓰기 요청에 필요한 시간도 늘어날 수 있습니다.

RAID-1 볼륨을 사용하면 데이터를 양쪽 RAID-0 볼륨에서 동시에 읽을 수 있으므로(두 볼륨 모두 요청을 충족) 성능이 향상됩니다. 한 개의 물리적 디스크에 오류가 발생하면 성능 손상이나 데이터 손실 없이 계속해서 미러를 사용할 수 있습니다.

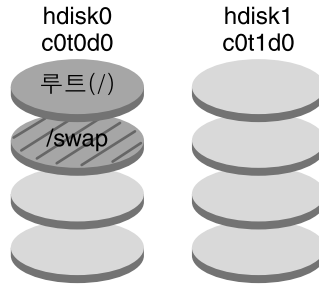
RAID-1 볼륨을 구성한 후 볼륨은 마치 물리적 슬라이스인 것처럼 사용할 수 있습니다.

기존 파일 시스템을 비롯하여 모든 파일 시스템을 복제할 수 있습니다. 데이터베이스와 같은 응용 프로그램에 대해서도 RAID-1 볼륨을 사용할 수 있습니다.

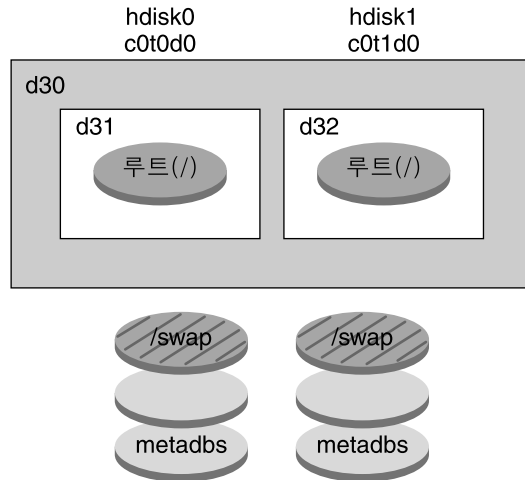
RAID-1 볼륨 디스크 레이아웃의 예

다음 그림은 2개의 물리적 디스크에 걸쳐 루트 파일 시스템(/)을 복제하는 RAID-1 볼륨을 보여줍니다. 상태 데이터베이스 복제본(metadb)은 양쪽 디스크에 놓입니다.

두 개의 물리적 디스크를 가진 원본 시스템



두 개의 RAID-0 볼륨과 상태 데이터베이스 복제본(metadbs)을 가진 미러된 파일 시스템



d30 - RAID-1 볼륨(미러)

d31 - 단일 슬라이스 연결(하위 미러)

d32 - 단일 슬라이스 연결(하위 미러)

그림 12-2 RAID-1 볼륨 디스크 레이아웃

그림 12-2는 다음 구성을 가진 시스템을 보여줍니다.

- hdisk0의 루트 파일 시스템(/)은 이름이 d31인 단일 슬라이스 연결에 포함됩니다.
- 이름이 d32인 단일 슬라이스 연결은 hdisk1이라는 하드 디스크에 만들어집니다.
- 이름이 d30인 미러는 d31과 d32라는 이름의 하위 미러로 구성됩니다.
- 미러는 양쪽 하위 미러의 루트 파일 시스템에 데이터를 복제합니다.
- 상태 데이터베이스 복제본은 hdisk0과 hdisk1 슬라이스 모두에 만들어집니다.

사용자 정의 JumpStart 설치 방법을 사용하여 이 구성을 만드는 프로파일의 예

예 6-13

Solaris Live Upgrade를 사용하여 RAID-1 볼륨을 만드는 방법에 대한 지침

Solaris 10 6/06 설치 설명서: **Solaris Live Upgrade** 및 업그레이드 계획의 “RAID-1 볼륨(미러)으로 부트 환경 만들기(명령줄 인터페이스)”

설치 중 RAID-1 볼륨(미러) 만들기(계획)

이 장은 사용자 정의 JumpStart 또는 Solaris Live Upgrade를 사용하여 RAID-1 볼륨을 만드는 데 필요한 요구 사항과 지침에 대해 설명합니다.

이 장은 다음 내용으로 구성되어 있습니다.

- 201 페이지 “시스템 요구 사항”
- 201 페이지 “상태 데이터베이스 복제본 지침 및 요구 사항”
- 203 페이지 “RAID-1 및 RAID-0 볼륨 요구 사항 및 지침”
- 207 페이지 “단일 사용자 모드로 부트가 RAID-1 볼륨에 미치는 영향”

Solaris Live Upgrade 설치 방법을 사용하여 미러된 파일 시스템을 작성하기 위한 계획에 대한 추가 정보는 **Solaris 10 6/06** 설치 설명서: **Solaris Live Upgrade** 및 업그레이드 계획의 “RAID-1 볼륨(미러) 파일 시스템 생성 일반 지침”을 참조하십시오.

사용자 정의 JumpStart 설치 방법으로 미러된 파일 시스템을 작성하는 방법의 자세한 내용은 168 페이지 “filesys 프로필 키워드(RAID-1 볼륨 작성)” 및 173 페이지 “metadb 프로필 키워드(상태 데이터베이스 복제본 만들기)”를 참조하십시오.

시스템 요구 사항

RAID-1 볼륨을 만들어 특정한 슬라이스의 데이터를 복제하려면 사용을 계획하는 디스크는 설치 중에 시스템에 직접 연결되고 시스템에서 사용 가능해야 합니다.

상태 데이터베이스 복제본 지침 및 요구 사항

단일 지점 오류를 피하려면 슬라이스, 드라이브 및 제어기에 상태 데이터베이스 복제본을 분산시켜야 합니다. 단일 구성 요소 오류를 피하려면 다수의 복제본이 필요합니다. 장치에 오류가 발생하여 복제본을 잃는 경우 오류로 인해 Solaris 볼륨 관리자 소프트웨어를 실행하거나 시스템을 재부트할 때 문제가 발생할 수 있습니다. Solaris 볼륨 관리자 소프트웨어는 최소한 복제본의 반을 실행할 것을 요구하지만 여러 사용자 모드로 재부트하려면 과반수 이상(반 + 1)을 요구합니다.

상태 데이터베이스 복제본 만들기 및 관리에 대한 자세한 내용은 **Solaris Volume Manager Administration Guide**를 참조하십시오.

상태 데이터베이스 복제본에 대한 슬라이스 선택

상태 데이터베이스 복제본에 대한 슬라이스를 선택하기 전에 다음 지침 및 권장 사항을 고려하십시오.

- 복제본 당 최소 4MB의 전용 슬라이스에 상태 데이터베이스 복제본을 만들어야 합니다. 필요한 경우 RAID-0이나 RAID-1 볼륨의 일부로 사용될 슬라이스에 상태 데이터베이스 복제본을 만들 수 있습니다. 슬라이스를 볼륨에 추가하기 전에 먼저 복제본을 만들어야 합니다.
- 기본적으로 상태 데이터베이스 복제본의 크기는 4MB나 8192 디스크 블록입니다. 사용자 디스크 슬라이스의 크기가 그 정도로 작지 않을 수도 있으므로 상태 데이터베이스 복제본을 포함하도록 슬라이스의 크기를 조정할 수 있습니다. 슬라이스 크기 조정에 대한 자세한 내용은 **System Administration Guide: Devices and File Systems**의 12 장, “Administering Disks (Tasks)”를 참조하십시오.
- 사용 중이 아닌 슬라이스에 상태 데이터베이스 복제본을 만들 수 있습니다. 상태 데이터베이스 복제본을 위해 예약된 슬라이스의 일부를 다른 용도로 사용하면 안 됩니다.
- 기존 파일 시스템이나 루트(/), /usr 및 swap 파일 시스템에 상태 데이터베이스 복제본을 만들 수 없습니다. 필요한 경우 swap에서 공간을 할당하여 새 슬라이스를 만든 다음(슬라이스 이름을 사용할 수 있는 경우) 상태 데이터베이스 복제본을 새 슬라이스에 놓을 수 있습니다.
- 상태 데이터베이스 복제본이 볼륨의 일부가 될 슬라이스에 있는 경우 볼륨 용량은 복제본이 차지하는 공간 만큼 줄어듭니다. 복제본이 차지하는 공간은 다음 실린더 경계까지 반올림되므로 볼륨에서 이 공간은 건너뛴니다.

상태 데이터베이스 복제본 수 선택

상태 데이터베이스 복제본의 수를 선택하기 전에 다음 지침을 고려하십시오.

- 최소 3개의 상태 데이터베이스 복제본이 필요하며 Solaris 볼륨 관리자 디스크 집합 당 최대 50개의 복제본이 허용됩니다. 다음은 권장되는 지침입니다.
 - 단일 드라이브 사용 시스템의 경우: 세 개의 복제본을 모두 하나의 슬라이스에 둡니다.
 - 2-4개 드라이브 사용 시스템의 경우: 각 드라이브에 두 개의 복제본을 둡니다.
 - 5개 이상 드라이브 사용 시스템의 경우: 각 드라이브에 한 개의 복제본을 둡니다.
- 추가 상태 데이터베이스 복제본은 미러의 성능을 향상시킬 수 있습니다. 일반적으로 시스템에 추가하는 각 미러에 두 개의 복제본을 추가해야 합니다.
- 소규모의 임의 입출력에 사용할 RAID-1 볼륨이 있는 경우(예: 데이터베이스) 복제본 수를 고려하십시오. 최적의 성능을 위해서는 RAID-1 볼륨에 연결되지 않은 슬라이스(특히 디스크 및 제어기)에 있는 RAID-1 볼륨당 적어도 2개의 추가 복제본이 있어야 합니다.

제어기에 상태 데이터베이스 복제본 분산

여러 개의 제어기가 있는 경우 가능하면 복제본이 모든 제어기에 균등하게 분산되도록 합니다. 이러한 전략은 제어기에 오류가 발생하는 경우 중복성을 제공하여 로드 균형을 조정하는 데도 도움이 됩니다. 하나의 제어기에 여러 개의 디스크가 있는 경우 각 제어기에 있는 최소 두 개의 디스크에서 복제본을 저장해야 합니다.

RAID-1 및 RAID-0 볼륨 요구 사항 및 지침

RAID-1 볼륨(미러) 및 RAID-0 볼륨(단일 슬라이스 연결)으로 작업할 경우 다음 지침을 고려하십시오.

사용자 정의 JumpStart 및 Solaris Live Upgrade 지침

사용자 정의 JumpStart 설치 방법 및 Solaris Live Upgrade는 Solaris 볼륨 관리자 소프트웨어에서 사용할 수 있는 일련의 기능을 지원합니다. 이러한 설치 프로그램을 사용하여 미러된 파일 시스템을 만드는 경우 다음 지침을 고려하십시오.

설치 프로그램	지원되는 기능	지원되지 않는 기능
사용자 정의 JumpStart 및 Solaris Live Upgrade	<ul style="list-style-type: none"> ■ RAID-0 볼륨과 RAID-1 볼륨을 지원하지만 RAID-5 볼륨과 같은 다른 Solaris 볼륨 관리자 구성 요소는 지원하지 않습니다. ■ RAID-0 볼륨이 지원되지만 단일 슬라이스 연결로서만입니다. 	Solaris 볼륨 관리자에서 RAID-0 볼륨은 디스크 스트라이프 및 디스크 연결을 참조할 수 있습니다. 설치나 업그레이드 중에 RAID-0 스트라이프 볼륨을 만들 수 없습니다.
사용자 정의 JumpStart	<ul style="list-style-type: none"> ■ 초기 설치 중에만 RAID-1 볼륨 만들기를 지원합니다. ■ 각 RAID-1 볼륨에 대하여 최고 2개의 RAID-0 볼륨(하위 미러)을 만들 수 있습니다. 대체로 두 개의 하위 미러는 대부분의 응용 프로그램에 충분한 데이터 중복성을 제공하며 디스크 드라이브 비용도 덜 듭니다. 	<ul style="list-style-type: none"> ■ RAID-1 볼륨이 구성되어 있으면 업그레이드를 지원하지 않습니다. ■ 2개 이상의 RAID-0 볼륨은 지원되지 않습니다.

설치 프로그램	지원되는 기능	지원되지 않는 기능
Solaris Live Upgrade	<ul style="list-style-type: none"> ■ 각 RAID-1 볼륨에 대하여 최고 3개의 RAID-0 볼륨(하위 미러)을 만들 수 있습니다. 세 개의 하위 미러를 사용하면 데이터 중복성을 지속하기 위해 두 개의 나머지 하위 미러를 유지 관리하면서 한 개의 하위 미러를 오프라인으로 만들어 백업을 수행할 수 있습니다. ■ 업그레이드 중 RAID-1 볼륨 만들기를 지원합니다. <p>예를 들어, Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 “RAID-1 볼륨(미러)으로 부트 환경 만들기(명령줄 인터페이스)”를 참조하십시오.</p>	3개 이상의 RAID-0 볼륨은 지원되지 않습니다.
RAID-1 볼륨을 사용한 Solaris Flash 만들기 및 설치	<p>Solaris 볼륨 관리자 RAID-1 볼륨이 구성되어 있는 마스터 시스템에서 작성되는 Solaris Flash 아카이브를 작성할 수 있습니다. Solaris Flash 작성 소프트웨어는 아카이브에서 모든 RAID-1 볼륨 정보를 제거하여 복제 시스템의 무결성을 유지합니다. 사용자 정의 JumpStart를 사용하면 JumpStart 프로필을 사용하여 RAID-1 볼륨을 재구축할 수 있습니다. Solaris Live Upgrade를 사용하면 구성된 RAID-1 볼륨으로 부트 환경을 만들고 아카이브를 설치합니다. Solaris Flash 아카이브로 RAID-1 볼륨을 설치할 때 Solaris 설치 프로그램을 사용할 수 없습니다.</p> <p>JumpStart 프로필의 RAID-1 볼륨의 예는 83 페이지 “프로필의 예”를 참조하십시오.</p>	Veritas VxVM은 Solaris Flash이 사용할 수 없는 영역의 구성 정보를 저장합니다. Veritas VxVM 파일 시스템이 구성된 경우 Solaris Flash 아카이브를 만들면 안됩니다. JumpStart 및 Solaris Live Upgrade를 포함하는 Solaris 설치도 설치 시에 VxVM 볼륨 재구축을 지원하지 않습니다. 따라서 Solaris Flash 아카이브를 사용해 Veritas VxVM 소프트웨어를 배포할 예정이면 VxVM 파일 시스템을 구성하기 전에 아카이브를 만들어야 합니다. 복제 시스템은 아카이브를 적용하고 시스템을 재부트한 후에 개별적으로 구성해야 합니다.

사용자 정의 JumpStart 및 Solaris Live Upgrade에 대한 RAID 볼륨 이름 요구 사항 및 지침

볼륨에 이름을 할당할 때 다음 규칙을 준수하십시오.

- 슬라이스 번호 및 디스크 번호를 볼륨 번호에 매핑하는 이름 지정 메소드를 사용합니다.
- 볼륨 이름은 문자 d로 시작하고 숫자가 뒤에 와야 합니다(예: d0).
- Solaris 볼륨 관리자는 0-127까지 128개의 기본 볼륨 이름을 갖습니다. 다음 목록은 몇 가지 예제 볼륨 이름을 보여 줍니다.
 - Device /dev/md/dsk/d0 - block volume d0
 - 장치 /dev/md/dsk/d1 - 블록 볼륨 d1
- 특정 볼륨 유형 각각의 범위를 사용합니다. 예를 들어 RAID-1 볼륨에는 0-20, RAID-0 볼륨에는 21-40을 할당합니다.

- `/dev/md/dsk/d1` 같은 전체 볼륨 이름을 지정하는 대신 `d1` 같은 약어로 된 볼륨 이름을 사용할 수 있습니다.

Solaris Live Upgrade에 대한 RAID 볼륨 이름 지정 규약

물리적 디스크 슬라이스와 Solaris 볼륨 관리자의 볼륨 이름은 약어로 만들 수 있습니다. 약어는 장치를 고유하게 식별하는 짧은 이름입니다. 다음 예를 참고하십시오.

- Solaris 볼륨 관리자 볼륨은 `dnum` 형식으로 식별할 수 있습니다. 예를 들어 `/dev/md/dsk/d10`은 간단하게 `d10`으로 줄일 수 있습니다.
- 시스템에 하나의 제어기와 여러 디스크가 있는 경우에는 `t0d0s0`을 사용할 수 있지만 제어기가 여러 개일 경우에는 `c0t0d0s0`을 사용합니다.

Solaris Live Upgrade를 사용하여 RAID-1 볼륨(미러) 및 RAID-0 볼륨(하위 미러)을 만들 경우 소프트웨어가 볼륨 이름을 감지 및 할당하도록 하거나 직접 이름을 할당할 수 있습니다. 소프트웨어가 이름을 감지하게 할 경우 소프트웨어는 사용 가능한 첫 번째 미러 또는 하위 미러를 지정합니다. 직접 이름을 할당할 경우 미러 이름이 0으로 끝나게 할당해야 설치 과정에서 하위 미러에 1 및 2로 끝나는 이름을 사용할 수 있습니다. 하위 미러 이름을 할당할 경우 1 또는 2로 끝나게 이름을 할당합니다. 숫자를 잘못 할당하면 미러가 만들어지지 않을 수 있습니다. 예를 들어, 1 또는 2로 끝나는 번호(`d1` 또는 `d2`)를 미러 이름으로 지정할 경우 미러 이름이 하위 미러 이름과 동일하다면 Solaris Live Upgrade는 미러를 생성할 수 없습니다.

이 예에서, Solaris Live Upgrade가 볼륨 이름을 할당합니다. RAID-1 볼륨 `d0`과 `d1`이 유일한 사용중 볼륨입니다. 미러 `d10`의 경우, Solaris Live Upgrade는 장치 `c0t0d0s0`에 대한 하위 미러용으로 `dd2`를 선택하고 장치 `c1t0d0s0`에 대한 하위 미러용으로 `d3`를 선택합니다.

```
lucreate -n newbe -m /:d10:mirror,ufs -m /:c0t0d0s0:attach -m
/:c1t0d0s0:attach
```

이 예에서, 볼륨 이름은 명령어에서 할당됩니다. 미러 `d10`의 경우, `d11`은 장치 `c0t0d0s0`의 하위 미러의 이름이고 `d12`는 장치 `c1t0d0s0`의 하위 미러의 이름입니다.

```
lucreate -n newbe -m /:d10:mirror,ufs -m /:c0t0d0s0,d11:attach -m
/:c1t0d0s0,d12:attach
```

Solaris 볼륨 관리자 이름 지정 요구 사항에 대한 자세한 내용은 **Solaris Volume Manager Administration Guide**를 참조하십시오.

사용자 정의 JumpStart에 대한 RAID 볼륨 이름 지정 규약

사용자 정의 JumpStart 설치 방법을 사용하여 RAID-1 볼륨(미러) 및 RAID-0 볼륨(하위 미러)을 생성할 때 소프트웨어가 미러에 대한 볼륨 이름을 감지하여 지정하게 하거나 프로필에서 이름을 지정할 수 있습니다. 소프트웨어가 이름을 감지하게 할 경우 소프트웨어는 사용 가능한 첫 번째 볼륨 번호를 지정합니다. 프로필에서 이름을 지정할 경우 미러 이름이 0으로 끝나게 지정해야 설치 과정에서 하위 미러에 1 및 2로 끝나는 이름을 사용할 수 있습니다. 올바르지 않은 번호를 지정할 경우 미러가 생성되지 않을 수 있습니다. 예를 들어, 1 또는 2로 끝나는 번호(`d1` 또는 `d2`)를 미러 이름으로 지정할 경우 미러 이름이 하위

미러 이름과 동일하다면 JumpStart는 미러를 생성할 수 없습니다. 다음 프로필 예에서는 사용 가능한 첫 번째 볼륨 번호가 미러에 지정되었습니다. 0으로 끝나는 사용 가능한 미러가 d10이라면 d11 및 d12라는 이름이 하위 미러에 지정됩니다.

```
filesys                mirror c0t0d0s1 /
```

다음 프로필 예에서는 미러 번호가 프로필에서 d30으로 지정됩니다. 소프트웨어는 미러 번호와 첫 번째로 사용 가능한 하위 미러를 기준으로 하위 미러 이름을 지정합니다. 여기서 하위 미러의 이름은 d31 및 d32가 됩니다.

```
filesys                mirror:d30 c0t1d0s0 c0t0d0s0 /
```

Solaris 볼륨 관리자 이름 지정 요구 사항에 대한 자세한 내용은 **Solaris Volume Manager Administration Guide**를 참조하십시오.

