



Logical Domains(LDoms) 1.0.2 관리 설명서

Sun Microsystems, Inc.
www.sun.com

부품 번호: 820-4453-10
2008년 3월, 개정판 01

본 설명서에 대한 의견은 다음 사이트로 보내 주십시오. <http://www.sun.com/hwdocs/feedback>

Copyright © 2008 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 모든 권리는 저작권자의 소유입니다.

Sun Microsystems, Inc.는 본 설명서에서 사용하는 기술과 관련된 지적 재산권을 보유하고 있습니다. 특히 이러한 지적 재산권에는 <http://www.sun.com/patents> 나열된 하나 이상의 미국 특허 및 추가 특허 또는 미국 및 기타 국가에서 특허 출원중인 응용 프로그램이 포함될 수 있습니다.

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

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, Java, Solaris, JumpStart, OpenBoot, Sun Fire, Netra, SunSolve, Sun BluePrints, Sun Blade, Sun Ultra 및 SunVTS는 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.

Adobe PostScript 로고는 Adobe Systems, Incorporated의 상표입니다.

본 서비스 매뉴얼에서 언급한 제품 및 해당 정보는 미국 수출 규제법에 의해 다뤄지고 규제되며 다른 국가에서 수출입 법률의 적용을 받을 수 있습니다. 직접적 또는 간접적인 핵, 미사일, 생화학 무기 또는 해상 핵에 사용을 엄격히 금지합니다. 미국 수출입 금지 대상 국가 또는 추방 인사와 특별히 지명된 교포를 포함하여(그러나 이에 국한되지 않음) 미국 수출 제외 대상으로 지목된 사람에 대한 수출이나 재수출은 엄격히 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.



Adobe PostScript

목차

머리말 xvii

1. **Logical Domains** 소프트웨어 개요 1
 - 하이퍼바이저(hypervisor) 및 Logical Domains 1
 - Logical Domains Manager 3
 - Logical Domains 역할 3
 - 명령줄 인터페이스 4
 - 가상 입/출력 4
 - 가상 네트워크 5
 - 가상 저장소 5
 - 가상 콘솔 5
 - 동적 재구성 5
 - 지연된 재구성 6
 - 영구 구성 6
2. 보안 7
 - 보안 고려 사항 7
 - Solaris Security Toolkit 및 Logical Domains Manager 8
 - 강화 9
 - Logical Domains 최소화 10

인증	10
감사	11
준수	12

3. 소프트웨어 설치 및 활성화 13

Solaris OS 업그레이드 13

Logical Domains 제약 조건 데이터베이스 파일 저장 및 복구 13

컨트롤 도메인에서 라이브 업그레이드 사용 14

LDoms 1.0.2 소프트웨어로 업그레이드 14

▼ LDoms 1.0에서 LDoms 1.0.2 소프트웨어로 업그레이드하려면 14

컨트롤 도메인에서 소프트웨어 새로 설치 16

▼ Solaris 10 OS를 설치하려면 16

▼ 시스템 펌웨어를 업그레이드하려면 17

▼ FTP 서버를 사용하지 않고 시스템 펌웨어를 업그레이드하려면 18

▼ 시스템 펌웨어를 다운그레이드하려면 19

Logical Domains Manager 및 Solaris Security Toolkit 다운로드 19

▼ Logical Domains Manager, Solaris Security Toolkit 및 Logical Domains MIB를 다운로드하려면 19

Logical Domains Manager 및 Solaris Security Toolkit 설치 20

설치 스크립트를 사용하여 Logical Domains Manager 1.0.2 및 Solaris Security Toolkit 4.2 소프트웨어 설치 21

▼ 옵션 없이 install-ldm 스크립트를 사용하여 설치하려면 22

▼ install-ldm 스크립트를 -d 옵션과 함께 사용하여 설치하려면 25

▼ install-ldm 스크립트를 -d none 옵션과 함께 사용하여 설치하려면 26

▼ install-ldm 스크립트를 -p 옵션과 함께 사용하여 설치하려면 27

JumpStart를 사용하여 Logical Domains Manager 1.0.2 및 Solaris Security Toolkit 4.2 소프트웨어 설치 27

▼ JumpStart 서버를 설정하려면 27

▼ JumpStart 소프트웨어를 사용하여 설치하려면 28

Logical Domains Manager 및 Solaris Security Toolkit 소프트웨어 수동 설치 30

- ▼ Logical Domains Manager(LDoms) 1.0.2 소프트웨어를 수동으로 설치하려면 30
- ▼ (선택 사항) Solaris Security Toolkit 4.2 소프트웨어를 수동으로 설치하려면 30
- ▼ (선택 사항) 컨트롤 도메인을 수동으로 강화하려면 31
- ▼ 강화를 검증하려면 32
- ▼ 강화를 실행 취소하려면 32

Logical Domains Manager 데몬 활성화 32

- ▼ Logical Domains Manager 데몬을 활성화하려면 33

인증 및 프로필 만들기와 사용자 계정에 역할 할당 33

사용자 인증 관리 34

- ▼ 사용자 인증을 추가하려면 34
- ▼ 사용자의 모든 인증을 삭제하려면 34

사용자 프로필 관리 35

- ▼ 사용자의 프로필을 추가하려면 35
- ▼ 사용자의 모든 프로필을 삭제하려면 35

사용자에 역할 할당 35

- ▼ 역할을 만들어 사용자에게 해당 역할을 할당하려면 35

4. 서비스 및 논리적 도메인 설정 37

출력 메시지 37

Sun UltraSPARC T1 프로세서 37

Sun UltraSPARC T2 프로세서 38

기본 서비스 만들기 38

- ▼ 기본 서비스를 만들려면 38

컨트롤 도메인의 초기 구성 40

- ▼ 컨트롤 도메인을 설정하려면 40

논리적 도메인 사용을 위한 재부트 41

- ▼ 논리적 도메인 사용을 위해 재부트하려면 41

컨트롤/서비스 도메인과 다른 도메인 간의 네트워킹 활성화 42

▼ 가상 스위치를 기본 인터페이스로 구성하려면 42

가상 네트워크 터미널 서버 데몬 활성화 43

▼ 가상 네트워크 터미널 서버 데몬을 활성화하려면 43

게스트 도메인 만들기 및 시작 44

▼ 게스트 도메인을 만들고 시작하려면 44

게스트 도메인 점프 스타팅(Jump-Starting) 48

5. 기타 정보 및 작업 49

CLI 이름 입력 제한 사항 49

파일 이름(*file*) 및 변수 이름(*var_name*) 49

가상 디스크 서버 *file|device* 및 가상 스위치 장치 이름 49

구성 이름(*config_name*) 49

기타 모든 이름 50

`ldm list` 하위 명령 사용 50

시스템 읽기 가능 출력 50

▼ `ldm` 하위 명령 구문 사용법을 표시하려면 50

플래그 정의 53

사용률 통계 정의 54

다양한 목록의 예 54

▼ 소프트웨어 버전(`-v`)을 표시하려면 54

▼ 짧은 목록을 생성하려면 54

▼ 긴 목록(`-l`)을 생성하려면 55

▼ 확장 목록(`-e`)을 생성하려면 56

▼ 구문 분석할 수 있는 시스템 읽기 가능 목록(`-p`)을 생성하려면 58

▼ 도메인의 상태를 표시하려면 59

▼ 변수를 나열하려면 59

▼ 바인딩을 나열하려면 59

▼ 구성을 나열하려면 60

▼ 장치를 나열하려면	61
▼ 서비스를 나열하려면	62
계약 조건 목록	62
▼ 한 개의 도메인에 대한 계약 조건을 나열하려면	63
▼ XML 형식으로 계약 조건을 나열하려면	64
▼ 시스템 읽기 가능 형식으로 계약 조건을 나열하려면	65
도메인이 과도하게 로드된 경우 ldm stop-domain 명령이 시간 초과될 수 있음	65
가상 네트워크 장치에 해당하는 Solaris 네트워크 인터페이스 이름 확인	66
▼ Solaris OS 네트워크 인터페이스 이름을 찾으려면	66
자동 또는 수동으로 MAC 주소 할당	67
논리적 도메인 소프트웨어에 할당된 MAC 주소 범위	68
자동 할당 알고리즘	68
중복 MAC 주소 감지	69
해제된 MAC 주소	70
수동 MAC 주소 할당	70
▼ 수동으로 MAC 주소를 할당하려면	70
CPU 및 메모리 주소 매핑	71
CPU 매핑	71
▼ CPU 번호를 확인하려면	71
메모리 매핑	71
▼ 실제 메모리 주소를 확인하려면	72
CPU 및 메모리 매핑의 예	72
분할 PCI Express 버스를 구성하여 다중 Logical Domains 사용	74
▼ 분할 PCI 구성을 만들려면	74
PCI 버스에서 I/O MMU 바이패스 모드 활성화	77
콘솔 그룹 사용	77
▼ 여러 콘솔을 하나의 그룹으로 결합하려면	78

- 한 서버에서 다른 서버로 논리적 도메인 이동 78
 - ▼ 이동할 도메인을 설정하려면 79
 - ▼ 도메인을 이동하려면 79
- 논리적 도메인 제거 79
 - ▼ 모든 게스트 논리적 도메인을 제거하려면 80
- Logical Domains로 Solaris OS 작동 81
 - 도메인 지정이 활성화된 경우 Solaris OS를 시작한 후에 OpenBoot 펌웨어를 사용할 수 없음 81
 - 서버 전원 순환 81
 - ▼ 현재 논리적 도메인 구성을 SC에 저장하려면 81
 - OpenBoot power-off 명령의 결과 82
 - Solaris OS 중단 결과 82
 - 컨트롤 도메인 정지 또는 재부트 결과 83
 - 일부 format(1M) 명령 옵션이 가상 디스크에서 작동하지 않음 84
- ALOM CMT와 함께 Ldoms를 사용 84
 - ▼ 논리적 도메인 구성을 기본 또는 다른 구성으로 재설정하려면 85
- BSM 감사 활성화 및 사용 85
 - ▼ enable-bsm.fin 완료 스크립트를 사용하려면 86
 - ▼ Solaris OS bsmconv(1M) 명령을 사용하려면 87
 - ▼ BSM 감사가 활성화되었는지 확인하려면 87
 - ▼ 감사를 비활성화하려면 87
 - ▼ 감사 출력을 인쇄하려면 88
 - ▼ 감사 로그를 회전하려면 88
- NAT 및 라우팅을 위해 가상 스위치 및 서비스 도메인 구성 88
 - ▼ 도메인에 외부 연결을 제공하도록 가상 스위치를 설정하려면 89
- 가상 디스크와 함께 ZFS 사용 89
 - ZFS 볼륨 상단에 가상 디스크 만들기 90
 - ▼ ZFS 볼륨 상단에 가상 디스크를 만들려면 90

가상 디스크를 통해서 ZFS 사용	91
▼ 가상 디스크를 통해서 ZFS 사용하려면	91
부트 디스크로 ZFS 사용	93
▼ 부트 디스크로 ZFS를 사용하려면	93
Logical Domains 환경에서 Volume Manager 사용	94
Volume Manager 상단에서 가상 디스크 사용	95
SVM의 상단에서 가상 디스크 사용	96
VxVM이 설치된 경우 가상 디스크 사용	97
가상 디스크 상단에서 Volume Manager 사용	97
가상 디스크 상단에서 ZFS 사용	98
가상 디스크 상단에서 SVM 사용	98
가상 디스크 상단에서 VxVM 사용	98
Logical Domains 환경에서 IPMP 구성	98
논리적 도메인에서 IPMP 그룹에 가상 네트워크 장치 구성	99
서비스 도메인에서 IPMP 구성 및 사용	100
용어집	101

그림

- 그림 1-1 2개의 논리적 도메인을 지원하는 하이퍼바이저 2
- 그림 5-1 별도의 가상 스위치 인스턴스에 연결된 두 개의 가상 네트워크 99
- 그림 5-2 서로 다른 서비스 도메인에 연결된 각각의 가상 네트워크 장치 100
- 그림 5-3 IPMP 그룹의 일부로 구성된 두 개의 네트워크 인터페이스 100

표

표 1-1	논리적 도메인 역할	3
표 2-1	ldm 하위 명령 및 사용자 인증	11
표 5-1	컨트롤(primary) 도메인의 예상되는 중지 또는 재부트 동작	83

코드 예

코드 예 3-1	다운로드한 Logical Domains 1.0.2 소프트웨어의 디렉토리 구조	20
코드 예 3-2	LDoms용으로 강화된 Solaris 구성 출력	22
코드 예 3-3	사용자 정의된 구성 프로파일 선택 출력	23
코드 예 3-4	install-ldm -d 스크립트의 실행 출력	25
코드 예 3-5	install-ldm -d none 스크립트의 실행 출력	26
코드 예 5-1	모든 ldm 하위 명령의 구문 사용법	50
코드 예 5-2	설치된 소프트웨어 버전	54
코드 예 5-3	모든 도메인에 대한 짧은 목록	54
코드 예 5-4	모든 도메인에 대한 긴 목록	55
코드 예 5-5	모든 도메인에 대한 확장 목록	56
코드 예 5-6	시스템 읽기 가능 목록	58
코드 예 5-7	도메인 상태	59
코드 예 5-8	도메인에 대한 변수 목록	59
코드 예 5-9	도메인에 대한 바인딩 목록	59
코드 예 5-10	구성 목록	60
코드 예 5-11	모든 서버 자원에 대한 목록	61
코드 예 5-12	서비스 목록	62
코드 예 5-13	한 개의 도메인에 대한 제약 조건 목록	63
코드 예 5-14	도메인에 대한 제약 조건(XML 형식)	64
코드 예 5-15	모든 도메인에 대한 제약 조건(시스템 읽기 가능 형식)	65
코드 예 5-16	논리적 도메인 구성의 구문 분석 가능한 긴 목록	72

머리말

Logical Domains(LDoms) 1.0.2 관리 설명서에서는 지원되는 서버, 블레이드 및 서버 모듈에서 Logical Domains Manager 1.0.2 소프트웨어에 대한 개요, 보안 고려 사항, 설치, 구성, 수정 및 일반 작업 실행에 대해 설명하는 자세한 정보와 절차를 제공합니다. 목록은 Logical Domains(LDoms) 1.0.2 릴리스 노트의 "지원 서버"를 참조하십시오. 본 설명서는 UNIX® 시스템 및 Solaris™ 운영 체제(Solaris OS)에 대한 작업 지식을 보유한 해당 서버의 시스템 관리자를 대상으로 합니다.

이 문서를 읽기 전에

UNIX 명령 및 프로시저와 Solaris 운영 체제에 대한 실제 작업 지식을 보유하지 않은 경우에는 시스템 하드웨어와 함께 제공된 Solaris OS 사용자 및 시스템 관리자 설명서를 읽고 UNIX 시스템 관리 교육 이수를 고려하십시오.

이 설명서의 구성

1장에서는 Logical Domains 소프트웨어의 개요에 대해 설명합니다.

2장에서는 Solaris Security Toolkit과 여기에서 논리적 도메인에 있는 Solaris OS에 보안을 제공하는 방법에 대해 설명합니다.

3장에서는 Logical Domains Manager 소프트웨어를 업그레이드 또는 설치 및 활성화하는 자세한 절차에 대해 설명합니다.

4장에서는 서비스 및 논리적 도메인을 설정하는 자세한 절차에 대해 설명합니다.

5장에서는 논리적 도메인을 관리하기 위해 Logical Domains 소프트웨어를 사용하여 일반 작업을 수행하는 기타 정보와 절차에 대해 설명합니다.

용어집은 LDoms 특정 약어, 머리글자, 용어 및 해당 정의에 대한 목록입니다.

UNIX 명령어 사용

이 설명서에는 시스템 종료, 시스템 부팅 및 장치 구성과 같은 기본적인 UNIX 명령어 및 절차에 대한 정보는 포함되어 있지 않을 수 있습니다. 이러한 정보에 대해서는 다음을 참조하여 주십시오.

- 시스템에 포함되어 있는 소프트웨어 설명서
- Solaris 운영 체제 설명서는 다음 url을 참조하여 주시기 바랍니다.

<http://docs.sun.com>

셸 프롬프트

셸	프롬프트
C 셸	<i>machine-name%</i>
C 셸 슈퍼유저	<i>machine-name#</i>
Bourne 셸 및 Korn 셸	\$
Bourne 셸 및 Korn 셸 슈퍼유저	#

활자체 규약

활자체 또는 기호*	의미	예
AaBbCc123	명령 및 파일, 디렉토리 이름; 컴퓨터 화면에 출력되는 내용입니다.	.login 파일을 편집하십시오. 모든 파일 목록을 보려면 <code>ls -a</code> 명령을 사용하십시오. % You have mail.
AaBbCc123	사용자가 입력하는 내용으로 컴퓨터 화면의 출력 내용과 대조됩니다.	% su Password:
AaBbCc123	새로 나오는 용어, 강조 표시할 용어입니다. 명령줄 변수를 실제 이름이나 값으로 바꾸십시오.	<i>class</i> 옵션입니다. 이를 실행하기 위해서는 반드시 슈퍼유저여야 합니다. 파일 삭제 명령은 rm filename 입니다.
AaBbCc123	책 제목, 장, 절	Solaris 사용자 설명서 6장 데이터 관리를 참조하시기 바랍니다.

* 사용자가 사용하는 브라우저의 설정과 이 설정은 다를 수 있습니다.

관련 문서

Logical Domains(LDoms) 1.0.2 관리 설명서 및 릴리스 노트는 다음 사이트에 있습니다.

<http://docs.sun.com>

Beginners Guide to LDoms: Understanding and Deploying Logical Domains Software는 다음 주소의 Sun BluePrints™ 사이트에 있습니다.

<http://www.sun.com/blueprints/0207/820-0832.html>

서버, 소프트웨어 또는 Solaris OS와 관련된 문서는 다음 사이트에 있습니다.

<http://docs.sun.com>

원하는 문서를 찾으려면 검색 상자에 서버, 소프트웨어 또는 Solaris OS 이름을 입력하십시오.

응용 프로그램	제목	부품 번호	형식	위치
LDoms 릴리스 노트	Logical Domains(LDoms) 1.0.2 릴리스 노트	820-4459-10	HTML PDF	온라인
LDoms용 Solaris 매뉴얼 페이지	Solaris 10 Reference Manual 모음: <ul style="list-style-type: none">• drd(1M) 매뉴얼 페이지• vntsd(1M) 매뉴얼 페이지	해당 없음	HTML	온라인
LDoms 매뉴얼 페이지	ldm(1M) 매뉴얼 페이지	해당 없음	SGML	온라인
	Logical Domains(LDoms) 1.0.1 Manager Man Page Guide	819-7679-10	PDF	온라인
Logical Domains 소프트웨어 기본 사항	Beginners Guide to LDoms: Understanding and Deploying Logical Domains Software	820-0832-20	PDF	온라인
LDoms MIB 관리	Logical Domains(LDoms) MIB 1.0.1 관리 설명서	820-2319-10	HTML PDF	온라인
LDoms MIB 릴리스 노트	Logical Domains(LDoms) MIB 1.0.1 릴리스 노트	820-2320-10	HTML PDF	온라인
Solaris OS(설치, JumpStart™ 사용 및 SMF 사용)	Solaris 10 모음	해당 없음	HTML PDF	온라인
보안	Solaris Security Toolkit 4.2 관리 안내서	819-3709-10	HTML PDF	온라인

응용 프로그램	제목	부품 번호	형식	위치
보안	Solaris Security Toolkit 4.2 Reference Manual	819-1503-10	HTML PDF	온라인
보안	Solaris Security Toolkit 4.2 릴리스 노트	819-3797-10	HTML PDF	온라인
보안	Solaris Security Toolkit 4.2 Man Page Guide	819-1505-10	HTML PDF	온라인

설명서, 지원 및 교육

Sun 기능	URL
설명서	http://docs.sun.com
지원	http://www.sun.com/support
교육	http://www.sun.com/training

타사 웹 사이트

Sun은 본 설명서에서 언급된 타사 웹 사이트의 가용성 여부에 대해 책임을 지지 않습니다. 또한 해당 사이트나 리소스를 통해 제공되는 내용, 광고, 제품 및 기타 자료에 대해 어떠한 보증도 하지 않으며 그에 대한 책임도 지지 않습니다. 따라서 타사 웹 사이트의 내용, 제품 또는 리소스의 사용으로 인해 발생한 실제 또는 주장된 손상이나 피해에 대해서도 책임을 지지 않습니다.

Sun은 여러분의 의견을 환영합니다

Sun은 설명서의 내용 개선에 노력을 기울이고 있으며, 여러분의 의견과 제안을 환영합니다. 다음 사이트에 여러분의 의견을 제출하여 주십시오.

<http://www.sun.com/hwdocs/feedback>

아래와 같이 설명서의 제목과 부품 번호를 함께 적어 보내주시기 바랍니다.

Logical Domains(LDoms) 1.0.2 관리 설명서, 부품 번호 820-4453-10

Logical Domains 소프트웨어 개요

이 장에서는 Logical Domains 소프트웨어의 개요에 대해 설명합니다. Sun의 Logical Domains 기술을 사용하는 데 필요한 모든 Solaris OS 기능은 필요한 패치가 추가되어 있는 Solaris 10 11/06 릴리스(최소)에서 제공됩니다. 하지만 논리적 도메인을 사용하려면 시스템 펌웨어와 Logical Domains Manager도 필요합니다. 목록은 Logical Domains(LDoms) 1.0.2 릴리스 노트의 지원 서버 절을 참조하십시오.

하이퍼바이저(hypervisor) 및 Logical Domains

이 절에서는 SPARC® 하이퍼바이저 및 하이퍼바이저에서 지원하는 논리적 도메인에 대해 간략하게 설명합니다.

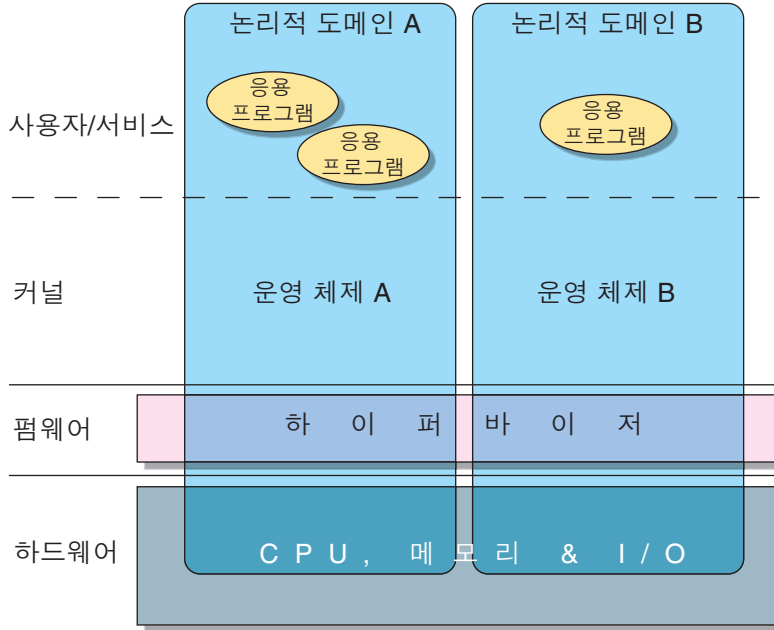
SPARC 하이퍼바이저는 운영 체제를 기록할 수 있는 안정된 가상 시스템 아키텍처를 제공하는 작은 펌웨어 계층입니다. 하이퍼바이저를 사용하는 Sun 서버는 논리적 운영 체제 활동에 대한 하이퍼바이저의 제어를 지원하는 하드웨어 기능을 제공합니다.

논리적 도메인은 단일 컴퓨터 시스템 내의 운영 체제, 자원 및 ID가 있는 개별 논리적 그룹화입니다. 서버의 전원을 순환시키지 않아도 독립적으로 각 논리적 도메인을 만들고, 삭제하고, 다시 구성하고, 재부트할 수 있습니다. 서로 다른 논리적 도메인에서 다양한 응용 프로그램 소프트웨어를 실행할 수 있으며, 성능과 보안을 위해 독립적으로 유지할 수 있습니다.

하이퍼바이저를 사용하면 각 논리적 도메인은 논리적 도메인에서 사용할 수 있는 서버 자원만 확인하고 상호 작용할 수 있습니다. Logical Domains Manager를 사용하여 시스템 관리자는 하이퍼바이저가 컨트롤 도메인을 통해 수행해야 하는 작업을 지정합니다. 즉, 하이퍼바이저는 서버 자원의 영역 분할을 강제로 시행하여 다중 운영 체제 환경에 제한된 일부를 제공합니다. 이 작업은 논리적 도메인을 만들기 위한 근본적인 메커니즘입니다. 다음 다이어그램은 2개의 논리적 도메인을 지원하는 하이퍼바이저를 보여 줍니다. 또한 Logical Domains 기능을 구성하는 계층도 보여 줍니다.

- 응용 프로그램 또는 사용자/서비스
- 커널 또는 운영 체제
- 펌웨어 또는 하이퍼바이저
- 하드웨어(CPU, 메모리 및 I/O 포함)

그림 1-1 2개의 논리적 도메인을 지원하는 하이퍼바이저



특정 SPARC 하이퍼바이저에서 지원하는 논리적 도메인의 수와 기능은 서버 기능에 따라 다릅니다. 하이퍼바이저는 지정된 논리적 도메인에 서버의 전체 CPU, 메모리 및 I/O 자원 중 일부를 할당합니다. 이를 통해 고유한 논리적 도메인 내에서 각각 다중 운영 체제가 동시에 지원됩니다. 개별 논리적 도메인 간 자원을 임의 단위로 재배열할 수 있습니다. 예를 들어 메모리는 8KB 단위로 논리적 도메인에 할당할 수 있습니다.

각 가상 시스템은 다음과 같은 고유한 자원을 사용하여 완전히 독립적인 시스템으로 관리할 수 있습니다.

- 커널, 패치 및 조정 매개 변수
- 사용자 계정 및 관리자
- 디스크
- 네트워크 인터페이스, MAC 주소 및 IP 주소

서버의 전원을 순환시키지 않아도 서로에 대해 독립적으로 각 가상 시스템을 중지하고, 시작하고 재부트할 수 있습니다.

하이퍼바이저 소프트웨어는 논리적 도메인 간 분리를 유지 관리하는 작업을 담당합니다. 또한 논리적 도메인 채널(Logical Domain Channel, LDC)을 제공하여 논리적 도메인이 서로 통신할 수 있습니다. 논리적 도메인 채널을 사용하면 도메인은 네트워킹 또는 디스크 서비스와 같은 서비스를 서로 제공할 수 있습니다.

시스템 제어기는 물리적 시스템을 모니터하고 실행하지만 가상 시스템은 관리하지 않습니다. 가상 시스템은 Logical Domains Manager에서 실행됩니다.

Logical Domains Manager

Logical Domains Manager는 논리적 도메인을 만들고 관리하는 데 사용되며 서버당 하나만 있을 수 있습니다. Logical Domains Manager는 논리적 도메인을 물리적 자원에 매핑합니다.

Logical Domains 역할

모든 논리적 도메인은 각 도메인에 대해 사용자가 지정한 역할을 제외하고는 역할이 모두 같습니다. 논리적 도메인에서 수행할 수 있는 여러 가지 역할이 있습니다.

표 1-1 논리적 도메인 역할

도메인 역할	설명
컨트롤 도메인	Logical Domains Manager에서 실행하여 사용자가 다른 논리적 도메인을 만들고 관리하며 가상 자원을 다른 도메인에 할당할 수 있게 하는 도메인입니다. 컨트롤 도메인은 서버당 하나만 있을 수 있습니다. Logical Domains 소프트웨어를 설치할 때 만들어진 초기 도메인이 컨트롤 도메인이며, primary로 이름이 지정됩니다.
서비스 도메인	다른 도메인에 가상 장치 서비스(예: 가상 스위치, 가상 콘솔 집중 장치 및 가상 디스크 서버)를 제공하는 도메인입니다.
I/O 도메인	물리적 I/O 장치(예: PCI Express 제어기의 네트워크 카드)의 직접 소유권과 직접 액세스 권한이 있는 도메인입니다. I/O 도메인이면서 컨트롤 도메인인 경우에는 가상 장치 형식으로 다른 도메인과 장치를 공유합니다. 보유할 수 있는 I/O 도메인 수는 사용자 플랫폼 아키텍처에 따라 다릅니다. 예를 들어 Sun UltraSPARC® T1 프로세서를 사용 중인 경우에는 최대 2개의 I/O 도메인을 가질 수 있으며, 그 중 하나는 컨트롤 도메인이어야 합니다.
게스트 도메인	컨트롤 도메인에서 관리되며 I/O 및 서비스 도메인에서 서비스를 사용하는 도메인입니다.

