

Oracle® Solaris 영역 소개

ORACLE®

부품 번호: E54012-02
2014년 12월

Copyright © 2004, 2014, Oracle and/or its affiliates. All rights reserved.

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

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

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

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

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

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

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

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

목차

이 설명서 사용	7
1 Oracle Solaris 영역 소개	9
영역 개요	9
이 릴리스에서 지원되는 영역	10
변경 불가능한 영역	11
ipkg 영역을 solaris 영역으로 변환에 대한 정보	11
이 릴리스의 브랜드 영역	12
Oracle Solaris 커널 영역	12
기본 Oracle Solaris 영역	12
Oracle Solaris 10 영역	13
영역 브랜드 개요	13
Oracle Solaris Trusted Extensions 시스템에서 Oracle Solaris 영역 사 용	13
Oracle Solaris Cluster 영역 클러스터	14
브랜드 영역 정보	14
브랜드 영역에서 실행 중인 프로세스	15
영역을 사용해야 하는 경우	15
영역 작동 방식	17
기능별 영역 요약	18
비전역 영역이 관리되는 방식	19
비전역 영역이 생성되는 방식	19
비전역 영역 상태 모델	19
비전역 영역 특성	22
리소스 관리 기능을 비전역 영역과 함께 사용	23
영역 관련 SMF 서비스	23
비전역 영역 모니터링	24
비전역 영역에서 제공하는 기능	24
이 Oracle Solaris 영역 릴리스 정보	25
라이브 영역 재구성	28

2 비전역 영역 구성 개요	29
영역 내 리소스 정보	29
영역 관리 시 권한 프로파일과 역할 사용	29
zonecfg template 등록 정보	30
설치 전 구성 프로세스	30
영역 구성 요소	31
영역 이름 및 경로	31
영역 자동 부트	31
변경 불가능한 영역에 대한 file-mac-profile 등록 정보	31
admin 리소스	32
dedicated-cpu 리소스	32
solaris-kz만 해당: virtual-cpu 리소스	33
capped-cpu 리소스	34
예약 클래스	34
물리적 메모리 제어 및 capped-memory 리소스	35
solaris 및 solaris10만 해당:rootzpool 리소스	35
zpool 리소스 자동 추가	37
영역 네트워크 인터페이스	38
영역에서 마운트된 파일 시스템	43
파일 시스템 마운트 및 업데이트	44
영역의 호스트 ID	44
비전역 영역의 /dev 파일 시스템	44
비전역 영역의 이동식 lofi 장치	45
비전역 영역의 디스크 포맷 지원	45
저장소 URI를 사용하는 커널 영역 장치 리소스	46
구성 가능한 권한	46
리소스 풀 연결	46
영역 전체 리소스 제어 설정	47
영역에 대한 설명 포함	49
zonecfg 명령 사용	49
zonecfg 모드	50
zonecfg 대화식 모드	51
zonecfg 명령 파일 모드	53
영역 구성 데이터	53
리소스 유형과 등록 정보	53
리소스 유형 등록 정보	58
명령줄 편집 라이브러리	68
용어해설	69

색인 73

이 설명서 사용

- **개요** - 영역 기술 및 영역에서 사용 가능한 리소스에 대해 설명합니다.
- **대상** - 시스템 관리자, 기술자 및 공인 서비스 공급자
- **필요한 지식** - 네트워크 구성 및 리소스 할당 지식을 비롯한 Oracle Solaris 운영 체제 사용 경험이 필요합니다.

제품 설명서 라이브러리

이 제품에 대한 최신 정보 및 알려진 문제는 설명서 라이브러리(<http://www.oracle.com/pls/topic/lookup?ctx=E56343>)에서 확인할 수 있습니다.

Oracle 지원 액세스

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

피드백

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

Oracle Solaris 영역 소개

Oracle Solaris 운영 체제의 Oracle™ Solaris 영역 기능은 시스템에서 응용 프로그램을 실행할 분리된 환경을 제공합니다.

이 장에서는 영역에 대해 개괄적으로 설명합니다.

이 장에서는 다음과 같은 일반적인 영역의 내용을 다룹니다.

- “영역 개요” [9]
- “이 릴리스에서 지원되는 영역” [10]
- “이 Oracle Solaris 영역 릴리스 정보” [25]
- “변경 불가능한 영역” [11]
- “ipkg 영역을 solaris 영역으로 변환에 대한 정보” [11]
- “영역을 사용해야 하는 경우” [15]
- “영역 작동 방식” [17]
- “비전역 영역에서 제공하는 기능” [24]
- “이 릴리스의 브랜드 영역” [12]

다음 장에서는 영역 구성 리소스 및 등록 정보에 대해 설명합니다.