디스크 및 제어기 선택 지침

파일 시스템을 미러하기 위해 사용할 디스크와 제어기를 선택하는 경우 다음 지침을 고려하십시오.

- 수행할 수 있는 동시 읽기 및 쓰기 수를 높이려면 다른 제어기에 있는 구성 요소를 사용합니다.
- 다양한 디스크와 제어기에 여러 하위 미러 슬라이스를 보관합니다. 동일한 미러에 있는 두 개 이상의 하위 미러 슬라이스가 동일한 디스크에 있는 경우 데이터 보호가 상당히 감소됩니다.
- 제어기 및 연결된 케이블은 디스크보다 자주 오류를 일으키는 경향이 있으므로 여러 제어기에 하위 미러를 구성합니다. 이렇게 하면 미러의 성능도 향상됩니다.
- 단일 미러에 같은 종류의 디스크와 제어기를 사용합니다. 특히 이전 SCSI 저장 장치에서 디스크나 제어기의 모델이나 브랜드에 따라 그 성능이 상당히 다를 수 있습니다. 단일 미러에서 여러 성능 수준을 함께 사용하는 것은 성능을 상당히 저하시킬 수 있습니다.

슬라이스 선택 지침

파일 시스템을 미러하기 위해 사용할 슬라이스를 선택하는 경우 다음 지침을 고려하십시오.

- 루트(/), swap 및 /usr을 비롯한 모든 파일 시스템은 미러를 사용할 수 있습니다. 데이터베이스 같은 모든 응용 프로그램도 미러를 사용할 수 있습니다.
- 하위 미러 슬라이스의 크기가 같은지 확인합니다. 크기가 다른 하위 미러를 사용하면 사용되지 않는 디스크 공간이 발생합니다.
- 첫 번째 연결된 하위 미러가 실린더 0으로 시작하지 않는 미러된 파일 시스템인 경우 연결한 모든 추가 하위 미러도 실린더 0으로 시작하지 말아야 합니다. 실린더 0으로 시작하는 하위 미러를 원래 하위 미러가 0으로 시작하지 않는 미러에 연결하려고 하면 다음 오류 메시지가 표시됩니다.

```
can't attach
labeled submirror to an unlabeled mirror
```

연결하려는 하위 미러가 모두 실린더 0에서 시작하거나 모두 실린더 0에서 시작하지 않아야 합니다.

시작 실린더가 모든 하위 미러에서 동일할 필요는 없지만 모든 하위 미러는 실린더 0을 포함하거나 포함하지 않아야 합니다.

단일 사용자 모드로 부트가 RAID-1 볼륨에 미치는 영향

루트(/), /usr 및 swap용 미러가 있는 시스템이 단일 사용자 모드로 부트된 경우 시스템은 이러한 미러의 유지 관리가 필요하다는 것을 표시합니다. `metastat` 명령어를 사용하여 이러한 미러를 보면 이러한 미러 및 시스템의 모든 미러가 “유지 관리 필요” 상태로 표시됩니다.

이러한 상황은 잠재적으로 위험할 수 있지만 문제가 되지는 않습니다. 미러를 재동기화하기 위해 부트 중에 일반적으로 발생하는 `metasync -r` 명령어는 시스템이 단일 사용자 모드로 부트되면 중단됩니다. 시스템을 재부트하면 `metasync -r` 명령어가 실행되고 모든 미러가 재동기화됩니다.

이러한 중단이 문제가 되면 `metasync -r` 명령어를 수동으로 실행합니다.

`metasync`에 대한 자세한 내용은 `metasync(1M)` 매뉴얼 페이지 및 **Solaris Volume Manager Administration Guide**를 참조하십시오.

4

부록

이 부분은 문제 해결 및 참조 정보를 포함합니다.

문제 해결(작업)

이 장에는 목록과 Solaris 10 6/06 소프트웨어를 설치할 때 나타날 수 있는 특정 오류 메시지 및 일반적인 문제의 목록이 포함되어 있습니다. 또한 그런 문제들에 대한 해결 방법도 설명합니다. 다음 절 목록을 사용하여 설치 프로세스에서 문제가 발생한 부분을 먼저 확인할 수 있습니다.

- 211 페이지 “네트워크 설치 설정 문제”
- 211 페이지 “시스템 부트 관련 문제”
- 217 페이지 “Solaris OS의 초기 설치”
- 219 페이지 “Solaris OS 업그레이드”

주 - “bootable media” 라는 문구는 Solaris 설치 프로그램과 JumpStart 설치 방법을 의미합니다.

네트워크 설치 설정 문제

Unknown client *host_name*

원인: `add_install_client` 명령어의 *host_name* 인자가 이름 서비스의 호스트가 아닙니다.

설명: 호스트 *host_name*을 이름 서비스에 추가하고 `add_install_client` 명령어를 다시 실행합니다.

시스템 부트 관련 문제

매체에서 부트할 때 나타나는 오류 메시지

le0: No carrier - transceiver cable problem

원인: 시스템이 네트워크에 연결되지 않았습니다.

해결책: 네트워크에 연결되지 않은 시스템인 경우 이 메시지를 무시합니다. 네트워크에 연결된 시스템인 경우 이더넷 케이블이 제대로 연결되어 있는지 확인합니다.

The file just loaded does not appear to be executable

원인: 시스템에서 부트를 위한 적합한 매체를 찾을 수 없습니다.

해결책: 네트워크상의 설치 서버를 통해 Solaris 10 6/06 소프트웨어를 설치할 수 있도록 시스템을 적절히 설정했는지 확인합니다. 다음은 만들 수 있는 checks의 예입니다.

- Solaris Operating System DVD 또는 Solaris Software CD의 이미지를 설치 서버에 복사한 경우 설정할 때 시스템에 대한 올바른 플랫폼 그룹을 지정했는지 확인합니다.
- DVD 또는 CD 매체를 사용 중이면 Solaris Operating System DVD 또는 Solaris Software - 1 CD가 설치 서버에 마운트되고 액세스 가능한지 확인합니다.

boot: <filename>을 열 수 없음 (SPARC 기반 시스템에만 적용)

원인: 이 오류는 boot-file의 위치를 명시적으로 설정하여 해당 위치를 무시할 때 발생합니다.

주 -filename은 영향을 받는 파일의 이름의 변수입니다.

해결책: 다음 지침을 따르십시오.

- PROM에 있는 boot-파일을 “ (공백)으로 재설정합니다.
- 진단 스위치가 꺼져 있고 참으로 설정되어 있는지 확인합니다.

Can't boot from file/device

원인: 설치 매체가 부트 가능한 매체를 찾을 수 없습니다.

해결책: 다음 조건이 충족되어 있는지 확인합니다.

- DVD-ROM 또는 CD-ROM 드라이브가 제대로 설치되어 있으며 켜져 있습니다.
- Solaris Operating System DVD 또는 Solaris Software - 1 CD가 드라이브에 들어 있습니다.
- 디스크가 손상되거나 먼지가 묻어있지 않습니다.

WARNING: clock gained xxx days -- CHECK AND RESET DATE! (SPARC 기반 시스템에만 적용)

설명: 이 메시지는 정보를 제공하는 메시지입니다.

해결책: 이 메시지를 무시하고 설치를 계속합니다.

Not a UFS file system (x86 기반 시스템에만 적용)

원인: 설치 프로그램 또는 사용자 정의 JumpStart를 통하여 Solaris 10 6/06 소프트웨어를 설치할 때 부트 디스크를 선택하지 않았습니다. 이제 시스템을 부트하려면 BIOS를 편집해야 합니다.

해결책: 부트할 BIOS를 선택합니다. 지침은 BIOS 설명서를 참조하십시오.

매체에서 부트할 때 발생하는 일반적인 문제

The system does not boot.

설명: 사용자 정의 JumpStart 서버를 처음 설정할 때 오류 메시지를 반환하지 않는 부트 문제가 발생할 수 있습니다. 시스템과 시스템 부트 방법에 관한 정보를 확인하려면 `-v` 옵션과 함께 부트 명령어를 실행합니다. `-v` 옵션을 사용할 때 부트 명령어는 화면에 자세한 디버깅 정보를 표시합니다.

주 - 이 플래그가 제공되지 않으면 메시지는 계속 인쇄되지만 출력은 시스템 로그 파일로 지정됩니다. 자세한 내용은 `syslogd(1M)`을 참조하십시오.

해결책: SPARC 기반 시스템의 경우 `ok` 프롬프트에 다음 명령어를 입력합니다.

```
ok boot net -v - install
```

Toshiba SD-M 1401 DVD-ROM이 장착된 시스템에서 DVD 매체로부터의 부트가 실패합니다.

설명: 펌웨어 개정 1007인 Toshiba SD-M1401 DVD-ROM이 시스템에 설치되어 있는 경우 시스템이 Solaris Operating System DVD에서 부트할 수 없습니다.

해결책: 패치 111649-03 또는 이후 버전을 적용하여 Toshiba SD-M1401 DVD-ROM 드라이브의 펌웨어를 업데이트합니다. 경로 111649-03은 sunsolve.sun.com에서 사용 가능합니다.

The system hangs or panics when nonmemory PC cards are inserted. (x86 기반 시스템에만 적용)

원인: 비메모리 PC 카드는 다른 장치에서 사용하는 동일한 메모리 자원을 사용할 수 없습니다.

해결책: 이 문제를 보정하려면 PC 카드에 대한 지침을 참조하고 주소 범위를 확인합니다.

The system hangs before displaying the system prompt. (x86 기반 시스템에만 적용)

해결책: 지원되지 않는 하드웨어가 있습니다. 하드웨어 제조업체의 설명서를 확인하십시오.

네트워크에서 부트할 때 나타나는 오류 메시지

WARNING: getfile: RPC failed: error 5 (RPC Timed out).

설명: 이 오류는 설치 클라이언트의 부트 요청에 응답하는 서버가 네트워크상에 두 대 이상 있을 때 발생합니다. 설치 클라이언트가 잘못된 부트 서버에 연결하고 설치가 중단됩니다. 이 오류를 유발할 수 있는 구체적인 원인은 다음과 같습니다.

원인: 이유 `I:/etc/bootparams` 파일이 이 설치 클라이언트 항목을 갖고 있는 다른 서버에 존재할 수 있습니다.

해결책: 이유 1:네트워크 상의 서버에 설치 클라이언트용 /etc/bootparams 항목이 여러 개 있는지 확인합니다. 여러 항목이 있는 경우 설치 클라이언트에서 사용할 서버를 제외한 모든 설치 서버 및 부트 서버의 /etc/bootparams 파일에 있는 중복된 클라이언트 항목을 제거합니다.

원인: 이유 2:이 설치 클라이언트용으로 여러 /tftpboot 또는 /rplboot 디렉토리 항목이 있을 수 있습니다.

해결책: 이유 2:네트워크상의 서버에 설치 클라이언트용 /tftpboot 또는 /rplboot 디렉토리 항목이 여러 개 있는지 확인합니다. 여러 항목이 있는 경우 설치 클라이언트에서 사용할 서버를 제외한 모든 설치 서버와 부트 서버의 /tftpboot 또는 /rplboot 디렉토리에 있는 중복된 클라이언트 항목을 제거합니다.

원인: 이유 3:서버의 /etc/bootparams 파일에 설치 클라이언트 항목이 있고 모든 시스템이 프로필 서버에 액세스할 수 있는 또 다른 /etc/bootparams 파일에 항목이 있습니다. 이러한 항목은 다음과 유사합니다.

```
* install_config=profile_server:path
```

NIS 또는 NIS+ bootparams 테이블의 이전 항목과 유사한 행 역시 이런 오류를 유발할 수 있습니다.

해결책: 이유 3:와일드카드 항목이 이름 서비스 bootparams 맵 또는 테이블에 있습니다(예: *install_config=). 해당 항목을 삭제하고 부트 서버의 /etc/bootparams 파일에 추가하십시오.

No network boot server. Unable to install the system. See installation instructions. (SPARC 기반 시스템에만 적용)

원인: 이 오류는 네트워크에서 설치를 시도할 때 발생합니다. 시스템이 올바르게 설정되지 않았습니니다.

해결책: 네트워크에서 설치할 시스템을 올바르게 설정했는지 확인합니다. **Solaris 10 6/06** 설치 설명서: 네트워크 기반 설치의 “CD 이미지로 네트워크에서 설치할 시스템 추가”를 참조하십시오.

prom_panic: Could not mount file system(SPARC 기반 시스템에만 적용)

원인: 이 오류는 네트워크에서 Solaris 10 6/06를 설치하려는데 부트 소프트웨어에서 다음을 찾을 수 없는 경우 발생합니다.

- Solaris Operating System DVD, DVD 또는 설치 서버의 DVD 이미지 복사본
- 설치 서버의 Solaris Software - 1 CD 이미지, Solaris Software - 1 CD 또는 CD 이미지의 복사본

해결책: 설치 소프트웨어가 마운트 및 공유되어 있는지 확인합니다.

- 설치 서버의 DVD-ROM이나 CD-ROM 드라이브에서 Solaris를 설치하려면 Solaris Operating System DVD 또는 Solaris Software - 1 CD가 CD-ROM 드라이브에 들어 있고 마운트되어 있으며 /etc/dfs/dfstab 파일에서 공유되어 있는지 확인합니다.

- 설치 서버 디스크의 Solaris Operating System DVD 이미지나 Solaris Software - 1 CD 이미지 복사본을 이용해 설치하려면 해당 복사본의 디렉토리 경로가 `/etc/dfs/dfstab` 파일에서 공유되어 있는지 확인합니다.

Timeout waiting for ARP/RARP packet... (SPARC 기반 시스템에만 적용)

원인: 이유 1: 클라이언트가 네트워크에서 부트를 시도하지만 해당 클라이언트를 인식하는 시스템이 없습니다.

해결책: 이유 1: 시스템의 호스트 이름이 NIS 또는 NIS+ 이름 서비스에 있는지 확인합니다. 또한 부트 서버의 `/etc/nsswitch.conf` 파일에서 `bootparams` 검색 순서를 확인합니다.

예를 들어, `/etc/nsswitch.conf` 파일의 다음 행은 `JumpStart` 또는 Solaris 설치 프로그램이 `bootparams` 정보를 조회할 때 우선 NIS 맵을 조회한다는 것을 의미합니다. 프로그램이 아무런 정보도 찾지 못하면 설치 프로그램이 부트 서버의 `/etc/bootparams` 파일에서 찾습니다.

`bootparams: nis files`

원인: 이유 2: 클라이언트의 이더넷 주소가 올바르지 않습니다.

해결책: 이유 2: 설치 서버의 `/etc/ethers` 파일에 있는 클라이언트의 이더넷 주소가 올바른지 확인합니다.

원인: 이유 3: 사용자 정의 `JumpStart` 설치에서 `add_install_client` 명령어는 지정된 서버를 설치 서버로 사용하는 플랫폼 그룹을 지정합니다. `add_install_client`를 사용할 때 잘못된 구조 값이 사용되면 이 문제가 발생합니다. 예를 들어, 설치하고자 하는 시스템이 `sun4u`이지만 `sun4u`를 사용하는 경우에 문제가 발생합니다.

해결책: 이유 3: 올바른 구조 값으로 `add_install_client`를 반환합니다.

`ip: joining multicasts failed on tr0 - will use link layer broadcasts for multicast` (x86 기반 시스템에만 적용)

원인: 이 오류 메시지는 토큰 링 카드로 시스템을 부트할 때 표시됩니다. 이더넷 멀티캐스트 및 토큰 링 멀티캐스트는 동일한 방법으로 작동하지 않습니다. 드라이브에서 잘못된 멀티캐스트 주소를 받았기 때문에 이 오류 메시지를 반환합니다.

해결책: 이 오류 메시지를 무시하십시오. 멀티캐스트가 작동하지 않으면 IP는 대신 계층 브로드캐스트를 사용하고 설치 실패를 유발하지 않습니다.

Requesting Internet address for *Ethernet_Address* (x86 기반 시스템에만 적용)

원인: 클라이언트가 네트워크에서 부트를 시도하지만 해당 클라이언트를 인식하는 시스템이 없습니다.

해결책: 시스템의 호스트 이름이 이름 서비스에 나열되어 있는지 확인합니다. 시스템의 호스트 이름이 NIS 또는 NIS+ 이름 서비스에 나열되어 있지 않고 시스템이 계속해서 이 오류 메시지를 출력하면 재부트를 시도합니다.

RPC: Timed out No bootparams (whoami) server responding; still trying... (x86 기반 시스템에만 적용)

원인: 클라이언트가 네트워크에서 부트를 시도하고 있지만 설치 서버의 `/etc/bootparams` 파일에 항목이 있는 시스템을 찾을 수 없습니다.

해결책: 설치 서버에서 `add_install_client`를 사용합니다. 이 명령어를 사용하면 `/etc/bootparams` 파일에 올바른 항목을 추가하여 클라이언트를 네트워크에서 부트할 수 있습니다.

Still trying to find a RPL server... (x86 기반 시스템에만 적용)

원인: 시스템이 네트워크에서 부트를 시도하지만 서버가 이 시스템을 부트하도록 설정되지 않았습니다.

해결책: 설치 서버에서 시스템이 설치되도록 `add_install_client`를 실행합니다. `add_install_client` 명령어가 필요한 네트워크 부트 프로그램을 포함하는 `/rplboot` 디렉토리를 설정합니다.

CLIENT MAC ADDR: FF FF FF FF FF FF (DHCP를 사용하는 네트워크 설치에만 적용)

원인: DHCP 서버가 제대로 구성되어 있지 않습니다. 옵션이나 매크로가 DHCP Manager 소프트웨어에 제대로 정의되어 있지 않은 경우 이 오류가 발생할 수 있습니다.

해결책: DHCP Manager 소프트웨어에서 옵션과 매크로가 제대로 정의되어 있는지 확인합니다. 라우터 옵션이 정의되어 있고 네트워크 설치에 사용할 서브넷에 대해 라우터 옵션 값이 올바른지 확인합니다.

네트워크에서 부트할 때 발생하는 일반적인 문제

시스템이 네트워크에서 부트되지만 지정된 설치 서버가 아닌 시스템에서 부트됩니다.

원인: `/etc/bootparams` 및 `/etc/ethers` 항목이 클라이언트에 대한 다른 시스템에 있습니다.

해결책: 이름 서버에서 설치되는 시스템용 `/etc/bootparams` 항목을 업데이트합니다. 항목은 다음 구문과 일치해야 합니다.

```
install_system root=boot_server:path install=install_server:path
```

또한 `bootparams` 항목이 설치 클라이언트용 서브넷에 있는지 확인합니다.

The system does not boot from the network (DHCP를 사용하는 네트워크 설치에만 적용).

원인: DHCP 서버가 제대로 구성되어 있지 않습니다. 이 오류는 시스템이 DHCP 서버에서 설치 클라이언트로 구성되지 않은 경우에 발생할 수 있습니다.

해결책: DHCP 관리자 소프트웨어에서 클라이언트 시스템의 설치 옵션과 매크로가 정의되어 있는지 확인합니다. 자세한 내용은 **Solaris 10 6/06 설치 설명서: 네트워크 기반 설치의 “DHCP 서비스를 사용하여 시스템 구성 정보를 미리 구성(작업)”**을 참조하십시오.