기존 시스템이 있고 다른 소프트웨어와 운영 체제가 서버에 이미 실행 중인 경우 Logical Domains Manager를 설치한 후 컨트롤 도메인이 됩니다. 설정되면 컨트롤 도메인에서 응용 프로그램 중 일부를 제거하고 도메인 전체에 대한 응용 프로그램의 로드 균형을 조정하여 시스템을 가장 효율적으로 사용할 수 있도록 만듭니다.

명령줄 인터페이스

Logical Domains Manager에서는 시스템 관리자가 논리적 도메인을 만들고 구성할 수 있도록 명령줄 인터페이스(command-line interface, CLI)를 제공합니다. CLI는 여러 개의 하위 명령이 있는 단일 명령인 `ldm(1M)`입니다.

Logical Domains Manager CLI를 사용하려면 Logical Domains Manager 데몬인 `ldmd`가 실행 중이어야 합니다. `ldm(1M)` 명령과 해당하는 하위 명령에 대한 자세한 내용은 `ldm(1M)` 매뉴얼 페이지와 Logical Domains(LDoms) 1.0.1 Manager Man Page Guide에서 설명합니다. `ldm(1M)` 매뉴얼 페이지는 `SUNWldm` 패키지의 일부로 `SUNWldm package` 패키지가 설치될 때 함께 설치됩니다.

`ldm` 명령을 실행하려면 UNIX `$PATH` 변수에 `/opt/SUNWldm/bin` 디렉토리가 있어야 합니다. `ldm(1M)` 매뉴얼 페이지에 액세스하려면 `$MANPATH` 변수에 디렉토리 경로 `/opt/SUNWldm/man`를 추가합니다. 다음과 같이 표시됩니다.

```
$ PATH=$PATH:/opt/SUNWldm/bin; export PATH (for Bourne or K shell)
$ MANPATH=$MANPATH:/opt/SUNWldm/man; export MANPATH
% set PATH=($PATH /opt/SUNWldm/bin) (for C shell)
% set MANPATH=($MANPATH /opt/SUNWldm/man)
```

가상 입/출력

논리적 도메인 환경에서 관리자는 Sun Fire™나 SPARC Enterprise T1000 또는 T2000 서버에 최대 32개의 도메인을 지정할 수 있습니다. 각 도메인에 전용 CPU와 메모리를 할당할 수 있으나 이러한 시스템의 제한된 I/O 버스와 물리적 I/O 슬롯의 수로 인해 모든 도메인에 디스크와 네트워크 장치에 대한 단독 액세스를 제공할 수는 없습니다. PCI Express®(PCI-E) 버스를 2개로 분할(74페이지의 "분할 PCI Express 버스를 구성하여 다중 Logical Domains 사용" 참조)하여 일부 물리적 장치를 공유할 수 있으나 모든 도메인에 단독 장치 액세스를 제공하기에는 충분하지 않습니다. 직접 물리적 I/O 장치의 액세스 부족은 가상 I/O 모델을 구현하여 해결합니다.

직접 I/O 액세스가 없는 논리적 도메인은 모두 서비스 도메인과 통신하는 가상 I/O 장치로 구성됩니다. 이를 통해 서비스가 실행되어 액세스를 물리적 장치 또는 해당 기능에 내보냅니다. 이 클라이언트-서버 모델에서 가상 I/O 장치는 논리적 도메인 채널(Logical Domain Channel, LDC)이라고 하는 도메인 간 통신 채널을 통해 서로 통신하거나 서비스 대응 항목과 통신합니다. Logical Domains 1.0.2 소프트웨어에서 가상 I/O 기능은 가상 네트워킹, 저장소 및 콘솔에 대한 지원을 구성합니다.

가상 네트워크

가상 네트워크 지원은 두 가지 구성 요소인 가상 네트워크와 가상 네트워크 스위치 장치를 사용하여 구현됩니다. 가상 네트워크(vnet) 장치는 이더넷 장치를 에뮬레이트하고 지점 간 채널을 사용하여 시스템의 다른 vnet 장치와 통신합니다. 가상 스위치(vsw) 장치는 주로 가상 네트워크의 모든 수신과 발신 패킷의 다중화 장치로 작동합니다. vsw 장치는 서비스 도메인에서 물리적 네트워크 어댑터와 직접 상호 작용하고 가상 네트워크 대신 패킷을 보내고 받습니다. vsw 장치는 단순 계층 2 스위치로도 작동하여 시스템 내에서 해당 장치에 연결된 vnet 장치 간에 패킷을 전환합니다.

가상 저장소

가상 저장소 인프라를 사용하여 논리적 도메인이 클라이언트-서버 모델에서 논리적 도메인에 직접 할당되지 않는 블록 레벨 저장소에 액세스할 수 있습니다. 가상 저장소는 두 가지 구성 요소인 가상 디스크 클라이언트와 가상 디스크 서비스로 구성됩니다. 가상 디스크 클라이언트(vdc)에서는 블록 장치 인터페이스로 내보내고, 가상 디스크 서비스(vds)에서는 가상 디스크 클라이언트 대신 디스크 요청을 처리하여 해당 요청을 서비스 도메인에 있는 물리적 저장소에 전송합니다. 가상 디스크가 클라이언트 도메인에서 정상적인 디스크로 나타나지만 모든 디스크 작업은 가상 디스크 서비스를 통해 물리적 디스크에 전달됩니다.

가상 콘솔

Logical Domains 환경에서 primary 도메인을 제외한 모든 도메인의 콘솔 I/O는 시스템 제어기 대신 가상 콘솔 집중 장치(vcc)와 가상 네트워크 터미널 서버를 실행하는 서비스 도메인으로 리디렉션됩니다. 가상 콘솔 집중 장치 서비스는 모든 도메인의 콘솔 트래픽에 대한 집중 장치로 작동하며, 가상 네트워크 터미널 서버 데몬(vntsd)과 상호 작용하고, UNIX 소켓을 통해 각 콘솔에 대한 액세스를 내보냅니다.

동적 재구성

동적 재구성(dynamic reconfiguration, DR)은 운영 체제가 실행하는 동안 자원을 추가하고 제거하는 기능입니다. Solaris 10 OS에서는 가상 CPU(vcpu)의 추가 및 제거만 지원됩니다. 메모리와 입/출력의 동적 재구성은 Solaris 10 OS에서 지원되지 않습니다. Logical Domains Manager CLI에서 동적 재구성 기능을 사용하려면 변경하려는 도메인에 Logical Domains 동적 재구성 데몬인 drd(1M)가 실행 중이어야 합니다.

지연된 재구성

즉시 발생하는 동적 재구성 작업과는 반대로 지연된 재구성 작업은 OS의 다음 재부트 이후에 적용되거나 OS가 실행 중이 아니면 논리적 도메인의 중지 및 시작 이후에 적용됩니다. 활성 논리적 도메인에서 `add-vcpu`, `set-vcpu` 및 `remove-vcpu` 하위 명령을 제외한 추가 또는 제거 작업은 모두 지연된 재구성 작업으로 간주됩니다. 활성 논리적 도메인에서 `set-vswitch` 하위 명령도 지연된 재구성 작업으로 간주됩니다.

Sun UltraSPARC T1 프로세서를 사용 중인 경우 Logical Domains Manager가 처음 설치되고 활성화되거나 구성이 `factory-default`)로 복구되면 LDoms Manager가 구성 모드에서 실행됩니다. 이 모드에서 재구성 요청이 허용되고 대기열에 기록되거나 작업을 수행하지는 않습니다. 이를 통해 실행 중인 시스템의 상태에 영향을 미치지 않고, 즉 I/O 도메인의 지연된 재구성 또는 재부트와 같이 주변의 제한 사항으로 방해받지 않고 새 구성이 생성되어 SC에 저장됩니다.

지연된 재구성이 특정 논리적 도메인에서 진행 중이면 해당 논리적 도메인의 다른 모든 재구성 요청도 도메인이 재부트되거나 중지 및 시작할 때까지 지연됩니다. 또한 한 개의 논리적 도메인에 미해결된 지연된 재구성이 있는 경우 다른 논리적 도메인의 재구성 요청도 심하게 제한되어 적절한 오류 메시지와 함께 장애가 발생합니다.

활성 논리적 도메인에서 가상 I/O 장치를 제거하려는 시도가 지연된 재구성 작업으로 처리된다 하더라도 일부 구성 변경이 즉시 발생합니다. 이는 관련 Logical Domains Manager CLI 작업이 호출되자마자 장치가 사실상 작동을 중지하게 됨을 의미합니다.

Logical Domains Manager의 하위 명령 `remove-reconf`는 지연된 재구성 작업을 취소합니다. `ldm list-domain` 명령을 사용하여 지연된 재구성 작업을 나열할 수 있습니다. 지연된 재구성 기능을 사용하는 방법에 대한 자세한 내용은 `ldm(1M)` 매뉴얼 페이지 또는 Logical Domains(LDoms) 1.0.1 Manager Man Page Guide를 참조하십시오.

주 - 다른 `ldm remove-*` 명령이 가상 I/O 장치에서 실행되면 `ldm remove-reconf` 명령을 사용할 수 없습니다. 이 경우 `ldm remove-reconf` 명령이 실패합니다.

영구 구성

현재 논리적 도메인 구성은 Logical Domains Manager CLI 명령을 사용하여 시스템 제어기(System Controller, SC)에 저장할 수 있습니다. 구성을 추가하고, 사용할 수 있는 구성을 지정하고, 구성을 제거하고, 시스템 제어기에 구성을 나열할 수 있습니다. `ldm(1M)` 매뉴얼 페이지 또는 Logical Domains(LDoms) 1.0.1 Manager Man Page Guide를 참조하십시오. 추가로 구성을 선택하여 부트할 수 있게 하는 ALOM CMT 버전 1.3 명령이 있습니다(84페이지의 "ALOM CMT와 함께 Ldoms를 사용" 참조).

보안

이 장에서는 Solaris Security Toolkit 소프트웨어와 논리적 도메인에서 해당 소프트웨어를 사용하여 Solaris OS를 보호하는 방법에 대해 설명합니다.

보안 고려 사항

JASS(JumpStart™ Architecture and Security Scripts) 툴킷이라는 이전 명칭으로도 알려진 Solaris Security Toolkit 소프트웨어는 자동화되고 확장 가능한 메커니즘을 제공하여 보안 Solaris OS 시스템을 구축하고 유지 관리할 수 있습니다. Solaris Security Toolkit은 Logical Domains Manager의 컨트롤 도메인을 비롯하여 서버 관리에 중요한 장치에 대한 보안을 제공합니다.

Solaris Security Toolkit 4.2 소프트웨어 패키지인 SUNWjass는 다음을 수행하여 `install-ldm` 스크립트의 사용을 통해 컨트롤 도메인에 Solaris 운영 체제를 보호하는 방법을 제공합니다.

- Logical Domains Manager 설치 스크립트(`install-ldm`)와 Logical Domains Manager에 한정된 제어 드라이버(`ldm_control-secure.driver`)를 사용하여 Solaris Security Toolkit에서 사용 중인 컨트롤 도메인 강화
- 설치 스크립트를 사용 중인 경우 대체 드라이버 선택
- 설치 스크립트를 사용 중이고 자체 Solaris 강화를 적용하는 경우 드라이버를 선택하지 않음

SUNWjass 패키지는 Logical Domains(LDoms) Manager 1.0.2 소프트웨어 패키지인 SUNWldm과 함께 Sun 소프트웨어 다운로드 웹 사이트에 있습니다. Logical Domains Manager 1.0.2 소프트웨어를 다운로드하여 설치할 때 Solaris Security Toolkit 4.2 소프트웨어 패키지를 같이 다운로드하여 설치할 수 있는 옵션이 있습니다. Solaris Security Toolkit 4.2 소프트웨어 패키지에는 Solaris Security Toolkit 소프트웨어가 Logical Domains Manager와 함께 작동할 수 있게 하는 필수 패치가 포함되어 있습니다. 소프트웨어가 설치되면 Solaris Security Toolkit 4.2 소프트웨어를 사용하여 시스템을 강화할 수 있습니다. 3장에서는 Solaris Security Toolkit을 설치하고 구성하는 방법과 컨트롤 도메인을 강화하는 방법에 대해 설명합니다.

다음은 Logical Domains Manager 사용자가 사용할 수 있는 보안 기능으로 Solaris Security Toolkit에서 제공됩니다.

- 강화 - Solaris Security Toolkit이 Logical Domains Manager와 함께 작동할 수 있게 하는 필수 패치가 있는 Solaris Security Toolkit 4.2 소프트웨어를 사용하여 Solaris OS 구성을 수정하여 시스템의 보안을 향상시킵니다.
- 최소화 - LDoms 및 LDoms Management Information Base(MIB) 지원을 위해 필요한 최소 수의 핵심 Solaris OS 패키지를 설치합니다.
- 인증 - Logical Domains Manager에 대해 조정된 Solaris OS 역할별 접근 제어(Role-Based Access Control, RBAC)를 사용하여 인증을 설정합니다.
- 감사 - Logical Domains Manager에 대해 조정된 Solaris OS 기본 보안 모듈(Basic Security Module, BSM)을 사용하여 시스템에서 변경된 보안 사항의 소스를 식별하여 수행된 작업, 작업 시기, 작업 수행자 및 해당 결과를 확인합니다.
- 준수 - Solaris Security Toolkit의 감사 기능을 사용하여 시스템의 구성이 미리 정의된 보안 프로필을 준수하는지 확인합니다.

Solaris Security Toolkit 및 Logical Domains Manager

3장에서는 Solaris Security Toolkit을 설치하여 Logical Domains Manager와 함께 작동하는 방법을 알려 줍니다. Logical Domains Manager가 실행 중인 컨트롤 도메인에 Solaris Security Toolkit을 설치합니다. 다른 논리적 도메인에 Solaris Security Toolkit을 설치할 수도 있습니다. 유일한 차이는 컨트롤 도메인을 강화하려면 `ldm_control-secure.driver`를 사용하고, 다른 논리적 도메인을 강화하려면 `secure.driver`와 같은 다른 드라이버를 사용한다는 점입니다. `ldm_control-secure.driver`가 컨트롤 도메인에만 국한되기 때문입니다. `ldm_control-secure.driver`는 `secure.driver`를 기반으로 하며, Logical Domains Manager와 함께 사용하기 위해 사용자 정의되고 테스트되었습니다. `secure.driver`에 대한 자세한 내용은 Solaris Security Toolkit 4.2 Reference Manual을 참조하십시오.

강화

컨트롤 도메인에서 Solaris OS를 강화하기 위해 Solaris Security Toolkit에서 사용하는 드라이버(ldm_control-secure.driver)는 특별히 맞춤형으로 조정되어 Logical Domains Manager가 OS와 함께 실행될 수 있습니다. ldm_control-secure.driver는 Solaris Security Toolkit 4.2 Reference Manual에서 설명한 secure.driver와 유사합니다.

ldm_control-secure.driver는 Logical Domains Manager 소프트웨어를 실행하는 시스템의 컨트롤 도메인에 기본 구성을 제공합니다. 일반적인 Solaris OS 도메인의 경우보다 더 적은 시스템 서비스를 제공하여 일반적인 사용 보다는 Logical Domains Manager 작업을 위한 컨트롤 도메인을 확보합니다.

Logical Domains Manager 소프트웨어가 아직 설치되어 있지 않은 경우 install-ldm 스크립트로 해당 소프트웨어를 설치하고 활성화합니다.

다음은 secure.driver에서 변경된 기타 주목할 만한 사항에 대해 간략히 요약합니다.

- 텔넷 서버가 실행할 수 없게 됩니다. 대신 보안 셸(ssh)을 사용할 수 있습니다. 텔넷 클라이언트를 사용하여 Logical Domains 가상 네트워크 터미널 서버 데몬(vntsd)에서 시작된 가상 콘솔에 액세스할 수도 있습니다. 예를 들어 로컬 시스템의 TCP 포트 5001을 수신 중인 가상 콘솔이 실행 중인 경우 다음과 같이 해당 콘솔에 액세스할 수 있습니다.

```
# telnet localhost 5001
```

vntsd를 활성화하는 지침에 대한 자세한 내용은 [32페이지의 "Logical Domains Manager 데몬 활성화"](#)를 참조하십시오. 자동으로 활성화되지 않습니다.

- 다음 완료 스크립트가 추가되었습니다. 이러한 스크립트를 통해 Logical Domains Manager를 설치하고 시작합니다. 추가된 스크립트 중 일부는 사용자 정의된 드라이버에 반드시 추가되어야 하며, 일부는 선택 사항입니다. 스크립트는 필수 또는 선택 사항인지 여부에 대해 표시됩니다.
 - install-ldm.fin - SUNWldm 패키지를 설치합니다(필수).
 - enable-ldmd.fin - Logical Domains Manager 데몬(ldmd)을 활성화합니다(필수).
 - enable-ssh-root-login.fin - 슈퍼유저가 보안 셸(ssh)을 통해 직접 로그인할 수 있습니다(선택 사항).
- 다음 파일이 변경되었습니다. 이러한 변경 사항은 선택 사항으로 사용자에게 있는 사용자 정의된 드라이버에서 변경되며 선택 사항으로 표시됩니다.
 - /etc/ssh/sshd_config - 루트 계정 액세스가 전체 네트워크에서 허용됩니다. 이 파일은 다른 드라이버에서 사용되지 않습니다(선택 사항).
 - /etc/ipf/ipf.conf - UDP 포트 161(SNMP)이 열려 있습니다(선택 사항).
 - /etc/host.allow - 보안 셸 데몬(sshd)이 로컬 서브넷이 아니라 전체 네트워크에서 열려 있습니다(선택 사항).

- 다음 완료 스크립트가 비활성화됩니다(주석 처리됨). 사용자 정의된 드라이버에 `disable-rpc.fin` 스크립트를 주석 처리해야 합니다. 다른 변경 사항은 선택 사항입니다. 스크립트는 필수 또는 선택 사항인지 여부에 대해 표시됩니다.
 - `enable-ipfilter.fin` - IP 필터, 즉 네트워크 패킷 필터가 활성화되지 않습니다(선택 사항).
 - `disable-rpc.fin` - 원격 프로시저 호출(Remote Procedure Call, RPC)을 활성화 상태로 유지합니다. RPC 서비스는 네트워크 정보 서비스(Network Information Service, NIS) 및 네트워크 파일 시스템(Network File System, NFS) 과 같은 기타 많은 시스템에서 사용됩니다(필수).
 - `disable-sma.fin` - 시스템 관리 에이전트(System Management Agent, SMA)를 활성화 상태로 유지합니다(선택 사항).
 - `disable-ssh-root-login.fin` - ssh 루트 로그인을 비활성화할 수 없습니다.
 - `set-term-type.fin` - 필요하지 않은 레거시 스크립트입니다(선택 사항).

Logical Domains 최소화

Solaris OS는 사용자의 필요에 따라 서로 다른 패키지 수량으로 구성됩니다. 최소화는 이러한 패키지 세트를 사용자가 원하는 응용 프로그램을 실행하는 데 필요한 최소 수량으로 줄입니다. 최소화는 잠재적인 보안 위험성이 포함된 소프트웨어의 양을 줄이고 설치된 소프트웨어의 패치 적용이 적절히 유지되도록 하는 작업에 대한 수고를 덜어주므로 중요합니다. 논리적 도메인 최소화 작업에서는 도메인을 모두 지원하는 최소화된 Solaris OS를 설치할 수 있도록 JumpStart™ 지원을 제공합니다.

Solaris Security Toolkit은 LDoms의 논리적 도메인을 최소화기 위해 JumpStart 프로파일인 `minimal-ldm_control.profile`을 제공하여 LDoms 및 LDoms MIB 지원에 필요한 모든 Solaris OS 패키지를 설치합니다. 컨트롤 도메인에서 LDoms MIB를 사용하면 LDoms 및 Solaris Security Toolkit 패키지를 설치한 후 해당 패키지를 각각 추가해야 합니다. 다른 소프트웨어와 함께 자동으로 설치되지 않습니다. LDoms MIB를 설치하고 사용하는 방법에 대한 자세한 내용은 Logical Domains(LDoms) MIB 1.0.1 관리 설명서를 참조하십시오.

인증

Logical Domains Manager에 대한 인증에는 다음 두 가지 수준이 있습니다.

- 읽기 - 구성을 볼 수 있으나 수정할 수 없습니다.
- 읽기 및 쓰기 - 구성을 보고 변경할 수 있습니다.

변경 사항은 Solaris OS에는 적용되지 않으나 Logical Domains Manager가 설치되면 패키지 스크립트 `postinstall`에 의해 인증 파일에 추가됩니다. 이와 유사하게 인증 항목은 패키지 스크립트 `preremove`에 의해 제거됩니다.

다음 표에는 명령을 수행하기 위해 필요한 해당 사용자 인증이 있는 ldm 하위 명령이 나열되어 있습니다.

표 2-1 ldm 하위 명령 및 사용자 인증

ldm 하위 명령*	사용자 인증
add-*	solaris.ldoms.write
bind-domain	solaris.ldoms.write
list	solaris.ldoms.read
list-*	solaris.ldoms.read
panic-domain	solaris.ldoms.write
remove-*	solaris.ldoms.write
set-*	solaris.ldoms.write
start-domain	solaris.ldoms.write
stop-domain	solaris.ldoms.write
unbind-domain	solaris.ldoms.write

* 추가, 나열, 제거 및 설정할 수 있는 자원을 모두 참조합니다.

감사

Logical Domains Manager CLI 명령의 감사는 Solaris OS 기본 보안 모듈(Basic Security Module, BSM) 감사로 수행됩니다. IPMP 그룹 구성 및 사용 방법에 대한 자세한 내용은 Solaris 10 System Administration Guide: Security Services를 참조하십시오.

BSM 감사는 Logical Domains Manager에서 기본적으로 활성화되어 있지 않지만 인프라가 제공됩니다. 다음 두 가지 방법 중 하나를 사용하여 BSM 감사를 활성화할 수 있습니다.

- Solaris Security Toolkit에서 `enable-bsm.fin` 완료 명령 실행
- Solaris OS `bsmconv(1M)` 명령을 사용합니다.

Logical Domains Manager로 BSM 감사를 사용하여 로그를 활성화, 확인, 비활성화, 출력 인쇄 및 회전하는 방법에 대한 자세한 내용은 85페이지의 "BSM 감사 활성화 및 사용"을 참조하십시오.

준수

Solaris Security Toolkit에는 고유한 감사 기능이 있습니다. Solaris Security Toolkit 소프트웨어에서는 Solaris OS를 실행하는 시스템의 보안 스타일을 미리 정의된 보안 프로필과 비교하여 자동으로 검증할 수 있습니다. 이 준수 기능에 대한 자세한 내용은 Solaris Security Toolkit 4.2 Administration Guide의 "Auditing System Security"를 참조하십시오.

소프트웨어 설치 및 활성화

이 장은 지원되는 서버의 컨트롤 도메인에서 Logical Domains Manager 1.0.2 소프트웨어와 기타 소프트웨어를 설치하고 활성화하는 방법에 대해 설명합니다. 지원 되는 서버 목록은 Logical Domains(LDoms) 1.0.2 릴리스 노트의 "지원 서버"를 참조하십시오.

사용자 플랫폼에 따라 이 장에서 필요한 내용을 사용할 수 있습니다. 새 Sun UltraSPARC T2 플랫폼에서 Logical Domains 소프트웨어를 사용 중인 경우에는 모든 소프트웨어가 출하 시 사전 설치되어 있어야 합니다.

Solaris OS 업그레이드

이 절에는 Logical Domains 제약 조건 데이터베이스 파일을 저장 및 복구하거나 컨트롤 도메인에서 라이브 업그레이드를 수행하는 데 필요한 정보가 들어 있습니다.

Logical Domains 제약 조건 데이터베이스 파일 저장 및 복구

컨트롤 도메인에서 운영 체제를 업그레이드할 때마다 `/var/opt/SUNWldm/ldom-db.xml`에서 찾을 수 있는 Logical Domains 제약 조건 데이터베이스 파일을 저장하고 복구해야 합니다.

주 - 또한 디스크 스왑 등 컨트롤 도메인의 파일 데이터를 손상시키는 기타 작업을 수행할 때도 `/var/opt/SUNWldm/ldom-db.xml` 파일을 저장 및 복구해야 합니다.

컨트롤 도메인에서 라이브 업그레이드 사용

컨트롤 도메인에서 라이브 업그레이드를 사용하는 경우에는 `/etc/lu/synclist` 파일에 다음 행 추가를 고려하십시오.

```
/var/opt/SUNWldm/ldm-db.xml OVERWRITE
```

이 행을 추가하면 부트 환경을 전환할 때 해당 데이터베이스가 자동으로 활성 부트 환경에서 새 부트 환경으로 복사됩니다. `/etc/lu/synclist`와 부트 환경 간의 파일 동기화에 대해서는 Solaris 10 8/07 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획에서 "부트 환경 사이의 파일 동기화"를 참조하십시오.

LDoms 1.0.2 소프트웨어로 업그레이드

기존 LDoms 1.0.1 구성이 LDoms 1.0.2 소프트웨어에서 작동하므로 LDoms 1.0.1 소프트웨어에서 LDoms 1.0.2 소프트웨어로 업그레이드하는 경우 다음 절차를 수행하지 않아도 됩니다. 그러나 기존 LDoms 1.0 구성을 LDoms 1.0.2 소프트웨어에서 사용하려면 다음 절차를 사용해야 합니다.

▼ LDoms 1.0에서 LDoms 1.0.2 소프트웨어로 업그레이드하려면

기존 LDoms 1.0 구성이 LDoms 1.0.2 소프트웨어에서 작동하지 않습니다. 다음 절차는 XML 제약 조건 파일과 `ldm start-domain` 명령에 `-i` 옵션을 사용하여 구성을 저장하고 재구성하는 방법에 대해 설명합니다. 이 방법은 실제 바인딩은 보존되지 않고 해당 바인딩을 만드는 데 사용되는 제약 조건만 보존됩니다. 즉 이 절차 후 도메인에 동일한 가상 자원이 있게 되지만 반드시 동일한 물리적 자원으로 바인드되지는 않습니다.

기본 프로세스는 각 도메인의 제약 조건 정보를 XML 파일에 저장하는 것입니다. 이 프로세스는 원하는 구성을 재구성하기 위해 업그레이드한 후 Logical Domains Manager에서 다시 실행할 수 있습니다. 이 절차는 컨트롤 도메인이 아니라 게스트 도메인에서 작동합니다. 제어(primary) 도메인의 제약 조건을 XML 파일에 저장할 수 있지만 `ldm start-domain -i` 명령으로 피드백을 전송할 수 없습니다.

1. Solaris OS의 최신 버전으로 업데이트합니다. 자세한 내용은 2단계, 16페이지의 "Solaris 10 OS를 설치하려면"을 참조하십시오.
2. 각 도메인에 대해 도메인 제약 조건을 포함하는 XML 파일을 만듭니다.

```
# ldm ls-constraints -x ldom > ldom.xml
```

3. 시스템 제어기에 저장된 논리적 도메인 구성을 모두 나열합니다.

```
# ldm ls-config
```

4. 시스템 제어기에 저장된 논리적 도메인 구성을 제거합니다.

```
# ldm rm-config config_name
```

5. Logical Domains Manager 데몬(ldmd)을 비활성화합니다.

```
# svcadm disable ldmd
```

6. Logical Domains Manager 패키지(SUNWldm)를 제거합니다.

```
# pkgrm SUNWldm
```

7. Solaris Security Toolkit 패키지(SUNWjass)를 사용 중인 경우에는 제거합니다.

```
# pkgrm SUNWjass
```

8. 시스템 펌웨어를 플래시 업데이트합니다. 전체 절차에 대한 자세한 내용은 [17페이지](#)의 "시스템 펌웨어를 업그레이드하려면" 또는 [18페이지](#)의 "FTP 서버를 사용하지 않고 시스템 펌웨어를 업그레이드하려면"을 참조하십시오.

9. LDoms 1.0.2 소프트웨어 패키지를 다운로드합니다.

Logical domains Manager, Solaris Security Toolkit 및 Logical Domains MIB를 다운로드하여 설치하는 절차에 대한 자세한 내용은 [19페이지](#)의 "Logical Domains Manager, Solaris Security Toolkit 및 Logical Domains MIB를 다운로드하려면"을 참조하십시오.