시스템에 영역을 만들 준비가 된 경우 “Oracle Solaris 커널 영역 만들기 및 사용” 및 “Oracle Solaris 영역 만들기 및 사용”으로 건너뛰십시오. Oracle Solaris 10을 실행하는 시스템(해당 시스템에 비전역 native 영역이 포함될 수 있음)을 Oracle Solaris 11 시스템의 영역으로 마이그레이션할 준비가 된 경우, “Oracle Solaris 10 영역 만들기 및 사용”을 참조하십시오.

참고 - Oracle Solaris Trusted Extensions 시스템에서의 영역 사용에 대한 자세한 내용은 “Trusted Extensions 구성 및 관리”의 13 장, “Trusted Extensions에서 영역 관리”를 참조하십시오.

영역 개요

Oracle Solaris 영역 분할 기술은 운영 체제 서비스를 가상화하고 실행 중인 응용 프로그램을 위해 안전한 격리 환경을 제공하는 데 사용됩니다. 영역이라고 하는 비전역 영역은 Oracle

Solaris 운영 체제의 단일 인스턴스 내에서 만들어진 가상 운영 체제 환경입니다 운영 체제의 인스턴스를 전역 영역이라고 합니다. Oracle Solaris 커널 영역에서는 호스트 버전과 다른 SRU(Support Repository Update) 또는 커널 버전을 실행할 수 있습니다.

가상화의 목표는 개별 데이터 센터 구성 요소 관리에서 리소스 풀 관리로 이동하는 것입니다. 서버 가상화를 성공적으로 수행하면 서버 활용을 개선하고 서버 자산을 더욱 효율적으로 사용할 수 있습니다. 서버 가상화는 개별 시스템의 격리를 유지 관리하는 성공적인 서버 통합 프로젝트에도 중요합니다.

가상화는 단일 시스템에서 여러 호스트와 서비스를 통합해야 하는 필요에 따라 수행됩니다. 가상화는 하드웨어, 기반구조 및 관리를 공유하여 비용을 축소합니다. 혜택은 다음과 같습니다.

- 하드웨어 활용 증가
- 리소스 할당의 유연성 향상
- 전원 요구 사항 축소
- 관리 비용 감소
- 소유 비용 감소
- 시스템에서 응용 프로그램 간에 관리 및 리소스 경계 설정

영역을 만들면 프로세스를 나머지 시스템으로부터 격리시키는 응용 프로그램 실행 환경이 만들어집니다. 이러한 분리는 하나의 영역에서 실행되는 프로세스가 다른 영역에서 실행되는 프로세스를 모니터링하거나 영향을 미치는 것을 방지합니다. 루트 자격 증명으로 실행되는 프로세스라도 다른 영역의 작업을 보거나 영향을 미칠 수 없습니다. Oracle Solaris 영역을 사용하면 하드웨어 리소스를 동시에 공유하면서 서버 배포 모델당 하나의 응용 프로그램을 유지할 수 있습니다.

또한, 영역은 응용 프로그램과 응용 프로그램이 배포된 시스템의 물리적 속성을 분리하는 추상 계층을 제공합니다. 이러한 속성의 예로는 물리적 장치 경로가 있습니다.

영역은 Oracle Solaris 10 또는 Oracle Solaris 11 릴리스를 실행 중인 모든 시스템에서 영역을 사용할 수 있습니다. 시스템에 있는 solaris 및 solaris10 영역 수에 대한 상한은 8192입니다. 단일 시스템에 효과적으로 호스팅할 수 있는 영역 수는 시스템의 모든 영역과 크기에서 실행 중인 응용 프로그램 소프트웨어의 전체 리소스 요구 사항에 따라 결정됩니다. 이러한 개념은 [“Oracle Solaris 영역 만들기 및 사용”](#)의 1 장, [“비전역 영역을 계획 및 구성하는 방법”](#)에 설명되어 있습니다.

Oracle Solaris 커널 영역을 실행 중인 경우 이러한 개념에 대한 자세한 내용은 [“Oracle Solaris 커널 영역 만들기 및 사용”](#)의 [“Oracle Solaris 커널 영역에 대한 하드웨어 및 소프트웨어 요구 사항”](#)을 참조하십시오.