Solaris OS의 초기 설치

Initial installation fails

해결책: Solaris 설치가 실패하면 설치를 다시 시작해야 합니다. 설치를 다시 시작하려면 Solaris Operating System DVD, the Solaris Software - 1 CD 또는 네트워크를 통해 시스템을 부트합니다.

소프트웨어를 부분적으로 설치한 뒤 Solaris 소프트웨어를 설치 해제할 수 없습니다. 백업을 사용하여 시스템을 복원하거나 Solaris 설치 프로세스를 다시 시작해야 합니다.

/cdrom/10_606/SUNWxxxx/reloc.cpio: Broken pipe

설명: 이 오류 메시지는 정보용이고 설치에 영향을 주지 않습니다. 파이프에 쓰기에 읽기 프로세스가 없을 때 발생합니다.

해결책: 이 메시지를 무시하고 설치를 계속합니다.

WARNING: CHANGE DEFAULT BOOT DEVICE(x86 기반 시스템에만 적용)

원인: 이 메시지는 정보를 제공하는 메시지입니다. 시스템의 BIOS에 설정된 기본 부트 장치가 Solaris 10 3/05 Device Configuration Assistant 디스켓을 사용하여 시스템을 부트해야 하는 장치에 대해 설정되었을 수 있습니다.

해결책: 설치를 계속하고 필요한 경우 Solaris 소프트웨어를 설치한 후 BIOS에 지정된 시스템의 기본 부트 장치를 Solaris 10 3/05 Device Configuration Assistant 디스켓이 필요하지 않은 장치로 변경합니다.

x86 전용 - locale 키워드를 사용하여 초기 설치의 사용자 정의 JumpStart 프로필을 테스트하는 경우 `pfinstall -D` 명령어가 프로필 테스트에 실패합니다. 이에 대한 해결책은 [219 페이지](#) “Solaris OS 업그레이드” 섹션의 오류 메시지 “could not select locale” 을 참조하십시오.

▼ x86: IDE 디스크의 잘못된 블록 확인

IDE 디스크는 Solaris 소프트웨어가 지원하는 다른 장치와 같이 잘못된 블록을 자동으로 매핑하지 않습니다. Solaris 10 6/06를 IDE 디스크에 설치하기 전에 디스크에서 표면 분석을 수행할 수 있습니다. IDE 디스크에서 표면 분석을 수행하려면 다음 절차를 따릅니다.

- 1 설치 매체로 부트합니다.
- 2 설치 유형을 선택하는 프롬프트가 표시되면 옵션 6, 단일 사용자 셸을 선택합니다.
- 3 `format(1M)` 프로그램을 시작합니다.
`format`
- 4 표면 분석을 수행하려는 IDE 디스크 드라이브를 지정합니다.
`cxdy`

cx 제어기 번호
dy 장치 번호

5 fdisk 분할 영역을 가질지 결정합니다.

- Solaris fdisk 파티션이 이미 있는 경우 단계 6로 진행합니다.
- Solaris fdisk 분할 영역이 존재하지 않으면 fdisk 명령어를 사용하여 디스크에 Solaris 분할 영역을 만듭니다.

```
format> fdisk
```

6 표면 분석을 시작하려면 다음을 입력합니다.

```
format> analyze
```

7 현재 설정을 결정하고 다음을 입력합니다.

```
analyze> config
```

8 (선택 사항) 설정을 변경하려면 다음을 입력합니다.

```
analyze> setup
```

9 잘못된 블록을 찾으려면 다음을 입력합니다.

```
analyze> type_of_surface_analysis
```

type_of_surface_analysis 읽기, 쓰기 또는 비교

format이 잘못된 블록을 발견하면 다시 매핑합니다.

10 분석을 종료하려면 다음을 입력합니다.

```
analyze> quit
```

11 다시 매핑할 블록을 지정할 것인지 결정합니다.

- 없는 경우 단계 12로 진행합니다.
- 지정하려면 다음을 입력합니다.

```
format> repair
```

12 포맷 프로그램을 종료하려면 다음을 입력합니다.

```
quit
```

13 다음 명령어를 입력하여 다중 사용자 모드에서 매체를 재시작합니다.

```
# exit
```

Solaris OS 업그레이드

업그레이드할 때 나타나는 오류 메시지

No upgradable disks

원인: /etc/vfstab 파일에 있는 스왑 항목으로 인해 업그레이드가 실패하고 있습니다.

해결책: /etc/vfstab 파일에서 다음 행을 주석 처리합니다.

- 업그레이드되지 않는 디스크의 모든 스왑 파일 및 슬라이스
- 존재하지 않는 스왑 파일
- 사용하지 않는 스왑 슬라이스

usr/bin/bzcat not found

원인: 패치 클러스터가 없어 Solaris Live Upgrade가 실패하고 있습니다.

해결책: Solaris Live Upgrade 설치를 위한 패치가 필요합니다. <http://sunsolve.sun.com>을 참조하여 가장 최근에 업데이트된 패치 목록이 있는지 확인합니다. SunSolve 웹 사이트에서 정보 문서 72099를 검색합니다.

Upgradeable Solaris root devices were found, however, no suitable partitions to hold the Solaris install software were found. Upgrading using the Solaris Installer is not possible. Solaris Software 1 CDROM을 사용하는 업그레이드가 가능할 수 있습니다. (x86 기반 시스템에만 적용)

원인: 충분한 공간이 없기 때문에 Solaris Software - 1 CD를 사용하여 업그레이드할 수 없습니다.

해결책: 업그레이드하려면 512 MB보다 크거나 같은 스왑 슬라이스를 만들거나 Solaris Operating System DVD의 Solaris 설치 프로그램, 네트워크 설치 이미지 또는 JumpStart와 같은 다른 업그레이드 방법을 사용할 수 있습니다.

오류: Could not select locale(x86 기반 시스템에만 적용)

원인: pfinstall -D 명령어를 사용하여 JumpStart 프로필을 테스트할 경우 다음 조건에서 Dry Run 테스트가 실패합니다.

- 프로필에 로캘 키워드가 포함되어 있습니다.
- GRUB 소프트웨어가 포함된 릴리스를 테스트하고 있습니다. Solaris 10 1/06 릴리스부터 시작하여 GRUB 부트 로더에서 GRUB 메뉴로 시스템에 설치된 다른 운영 체제 부팅이 가능해졌습니다.

GRUB 소프트웨어가 추가되어 미니루트가 압축됩니다. 소프트웨어는 더 이상 압축된 미니루트에서 로캘 목록을 찾을 수 없습니다. 미니루트는 가능한 가장 작은 Solaris 루트(/) 파일 시스템이며 Solaris 설치 매체에서 찾을 수 있습니다.

해결책: 다음 단계를 수행합니다. 다음 값을 사용합니다.

- MEDIA_DIR은 /cdrom/cdrom0/
- MINIROOT_DIR은 \$MEDIA_DIR /Solaris_10 6/06/Tools/Boot
- MINIROOT_ARCHIVE는 \$MEDIA_DIR /boot/x86.miniroot
- TEMP_FILE_NAME은 /tmp/test

1. 미니루트 아카이브 압축을 해제합니다.

```
# /usr/bin/gzcat $MINIROOT_ARCHIVE > $TEMP_FILE_NAME
```

2. lofiadm 명령어를 사용하여 미니루트 장치를 만듭니다.

```
# LOFI_DEVICE=/usr/sbin/lofiadm -a $TEMP_FILE_NAME
# echo $LOFI_DEVICE
/dev/lofi/1
```

3. 미니루트를 Miniroot 디렉토리 아래에서 lofi 명령어를 사용하여 마운트합니다.

```
# /usr/sbin/mount -F ufs $LOFI_DEVICE $MINIROOT_DIR
```

4. 프로필을 테스트합니다.

```
# /usr/sbin/install.d/pfinstall -D -c $MEDIA_DIR $path-to-jumpstart_profile
```

5. 테스트가 끝나면 lofi 장치를 마운트 해제합니다.

```
# umount $LOFI_DEVICE
```

6. lofi 장치를 삭제합니다.

```
# lofiadm -d $TEMP_FILE_NAME
```

업그레이드할 때 발생하는 일반적인 문제

The upgrade option is not presented even though there is a version of Solaris software that's upgradable on the system.

원인: 이유 1: /var/sadm 디렉토리가 symlink이거나 다른 파일 시스템으로부터 마운트되었습니다.

해결책: 이유 1: /var/sadm 디렉토리를 루트(/) 또는 /var 파일 시스템으로 이동합니다.

원인: 이유 2: /var/sadm/softinfo/INST_RELEASE 파일이 없습니다.

해결책: 이유 2: 다음 템플리트를 사용하여 새로운 INST_RELEASE 파일을 만듭니다.

```
OS=Solaris
VERSION=x
REV=0
```

x
시스템상의 Solaris 소프트웨어 버전입니다.

원인: 이유 3: SUNWusr이 /var/sadm/softinfo에 없습니다.

해결책: 해결 방법 3: 초기 설치를 수행해야 합니다. Solaris 소프트웨어를 업그레이드할 수 없습니다.

Couldn't shut down or initialize the md driver

해결책: 다음 지침을 따르십시오.

- 파일 시스템이 RAID-1 볼륨이 아니면 vsftab 파일에서 주석 처리합니다.
- 파일 시스템이 RAID-1 볼륨이면 미러를 차단하고 다시 설치합니다. 미러 해제에 대한 자세한 내용은 **Solaris Volume Manager Administration Guide**의 “Removing RAID-1 Volumes (Unmirroring)”를 참조하십시오.

The upgrade fails because the Solaris installation program cannot mount a file system.

원인: 업그레이드 동안 스크립트는 업그레이드되는 루트(/) 파일 시스템의 /etc/vfstab 파일에 나열된 모든 파일 시스템을 마운트하려 시도합니다. 설치 스크립트가 파일 시스템을 마운트할 수 없으면, 업그레이드가 실패하여 종료됩니다.

해결책: 시스템의 /etc/vfstab 파일 내의 모든 파일 시스템이 마운트될 수 있는지 확인하십시오. 마운트될 수 없거나 문제를 야기하는 /etc/vfstab 파일의 파일 시스템을 주석 처리하여 Solaris 설치 프로그램이 업그레이드를 하는 동안에 이들에 대한 마운트 시도를 하지 않도록 합니다. 업그레이드될 소프트웨어를 포함하는 시스템 기반 파일 시스템(예를 들어, /usr)을 주석 처리해서는 안 됩니다.

The upgrade fails

설명: 시스템에 업그레이드를 위한 충분한 공간이 없습니다.

원인: 공간 문제는 40 페이지 “디스크 공간 재할당으로 업그레이드”를 확인하고 자동 레이아웃을 사용하여 공간을 재할당하지 않고 수정할 수 있는지 확인합니다.

RAID-1 볼륨 루트(/) 파일 시스템 업그레이드 문제

해결책: 루트(/) 파일 시스템인 Solaris Volume Manager RAID-1 볼륨을 사용하여 업그레이드하는 데 문제가 있으면 **Solaris Volume Manager Administration Guide**의 25 장, “Troubleshooting Solaris Volume Manager (Tasks)”를 참조하십시오.

▼ 업그레이드 실패 후에 업그레이드 계속 진행

업그레이드가 실패하고 시스템을 소프트웨어 부트할 수 없습니다. 정전이나 네트워크 연결 오류와 같이 사용자가 해결할 수 있는 수준의 오류가 아닙니다.

- 1 **Solaris Operating System DVD, Solaris Software - 1 CD** 또는 네트워크에서 시스템을 재부트합니다.

2 설치를 위한 업그레이드 옵션을 선택합니다.

시스템이 부분적으로 업그레이드되었는지 여부를 Solaris 설치 프로그램이 확인하고 업그레이드를 계속합니다.

x86: GRUB를 사용하는 경우의 Solaris Live Upgrade 문제

x86 기반 시스템에서 Solaris Live Upgrade 및 GRUB 부트 로더를 사용하면 다음 오류가 발생할 수 있습니다.

오류: The media product tools installation directory *path-to-installation-directory* does not exist.

오류: The media *dirctory* does not contain an operating system upgrade image.

설명: luupgrade 명령어를 사용하여 새 부트 환경으로 업그레이드할 때 오류 메시지가 표시됩니다.

원인: 이전 버전의 Solaris Live Upgrade가 사용 중입니다. 시스템에 설치한 Solaris Live Upgrade 패키지가 매체 및 매체 릴리스와 호환되지 않습니다.

해결책: 항상 업그레이드할 릴리스의 Solaris Live Upgrade 패키지를 사용해야 합니다.

예: 다음 예에서 오류 메시지는 시스템의 Solaris Live Upgrade 패키지 버전이 매체에 있는 것과 일치하지 않는 것을 나타냅니다.

```
# luupgrade -u -n s10u1 -s /mnt
Validating the contents of the media </mnt>.
The media is a standard Solaris media.
ERROR: The media product tools installation directory
</mnt/Solaris_10/Tools/Boot/usr/sbin/install.d/install_config> does
not exist.
ERROR: The media </mnt> does not contain an operating system upgrade
image.
```

오류: Cannot find or is not executable: </sbin/biosdev>.

오류: One or more patches required by Solaris Live Upgrade has not been installed.

원인: Solaris Live Upgrade에 필요한 패치 중 하나 이상이 시스템에 설치되어 있지 않습니다. 이 오류 메시지가 누락된 패치를 모두 찾는 것은 아닙니다.

해결책: Solaris Live Upgrade를 사용하기 전에 항상 필요한 패치를 모두 설치해야 합니다. <http://sunsolve.sun.com>을 참조하여 가장 최근에 업데이트된 패치 목록이 있는지 확인합니다. SunSolve 웹 사이트에서 정보 문서 72099를 검색합니다.

오류: Device mapping command `</sbin/biosdev>` failed. **재부트한 다음 다시 시도하십시오.**

원인: 이유 1: 이전 관리 작업 때문에 Solaris Live Upgrade에서 장치를 매핑할 수 없습니다.

해결책: 이유 1: 시스템을 재부트하고 Solaris Live Upgrade를 다시 시도하십시오.

원인: 이유 2: 시스템을 재부트해도 같은 오류 메시지가 표시되는 경우에는 같은 디스크가 두 개 이상 있는 것입니다. 장치 매핑 명령어가 해당 디스크를 구별할 수 없습니다.

해결책: 이유 2: 디스크 중 하나에 새 임시 `fdisk` 분할 영역을 만듭니다. `fdisk(1M)` 매뉴얼 페이지를 참조하십시오. 그런 다음 시스템을 재부트 합니다.

GRUB 메뉴가 포함된 부트 환경을 삭제할 수 없습니다.

원인: Solaris Live Upgrade에는 부트 환경에 GRUB 메뉴가 포함되어 있으면 부트 환경을 삭제할 수 없다는 제한이 적용됩니다.

해결책: `lumake(1M)` 또는 `luupgrade(1M)` 명령어를 사용하여 부트 환경을 다시 사용합니다.

GRUB 메뉴가 포함된 파일 시스템을 실수로 재구성했습니다. 하지만 디스크의 슬라이스는 이전과 같습니다. 예를 들어, 디스크가 다시 슬라이스되지 않았습니다.

원인: GRUB 메뉴를 포함하는 파일 시스템은 시스템을 부트 가능한 상태로 유지하는 데 핵심적인 역할을 합니다. Solaris Live Upgrade 명령어는 GRUB 메뉴를 완전 삭제하지 않습니다. 하지만 실수로 재구성 등의 방법으로 Solaris Live Upgrade 명령어가 아닌 명령어를 사용하여 GRUB 메뉴가 포함된 파일 시스템을 완전 삭제한 경우에는 복구 소프트웨어에서 GRUB 메뉴를 다시 설치하려 합니다. 복구 소프트웨어는 다음으로 재부트할 때 GRUB 메뉴를 다시 같은 파일 시스템에 넣습니다. 예를 들어, 파일 시스템에서 `newfs` 또는 `mkfs` 명령어를 사용하여 실수로 GRUB 메뉴를 완전 삭제하는 경우가 있을 수 있습니다. GRUB 메뉴를 제대로 복원하려면 슬라이스에서 다음 조건을 준수해야 합니다.

- 마운트 가능 파일 시스템을 포함합니다.
- 이전에 슬라이스가 있던 것과 같은 Solaris Live Upgrade 부트 환경의 일부로 남아 있습니다.

시스템을 재부트하기 전에 슬라이스에서 필요한 사항을 보정합니다.

해결책: 시스템을 다시 부트합니다. GRUB 메뉴의 백업 복사본이 자동으로 설치됩니다.

GRUB 메뉴의 `menu.lst` 파일이 실수로 삭제되었습니다.

해결책: 시스템을 다시 부트합니다. GRUB 메뉴의 백업 복사본이 자동으로 설치됩니다.

▼ Veritas VxVm을 실행하는 Solaris Live Upgrade를 사용하여 업그레이드하는 중에 시스템 장애 발생

다음 절차에 따라 업그레이드를 수행하지 않으면 Solaris Live Upgrade를 사용하여 업그레이드를 수행하고 Veritas VxVM을 실행하는 경우 재부트 시 시스템 장애가 발생합니다. 패키지가 Solaris 고급 패키지 작성 지침 사항을 따르지 않으면 문제가 발생합니다.

1 비활성 부트 환경을 만듭니다. Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 “새 부트 환경 만들기”를 참조하십시오.

2 비활성 부트 환경을 업그레이드하기 전에 비활성 부트 환경에서 기존 Veritas 소프트웨어를 비활성화시켜야 합니다.

a. 비활성 부트 환경을 마운트합니다.

```
# lumount inactive_boot_environment_name mount_point
```

예를 들면 다음과 같습니다.

```
# lumount solaris8 /mnt
```

b. vfstab를 포함하는 디렉토리로 변경합니다. 예를 들면 다음과 같습니다.

```
# cd /mnt/etc
```

c. 비활성 부트 환경의 vfstab 파일 사본을 만듭니다. 예를 들면 다음과 같습니다.

```
# cp vfstab vfstab.501
```

d. 복사한 vfstab에서 Veritas 파일 시스템 항목을 모두 주석 처리합니다. 예를 들면 다음과 같습니다.

```
# sed '/vx\dsk/s/^/#/g' < vfstab > vfstab.novxfs
```

각 행의 첫 번째 문자는 해당 행을 주석 행으로 만드는 #으로 변경됩니다. 이 주석 행은 시스템 파일 주석 행과 다르다는 점에 주의하십시오.

e. 변경된 vfstab 파일을 복사합니다. 예를 들면 다음과 같습니다.

```
# cp vfstab.novxfs vfstab
```

f. 비활성 부트 환경의 시스템 파일로 디렉토리를 변경합니다. 예를 들면 다음과 같습니다.

```
# cd /mnt/etc
```

g. 비활성 부트 환경의 시스템 파일 사본을 만듭니다. 예를 들면 다음과 같습니다.

```
# cp system system.501
```

h. “forceload:”를 모두 주석 처리합니다. drv/vx를 포함하는 항목입니다.

```
# sed '/forceload: drv/vx/s/^*/' <system> system.novxfs
```

각 행의 첫 번째 문자는 해당 행을 명령줄로 만드는 *로 변경됩니다. 이 주석 행은 vfstab 파일 주석 행과 다르다는 점에 주의하십시오.

i. Veritas install-db 파일을 만듭니다. 예를 들면 다음과 같습니다.

```
# touch vx/reconfig.d/state.d/install-db
```


j. 비활성 부트 환경을 마운트 해제합니다.

```
# luumount inactive_boot_environment_name
```

3 비활성 부트 환경을 업그레이드합니다. Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 9 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

4 비활성 부트 환경을 활성화시킵니다. Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 “부트 환경 활성화”를 참조하십시오.