10. primary 도메인을 수동으로 재구성합니다. 방법은 [40페이지](#)의 "컨트롤 도메인을 설정하려면"을 참조하십시오.

11. 2단계에서 만든 각 게스트 도메인의 XML 파일에 다음 명령을 실행합니다.

```
# ldm create -i ldom.xml
# ldm bind-domain ldom
# ldm start-domain ldom
```

컨트롤 도메인에서 소프트웨어 새로 설치

Logical Domains Manager 소프트웨어를 설치할 때 만들어지는 첫 번째 도메인이 컨트롤 도메인입니다. 첫 번째 도메인의 이름은 `primary`로 지정되며 이름을 변경할 수 없습니다. 다음 주요 구성 요소가 컨트롤 도메인에 설치됩니다.

- Solaris 10 OS. 필요한 경우 Logical Domains(LDoms)1.0.2 릴리스 노트에서 권장하는 패치 추가. [16페이지의 "Solaris 10 OS를 설치하려면"](#)을 참조하십시오.
- 사용자 Sun UltraSPARC T1 플랫폼용 시스템 펌웨어 버전 6.5 또는 Sun UltraSPARC T2 플랫폼용 시스템 펌웨어 버전 7.0. [17페이지의 "시스템 펌웨어를 업그레이드하려면"](#)을 참조하십시오.
- Logical Domain Manager 1.0.2 소프트웨어. [20페이지의 "Logical Domains Manager 및 Solaris Security Toolkit 설치"](#)를 참조하십시오.
- (선택 사항) Solaris Security Toolkit 4.2 소프트웨어. [20페이지의 "Logical Domains Manager 및 Solaris Security Toolkit 설치"](#)를 참조하십시오.
- (선택 사항) Logical Domains(LDoms) Management Information Base(MIB) 소프트웨어 패키지. LDoms MIB 설치 및 사용에 대한 자세한 내용은 Logical Domains (LDoms) Management Information Base(MIB) 1.0.1 관리 설명서를 참조하십시오.

Solaris OS 및 시스템 펌웨어는 Logical Domains Manager를 설치하기 전에 서버에 설치되어 있어야 합니다. Solaris OS, 시스템 펌웨어 및 Logical Domains Manager가 설치된 후 원래 도메인은 컨트롤 도메인이 됩니다.

▼ Solaris 10 OS를 설치하려면

Solaris 10 OS를 아직 설치하지 않은 경우 해당 OS를 설치합니다. 이 버전의 Logical Domains 소프트웨어에 대해 사용해야 하는 Solaris 10 OS를 찾으려면 Logical Domains(LDoms) 1.0.2 릴리스 노트의 "필수 소프트웨어"를 참조하십시오. Solaris OS를 설치하는 데 필요한 전체 지침은 Solaris 10 OS 설치 설명서를 참조하십시오. 시스템의 필요에 따라 맞춤형 설치를 수행할 수 있습니다.

주 - 논리적 도메인의 경우 블록 장치로 내보내진 전체 디스크 또는 파일에만 Solaris OS를 설치할 수 있습니다.

1. Solaris 10 OS를 설치합니다.

최소화는 선택 사항입니다. Solaris Security Toolkit은 Logical Domains 소프트웨어에 대해 다음과 같은 JumpStart 최소화 프로필을 제공합니다.

```
/opt/SUNWjass/Profiles/minimal-ldm_control.profile
```

2. Solaris 10 11/06 OS를 설치하는 중이면 필수 패치를 설치합니다. 필수 패치 목록은 Logical Domains(LDoms) 1.0.2 릴리스 노트의 "필수 Solaris 10 11/06 OS 패치"를 참조하십시오.

주 - 게스트 도메인에 영어가 아닌 언어 운영 체제를 설치하는 경우 콘솔의 단말기는 운영 체제 설치 프로그램 필요한 로케일에 있어야 합니다. 예를 들어 Solaris OS 설치 프로그램은 EUC 로케일을 필요하고 Linux 설치 프로그램은 유니코드를 필요합니다.

▼ 시스템 펌웨어를 업그레이드하려면

다음 SunSolve 사이트에서 사용자 플랫폼용 시스템 펌웨어를 찾을 수 있습니다.

<http://sunsolve.sun.com>

지원 되는 서버에서 필요한 시스템 펌웨어는 Logical Domains(LDoms) 1.0.2 릴리스 노트의 "필수 시스템 펌웨어 패치"를 참조하십시오.

이 절차에서는 시스템 제어기에서 flashupdate(1M) 명령을 사용하여 시스템 펌웨어를 업그레이드하는 방법에 대해 설명합니다.

- 로컬 FTP 서버에 대한 액세스 권한이 없으면 18페이지의 "FTP 서버를 사용하지 않고 시스템 펌웨어를 업그레이드하려면"을 참조하십시오.
- 컨트롤 도메인에서 시스템 펌웨어를 업데이트하려는 경우에는 시스템 펌웨어 릴리스 노트를 참조하십시오.

이러한 서버의 시스템 펌웨어를 설치하고 업데이트하는 방법에 대한 자세한 내용은 지원되는 서버의 관리 설명서 또는 제품 안내서를 참조하십시오.

1. 시스템 제어기에 연결된 관리 포트(직렬 또는 네트워크 중 하나)에서 호스트 서버를 종료하고 전원을 끕니다.

```
# shutdown -i5 -g0 -y
```

2. flashupdate(1M) 명령을 사용하여 사용 중인 서버에 따라 시스템 펌웨어를 업그레이드합니다.

```
sc> flashupdate -s IP-address -f path/Sun_System_Firmware-  
x_x_x_build_nm-server-name.bin  
username: your-userid  
password: your-password
```

여기서,

- *IP-address*는 FTP 서버의 IP 주소입니다.
- *path*는 시스템 펌웨어 이미지를 구할 수 있는 고유 디렉토리 또는 SunSolveSM의 위치입니다.
- *x_x_x*는 시스템 펌웨어의 버전 번호입니다.
- *nn*은 이 릴리스에 적용되는 빌드 번호입니다.
- *server-name*은 사용 중인 서버 이름입니다. 예를 들어 Sun Fire T2000 서버의 *server-name*은 Sun_Fire_T2000입니다.

3. 시스템 제어기를 재설정합니다.

```
sc> resetsc -y
```

4. 호스트 서버의 전원을 켜고 부트합니다.

```
sc> poweron -c  
ok boot disk
```

▼ FTP 서버를 사용하지 않고 시스템 펌웨어를 업그레이드하려면

시스템 제어기로 펌웨어를 업로드하는 로컬 FTP 서버에 대한 액세스 권한이 없는 경우 `sysfwdownload` 유틸리티를 사용할 수 있으며, 이 유틸리티는 다음 SunSolve 사이트에서 시스템 펌웨어 업그레이드 패키지과 함께 제공됩니다.

<http://sunsolve.sun.com>

1. 다음 명령을 Solaris OS에서 실행합니다.

```
# cd firmware_location  
# sysfwdownload system_firmware_file
```

2. Solaris OS 인스턴스를 종료합니다.

```
# shutdown -i5 -g0 -y
```

3. 시스템 제어기에서 해당 펌웨어의 전원을 끄고 업데이트합니다.

```
sc> poweroff -fy  
sc> flashupdate -s 127.0.0.1
```


4. 시스템 제어를 재설정한 다음 전원을 켭니다.

```
sc> resetsc -y
sc> poweron
```

▼ 시스템 펌웨어를 다운그레이드하려면

Logical Domains 소프트웨어와 함께 사용하기 위해 시스템 펌웨어를 업그레이드한 후 해당 펌웨어를 원래 비Logical Domains 펌웨어로 다운그레이드할 수 있습니다.

- **flashupdate(1M)** 명령을 실행하고 원래 비Logical Domains 펌웨어에 대한 경로를 지정합니다.

Logical Domains Manager 및 Solaris Security Toolkit 다운로드

▼ Logical Domains Manager, Solaris Security Toolkit 및 Logical Domains MIB를 다운로드하려면

1. Sun 소프트웨어 다운로드 사이트에서 **Logical Domains Manager** 패키지(SUNWldm)가 포함된 **tar** 파일(LDoms_Manager-1_0_2.zip), **Solaris Security Toolkit**(SUNWjass)과 설치 스크립트(install-ldm) 및 **Logical Domains Management Information Base** 패키지(SUNWldmib.v)를 다운로드합니다. 다음 웹 사이트에서 해당 소프트웨어를 찾을 수 있습니다.

<http://www.sun.com/ldoms>

2. zip 파일의 압축을 풉니다.

```
$ unzip LDoms_Manager-1_0_2.zip
```

다운로드한 소프트웨어의 디렉토리 구조는 다음과 유사합니다.

코드 예 3-1 다운로드한 Logical Domains 1.0.2 소프트웨어의 디렉토리 구조

```
LDomains_Manager-1_0_2/  
  Install/  
    install-ldm  
  Legal/  
    Ldomains_1.0.2_Entitlement.txt  
    Ldomains_1.0.2_SLA_Entitlement.txt  
  Product/  
    SUNWjass/  
    SUNWldm.v/  
    SUNWldmib.v  
  README
```

Logical Domains Manager 및 Solaris Security Toolkit 설치

다음은 Logical Domains Manager 및 Solaris Security Toolkit 소프트웨어를 설치하는 세 가지 방법입니다.

- 설치 스크립트를 사용하여 패키지 및 패치를 설치합니다. 이를 통해 Logical Domains Manager 및 Solaris Security Toolkit 소프트웨어를 모두 자동으로 설치합니다. 21페이지의 "설치 스크립트를 사용하여 Logical Domains Manager 1.0.2 및 Solaris Security Toolkit 4.2 소프트웨어 설치"를 참조하십시오.
- JumpStart를 사용하여 패키지를 설치합니다. 27페이지의 "JumpStart를 사용하여 Logical Domains Manager 1.0.2 및 Solaris Security Toolkit 4.2 소프트웨어 설치"를 참조하십시오.
- 수동으로 각 패키지를 설치합니다. 30페이지의 "Logical Domains Manager 및 Solaris Security Toolkit 소프트웨어 수동 설치"를 참조하십시오.

주 - LDoms 및 Solaris Security Toolkit 패키지를 설치한 후 LDoms MIB 소프트웨어 패키지를 수동으로 설치해야 한다는 점에 주의하십시오. 다른 패키지와 함께 자동으로 설치되지 않습니다. LDoms MIB를 설치하고 사용하는 방법에 대한 자세한 내용은 Logical Domains(LDoms) Management Information Base 1.0.1 관리 설명서를 참조하십시오.

설치 스크립트를 사용하여 Logical Domains Manager 1.0.2 및 Solaris Security Toolkit 4.2 소프트웨어 설치

install-ldm 설치 스크립트를 사용하는 경우 여러 가지 선택으로 스크립트를 실행할 방법을 지정할 수 있습니다. 각 선택 사항에 대한 내용은 다음 절차에서 설명합니다.

- 옵션 없이 install-ldm 스크립트를 사용하여 다음을 자동으로 수행합니다.
 - Solaris OS 릴리스가 Solaris 10 11/06인지 확인
 - 패키지의 하위 디렉토리 SUNWldm/ 및 SUNWjass/가 있는지 확인
 - 필수 Solaris Logical Domains 드라이버 패키지인 SUNWldomr 및 SUNWldomu가 있는지 확인
 - SUNWldm 및 SUNWjass 패키지가 설치되어 있지 않은지 확인

주 - 해당 스크립트에서 설치 중 이전 버전의 SUNWjass를 감지하면 제거해야 합니다. 이전의 Solaris OS 강화를 실행 취소할 필요는 없습니다.

- Logical Domains Manager 1.0.2 소프트웨어(SUNWldm 패키지) 설치
- 필수 패키지가 포함된 Solaris Security Toolkit 4.2 소프트웨어(SUNWjass 패키지) 설치
- 모든 패키지가 설치되어 있는지 확인
- Logical Domains Manager 데몬 ldmd 활성화
- Solaris Security Toolkit ldmd_control-secure.driver 또는 선택한 -secure.driver로 끝나는 다른 드라이버 중 하나를 사용하여 컨트롤 도메인에서 Solaris OS 강화
- install-ldm 스크립트를 옵션 -d와 함께 사용하면 -secure.driver로 끝나는 드라이버를 제외한 Solaris Security Toolkit 드라이버를 지정할 수 있습니다. 이 옵션은 이전의 선택 사항에 나열된 모든 기능을 추가된 옵션으로 자동 수행합니다.
- 지정한 Solaris Security Toolkit 사용자 정의 드라이버를 사용하여 컨트롤 도메인에서 Solaris OS를 강화합니다(예:server-secure-myname.driver).
- install-ldm 스크립트를 옵션 -d와 함께 사용하고 none을 지정하면 Solaris Security Toolkit을 사용하여 컨트롤 도메인에서 실행 중인 Solaris OS를 강화하지 않도록 지정합니다. 이 옵션은 이전 선택 사항에 나열된 강화를 제외한 모든 기능을 자동으로 수행합니다. Solaris Security Toolkit 사용의 바이패스는 권장되지 않으며 대체 프로세스를 사용하여 컨트롤 도메인을 강화하려고 하는 경우에만 수행해야 합니다.
- install-ldm 스크립트를 옵션 -p와 함께 사용하면 Logical Domains Manager 데몬(ldmd)을 활성화하고 Solaris Security Toolkit을 실행하는 사후 설치 작업만 수행하려는 것을 지정합니다. 예를 들어 SUNWldm 및 SUNWjass 패키지가 사용 중인 서버에 사전 설치된 경우 이 옵션을 사용합니다. [27페이지의 "install-ldm 스크립트를 -p 옵션과 함께 사용하여 설치하려면"](#)을 참조하십시오.

▼ 옵션 없이 install-ldm 스크립트를 사용하여 설치하려면

- 옵션 없이 설치 스크립트를 실행합니다.

설치 스크립트는 SUNWldm 패키지의 일부로 Install 하위 디렉토리에 있습니다.

```
# Install/install-ldm
```

- a. 하나 이상의 패키지가 이전에 설치되어 있는 경우 다음 메시지를 받습니다.

```
# Install/install-ldm
ERROR: One or more packages are already installed: SUNWldm SUNWjass.
If packages SUNWldm.v and SUNWjass are factory pre-installed, run
install-ldm -p to perform post-install actions. Otherwise remove the
package(s) and restart install-ldm.
```

사후 설치 작업만 수행하려는 경우 27페이지의 "install-ldm 스크립트를 -p 옵션과 함께 사용하여 설치하려면"으로 이동합니다.

- b. 프로세스가 완료되면 다음 예와 유사한 메시지를 받습니다.

- 코드 예 3-2는 다음 기본 보안 프로필을 선택하는 경우 install-ldm 스크립트의 실행을 보여 줍니다.

a) Hardened Solaris configuration for LDoms (recommended)

- 코드 예 3-3은 다음 보안 프로필을 선택하는 경우 install-ldm 스크립트의 실행을 보여 줍니다.

c) Your custom-defined Solaris security configuration profile

선택을 위해 표시되는 드라이버는 -secure.driver로 끝나는 드라이버입니다. -secure.driver로 끝나지 않는 사용자 정의 드라이버를 기록하는 경우 install-ldm -d 옵션으로 사용자 정의 드라이버를 지정해야 합니다. 25페이지의 "install-ldm 스크립트를 -d 옵션과 함께 사용하여 설치하려면"을 참조하십시오.

코드 예 3-2

LDoms용으로 강화된 Solaris 구성 출력

```
# Install/install-ldm
Welcome to the LDoms installer.

You are about to install the domain manager package that will enable
you to create, destroy and control other domains on your system. Given
the capabilities of the domain manager, you can now change the security
configuration of this Solaris instance using the Solaris Security
Toolkit.
```

```

Select a security profile from this list:

a) Hardened Solaris configuration for LDoms (recommended)
b) Standard Solaris configuration
c) Your custom-defined Solaris security configuration profile

Enter a, b, or c [a]: a
The changes made by selecting this option can be undone through the
Solaris Security Toolkit's undo feature. This can be done with the
'/opt/SUNWjass/bin/jass-execute -u' command.

Installing LDoms and Solaris Security Toolkit packages.
pkgadd -n -d "/var/tmp/install/Product/Logical_Domain_Manager" -a pkg_admin
SUNWldm.v
Copyright 2006 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

Installation of <SUNWldm> was successful.
pkgadd -n -d "/var/tmp/install/Product/Solaris_Security_Toolkit" -a pkg_admin
SUNWjass
Copyright 2005 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

Installation of <SUNWjass> was successful.

Verifying that all packages are fully installed. OK.
Enabling services: svc:/ldoms/ldmd:default
Running Solaris Security Toolkit 4.2.0 driver ldm_control-secure.driver.
Please wait. . .
/opt/SUNWjass/bin/jass-execute -q -d ldm_control-secure.driver
Executing driver, ldm_control-secure.driver
Solaris Security Toolkit hardening executed successfully; log file
/var/opt/SUNWjass/run/20070208142843/jass-install-log.txt. It will not
take effect until the next reboot. Before rebooting, make sure SSH or
the serial line is setup for use after the reboot.

```

```

# Install/install-ldm
Welcome to the LDoms installer.

You are about to install the domain manager package that will enable
you to create, destroy and control other domains on your system. Given
the capabilities of the domain manager, you can now change the security
configuration of this Solaris instance using the Solaris Security
Toolkit.

```

```
Select a security profile from this list:

a) Hardened Solaris configuration for LDoms (recommended)
b) Standard Solaris configuration
c) Your custom-defined Solaris security configuration profile

Enter a, b, or c [a]: c
Choose a Solaris Security Toolkit .driver configuration profile from
this list
1) ldm_control-secure.driver
2) secure.driver
3) server-secure.driver
4) suncluster3x-secure.driver
5) sunfire_15k_sc-secure.driver

Enter a number 1 to 5: 2
The driver you selected may not perform all the LDoms-specific
operations specified in the LDoms Administration Guide.
Is this OK (yes/no)? [no] y
The changes made by selecting this option can be undone through the
Solaris Security Toolkit's undo feature. This can be done with the
'/opt/SUNWjass/bin/jass-execute -u' command.

Installing LDoms and Solaris Security Toolkit packages.
pkgadd -n -d "/var/tmp/install/Product/Logical_Domain_Manager" -a pkg_admin
SUNWldm.v
Copyright 2006 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

Installation of <SUNWldm> was successful.
pkgadd -n -d "/var/tmp/install/Product/Solaris_Security_Toolkit" -a pkg_admin
SUNWjass
Copyright 2005 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

Installation of <SUNWjass> was successful.

Verifying that all packages are fully installed. OK.
Enabling services: svc:/ldoms/ldmd:default
Running Solaris Security Toolkit 4.2.0 driver secure.driver.
Please wait. . .
/opt/SUNWjass/bin/jass-execute -q -d secure.driver
Executing driver, secure.driver
Solaris Security Toolkit hardening executed successfully; log file
/var/opt/SUNWjass/run/20070102142843/jass-install-log.txt. It will not
take effect until the next reboot. Before rebooting, make sure SSH or
the serial line is setup for use after the reboot.
```

▼ **install-ldm** 스크립트를 **-d** 옵션과 함께 사용하여 설치하려면

- 설치 스크립트를 **-d** 옵션과 함께 사용하여 실행하면 **Solaris Security Toolkit** 사용자 정의 강화 드라이버를 지정할 수 있습니다(예:server-secure-myname.driver).

설치 스크립트는 SUNWldm 패키지의 일부로 Install 하위 디렉토리에 있습니다.

```
# Install/install-ldm -d server-secure-myname.driver
```

프로세스가 완료되면 다음 코드 예 3-4에 있는 메시지와 유사한 메시지를 받습니다.

코드 예 3-4 install-ldm -d 스크립트의 실행 출력

```
# Install/install-ldm -d server-secure.driver
The driver you selected may not perform all the LDoms-specific
operations specified in the LDoms Administration Guide.
Installing LDoms and Solaris Security Toolkit packages.
pkgadd -n -d "/var/tmp/install/Product/Logical_Domain_Manager" -a pkg_admin
SUNWldm.v
Copyright 2006 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

Installation of <SUNWldm> was successful.
pkgadd -n -d "/var/tmp/install/Product/Solaris_Security_Toolkit" -a pkg_admin
SUNWjass
Copyright 2005 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

Installation of <SUNWjass> was successful.

Verifying that all packages are fully installed. OK.
Enabling services: svc:/ldoms/ldmd:default
Running Solaris Security Toolkit 4.2.0 driver server-secure-myname.driver.
Please wait. . .
/opt/SUNWjass/bin/jass-execute -q -d server-secure-myname.driver
Executing driver, server-secure-myname.driver
Solaris Security Toolkit hardening executed successfully; log file
/var/opt/SUNWjass/run/20061114143128/jass-install-log.txt. It will not
take effect until the next reboot. Before rebooting, make sure SSH or
the serial line is setup for use after the reboot.
```

▼ **install-ldm** 스크립트를 `-d none` 옵션과 함께 사용하여 설치하려면

- 설치 스크립트를 `-d none` 옵션과 함께 사용하여 실행하면 **Solaris Security Toolkit** 드라이버를 사용하는 시스템을 강화하지 않도록 지정합니다.

설치 스크립트는 `SUNWldm` 패키지의 일부로 `Install` 하위 디렉토리에 있습니다.

```
# Install/install-ldm -d none
```

프로세스가 완료되면 다음 코드 예 3-5에서 표시되는 예와 유사한 메시지를 받습니다.

코드 예 3-5 `install-ldm -d none` 스크립트의 실행 출력

```
# Install/install-ldm -d none
Installing LDoms and Solaris Security Toolkit packages.
pkgadd -n -d "/var/tmp/install/Product/Logical_Domain_Manager" -a pkg_admin
SUNWldm.v
Copyright 2006 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

Installation of <SUNWldm> was successful.
pkgadd -n -d "/var/tmp/install/Product/Solaris_Security_Toolkit" -a pkg_admin
SUNWjass
Copyright 2005 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

Installation of <SUNWjass> was successful.

Verifying that all packages are fully installed. OK.
Enabling services: svc:/ldoms/ldmd:default
Solaris Security Toolkit was not applied. Bypassing the use of the
Solaris Security Toolkit is not recommended and should only be
performed when alternative hardening steps are to be taken.
```


▼ install-ldm 스크립트를 -p 옵션과 함께 사용하여 설치하려면

SUNWldm 및 SUNWjass 패키지가 사용 중인 서버에 사전 설치되어 있고 Logical Domains Manager 데몬(ldmd)을 활성화하고 Solaris Security Toolkit을 실행하는 사후 설치 작업만 수행하려는 경우 이 옵션을 사용할 수 있습니다.

- 설치 스크립트를 -p 옵션과 함께 사용하여 실행하면 ldmd를 활성화하고 시스템 강화를 위해 **Solaris Security Toolkit**을 실행하는 사후 설치 작업만 수행합니다.

```
# Install/install-ldm -p
Verifying that all packages are fully installed. OK.
Enabling services: svc:/ldoms/ldmd:default
Running Solaris Security Toolkit 4.2.0 driver ldm_control-secure.driver.
Please wait. . .
/opt/SUNWjass/bin/jass-execute -q -d ldm_control-secure.driver
Solaris Security Toolkit hardening executed successfully; log file
var/opt/SUNWjass/run/20070515140944/jass-install-log.txt. It will not
take effect until the next reboot. Before rebooting, make sure SSH or
the serial line is setup for use after the reboot.
```

JumpStart를 사용하여 Logical Domains Manager 1.0.2 및 Solaris Security Toolkit 4.2 소프트웨어 설치

JumpStart 사용에 대한 자세한 내용은 JumpStart Technology: Effective Use in the Solaris Operating Environment를 참조하십시오.



주의 - 네트워크 설치 중 가상 콘솔의 연결을 해제하면 안 됩니다.

▼ JumpStart 서버를 설정하려면

- 이미 JumpStart 서버를 설정한 경우 이 관리 설명서의 28페이지의 "**JumpStart 소프트웨어를 사용하여 설치하려면**"으로 이동합니다.
- JumpStart 서버를 이미 설정하지 않은 경우 다음을 수행해야 합니다.
이 절차에 대한 자세한 내용은 Solaris 10 11/06 설치 설명서: 사용자 정의 JumpStart 및 고급 설치를 참조하십시오. 이 설치 설명서는 다음 사이트에서 찾을 수 있습니다.
<http://docs.sun.com/app/docs/doc/819-6397>

1. Solaris 10 11/06 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 3장 "사용자 정의 JumpStart 설치 준비(작업)"를 참조하여 다음 단계를 수행합니다.
 - a. "작업 맵: 사용자 정의 JumpStart 설치 준비"에서 작업 맵을 읽습니다.
 - b. "네트워크 시스템에 대한 프로필 서버 만들기"의 절차대로 네트워크에 연결된 시스템을 설정합니다.
 - c. "rules 파일 만들기" 절차대로 rules 파일을 만듭니다.
2. "rules 파일 검증" 절차대로 rules 파일을 검증합니다.
Solaris Security Toolkit은 프로필과 완료 스크립트를 제공합니다. 프로필과 완료 스크립트에 대한 자세한 내용은 Solaris Security Toolkit 4.2 Reference Manual을 참조하십시오.

▼ JumpStart 소프트웨어를 사용하여 설치하려면

1. Solaris Security Toolkit 패키지(SUNWjass)를 다운로드한 디렉토리로 변경합니다.

```
# cd /path-to-download
```

2. SUNWjass를 설치하여 JumpStart(jumpstart) 디렉토리 구조를 만듭니다.

```
# pkgadd -R /jumpstart -d . SUNWjass
```

3. 텍스트 편집기를 사용하여
/jumpstart/opt/SUNWjass/Sysidcfg/Solaris_10/sysidcfg 파일을 수정하고 네트워크 환경을 반영합니다.
4. /jumpstart/opt/SUNWjass/Drivers/user.init.SAMPLE 파일을
/jumpstart/opt/SUNWjass/Drivers/user.init 파일로 복사합니다.

```
# cp user.init.SAMPLE user.init
```

5. user.init 파일을 편집하여 경로를 반영합니다.
6. JumpStart 설치 중 Solaris Security Toolkit 패키지(SUNWjass)를 대상 시스템에서 설치하려면 해당 패키지를 user.init 파일에서 정의한 JASS_PACKAGE_MOUNT 디렉토리에 배치해야 합니다. 예를 들면 다음과 같습니다.

```
# cp -r /path/to/LDoms_Manager-1_0_2/Product/SUNWjass  
/jumpstart/opt/SUNWjass/Packages
```

7. **JumpStart** 설치 중 **Logical Domains Manager** 패키지(SUNWldm.v)를 대상 시스템에서 설치하려면 다운로드 영역에 있는 해당 패키지를 `user.init` 파일에서 정의한 `JASS_PACKAGE_MOUNT` 디렉토리에 배치해야 합니다. 예를 들면 다음과 같습니다.

```
# cp -r /path/to/LDoms_Manager-1_0_2/Product/SUNWldm.v
/jumpstart/opt/SUNWjass/Packages
```

8. 멀티홈 **JumpStart** 서버에서 문제가 발생하면 `user.init` 파일에 있는 `JASS_PACKAGE_MOUNT` 및 `JASS_PATCH_MOUNT`에 대한 두 개의 항목을 `JASS_HOME_DIR/Patches` 및 `JASS_HOME_DIR/Packages` 디렉토리의 올바른 경로로 수정합니다. 자세한 내용은 `user.init.SAMPLE` 파일에 있는 주석을 참조하십시오.
9. **Logical Domains Manager** 컨트롤 도메인의 기본 드라이버로 `ldm_control-secure.driver`를 사용합니다.
사용을 위해 드라이버를 수정하는 방법에 대한 자세한 내용은 **Solaris Security Toolkit 4.2 Reference Manual**의 4장을 참조하십시오. `ldm_control-secure.driver`의 대응 항목인 **Solaris Security Toolkit**의 주 드라이버는 `secure.driver`입니다.
10. `ldm_control-secure.driver`의 수정을 완료한 후 **rules** 파일의 항목을 수정합니다.
- **LDoms** 컨트롤 도메인을 최소화하려는 경우 다음과 유사하게 **rules** 파일에서 `minimal-ldm-control.profile`을 지정합니다.

```
hostname imbulu - Profiles/minimal-ldm_control.profile Drivers/ldm_control-secure-abc.driver
```

주 – **LDoms** 및 **Solaris Security Toolkit** 패키지를 설치한 후 **LDoms MIB** 소프트웨어 패키지를 수동으로 설치해야 한다는 점에 주의하십시오. 다른 패키지와 함께 자동으로 설치되지 않습니다. **LDoms MIB**를 설치하고 사용하는 방법에 대한 자세한 내용은 **Logical Domains(LDoms) Management Information Base 1.0.1** 관리 설명서를 참조하십시오.