이 릴리스에서 지원되는 영역

단일 호스트 전역 영역 내에서 실행되는 solaris 및 solaris10 비전역 브랜드 영역은 Oracle Solaris 11.2 릴리스가 지원되는 플랫폼으로 정의된 모든 아키텍처에서 지원됩니다.

Oracle Solaris 커널 영역은 T4+ 및 M5+ SPARC 시스템, Nehalem+ Intel 시스템 및 Barcelona+ AMD 시스템에서 실행될 수 있습니다. 커널 영역 시스템 요구 사항에 대한 자세한 내용은 “Oracle Solaris 커널 영역 만들기 및 사용”의 “Oracle Solaris 커널 영역에 대한 하드웨어 및 소프트웨어 요구 사항”을 참조하십시오.

변경 불가능한 영역

변경 불가능한 영역은 읽기 전용 루트를 사용하는 solaris 영역입니다. 전역 영역과 비전역 영역은 변경 불가능한 영역일 수 있습니다. 읽기 전용 영역은 file-mac-profile 등록 정보를 설정하여 구성할 수 있습니다. 몇 가지 구성을 사용할 수 있습니다. 읽기 전용 영역 루트는 보안 런타임 경계를 확장합니다.

Oracle Solaris 전역 변경 불가능한 영역은 변경 불가능한 영역 기능을 전역 영역까지 확장했습니다. 변경 불가능한 영역 및 변경 불가능한 커널 영역의 경우 zlogin 명령 `zlogin(1)`을 통해 신뢰할 수 있는 경로 로그인을 호출할 수 있습니다.

`zonecfg add dataset`을 사용하여 추가 데이터 세트가 지정되는 영역에는 여전히 이러한 데이터 세트에 대한 모든 권한이 부여됩니다. `zonecfg add fs`를 사용하여 추가 파일 시스템이 지정되는 영역에는 파일 시스템이 읽기 전용으로 설정된 경우가 아니면 이러한 파일 시스템에 대한 모든 권한이 부여됩니다.

자세한 내용은 “Oracle Solaris 영역 만들기 및 사용”의 12 장, “변경할 수 없는 영역 구성 및 관리”를 참조하십시오.

ipkg 영역을 solaris 영역으로 변환에 대한 정보

Oracle Solaris 11 Express 릴리스 고객을 지원하기 위해 ipkg 영역으로 구성된 영역이 solaris 영역으로 변환되고 Oracle Solaris 11.2로의 zoneadm attach 또는 pkg 업데이트 시 solaris로 보고됩니다. ipkg 이름은 영역을 구성할 때 사용된 경우 solaris 이름으로 매핑됩니다. Oracle Solaris 11 Express 호스트에서 내보낸 zonecfg 파일의 가져오기가 지원됩니다.

`zonecfg info` 또는 `zoneadm list -v` 등의 명령 출력은 Oracle Solaris 11.2 시스템에서 기본 영역의 solaris 브랜드를 표시합니다.

이 릴리스의 브랜드 영역

Oracle Solaris 커널 영역

Oracle Solaris 커널 영역 기능은 영역 내에서 완전한 커널 및 사용자 환경을 제공하며 호스트와 영역 간 커널 분리도 향상시킵니다. 브랜드 이름은 `solaris-kz`입니다. 커널 영역은 기존 도구인 `zonecfg`, `zoneadm` 및 `zlogin`을 사용하여 전역 영역에서 관리할 수 있습니다. 커널 영역 관리자는 기본 `solaris` 영역 관리자보다 더 유연성 있게 영역을 구성하고 관리할 수 있습니다. 예를 들어 전역 영역에 설치된 패키지로 국한되지 않고, 커널 버전을 비롯한 영역의 설치된 패키지를 완전히 업데이트하고 수정할 수 있습니다. 영역 전용 저장소를 관리하고 ZFS 풀을 만들고 삭제하며 iSCSI 및 CIFS를 구성할 수 있습니다. 커널 영역 내에 `solaris` 영역을 설치하여 계층형(중첩) 영역을 생성할 수 있습니다. 커널 영역은 일시 중지 및 다시 시작을 지원합니다. 소스 시스템에 있는 영역을 일시 중지한 다음 대상 시스템에 있는 영역을 다시 시작하는 방식으로 커널 영역을 마이그레이션할 수 있습니다.