5 시스템을 종료합니다.

```
# init 0
```

6 단일 사용자 모드에서 비활성 부트 환경을 부트합니다.

```
OK boot -s
```

“vxvm” 또는 “VXVM”을 포함하는 몇 개의 메시지와 오류 메시지는 무시할 수 있습니다. 비활성 부트 환경이 활성화됩니다.

7 Veritas를 업그레이드합니다.

a. 시스템에서 Veritas VRTSvmsa 패키지를 제거합니다. 예를 들면 다음과 같습니다.

```
# pkgrm VRTSvmsa
```

b. 디렉토리를 Veritas 패키지로 변경합니다.

```
# cd /location_of_Veritas_software
```

c. 시스템에 최신 Veritas 패키지를 추가합니다.

```
# pkgadd -d 'pwd' VRTSvxvm VRTSvmsa VRTSvmdoc VRTSvman VRTSvmdv
```

8 원래의 vfstab 및 시스템 파일을 복원합니다.

```
# cp /etc/vfstab.original /etc/vfstab
# cp /etc/system.original /etc/system
```

9 시스템을 다시 부트합니다.

```
# init 6
```

x86: 기존 Service 분할 영역이 없는 시스템에서 기본적으로 만들지 않는 Service 분할 영역

현재 서비스 또는 진단 분할 영역이 없는 시스템에 Solaris 10 6/06 OS를 설치하는 경우 설치 프로그램에서 기본적으로 서비스 분할 영역을 만들 수 없을 것입니다. Solaris 분할 영역과 같은 디스크에 서비스 분할 영역이 있으면 Solaris 10 6/06 OS를 설치하기 전에 서비스 분할 영역을 다시 작성해야 합니다.

서비스 분할 영역이 있는 시스템에 Solaris 8 2/02 OS를 설치했으면 설치 프로그램이 설치 분할 영역을 보존하지 않았을 수 있습니다. 서비스 분할 영역을 보존하기 위한 fdisk 부트 분할 영역 레이아웃 수동 편집을 수행하지 않은 경우 설치 프로그램에서 설치하는 동안 서비스 분할 영역을 삭제합니다.

주 - Solaris 8 2/02 OS를 설치할 때 서비스 분할 영역을 별도로 보존하지 않았으면 서비스 분할 영역을 다시 작성하고 Solaris 10 6/06 OS로 업그레이드할 수 없습니다.

Solaris 분할 영역이 있는 디스크에 서비스 분할 영역을 두려면 다음 조치 중 하나를 선택합니다.

▼ 네트워크 설치 이미지 또는 Solaris Operating System DVD에서 소프트웨어를 설치하는 방법

네트워크 설치 이미지 또는 네트워크를 통해 Solaris Operating System DVD에서 소프트웨어를 설치하려면 다음 단계를 따르십시오.

- 1 디스크의 내용을 삭제합니다.
- 2 설치하기 전에 시스템의 진단 CD를 사용하여 서비스 분할 영역을 작성합니다.
서비스 분할 영역을 작성하는 방법에 대한 자세한 내용은 하드웨어 설명서를 참조하십시오.
- 3 네트워크에서 시스템을 부트합니다.
Customize fdisk Partitions 화면이 표시됩니다.
- 4 기본 부트 디스크 분할 영역 레이아웃을 로드하려면 기본값을 누릅니다.
설치 프로그램이 서비스 분할 영역을 보존하고 Solaris 분할 영역을 만듭니다.

▼ Solaris Software - 1 CD 또는 네트워크 설치 이미지에서 설치하는 방법

Solaris 설치 프로그램을 사용하여 부트 서버의 Solaris Software - 1 CD 또는 네트워크 설치 이미지에서 설치하려면 다음 단계를 따릅니다.

- 1 디스크의 내용을 삭제합니다.
- 2 설치하기 전에 시스템의 진단 CD를 사용하여 서비스 분할 영역을 작성합니다.
서비스 분할 영역을 작성하는 방법에 대한 자세한 내용은 하드웨어 설명서를 참조하십시오.
- 3 설치 프로그램이 Solaris 분할 영역을 만드는 방법을 선택하라는 메시지를 표시합니다.

- 4 시스템을 부트합니다.
- 5 Use rest of disk for Solaris partition 옵션을 선택합니다.
설치 프로그램이 서비스 분할 영역을 보존하고 Solaris 분할 영역을 만듭니다.
- 6 설치를 완료합니다.

추가 SVR4 패키징 요구 사항(참조)

이 부록은 특히 다른 업체 패키지를 포함하여 패키지를 설치 또는 제거하는 시스템 관리자를 위한 것입니다. 이러한 패키징 요구 사항을 따르면 다음을 수행할 수 있습니다.

- Solaris Live Upgrade로 업그레이드하고 비전역 영역 및 디스크 없는 클라이언트를 만들고 유지할 수 있으므로 현재 실행 중인 시스템을 수정하지 않도록 합니다.
- 사용자 정의 JumpStart와 같은 설치 프로그램을 사용할 때 패키지가 설치를 자동화하는 대화식이 되지 못하도록 합니다.

이 장은 다음 내용으로 구성되어 있습니다.

- 229 페이지 “현재 OS의 수정 방지”.
- 233 페이지 “설치 또는 업그레이드 시 사용자 상호 작용 방지”.
- 234 페이지 “영역에 대한 패키지 매개 변수 설정”

현재 OS의 수정 방지

이 절의 요구 사항을 따르면 현재 실행 중인 OS가 변경되지 않고 유지됩니다.

절대 경로 사용

운영 체제 설치가 성공하려면 패키지가 Solaris Live Upgrade 비활성 부트 환경 등의 대체 루트(/) 파일 시스템을 인지하고 올바르게 처리해야 합니다.

패키지는 pkgmap 파일(패키지 맵)에 절대 경로를 포함할 수 있습니다. 이 파일이 존재할 경우 해당 파일은 pkgadd 명령어의 -R 옵션에 상대적으로 작성됩니다. 절대 및 상대(재할당 가능) 경로를 모두 포함하는 패키지를 대체 루트(/) 파일 시스템에 설치할 수도 있습니다.

\$PKG_INSTALL_ROOT는 pkgadd에 의해 설치될 때 모든 경로가 제대로 해결되도록 절대 및 재할당 파일 모두의 앞에 붙습니다.

pkgadd -R 명령어 사용

pkgadd -R 옵션을 사용하여 설치되거나 pkgrm -R 옵션을 사용하여 제거되는 패키지는 현재 실행 중인 시스템을 변경하지 말아야 합니다. 이 기능은 사용자 정의 JumpStart, Solaris Live Upgrade, 비전역 영역 및 디스크 없는 클라이언트에서 사용됩니다.

pkgadd 명령어 -R 옵션으로 설치되거나 pkgrm 명령어 -R 옵션을 사용하여 제거되는 모든 패키지에 포함되는 모든 프로시저 스크립트는 현재 실행 중인 시스템을 변경하지 말아야 합니다. 사용자가 제공하는 모든 설치 스크립트는 접두어가 \$PKG_INSTALL_ROOT 변수인 모든 디렉토리 또는 파일을 참조해야 합니다. 패키지는 \$PKG_INSTALL_ROOT 접두어를 가진 모든 디렉토리 및 파일을 작성해야 합니다. 패키지는 \$PKG_INSTALL_ROOT 접두어 없이 디렉토리를 제거하지 말아야 합니다.

표 B-1은 스크립트 구문의 예를 제공합니다.

표 B-1 설치 스크립트 구문의 예

스크립트 유형	올바른 구문	잘못된 구문
Bourne 셸 “if” 명령문 조각	<pre>if [-f \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf] ; then</pre>	<pre>if [-f /etc/myproduct.conf] ; \ then</pre>
파일 제거	<pre>/bin/rm -f \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf</pre>	<pre>/bin/rm -f /etc/myproduct.conf</pre>
파일 변경	<pre>echo "test=no" > \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf</pre>	<pre>echo "test=no" > \ /etc/myproduct.conf</pre>

\$PKG_INSTALL_ROOT와 \$BASEDIR 사이의 차이점 개요

\$PKG_INSTALL_ROOT는 패키지를 추가할 컴퓨터의 루트(/) 파일 시스템의 위치입니다. 위치는 pkgadd 명령어의 -R 인자로 설정됩니다. 예를 들어, 다음 명령어가 호출되면 \$PKG_INSTALL_ROOT가 패키지를 설치하는 동안 /a가 됩니다.

```
# pkgadd -R /a SUNWvsvm
```

\$BASEDIR은 위치 변경이 가능한 패키지 객체가 설치된 위치 변경이 가능한 기본 디렉토리를 가리킵니다. 위치 변경이 가능한 객체만 여기에 설치됩니다. 위치 변경이 불가능한 객체(pkgmap 파일의 절대 경로를 가진 객체)는 항상 비활성 부트 환경에 상대적으로 설치되지만 실제로 \$BASEDIR에 상대적이지 않습니다. 패키지에 위치 변경이 가능한 객체가 없는 경우 해당 패키지는 절대 패키지(즉, 위치 변경이 불가능한 패키지)라고 하며 \$BASEDIR은 정의되지 않고 패키지 프로시저 스크립트에 사용할 수 없습니다.

예를 들어, 패키지의 pkgmap 파일은 다음 두 항목을 가지는 것으로 가정합니다.

```
1 f none sbin/ls 0555 root sys 3541 12322 1002918510
1 f none /sbin/ls2 0555 root sys 3541 12322 2342423332
```

pkginfo 파일은 \$BASEDIR에 대한 다음 사양을 갖습니다.

```
BASEDIR=/opt
```

이 패키지가 다음 명령어로 설치되면 ls는 /a/opt/sbin/ls에 설치되지만 ls2는 /a/sbin/ls2에 설치됩니다.

```
# pkgadd -R /a SUNWtest
```

스크립트 작성 지침

패키지 프로시저 스크립트는 반드시 현재 실행 중인 OS와 독립적이어서 OS가 수정되지 않도록 해야 합니다. 프로시저 스크립트는 패키지 설치 및 제거 동안 특정 지점에서 발생하는 작업을 정의합니다. 이러한 미리 지정된 이름으로 만들 수 있는 프로시저 스크립트는 preinstall, postinstall, preremove 및 postremove 등 4개입니다.

표 B-2 스크립트 작성 지침

지침	Solaris Live Upgrade에 미치는 영향	비전역 영역에 미치는 영향
스크립트는 Bourne 셸(/bin/sh)에서 작성되어야 합니다. Bourne 셸은 프로시저 스크립트를 실행하기 위해 pkgadd 명령어가 사용하는 해석기입니다.	X	X
스크립트는 어떤 프로세스도 시작 또는 정지하지 말아야 합니다. 또는 운영 체제 종속적이고 현재 실행 중인 시스템에 대한 정보를 보고하는 ps 또는 truss와 같은 명령어의 출력에 의존하지 말아야 합니다.	X	X
스크립트는 expr, cp, ls 및 셸 스크립트 작성을 용이하게 하는 기타 명령어와 같은 기타 표준 UNIX 명령어를 자유롭게 사용합니다.	X	X
스크립트가 호출하는 모든 명령어는 패키지가 해당하는 모든 릴리스에서 실행해야 하기 때문에 지원되는 모든 릴리스에서 사용 가능해야 합니다. 따라서 Solaris 8 릴리스 이후에 추가 또는 제거된 명령어는 사용할 수 없습니다.	X	
특정 명령어 또는 옵션이 Solaris 8, 9 또는 10 릴리스에서 지원되는지를 확인하려면 http://docs.sun.com 의 Solaris Reference Manual AnswerBook 의 특정 버전을 참조하십시오.		

디스크 없는 클라이언트 호환성 유지 관리

패키지가 패키지 자체가 제공하는 명령어를 실행하면 안 됩니다. 이는 디스크 없는 클라이언트 호환성을 유지하고 아직 설치되지 않은 공유 라이브러리가 필요할 수 있는 명령어 실행을 방지하기 위한 것입니다.

패키지 확인

모든 패키지는 `pkgchk` 검증을 통과해야 합니다. 패키지가 작성되면 설치되기 전에 다음 명령어로 검사해야 합니다.

```
# pkgchk -d dir_name pkg_name
```

`dir_name` 패키지가 상주하는 디렉토리의 이름을 지정합니다.

`pkg_name` 패키지의 이름을 지정합니다.

예 B-1 패키지 테스트

패키지가 작성된 후 `pkgadd`에 대한 `-R dir_name` 옵션을 사용하여 대체 루트(/) 위치로 설치하여 테스트해야 합니다. 패키지가 설치된 후 다음 예에서와 같이 `pkgchk`를 사용하여 정확성을 검사해야 합니다.

```
# pkgadd -d . -R /a SUNWvxvm
```

```
# pkgchk -R /a SUNWvxvm
```

오류가 표시되어서는 안 됩니다.

예 B-2 /export/SUNWvxvm에서 패키지 테스트

패키지가 /export/SUNWvxvm에 있는 경우 다음 명령을 실행해야 합니다.

```
# pkgchk -d /export SUNWvxvm
```

오류가 표시되어서는 안 됩니다.

파일을 작성, 수정 및 삭제할 때 기타 명령어를 사용하여 패키지를 확인할 수 있습니다. 다음 명령어를 예로 들 수 있습니다.

- 예를 들어, `dircmp` 또는 `fssnap` 명령어는 패키지가 제대로 작동하는지 검증하는 데 사용할 수 있습니다.
- 또한 `ps` 명령어는 데몬이 패키지에 의해 정지 또는 시작되지 않았음을 확인하여 데몬 적합성을 테스트하기 위해 사용될 수 있습니다.
- `truss`, `pkgadd -v` 및 `pkgrm` 명령어는 런타임 패키지 설치 적합성을 테스트할 수 있지만 일부 상황에서 작동하지 않을 수 있습니다. 다음 예에서, `truss` 명령어는 모든 읽기 전용, 비`$TMPDIR` 액세스를 제거하고 지정된 비활성 부트 환경 내에 있지 않는 경로에 대한 비읽기 전용 액세스만 보여 줍니다.

```
# TMPDIR=/a; export TMPDIR
```

```
# truss -t open /usr/sbin/pkgadd -R ${TMPDIR} SUNWvxvm \
2>&1 > /dev/null | grep -v O_RDONLY | grep -v \
'open("${TMPDIR}'
```


설치 또는 업그레이드 시 사용자 상호 작용 방식

패키지는 다음 Solaris 유틸리티를 사용할 경우 사용자에게 정보를 묻는 메시지가 표시되지 않고 추가 또는 제거되어야 합니다.

- 사용자 정의 JumpStart 프로그램
- Solaris Live Upgrade
- Solaris 설치 프로그램 프로그램
- Solaris 영역

사용자 상호 작용 없이 설치할 수 있도록 패키지를 테스트하기 위해 `pkgadd` 명령어 `-a` 옵션으로 새 관리 파일을 설정할 수 있습니다. `-a` 옵션은 기본 관리 파일 대신 사용될 설치 관리 파일을 정의합니다. 기본 파일을 사용하면 사용자에게 추가 정보를 묻는 메시지가 표시될 수 있습니다. 이들 검사를 무시하고 사용자 확인이 없이 패키지를 설치해야 함을 `pkgadd`에게 알려주는 관리 파일을 만들 수 있습니다. 자세한 내용은 매뉴얼 페이지 `admin(4)` 또는 `pkgadd(1M)`을 참조하십시오.

다음 예는 `pkgadd` 명령어가 관리 파일을 사용하는 방법을 보여 줍니다.

- 관리 파일이 제공되지 않으면 `pkgadd`는 `/var/sadm/install/admin/default`를 사용합니다. 이 파일을 사용하면 사용자 상호 작용이 생길 수 있습니다.

```
# pkgadd
```

- 상대 관리 파일이 명령줄에 제공되면 `pkgadd`는 `/var/sadm/install/admin`에서 파일 이름을 찾아 사용합니다. 이 예에서 상대 관리 파일의 이름은 `nocheck`이고 `pkgadd`는 `/var/sadm/install/admin/nocheck`를 찾습니다.

```
# pkgadd -a nocheck
```

- 절대 파일이 제공되면 `pkgadd`가 해당 파일을 사용합니다. 이 예에서 `pkgadd`는 `/tmp`에서 `nocheck` 관리 파일을 찾습니다.

```
# pkgadd -a /tmp/nocheck
```

예 B-3 설치 관리 파일

다음은 `pkgadd` 유틸리티와의 사용자 상호 작용이 거의 필요 없는 설치 관리 파일의 예입니다. 패키지가 시스템에서 사용 가능한 공간보다 더 많은 공간을 필요로 하는 경우를 제외하고 `pkgadd` 유틸리티는 이 파일을 사용하고 사용자에게 추가 정보를 묻는 메시지를 표시하지 않고 패키지를 설치합니다.

```
mail=
instance=overwrite
partial=nocheck
runlevel=nocheck
idepend=nocheck
space=ask
```

예 B-3 설치 관리 파일 (계속)

```
setuid=nocheck
confiict=nocheck
action=nocheck
basedir=default
```

영역에 대한 패키지 매개 변수 설정

패키지에는 비전역 영역이 설치된 시스템에서 해당 내용이 배포되고 표시되는 방식을 제어하는 매개 변수가 있습니다. `SUNW_PKG_ALLZONES`, `SUNW_PKG_HOLLOW` 및 `SUNW_PKG_THISZONE` 패키지 매개 변수는 영역이 설치된 시스템의 패키지 특성을 정의합니다. 비전역 영역이 설치된 시스템에서 패키지를 관리할 수 있도록 이 매개 변수를 설정해야 합니다.

다음 표에는 패키지 매개 변수 설정을 위한 네 가지 유효한 조합이 나열되어 있습니다. 다음 표에 나열되지 않은 설정 조합을 선택한 경우, 해당 설정은 무효화되고 패키지 설치에 실패합니다.

주 - 세 개의 패키지 매개 변수 모두를 설정해야 합니다. 세 개의 패키지 매개 변수 모두를 비워둘 수 있습니다. 패키지 도구는 누락된 영역 패키지 매개 변수를 설정이 “false” 인 것으로 해석하지만 매개 변수를 설정하지 않는 것은 권장하지 않습니다. 세 개의 패키지 매개 변수 모두를 설정하여 패키지를 설치하거나 제거할 때의 패키지 도구의 정확한 동작을 지정합니다.