- **LDoms** 컨트롤 도메인을 최소화하지 않으려면 항목은 다음과 유사하게 나타나야 합니다.

```
hostname imbulu - Profiles/oem.profile Drivers/ldm_control-secure-abc.driver
```

11. **JumpStart** 설치 중 강화를 실행 취소하려면 다음 **SMF** 명령을 실행하여 **Logical Domains Manager**를 다시 시작해야 합니다.

```
# svcadm enable svc:/ldoms/ldmd:default
```

Logical Domains Manager 및 Solaris Security Toolkit 소프트웨어 수동 설치

다음 절차를 수행하여 Logical Domains Manager 및 Solaris Security Toolkit 소프트웨어를 수동으로 설치합니다.

- 30페이지의 "[Logical Domains Manager\(LDoms\) 1.0.2 소프트웨어를 수동으로 설치하려면](#)"
- 30페이지의 "(선택 사항) [Solaris Security Toolkit 4.2 소프트웨어를 수동으로 설치하려면](#)"
- 31페이지의 "(선택 사항) [컨트롤 도메인을 수동으로 강화하려면](#)"

▼ Logical Domains Manager(LDoms) 1.0.2 소프트웨어를 수동으로 설치하려면

SUN 소프트웨어 다운로드 사이트에서 Logical Domains Manager 1.0.2 소프트웨어인 SUNWldm 패키지를 다운로드합니다. 자세한 내용은 [19페이지의 "Logical Domains Manager, Solaris Security Toolkit 및 Logical Domains MIB를 다운로드하려면"](#)을 참조하십시오.

1. pkgadd(1M) 명령을 사용하여 SUNWldm.v 패키지를 설치합니다. -G 옵션을 사용하여 전역 영역에만 패키지를 설치하고 -d 옵션을 사용하여 SUNWldm.v 패키지가 포함된 디렉토리에 대한 경로를 지정합니다.

```
# pkgadd -Gd . SUNWldm.v
```

2. 대화형 프롬프트에서 모든 질문에 예에 해당하는 대답으로 y를 입력합니다.
3. pkginfo(1) 명령을 사용하여 Logical Domains Manager 1.0.2 소프트웨어용 SUNWldm 패키지를 설치했는지 확인합니다.
아래의 예에 개정판(REV) 정보가 표시됩니다.

```
# pkginfo -l SUNWldm | grep VERSION  
VERSION=1.0.2,REV=2007.08.23.10.20
```

▼ (선택 사항) Solaris Security Toolkit 4.2 소프트웨어를 수동으로 설치하려면

시스템을 보호하려는 경우 SUNWjass 패키지를 다운로드하고 설치합니다. 필수 패치(122608-03 및 125672-01)는 SUNWjass 패키지에 포함되어 있습니다. 소프트웨어를 다운로드하는 방법에 대한 자세한 내용은 [19페이지의 "Logical Domains Manager, Solaris Security Toolkit 및 Logical Domains MIB를 다운로드하려면"](#)을 참조하십시오.

Logical Domains Manager 소프트웨어를 사용 중인 경우 보안 고려 사항에 대한 자세한 내용은 2장을 참조하십시오. 자세한 내용은 다음 사이트에 있는 Solaris Security Toolkit 4.2 설명서를 참조하십시오.

<http://docs.sun.com>

1. pkgadd(1M) 명령을 사용하여 SUNWjass 패키지를 설치합니다.

```
# pkgadd -d . SUNWjass
```

2. pkginfo(1) 명령을 사용하여 Solaris Security Toolkit 4.2 소프트웨어용 SUNWjass 패키지를 설치했는지 확인합니다.

```
# pkginfo -l SUNWjass | grep VERSION  
VERSION: 4.2.0
```

▼ (선택 사항) 컨트롤 도메인을 수동으로 강화하려면

Solaris Security Toolkit 4.2 패키지를 설치한 경우에만 이 절차를 수행합니다.

주 – Solaris Security Toolkit을 사용하여 컨트롤 도메인을 강화한 경우 많은 시스템 서비스를 비활성화하고 네트워크 액세스에 대한 특정 제한 사항을 설정합니다. 자세한 내용이 있는 Solaris Security Toolkit 4.2 설명서를 찾으려면 이 설명서의 [xx페이지의 "관련 문서"](#)를 참조하십시오.

1. ldm_control-secure.driver를 사용하여 강화합니다.

```
# /opt/SUNWjass/bin/jass-execute -d ldm_control-secure.driver
```

다른 드라이버를 사용하여 시스템을 강화할 수 있습니다. 드라이버를 사용자 정의하여 환경 보안을 조정할 수도 있습니다. 드라이버와 드라이버의 사용자 정의에 대한 자세한 내용은 Solaris Security Toolkit 4.2 Reference Manual을 참조하십시오.

2. 대화형 프롬프트에서 모든 질문에 예에 해당하는 대답으로 y를 입력합니다.
3. 강화를 적용하기 위해 해당 서버를 종료하고 재부트합니다.

```
# /usr/sbin/shutdown -y -g0 -i6
```

▼ 강화를 검증하려면

- **Logical Domains** 강화 드라이버(`ldom_control-secure.driver`)에서 강화가 제대로 적용되었는지 확인합니다.
다른 드라이버에서 확인하려면 다음 명령 예에서 드라이버 이름을 대체합니다.

```
# /opt/SUNWjass/bin/jass-execute -a ldom_control-secure.driver
```

▼ 강화를 실행 취소하려면

1. **Solaris Security Toolkit**에서 적용된 구성 변경 사항을 실행 취소합니다.

```
# /opt/SUNWjass/bin/jass-execute -u
```

Solaris Security Toolkit에서 실행 취소하려는 강화 실행을 묻습니다.

2. 실행 취소하려는 강화 실행을 선택합니다.
3. 강화가 취소된 구성이 적용되도록 시스템을 재부트합니다.

```
# /usr/sbin/shutdown -y -g0 -i6
```

주 - JumpStart 설치 중 수행된 강화를 실행 취소하려면 다음 SMF 명령을 실행하여 Logical Domains Manager와 가상 네트워크 터미널 서버 데몬을 다시 시작해야 합니다.

```
# svcadm enable svc:/ldoms/ldmd:default
```

Logical Domains Manager 데몬 활성화

설치 스크립트 `install-ldm`은 자동으로 Logical Domains Manager 데몬(`ldmd`)을 활성화합니다. Logical Domains Manager 소프트웨어를 수동으로 설치한 경우 Logical Domains Manager 데몬인 `ldmd`를 활성화해야 논리적 도메인을 만들고, 수정하고, 제어할 수 있습니다.

▼ Logical Domains Manager 데몬을 활성화하려면

1. `svcadm(1M)` 명령을 사용하여 **Logical Domains Manager** 데몬인 `ldmd`를 활성화합니다.

```
# svcadm enable ldmd
```

2. `ldm list` 명령을 사용하여 **Logical Domains Manager**가 실행 중인지 확인합니다.
`factory-default` 구성에 대해 다음과 유사한 메시지를 받습니다. `primary` 도메인이 `active`인 경우에는 **Logical Domains Manager**가 실행 중임을 의미합니다.

```
# /opt/SUNWldm/bin/ldm list
NAME                STATE   FLAGS   CONS   VCPU  MEMORY  UTIL  UPTIME
primary            active  ---c-  SP     32    3264M   0.3%  19d 9m
```

인증 및 프로파일 만들기와 사용자 계정에 역할 할당

Logical Domains Manager에 대해 조정된 Solaris OS 역할별 접근 제어(Role-Based Access Control, RBAC)를 사용하여 인증 및 프로파일을 설정하고 사용자 계정에 역할을 할당합니다. RBAC에 대한 자세한 내용은 Solaris 10 시스템 관리자 모음을 참조하십시오.

Logical Domains Manager에 대한 인증에는 다음 두 가지 수준이 있습니다.

- 읽기 - 구성을 볼 수 있으나 수정할 수 없습니다.
- 읽기 및 쓰기 - 구성을 보고 변경할 수 있습니다.

다음은 Solaris OS `/etc/security/auth_attr` 파일에 자동으로 추가되는 Logical Domains 항목입니다.

- `solaris.ldoms.:::LDom administration::`
- `solaris.ldoms.grant.:::Delegate LDom configuration::`
- `solaris.ldoms.read.:::View LDom configuration::`
- `solaris.ldoms.write.:::Manage LDom configuration::`

사용자 인증 관리

▼ 사용자 인증을 추가하려면

필요한 경우 다음 단계를 사용하여 Logical Domains Manager 사용자에게 대한 인증을 /etc/security/auth_attr 파일에 추가합니다. 슈퍼유저는 이미 solaris.* 인증이 있으므로 solaris.ldoms.* 인증에 대한 권한이 있습니다.

1. **ldm(1M)** 하위 명령을 사용할 수 있는 인증이 필요한 각 사용자에게 대해 로컬 사용자 계정을 만듭니다.

주 - 사용자의 Logical Domains Manager 인증을 추가하려면 로컬(비LDAP) 계정을 해당 사용자용으로 만들어야 합니다. 자세한 내용은 Solaris 10 시스템 관리자 모음을 참조하십시오.

2. 사용자에게 액세스할 수 있게 하려는 **ldm(1M)** 하위 명령에 따라 다음 중 하나를 수행합니다.

ldm(1M) 명령 목록과 해당 사용자 인증은 [표 2-1](#)을 참조하십시오.

- **usermod(1M)** 명령을 사용하여 사용자의 읽기 전용 인증을 추가합니다.

```
# usermod -A solaris.ldoms.read username
```

- **usermod(1M)** 명령을 사용하여 사용자의 읽기 및 쓰기 인증을 추가합니다.

```
# usermod -A solaris.ldoms.write username
```

▼ 사용자의 모든 인증을 삭제하려면

- 로컬 사용자 계정에 대한 모든 인증을 삭제합니다(가능한 옵션만).

```
# usermod -A `` username
```


사용자 프로필 관리

SUNwldm 패키지는 슈퍼유저가 아닌 사용자로 Logical Domains Manager에 대한 액세스 권한을 인증하는 경우 사용할 /etc/security/prof_attr 파일에 2개의 시스템 정의 RBAC 프로필을 추가합니다. 2개의 LDoms 특정 프로필은 다음과 같습니다.

- LDoms Review:::Review LDoms configuration:auths=solaris.ldoms.read
- LDoms Management:::Manage LDoms domains:auths=solaris.ldoms.*

다음 절차를 사용하여 앞의 프로필 중 하나를 사용자 계정에 할당할 수 있습니다.

▼ 사용자의 프로필을 추가하려면

- 로컬 사용자 계정에 대한 관리 프로필(예: **LDoms Management**)을 추가합니다.

```
# usermod -P "LDoms Management" username
```

▼ 사용자의 모든 프로필을 삭제하려면

- 로컬 사용자 계정에 대한 모든 프로필을 삭제합니다(가능한 옵션만).

```
# usermod -P `` username
```

사용자에 역할 할당

이 절차 사용의 장점은 특정 역할이 할당된 사용자만 해당 역할을 맡을 수 있다는 점입니다. 역할을 맡을 때 역할에 암호가 지정되면 암호가 필요합니다. 2가지 보안 계층을 제공합니다. 사용자에게 역할이 할당되지 않으면 사용자는 정확한 암호를 알고 있더라도 su *role_name* 명령을 수행하여 해당 역할을 맡을 수 없습니다.

▼ 역할을 만들어 사용자에게 해당 역할을 할당하려면

1. 역할을 만듭니다.

```
# roleadd -A solaris.ldoms.read ldm_read
```

2. 역할에 암호를 할당합니다.

```
# passwd ldm_read
```

3. 사용자에게 역할을 할당합니다(예: user_1).

```
# useradd -R ldm_read user_1
```

4. 사용자에게 암호를 할당합니다(user_1).

```
# passwd user_1
```

5. ldm_read 계정이 될 user_1 계정에만 액세스 권한을 할당합니다.

```
# su user_1
```

6. 프롬프트가 나타나면 사용자 암호를 입력합니다.
7. 사용자 ID와 ldm_read 역할에 대한 액세스 권한을 확인합니다.

```
$ id
uid=nn(user_1) gid=nn(<group name>)
$ roles
ldm_read
```

8. 읽기 인증이 있는 ldm 하위 명령에 사용자에게 대한 액세스를 제공합니다.

```
# su ldm_read
```

9. 프롬프트가 나타나면 사용자 암호를 입력합니다.
10. id 명령을 입력하여 사용자를 표시합니다.

```
$ id
uid=nn(ldm_read) gid=nn(<group name>)
```

서비스 및 논리적 도메인 설정

이 장은 기본 서비스, 컨트롤 도메인 및 게스트 도메인을 설정하는 데 필요한 절차를 설명합니다.

출력 메시지

사용자 플랫폼에 따라 기본 서비스를 만들고 컨트롤(primary) 도메인을 설정하는 데 사용하는 명령을 통해 서로 다른 출력 메시지를 받습니다.

- Sun UltraSPARC T1 프로세서
- Sun UltraSPARC T2 프로세서

Sun UltraSPARC T1 프로세서

Sun UltraSPARC T1 프로세서와 함께 서버를 사용 중인 경우 primary 도메인에 대한 설치 명령 후 다음 알림을 받습니다.

```
Notice: the LDom Manager is running in configuration mode. Any
configuration changes made will only take effect after the machine
configuration is downloaded to the system controller and the host
is reset.
```

Sun UltraSPARC T2 프로세서

첫 번째 작업 - Sun UltraSPARC T2 프로세서와 함께 서버를 사용 중인 경우 primary 도메인의 장치 또는 서비스에서 첫 번째 작업 이후 다음 메시지를 받습니다.

```
Initiating delayed reconfigure operation on LDom primary. All configuration changes for other LDomS are disabled until the LDom reboots, at which time the new configuration for LDom primary will also take effect.
```

재부트 전까지의 후속 작업 - Sun UltraSPARC T2 프로세서와 함께 서버를 사용 중인 경우 재부트할 때까지 primary 도메인에 대한 모든 후속 작업 후 다음과 같은 알림을 받습니다.

```
Notice: LDom primary is in the process of a delayed reconfiguration. Any changes made to this LDom will only take effect after it reboots.
```

기본 서비스 만들기

처음에 다음과 같은 가상 기본 서비스를 만들어야 나중에 사용할 수 있습니다.

- vdiskserver - 가상 디스크 서버
- vswitch - 가상 스위치 서비스
- vconscon - 가상 콘솔 집중 장치 서비스

▼ 기본 서비스를 만들려면

1. 가상 디스크 서버(vds)를 만들어 논리적 도메인으로 가상 디스크 가져오기를 허용합니다.

예를 들어 다음 명령은 컨트롤 도메인(primary)에 가상 디스크 서버(primary-vds0)를 추가합니다.

```
primary$ ldm add-vds primary-vds0 primary
```

- 가상 콘솔 집중 장치 서비스(vcc)를 가상 네트워크 터미널 서버 데몬(vntsd)에서 사용할 수 있게 만들거나 모든 논리적 도메인 콘솔에 대한 집중 장치로 만듭니다. 예를 들어 다음 명령은 컨트롤 도메인(primary)에 포트 범위 5000~5100에서 가상 콘솔 집중 장치 서비스(primary-vcc0)를 추가합니다.

```
primary$ ldm add-vcc port-range=5000-5100 primary-vcc0 primary
```

- 가상 스위치 서비스(vsw)를 만들어 논리적 도메인의 가상 네트워크(vnet) 장치 간의 네트워킹을 활성화합니다. 각 논리적 도메인에서 가상 스위치를 통해 상자 외부로 통신해야 하는 경우 GLDv3 호환 네트워크 어댑터를 가상 스위치에 할당합니다. 예를 들어 다음 명령은 컨트롤 도메인(primary)에 네트워크 어댑터 드라이버(e1000g0)에 있는 가상 스위치 서비스(primary-vsw0)를 추가합니다.

```
primary$ ldm add-vsw net-dev=e1000g0 primary-vsw0 primary
```

이 명령은 MAC 주소를 가상 스위치에 자동으로 할당합니다. ldm add-vsw 명령에 대한 옵션으로 사용자 고유의 MAC 주소를 지정할 수 있습니다. 하지만 이 경우 사용자는 지정된 MAC 주소가 이미 존재하는 MAC 주소와 충돌하지 않는지 확인해야 합니다.

추가할 가상 스위치가 기본 물리적 어댑터를 기본 네트워크 인터페이스로 교체하는 경우 물리적 어댑터의 MAC 주소를 할당해야 동적 호스트 구성 프로토콜(Dynamic Host Configuration Protocol, DHCP) 서버에서 같은 IP 주소를 도메인에 할당합니다. [42페이지의 "컨트롤/서비스 도메인과 다른 도메인 간의 네트워킹 활성화"](#)를 참조하십시오.

```
primary$ ldm add-vsw mac-addr=2:04:4f:fb:9f:0d net-dev=e1000g0 primary-vsw0 primary
```

- list-services 하위 명령을 사용하여 만든 서비스를 확인합니다. 출력은 다음과 유사하게 나타나야 합니다.

```
primary$ ldm list-services primary
VDS
  NAME                VOLUME                OPTIONS                DEVICE
  primary-vds0
VCC
  NAME                PORT-RANGE
  primary-vcc0        5000-5100
VSW
  NAME                MAC                    NET-DEV                DEVICE                MODE
  primary-vsw0        02:04:4f:fb:9f:0d    e1000g0                switch@0                prog,promisc
```

컨트롤 도메인의 초기 구성

처음에는 모든 시스템 자원이 컨트롤 도메인에 할당됩니다. 다른 논리적 도메인을 만들 수 있으려면 이러한 자원 중 일부를 릴리스해야 합니다.

주 - 다음 예의 출력에서 LDoms Manager가 구성 모드에서 실행 중임을 나타내는 알림은 Sun UltraSPARC T1 프로세서에만 적용됩니다.

▼ 컨트롤 도메인을 설정하려면

주 - 이 절차에는 컨트롤 도메인에 설정할 자원의 예가 포함되어 있습니다. 여기에 나오는 숫자는 예일 뿐이며, 사용되는 값은 사용자 컨트롤 도메인에서는 적절하지 않을 수도 있습니다.

1. 컨트롤 도메인에 암호화 자원을 할당합니다.

주 - 컨트롤 도메인에 암호화 장치가 있는 경우 CPU를 동적으로 재구성할 수 없습니다. 따라서 암호화 장치를 사용하고 있지 않은 경우에는 `set-mau`를 0으로 설정합니다.

다음 예에서는 컨트롤 도메인 `primary`에 암호화 자원을 한 개 할당합니다. 이 경우 사용할 수 있는 나머지 암호화 자원은 게스트 도메인에서 유지됩니다.

```
primary$ ldm set-mau 1 primary
```

2. 컨트롤 도메인에 가상 CPU를 할당합니다.

예를 들어 다음 명령은 컨트롤 도메인 `primary`에 가상 CPU 4개를 할당합니다. 이 경우 사용할 수 있는 나머지 가상 CPU는 게스트 도메인에서 유지됩니다.

```
primary$ ldm set-vcpu 4 primary
```

3. 컨트롤 도메인에 메모리를 할당합니다.

예를 들어 다음 명령은 컨트롤 도메인 `primary`에 1GB의 메모리를 할당합니다. 이 경우 사용할 수 있는 나머지 메모리는 게스트 도메인에서 유지됩니다.

```
primary$ ldm set-memory 1G primary
```

주 - 디스크 서비스를 전달하는 데 ZFS를 사용하고 있지 않은 경우에는 1GB의 메모리가 적절합니다. 디스크 서비스를 전달하는 데 ZFS를 사용 중인 경우에는 4개의 가상 CPU와 4GB 이상의 메모리가 있는 전체 코어를 할당합니다. I/O 로드가 더 과도한 경우에는 전체 코어를 추가로 할당해야 할 수 있습니다.

4. 시스템 제어기(System Controller, SC)에 논리적 도메인 시스템 구성을 추가합니다. 예를 들어 다음 명령은 `initial`이라고 하는 구성을 추가합니다.

```
primary$ ldm add-config initial
```

주 - 현재 `factory-default` 구성을 제외하고 SC에 저장할 수 있는 구성은 8개로 제한됩니다.

5. 다음 재부트 시 구성을 사용할 준비가 되었는지 확인합니다.

```
primary$ ldm list-config
factory-default [current]
initial [next]
```

이 하위 명령 목록은 `factory-default` 구성 세트가 현재 사용 중이고, `initial` 구성 세트는 재부트 시 사용됨을 보여 줍니다.

논리적 도메인 사용을 위한 재부트

구성 변경 사항을 적용하기 위해 제어/서비스 도메인을 재부트하고, 다른 논리적 도메인에서 사용할 수 있도록 릴리스할 자원을 재부트해야 합니다.

▼ 논리적 도메인 사용을 위해 재부트하려면

- `primary` 도메인을 종료하고 재부트합니다. 이 예의 경우 이 도메인은 서비스 도메인이기도 합니다.

```
primary# shutdown -y -g0 -i6
```

주 - 지정된 명령을 사용한 후 재부트하면 변경된 사항이 적용될 수 있지만 `ldm list-config` 명령은 여전히 재부트 전과 같은 출력을 보여 줍니다. 전원을 끄고 켜는 작업에는 표시된 구성을 업데이트하기 위해 `ldm list-config` 명령이 필요합니다.

컨트롤/서비스 도메인과 다른 도메인 간의 네트워킹 활성화

기본적으로 시스템의 컨트롤/서비스 도메인과 다른 도메인 간의 네트워킹이 비활성화되어 있습니다. 이를 활성화하려면 가상 스위치 장치를 네트워크 장치로 구성해야 합니다. 가상 스위치는 기본 물리적 장치(이 예의 경우 e1000g0)를 기본 인터페이스로 교체하거나 도메인의 추가 네트워크 인터페이스로 구성될 수 있습니다.

주 - 다음 구성 단계를 도메인 콘솔에서 수행합니다. 이 절차로 인해 도메인의 네트워크 연결이 임시로 끊길 수 있습니다.

▼ 가상 스위치를 기본 인터페이스로 구성하려면

1. 모든 인터페이스에 대한 주소 정보를 인쇄합니다.

```
primary# ifconfig -a
```

2. 가상 스위치를 연결(plumb)합니다. 이 예에서 vsw0는 구성할 가상 스위치입니다.

```
primary# ifconfig vsw0 plumb
```

3. (선택 사항)도메인에 있는 모든 가상 스위치 인스턴스 목록을 얻기 위해 이들을 나열할 수 있습니다.

```
primary# /usr/sbin/dladm show-link | grep vsw
vsw0                type: non-vlan  mtu: 1500      device: vsw0
```

4. 가상 스위치(net-dev)에 할당된 물리적 네트워크 장치(이 예에서는 e1000g0)의 연결을 해제(unplumb)합니다.

```
primary# ifconfig e1000g0 down unplumb
```


5. 물리적 네트워크 장치(e1000g0)의 등록 정보를 가상 스위치(vsw0) 장치로 이전하려면 다음 중 하나를 수행합니다.
 - 네트워킹이 정적 IP 주소를 사용하여 구성되어 있는 경우 vsw0에 대한 e1000g0의 IP 주소와 넷마스크를 다시 사용합니다.

```
primary# ifconfig vsw0 IP_of_e1000g0 netmask netmask_of_e1000g0 broadcast + up
```

- 네트워킹이 DHCP를 사용하여 구성된 경우 vsw0에 대한 DHCP를 활성화합니다.

```
primary# ifconfig vsw0 dhcp start
```

6. 필수 구성 파일을 수정하여 이 변경 사항을 영구적으로 만듭니다.

```
primary# mv /etc/hostname.e1000g0 /etc/hostname.vsw0
primary# mv /etc/dhcp.e1000g0 /etc/dhcp.vsw0
```

주 - 필요한 경우 물리적 네트워크 장치 뿐 아니라 가상 스위치도 구성할 수 있습니다. 이 경우 2단계와 같이 가상 스위치를 연결(plumb)합니다. 물리적 장치의 연결을 해제(unplumb)하면 안 됩니다(4단계 건너뛴). 가상 스위치는 정적 IP 주소로 구성되거나 DHCP 서버에서 동적 IP 주소를 얻어야 합니다.

가상 네트워크 터미널 서버 데몬 활성화

가상 네트워크 터미널 서버 데몬(vntsd)을 활성화하여 각 논리적 도메인의 가상 콘솔에 대한 액세스를 제공해야 합니다. 이 데몬을 사용하는 방법에 대한 자세한 내용은 Solaris 10 OS 참조 설명서 모음 또는 vntsd(1M) 매뉴얼 페이지를 참조하십시오.

▼ 가상 네트워크 터미널 서버 데몬을 활성화하려면

주 - vntsd를 활성화하기 전에 컨트롤 도메인에서 기본 서비스 vconscn을 만들었는지 확인하십시오. 자세한 내용은 [38페이지의 "기본 서비스 만들기"](#)를 참조하십시오.

1. svcadm(1M) 명령을 사용하여 가상 네트워크 터미널 서버 데몬인 vntsd(1M)를 활성화합니다.

```
# svcadm enable vntsd
```

2. `svcs(1)` 명령을 사용하여 `vntsd`가 활성화되었는지 확인합니다.

```
# svcs -l vntsd
fmri          svc:/ldoms/vntsd:default
enabled      true
state        online
next_state    none
state_time    Sat Jan 27 03:14:17 2007
logfile       /var/svc/log/ldoms-vntsd:default.log
restarter    svc:/system/svc/restarter:default
contract_id   93
dependency    optional_all/error svc:/milestone/network (online)
dependency    optional_all/none svc:/system/system-log (online)
```

게스트 도메인 만들기 및 시작

게스트 도메인은 하이퍼바이저(hypervisor)에서 나타내는 sun4v 플랫폼과 가상 장치를 모두 이해하는 운영 체제를 실행해야 합니다. 현재 이러한 운영 체제는 최소 Solaris 10 11/06 OS입니다. 필요한 특정 패치에 대한 자세한 내용은 Logical Domains(LDoms) 1.0.2 릴리스 노트를 참조하십시오. 기본 서비스를 만들어 컨트롤 도메인에서 자원을 재할당한 경우 게스트 도메인을 만들어 시작할 수 있습니다.

▼ 게스트 도메인을 만들고 시작하려면

1. 논리적 도메인을 만듭니다.

예를 들어 다음 명령은 `ldg1`로 이름이 지정된 게스트 도메인을 만듭니다.

```
primary$ ldm add-domain ldg1
```

2. 게스트 도메인에 CPU를 추가합니다.

예를 들어 다음 명령은 게스트 도메인 `ldg1`에 4개의 가상 CPU를 추가합니다.

```
primary$ ldm add-vcpu 4 ldg1
```

3. 게스트 도메인에 메모리를 추가합니다.

예를 들어 다음 명령은 게스트 도메인 `ldg1`에 512MB의 메모리를 추가합니다.

```
primary$ ldm add-memory 512m ldg1
```

4. 게스트 도메인에 가상 네트워크 장치를 추가합니다.

예를 들어 다음 명령은 게스트 도메인 ldg1에 이러한 세부사항이 있는 가상 네트워크 장치를 추가합니다.

```
primary$ ldm add-vnet vnet1 primary-vsw0 ldg1
```

여기서,

- vnet1은 논리적 도메인에 대한 고유 인터페이스 이름으로, 후속 set-vnet 또는 remove-vnet 하위 명령에 대한 참조용으로 이 가상 네트워크 장치 인스턴스에 할당됩니다.
- primary-vsw0은 연결할 기존 네트워크 서비스(가상 스위치) 이름입니다.

5. 가상 디스크 서버에서 게스트 도메인에 가상 디스크로 내보낼 장치를 지정합니다.

물리적 디스크, 디스크 슬라이스, 볼륨 또는 파일을 블록 장치로 내보낼 수 있습니다. 루프백(lofi) 장치를 블록 장치로 내보내는 작업은 Logical Domains 소프트웨어의 이번 릴리스에서는 지원되지 않습니다. 다음 예는 물리적 디스크와 파일을 보여 줍니다.