표 B-3 영역에 대한 유효한 패키지 매개 변수 설정

<code>SUNW_PKG_ALLZONES</code> 설정	<code>SUNW_PKG_HOLLOW</code> 설정	<code>SUNW_PKG_THISZONE</code> 설정	패키지 설명
false	false	false	<p>모든 영역 패키지 매개 변수에 대한 값을 지정하지 않는 패키지에 대한 기본 설정입니다.</p> <p>이러한 설정의 패키지는 전역 영역 또는 비전역 영역에 설치할 수 있습니다.</p> <ul style="list-style-type: none"> ■ <code>pkgadd</code> 명령어를 전역 영역에서 실행하면 패키지는 전역 영역 및 모든 비전역 영역에 설치됩니다. ■ <code>pkgadd</code> 명령어를 비전역 영역에서 실행하면 패키지는 비전역 영역에만 설치됩니다. <p>두 경우 모두, 패키지가 설치된 모든 영역에서 패키지의 전체 내용이 표시됩니다.</p>

표 B-3 영역에 대한 유효한 패키지 매개 변수 설정 (계속)

SUNW_PKG_ALLZONES 설정	SUNW_PKG_HOLLOW 설정	SUNW_PKG_THISZONE 설정	패키지 설명
false	false	true	<p>이러한 설정의 패키지는 전역 영역 또는 비전역 영역에 설치될 수 있습니다. 설치 이후 새 비전역 영역이 작성되면 패키지는 해당하는 새 비전역 영역에 전파되지 않습니다.</p> <ul style="list-style-type: none"> ■ pkgadd 명령어를 전역 영역에서 실행하면 패키지가 전역 영역에만 설치됩니다. ■ pkgadd 명령어를 비전역 영역에서 실행하면 패키지가 비전역 영역에만 설치됩니다. <p>두 경우 모두, 패키지가 설치된 영역에서 패키지의 전체 내용이 표시됩니다.</p>
true	false	false	<p>이러한 설정을 가진 패키지는 반드시 다음과 같은 요구 사항에 부합되어야 합니다.</p> <ul style="list-style-type: none"> ■ 모든 영역에서 반드시 동일한 패키지가 존재해야 합니다. ■ 모든 영역에서 패키지 버전은 반드시 일치해야 합니다. ■ 모든 영역에서 패키지에 대한 임의의 패치는 반드시 존재하며 일치해야 합니다. <p>이러한 설정을 가진 패키지는 전역 영역에서 pkgadd 명령을 실행해야만 설치될 수 있습니다. 비전역 영역에서 pkgadd 명령을 실행하여 이 패키지를 설치하기 위한 모든 시도가 실패합니다.</p> <p>전역 영역에서 pkgadd 명령을 실행한 경우 패키지는 먼저 전역 영역에 설치된 다음 모든 비전역 영역에 설치됩니다. 패키지의 전체 내용이 모든 영역에 표시됩니다.</p>

표 B-3 영역에 대한 유효한 패키지 매개 변수 설정 (계속)

SUNW_PKG_ALLZONES 설정	SUNW_PKG_HOLLOW 설정	SUNW_PKG_THISZONE 설정	패키지 설명
true	true	false	<p>이러한 설정의 패키지는 전역 관리자에 의해 전역 영역에만 설치될 수 있습니다. pkgadd 명령어를 실행하면 패키지의 내용이 전역 영역에 전체 설치됩니다. 패키지에 이 값으로 설정된 패키지 매개 변수가 있는 경우에는 패키지 내용 자체가 비전역 영역에서 전달되지 않습니다. 패키지가 설치된 것처럼 보이도록 하는 데 필요한 패키지 설치 정보만이 모든 비전역 영역에 설치됩니다. 이렇게 하면 이 패키지에 종속된 다른 패키지의 설치가 가능해집니다. “비어 있는” 패키지에 대한 자세한 정보는 System Administration Guide: Solaris Containers-Resource Management and Solaris Zones의 23 장, “About Packages and Patches on a Solaris System with Zones Installed (Overview)”를 참조하십시오.</p> <p>패키지 종속성 검사를 위해 패키지가 모든 영역에 설치된 것처럼 나타납니다.</p> <ul style="list-style-type: none"> ■ 전역 영역에서는 패키지의 전체 내용이 표시됩니다. ■ 전체 루트 비전역 영역에서는 패키지의 전체 내용이 표시되지 않습니다. ■ 비전역 영역이 전역 영역에서 파일 시스템을 상속하는 경우, 해당 파일 시스템에 설치된 패키지가 비전역 영역에 표시됩니다. 패키지에 의해 전달된 다른 모든 파일은 비전역 영역 내에서 표시되지 않습니다. <p>예를 들어 희소 루트 비전역 영역은 전역 영역과 특정 디렉토리를 공유합니다. 이 디렉토리는 읽기 전용입니다. 희소 루트 비전역 영역은 다른 영역과 /platform 파일 시스템을 공유합니다. 다른 예는 부팅 하드웨어에만 관련된 파일을 배포하는 패키지입니다.</p> <p>주 - 비전역 영역에 패키지를 설치하기 위한 모든 시도가 실패합니다.</p>
설명			자세한 내용
패키지 및 영역에 대한 자세한 내용			System Administration Guide: Solaris Containers-Resource Management and Solaris Zones 의 23 장, “About Packages and Patches on a Solaris System with Zones Installed (Overview)”

설명	자세한 내용
희소 및 전체 루트 영역의 개요	System Administration Guide: Solaris Containers-Resource Management and Solaris Zones 의 16 장, “Introduction to Solaris Zones”
패키지 특성 및 매개 변수에 대한 정보	pkginfo(4)
패키지 매개 변수 값 표시에 대한 정보	pkgparam(1)

배경 정보

다음 참조는 패키징 요구 사항과 특정 명령어 구문에 대한 배경 정보를 제공합니다.

패키징 요구 사항과 용어 정의에 대한 자세한 정보	Application Packaging Developer’s Guide 의 6 장, “Advanced Techniques for Creating Packages”
패키지 추가 및 제거와 설치 관리 파일에 대한 기본 정보	System Administration Guide: Basic Administration 의 16 장, “Managing Software (Overview)”
이 부록에서 참조하는 특정 명령어에 대한 자세한 내용은 다음 매뉴얼 페이지를 참조하십시오.	dircmp(1), fssnap(1M), ps(1), 또는 truss(1) pkgadd(1M), pkgchk(1M), 또는 pkgrm(1M)
Solaris Live Upgrade의 개요	Solaris 10 6/06 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획의 6 장, “Solaris Live Upgrade(개요)”
사용자 정의 JumpStart의 개요	5 장
Solaris 영역의 개요	System Administration Guide: Solaris Containers-Resource Management and Solaris Zones 의 16 장, “Introduction to Solaris Zones”

용어집

3DES	Triple-Data Encryption Standard의 약자입니다. 168비트의 키 길이를 제공하는 대칭 키 암호화 방법입니다.
AES	Advanced Encryption Standard의 약자입니다. 대칭 128비트 블록 데이터 암호화 기술입니다. 미국 정부는 2000년 10월 알고리즘의 Rijndael 변형을 암호화 표준으로 채택했습니다. AES가 정부 표준으로 DES 암호화를 대체합니다.
bootlog-cgi	WAN 부트 설치 중에 웹 서버가 원격 클라이언트 부트 및 설치 콘솔 메시지를 수집 및 저장할 수 있게 해주는 CGI 프로그램입니다.
certstore	특정 클라이언트 시스템의 디지털 인증서를 포함하는 파일입니다. SSL 협상 중에 클라이언트는 인증서 파일을 서버에 제공하라는 요청을 받을 수 있습니다. 서버에서는 이 파일을 사용하여 클라이언트 ID를 확인합니다.
CGI	Common Gateway Interface의 약자입니다. 외부 프로그램이 HTTP 서버와 통신하기 위한 인터페이스입니다. CGI를 사용하기 위해 작성한 프로그램을 CGI 프로그램 또는 CGI 스크립트라고 합니다. CGI 프로그램은 서버에서 정상적으로 처리 또는 구문 분석하지 않는 출력을 구문 분석하거나 양식을 처리합니다.
DES	Data Encryption Standard의 약자입니다. 1975년에 개발되어서 1981년에 ANSI에 의해 ANSI X3.92로 표준화된 대칭 키 암호화 방법입니다. DES에서는 56비트 키를 사용합니다.
DHCP	Dynamic Host Configuration Protocol의 약자입니다. 응용 프로그램 계층 프로토콜로서 TCP/IP 네트워크 상의 개별 컴퓨터 또는 클라이언트가 지정된 중앙 관리되는 DHCP 서버로부터 IP 주소와 기타 네트워크 구성 정보를 추출할 수 있도록 합니다. DHCP는 대형 IP 네트워크를 유지 및 관리하는 총 비용을 절감시켜 줍니다.
암호화	정보를 알아볼 수 없도록 하여 권한이 없는 사용자가 정보를 사용하지 못하게 하는 과정입니다. 암호화는 정보를 해독하는 데 사용되는 키라고 하는 코드를 기반으로 합니다. 암호 해독도 참조하십시오.
/etc	코어 시스템 구성 파일과 유지 보수 명령을 포함하는 디렉토리입니다.
/etc/netboot 디렉토리	WAN 부트 설치에 필요한 클라이언트 구성 정보와 보안 데이터가 포함된 WAN 부트 서버의 디렉토리입니다.

/export	네트워크의 다른 시스템과 공유되는 OS 서버의 파일 시스템입니다. 예를 들어, /export 파일 시스템은 네트워크 상의 루트(/) 파일 시스템, 디스크 없는 클라이언트에 대한 스왑 공간, 사용자에 대한 홈 디렉토리를 포함할 수 있습니다. 디스크 없는 클라이언트는 부트 및 실행을 위해 OS 서버에 있는 /export 파일 시스템에 의존합니다.
fdisk 분할 영역	x86 기반 시스템의 특정 운영 체제 전용인 디스크 드라이브의 논리적 분할 영역입니다. Solaris 소프트웨어를 설치하려면 x86 기반 시스템에서 Solaris fdisk 분할 영역을 적어도 하나 설정해야 합니다. x86 기반 시스템은 디스크에서 최고 4개의 fdisk 분할 영역을 허용합니다. 개별 운영 체제를 보유하기 위해 이러한 분할 영역을 사용할 수 있습니다. 각 운영 체제는 고유한 fdisk 분할 영역에 위치해야 합니다. 시스템은 디스크당 하나의 Solaris fdisk 분할 영역만 가질 수 있습니다.
파일 시스템	SunOS™ 운영 체제에서 사용자가 액세스할 수 있는 파일 및 디렉토리의 트리 구조 네트워크입니다.
GRUB	x86 전용: GNU GRUB(Grand Unified Bootloader)는 단순 메뉴 인터페이스가 있는 오픈 소스 부트로더입니다. 메뉴에는 시스템에 설치된 운영 체제의 목록이 표시됩니다. GRUB를 사용하면 Solaris OS, Linux 또는 Microsoft Windows 등의 다양한 운영 체제를 쉽게 부트할 수 있습니다.
GRUB 주 메뉴	x86 전용: 시스템에 설치된 운영 체제의 목록을 표시하는 부트 메뉴이 메뉴에서는 BIOS 또는 fdisk 분할 영역 설정을 수정하지 않고 운영 체제를 쉽게 부트할 수 있습니다.
GRUB 편집 메뉴	x86 전용: GRUB 주 메뉴의 하위 메뉴인 부트 메뉴 GRUB 명령이 이 메뉴에 표시됩니다. 이 명령을 편집하여 부트 동작을 변경할 수 있습니다.
HMAC	메시지 인증을 위해 입력한 해싱 방법입니다. HMAC는 비밀 공유 키와 조합하여 MD5 또는 SHA-1과 같은 반복 암호화 해시 기능과 함께 사용합니다. 기본 해시 기능의 등록 정보에 따라 HMAC의 암호화 강도가 달라집니다.
HTTP	(하이퍼텍스트 전송 프로토콜) (n). 원격 호스트에서 하이퍼텍스트 객체를 불러오는 인터넷 프로토콜입니다. 이 프로토콜은 TCP/IP를 기반으로 합니다.
HTTPS	SSL(Secure Sockets Layer)을 사용하여 구현된 보안 버전의 HTTP입니다.
IP 주소	Internet Protocol Address의 약자로서 TCP/IP에서 네트워크의 각 호스트를 식별하는 고유한 32비트 번호입니다. IP 주소는 마침표로 구분되는 숫자 4개로 구성됩니다(예를 들어, 192.168.0.0). 흔히 IP 주소의 각 부분은 0에서 225 사이의 숫자를 사용합니다. 그러나 첫 번째 숫자는 224 이하여야 하고 마지막 숫자는 0일 수 없습니다. IP 주소는 논리적으로 네트워크(지역 번호와 유사)와 네트워크상의 로컬 시스템(전화 번호와 유사)이라는 다음 두 부분으로 나뉩니다. 예를 들어, 클래스 A IP 주소는 "network.local.local.local"이고 클래스 C IP 주소는 "network.network.network.local"을 나타냅니다.

클래스	범위(<i>xxx</i> 는 0에서 255 사이의 숫자)	사용 가능 IP 주소의 숫자
클래스 A	1. <i>xxx.xxx.xxx</i> - 126. <i>xxx.xxx.xxx</i>	16,000,000 이상
클래스 B	128.0. <i>xxx.xxx</i> - 191.255. <i>xxx.xxx</i>	65,000 이상
클래스 C	192.0.0. <i>xxx</i> - 223.255.255. <i>xxx</i>	256

IPv6

IPv6은 현재 버전인 IPv4(버전 4)에서 혁신적으로 향상이 이루어지도록 설계된 새로운 버전(버전 6)의 인터넷 프로토콜(IP)입니다. 정의된 전환 메커니즘을 사용하여 IPv6을 배치하면 현재 작동이 중단되지 않습니다. 또한 IPv6은 새 인터넷 기능을 위한 플랫폼을 제공합니다.

IPv6은 **System Administration Guide: IP Services**의 제1부, “Introducing System Administration: IP Services”에서 자세히 설명합니다.

JumpStart 디렉토리

사용자 정의 JumpStart 설치용 프로필 디스켓을 사용할 때 JumpStart 디렉토리는 모든 핵심 사용자 정의 JumpStart 파일을 포함하는 디스켓상의 루트 디렉토리입니다. 사용자 정의 JumpStart 설치용 프로필 서버를 사용할 때 JumpStart 디렉토리는 모든 핵심 사용자 정의 JumpStart 파일을 포함하는 서버상의 디렉토리입니다.

JumpStart 설치

Solaris소프트웨어가 출하시 설치된 JumpStart 소프트웨어를 사용하여 시스템에 자동으로 설치되는 설치 유형입니다.

keystore

클라이언트와 서버에서 공유하는 키가 포함된 파일입니다. WAN 부트 설치 중에 클라이언트 시스템은 키를 사용하여 서버에서 전송된 데이터와 파일을 해독하거나 무결성을 검증합니다.

LAN

Local Area Network의 약자입니다. 하드웨어 및 소프트웨어를 연결하여 통신할 수 있는 근거리 컴퓨터 시스템 그룹입니다.

LDAP

Lightweight Directory Access Protocol은 LDAP 이름 지정 서비스 클라이언트와 서버가 서로 통신하기 위해 사용하는 표준 확장 가능 액세스 프로토콜입니다.

MD5

Message Digest 5의 약자로서 디지털 서명을 포함하여 메시지 인증에 사용되는 반복적인 암호화 해시 기능입니다. 이 기능은 1991년 Rivest가 개발했습니다.

menu.lst 파일

x86 전용: 시스템에 설치된 모든 운영 체제의 목록을 표시하는 파일이 파일의 내용에 따라 GRUB 메뉴에 표시되는 운영 체제의 목록이 결정됩니다. GRUB 메뉴에서는 BIOS 또는 fdisk 분할 영역 설정을 수정하지 않고 운영 체제를 쉽게 부트할 수 있습니다.

NIS

SunOS 4.0(최소) 네트워크 정보 서비스입니다. NIS는 네트워크상의 시스템과 사용자에게 대한 핵심 정보를 포함하는 분산 네트워크 데이터베이스입니다. NIS 데이터베이스는 마스터 서버와 모든 슬레이브 서버에 저장됩니다.

NIS+	SunOS 5.0(최소) 네트워크 정보 서비스입니다. NIS+는 NIS, SunOS 4.0(최소) 네트워크 정보 서비스를 대체합니다.
/opt	타사 소프트웨어 및 번들화되지 않은 소프트웨어에 대한 마운트 지점을 포함하는 파일 시스템입니다.
OS 서버	네트워크의 시스템에 서비스를 제공하는 시스템입니다. 디스크 없는 클라이언트에 서브하려면 OS 서버에 각 디스크 없는 클라이언트의 루트(/) 파일 시스템과 스왑 공간(/export/root, /export/swap)을 위한 디스크 공간을 따로 두어야 합니다.
Patch Analyzer	수동으로 또는 Solaris 설치 프로그램의 일부로 실행할 수 있는 스크립트입니다. Patch Analyzer는 시스템에 대한 분석을 수행하여 Solaris 업데이트로 업그레이드하여 제거할 패치를 결정합니다.
Power Management	<p>시스템의 유휴 상태가 30분 이상 지속되면 시스템 상태를 자동 저장한 뒤 시스템을 끄는 소프트웨어입니다. 미국 환경 보호 기관 Energy Star 지침의 버전 2(예를 들어 sun4u SPARC 시스템)를 준수하는 시스템에 Solaris 소프트웨어를 설치할 때 Power Management 소프트웨어는 기본적으로 설치됩니다. 이후 재부트하면 Power Management 소프트웨어를 활성화 또는 비활성화할 것인지를 묻는 메시지가 나타납니다.</p> <p>Energy Star 지침에 의하면 시스템 또는 모니터가 비활성화되면 자동으로 “휴면 상태”(30와트 미만의 전력을 소모)에 들어가야 합니다.</p>
RAID-0 볼륨	스트라이프 또는 연결일 수 있는 볼륨 클래스입니다. 이 구성 요소는 하위 미러라고도 합니다. 스트라이프 또는 연결은 미러를 위한 기본 빌딩 블록입니다.
RAID-1 볼륨	여러 복사본을 유지하여 데이터를 복제하는 볼륨 클래스입니다. RAID-1 볼륨은 하위 미러라고 하는 하나 이상의 RAID-0 볼륨으로 구성됩니다. RAID-1 볼륨을 미러라고도 합니다.
rules.ok 파일	rules 파일의 생성된 버전입니다. rules.ok 파일은 시스템을 프로필에 일치시키기 위해 사용자 정의 JumpStart 설치 소프트웨어를 필요로 합니다. rules.ok 파일을 작성하기 위해 check 스크립트를 반드시 사용해야 합니다.
rules 파일	자동으로 설치하려는 각 시스템 그룹(또는 단일 시스템)에 대한 규칙을 포함하는 텍스트 파일입니다. 각 규칙은 하나 이상의 시스템 속성에 기반한 시스템 그룹을 구별합니다. rules 파일은 각 그룹을 프로필에 연결합니다. 프로필은 그룹의 각 시스템에 Solaris 소프트웨어를 설치할 방법을 정의하는 텍스트 파일입니다. rules 파일은 사용자 정의 JumpStart 설치에 사용됩니다. 프로필도 참조하십시오.
SHA1	Secure Hashing Algorithm의 약자로 2 ⁶⁴ 이하의 입력 길이에서 작동하여 메시지 다이제스트를 생성하는 알고리즘입니다.
Solaris DVD 또는 CD 이미지	시스템에 설치된 Solaris 소프트웨어입니다. 이 소프트웨어는 Solaris DVD 또는 CD에서 액세스하거나 Solaris DVD 또는 CD 이미지를 복사한 설치 서버의 하드 디스크에서 액세스할 수 있습니다.