- 물리적 디스크 예. 첫 번째 예는 이러한 세부사항이 있는 물리적 디스크를 추가합니다.

```
primary$ ldm add-vdsdev /dev/dsk/c0t0d0s2 vol1@primary-vds0
```

여기서,

- /dev/dsk/c0t0d0s2는 실제 물리적 장치의 경로 이름입니다. 장치를 추가하는 경우 경로 이름은 장치 이름과 쌍을 이루어야 합니다.
- vol1은 가상 디스크 서버에 추가할 장치에 대해 지정해야 하는 고유 이름입니다. 해당 가상 디스크 서버에서 추가를 위해 장치 이름을 클라이언트로 내보내므로 이 이름은 이 가상 디스크 서버 인스턴스에 고유해야 합니다. 장치를 추가하는 경우 장치 이름은 실제 장치의 경로 이름과 쌍을 이루어야 합니다.
- primary-vds0은 이 장치를 추가할 가상 디스크 서버의 이름입니다.
- 파일 예. 두 번째 예는 블록 장치로 파일을 내보냅니다.

```
primary$ ldm add-vdsdev path-to-file/filename vol1@primary-vds0
```

여기서,

- path-to-file/filename은 블록 장치로 내보내는 실제 파일 경로 이름입니다. 장치를 추가하는 경우 경로 이름은 장치 이름과 쌍을 이루어야 합니다.
- vol1은 가상 디스크 서버에 추가할 장치에 대해 지정해야 하는 고유 이름입니다. 해당 가상 디스크 서버에서 추가를 위해 장치 이름을 클라이언트로 내보내므로 이 이름은 이 가상 디스크 서버 인스턴스에 고유해야 합니다. 장치를 추가하는 경우 장치 이름은 실제 장치의 경로 이름과 쌍을 이루어야 합니다.
- primary-vds0은 이 장치를 추가할 가상 디스크 서버의 이름입니다.

6. 게스트 도메인에 가상 디스크를 추가합니다.

다음 예에서는 게스트 도메인 ldg1에 가상 디스크를 추가합니다.

```
primary$ ldm add-vdisk vdisk1 vol1@primary-vds0 ldg1
```

여기서,

- vdisk1은 가상 디스크 이름입니다.
- vol1은 연결할 기존 가상 디스크 서버 장치의 이름입니다.
- primary-vds0은 연결할 기존 가상 디스크 서버의 이름입니다.

주 - 가상 디스크는 물리적 장치, 볼륨 또는 파일의 서로 다른 유형에 의해 지원되는 일반 블록 장치입니다. 가상 디스크는 SCSI 디스크와 같지 않으므로 디스크 레이블의 대상 ID는 제외합니다. 논리적 도메인에서 가상 디스크의 형식은 cNdNsN이며 여기서 cN은 가상 제어기, dN은 가상 디스크 번호, sN은 슬라이스를 나타냅니다.

7. 게스트 도메인에 대한 auto-boot 및 boot-device 변수를 설정합니다.

첫 번째 명령 예에서 게스트 도메인 ldg1에 대해 auto-boot\?를 true로 설정합니다.

```
primary$ ldm set-var auto-boot\?=true ldg1
```

두 번째 명령 예에서 게스트 도메인 ldg1에 대해 boot-device를 vdisk로 설정합니다.

```
primary$ ldm set-var boot-device=vdisk ldg1
```

8. 자원을 게스트 도메인 ldg1로 바인드한 다음 도메인을 나열하여 바인드되었는지 확인합니다.

```
primary$ ldm bind-domain ldg1
primary$ ldm list-domain ldg1
NAME          STATE    FLAGS    CONS    VCPU  MEMORY    UTIL    UPTIME
ldg1          bound   -----  5001    4     512M
```

9. 게스트 도메인의 콘솔 포트를 찾으려면 이전의 list-domain 하위 명령의 출력을 살펴봅니다.

제목 Cons 아래에서 논리적 도메인 게스트 1(ldg1)이 포트 5001에 바인드된 콘솔 출력을 확인할 수 있습니다.

10. 게스트 도메인 ldg1을 시작합니다.

```
primary$ ldm start-domain ldg1
```

11. 게스트 도메인의 콘솔에 연결합니다. 이를 수행하는 방법은 여러 가지가 있습니다.

- 컨트롤 도메인으로 로그인하여 로컬 호스트에서 콘솔 포트에 직접 연결할 수 있습니다.

```
$ ssh admin@controldom.domain  
$ telnet localhost 5001
```

- 게스트 콘솔이 vntsd(1M) SMF 매니페스트에서 활성화되어 있는 경우 네트워크 상에서 연결할 수 있습니다. 예를 들면 다음과 같습니다.

```
$ telnet host-name 5001
```

서비스 관리 기능 매니페스트는 서비스에 대해 설명하는 XML 파일입니다. SMF 매니페스트를 만드는 방법에 대한 자세한 내용은 Solaris 10 시스템 관리자 모음을 참조하십시오.

주 - 게스트 도메인에서 콘솔을 통해 영어가 아닌 언어 운영 체제에 액세스하려면 콘솔의 단말기는 운영 체제 필요한 로컬에 있어야 합니다.

게스트 도메인 점프 스타팅(Jump-Starting)

게스트 도메인을 점프 스타팅하는 중이면 다음 두 가지 예에서 볼 수 있듯이 정규 Solaris OS JumpStart 절차에서 Ldoms 특정 JumpStart 절차로 변경한 다음 프로파일 구문으로 일반 JumpStart 절차를 사용하게 됩니다.

일반 JumpStart 프로파일

```
filesys c1t1d0s0 free /
filesys c1t1d0s1 2048 swap
filesys c1t1d0s5 120 /spare1
filesys c1t1d0s6 120 /spare2
```

논리적 도메인의 가상 디스크 장치 이름은 해당 이름에 장치 이름의 대상 ID(*tN*)가 없다는 점에서 물리적 디스크 장치 이름과 다릅니다. 일반 *cNtNdNsN* 형식 대신 가상 디스크 장치 이름은 *cNdNsN* 형식으로 되어 있으며 여기서 *cN*은 가상 제어기, *dN*은 가상 디스크 번호, *sN*은 슬라이스입니다. 사용 중인 JumpStart 프로파일을 수정하여 다음 프로파일 예와 같이 이 변경 사항을 반영합니다.

논리적 도메인에 사용된 실제 프로파일

```
filesys c0d0s0 free /
filesys c0d0s1 2048 swap
filesys c0d0s5 120 /spare1
filesys c0d0s6 120 /spare2
```

기타 정보 및 작업

이 장에는 이전 장에서 설명하지 않은 Logical Domains 소프트웨어 사용에 대한 정보 및 작업이 포함되어 있습니다.

CLI 이름 입력 제한 사항

다음 절에서는 Logical Domains Manager CLI에 이름을 입력할 때 적용되는 제한 사항에 대해 설명합니다.

파일 이름(*file*) 및 변수 이름(*var_name*)

- 첫 번째 문자는 문자, 숫자 또는 슬래시(/)여야 합니다.
- 후속 문자는 문자, 숫자 또는 문장 부호여야 합니다.

가상 디스크 서버 *file|device* 및 가상 스위치 장치 이름

- 문자, 숫자 또는 문장 부호를 포함해야 합니다.

구성 이름(*config_name*)

시스템 제어기에 저장된 구성에 할당한 논리적 도메인 구성 이름(*config_name*)은 64자 이하여야 합니다.

기타 모든 이름

논리적 도메인 이름(*ldom*), 서비스 이름(*vswitch_name*, *service_name*, *vdpcs_service_name* 및 *vcc_name*), 가상 네트워크 이름(*if_name*) 및 가상 디스크 이름(*disk_name*)과 같은 나머지 이름은 다음 형식을 따라야 합니다.

- 첫 번째 문자는 문자 또는 숫자여야 합니다.
- 후속 문자는 문자, 숫자 또는 `'_+#.~:~()'` 중 하나여야 합니다.

ldm list 하위 명령 사용

이 절에서는 ldm 하위 명령의 구문 사용법을 표시하고 플래그 및 사용률 통계 등의 일부 출력 조건을 정의하고 해당 출력의 예를 제공합니다.

시스템 읽기 가능 출력

ldm list 명령 출력을 사용하는 스크립트를 만드는 경우 출력의 시스템 읽기 가능 형식을 생성하려면 항상 `-p` 옵션을 사용합니다. 자세한 내용은 [58페이지의 "구문 분석할 수 있는 시스템 읽기 가능 목록\(-p\)을 생성하려면"](#)을 참조하십시오.

▼ ldm 하위 명령 구문 사용법을 표시하려면

- 모든 ldm 하위 명령의 구문 사용법을 보려면 다음을 수행합니다.

코드 예 5-1 모든 ldm 하위 명령의 구문 사용법

```
primary# ldm --help

Usage:
  ldm [--help] command [options] [properties] operands

Command(s) for each resource (aliases in parens):

  bindings
    list-bindings [-e] [-p] [<ldom>...]

  services
    list-bindings [-e] [-p] [<ldom>...]

  constraints
    list-constraints ([-x] | [-e] [-p]) [<ldom>...]
```



```

devices
  list-devices [-a] [-p] [cpu] [mau] [memory] [io]

domain      ( dom )
  add-domain (-i <file> | mac-addr=<num> <ldom> | <ldom>...)
  remove-domain (-a | <ldom>...)
  list-domain [-e] [-l] [-p] [<ldom>...]
  start-domain start-domain (-a | -i <file> | <ldom>...)
  stop-domain stop-domain [-f] (-a | <ldom>...)
  bind-domain (-i <file> | <ldom>)
  unbind-domain <ldom>
  panic-domain <ldom>

io
  add-io [bypass=on] <bus> <ldom>
  remove-io <bus> <ldom>

mau
  add-mau <number> <ldom>
  set-mau <number> <ldom>
  remove-mau <number> <ldom>

memory      ( mem )
  add-memory <number>[GMK] <ldom>
  set-memory <number>[GMK] <ldom>
  remove-memory <number>[GMK] <ldom>

reconf
  remove-reconf <ldom>

spconfig    ( config )
  add-spconfig <config_name>
  set-spconfig <config_name>
  remove-spconfig <config_name>
  list-spconfig

variable    ( var )
  add-variable <var_name>=<value> <ldom>
  set-variable <var_name>=<value> <ldom>
  remove-variable <var_name> <ldom>
  list-variable [<var_name>...] <ldom>

vconscon    ( vcc )
  add-vconscon port-range=<x>-<y> <vcc_name> <ldom>
  set-vconscon port-range=<x>-<y> <vcc_name>
  remove-vconscon [-f] <vcc_name>

```

```

vconsole      ( vcons )
    set-vcons [port=[<port-num>]] [group=<group>] [service=<vcc_server>]
<ldom>

vcpu
    add-vcpu <number> <ldom>
    set-vcpu <number> <ldom>
    remove-vcpu <number> <ldom>

vdisk
    add-vdisk [timeout=<seconds>] <disk_name>
<volume_name>@<service_name> <ldom>
    remove-vdisk [-f] <disk_name> <ldom>

vdiskserver ( vds )
    add-vdiskserver <service_name> <ldom>
    remove-vdiskserver [-f] <service_name>

vdpcc         ( ndpsldcc )
    add-vdpcc <vdpcc_name> <service_name> <ldom>
    remove-vdpcc [-f] <vdpcc_name> <ldom>

vdpcs         ( ndpsldcs )
    add-vdpcs <vdpcs_name> <ldom>
    remove-vdpcs [-f] <vdpcs_name>

vdiskserverdevice ( vdsdev )
    add-vdiskserverdevice [options=<opts>] <file|device>
<volume_name>@<service_name>
    remove-vdiskserverdevice [-f] <volume_name>@<service_name>

vnet
    add-vnet [mac-addr=<num>] <if_name> <vswitch_name> <ldom>
    set-vnet [mac-addr=<num>] [vswitch=<vswitch_name>] <if_name> <ldom>
    remove-vnet [-f] <if_name> <ldom>

vswitch      ( vsw )
    add-vswitch [mac-addr=<num>] [net-dev=<device>] <vswitch_name> <ldom>
    set-vswitch [mac-addr=<num>] [net-dev=<device>] <vswitch_name>
    remove-vswitch [-f] <vswitch_name>

Verb aliases:
    Alias          Verb
    -----
    rm             remove
    ls             list

```

```

Command aliases:
  Alias           Command
  -----
create           add-domain
destroy          remove-domain
cancel-reconf   remove-reconf
start            start-domain
stop             stop-domain
bind             bind-domain
unbind          unbind-domain
panic           panic-domain

```

플래그 정의

다음 플래그는 도메인에 대한 출력에 표시될 수 있습니다.

- 자리 표시자
- c 컨트롤 도메인(control domain)
- d 지연된 재구성(delayed reconfiguration)
- n 일반(normal)
- s 시작(starting) 또는 중지(stopping)
- t 전환(transition)
- v 가상 I/O 도메인(virtual I/O domain)

명령에 긴(-1) 옵션을 사용하는 경우에는 플래그의 전체 이름이 나타납니다. 그렇지 않은 경우에는 약어가 표시됩니다.

목록 플래그 값은 위치에 따라 달라집니다. 다음은 왼쪽에서 오른쪽으로 5개의 열 각각에 나타날 수 있는 값입니다.

열 1	열 2	열 3	열 4	열 5
s 또는 -	n 또는 t	d 또는 -	c 또는 -	v 또는 -

사용률 통계 정의

가상 CPU당 사용률 통계(UTIL)는 `ldm list` 명령의 `-l` 옵션에 표시됩니다. 통계는 마지막 통계 표시 이후 게스트 운영 체제 대신 실행하는 데 사용된 가상 CPU의 시간 백분율입니다. 가상 CPU는 하이퍼바이저로 생성된 경우를 제외하고 게스트 운영 체제를 대신하여 실행되는 것으로 간주합니다. 게스트 운영 체제는 가상 CPU를 하이퍼바이저로 생성하지 않으며 게스트 운영 체제의 CPU 사용률은 항상 100%로 표시됩니다.

논리적 도메인에 대해 보고된 사용률 통계는 해당 도메인에서 가상 CPU에 대한 평균 가상 CPU 사용률입니다.

다양한 목록의 예

▼ 소프트웨어 버전(-v)을 표시하려면

- 현재 설치된 소프트웨어 버전을 표시하려면 다음을 수행합니다. 그러면 다음과 유사한 목록을 받습니다.

코드 예 5-2

설치된 소프트웨어 버전

```
primary$ ldm -v

Logical Domain Manager (v 1.0.2)
  Hypervisor control protocol v 1.0

System PROM:
  Hypervisor      v. 1.5.2           @(#)Hypervisor 1.5.2 2007/09/25 08:39/015
  OpenBoot        v. 4.27.2          @(#)OBP 4.27.2 2007/09/24 16:28
```

▼ 짧은 목록을 생성하려면

- 모든 도메인에 대해 짧은 목록을 생성하려면 다음을 수행합니다.

코드 예 5-3

모든 도메인에 대한 짧은 목록

```
primary$ ldm list
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-t-cv		4	1G	0.5%	3d 21h 7m
ldg1	active	-t---	5000	8	1G	23%	2m

▼ 긴 목록(-l)을 생성하려면

- 모든 도메인에 대해 긴 목록을 생성하려면 다음을 수행합니다.

코드 예 5-4

모든 도메인에 대한 긴 목록

```
primary$ ldm list -l
NAME                STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary             active   -t-cv                    1      768M     0.0%    0s

VCPU
  VID    PID    UTIL  STRAND
  0      0      0.0%  100%

MEMORY
  RA                PA                SIZE
  0x4000000         0x4000000         768M

IO
  DEVICE                PSEUDONYM        OPTIONS
  pci@780                bus_a
  pci@7c0                bus_b             bypass=on

VCC
  NAME                PORT-RANGE
  vcc0                 5000-5100

VSW
  NAME                MAC                NET-DEV    DEVICE    MODE
  vsw0                 08:00:20:aa:bb:e0 e1000g0    switch@0  prog,promisc
  vsw1                 08:00:20:aa:bb:e1                                routed

VDS
  NAME                VOLUME            OPTIONS        DEVICE
  vds0                 myvol-a           slice          /disk/a
                     myvol-b                                /disk/b
                     myvol-c           ro,slice,excl /disk/c
  vds1                 myvol-d

VDPCS
  NAME
  vdpcs0
  vdpcs1

-----
NAME                STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
ldg1                bound   -----  5000    1      512M
```

```

VCPU
  VID    PID    UTIL STRAND
  0      1      100%

MEMORY
  RA          PA          SIZE
  0x4000000  0x34000000  512M

NETWORK
  NAME          SERVICE          DEVICE          MAC
  mynet-b       vsw0@primary    network@0       08:00:20:ab:9a:12
  mynet-a       vsw0@primary    network@1       08:00:20:ab:9a:11

DISK
  NAME          VOLUME          DEVICE          SERVER
  mydisk-a      myvol-a@vds0    disk@0          primary
  mydisk-b      myvol-b@vds0    disk@1          primary

VDPCC
  NAME          SERVICE
  myvdpcc-a     vdpcs0@primary
  myvdpcc-b     vdpcs0@primary

VCONS
  NAME          SERVICE          PORT
  mygroup       vcc0@primary     5000
    
```

▼ 확장 목록(-e)을 생성하려면

- 모든 도메인에 대해 확장 목록을 생성하려면 다음을 수행합니다.

```

primary$ ldm list -e
NAME          STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    UPTIME
primary       active  -t-cv           1      768M     0.0%    0s

VCPU
  VID    PID    UTIL STRAND
  0      0      0.0% 100%

MEMORY
  RA          PA          SIZE
  0x4000000  0x4000000  768M
    
```

```

IO
  DEVICE          PSEUDONYM          OPTIONS
  pci@780         bus_a
  pci@7c0         bus_b              bypass=on

VLDC
  NAME
  primary

VCC
  NAME          PORT-RANGE
  vcc0          5000-5100

VSW
  NAME          MAC          NET-DEV  DEVICE  MODE
  vsw0          08:00:20:aa:bb:e0  e1000g0  switch@0  prog,promisc
  vsw1          08:00:20:aa:bb:e1

VDS
  NAME          VOLUME          OPTIONS          DEVICE
  vds0          myvol-a          slice            /disk/a
                myvol-b                                /disk/b
                myvol-c          ro,slice,excl   /disk/c
  vds1          myvol-d                                /disk/d

VDPCS
  NAME
  vdpcs0
  vdpcs1

VLDCC
  NAME          SERVICE          DESC
  hvctl         primary@primary  hvctl
  vldcc0        primary@primary  ds
-----
NAME          STATE  FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
ldg1          bound  -----  5000  1     512M

VCPU
  VID  PID  UTIL  STRAND
  0    1   100%

MEMORY
  RA          PA          SIZE
  0x4000000  0x34000000  512M
    
```

VLDCC				
NAME	SERVICE	DESC		
vldcc0	primary@primary	ds		
NETWORK				
NAME	SERVICE	DEVICE	MAC	
mynet-b	vsw0@primary	network@0	08:00:20:ab:9a:12	
mynet-a	vsw0@primary	network@1	08:00:20:ab:9a:11	
DISK				
NAME	VOLUME	DEVICE	SERVER	
mydisk-a	myvol-a@vds0	disk@0	primary	
mydisk-b	myvol-b@vds0	disk@1	primary	
VDPCC				
NAME	SERVICE			
myvdpcc-a	vdpcs0@primary			
myvdpcc-b	vdpcs0@primary			
VCONS				
NAME	SERVICE	PORT		
mygroup	vcc0@primary	5000		

▼ 구문 분석할 수 있는 시스템 읽기 가능 목록(-p)을 생성하려면

- 모든 도메인에 대해 구문 분석할 수 있는 시스템 읽기 가능 목록을 생성하려면 다음을 수행합니다.

```
primary$ ldm list -p
VERSION 1.0
DOMAIN|name=primary|state=active|flags=-t-cv|cons=|ncpu=1|mem=805306368|util=
0.0|uptime=0
DOMAIN|name=ldg1|state=bound|flags=-----|cons=5000|ncpu=1|mem=536870912|util=
|uptime=
```


▼ 도메인의 상태를 표시하려면

- 도메인의 상태(예: 게스트 도메인 ldg1)를 표시하려면 다음을 수행합니다.

코드 예 5-7 도메인 상태

```
primary# ldm list-domain ldg1
NAME          STATE   FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
ldg1         active -t---  5000   8     1G     0.3%  2m
```

▼ 변수를 나열하려면

- 도메인(예: ldg1)에 대한 변수(예: boot-device)를 나열하려면 다음을 수행합니다.

코드 예 5-8 도메인에 대한 변수 목록

```
primary$ ldm list-variable boot-device ldg1
boot-device=/virtual-devices@100/channel-devices@200/disk@0:a
```

▼ 바인딩을 나열하려면

- 도메인(예: ldg1)에 바인드된 자원을 나열하려면 다음을 수행합니다.

코드 예 5-9 도메인에 대한 바인딩 목록

```
primary$ ldm list-bindings ldg1
NAME          STATE   FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
ldg1         bound  -----  5000   1     512M

VCPU
  VID  PID  UTIL STRAND
  0    1   100%

MEMORY
  RA          PA          SIZE
  0x4000000   0x3400000   512M

NETWORK
  NAME          SERVICE          DEVICE          MAC
  mynet-b       vsw0@primary     network@0       08:00:20:ab:9a:12
  PEER
  vsw0@primary     08:00:20:aa:bb:e0
  mynet-a@ldg1    08:00:20:ab:9a:11
  mynet-c@ldg2    08:00:20:ab:9a:22
  NAME          SERVICE          DEVICE          MAC
  mynet-a       vsw0@primary     network@1       08:00:20:ab:9a:11
```

PEER	MAC		
vsw0@primary	08:00:20:aa:bb:e0		
mynet-b@ldg1	08:00:20:ab:9a:12		
mynet-c@ldg2	08:00:20:ab:9a:22		
DISK			
NAME	VOLUME	DEVICE	SERVER
mydisk-a	myvol-a@vds0	disk@0	primary
mydisk-b	myvol-b@vds0	disk@1	primary
VDPCC			
NAME	SERVICE		
myvdpcc-a	vdpcs0@primary		
myvdpcc-b	vdpcs0@primary		
VCONS			
NAME	SERVICE	PORT	
mygroup	vcc0@primary	5000	

▼ 구성을 나열하려면

- SC에 저장된 논리적 도메인 구성을 나열하려면 다음을 수행합니다.

코드 예 5-10 구성 목록

```
primary$ ldm list-config
factory-default [current]
initial [next]
```

레이블의 의미

구성 이름 오른쪽의 레이블은 다음을 의미합니다.

- current - 현재 사용 중인 구성
- next - 다음 번 전원 순환에 사용할 구성

▼ 장치를 나열하려면

- 모든 서버 자원(제한 및 무제한)을 나열하려면 다음을 수행합니다.

코드 예 5-11 모든 서버 자원에 대한 목록

```
primary$ ldm list-devices -a
VCPU
  PID  %FREE
  0      0
  1      0
  2      0
  3      0
  4     100
  5     100
  6     100
  7     100
  8     100
  9     100
 10     100
 11     100
 12     100
 13     100
 14     100
 15     100
 16     100
 17     100
 18     100
 19     100
 20     100
 21     100
 22     100
 23     100
 24     100
 25     100
 26     100
 27     100
 28     100
 29     100
 30     100
 31     100

MAU
  CPUSET                                BOUND
  (0, 1, 2, 3)                          ldg2
  (4, 5, 6, 7)
  (8, 9, 10, 11)
  (12, 13, 14, 15)
  (16, 17, 18, 19)
```

```
(20, 21, 22, 23)
(24, 25, 26, 27)
(28, 29, 30, 31)

MEMORY
  PA                SIZE                BOUND
  0x0                512K                _sys_
  0x80000            1536K               _sys_
  0x200000           62M                 _sys_
  0x4000000          768M                primary
  0x34000000         512M                ldg1
  0x54000000         8M                  _sys_
  0x54800000         2G                  ldg2
  0xd4800000         29368M

IO
  DEVICE            PSEUDONYM          BOUND  OPTIONS
  pci@780           bus_a              yes    bypass=on
  pci@7c0           bus_b              yes
```

▼ 서비스를 나열하려면

- 사용 가능한 서비스를 나열하려면 다음을 수행합니다.

```
primary$ ldm list-services
VDS
  NAME                VOLUME                OPTIONS                DEVICE
  primary-vds0
VCC
  NAME                PORT-RANGE
  primary-vcc0        5000-5100
VSW
  NAME                MAC                NET-DEV  DEVICE  MODE
  primary-vsw0        00:14:4f:f9:68:d0 e100g0  switch@0 prog,promisc
```

제약 조건 목록

Logical Domains Manager에 대한 제약 조건은 특정 도메인에 할당하려는 하나 이상의 자원입니다. 사용 가능한 자원에 따라 도메인에 추가하려고 요청한 자원을 모두 받거나 어떠한 자원도 받지 않습니다. list-constraints 하위 명령은 사용자가 도메인에 할당을 요청한 자원을 나열합니다.

▼ 한 개의 도메인에 대한 제약 조건을 나열하려면

- 한 개의 도메인(예: ldg1)에 대한 제약 조건을 나열하려면 다음을 수행합니다.

코드 예 5-13

한 개의 도메인에 대한 제약 조건 목록

```
primary$ ldm list-constraints ldg1
DOMAIN
ldg1

VCPU
  COUNT
  1

MEMORY
  SIZE
  512M

NETWORK
  NAME          SERVICE          DEVICE          MAC
  mynet-b       vsw0             network@0       08:00:20:ab:9a:12
  mynet-b       vsw0             network@0       08:00:20:ab:9a:12

DISK
  NAME          VOLUME
  mydisk-a      myvol-a@vds0
  mydisk-b      myvol-b@vds0

VDPCC
  NAME          SERVICE
  myvdpcc-a     vdpcs0@primary
  myvdpcc-b     vdpcs0@primary

VCONS
  NAME          SERVICE
  mygroup       vcc0
```

▼ XML 형식으로 제약 조건을 나열하려면

- XML 형식으로 특정 도메인(예: ldg1)에 대한 제약 조건을 나열하려면 다음을 수행합니다.

코드 예 5-14

도메인에 대한 제약 조건(XML 형식)

```
primary$ ldm list-constraints -x ldg1
<?xml version="1.0"?>
<LDM_interface version="1.0">
  <data version="2.0">
    <ldom>
      <ldom_info>
        <ldom_name>ldg1</ldom_name>
      </ldom_info>
      <cpu>
        <number>8</number>
      </cpu>
      <memory>
        <size>1G</size>
      </memory>
      <network>
        <vnet_name>vnet0</vnet_name>
        <service_name>primary-vsw0</service_name>
        <mac_address>01:14:4f:fa:0f:55</mac_address>
      </network>
      <disk>
        <vdisk_name>vdisk0</vdisk_name>
        <service_name>primary-vds0</service_name>
        <vol_name>vol0</vol_name>
      </disk>
      <var>
        <name>boot-device</name>
        <value>/virtual-devices@100/channel-devices@200/disk@0:a</value>
      </var>
      <var>
        <name>nvrarc</name>
        <value>devalias vnet0 /virtual-devices@100/channel-devices@200/
network@0</value>
      </var>
      <var>
        <name>use-nvrarc?</name>
        <value>>true</value>
      </var>
    </ldom>
  </data>
</LDM_interface>
```

▼ 시스템 읽기 가능 형식으로 제약 조건을 나열하려면

- 모든 도메인에 대한 제약 조건을 구문 분석 가능 형식으로 나열하려면 다음을 수행합니다.

코드 예 5-15 모든 도메인에 대한 제약 조건(시스템 읽기 가능 형식)

```
primary$ ldm list-constraints -p
VERSION 1.0
DOMAIN|name=primary
MAC|mac-addr=00:03:ba:d8:b1:46
VCPU|count=4
MEMORY|size=805306368
IO
|dev=pci@780|alias=
|dev=pci@7c0|alias=
VDS|name=primary-vds0
|vol=disk-ldg2|opts=|dev=/ldoms/nv72-ldg2/disk
|vol=vol0|opts=|dev=/ldoms/nv72-ldg1/disk
VCC|name=primary-vcc0|port-range=5000-5100
VSW|name=primary-vsw0|mac-addr=|net-dev=e1000g0|dev=switch@0
DOMAIN|name=ldg1
VCPU|count=8
MEMORY|size=1073741824
VARIABLES
|boot-device=/virtual-devices@100/channel-devices@200/disk@0:a
|nvramrc=devalias vnet0 /virtual-devices@100/channel-devices@200/network@0
|use-nvramrc?=true
VNET|name=vnet0|dev=network@0|service=primary-vsw0|mac-addr=01:14:4f:fa:0f:55
VDISK|name=vdisk0|vol=vol0@primary-vds0
```

도메인이 과도하게 로드된 경우 ldm stop-domain 명령이 시간 초과될 수 있음

도메인이 완전히 종료되기 전에 ldm stop-domain 명령이 시간 초과될 수 있습니다. 이 경우 Logical Domains Manager에서 다음과 유사한 오류를 반환합니다.

```
LDom ldg8 stop notification failed
```

하지만 도메인은 계속해서 종료 요청을 처리할 수 있습니다. 도메인의 상태를 확인하려면 `ldm list-domain` 명령을 사용합니다. 예를 들면 다음과 같습니다.

```
# ldm list-domain ldg8
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
ldg8	active	s----	5000	22	3328M	0.3%	1d 14h 31m

이전 목록에는 도메인이 활성으로 표시되지만 `s` 플래그는 해당 도메인을 중지하는 중임을 나타냅니다. 일시적인 상태여야 합니다.

다음 예에서는 이제 도메인이 중지되었음을 표시합니다.

```
# ldm list-domain ldg8
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
ldg8	bound	-----	5000	22	3328M		

가상 네트워크 장치에 해당하는 Solaris 네트워크 인터페이스 이름 확인

게스트에서 `ldm list-*` 명령으로 제공된 출력에서 직접 지정된 가상 장치에 해당하는 Solaris OS 네트워크 인터페이스 이름을 확인할 수 없습니다. 하지만 `ldm list -l` 명령의 출력과 Solaris OS 게스트의 `/devices` 아래에 있는 항목의 조합을 사용하여 이를 수행할 수 있습니다.

▼ Solaris OS 네트워크 인터페이스 이름을 찾으려면

이 예에서 게스트 도메인 `ldg1`에는 두 개의 가상 네트워크 장치(`net-a` 및 `net-c`)가 포함되어 있으며 다음을 수행하여 `ldg1`에서 `net-c`에 해당하는 Solaris OS 네트워크 인터페이스 이름을 찾습니다.

1. ldm 명령을 사용하여 net-c에 대한 가상 네트워크 장치 인스턴스를 찾습니다.

```
# ldm list -l ldg1
...
NETWORK
NAME          SERVICE          DEVICE          MAC
net-a         primary-vsw0@primary  network@0      00:14:4f:f8:91:4f
net-c         primary-vsw0@primary  network@2      00:14:4f:f8:dd:68
...
#
```

net-c에 대한 가상 네트워크 장치 인스턴스는 network@2입니다.

2. ldg1에서 해당 네트워크 인터페이스를 찾으려면 ldg1로 로그인하여 /devices 아래에서 이 인스턴스에 대한 항목을 찾습니다.

```
# uname -n
ldg1
# find /devices/virtual-devices@100 -type c -name network@2\*
/devices/virtual-devices@100/channel-devices@200/network@2:vnet1
#
```

네트워크 인터페이스 이름은 콜론 다음에 오는 항목의 일부로 vnet1입니다.

3. MAC 주소가 1단계에서 net-c에 대한 ldm list -l 출력에 표시된 00:14:4f:f8:dd:68인지 확인하려면 vnet1을 연결(plumb)합니다.

```
# ifconfig vnet1
vnet1: flags=1000842<BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
inet 0.0.0.0 netmask 0
ether 0:14:4f:f8:dd:68
#
```

자동 또는 수동으로 MAC 주소 할당

사용하려는 논리적 도메인, 가상 스위치 및 가상 네트워크 수를 할당하려면 충분한 매체 액세스 제어(media access control, MAC) 주소가 있어야 합니다. Logical Domains Manager에서 논리적 도메인, 가상 네트워크(vnet) 및 가상 스위치(vswitch)에 자동으로 MAC 주소를 할당하거나 사용자의 자체 MAC 주소 할당 풀에서 수동으로 MAC 주소를 할당할 수 있습니다. MAC 주소를 설정하는 ldm 하위 명령은 add-domain, add-vsw, set-vsw, add-vnet 및 set-vnet입니다. 이러한 하위 명령으로 MAC 주소를 지정하지 않으면 Logical Domains Manager에서 자동으로 주소를 할당합니다.

Logical Domains Manager로 MAC 주소를 할당하면 논리적 도메인에 사용할 MAC 주소 블록을 활용할 수 있다는 장점이 있습니다. 또한 Logical Domains Manager는 동일한 서브넷에서 다른 Logical Domains Manager 인스턴스와의 MAC 주소 충돌을 감지하고 이를 방지합니다. 이렇게 하면 수동으로 사용자의 MAC 주소 풀을 관리하지 않아도 됩니다.

MAC 주소는 논리적 도메인이 만들어지거나 도메인에 네트워크 장치가 구성되는 즉시 할당됩니다. 또한 할당된 주소는 해당 장치나 논리적 도메인 자체가 제거될 때까지 유지됩니다.

이 절은 다음 내용으로 구성되어 있습니다.

- 68페이지의 "논리적 도메인 소프트웨어에 할당된 MAC 주소 범위"
- 68페이지의 "자동 할당 알고리즘"
- 69페이지의 "중복 MAC 주소 감지"
- 70페이지의 "해제된 MAC 주소"
- 70페이지의 "수동 MAC 주소 할당"

논리적 도메인 소프트웨어에 할당된 MAC 주소 범위

논리적 도메인이 다음 512K MAC 주소 블록에 할당되었습니다.

00:14:4F:F8:00:00 ~ 00:14:4F:FF:FF:FF

256K 이하 주소는 Logical Domains Manager에서 자동 MAC 주소 할당에 사용되며 수동으로 이 범위의 주소를 요청할 수 없습니다.

00:14:4F:F8:00:00 - 00:14:4F:FB:FF:FF

이 범위의 상반부는 수동 MAC 주소 할당에 사용할 수 있습니다.

00:14:4F:FC:00:00 - 00:14:4F:FF:FF:FF

자동 할당 알고리즘

논리적 도메인이나 네트워크 장치를 만들 때 MAC 주소를 지정하지 않으면 Logical Domains Manager에서 해당 논리적 도메인이나 네트워크 장치에 자동으로 MAC 주소를 할당합니다. 이 MAC 주소를 얻기 위해 Logical Domains Manager는 반복해서 주소를 선택한 다음 충돌 가능성이 있는지 검사합니다.

Logical Domains Manager는 가능한 주소를 선택하기 전에 이 용도로 사용하기 위한 데이터베이스에 최근에 해제되어 저장된 자동 할당 주소가 있는지 확인합니다(70페이지의 "해제된 MAC 주소" 참조). 있으면 Logical Domains Manager는 데이터베이스에서 해당 후보 주소를 선택합니다.

최근에 해제된 주소가 없는 경우에는 이를 위해 별도로 설정된 256K 주소 범위에서 임의로 MAC 주소가 선택됩니다. MAC 주소가 임의로 선택되면 중복된 MAC 주소가 후보로 선택될 확률이 낮아집니다.

그런 다음 선택한 주소를 다른 시스템에 있는 다른 Logical Domains Manager에 대해 검사하여 실제로 중복된 MAC 주소가 할당되지 않도록 합니다. 사용된 알고리즘은 69페이지의 "중복 MAC 주소 감지"에 설명되어 있습니다. 주소가 이미 할당된 경우 Logical Domains Manager는 반복해서 다른 주소를 선택하고 다시 충돌 가능성을 검사합니다. 이 작업은 이미 할당되지 않은 MAC 주소를 찾거나 30초의 제한 시간이 경과될 때까지 지속됩니다. 제한 시간에 도달할 때까지 장치를 만들지 못하면 다음과 유사한 오류 메시지가 표시됩니다.

```
Automatic MAC allocation failed. Please set the vnet MAC address manually.
```

중복 MAC 주소 감지

여러 장치에 동일한 MAC 주소가 할당되지 않도록 해당 컨트롤 도메인의 기본 네트워크 인터페이스에 멀티캐스트 메시지를 보내 Logical Domains Manager에서 해당 장치에 할당하려는 주소를 비롯하여 다른 시스템의 Logical Domain Manager를 검사합니다. MAC 주소 할당을 시도하는 Logical Domains Manager는 1초 동안 응답을 기다립니다. 다른 LDoms 활성화 시스템에 있는 다른 장치에 해당 MAC 주소가 이미 할당된 경우에는 해당 시스템의 Logical Domains Manager에서 해당 MAC 주소가 포함된 응답을 다시 보냅니다. 요청하는 Logical Domains Manager에서 응답을 받으면 선택된 MAC 주소가 이미 할당된 경우 다른 주소를 선택하고 이를 반복합니다.

기본적으로 이러한 멀티캐스트 메시지는 동일한 서버넷의 다른 관리자에게만 보내지며 기본 TTL(time-to-live)은 1입니다. TTL은 서비스 관리 기능(Service Management Facility, SMF) 등록 정보 ldmd/hops를 사용하여 구성할 수 있습니다.

각 Logical Domains Manager는 다음을 담당합니다.

- 멀티캐스트 메시지 수신
- 도메인에 할당된 MAC 주소 추적
- 중복 항목 검색
- 중복이 발생하지 않도록 응답

기타 사유로 시스템의 Logical Domains Manager가 종료된 경우 Logical Domains Manager가 중단되어 있는 동안 중복 MAC 주소가 발생할 수 있습니다.

논리적 도메인이나 네트워크 장치가 만들어질 때 자동 MAC 할당이 발생하며 해당 장치나 논리적 도메인이 제거될 때까지 지속됩니다.

해제된 MAC 주소

자동 MAC 주소와 연결된 논리적 도메인이나 장치가 제거될 때 해당 MAC 주소를 나중
에 해당 시스템에서 사용하기 위해 최근에 해제된 MAC 주소의 데이터베이스에 저장합
니다. 이러한 MAC 주소는 동적 호스트 구성 프로토콜(Dynamic Host Configuration
Protocol, DHCP) 서버에서 인터넷 프로토콜(Internet Protocol, IP) 주소를 모두 소모하
지 않도록 저장됩니다. DHCP 서버에서 IP 주소를 할당할 때 일정 기간 동안(임대 기간)
수행됩니다. 임대 기간은 주로 상당히 긴 시간 또는 일로 구성됩니다. Logical Domains
Manager에서 할당된 MAC 주소를 다시 사용하지 않고 매우 빠른 속도로 네트워크 장치
를 만들고 제거한 경우에는 할당된 MAC 주소 수가 곧 일반적으로 구성된 DHCP 서버를
압도할 수 있습니다.

Logical Domains Manager에서 논리적 도메인이나 네트워크 장치에 대해 자동으로 MAC
주소를 가져오도록 요청된 경우 먼저 해제된 MAC 주소 데이터베이스를 찾아 다시 사용
할 수 있는 이전에 할당된 MAC 주소가 있는지 확인합니다. 이 데이터베이스에 사용 가능
한 MAC 주소가 있으면 중복 MAC 주소 감지 알고리즘이 실행됩니다. MAC 주소가 이전
에 해제된 후 할당되지 않은 경우에는 데이터베이스에서 다시 사용되고 제거됩니다. 충돌
이 감지되면 데이터베이스에서 주소가 간단히 제거됩니다. 그런 다음 Logical Domains
Manager에서 데이터베이스에 있는 다음 주소를 시도하거나 사용 가능한 주소가 없는 경
우 임의로 새 MAC 주소를 선택합니다.

수동 MAC 주소 할당

다음 절차는 수동으로 MAC 주소를 만드는 방법을 알려줍니다.

▼ 수동으로 MAC 주소를 할당하려면

1. 물리적 호스트에 대한 IP 주소의 서브넷 부분을 16진수 형식으로 변환하여 결과를
저장합니다.

```
# grep $hostname /etc/hosts | awk '{print $1}' | awk -F. '{printf("%x", $4)}'  
27
```

2. 컨트롤 도메인을 제외한 현재 도메인 수를 확인합니다.

```
# /opt/SUNWldm/bin/ldm list-domain  
NAME          STATE  FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME  
primary       active -n-cv  SP    4     768M   0.3%  4h 54m  
myldom1      active -n---  5000  2     512M   1.9%  1h 12m
```

하나의 게스트 도메인이 있고 만들려는 도메인을 포함해야 하므로 도메인 수는 2입
니다.

3. 변환된 IP 주소(27)를 공급업체 문자열(0x08020ab)에 추가하고 그 뒤에 논리적 도메인 수(이 예의 경우 2)에 10을 더한 12가 옵니다.

```
0x08020ab and 27 and 12 = 0x08020ab2712 or 8:0:20:ab:27:12
```

CPU 및 메모리 주소 매핑

Solaris 오류 관리 아키텍처(Fault Management Architecture, FMA)는 물리적 CPU 개수 측면에서 CPU 오류를 보고하고 물리적 메모리 주소 측면에서 메모리 오류를 보고합니다.

오류가 발생한 논리적 도메인 및 해당 도메인 내의 가상 CPU 번호 또는 실제 메모리 주소를 확인하려면 매핑을 수행해야 합니다.

CPU 매핑

도메인 및 해당 도메인 내의 가상 CPU 번호는 지정된 물리적 CPU 번호에 해당하며 다음 절차로 확인할 수 있습니다.

▼ CPU 번호를 확인하려면

1. 모든 도메인에 대해 구문 분석 가능한 긴 목록을 생성합니다.

```
primary$ ldm ls -l -p
```

2. 물리적 CPU 번호와 동일한 pid 필드가 있는 목록의 vCPU 섹션에서 해당 항목을 찾습니다.
 - 해당 항목을 찾은 경우 CPU는 아래에 항목이 나열된 도메인에 있으며 해당 도메인 내의 가상 CPU 번호는 항목의 vid 필드에 의해 지정됩니다.
 - 해당 항목을 찾을 수 없는 경우 CPU가 도메인에 없습니다.

메모리 매핑

도메인 및 해당 도메인 내의 실제 메모리 주소는 지정된 물리적 메모리 주소(PA)에 해당하며 다음과 같이 확인할 수 있습니다.

▼ 실제 메모리 주소를 확인하려면

1. 모든 도메인에 대해 구문 분석 가능한 긴 목록을 생성합니다.

```
primary$ ldm ls -l -p
```

2. PA가 포함 범위 pa 에서 $(pa + size - 1)$ 내에 해당하는 목록의 MEMORY 섹션에서 해당 행을 찾습니다. 즉 $pa \leq PA < (pa + size - 1)$ 입니다.

여기에서 pa 및 $size$ 는 행의 해당 필드에 있는 값을 참조합니다.

- 해당 항목을 찾은 경우 PA는 아래에 항목이 나열된 도메인에 있으며 도메인 내의 해당 실제 주소는 $ra + (PA - pa)$ 에 의해 지정됩니다.
- 이러한 항목을 찾을 수 없는 경우 PA가 도메인에 없습니다.

CPU 및 메모리 매핑의 예

코드 예 5-16에 표시된 논리적 도메인 구성이 있고 물리적 CPU 번호 5에 해당하는 도메인 및 가상 CPU와 물리적 주소 0x7e816000에 해당하는 도메인 및 실제 주소를 확인한다고 가정합니다.

목록에서 5와 동일한 pid 필드가 있는 vCPU 항목을 조사하여 논리적 도메인 1dg1 아래에서 다음 항목을 찾을 수 있습니다.

따라서 물리적 CPU 번호 5는 도메인 1dg1에 있으며 해당 도메인 내에 가상 CPU 번호 1을 가집니다.

```
|vid=1|pid=5|util=29|strand=100
```

목록에서 MEMORY 항목을 조사하면 1dg2 도메인 아래에서 다음 항목을 찾을 수 있습니다.

```
ra=0x8000000|pa=0x78000000|size=1073741824
```

여기서 $0x78000000 \leq 0x7e816000 \leq (0x78000000 + 1073741824 - 1)$, 즉 $pa \leq PA \leq (pa + size - 1)$ 입니다.

따라서 PA는 도메인 ldg2에 있으며 해당 실제 주소는 $0x8000000 + (0x7e816000 - 0x78000000) = 0xe816000$ 입니다.

코드 예 5-16

논리적 도메인 구성의 구문 분석 가능한 긴 목록

```
primary$ ldm ls -l -p
VERSION 1.0
DOMAIN|name=primary|state=active|flags=normal,control,vio-service|cons=
SP|ncpu=4|mem=1073741824|util=0.6|uptime=64801|softstate=Solaris running
VCPU
|vid=0|pid=0|util=0.9|strand=100
|vid=1|pid=1|util=0.5|strand=100
|vid=2|pid=2|util=0.6|strand=100
|vid=3|pid=3|util=0.6|strand=100
MEMORY
|ra=0x8000000|pa=0x8000000|size=1073741824
IO
|dev=pci@780|alias=bus_a
|dev=pci@7c0|alias=bus_b
VDS|name=primary-vds0|nclients=2
|vol=disk-ldg1|opts=|dev=/opt/ldoms/testdisk.1
|vol=disk-ldg2|opts=|dev=/opt/ldoms/testdisk.2
VCC|name=primary-vcc0|nclients=2|port-range=5000-5100
VSW|name=primary-vsw0|nclients=2|mac-addr=00:14:4f:fb:42:5c|net-dev=
e1000g0|dev=switch@0|mode=prog,promisc
VCONS|type=SP
DOMAIN|name=ldg1|state=active|flags=normal|cons=5000|ncpu=2|mem=
805306368|util=29|uptime=903|softstate=Solaris running
VCPU
|vid=0|pid=4|util=29|strand=100
|vid=1|pid=5|util=29|strand=100
MEMORY
|ra=0x8000000|pa=0x4800000|size=805306368
VARIABLES
|auto-boot?=true
|boot-device=/virtual-devices@100/channel-devices@200/disk@0
VNET|name=net|dev=network@0|service=primary-vsw0@primary|mac-addr=
00:14:4f:f9:8f:e6
VDISK|name=vdisk-1|vol=disk-ldg1@primary-vds0|dev=disk@0|server=primary
VCONS|group=group1|service=primary-vcc0@primary|port=5000
DOMAIN|name=ldg2|state=active|flags=normal|cons=5001|ncpu=3|mem=
1073741824|util=35|uptime=775|softstate=Solaris running
VCPU
|vid=0|pid=6|util=35|strand=100
|vid=1|pid=7|util=34|strand=100
|vid=2|pid=8|util=35|strand=100
MEMORY
|ra=0x8000000|pa=0x7800000|size=1073741824
VARIABLES
```

```
| auto-boot?=true
| boot-device=/virtual-devices@100/channel-devices@200/disk@0
VNET|name=net|dev=network@0|service=primary-vsw0@primary|mac-addr=
00:14:4f:f9:8f:e7
VDISK|name=vdisk-2|vol=disk-ldg2@primary-vds0|dev=disk@0|server=primary
VCONS|group=group2|service=primary-vcc0@primary|port=5000
```

분할 PCI Express 버스를 구성하여 다중 Logical Domains 사용

주 - Sun SPARC Enterprise T5120 및 T5220과 같은 Sun UltraSPARC T2 기반 서버의 경우 이 절차를 사용하는 대신 논리적 도메인에 네트워크 인터페이스 장치(Network Interface Unit, NIU)를 할당합니다.

Sun UltraSPARC T1 기반 서버의 PCI Express(PCI-E) 버스는 여러 최하위 장치가 연결된 두 개의 포트에 구성됩니다. 이들은 서버에서 pci@780 (bus_a) 및 pci@7c0 (bus_b)로 구분됩니다. 다중 도메인 환경에서 Logical Domains Manager를 사용하여 별도의 도메인에 각각의 최하위 항목을 할당하도록 PCI-E 버스를 프로그래밍할 수 있습니다. 따라서 I/O 가상화를 사용하는 대신 물리적 장치에 직접 액세스된 두 개 이상의 도메인을 활성화할 수 있습니다.

Logical Domains 시스템의 전원이 켜져 있으면 컨트롤(primary) 도메인에서 모든 물리적 장치 자원을 사용하므로 기본 도메인이 두 개의 PCI-E 버스 최하위 항목 모두를 소유하게 됩니다.



주의 - 지원 서버의 모든 내부 디스크는 단일 최하위 항목에 연결되어 있습니다. 컨트롤 도메인이 내부 디스크에서 부트되는 경우에는 도메인에서 해당 최하위 항목을 제거하지 마십시오. 또한 기본 네트워크 포트가 있는 최하위 항목을 제거하면 안 됩니다. 제어 또는 서비스 도메인에서 잘못된 최하위 항목을 제거하면 해당 도메인에서 필수 장치에 액세스할 수 없으므로 사용할 수 없게 됩니다. 기본 네트워크 포트가 시스템 디스크가 아닌 다른 버스에 있으면 장착된 네트워크 포트에서 네트워크 케이블을 이동하고 Logical Domains Manager를 사용하여 이 변경 사항이 반영되도록 가상 스위치(vsw)를 재구성합니다.

▼ 분할 PCI 구성을 만들려면

다음 예는 Sun Fire T2000 서버를 사용한 경우입니다. 또한 이 절차를 Sun Fire T1000 서버 및 Netra T2000 서버와 같은 기타 Sun UltraSPARC T1 기반 서버에서 사용할 수 있습니다. 다른 서버의 경우 지침이 약간 달라질 수 있지만 이 예에서 기본 원칙을 참조할 수 있습니다. 기본적으로 부트 디스크가 있는 최하위 항목을 유지하고 기본 도메인에서 다른 최하위 항목을 제거하여 이를 다른 도메인에 할당해야 합니다.

1. primary 도메인이 두 개의 PCI Express 버스를 모두 소유하는지 확인합니다.

```
primary# ldm list-bindings primary
...
IO
    DEVICE          PSEUDONYM      OPTIONS
    pci@780         bus_a
    pci@7c0         bus_b
...
```

2. 보존해야 하는 부트 디스크의 장치 경로를 확인합니다.

```
primary# df /
/                (/dev/dsk/c1t0d0s0 ): 1309384 blocks  457028 files
```

3. 블록 장치 c1t0d0s0이 연결된 물리적 장치를 확인합니다.

```
primary# ls -l /dev/dsk/c1t0d0s0
lrwxrwxrwx  1 root    root          65 Feb  2 17:19 /dev/dsk/c1t0d0s0 -> ../
../devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/sd@0,0:a
```

이 예에서 primary 도메인의 부트 디스크에 대한 물리적 장치는 최하위 항목 pci@7c0 아래에 있으며 이는 bus_b의 이전 목록에 해당합니다. 이것은 PCI-Express 버스의 bus_a(pci@780)를 다른 도메인에 할당할 수 있음을 의미합니다.

4. /etc/path_to_inst를 확인하여 장착된 네트워크 포트의 물리적 경로를 찾습니다.

```
primary# grep e1000g /etc/path_to_inst
```

5. primary 도메인에서 부트 디스크가 없는 최하위 항목(이 예의 경우 pci@780)을 제거합니다.

```
primary# ldm remove-io pci@780 primary
```

6. 이 분할 PCI 구성(이 예의 경우 split-cfg)을 시스템 제어기에 추가합니다.

```
primary# ldm add-config split-cfg
```

이 구성(split-cfg)을 재부트 후에 사용할 다음 구성으로 설정할 수도 있습니다.

주 - 현재 factory-default 구성을 제외하고 SC에 저장할 수 있는 구성은 8개로 제한됩니다.

7. primary 도메인을 재부트하여 변경 사항을 적용합니다.

```
primary# shutdown -i6 -g0 -y
```

8. 최하위 항목(이 예의 경우 pci@780)을 직접 액세스가 필요한 도메인(이 예의 경우 ldg1)에 추가합니다.

```
primary# ldm add-io pci@780 ldg1
```

```
Notice: the LDom Manager is running in configuration mode. Any
configuration changes made will only take effect after the machine
configuration is downloaded to the system controller and the
host is reset.
```

Infiniband 카드가 있는 경우에는 pci@780 버스에서 바이패스 모드를 활성화해야 할 수 있습니다. 바이패스 모드 활성화 여부에 대한 자세한 내용은 [77페이지의 "PCI 버스에서 I/O MMU 바이패스 모드 활성화"](#)를 참조하십시오.

9. ldg1 도메인을 재부트하여 변경 사항을 적용합니다.

재부트 시 모든 도메인이 비활성 상태여야 합니다. 처음으로 이 도메인을 구성하는 경우 도메인은 비활성 상태가 됩니다.

```
ldg1# shutdown -i6 -g0 -y
```

10. 올바른 최하위 항목이 여전히 primary 도메인과 ldg1 도메인에 할당되어 있는지 확인합니다.

```
primary# ldm list-bindings primary
NAME          STATE  FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
primary       active -n-cv  SP    4     4G     0.4%  18h 25m
...
IO
  DEVICE          PSEUDONYM      OPTIONS
  pci@7c0         bus_b
...
-----
NAME          STATE  FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
ldg1         active -n---  5000  4     2G     10%   35m
...
IO
  DEVICE          PSEUDONYM      OPTIONS
  pci@780         bus_a
...

```

이 출력에서 PCI-E 최하위 bus_b 및 그 아래의 장치가 primary 도메인에 할당되고 bus_a 및 해당 장치가 ldg1에 할당되어 있는지 확인합니다.

PCI 버스에서 I/O MMU 바이패스 모드 활성화

Infiniband 호스트 채널 어댑터(Host Channel Adapter, HCA) 카드가 있는 경우 I/O 메모리 관리 장치(memory management unit, MMU) 바이패스 모드를 켜야 할 수 있습니다. 기본적으로 Logical Domains 소프트웨어는 PCI-E 트랜잭션을 제어하므로 지정된 I/O 장치나 PCI-E 옵션이 해당 I/O 도메인 내에 할당된 물리적 메모리에만 액세스할 수 있습니다. I/O MMU에서 다른 게스트 도메인의 메모리에 액세스하지 못하도록 방지합니다. 따라서 I/O 도메인과 기타 모든 도메인 간에 높은 수준의 보안이 제공됩니다. 드물지만 PCI-E 또는 PCI-X 옵션 카드가 로드되지 않거나 I/O MMU 바이패스 모드가 꺼져 있어 작동하지 않는 경우에는 이 옵션으로 I/O MMU 바이패스 모드를 켤 수 있습니다. 하지만 바이패스 모드를 켜면 I/O 도메인에서 메모리 액세스의 하드웨어 강화 보호가 더 이상 제공되지 않습니다.

bypass=on 옵션으로 I/O MMU 바이패스 모드를 켭니다. 이 바이패스 모드는 모든 게스트 도메인에서 해당 I/O 도메인 내의 각 I/O 도메인과 I/O 장치를 신뢰하는 경우에만 활성화해야 합니다. 이 예에서는 바이패스 모드를 켭니다.

```
primary# ldm add-io bypass=on pci@780 ldg1
```

출력에는 OPTIONS 아래에 bypass=on이 표시됩니다.

콘솔 그룹 사용

가상 네트워크 터미널 서버 데몬 vntsd(1M)를 통해 단일 TCP 포트를 사용하여 여러 도메인 콘솔에 액세스할 수 있습니다. 도메인을 만들 때 Logical Domains Manager는 해당 도메인의 콘솔에 새 기본 그룹을 만들어 각 콘솔에 고유한 TCP 포트를 할당합니다. 그러면 TCP 포트가 콘솔 자체와 반대로 콘솔 그룹에 할당됩니다. 해당 콘솔을 set-vcons 하위 명령을 사용하여 기존 그룹에 바인드할 수 있습니다.

▼ 여러 콘솔을 하나의 그룹으로 결합하려면

1. 도메인의 콘솔을 하나의 그룹으로 바인드합니다.

다음 예에서는 서로 다른 세 도메인(ldg1, ldg2 및 ldg3)의 콘솔을 동일한 콘솔 그룹(group1)으로 바인드하는 것을 보여 줍니다.

```
primary# ldm set-vcons group=group1 service=primary-vc0 ldg1
primary# ldm set-vcons group=group1 service=primary-vc0 ldg2
primary# ldm set-vcons group=group1 service=primary-vc0 ldg3
```

2. 관련 TCP 포트(이 예의 경우 localhost의 포트 5000)에 연결합니다.

```
# telnet localhost 5000
primary-vnts-group1: h, l, c{id}, n{name}, q:
```

하나의 도메인 콘솔을 선택하라는 메시지가 표시됩니다.

3. 1(list)을 선택하여 그룹 내의 도메인을 나열합니다.