Solaris Flash	사용자가 마스터 시스템이라고 하는 시스템의 파일의 아카이브를 만들 수 있도록 하는 Solaris 설치 프로그램입니다. 그 다음 아카이브를 다른 시스템 설치에 사용하여 다른 시스템이 구성 면에서 마스터 시스템과 동일하도록 합니다. 아카이브도 참조하십시오.
Solaris Live Upgrade	활성화된 부트 환경이 실행 중인 상태에서 복제 부트 환경을 업그레이드할 수 있어 생산 환경의 중단 시간을 없앨 수 있는 업그레이드 방법입니다.
Solaris 설치 프로그램	Solaris 소프트웨어 및 타사 소프트웨어 설치를 단계별로 안내하는 마법사 패널을 사용하는 그래픽 사용자 인터페이스(GUI) 또는 명령줄 인터페이스(CLI) 설치 프로그램입니다.
Solaris 영역	운영 체제 서비스를 가상화하고 실행 중 응용 프로그램에 대하여 고립되고 안전한 환경을 제공하는 데 사용되는 소프트웨어 분할 기술입니다. 비전역 영역을 작성할 경우 프로세스가 다른 모든 영역에서 고립되는 응용 프로그램 실행 환경을 만듭니다. 이 고립으로 한 영역에서 실행 중인 프로세스가 다른 영역에서 실행 중인 프로세스를 모니터링하거나 영향을 미칠 수 없게 됩니다. 전역 영역 및 비전역 영역을 참조하십시오.
SSL (Secure Sockets Layer)	보안 버전의 HTTP인 HTTPS를 구현하기 위해 사용한 두 당사자(클라이언트와 서버)간의 보안 연결을 설정하는 소프트웨어 라이브러리입니다.
sysidcfg 파일	시스템을 미리 구성하는 특수 시스템 구성 키워드 집합을 지정하는 파일입니다.
truststore	하나 이상의 디지털 인증서가 포함된 파일입니다. WAN 부트 설치 중에 클라이언트 시스템은 truststore 파일의 데이터를 확인하여 설치 수행을 시도하는 서버의 ID를 확인합니다.
URL	Uniform Resource Locator의 약자입니다. 문서를 요청하기 위해 서버와 클라이언트에서 사용한 주소 지정 시스템입니다. URL을 위치라고도 합니다. URL의 형식은 <i>protocol://machine:port/document</i> 입니다. 샘플 URL은 http://www.example.com/index.html 입니다.
/usr	많은 표준 UNIX 프로그램을 포함하는 독립 실행형 시스템 또는 서버의 파일 시스템입니다. 로컬 복사본을 유지하기보다 대형 /usr 파일 시스템을 서버와 공유하면 시스템에 Solaris 소프트웨어를 설치 및 실행하는 데 필요한 전체 디스크 공간이 최소화됩니다.
/var	시스템에서 계속 변경 또는 증가될 가능성이 있는 시스템 파일을 포함하는 파일 시스템 또는 디렉토리(독립 실행형 시스템의 경우)입니다. 이러한 파일로는 시스템 로그, vi 파일, 메일 파일 및 uucp 파일 등이 있습니다.
WAN	Wide Area Network의 약자입니다. 전화, 광섬유 또는 위성 연결을 사용하여 서로 다른 지리적 위치에 있는 여러 개의 LAN(local-area network)이나 시스템을 연결하는 네트워크입니다.
WAN 부트 미니루트	WAN 부트 설치를 수행할 수 있도록 수정된 미니루트입니다. WAN 부트 미니루트에는 Solaris 미니루트에 있는 소프트웨어의 일부가 들어 있습니다. 미니루트 도 참조하십시오.
WAN 부트 서버	WAN 부트 설치 중에 사용한 구성 및 보안 파일을 제공하는 웹 서버입니다.

WAN 부트 설치	HTTP 또는 HTTPS를 사용하여 WAN(wide area network)에서 소프트웨어를 부트 및 설치할 수 있게 해 주는 설치 유형입니다. WAN 부트 설치 방법을 사용하면 공용 네트워크에서 암호화된 Solaris Flash 아카이브를 전송하고 원격 클라이언트에서 사용자 정의 JumpStart 설치를 수행할 수 있습니다.
wanboot-cgi 프로그램	WAN 부트 설치에서 사용한 데이터와 파일을 검색 및 전송하는 CGI 프로그램입니다.
wanboot.conf 파일	WAN 부트 설치를 수행하는 데 필요한 구성 정보와 보안 설정을 지정하는 텍스트 파일입니다.
wanboot 프로그램	WAN 부트 설치를 수행하는 데 필요한 WAN 부트 미니루트, 클라이언트 구성 파일 및 설치 파일을 로드하는 두 번째 수준의 부트 프로그램입니다. WAN 부트 설치의 경우 wanboot 바이너리에서는 ufsboot 또는 inetboot 두 번째 수준 부트 프로그램과 유사한 작업을 수행합니다.
개발자 Solaris 소프트웨어 그룹	최종 사용자 Solaris 소프트웨어 그룹과 라이브러리를 포함하고 파일, 매뉴얼 페이지 및 소프트웨어 개발용 프로그래밍 도구를 포함하는 소프트웨어 그룹입니다.
개인 키	공용 키 암호화에 사용한 암호 해독 키입니다.
검사 키워드	사용자 정의 JumpStart 방법을 사용하여 설치할 때 시스템에 대한 속성 정보를 추출하는 구문 요소입니다. 검사 키워드를 사용하는 경우 일치 조건을 설정하고 규칙에 필요한 프로필을 실행할 필요가 없습니다. 규칙도 참조하십시오.
공용 키	공용 키 암호화에 사용한 암호 키입니다.
공용 키 암호화	모든 사용자에게 알려진 공용 키와 메시지 수신자에게만 알려진 개인 키를 사용하는 암호 체계입니다.
공유 가능 파일 시스템	/export/home 및 /swap과 같은 사용자 정의 파일인 파일 시스템입니다. 이러한 파일 시스템은 Solaris Live Upgrade를 사용할 때 활성 및 비활성 부트 환경 사이에서 공유됩니다. 공유 가능 파일 시스템은 활성 및 비활성 부트 환경 모두에서 vfstab에 동일한 마운트 지점을 포함합니다. 활성 부트 환경에서 공유된 파일을 업데이트하면 비활성 부트 환경에서도 데이터가 업데이트됩니다. 공유 가능 파일 시스템은 기본적으로 공유되지만 대상 슬라이스를 지정할 수 있고 그런 다음 파일 시스템이 공유됩니다.
규칙	프로필에 하나 이상의 시스템 속성을 할당하는 일련의 값입니다. 규칙은 사용자 정의 JumpStart 설치에 사용됩니다.
기능 키	특정 작업에 매핑된, F1, F2, F3 등의 레이블이 붙은 10개 이상의 키보드 키 중 하나입니다.
기본 부트 아카이브	시스템에서 Solaris OS 부트에 사용되는 부트 아카이브. 이 부트 아카이브를 기본 부트 아카이브라고도 합니다. 부트 아카이브를 참조하십시오.
네트워크 설치	네트워크를 통해 CD-ROM 또는 DVD-ROM 드라이브를 가진 시스템에서 CD-ROM이나 DVD-ROM 드라이브가 없는 시스템으로 소프트웨어를 설치하는 방법입니다. 네트워크 설치 시에는 이름 서버 및 설치 서버가 필요합니다.

네트워크에 연결되지 않은 시스템	네트워크에 연결되지 않았거나 다른 시스템에 의존하지 않는 시스템입니다.
네트워크에 연결된 시스템	시스템이 통신하고 정보를 공유할 수 있도록 하드웨어 및 소프트웨어를 통해 연결된 시스템(호스트라고 함) 그룹입니다. LAN(local area network)이라고 합니다. 시스템이 네트워크에 연결되면 일반적으로 하나 이상의 서버가 필요합니다.
논리적 장치	시스템에는 하나의 장치로 보이는 한 개 이상의 디스크에 있는 물리적 슬라이스의 그룹입니다. 논리적 장치는 Solaris 볼륨 관리자에서 볼륨이라고 합니다. 볼륨은 응용 프로그램 또는 파일 시스템의 관점에서 보면 물리적 디스크와 기능적으로 동일합니다.
도메인	인터넷 이름 지정 계층의 일부입니다. 도메인은 관리 파일을 공유하는 로컬 네트워크상의 시스템 그룹을 표현합니다.
도메인 이름	관리 파일을 공유하는 로컬 네트워크상의 시스템 그룹에 할당된 이름입니다. 네트워크 정보 서비스(NIS) 데이터베이스가 제대로 작동하려면 도메인 이름이 필요합니다. 도메인 이름은 마침표로 구분되는 구성 요소 이름의 시퀀스로 구성됩니다(예: tundra.mpk.ca.us). 도메인 이름을 왼쪽에서 오른쪽으로 읽음에 따라 구성 요소 이름은 관리 기관의 보다 일반적인(일반적으로 왼쪽) 영역을 식별합니다.
독립 실행형	다른 모든 시스템의 지원을 필요로 하지 않는 컴퓨터입니다.
디스크(disc)	자기 디스크와 구분되는 광 디스크로서 CD(컴팩트 디스크) 시장에서 사용되는 일반적인 철자를 인식합니다. 예를 들어, CD-ROM 또는 DVD-ROM은 광 디스크입니다.
디스크(disk)	파일과 같이 데이터를 저장하기 위한 집중 트랙과 섹터로 구성되는 자기화된 매체의 둥근 레코드 또는 레코드 세트입니다. 디스크(disc)를 참조하십시오.
디스크 구성 파일	디스크의 구조를 표현하는 파일(예: 바이트/섹터, 플래그, 슬라이스)입니다. 디스크 구성 파일을 통해 단일 시스템의 pfinstall을 사용하여 다른 크기의 디스크에서 프로필을 테스트할 수 있습니다.
디스크 없는 클라이언트	모든 디스크 저장소에 대하여 서버에 의존하는 네트워크상의 클라이언트입니다.
디지털 인증서	통신 당사자가 이미 인증하는 타사에서 발급된 양도 및 위조 불가능한 디지털 파일입니다.
로컬	동일한 언어, 관습 또는 문화적 관례(미국용 영어는 en_US, 영국용 영어는 en_UK)를 공유하는 지리적 또는 정치적 지역 또는 커뮤니티입니다.
루트	항목 계층의 최상위 수준입니다. 루트는 다른 모든 항목이 파생되는 항목입니다. 루트 디렉토리 또는 루트(/) 파일 시스템을 참조하십시오.
루트(/) 파일 시스템	다른 모든 파일 시스템이 파생되는 최상위 수준 파일 시스템입니다. 루트(/) 파일 시스템은 다른 모든 파일 시스템이 마운트되는 기본이며 마운트 해제되지 않습니다. 루트(/) 파일

시스템에는 커널, 장치 드라이버, 시스템 시작(부트)에 사용되는 프로그램과 같이 시스템 작업에 핵심적인 디렉토리 및 파일이 포함됩니다.

루트 디렉토리	다른 모든 디렉토리가 파생되는 최상위 수준 디렉토리입니다.
마스터 시스템	Solaris Flash 아카이브를 설치하는 데 사용되는 시스템입니다. 시스템 구성은 아카이브에 저장됩니다.
마운트	마운트 요청을 하는 시스템 또는 네트워크상의 원격 디스크에 첨부한 디스크에서 디렉토리에 액세스하는 프로세스입니다. 파일 시스템을 마운트하려면 로컬 시스템의 마운트 지점과 마운트될 파일 시스템의 이름(예: /usr)이 필요합니다.
마운트 지점	원격 시스템에 존재하는 파일 시스템을 마운트하는 워크스테이션 디렉토리입니다.
마운트 해제	네트워크상의 컴퓨터 또는 원격 디스크에 연결된 디스크의 디렉토리에 대한 액세스 권한을 제거하는 프로세스입니다.
매니페스트 섹션	복제 시스템의 유효성을 확인하기 위해 사용되는 Solaris Flash 아카이브의 섹션입니다. 매니페스트 섹션은 복제 시스템에서 보관, 추가 또는 삭제된 시스템상의 파일을 나열합니다. 이 섹션은 정보에 불과합니다. 이 섹션은 내부 형식을 사용하여 파일을 나열하며 스크립트에는 사용할 수 없습니다.
매체 서버	설치 서버를 참조하십시오.
메타 장치	볼륨을 참조하십시오.
명령줄	명령으로 시작하는 문자열로서 흔히 옵션, 파일 이름 및 기타 표현식을 포함하는 인자가 뒤에 오며 행 끝 문자로 끝납니다.
문서 루트 디렉토리	웹 서버를 액세스하는 사용자에게 제공할 파일, 이미지 및 데이터가 포함된 웹 서버 시스템의 계층 구조 루트입니다.
미니루트	Solaris 설치 매체에 포함된 부트 가능한 최소 루트(/) 파일 시스템입니다. 미니루트는 시스템을 설치하고 업그레이드하는 데 필요한 Solaris 소프트웨어로 구성됩니다. x86 기반 시스템에서 미니루트는 비상 안전 부트 아카이브로 사용될 시스템으로 복사됩니다. 비상 안전 부트 아카이브를 참조하십시오.
미러	RAID-1 볼륨을 참조하십시오.
복제 시스템	Solaris Flash 아카이브를 사용하여 설치한 시스템입니다. 복제 시스템은 마스터 시스템과 동일한 설치 구성을 갖습니다.
볼륨	시스템에 하나의 논리적 장치로 보이는 물리적 슬라이스 또는 다른 볼륨의 그룹입니다. 볼륨은 응용 프로그램 또는 파일 시스템의 관점에서 보면 물리적 디스크와 기능적으로 동일합니다.

일부 명령줄 유틸리티에서는 볼륨을 메타 장치라고도 합니다. 볼륨은 표준 UNIX 용어로 의사 장치 또는 가상 장치라고도 합니다.

볼륨 관리자	DVD-ROM, CD-ROM 및 디스켓의 데이터에 대한 액세스 권한을 관리하고 얻는 체계를 제공하는 프로그램입니다.
부트	시스템 소프트웨어를 메모리에 로드하고 시스템 소프트웨어를 시작하는 작동입니다.
부트로더	x86 전용: 부트로더는 시스템 전원을 켜 후에 처음으로 실행되는 소프트웨어 프로그램입니다. 이 프로그램은 부트 프로세스를 시작합니다.
부트 서버	동일한 네트워크 서브넷의 클라이언트 시스템에 해당 클라이언트 시스템이 시작해야 하는 프로그램 및 정보를 제공하는 서버 시스템입니다. 부트 서버는 설치 서버가 Solaris 소프트웨어가 설치될 시스템과 다른 서브넷에 있을 경우 네트워크를 통해 설치해야 합니다.
부트 아카이브	<p>x86 전용: 부트 아카이브는 Solaris OS 부트에 사용되는 중요 파일의 모음입니다. 이 파일은 루트(/) 파일 시스템을 마운트하기 전에 시스템을 시작하는 동안 필요합니다. 시스템에는 다음 두 부트 아카이브가 유지 관리됩니다.</p> <ul style="list-style-type: none"> ■ 시스템에서 Solaris OS 부트에 사용되는 부트 아카이브. 이 부트 아카이브를 기본 부트 아카이브라고도 합니다. ■ 기본 부트 아카이브가 손상된 경우 복구에 사용되는 부트 아카이브. 이 부트 아카이브는 루트(/) 파일 시스템을 마운트하지 않고 시스템을 시작합니다. GRUB 메뉴에서는 이 부트 아카이브를 비상 안전이라고 합니다. 아카이브의 주요 목적은 보통 시스템 부트에 사용되는 기본 부트 아카이브를 재생성하는 것입니다.
부트 환경	<p>Solaris OS의 작동에 결정적인 필수 파일 시스템(디스크 슬라이스 및 마운트 지점)의 모음입니다. 이 디스크 슬라이스들은 동일한 디스크에 있을 수도 있고 여러 디스크에 분산되어 있을 수 있습니다.</p> <p>활성 부트 환경은 현재 부트된 환경입니다. 정확히 하나의 활성 부트 환경만 부트될 수 있습니다. 비활성 부트 환경은 현재 부트되지 않았지만 다음 재부트 시 작동 대기 상태일 수 있습니다.</p>
비상 안전 부트 아카이브	x86 전용: 기본 부트 아카이브가 손상된 경우 복구에 사용되는 아카이브. 이 부트 아카이브는 루트(/) 파일 시스템을 마운트하지 않고 시스템을 시동합니다. 이 부트 아카이브는 GRUB 메뉴에서 비상 안전이라고 합니다. 아카이브의 주요 목적은 보통 시스템 부트에 사용되는 기본 부트 아카이브를 재생성하는 것입니다. 부트 아카이브를 참조하십시오.
비전역 영역	Solaris 운영 체제의 단일 인스턴스 내에서 작성된 가상 운영 체제 환경입니다. 하나 이상의 응용 프로그램이 나머지 시스템과 상호 작용하지 않고 비전역 영역에서 실행될 수 있습니다. 비전역 영역은 영역이라고도 합니다. Solaris 영역 및 전역 영역을 참조하십시오.
사용자 정의 JumpStart	Solaris 소프트웨어가 사용자 정의 프로필을 기반으로 시스템에 자동으로 설치되는 설치 유형입니다. 사용자 유형 및 시스템 유형에 따라 사용자 정의된 프로필을 만들 수 있습니다. 사용자 정의 JumpStart 설치하는 사용자가 생성한 JumpStart 설치입니다.

사용자 정의 검사 파일	동일한 JumpStart 디렉토리에 rules 파일로 위치해야 하는 파일은 검사 및 비교라는 두 가지 함수 유형을 포함하는 Bourne 셸 스크립트입니다. 검사 함수는 원하는 정보를 모으거나 실제 작업을 수행하고 사용자가 정의한 해당 ST_ 환경 변수를 설정합니다. 검사 함수는 검사 키워드가 됩니다. 비교 함수는 해당하는 검사 함수를 호출하고, 검사 함수의 출력을 비교하고, 키워드가 일치하면 0을 반환하고 키워드가 일치하지 않으면 1을 반환합니다. 비교 함수는 규칙 키워드가 됩니다. rules 파일도 참조하십시오.
상태 데이터베이스	Solaris 볼륨 관리자 구성의 상태에 대한 디스크 정보를 저장하는 데이터베이스입니다. 상태 데이터베이스는 복제된 여러 데이터베이스 복사본의 모음입니다. 각각의 복사본을 상태 데이터베이스 복제본이라고 합니다. 이 상태 데이터베이스는 모든 알려진 상태 데이터베이스 복제본의 위치 및 상태를 추적합니다.
상태 데이터베이스 복제본	상태 데이터베이스의 복사본입니다. 복제본은 데이터베이스에 있는 데이터가 유효한지 확인합니다.
서버	자원을 관리하고 클라이언트에 서비스를 제공하는 네트워크 장치입니다.
서브넷	경로 지정을 단순화하기 위해 단일 논리적 네트워크를 더 작은 물리적 네트워크로 나누는 작동 체계입니다.
서브넷 마스크	서브넷 주소 지정을 위해 인터넷 주소에서 비트를 선택하는 데 사용되는 비트 마스크입니다. 마스크는 32비트 길이이고, 인터넷 주소의 네트워크 부분과 로컬 부분의 1 또는 그 이상의 비트를 선택합니다.
설치 서버	네트워크상의 다른 시스템이 Solaris를 설치할 수 있는 Solaris DVD 또는 CD 이미지를 제공하는 서버(매체 서버라고도 함)입니다. Solaris DVD 또는 CD 이미지를 서버의 하드 디스크로 복사하여 설치 서버를 만들 수 있습니다.
소프트웨어 그룹	Solaris 소프트웨어의 논리적 그룹(클러스터 및 패키지)입니다. Solaris를 설치하는 동안 코어, 최종 사용자 Solaris 소프트웨어, 개발자 Solaris 소프트웨어 또는 전체 Solaris 소프트웨어, 그리고 SPARC 시스템의 경우 전체 Solaris 소프트웨어 그룹 및 OEM 지원을 설치할 수 있습니다.
수퍼유저	시스템에서 모든 관리 작업을 수행할 권한을 가진 특수 사용자입니다. 수퍼유저에게는 모든 파일의 읽기 및 쓰기, 모든 파일 실행, 모든 프로세스의 중단 신호 전송과 같은 권한이 있습니다.
스왑 공간	메모리 영역의 내용을 메모리로 다시 로드할 수 있을 때까지 해당 내용을 임시로 보유하는 슬라이스 또는 파일입니다. /swap 또는 swap 파일 시스템이라고도 합니다.
슬라이스	디스크 공간이 소프트웨어에 의해 나뉘는 단위입니다.
시스템 구성 파일	(system.conf) WAN 부트 설치에서 사용할 sysidcfg 파일 및 사용자 정의 JumpStart 파일 위치를 지정하는 텍스트 파일입니다.

시작 스크립트	rules 파일 내에 지정된 사용자 정의 Bourne 셸 스크립트로서 시스템에 Solaris 소프트웨어가 설치되기 전에 작업을 수행합니다. 사용자 정의 JumpStart 설치에 한하여 시작 스크립트를 사용할 수 있습니다.
아카이브	<p>마스터 시스템에서 복사한 파일의 모음을 포함하는 파일입니다. 이 파일에는 이름 및 아카이브를 만든 날짜 등과 같은 아카이브에 대한 식별 정보도 포함되어 있습니다. 시스템에 아카이브를 설치하면 해당 시스템이 마스터 시스템의 정확한 구성을 갖게 됩니다.</p> <p>아카이브는 차등 아카이브가 될 수 있습니다. 차등 아카이브는 원래의 마스터 이미지와 업데이트된 마스터 이미지라는 두 시스템 이미지 간의 차이만을 포함하는 Solaris Flash 아카이브입니다. 차등 아카이브에는 복제 시스템에서 보관, 수정 또는 삭제된 파일이 포함되어 있습니다. 차등 업데이트는 지정된 파일만을 변경하며, 원래의 마스터 이미지와 일치하는 소프트웨어가 포함된 시스템에만 제한됩니다.</p>
암호 해독	암호화된 데이터를 일반 텍스트로 변환하는 과정입니다. 암호화 도 참조하십시오.
업그레이드	<p>파일을 기존 파일과 병합하고 수정 사항을 가능한 위치에 저장하는 설치 방법입니다.</p> <p>Solaris OS의 업그레이드는 Solaris OS의 새 버전을 시스템의 디스크의 기존 파일과 병합합니다. 업그레이드하는 경우 이전 버전의 Solaris OS에 적용된 수정 사항을 최대한 많이 저장할 수 있습니다.</p>
업그레이드 옵션	Solaris 설치 프로그램 프로그램이 제시하는 옵션입니다. 업그레이드 절차는 Solaris의 새 버전을 디스크의 기존 파일과 병합합니다. 업그레이드는 또한 Solaris가 마지막 설치된 이후 가능한 많은 로컬 수정을 저장합니다.
업데이트	<p>시스템에서 동일한 유형의 소프트웨어를 변경하는 설치 또는 설치 작업입니다.</p> <p>업그레이드와 달리 업데이트는 시스템을 다운그레이드할 수 있습니다. 초기 설치와 달리 업데이트하려면 설치하고 있는 소프트웨어와 동일한 유형의 소프트웨어가 설치되어 있어야 합니다.</p>
연결	RAID-0 볼륨입니다. 슬라이스가 연결되면 첫 번째 사용 가능한 슬라이스가 가득 찰 때까지 해당 슬라이스에 데이터가 기록됩니다. 슬라이스가 가득 차면 데이터는 연속적으로 다음 슬라이스에 기록됩니다. 연결은 미래에 포함된 경우 외에는 데이터 중복을 제공하지 않습니다. RAID-0 볼륨을 참조하십시오.
영역	비전역 영역을 참조하십시오.
유틸리티	컴퓨터의 관리를 수행하는, 일반적으로 컴퓨터 구입과 함께 추가 비용 없이 제공되는 표준 프로그램입니다.
이름 서버	네트워크의 시스템에 이름 서비스를 제공하는 서버입니다.
이름 서비스	시스템 간 상호 통신할 수 있도록 네트워크상의 모든 시스템에 대한 코어 시스템 정보를 포함하는 분산 네트워크 데이터베이스입니다. 이름 서비스를 사용하여 네트워크상에서

시스템 정보를 유지, 관리 및 액세스할 수 있습니다. 이름 서비스가 없다면 각 시스템은 시스템 정보의 자체 복사본을 로컬 /etc 파일에 보관해야 합니다. Sun은LDAP, NIS 및 NIS+ 이름 서비스를 제공합니다.

인증 기관	디지털 서명과 공용-개인 키 쌍을 만드는 데 사용되는 디지털 인증서를 발급하는 인증된 제3의 조직이나 회사입니다. CA는 고유한 인증서를 부여받은 개인의 자격을 보증합니다.
작업	컴퓨터 시스템에 의해 완료될 사용자 정의 작업입니다.
전역 영역	Solaris 영역에서 전역 영역은 시스템의 기본 영역이자 시스템 전반 관리 제어에 사용되는 영역입니다. 전역 영역은 비전역 영역이 구성, 설치, 관리 또는 설치 해제될 수 있는 유일한 영역입니다. 물리적 장치, 라우팅 또는 동적 재구성(DR)과 같은 시스템 인프라의 관리는 전역 영역에서만 가능합니다. 전역 영역에서 실행 중인 적절한 권한의 프로세스는 모든 다른 영역과 연관된 객체에 액세스할 수 있습니다. Solaris 영역 및 비전역 영역을 참조하십시오.
전체 Solaris 소프트웨어 그룹	전체 Solaris 10 6/06 릴리스가 포함된 소프트웨어 그룹입니다.
전체 Solaris 소프트웨어 그룹 및 OEM 지원	전체 Solaris 10 6/06 릴리스와 OEM에 대한 추가 하드웨어 지원이 포함된 소프트웨어 그룹입니다. 이 소프트웨어 그룹은 SPARC 기반 서버에 Solaris 소프트웨어를 설치할 때 권장됩니다.
종료 스크립트	rules 파일 내에 지정된, Solaris 소프트웨어가 시스템에 설치된 후 그러나 시스템 재부트 전에 작업을 수행하는 사용자 정의 Bourne 셸 스크립트. 사용자 정의 JumpStart 설치와 함께 종료 스크립트를 사용할 수 있습니다.
차등 아카이브	원래의 마스터 이미지와 업데이트된 마스터 이미지라는 두 시스템 이미지 간의 차이만을 포함하는 Solaris Flash 아카이브입니다. 차등 아카이브에는 복제 시스템에서 보관, 수정 또는 삭제된 파일이 포함되어 있습니다. 차등 업데이트는 원래의 지정된 파일만 변경하며 마스터 이미지와 일관된 소프트웨어를 포함하는 시스템에 제한됩니다.
체크섬	그룹 검사에 사용되는 데이터 항목 그룹을 추가한 결과입니다. 데이터 항목은 숫자이거나, 체크섬 계산 동안 숫자로 처리되는 다른 문자열일 수 있습니다. 체크섬 값은 두 장치 사이의 통신이 성공적임을 나타냅니다.
초기 설치	현재 실행되는 소프트웨어를 덮어쓰거나 빈 디스크를 초기화하는 설치입니다. Solaris OS의 초기 설치에는 시스템의 디스크를 Solaris OS의 새 버전으로 겹쳐 씁니다. 시스템에 Solaris OS가 실행되고 있지 않은 경우 초기 설치를 수행해야 합니다. 시스템이 Solaris OS의 업그레이드 가능한 버전을 실행 중이면 초기 설치가 디스크를 겹쳐 쓰고 OS 또는 로컬 수정을 보존하지 않습니다.
최종 사용자 Solaris 소프트웨어 그룹	공통 데스크탑 환경(CDE) 및 DeskSet 소프트웨어를 포함하여 최종 사용자에게 대한 핵심 소프트웨어 그룹과 권장 소프트웨어를 포함하는 소프트웨어 그룹입니다.
축소 네트워크 지원 소프트웨어 그룹	제한된 네트워크 서비스 지원을 가진 Solaris 시스템을 부트 및 실행하는 데 필요한 최소 코드를 제공하는 소프트웨어 그룹입니다. 축소 네트워크 지원 소프트웨어 그룹은 다중

사용자 텍스트 기반 콘솔과 시스템 관리 유틸리티를 제공합니다. 이 소프트웨어 그룹은 또한 시스템이 네트워크 인터페이스를 인식하도록 하지만 네트워크 서비스를 활성화하지는 않습니다.

커버로스	강력한 비밀키 암호화 도구를 사용하여 클라이언트 및 서버가 비보안 네트워크 연결을 통해 서로에 대하여 자신을 식별할 수 있도록 하는 네트워크 인증 프로토콜입니다.
클라이언트	통신용 클라이언트 서버 모델에서 클라이언트는 연산 능력과 대형 메모리 용량과 같은 연산 서버의 자원에 원격으로 액세스하는 프로세스입니다.
클러스터	패키지의 논리적 모음(소프트웨어 모듈)입니다. Solaris 소프트웨어는 클러스터와 패키지로 구성되는 소프트웨어 그룹으로 나뉩니다.
키	데이터를 암호화하거나 해독하기 위한 코드입니다. 암호화도 참조하십시오.
파생된 프로필	사용자 정의 JumpStart 설치 중에 시작 스크립트가 동적으로 만든 프로필입니다.
파일 서버	네트워크상의 시스템에 소프트웨어와 파일 저장소를 제공하는 서버입니다.
패널	창, 대화 상자 또는 애플릿의 내용을 구성하기 위한 컨테이너입니다. 패널은 사용자 입력을 수집 및 확인할 수 있습니다. 패널은 마법사에 의해 사용될 수 있으며 지정된 작업을 완수하기 위해 순서가 정해진 시퀀스를 따릅니다.
패키지	모듈러 설치를 위한 단일 엔티티로 그룹화된 소프트웨어의 모음입니다. Solaris 소프트웨어는 클러스터와 패키지로 구성되는 소프트웨어 그룹으로 나뉩니다.
포맷	데이터를 구조화하거나 디스크를 데이터 수신용 섹터로 나누는 작업입니다.
폴백	이전에 실행된 환경으로의 복귀를 의미합니다. 부트 오류(또는 일부 원하지 않는 동작)에 대해 환경과 지정된 부트 환경을 활성화할 때 폴백을 사용합니다.
표준 시간대	표준 시간이 유지되는 지구 표면의 24 경도 분할 중 하나입니다.
프로필	사용자 정의 JumpStart 방법을 사용할 때 Solaris 소프트웨어를 설치하는 방법을 정의하는 텍스트 파일입니다. 예를 들어, 프로필은 설치할 소프트웨어 그룹을 정의합니다. 모든 규칙은 규칙이 일치할 때 시스템을 설치할 방법을 정의하는 프로필을 지정합니다. 일반적으로 모든 규칙에 대하여 다른 프로필을 만듭니다. 그러나 동일한 프로필이 하나 이상의 규칙에서 사용될 수 있습니다. <i>rules</i> 파일도 참조하십시오.
프로필 디스켓	루트 디렉토리(JumpStart 디렉토리)에 모든 핵심 사용자 정의 JumpStart 파일을 포함하는 디스켓입니다.
프로필 서버	JumpStart 디렉토리에 모든 핵심 사용자 정의 JumpStart 파일을 포함하는 서버입니다.
플랫폼 그룹	특정 소프트웨어를 배포할 목적으로 공급업체가 정의한 하드웨어 플랫폼 그룹입니다. 유효한 플랫폼 그룹의 예로는 i86pc 및 sun4u가 있습니다.

플랫폼 이름	uname -i 명령의 출력입니다. 예를 들어, Ultra 60의 플랫폼 이름은 SUNW,Ultra-60입니다.
하드 링크	디스크상의 파일을 참조하는 디렉토리 항목입니다. 그러한 하나 이상의 디렉토리 항목이 동일한 물리적 파일을 참조할 수 있습니다.
하위 미러	RAID-0 볼륨을 참조하십시오.
해시	일부 입력을 가져와서 입력보다 훨씬 더 짧은 숫자를 생성하여 만든 숫자입니다. 동일한 입력에 대해 항상 동일한 출력 값이 생성됩니다. 테이블 검색 알고리즘, 오류 감지 및 무단 변경 감지에 해시 기능을 사용할 수 있습니다. 무단 변경 감지에 사용할 경우 동일한 해시 결과를 만들어내는 두 개의 입력을 찾기 어려우므로 해시 기능을 선택합니다. MD5 및 SHA-1은 단방향 해시 기능의 예입니다. 예를 들어, 메시지 다이제스트에서는 디스크 파일과 같은 가변 길이 입력을 가져와서 이를 작은 값으로 줄입니다.
해싱	문자열을 원래 문자열을 나타내는 값이나 키로 변경하는 과정입니다.
핵심 소프트웨어 그룹	시스템에서 Solaris OS를 부트 및 실행하는 데 필요한 최소 소프트웨어를 포함하는 소프트웨어 그룹입니다. 핵심 소프트웨어는 공통 데스크탑 환경(CDE) 데스크탑을 실행하는 데 필요한 일부 네트워크 소프트웨어 및 드라이버를 포함합니다. CDE 소프트웨어는 포함하지 않습니다.
핵심 파일 시스템	Solaris OS가 필요로 하는 파일 시스템입니다. Solaris Live Upgrade를 사용할 때 이 파일 시스템은 활성 및 비활성 부트 환경의 vfstab에서 별도의 마운트 지점입니다. 파일 시스템의 예에는 root(/), /usr, /var 및 /opt가 있습니다. 이 파일 시스템은 항상 소스에서 비활성 부트 환경으로 복사됩니다.
호스트 이름	시스템이 네트워크상의 다른 시스템에 알려진 이름입니다. 이 이름은 특정 도메인 내(일반적으로 이것은 모든 단일 구성 내부를 뜻함)에서 모든 시스템 사이에 고유해야 합니다. 호스트 이름은 문자, 숫자 및 마이너스 기호(-)의 모든 조합이 될 수 있지만 마이너스 기호로 시작하거나 끝낼 수 없습니다.
화살표 키	숫자 키패드에 있는 4개의 방향 키 중 하나입니다.

색인

번호와 기호

#

rules 파일, 79
프로필, 82

&&(앰퍼샌드) 규칙 필드, 79

(/) 파일 시스템

JumpStart가 설정한 값, 185

!(느낌표) 규칙 필드, 79

A

add_install_client 명령어, JumpStart 디렉토리
액세스, 73

AND 규칙 필드, 79

any

검사 키워드, 설명 및 값, 190
규칙 키워드, 설명 및 값, 145, 189

arch 검사 키워드, 189

arch 규칙 키워드, 145, 189

archive_location 키워드, 151-156

auto_install_sample 디렉토리

check 스크립트, 98, 119
JumpStart 디렉토리에 파일 복사, 72, 76, 77
set_root_pw 종료 스크립트, 105

B

setup_install_server 명령어의 -b 옵션, 138

backup_media 키워드, 156-157

begin.log 파일, 100

begin 규칙 필드, 설명, 79

boot: cannot open /kernel/unix 메시지, 211

boot_device 키워드, 158

bootenv createbe 키워드, 159

bootparams 파일

JumpStart 디렉토리 액세스 활성화, 74
업데이트, 216

C

-c 옵션

pfinstall 명령, 96
add_install_client 명령어, 141, 142

Can't boot from file/device 메시지, 211

CHANGE DEFAULT BOOT DEVICE 메시지, 217

check 스크립트

custom_probes.ok 파일 만들기, 119
custom_probes 파일 검증, 118, 119
rules.ok 파일 만들기, 97
rules 파일 검증, 97, 98, 119
규칙 테스트, 98, 119
파생된 프로필 및, 101

client_arch 키워드, 160

CLIENT MAC ADDR 오류 메시지, 216

client_root 프로필 키워드, 160

clock gained xxx days 메시지, 211

cluster 프로필 키워드

설명 및 값, 161-162, 162
예, 83

CPU(프로세서)

검사 키워드, 189
규칙 키워드, 145, 189

.cshrc 파일, 104

custom_probes.ok 파일
만들기, 118, 119

custom_probes.ok 파일 (계속)
 설명, 118

custom_probes 파일
 check를 사용하여 검증, 118, 119
 custom_probes 테스트, 119
 요구 사항, 116
 이름 지정, 116

D

dfstab 파일, 71, 139
 disks 검사 키워드, 설명 및 값, 189
 disksize 규칙 키워드, 설명 및 값, 146, 189
 domainname 검사 키워드, 189
 domainname 규칙 키워드, 146, 189
 dontuse 프로필 키워드, 163, 186

E

eng_profile 예, 139
 /etc/bootparams 파일
 JumpStart 디렉토리 액세스 활성화, 74, 216
 /etc/dfs/dfstab 파일, 71, 139
 /etc/mnttab 파일, 75

F

fdisk 명령어, 110
 fdisk 프로필 키워드
 설명 및 값, 163-165
 예, 83
 filesys 키워드, 166-168, 168-169
 filesys 프로필 키워드
 설명 및 값, 165-166
 예, 83
 finish.log 파일, 101
 finish 규칙 필드, 설명, 80
 Flash, 참조 아카이브

G

geo 키워드, 169

getfile: RPC 오류: 오류 5: RPC 시간 초과 메시지, 74
 GRUB 기반 부팅

개요, 49-51
 계획, 52
 네트워크를 통해, 52
 방법, 50
 설명
 menu.lst 파일, 54-56
 주 메뉴, 53
 장치 이름 지정 규약, 50-51
 GRUB 기반 부팅
 menu.lst 파일 찾기, 56
 명령어 참조, 132
 설치, 129, 131
 프로필 디스켓 만들기, 76

H

hostaddress 검사 키워드, 189
 hostaddress 규칙 키워드, 146, 189
 hostname 검사 키워드, 설명 및 값, 189
 hostname 규칙 키워드
 설명 및 값, 146, 189
 예, 145-149

I

install_config 명령어, 74
 install_type 키워드, 170
 install_type 프로필 키워드
 예, 83
 요구 사항, 82, 83
 프로필 테스트, 96-97
 installed 검사 키워드, 설명 및 값, 189
 installed 규칙 키워드, 설명 및 값, 147, 189
 IP 주소
 검사 키워드, 189
 규칙 키워드, 146, 189

J

JumpStart 디렉토리
 rules 파일 예, 78

JumpStart 디렉토리 (계속)

- 공유, 71, 139
- 권한, 71, 75
- 만들기
 - SPARC 기반 시스템용 디스켓, 75
 - x86 기반 시스템용 디스켓, 75, 77
 - 서버, 71
 - 예, 139
- 종료 스크립트를 사용하여 파일 추가, 102
- 파일 복사
 - 설치 파일, 72, 76, 77
 - 종료 스크립트 사용, 102

JumpStart 디렉토리 공유, 71, 139**K**

- karch 검사 키워드, 189
- karch 규칙 키워드, 147, 189

L

- layout_constraint 키워드, 170-172
- le0: No carrier - transceiver cable problem 메시지, 211
- locale 키워드, 173

M

- marketing_profile 예, 140
- memsize 검사 키워드, 설명 및 값, 189
- memsize 규칙 키워드, 설명 및 값, 147, 189
- menu.lst 파일
 - 설명, 54-56
 - 위치, 56
- metadb 프로파일 키워드, 173-174
- metadevice, 참조 볼륨
- mnttab 파일, 75
- model 검사 키워드, 설명 및 값, 190
- model 규칙 키워드, 설명 및 값, 148, 190

N

- network 검사 키워드, 설명 및 값, 190

- network 규칙 키워드, 설명 및 값, 148, 190
- No carrier - transceiver cable problem 메시지, 211
- no_master_check 키워드, 174
- noneuclidean 프로파일 키워드, 174
- Not a UFS filesystem 메시지, 211

O

- osname 검사 키워드, 190
- osname 규칙 키워드, 148, 190

P

- check 스크립트의 -p 옵션, 98, 119
- partitioning 키워드, 179
- pfinstall 명령, 93
- probe 규칙 키워드, 설명 및 값, 149
- profile 필드의 =(등호), 100
- prtvtoc 명령어
 - SPARC: 디스크 구성 파일 만들기, 108
 - x86: 디스크 구성 파일 만들기, 110

R

- check 스크립트의 -r 옵션, 98, 119
- RAID-0 볼륨
 - 개념적 개요, 197
 - 정의, 197
 - 지침, 203-207
- RAID-1 볼륨
 - 개념적 개요, 198
 - 만들기 정보, 203
 - 및 디스크 구조, 202
 - 정의, 198
 - 지침, 203-207
- root_device 키워드, 184
- rootdisk
 - fileys에 대한 슬라이스 값, 166
 - JumpStart가 설정한 값, 185
 - 정의, 185
- RPC Timed out message, 216
- RPC 시간 초과 메시지, 74
- RPC 오류: 오류 5: RPC 시간 초과 메시지, 74