```
primary-vnts-group1: h, l, c{id}, n{name}, q: 1
DOMAIN ID          DOMAIN NAME        DOMAIN STATE
0                   ldg1               online
1                   ldg2               online
2                   ldg3               online
```

주 - 콘솔을 다른 그룹이나 vcc 인스턴스에 재할당하려면 해당 도메인을 바인드 해제하여 비활성 상태가 되도록 해야 합니다. vntsd 관리에 대한 SMF 구성 및 사용과 콘솔 그룹 사용에 대한 자세한 내용은 Solaris 10 OS vntsd(1M) 매뉴얼 페이지를 참조하십시오.

한 서버에서 다른 서버로 논리적 도메인 이동

실행 중이 아닌 논리적 도메인을 한 서버에서 다른 서버로 이동할 수 있습니다. 도메인을 이동하기 전에 두 개의 서버에 동일한 도메인을 설정한 경우 도메인 이동이 보다 용이합니다. 실제로 도메인 자체를 이동할 필요 없이 한 서버의 도메인을 바인드 해제한 후 중지하고 다른 서버에서 해당 도메인을 바인드한 후 시작하기만 하면 됩니다.

▼ 이동할 도메인을 설정하려면

1. 두 개의 서버에 동일한 이름을 가진 도메인을 만듭니다. 예를 들어 serverA 및 serverB에 domainA1을 만듭니다.
2. 두 서버에 가상 디스크 서버 장치와 가상 디스크를 추가합니다. 가상 디스크 서버는 바인드의 일부로 내보낼 기본 장치를 엽니다.
3. 하나의 서버(예: serverA)에만 도메인을 바인드합니다. 다른 서버의 도메인은 비활성 상태를 유지합니다.

▼ 도메인을 이동하려면

1. serverA에서 도메인을 바인드 해제한 후 중지합니다.
2. serverB에서 도메인을 바인드한 후 시작합니다.

도메인 바인드

주 - 도메인을 바인드할 때까지 자원을 사용하지 않습니다.

논리적 도메인 제거

이 절에서는 모든 게스트 도메인을 제거하고 전체 서버를 제어하는 단일 OS 인스턴스로 되돌리는 방법에 대해 설명합니다.

▼ 모든 게스트 논리적 도메인을 제거하려면

1. 시스템 제어기의 모든 논리적 도메인 구성을 나열합니다.

```
primary# ldm ls-config
```

2. 이전에 시스템 제어기(system controller, SC)에 저장한 모든 구성(config_name)을 제거합니다. 각 구성에 대해 다음 명령을 사용합니다.

```
primary# ldm rm-config config_name
```

이전에 SC에 저장한 모든 구성을 제거하면 컨트롤 도메인(primary)을 재부트할 때 다음으로 factory-default 도메인이 사용됩니다.

3. -a 옵션을 사용하여 모든 게스트 도메인을 중지합니다.

```
primary# ldm stop-domain -a
```

4. 모든 도메인을 나열하여 게스트 도메인에 연결된 모든 자원을 표시합니다.

```
primary# ldm ls
```

5. 게스트 도메인에 연결된 모든 자원을 릴리스합니다. 이를 실행하려면 시스템에 구성된 각 게스트 도메인(ldom)에 대해 ldm unbind-domain 명령을 사용합니다.

주 - 분할 PCI 구성의 I/O 도메인에서 컨트롤 도메인에 필요한 서비스를 제공하는 경우 이를 바인드 해제하지 못할 수 있습니다. 이 경우에는 이 단계를 건너뛸 수 있습니다.

```
primary# ldm unbind-domain ldom
```

6. 컨트롤 도메인을 중지합니다.

```
primary# shutdown -i1 -g0 -y
```

7. 시스템 제어기의 전원을 순환하여 `factory-default` 구성을 다시 로드합니다.

```
SC> poweroff  
SC> poweron
```

Logical Domains로 Solaris OS 작동

이 절에서는 Logical Domains Manager에서 만든 구성이 인스턴스화되어 도메인 지정이 활성화될 때 Solaris OS 사용 동작에 발생하는 변경 사항에 대해 설명합니다.

주 - 도메인 지정의 활성화 여부에 대한 설명은 Sun UltraSPARC T1 기반 플랫폼에만 연관됩니다. 그렇지 않은 경우에는 도메인 지정이 항상 활성화됩니다.

도메인 지정이 활성화된 경우 Solaris OS를 시작한 후에 OpenBoot 펌웨어를 사용할 수 없음

도메인 지정이 활성화된 경우에는 Solaris OS를 시작한 후에 OpenBoot가 메모리에서 제거되기 때문에 OpenBoot 펌웨어를 사용할 수 없습니다.

Solaris OS에서 `ok` 프롬프트에 도달하려면 해당 도메인을 정지해야 합니다. Solaris OS `halt` 명령을 사용하여 해당 도메인을 정지할 수 있습니다.

서버 전원 순환

서버 전원 순환에 필요한 LDoms 소프트웨어를 실행 중인 시스템에 유지 관리를 수행할 때마다 먼저 현재 논리적 도메인 구성을 SC에 저장해야 합니다.

▼ 현재 논리적 도메인 구성을 SC에 저장하려면

- 다음 명령을 사용합니다.

```
# ldm add-config config_name
```

OpenBoot power-off 명령의 결과

OpenBoot™ power-off 명령으로 시스템의 전원을 끌 수 없습니다. OpenBoot 펌웨어에 있을 때 시스템의 전원을 끄려면 시스템 제어기 또는 시스템 프로세서의 poweroff 명령을 사용합니다. OpenBoot power-off 명령은 다음과 같은 메시지를 표시합니다.

```
NOTICE: power-off command is not supported, use appropriate
NOTICE: command on System Controller to turn power off.
```

Solaris OS 중단 결과

도메인 지정이 활성화되지 않은 경우 Solaris OS는 일반적으로 중단이 실행된 후 OpenBoot 프롬프트로 이동합니다. 이 절에 설명된 동작은 다음 두 가지 상황에서 볼 수 있습니다.

1. 입력 장치가 keyboard로 설정된 경우에는 L1-A 키 시퀀스를 누릅니다.
2. 가상 콘솔이 telnet 프롬프트에 있을 때는 send break 명령을 입력합니다.

도메인 지정이 활성화된 경우에는 이러한 중단 유형 이후에 다음 프롬프트를 받게 됩니다.

```
c)ontinue, s)ync, r)eboot, h)alt?
```

이 중단 유형 이후에 시스템에서 수행하려는 작업을 나타내는 문자를 입력합니다.

컨트롤 도메인 정지 또는 재부트 결과

다음 표는 제어(primary) 도메인의 예상되는 중지 또는 재부트 동작을 보여 줍니다.

주 - 도메인 지정 활성화 여부와 관련된 표 5-1의 질문은 Sun UltraSPARC T1 프로세서에만 연관됩니다. 그렇지 않은 경우에는 도메인 지정이 항상 활성화됩니다.

표 5-1 컨트롤(primary) 도메인의 예상되는 중지 또는 재부트 동작

명령	도메인 지정 활성화 여부	기타 도메인 구성 여부	동작
halt	비활성화됨	해당 없음	Sun UltraSPARC T1 프로세서의 경우: ok 프롬프트로 드롭됩니다.
	활성화됨	아니요	Sun UltraSPARC T1 프로세서의 경우: 02의 메시지를 참조하십시오. Sun UltraSPARC T2 프로세서의 경우: SC에서 전원을 켜 때까지 호스트의 전원이 꺼져 있습니다.
	활성화됨	예	변수 auto-boot?=true인 경우 소프트웨어 재설 정 및 부트됩니다. 변수 auto-boot?=false 인 경우 ok 프롬프트에서 소프트웨어 재설정 및 정 지됩니다.
reboot	비활성화됨	해당 없음	Sun UltraSPARC T1 프로세서의 경우: 호스트의 전원을 끈 후 다시 켭니다.
	활성화됨	아니요	Sun UltraSPARC T1 프로세서의 경우: 호스트의 전원을 끈 후 다시 켭니다. Sun UltraSPARC T2 프로세서의 경우: 전원을 끄지 않고 호스트를 재부트합니다.
	활성화됨	예	Sun UltraSPARC T1 프로세서의 경우: 호스트의 전원을 끈 후 다시 켭니다. Sun UltraSPARC T2 프로세서의 경우: 전원을 끄지 않고 호스트를 재부트합니다.
shutdown -i 5	비활성화됨	해당 없음	Sun UltraSPARC T1 프로세서의 경우: 호스트의 전원을 끕니다.
	활성화됨	아니요	SC에서 전원을 켜 때까지 호스트의 전원이 꺼져 있습니다.
	활성화됨	예	소프트웨어 재설정 및 재부트됩니다.

일부 format(1M) 명령 옵션이 가상 디스크에서 작동하지 않음

Solaris OS format(1M) 명령은 가상 디스크가 있는 게스트 도메인에서 작동하지 않습니다.

- 일부 label, verify 또는 inquiry 등의 하위 명령이 가상 디스크에서 실패합니다.
- format(1M) 명령은 다음과 같은 메시지를 표시할 수 있습니다.
 - Inquiry failed
 - Disk unformatted
 - Current disk is unformatted
 - Drive type unknown
- 확장 펌웨어 인터페이스(extensible firmware interface, EFI) 디스크 레이블이 있는 가상 디스크를 선택하면 format(1M) 명령에서 오류가 발생합니다.
- 게스트 도메인에서 format(1M) 명령을 실행하면 모든 가상 디스크가 올바르게 포맷되고 유효한 디스크 레이블이 있는 경우에도 포맷되지 않은 것으로 표시됩니다.

가상 디스크의 볼륨 목록(volume table of contents, VTOC)을 가져오거나 설정하려면 format(1M) 명령 대신 prtvtoc(1M) 명령 및 fmthard(1M) 명령을 사용합니다. 실제 디스크의 서비스 도메인에서 format(1M) 명령을 사용할 수도 있습니다.

ALOM CMT와 함께 Ldoms를 사용

이 절에서는 Logical Domains Manager와 함께 Advanced Lights Out Manager(ALOM) 칩 다중 스레딩(chip multithreading, CMT)를 사용할 때 알아야 하는 정보에 대해 설명합니다. ALOM CMT 소프트웨어 사용에 대한 자세한 내용은 Advanced Lights Out Management(ALOM) CMT v1.3 안내서를 참조하십시오.



주의 - ALOM CMT 설명서는 하나의 도메인만 참조하므로 Logical Domains Manager에서 여러 도메인을 도입한다는 사실을 알고 있어야 합니다. 논리적 도메인을 다시 시작하는 경우 컨트롤 도메인을 다시 시작할 때까지 게스트 도메인에 대해 I/O 서비스를 사용하지 못할 수 있습니다. 이것은 컨트롤 도메인이 Logical Domains Manager 1.0.2 소프트웨어의 서비스 도메인으로 작동하기 때문입니다. 게스트 도메인은 재부트 프로세스 동안 고정된 것으로 나타납니다. 컨트롤 도메인이 완전히 다시 시작되면 게스트 도메인이 정상적인 작동을 재개합니다. 게스트 도메인은 전체 서버에서 전원이 꺼지는 경우에만 종료해야 합니다.

기존 ALOM CMT 명령에 추가 옵션을 사용할 수 있습니다.

```
bootmode [normal|reset_nvram|bootscript=strong|config="config-name"]
```

`config=config-name?` 옵션을 사용하여 해당 구성을 다음 번에 전원이 켜질 때 `factory-default` 기본 구성을 비롯하여 다른 구성으로 설정할 수 있습니다.

호스트의 전원 상태와 관계없이 해당 명령을 호출할 수 있습니다. 그러면 다음 번에 호스트를 재설정하거나 전원을 켤 때 적용됩니다.

▼ 논리적 도메인 구성을 기본 또는 다른 구성으로 재설정하려면

- ALOM CMT 소프트웨어에서 이 명령을 실행하여 다음 번에 전원을 켤 때 논리적 도메인 구성을 기본 제공 구성으로 재설정합니다.

```
sc> bootmode config="factory-default"
```

또한 `ldm add-config` 명령을 사용하여 Logical Domains Manager에서 만들고 시스템 제어기(system controller, SC)에 저장된 다른 구성을 선택할 수 있습니다.

Logical Domains Manager에서 `ldm add-config` 명령으로 지정한 이름을 사용하여 ALOM CMT `bootmode` 명령을 사용하는 해당 구성을 선택할 수 있습니다. 예를 들어 해당 구성을 `ldm-config1` 이름으로 저장했다고 가정합니다.

```
sc> bootmode config="ldm-config1"
```

`ldm add-config` 명령에 대한 자세한 내용은 `ldm(1M)` 매뉴얼 페이지 또는 Logical Domains(LDoms) Manager 1.0.1 Man Page Guide를 참조하십시오.

BSM 감사 활성화 및 사용

Logical Domains Manager는 Solaris OS 기본 보안 모듈(Basic Security Module, BSM) 감사 기능을 사용합니다. BSM 감사는 컨트롤 도메인에서 작업 및 이벤트 기록을 검사하여 수행된 작업을 확인할 수 있는 수단을 제공합니다. 기록은 수행된 작업, 작업 시기, 작업 수행자 및 해당 결과에 대한 로그로 보존됩니다.

이 절에서는 감사 기능을 사용하려는 경우 감사 로그를 활성화, 확인, 비활성화, 출력 인쇄 및 회전하는 방법에 대해 설명합니다. Solaris 10 System Administration Guide: Security Services에서 BSM 감사에 대한 자세한 내용을 확인할 수 있습니다.

두 가지 방법 중 하나를 사용하여 BSM 감사를 활성화할 수 있습니다. 감사를 비활성화하는 경우에는 감사를 활성화할 때 사용했던 동일한 방법을 사용해야 합니다. 다음과 같은 두 가지 방법이 있습니다.

- Solaris Security Toolkit의 `enable-bsd.fin` 완료 스크립트를 사용합니다.
`enable-bsd.fin` 스크립트는 기본적으로 `ldm_control-secure.driver`에 사용되지 않습니다. 선택한 드라이버에서 완료 스크립트를 활성화해야 합니다.
- Solaris OS `bsmconv(1M)` 명령을 사용합니다.

두 가지 방법을 수행하는 절차는 다음과 같습니다.

▼ `enable-bsd.fin` 완료 스크립트를 사용하려면

1. `ldm_control-secure.driver`를 `my-ldm.driver`로 복사합니다. 여기서 `my-ldm.driver`는 `ldm_control-secure.driver`의 복사본 이름입니다.
2. `ldm_control-config.driver`를 `my-ldm-config.driver`로 복사합니다. 여기서 `my-ldm-config.driver`는 `ldm_control-config.driver`의 복사본 이름입니다.
3. `ldm_control-hardening.driver`를 `my-ldm-hardening.driver`로 복사합니다. 여기서 `my-ldm-hardening.driver`는 `ldm_control-hardening.driver`의 복사본 이름입니다.
4. `my-ldm.driver`가 각각 새 구성 및 강화 드라이버 `my-ldm-config.driver` 및 `my-ldm-hardening.driver`를 참조하도록 편집합니다.
5. `my-ldm-hardening.driver`를 편집하고 해당 드라이버의 다음 행 앞에서 파운드 기호(`#`)를 제거합니다.

```
enable-bsd.fin
```

6. `my-ldm.driver`를 실행합니다.

```
# /opt/SUNWjass/bin/jass-execute -d my-ldm.driver
```

7. Solaris OS를 재부트하여 감사를 적용합니다.

▼ Solaris OS bsmconv(1M) 명령을 사용하려면

1. `vs`를 `/etc/security/audit_control` 파일에 `flags:` 행에 추가합니다.
2. `bsmconv(1M)` 명령을 실행합니다.

```
# /etc/security/bsmconv
```

이 명령에 대한 자세한 내용은 Solaris 10 참조 설명서 모음 또는 해당 매뉴얼 페이지를 참조하십시오.

3. Solaris 운영 체제를 재부트하여 감사를 적용합니다.

▼ BSM 감사가 활성화되었는지 확인하려면

1. 다음 명령을 입력합니다.

```
# auditconfig -getcond
```

2. 출력에 `audit condition = auditing`이 나타나는지 확인합니다.

▼ 감사를 비활성화하려면

감사를 활성화한 방법에 따라 다음 두 가지 방법 중 하나를 사용하여 감사를 비활성화할 수 있습니다. [85페이지의 "BSM 감사 활성화 및 사용"](#)을 참조하십시오.

1. 다음 중 하나를 수행합니다.
 - BSM 감사를 사용하는 Solaris Security Toolkit 강화 실행을 실행 취소합니다.

```
# /opt/SUNWjass/bin/jass-execute -u
```

- Solaris OS `bsmunconv(1M)` 명령을 사용합니다.

```
# /etc/security/bsmunconv
```

2. Solaris OS를 재부트하여 감사를 비활성화시킵니다.

▼ 감사 출력을 인쇄하려면

- 다음 중 한 가지 방법을 사용하여 **BSM** 감사 출력을 인쇄합니다.
 - Solaris OS 명령 `auditreduce(1M)` 및 `praudit(1M)`를 사용하여 감사 출력을 인쇄합니다. 예를 들면 다음과 같습니다.

```
# auditreduce -c vs | praudit
# auditreduce -c vs -a 2006050200000 | praudit
```

- Solaris OS `praudit -x` 명령을 사용하여 XML 출력을 인쇄합니다.

▼ 감사 로그를 회전하려면

- Solaris OS `audit -n` 명령을 사용하여 감사 로그를 회전합니다.

NAT 및 라우팅을 위해 가상 스위치 및 서비스 도메인 구성

가상 스위치(vswitch)는 계층-2 스위치로, 서비스 도메인에서 네트워크 장치로도 사용될 수 있습니다. 가상 스위치가 여러 논리적 도메인에서 가상 네트워크(vnet) 장치 간의 스위치로만 작동하고 물리적 장치를 통해 상자 외부의 네트워크에는 연결되지 않도록 구성할 수 있습니다. 이 모드에서 vswitch를 네트워크 장치로 연결(plumb)하고 서비스 도메인에서 IP 라우팅을 활성화하면 가상 네트워크에서 서비스 도메인을 라우터로 사용하여 상자 외부와 통신할 수 있습니다. 이 작업 모드는 물리적 네트워크 어댑터가 GLDv3과 호환되지 않을 때 도메인에 외부 연결을 제공하기 위해 반드시 필요합니다.

이 구성의 장점은 다음과 같습니다.

- 가상 스위치는 물리적 장치를 직접 사용할 필요가 없으며 기본 장치가 GLDv3과 호환되지 않을 때에도 외부 연결을 제공할 수 있습니다.
- 이 구성은 Solaris OS의 IP 라우팅과 필터링 기능을 활용할 수 있습니다.

▼ 도메인에 외부 연결을 제공하도록 가상 스위치를 설정하려면

1. 물리적 장치와 연결되지 않은 가상 스위치를 만듭니다.
주소를 할당하는 경우 해당 가상 스위치에 고유한 MAC 주소가 있는지 확인합니다.

```
primary# ldm add-vsw [mac-addr=xxxxxxxxxxxx] primary-vsw0 primary
```

2. 가상 스위치를 네트워크 장치 및 해당 도메인에 사용할 물리적 네트워크 장치에 연결(plumb)합니다.
가상 스위치 연결(plumb)에 대한 자세한 내용은 42페이지의 "가상 스위치를 기본 인터페이스로 구성하려면"을 참조하십시오.
3. 필요한 경우 DHCP용으로 가상 스위치 장치를 구성합니다.
DHCP용으로 가상 스위치 장치 구성에 대한 자세한 내용은 42페이지의 "가상 스위치를 기본 인터페이스로 구성하려면"을 참조하십시오.
4. 필요한 경우 /etc/dhcp.vsw 파일을 만듭니다.
5. 서비스 도메인에서 IP 라우팅을 구성하고 모든 도메인에 필수 라우팅 테이블을 설정합니다.
작업 수행 방법에 대한 자세한 내용은 Solaris Express 시스템 관리자 모음에서 System Administration Guide: IP Services의 5장 "Configuring TCP/IP Network Services and IPv4 Addressing"의 "Packet Forwarding and Routing on IPv4 Networks" 절을 참조하십시오.

가상 디스크와 함께 ZFS 사용

이 절은 논리적 도메인에서 가상 디스크와 함께 Zettabyte 파일 시스템(Zettabyte File System, ZFS) 사용에 대한 다음 내용으로 구성되어 있습니다.

- 90페이지의 "ZFS 볼륨 상단에 가상 디스크 만들기"
- 91페이지의 "가상 디스크를 통해서 ZFS 사용"
- 93페이지의 "부트 디스크로 ZFS 사용"

ZFS 볼륨 상단에 가상 디스크 만들기

다음 절차는 서비스 도메인에서 ZFS 볼륨을 만드는 방법과 해당 볼륨을 다른 도메인에서 가상 디스크로 사용할 수 있도록 만드는 방법에 대해 설명합니다. 이 예에서 서비스 도메인은 컨트롤 도메인과 동일하며 이름은 primary로 지정됩니다. 이 예에서 게스트 도메인의 이름은 ldg1입니다. 각 단계에 나타나는 프롬프트는 명령이 실행되는 도메인을 표시합니다.

▼ ZFS 볼륨 상단에 가상 디스크를 만들려면

1. ZFS 저장소 풀(pool)을 만듭니다.

```
primary# zpool create -f tank1 c2t42d1
```

2. ZFS 볼륨을 만듭니다.

```
primary# zfs create -V 100m tank1/myvol
```

3. zpool(이 예의 경우 tank1) 및 ZFS 볼륨(이 예의 경우 tank/myvol)이 만들어졌는지 확인합니다.

```
primary# zfs list
NAME                                USED  AVAIL  REFER  MOUNTPOINT
tank1                                100M  43.0G  24.5K  /tank1
tank1/myvol                          22.5K  43.1G  22.5K  -
```

4. tank1/myvol을 가상 디스크로 내보내는 서비스를 구성합니다.

```
primary# ldm add-vdsdev /dev/zvol/rdisk/tank1/myvol zvol@primary-vds0
```

5. 내보낸 디스크를 다른 도메인(이 예의 경우 ldg1)에 추가합니다.

```
primary# ldm add-vdisk vdisk zvol@primary-vds0 ldg1
```

6. 다른 도메인(이 예의 경우 ldg1)에서 해당 도메인을 시작하고 새 가상 디스크가 표시되는지 확인합니다. devfsadm 명령을 실행해야 할 수 있습니다.

이 예에서 새 디스크는 /dev/rdsk/c2d2s0으로 나타납니다.

```
ldg1# newfs /dev/rdsk/c2d2s0
newfs: construct a new file system /dev/rdsk/c2d2s0: (y/n)? y
Warning: 4096 sector(s) in last cylinder unallocated
Warning: 4096 sector(s) in last cylinder unallocated
```



```
/dev/rdisk/c2d2s0: 204800 sectors in 34 cylinders of 48 tracks, 128 sectors
100.0MB in 3 cyl groups (14 c/g, 42.00MB/g, 20160 i/g) super-block backups
(for fsck -F ufs -o b=#) at: 32, 86176, 172320,
```

```
ldg1# mount /dev/dsk/c2d2s0 /mnt
```

```
ldg1# df -h /mnt
```

Filesystem	size	used	avail	capacity	Mounted on
/dev/dsk/c2d2s0	93M	1.0M	82M	2%	/mnt

주 - ZFS 볼륨이 가상 디스크 슬라이스로 논리적 도메인에 내보내집니다. 따라서 format 명령을 사용하거나 Solaris OS를 zvol 지원 가상 디스크에 설치할 수 없습니다.

가상 디스크를 통해서 ZFS 사용

다음 절차는 가상 디스크 상단에 있는 도메인에서 직접 ZFS를 사용하는 방법을 보여 줍니다. Solaris 10 OS zpool(1M) 및 zfs(1M) 명령을 사용하여 가상 디스크 상단에 ZFS 풀, 파일 시스템 및 볼륨을 만들 수 있습니다. 저장소 백엔드가 달라져도(물리적 디스크 대신 가상 디스크 사용) ZFS 사용법은 변경되지 않습니다.

또한 기존 ZFS 파일 시스템이 있으면 서비스 도메인에서 이를 내보내 다른 도메인에서 사용할 수 있습니다.

이 예에서 서비스 도메인은 컨트롤 도메인과 동일하며 이름은 primary로 지정됩니다. 이 예에서 게스트 도메인의 이름은 ldg1입니다. 각 단계에 나타나는 프롬프트는 명령이 실행되는 도메인을 표시합니다.

▼ 가상 디스크를 통해서 ZFS 사용하려면

1. ZFS 풀(이 예의 경우 tank)을 만든 다음 제대로 만들어졌는지 확인합니다.

```
primary# zpool create -f tank c2t42d0
primary# zpool list
NAME                SIZE  USED  AVAIL  CAP  HEALTH  ALTROOT
tank                43.8G  108K  43.7G  0%   ONLINE  -
```

2. ZFS 파일 시스템(이 예의 경우 tank/test)을 만든 다음 제대로 만들어졌는지 확인합니다.

이 예에서는 서비스 도메인에서 다음 명령을 실행하여 c2t42d0 디스크 상단에 파일 시스템을 만듭니다.

```
primary# zfs create tank/test
primary# zfs list
NAME                USED  AVAIL  REFER  MOUNTPOINT
tank                 106K  43.1G  25.5K  /tank
tank/test            24.5K  43.1G  24.5K  /tank/test
```

3. ZFS 풀(이 예의 경우 tank)을 내보냅니다.

```
primary# zpool export tank
```

4. 물리적 디스크 c2t42d0s2를 가상 디스크로 내보내는 서비스를 구성합니다.

```
primary# ldm add-vdsdev /dev/rdisk/c2t42d0s2 volz@primary-vds0
```

5. 내보낸 디스크를 다른 도메인(이 예의 경우 ldg1)에 추가합니다.

```
primary# ldm add-vdisk vdiskz volz@primary-vds0 ldg1
```

6. 다른 도메인(이 예의 경우 ldg1)에서 해당 도메인을 시작하고 새 가상 디스크가 표시되는지 확인한 다음(devfsadm 명령을 실행해야 할 수 있음) ZFS 풀을 가져옵니다.

```
ldg1# zpool import tank
ldg1# zpool list
NAME                SIZE  USED  AVAIL  CAP  HEALTH  ALTROOT
tank                 43.8G  214K  43.7G   0%  ONLINE  -

ldg1# zfs list
NAME                USED  AVAIL  REFER  MOUNTPOINT
tank                 106K  43.1G  25.5K  /tank
tank/test            24.5K  43.1G  24.5K  /tank/test

ldg1# df -hl -F zfs
Filesystem          size  used  avail capacity  Mounted on
tank                 43G   25K   43G     1%    /tank
tank/test            43G   24K   43G     1%    /tank/test
```

이제 ZFS 풀(이 예의 경우 tank/test)을 가져와서 ldg1 도메인에서 사용할 수 있습니다.

부트 디스크로 ZFS 사용

논리적 도메인에서 큰 파일이 있는 ZFS 파일 시스템을 가상 디스크로 사용할 수 있습니다.

주 - ZFS 파일 시스템은 서비스 도메인에서 더 많은 메모리가 필요합니다. 서비스 도메인을 구성할 때 이 점을 고려하십시오.

ZFS에서는 다음 작업이 가능합니다.

- 신속하게 파일 시스템 복제
- 복제를 사용하여 추가 도메인 제공
- ZFS 파일 시스템 내의 파일에서 디스크로 넷 설치

▼ 부트 디스크로 ZFS를 사용하려면

다음 절차를 사용하여 논리적 도메인에 ZFS 디스크를 만들거나 스냅샷을 만들고 다른 도메인에 이를 복제할 수 있습니다.

1. primary 도메인에서 ZFS 풀의 저장소로 사용할 전체 디스크 또는 슬라이스를 확보합니다. 2단계에서는 디스크의 슬라이스 5를 사용합니다.
2. ZFS 풀(예: ldomspool)을 만듭니다.

```
# zpool create ldomspool /dev/dsk/c0t0d0s5
```

3. 첫 번째 도메인(이 예의 경우 ldg1)에 대해 ZFS 파일 시스템을 만듭니다.

```
# zfs create ldomspool/ldg1
```

4. 이 도메인의 디스크로 사용할 파일을 만듭니다.

```
# mkfile 1G /ldomspool/ldg1/bootdisk
```

5. 파일을 도메인을 만들 때 사용할 장치로 지정합니다.

```
primary# ldm add-vdsdev /ldomspool/ldg1/bootdisk vol1@primary-vds0
primary# ldm add-vdisk vdisk1 vol1@primary-vds0 ldg1
```

6. ldg1 도메인을 부트하고 vdisk1로 넷 설치합니다. 이 파일은 전체 디스크로 작동하고 root, usr, home, dump 및 swap에 별도의 분할 영역을 가질 수 있습니다.

7. 설치가 완료되면 파일 시스템을 스냅샷합니다.

```
# zfs snapshot ldomspool/ldg1@initial
```

주 - 도메인을 재부트하기 전에 스냅샷을 수행하면 해당 도메인 상태가 스냅샷의 일부로 저장되거나 스냅샷에서 만들어진 다른 복제로 저장되지 않습니다.

8. 스냅샷에서 추가 복제를 만들고 이를 다른 도메인(이 예의 경우 ldg2 및 ldg3)의 부트 디스크로 사용합니다.

```
# zfs clone ldomspool/ldg1@initial ldomspool/ldg2
# zfs clone ldomspool/ldg1@initial ldomspool/ldg3
```

9. 모든 항목이 성공적으로 만들어졌는지 확인합니다.

```
# zfs list
NAME                                USED  AVAIL  REFER  MOUNTPOINT
ldomspool                           1.07G  2.84G  28.5K  /ldomspool
ldomspool/ldg1                       1.03G  2.84G  1.00G  /ldomspool/ldg1
ldomspool/ldg1@initial                23.0M      -  1.00G  -
ldomspool/ldg2                       23.2M  2.84G  1.00G  /ldomspool/ldg2
ldomspool/ldg3                       21.0M  2.84G  1.00G  /ldomspool/ldg3
```

주 - ZFS 풀에 복제를 만들 충분한 공간이 있는지 확인합니다. ZFS는 기록 중 복사를 사용하고 해당 복제의 블록을 수정할 때만 해당 풀의 공간을 사용합니다. 도메인을 부트한 후라도 복제는 디스크에 필요한 최소 퍼센트만 사용하며 이는 대부분의 OS 이진 이 최초 스냅샷의 OS 이진과 동일하기 때문입니다.

Logical Domains 환경에서 Volume Manager 사용

이 절은 다음 내용으로 구성되어 있습니다.

- 95페이지의 "Volume Manager 상단에서 가상 디스크 사용"
- 97페이지의 "가상 디스크 상단에서 Volume Manager 사용"

Volume Manager 상단에서 가상 디스크 사용

Zettabyte 파일 시스템(Zettabyte File System, ZFS), Solaris™ Volume Manager(SVM) 또는 Veritas Volume Manager(VxVM) 볼륨을 가상 디스크로 서비스 도메인에서 게스트 도메인으로 내보낼 수 있습니다. 내보낸 볼륨은 게스트 도메인에서 단일 슬라이스(s0)가 있는 가상 디스크로 나타납니다.

주 - 이 절의 나머지 부분에서는 SVM 볼륨을 예로 사용합니다. 하지만 해당 설명이 ZFS 및 VxVM 볼륨에도 적용됩니다.

예를 들어 서비스 도메인에서 SVM 볼륨 /dev/md/dsk/d0을 domain1로 내보내고 domain1에서 해당 가상 디스크를 /dev/dsk/c0d2*로 보는 경우 domain1에는 s0 장치, 즉 /dev/dsk/c0d2s0 장치만 있습니다.

게스트 도메인의 가상 디스크(예: /dev/dsk/c0d2s0)는 관련 볼륨(예: /dev/md/dsk/d0)에 직접 매핑되고 게스트 도메인의 가상 디스크에 저장된 데이터는 추가 메타데이터가 없는 관련 볼륨에 직접 저장됩니다. 따라서 게스트 도메인의 가상 디스크에 저장된 데이터는 관련 볼륨을 통해 서비스 도메인에서 직접 액세스할 수도 있습니다.

예를 들면 다음과 같습니다.

- SVM 볼륨 d0을 primary 도메인에서 domain1로 내보내면 domain1을 구성할 때 추가 단계가 필요합니다.

```
primary# metainit d0 3 1 c2t70d0s6 1 c2t80d0s6 1 c2t90d0s6
primary# ldm add-vdsdev /dev/md/dsk/d0 vol3@primary-vds0
primary# ldm add-vdisk vdisk3 vol3@primary-vds0 domain1
```

- domain1을 바인드하고 시작한 후 내보낸 볼륨이 /dev/dsk/c0d2s0 등으로 나타나면 사용자가 이를 사용할 수 있습니다.

```
domain1# newfs /dev/rdisk/c0d2s0
domain1# mount /dev/dsk/c0d2s0 /mnt
domain1# echo test-domain1 > /mnt/file
```

- domain1을 중지하고 바인드 해제한 후 domain1에서 가상 디스크에 저장된 데이터를 SVM 볼륨 d0을 통해 기본 도메인에서 직접 액세스할 수 있습니다.

```
primary# mount /dev/md/dsk/d0 /mnt
primary# cat /mnt/file
test-domain1
```

주 - 이러한 가상 디스크를 `format(1M)` 명령으로 표시하거나 영역을 분할하거나 Solaris OS의 설치 디스크로 사용할 수 없습니다. 자세한 내용은 [84페이지의 "일부 format\(1M\) 명령 옵션이 가상 디스크에서 작동하지 않음"](#)을 참조하십시오.

SVM의 상단에서 가상 디스크 사용

다른 도메인에서 RAID 또는 미리 SVM 볼륨을 가상 디스크로 사용할 때 SVM 볼륨의 구성 요소 중 하나에서 오류가 발생하면 `metareplace` 명령을 사용하거나 핫 스페어를 사용하여 SVM 복구를 시작할 수 없습니다. `metastat` 명령에서 해당 볼륨을 재동기화 중으로 표시하지만 재동기화가 진행되지 않습니다.

예를 들어 `/dev/md/dsk/d0`은 다른 도메인에서 가상 디스크로 내보낸 RAID SVM 볼륨이고 `d0`은 일부 핫 스페어 장치로 구성되어 있습니다. `d0`의 구성 요소가 실패하면 SVM은 오류가 발생한 구성 요소를 핫 스페어로 바꾸고 SVM 볼륨을 재동기화합니다. 하지만 재동기화가 시작되지 않습니다. 해당 볼륨이 재동기화 중인 것으로 보고되지만 재동기화가 진행되지 않습니다.

```
# metastat d0
d0: RAID
  State: Resyncing
  Hot spare pool: hsp000
  Interlace: 32 blocks
  Size: 20097600 blocks (9.6 GB)
Original device:
  Size: 20100992 blocks (9.6 GB)
Device                               Start Block  Dbase   State Reloc
c2t2d0s1                               330         No      Okay   Yes
c4t12d0s1                              330         No      Okay   Yes
/dev/dsk/c10t600C0FF000000000015153295A4B100d0s1 330         No      Resyncing Yes
```

이 경우에는 SVM 볼륨을 가상 디스크로 사용하는 도메인을 중지하고 바인드 해제하여 재동기화를 완료해야 합니다. 그런 다음 `metasync` 명령을 사용하여 SVM 볼륨을 재동기화할 수 있습니다.

```
# metasync d0
```

VxVM이 설치된 경우 가상 디스크 사용

시스템에 VERITAS 볼륨 관리자(Veritas Volume Manager)가 설치된 경우 가상 디스크로 내보내려는 물리적 디스크나 분할 영역에서 Veritas 동적 다중 경로 지정(Dynamic Multipathing, DMP)이 활성화되지 못하도록 해야 합니다. 그렇지 않으면 해당 디스크를 사용하는 도메인을 바인드할 때 /var/adm/messages에서 오류가 발생합니다.

```
vd_setup_vd(): ldi_open_by_name(/dev/dsk/c4t12d0s2) = errno 16
vds_add_vd(): Failed to add vdisk ID 0
```

vxdisk list 명령의 출력에서 다중 경로 지정 정보를 확인하여 Veritas DMP가 활성화되었는지 확인할 수 있습니다. 예를 들면 다음과 같습니다.

```
# vxdisk list Disk_3
Device:      Disk_3
devicetag:   Disk_3
type:        auto
info:        format=none
flags:       online ready private autoconfig invalid
pubpaths:    block=/dev/vx/dmp/Disk_3s2 char=/dev/vx/rdmp/Disk_3s2
guid:        -
udid:        SEAGATE%5FST336753LSUN36G%5FDISKS%5F3032333948303144304E0000
site:        -
Multipathing information:
numpaths:    1
c4t12d0s2   state=enabled
```

가상 디스크로 내보내려는 디스크나 슬라이스에서 Veritas DMP를 활성화한 경우에는 vxdmpadm 명령을 사용하여 DMP를 비활성화해야 합니다. 예를 들면 다음과 같습니다.

```
# vxdmpadm -f disable path=/dev/dsk/c4t12d0s2
```

가상 디스크 상단에서 Volume Manager 사용

이 절에서는 Logical Domains 환경에서 다음과 같은 상황에 대해 설명합니다.

- 98페이지의 "가상 디스크 상단에서 ZFS 사용"
- 98페이지의 "가상 디스크 상단에서 SVM 사용"
- 98페이지의 "가상 디스크 상단에서 VxVM 사용"

가상 디스크 상단에서 ZFS 사용

모든 가상 디스크를 ZFS에서 사용할 수 있습니다. ZFS 저장소 풀(zpool)을 임의의 도메인으로 가져올 수 있으며, 이 때 도메인에서 모든 저장소 장치를 가상 장치 또는 실제 장치로 표시하는지에 관계 없이 해당 장치를 이 zpool의 일부로 표시합니다.

가상 디스크 상단에서 SVM 사용

모든 가상 디스크를 SVM 로컬 디스크 세트에서 사용할 수 있습니다. 예를 들어 가상 디스크를 사용하여 로컬 디스크 세트의 SVM 메타 데이터베이스(metadb)를 저장하거나 로컬 디스크 세트에서 SVM 볼륨을 만들 수 있습니다.

현재 가상 디스크를 로컬 디스크 세트에서만 사용할 수 있고 공유 디스크 세트(metaset)에서는 사용할 수 없습니다. 가상 디스크를 SVM 공유 디스크 세트에 추가할 수 없습니다. 가상 디스크를 SVM 공유 디스크에 추가하려고 하면 다음과 유사한 오류 메시지를 표시하며 실패합니다.

```
# metaset -s test -a c2d2
metaset: domain1: test: failed to reserve any drives
```

가상 디스크 상단에서 VxVM 사용

현재 VxVM은 가상 디스크에서 작동하지 않습니다. VxVM 소프트웨어를 가상 디스크가 있는 도메인에 설치할 수 있지만 VxVM에서 사용 가능한 가상 디스크를 표시할 수 없습니다.

Logical Domains 환경에서 IPMP 구성

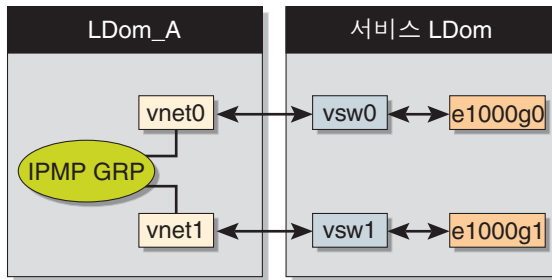
인터넷 프로토콜 네트워크 다중 경로 지정(Internet Protocol Network Multipathing, IPMP)은 여러 네트워크 인터페이스 카드에 결합 허용 및 로드 균형 조정을 제공합니다. IPMP를 사용하여 하나 이상의 인터페이스를 IP 다중 경로 지정 그룹에 구성할 수 있습니다. IPMP를 구성한 후 시스템에서 자동으로 IPMP 그룹에서 오류가 있는 인터페이스를 모니터링합니다. 그룹의 인터페이스에 오류가 있거나 유지 관리를 위해 제거된 경우 IPMP는 자동으로 오류가 발생한 인터페이스의 IP 주소를 마이그레이션하거나 페일오버합니다. Logical Domains 환경에서 IPMP를 사용하여 페일오버할 물리적 또는 가상 네트워크 인터페이스를 구성할 수 있습니다.

논리적 도메인에서 IPMP 그룹에 가상 네트워크 장치 구성

논리적 도메인에서 해당 가상 네트워크 장치를 IPMP 그룹에 구성하여 결합이 허용되도록 구성할 수 있습니다. 활성 대기 구성에서 가상 네트워크 장치로 IPMP 그룹을 설정할 때 해당 그룹에서 검사 기반 감지를 사용하도록 설정할 수 있습니다. 현재 Logical Domains 1.0.2 소프트웨어의 가상 네트워크 장치에 연결 기반 감지 및 페일오버가 지원되지 않습니다.

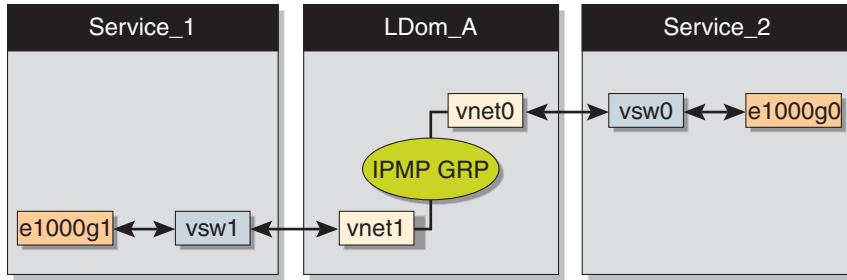
다음 다이어그램은 서비스 도메인에서 별도의 가상 스위치 인스턴스(vsw0 및 vsw1)에 연결되고 차례로 두 개의 다른 물리적 인터페이스(e1000g0 및 e1000g1)를 사용하는 두 개의 가상 네트워크(vnet0 및 vnet1)를 표시합니다. 물리적 인터페이스에 오류가 발생하는 경우 LDom_A의 IP 계층에서 검사 기반 감지를 통해 해당 vnet에서 오류 및 연결 손실을 감지하고 자동으로 두 번째 vnet 장치로 페일오버합니다.

그림 5-1 별도의 가상 스위치 인스턴스에 연결된 두 개의 가상 네트워크



서로 다른 서비스 도메인에서 각 가상 네트워크 장치(vnet0 및 vnet1)를 가상 스위치 인스턴스로 연결하여 논리적 도메인에서 추가로 신뢰성을 확보할 수 있습니다(다음 다이어그램에 표시). 분할 PCI 구성을 사용하여 두 개의 서비스 도메인(Service_1 및 Service_2)을 가상 스위치 인스턴스(vsw1 및 vsw2)와 함께 설정할 수 있습니다. 이 경우 LDom_A에서 네트워크 하드웨어 오류뿐만 아니라 가상 네트워크 오류도 감지하고 서비스 도메인 충돌 또는 종료 후에 페일오버를 트리거할 수 있습니다.

그림 5-2 서로 다른 서비스 도메인에 연결된 각각의 가상 네트워크 장치



IPMP 그룹 구성 및 사용 방법에 대한 자세한 내용은 Solaris 10 System Administration Guide: IP Services를 참조하십시오.

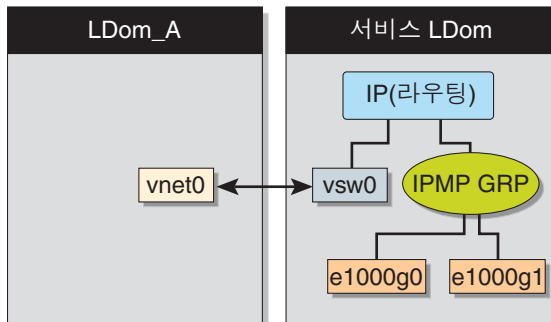
서비스 도메인에서 IPMP 구성 및 사용

또한 서비스 도메인에서 물리적 인터페이스를 IPMP 그룹으로 구성하여 Logical Domains 환경에서 네트워크 오류 감지 및 복구를 설정할 수 있습니다. 이를 수행하려면 서비스 도메인에서 가상 스위치를 네트워크 장치로 구성하고 해당 서비스 도메인 자체가 IP 라우터로 작동하도록 구성합니다. IP 라우팅 설정에 대한 자세한 내용은 Solaris 10 System Administration Guide: IP Services를 참조하십시오.

일단 구성되면 가상 스위치는 가상 네트워크(및 외부 시스템으로 향하는)에서 생성된 모든 패킷을 물리적 장치를 통해 직접 보내는 대신 해당 IP 계층으로 보냅니다. 물리적 인터페이스에 오류가 발생한 경우 IP 계층은 오류를 감지하고 자동으로 두 번째 인터페이스를 통해 패킷을 다시 라우팅합니다.

물리적 인터페이스가 직접 IPMP 그룹으로 구성되므로 해당 그룹을 연결 기반 또는 검사 기반 감지로 설정할 수 있습니다. 다음 다이어그램은 IPMP 그룹의 일부로 구성된 두 개의 네트워크 인터페이스(e1000g0 및 e1000g1)를 표시합니다. 네트워크 장치로 연결(plumb)된 가상 스위치 인스턴스(vsw0)는 해당 IP 계층으로 패킷을 보냅니다.

그림 5-3 IPMP 그룹의 일부로 구성된 두 개의 네트워크 인터페이스



용어집

이 목록에서는 Logical Domains 1.0.2 설명서에 나오는 용어, 약어 및 머리글자를 정의합니다.

A

- ALOM CMT** Advanced Lights Out Manager 칩 멀티스레딩(Advanced Lights Out Manager chip multithreading), 시스템 제어기에서 실행되며 이를 사용하여 CMT 서버를 모니터 및 제어할 수 있습니다.
- auditing(감사)** Solaris OS BSM을 사용하여 보안 변경 사항의 소스를 식별합니다.
- authorization(인증)** Solaris OS RBAC를 사용하여 인증을 설정합니다.

B

- bge** Broadcom BCM57xx 장치의 Broadcom Gigabit Ethernet 드라이버
- BSM** Basic Security Module(기본 보안 모듈)

C

- CLI** command-line interface(명령줄 인터페이스)
- compliance(준수)** 시스템의 구성이 미리 정의된 보안 프로필을 준수하는지 확인합니다.
- config** 시스템 제어기에 저장된 논리적 도메인 구성의 이름입니다.
- CMT** chip multithreading(칩 멀티스레딩)
- constraints(제약 조건)** Logical Domains Manager에 대한 제약 조건은 특정 도메인에 할당하려는 하나 이상의 자원입니다. 사용 가능한 자원에 따라 도메인에 추가하려고 요청한 자원을 모두 받거나 어떠한 자원도 받지 않습니다.
- control domain**
(컨트롤 도메인) 기타 논리적 도메인 및 서비스를 만들고 관리하는 도메인입니다.
- CPU** central processing unit(중앙 처리 장치)
- CWQ** Control Word Queue(제어 단어 대기열), Sun UltraSPARC T2 기반 플랫폼의 암호화 장치

D

- DHCP** Dynamic Host Configuration Protocol(동적 호스트 구성 프로토콜)
- DMP** Dynamic Multipathing(동적 다중 경로 지정)(Veritas)
- DR** dynamic reconfiguration(동적 재구성)
- drd(1M)** dynamic reconfiguration daemon(Logical Domains Manager(Solaris 10 OS)의 동적 재구성 데몬)
- DS** Domain Services(동적 서비스 모듈(Solaris 10 OS))

E

- e1000g** 네트워크 인터페이스 제어기의 Intel PRO/1000 Gigabit 제품군에 대한 드라이버
- EFI** extensible firmware interface(확장 가능 펌웨어 인터페이스)
- ETM** Encoding Table Management module(인코딩 테이블 관리 모듈(Solaris 10 OS))

F

- FC_AL** Fiber Channel Arbitrated Loop(광 섬유 채널 중재 루프)
- FMA** Fault Management Architecture(오류 관리 아키텍처)
- fmd(1M)** Fault Manager daemon(Fault Manager 데몬(Solaris 10 OS))
- FTP** File Transfer Protocol(파일 전송 프로토콜)

G

- guest domain**
(게스트 도메인) I/O 및 서비스 도메인에서 서비스를 사용하고 컨트롤 도메인에서 관리합니다.
- GLDv3** Generic LAN Driver version 3(일반 LAN 드라이버 버전 3)

H

- hardening(강화)** Solaris OS 구성을 수정하여 보안을 향상시킵니다.
- HDD** hard disk drive(하드 디스크 드라이브)
- hypervisor**
(하이퍼바이저) 운영 체제와 하드웨어 계층 사이에 삽입된 펌웨어 계층입니다.

I

- io** 내부 디스크, PCI-Express(PCI-E) 제어기, 연결된 어댑터 및 장치와 같은 I/O 장치입니다.
- IB** Infiniband
- I/O domain**
(I/O 도메인) 물리적 I/O 장치에 대한 직접 소유권이 있고 해당 장치에 직접 액세스하며 다른 논리적 도메인에 가상 장치의 형식으로 해당 장치를 공유하는 도메인입니다.

- ioctl input/output control call(입/출력 제어 호출)
- IP** Internet Protocol(인터넷 프로토콜)
- IPMP** Internet Protocol Network Multipathing(인터넷 프로토콜 네트워크 다중 경로 지정)

K

- kaio kernel asynchronous input/output(커널 비동기 입/출력)
- KB** kilobyte(킬로바이트)
- KU** kernel update(커널 업데이트)

L

- LAN** local-area network(로컬 영역 네트워크)
- LDAP** Lightweight Directory Access Protocol(단순 디렉토리 액세스 프로토콜)
- LDC** logical domain channel(논리적 도메인 채널)
- ldm(1M)** Logical Domain Manager utility(Logical Domain Manager 유틸리티)
- ldmd** Logical Domains Manager daemon(Logical Domains Manager 데몬)
- logical domain**
(논리적 도메인) 단일 컴퓨터 시스템 내의 운영 체제, 자원 및 ID가 있는 개별 논리적 그룹화입니다.
- Logical Domains(LDoms) Manager** CLI를 제공하여 논리적 도메인을 만들고 관리하며 도메인에 자원을 할당합니다.

M

- MAC** media access control(매체 액세스 제어) 주소, Ldoms에서 자동으로 할당하거나 사용자가 수동으로 할당할 수 있습니다.
- MAU** Modular Arithmetic Unit(모듈식 산술 장치, Sun UltraSPARC T1 기반 플랫폼의 암호화 장치)
- MB** megabyte(메가바이트)
- MD** 서버 데이터베이스의 시스템 설명(machine description)입니다.
- mem, memory 메모리 장치 - 바이트(기본 크기), 기가바이트(G), 킬로바이트(K) 또는 메가바이트(M)로 지정합니다. 게스트 도메인에 할당할 수 있는 서버의 가상 메모리입니다.
- MMF** Multimode Fiber(다중 모드 파이버)
- MIB** Management Information Base(관리 정보 기반)
- minimizing(최소화)** 필요한 핵심 Solaris OS 패키지의 수를 최소로 설치합니다.
- MMU** memory management unit(메모리 관리 장치)
- mtu maximum transmission unit(최대 전송 장치)

N

- NAT** Network Address Translation(네트워크 주소 변환)
- NDPSS** Netra Data Plane 소프트웨어 제품군
- ndpsldcc Netra Data Plane Software Logical Domain Channel Client(Netra Data Plane 소프트웨어 논리적 도메인 채널 클라이언트) vdpcc를 참조하십시오.
- ndpsldcs Netra Data Plane Software Logical Domain Channel Service(Netra Data Plane 소프트웨어 논리적 도메인 채널 서비스) vdpcc를 참조하십시오.
- NFS** Network File System(네트워크 파일 시스템)
- NIS** Network Information Services(네트워크 정보 서비스)
- NIU** Network Interface Unit(네트워크 인터페이스 장치(Sun SPARC Enterprise T5120 및 T5220 서버))
- NTS** network terminal server(네트워크 터미널 서버)
- NVRAM** non-volatile random-access memory(비휘발성 무작위 메모리)
- nxge Sun x8 Express 1/10G Ethernet Adapter용 드라이버

O

OS operation system(운영 체제)

P

PA physical address(물리적 주소)

PCI peripheral component interconnect(주변 구성 요소 상호 연결 버스)

PCI-E PCI Express 버스

PCI-X PCI Extended 버스

PICL Platform Information and Control Library(플랫폼 정보 및 제어 라이브러리)

picld(1M) PICL daemon(PICL 데몬)

PRI priority(우선 순위)

R

RA real address(실제 주소)

RAID Redundant Array of Inexpensive Disk(저렴한 디스크의 중복 어레이)

RBAC Role-Based Access Control(역할별 접근 제어)

RPC Remote Procedure Call(원격 프로시저 호출)

S

SC system controller(시스템 제어기입니다. 시스템 프로세서와 동일합니다.)

SCSI Small Computer System Interface(소형 컴퓨터 시스템 인터페이스)

service domain

(서비스 도메인) 가상 스위치, 가상 콘솔 커넥터 및 가상 디스크 서버와 같은 장치를 다른 논리적 도메인에 제공하는 논리적 도메인입니다.

SMA System Management Agent(시스템 관리 에이전트)

SMF Service Management Facility(Solaris 10 OS의 서비스 관리 기능)

SNMP Simple Network Management Protocol(단순 네트워크 관리 프로토콜)

SP system processor(시스템 프로세서입니다. 시스템 제어기와 동일합니다.)

SSH Secure Shell(보안 셸)

ssh(1) Secure Shell command(보안 셸 명령)

sshd(1M) Secure Shell daemon(보안 셸 데몬)

SunVTS Sun Validation Test Suite(Sun 검증 테스트 제품군)

SVM Solaris Volume Manager입니다.

T

TCP Transmission Control Protocol(전송 제어 프로토콜)

U

UDP User Datagram Protocol(사용자 다이어그램 프로토콜)

USB Universal Serial Bus(범용 직렬 버스)

UTP unshielded twisted pair(비차폐 연선)

V

vBSC	virtual blade system controller(가상 블레이드 시스템 제어기)
vcc, vconscon	virtual console concentrator(게스트 도메인에 할당할 특정 포트 범위가 있는 가상 콘솔 집중 장치 서비스)
vcons, vconsole	virtual console(시스템 수준 메시지를 액세스하기 위한 가상 콘솔) 특정 포트에서 컨트롤 도메인의 vconscon 서비스로 연결하여 연결이 수행됩니다.
vcpu	virtual central processing unit(가상 중앙 처리 장치) 서버의 각 코어가 가상 CPU로 나타납니다. 예를 들어 8-코어 Sun Fire T2000 서버에는 논리적 도메인 간 할당할 수 있는 가상 CPU가 32개 있습니다.
vdc	virtual disk client(가상 디스크 클라이언트)
vdppcc	virtual data plane channel client(NDPS 환경의 가상 데이터 플레인 채널 클라이언트)
vdpcs	virtual data plane channel service(NDPS 환경의 가상 데이터 플레인 채널 서비스)
vdisk	virtual disk(가상 디스크는 물리적 장치, 볼륨 또는 파일의 서로 다른 유형에 의해 지원되는 일반 블록 장치입니다.)
vds, vdiskserver	virtual disk server(가상 디스크를 논리적 도메인으로 가져올 수 있는 가상 디스크 서버입니다.)
vdsdev, vdiskserverdevice	virtual disk server device(가상 디스크 서버에서 내보낸 가상 디스크 서버 장치입니다.) 장치는 전체 디스크, 디스크의 슬라이스, 파일 또는 디스크 볼륨일 수 있습니다.
vnet	가상 네트워크 스위치(vswitch)를 사용하여 시스템에서 다른 vnet 장치와 통신하고 가상 이더넷 장치를 구현하는 가상 네트워크(virtual network) 장치입니다.
vntsd(1M)	virtual network terminal server daemon(Logical Domains 콘솔용(Solaris 10 OS) 가상 네트워크 터미널 서버 데몬입니다.)
vsw, vswitch	virtual network switch(가상 네트워크 장치를 외부 네트워크에 연결하고 해당 장치 간의 패킷을 전환하는 가상 네트워크 스위치입니다.)
VTOC	volume table of contents(볼륨 목차)
VxVM	Veritas Volume Manager(VERITAS 볼륨 관리자)

W

WAN wide-area network(광역 네트워크)

X

XFP eXtreme Fast Path(가장 빠른 경로)

XML Extensible Markup Language(확장 마크업 언어)

Z

ZFS Zettabyte File System(Zettabyte 파일 시스템(Solaris 10 OS))

zpool(1M) ZFS storage pool(ZFS 저장소 풀)