- rule_keyword 규칙 필드, 79
 - rule_value 규칙 필드, 79
 - rules
 - 구문, 79
 - 필드 설명, 79
 - rules.ok 파일
 - 규칙에 따른 순서 일치, 122, 128
 - rules.ok 파일, 규칙의 일치 순서, 80
 - rules.ok 파일
 - 만들기, 97
 - 설명, 97
 - rules 파일
 - check를 사용하여 검증, 98
 - 사용자 정의 JumpStart 예, 141
 - 파생된 프로필 및, 101
 - 구문, 79
 - 규칙 추가, 79
 - 규칙 테스트, 98
 - 다중 행 규칙, 79
 - 만들기, 78
 - 사용자 정의 JumpStart 예, 140, 141
 - 설명, 78
 - 예, 78
 - 이름 지정, 78, 79
 - 주석, 79
 - rules 파일의 다중 행, 79
 - rules 파일의 백슬래시, 79
 - rules 파일의 행 바꾸기, 79
- S**
- add_install_client 명령어의 -s 옵션, 142
 - set_root_pw 종료 스크립트, 105
 - share 명령어
 - JumpStart 디렉토리 공유, 71, 139
 - shareall 명령어, 71, 139
 - SI_PROFILE 환경 변수, 100
 - Solaris Flash, 참조 아카이브
 - Solaris Live Upgrades, 설치 프로그램 선택, 29
 - Solaris Zones 파티셔닝 기술, 업그레이드, 186
 - Solaris 대화식 설치 프로그램, 설치 프로그램 선택, 29
 - Solaris 볼륨 관리자, 사용자 정의 JumpStart 중에 볼륨
 - 만들기, 예, 83
 - Solaris 설치의 새로운 기능, 17
 - Solaris 소프트웨어
 - 그룹, 161-162
 - 업그레이드, 162
 - 프로필 예, 83
 - 릴리스 또는 버전
 - installed 검사 키워드, 189
 - installed 규칙 키워드, 147, 189
 - osname 검사 키워드, 190
 - osname 규칙 키워드, 148, 190
 - Solaris 소프트웨어의 릴리스
 - installed 검사 키워드, 189
 - installed 규칙 키워드, 147, 189
 - osname 검사 키워드, 190
 - osname 규칙 키워드, 148, 190
 - Solaris 소프트웨어의 버전
 - installed 검사 키워드, 189
 - installed 규칙 키워드, 147, 189
 - osname 검사 키워드, 190
 - osname 규칙 키워드, 148, 190
 - Solaris 영역 분할 기술
 - Solaris Flash 아카이브와 함께 설치, 45
 - 개요, 43
 - 디스크 공간 요구 사항, 45
 - 업그레이드, 44
 - stty 명령어, 125, 129
 - SUNWCall 그룹, 161-162
 - SUNWCprog 그룹, 161-162
 - SUNWCreq 그룹, 161-162
 - SUNWCrnet 그룹, 161-162
 - SUNWCuser 그룹, 161-162
 - SUNWCXall 그룹, 161-162
 - system_type 프로필 키워드
 - 설명 및 값, 186
 - 예, 83
- T**
- totaldisk 검사 키워드, 190
 - totaldisk 규칙 키워드, 149, 190
 - transceiver cable problem 메시지, 211
- U**
- UFS, 75
 - usedisk 프로필 키워드, 설명 및 값, 186

V

/var/sadm/system/logs/begin.log 파일, 100
 /var/sadm/system/logs/finish.log 파일, 101
 volcheck 명령어, 75,77

W

WARNING: CHANGE DEFAULT BOOT DEVICE, 217
 WARNING: clock gained xxx days 메시지, 211

개

개발자 Solaris 소프트웨어 그룹, 161-162
 설명, 36-38
 크기, 37
 프로파일 예, 83
 개요, GRUB 기반 부트, 49-51

검

검사 키워드
 arch, 189
 domainname, 189
 hostaddress, 189
 hostname, 189
 installed, 189
 karch, 189
 memsize, 189
 model, 190
 network, 190
 osname, 190
 rootdisk, 190
 totaldisk, 190
 디스크, 189

검증

custom_probes 파일
 check 사용, 119
 테스트, 119
 rules 파일
 check 사용, 97,98,119
 규칙 테스트, 98
 사용자 정의 JumpStart 예, 141
 파생된 프로파일 및, 101

경

경로, check 스크립트, 98,119

계**계획**

GRUB 기반 부트, 52
 네트워크를 통해 설치, 27-28
 디스크 공간, 34-38
 설치 프로그램 선택, 29
 작업 개요, 25
 초기 설치와 업그레이드 비교, 28

과

과반수 일치 알고리즘, 196

구

구성, 디스크 구성 파일 만들기, 108

권**권한**

JumpStart 디렉토리, 71,75
 시작 스크립트, 100
 종료 스크립트, 101

규**규칙**

rootdisk 일치 규칙, 185
 다중 행 규칙, 79
 순서 일치, 122,128
 예, 80
 유효성 테스트, 98,119
 일치 순서, 80
 파생된 프로파일, 100,101
 필드 설명, 80
 규칙 키워드, 145
 any, 설명 및 값, 145,189

규칙 키워드 (계속)

- arch, 145, 189
 - disksize, 설명 및 값, 146, 189
 - domainname, 146, 189
 - hostaddress, 146, 189
 - hostname, 145-149, 189
 - installed, 설명 및 값, 147, 189
 - karch, 147, 189
 - memsize, 147, 189
 - model, 148, 190
 - network, 148, 190
 - osname, 148, 190
 - probe, 149
 - totaldisk, 149, 190
- 규칙 필드의 Bourne 셸 스크립트, 79

기

- 기본값
 - 분할
 - 디스크 제외, 163
- 설치된 소프트웨어 그룹, 162
- 영역 분할
 - 디스크 지정, 186
- 파생된 프로파일 이름, 100

네

- 네트워크 번호, 148, 190
- 네트워크 부트, GRUB 사용, 52
- 네트워크 설치, 사용자 정의 JumpStart 설치, 예, 65

논

- 논리 AND 규칙 필드, 79

느

- 느낌표(!) 규칙 필드, 79

대

- 대체 설치 프로그램, 113

도

- 도메인
 - 검사 키워드, 189
 - 규칙 키워드, 146, 189

독

- 독립형 시스템
 - 사용자 정의 JumpStart 설치 예, 64
 - 프로필 예, 83

디

- 디렉토리
 - JumpStart
 - rules 파일 예, 78
 - 권한, 71, 75
 - 디렉토리 공유, 71, 139
 - 디렉토리 만들기, 139
 - 설치 파일 복사, 72, 76, 77
 - 시스템용 만들기, 75
 - 파일 복사, 102
 - 파일 추가, 102
 - 변경
 - JumpStart 디렉토리로, 98, 119
 - 로컬 디스크의 Solaris SPARC 소프트웨어 이미지로, 76
 - 로컬 디스크의 Solaris x86 기반 소프트웨어 이미지로, 77
 - 로컬 디스크의 Solaris 소프트웨어 이미지로, 72
 - 디렉토리 변경
 - JumpStart 디렉토리로, 98, 119
 - 로컬 디스크의 Solaris SPARC 소프트웨어 이미지, 76
 - 로컬 디스크의 Solaris x86 기반 소프트웨어 이미지로, 77
 - 로컬 디스크의 Solaris 소프트웨어 이미지로, 72
- 디스켓
 - JumpStart 디렉토리 액세스, 73

디스켓 (계속)

x86: *JumpStart* 디렉토리, 75

디스크 공간

계획, 34-38

비전역 영역용으로 계획, 45

요구 사항, 소프트웨어 그룹, 37

디스크 구성 파일

만들기

SPARC 기반 시스템, 108

x86 기반 시스템, 110

설명, 93, 108

디스크 없는 클라이언트

스왑 공간, 160

플랫폼, 160

로

로그 파일

시작 스크립트 출력, 100

종료 스크립트 출력, 101

루

루트(/) 파일 시스템, 비활성 부트 환경용 패키지 요구 사항, 229

루트(/) 파일 시스템, 프로필 예, 66

루트 비밀번호, 종료 스크립트를 사용하여 설정, 105

루트 환경, 종료 스크립트를 사용하여 사용자 정의, 104

마

마운트

Solaris 설치에 의해, 101

시작 스크립트 주의 사항, 100

원격 파일 시스템, 165-166

마이크로프로세서

검사 키워드, 189

규칙 키워드, 145, 189

만

만들기

custom_probes.ok 파일, 118, 119

JumpStart 디렉토리, 서버에, 71

RAID-1 볼륨, 168-169

rules.ok 파일, 97, 118

rules 파일, 78

UFS, 75

디스크 구성 파일, 108

로컬 파일 시스템, 166-168

프로필

설명, 81

파생, 100

메

메모리

검사 키워드, 189

규칙 키워드, 147, 189

스왑 공간 크기 및, 161

문

문제 해결

DHCP를 사용하여 네트워크에서 부트, 216

일반 설치 문제

DHCP를 사용하여 네트워크에서 부트, 216

시스템 부트, 216

잘못된 서버에서 부트, 216

미

미러, 참조 RAID-1 볼륨

번

변수

SI_PROFILE, 100

SYS_MEMSIZE, 95

보

보안, 루트 비밀번호, 105

복

복사

JumpStart 디렉토리 파일, 102

JumpStart 설치 파일, 72, 76, 77

복제본, 196

볼

볼륨

가상 디스크, 194

개념적 개요, 194

명명 규칙, 204

사용, 194

볼륨 관리자

복사, 75, 77

부

부트

GRUB 사용, 49-51

GRUB를 사용하여 네트워크에서, 52

RAID-1 볼륨 및 단일 사용자 모드, 207

부트 디스크 분할 영역 레이아웃, 새 기본값(x86 기반 시스템), 46

부트로더, GRUB, 49-51

부팅

GRUB 사용, 명령어 참조, 132

GRUB로 설치, 129, 131

프로필 디스켓 만들기, 76

분

분할

fdisk 분할 영역, 163-165

디스크 제외, 163

프로필 키워드, 179

비

비밀번호, 루트, 105

비전역 영역

Solaris Flash 아카이브와 함께 설치, 45

개요, 43

디스크 공간 요구 사항, 45

설명, 43

업그레이드, 44, 186

사

사용자 정의 JumpStart 설치, 121

개요, 66

부트 및 설치, 121

선택적 기능, 99

finis 스크립트, 101

개요, 99

사이트 특정 설치 프로그램, 113

시작 스크립트, 99, 101

종료 스크립트, 105

설명, 66

설치 프로그램 선택, 29

예, 135, 143

check 스크립트, 141

eng_profile 만들기, 139

JumpStart 디렉토리, 139

marketing_profile 만들기, 140

RAID-1 볼륨 프로필, 90, 92

rules 파일 편집, 140, 141

Solaris Flash 프로필, 86, 87, 88, 89

WAN 부트 설치 프로필, 87

네트워크에 연결되지 않은, 64

네트워크에 연결된, 65

독립형 시스템, 64

마케팅 시스템 설정, 138, 142

부트 및 설치, 143

사이트 설정, 136

엔지니어링 시스템 설정, 141

예제

사이트 설정, 135

준비, 66, 98

팁 행 연결 요구 사항, 125, 129

프로필 키워드, 149

사이트 특정 설치 프로그램, 113

삭

삭제, 업그레이드할 때 클러스터, 162

상

상태 데이터베이스

개념, 196-197

정의, 196-197

상태 데이터베이스 복제본, 196

기본 작업, 196

단일 슬라이스에서 여러 개 만들기, 201

사용, 196

위치, 202, 203

정의, 196

최소 수, 202

서

서버

JumpStart 디렉토리 만들기, 71

루트 공간, 160

서비스 분할 영역, 설치 중 보존(x86 기반 시스템), 46

설

설치

Solaris Flash 아카이브 사용, 40

네트워크를 통해

계획, 27-28

디스크 공간 권장 사항, 34-38

업그레이드와 비교, 28

작업 개요, 25

설치 준비, 사용자 정의 JumpStart 사용, 66, 98

소

소프트웨어 그룹

설명, 37

업그레이드, 42, 162

크기, 37

프로필 예, 83

프로필에 대한, 161-162

스

스왑 파일 시스템

디스크 없는 클라이언트 스왑 공간, 160

메모리 크기 및, 161

크기 결정, 161

프로필 예, 66

스크립트

규칙 필드의 Bourne 셸 스크립트, 79

시작 스크립트, 99, 101, 113

종료 스크립트, 101, 105, 113

슬

슬라이스

검사 키워드, 189

규칙 키워드, 147, 189

프로필 예, 83

시

시간 초과 RPC 오류, 216

시작, check 스크립트, 98

시작 스크립트

개요, 99

권한, 100

규칙 필드, 79

사이트 특정 설치 프로그램, 113

파생된 프로필 만들기, 100, 101

아

아카이브

JumpStart 프로필 예, 86, 87, 88, 89

Solaris Flash 아카이브와 함께 설치, 45

설명, 30

설치, 40

설치 프로그램 선택, 29

키워드, 사용자 정의 JumpStart, 151-156

알

알 수 없는 클라이언트 오류 메시지, 211

앰

앰퍼샌드(&&) 규칙 필드, 79

업

업그레이드

Solaris Flash 아카이브 사용

설명, 40

디스크 공간 권장 사항, 34-38

비전역 영역, 186

비전역 영역으로, 44

사용자 정의 JumpStart 설치, 121

업그레이드 실패, 221

작업 개요, 25

초기 설치와 비교, 28

프로필 키워드, 162, 170, 179

업그레이드 실패, 재부트 문제, 221

연

연결

개념적 개요, 197

만들기 정보, 203

정의, 197

영

영역 분할

fdisk 분할 영역, 83

예, 83

프로필 키워드, 186

요

요구 사항

custom_probes 파일, 116

디스크 공간, 34-38

메모리, 33

프로필, 78, 82

원

원격 파일 시스템, 마운트, 165-166

이

이름/이름 지정

custom_probes 파일, 116

rules 파일, 78, 79

시스템 모델 이름, 148, 190

파생된 프로필 이름, 100

호스트 이름, 146, 189

일

일치

rootdisk 값, 185

규칙 순서, 80

규칙의 순서, 122, 128

파생된 프로필, 100

장

장치 이름 지정 규약, GRUB, 50-51

장치의 이름 지정 규약, GRUB, 50-51

전

전역 영역, 설명, 43

전체 Solaris 소프트웨어 그룹, 161-162

설명, 36-38

크기, 37

전체 Solaris 소프트웨어 그룹 및 OEM 지원, 161-162

설명, 36-38

크기, 37

종

종료 스크립트

규칙 필드, 80

루트 환경 사용자 정의, 104

시스템의 루트 비밀번호 설정, 105

종료 스크립트 (계속)

패키지 및 패치 추가, 102

주

주석

rules 파일, 79

프로필, 82

최

최종 사용자 Solaris 소프트웨어 그룹, 161-162

설명, 36-38

크기, 37

추

추가

rules 파일에 규칙, 79

소프트웨어 그룹에서 패키지, 175

업그레이드할 때 클러스터, 162

종료 스크립트를 사용하여 패키지 또는 패치, 102

축

축소 네트워크 지원 소프트웨어 그룹, 161-162

설명, 36-38

크기, 37

출

출력 파일

시작 스크립트 로그, 100

종료 스크립트 로그, 101

코

코어 Solaris 소프트웨어 그룹, 161-162

코어 시스템 지원 소프트웨어 그룹

설명, 36-38

주석 (계속)

크기, 37

크

크기

메모리, 147, 189

스왑 공간

디스크 없는 클라이언트, 160

최대 크기, 161

프로필 예, 66

팁 행 연결 표시 크기, 125, 129

하드 디스크

검사 키워드, 189, 190

규칙 키워드, 146, 149, 189, 190

루트 공간, 160

키

키워드

Solaris Flash 아카이브, 사용자 정의
JumpStart, 151-156

검사, 115

테

테스트

custom_probes 파일 검증

check 사용, 118

custom_probes 테스트, 119

rules 파일 검증

check 사용, 97, 98, 119

규칙 테스트, 98

사용자 정의 JumpStart 예, 141

파생된 프로필 및, 101

프로필, 93, 96-97

토

토큰 링 카드, 부팅 오류, 215

팁

- 팁 행 연결 요구 사항, 125
- 팁 행 연결 표시 요구 사항, 129

파

- 파생된 프로필, 100, 101
- 파일 및 파일 시스템
 - UFS 만들기, 75
 - 만들기
 - RAID-1 볼륨, 168-169
 - 로컬 파일 시스템, 166-168
 - 복사
 - JumpStart 설치 파일, 72, 76, 77
 - 종료 스크립트를 사용하여 JumpStart 디렉토리 파일, 102
 - 시작 스크립트 출력, 100
 - 원격 파일 시스템 마운트, 165-166
 - 종료 스크립트 출력, 101

패

- 패치
 - 추가
 - chroot를 사용하여, 104
 - 종료 스크립트 사용, 102

패키지

- Solaris Live Upgrade
 - 요구 사항, 229
- 관리 파일, 99
- 사용자 정의 JumpStart를 사용할 경우 요구 사항, 229
- 추가
 - chroot 사용, 104
 - 종료 스크립트 사용, 102

표

- 표시
 - 팁 행 연결 요구 사항, 125, 129

프

- 프로세서
 - 검사 키워드, 189
 - 규칙 키워드, 145, 189
- 프로필
 - 규칙 필드, 79
 - 만들기, 81
 - 설명, 81
 - 시스템 일치, 80, 122, 128
 - 예, 83
 - eng_profile, 139
 - marketing_profile, 140
 - Solaris Flash, 86, 87, 88, 89
 - WAN 부트 설치, 87
 - 요구 사항, 78, 82
 - 이름 지정, 82
 - 주석, 82
 - 테스트, 96-97
 - 파생된 프로필, 100, 101
- 프로필 키워드, 149, 186
 - archive_location, 151-156
 - backup_media, 156-157
 - boot_device, 158
 - bootenv createbe, 159
 - client_arch, 160
 - client_root, 160
 - client_swap, 160
- cluster
 - 설명 및 값, 161-162, 162
 - 예, 83
- dontuse
 - usedisk 및, 186
 - 설명 및 값, 163
- fdisk
 - 설명 및 값, 163-165
 - 예, 83
- filesystems
 - RAID-1 볼륨, 168-169
 - 로컬 파일 시스템, 166-168
 - 설명 및 값, 165-166
 - 예, 83
 - 원격 파일 시스템, 165-166
- forced_deployment, 설명 및 값, 169
- geo
 - 설명 및 값, 169

프로필 키워드 (계속)

install_type
 설명 및 값, 170
 예, 83
 요구 사항, 82, 83
layout_constraint, 설명 및 값, 170-172
local_customization, 설명 및 값, 173
locale, 설명 및 값, 173
metadb
 설명 및 값, 173-174
 예, 83
no_master_check, 설명 및 값, 174
noneuclidean, 174
root_device, 184
system_type
 설명 및 값, 186
 예, 83
usedisk, 설명 및 값, 186
 대소문자 구분, 149
 분할
 디스크 제외, 163
 설명 및 값, 179
 상태 데이터베이스 복제본(meatball)
 만들기, 173-174
 영역 분할
 디스크 지정, 186
 예, 83

프로필 키워드, *install_type* (계속)

프로필 키워드, 179
 스왑 공간
 디스크 없는 클라이언트, 160
 최대 크기, 161
 프로필 예, 66, 83
 영역 분할
 영역 분할 기본값에 지정, 186
 예, 83
 크기
 검사 키워드, 189, 190
 규칙 키워드, 146, 149, 189, 190
 루트 공간, 160
 하위 미러, 참조 RAID-0 volume

플

플랫폼

검사 키워드, 189
 규칙 키워드, 147, 189
 디스크 없는 클라이언트, 160
 시스템 모델 이름, 148, 190
 시스템 속성 및 프로필 일치, 80, 122, 128

하

하드 디스크

rootdisk 값, 185
 마운트, 165-166
 분할
 영역 분할 기본값에서 제외, 163