Oracle Solaris 커널 영역을 사용하려면 `brand-solaris-kz` 패키지를 시스템에 설치해야 합니다. 사용 중인 시스템에서 커널 영역을 지원하는지 여부를 확인하려면 [“이 Oracle Solaris 영역 릴리스 정보” \[25\]](#)를 참조하십시오. Oracle Solaris 11.2가 설치된 경우 시스템에서 `virtinfo` 명령을 실행할 수도 있습니다. Oracle Solaris 커널 영역에 대한 자세한 내용은 [“Oracle Solaris 커널 영역 만들기 및 사용”](#) 및 `solaris-kz(5)` 매뉴얼 페이지를 참조하십시오. `virtinfo` 명령에 대한 자세한 내용은 [“Oracle Solaris 커널 영역 만들기 및 사용”](#)의 [“호스트에서 커널 영역 지원을 확인하는 방법”](#) 및 `virtinfo(1M)` 매뉴얼 페이지를 참조하십시오.

기본 Oracle Solaris 영역

Oracle Solaris 영역 기능은 응용 프로그램에 제공되는 완벽한 런타임 환경입니다. 영역은 응용 프로그램에서 플랫폼 리소스로의 가상 매핑을 제공합니다. 영역을 사용하면 영역에서 Oracle Solaris 운영 체제의 단일 인스턴스를 공유하는 경우라도 응용 프로그램 구성 요소를 서로 격리할 수 있습니다. 영역은 리소스 관리 구성 요소를 사용하여 응용 프로그램에서 사용할 가능한 시스템 리소스를 사용하는 방식을 제어합니다. 리소스 관리 기능에 대한 자세한 내용은 [“Oracle Solaris 11.2의 리소스 관리”](#)를 참조하십시오.

영역은 CPU 등의 리소스 소비에 대한 경계를 설정합니다. 이러한 경계는 영역에서 실행 중인 응용 프로그램의 변경되는 처리 요구 사항에 맞도록 확장될 수 있습니다.

추가 격리를 위해 읽기 전용 루트가 있는 영역(변경할 수 없는 영역이라고 함)을 구성할 수 있습니다.

Oracle Solaris 10 영역

solaris10 브랜드 비전역 영역이라고도 하는 Oracle Solaris 10 영역은 BrandZ 기술을 사용하여 Oracle Solaris 11 운영 체제에서 Oracle Solaris 10 응용 프로그램을 실행합니다. 즉, 응용 프로그램을 비전역 영역에서 제공한 안전한 환경에서 수정 없이 실행할 수 있습니다. 이로 인해 Oracle Solaris 10 시스템을 사용하여 응용 프로그램을 개발, 테스트 및 배치할 수 있습니다. 이러한 브랜드 영역 내에서 실행되는 작업 부하는 Oracle Solaris 11 릴리스에서만 사용할 수 있는 커널에 대한 향상된 기능을 활용하고 혁신적인 일부 기술을 사용할 수 있습니다. 이러한 영역은 Oracle Solaris 10 시스템을 Solaris 11의 영역으로 마이그레이션하는 데 사용됩니다. solaris10 브랜드 영역은 NFS 서버일 수 없습니다.

자세한 내용은 [“Oracle Solaris 10 영역 만들기 및 사용”](#)을 참조하십시오.

영역 브랜드 개요

solaris-kz 영역과 solaris 및 solaris10 브랜드 영역 간의 차이는 아래에 나와 있습니다.

표 1-1 Oracle Solaris 영역 브랜드 기능 비교

구성 요소	solaris-kz 브랜드	solaris 및 solaris10 브랜드
지원되는 하드웨어	지정된 하드웨어에서 지원됩니다. OTN 사이트 또는 HCL에 대한 링크를 참조하십시오.	모든 Oracle Solaris 11.2 시스템에서 지원됩니다. HCL을 참조하십시오.
메모리 관리	고정된 양의 물리적 RAM을 solaris-kz 가상 플랫폼에 할당해야 합니다.	전역 영역에 할당된 물리적 RAM을 공유할 수 있습니다.
커널 버전	커널 영역에서는 호스트와 다른 커널 버전 또는 SRU 레벨을 실행할 수 있습니다.	커널 버전이 전역 영역의 버전과 같아야 합니다.
저장소 및 장치 관리	모든 저장소 액세스를 수행합니다. 커널 영역에서 zpool 또는 rootzpool 리소스를 지원하지 않습니다.	fs, zpool 및 dataset zonecfg 리소스를 통해 파일 시스템 레벨에서 저장소를 사용할 수 있습니다.
네트워킹	배타적 IP 영역만 지원됩니다.	배타적 IP 영역과 공유 IP 영역이 지원됩니다.

Oracle Solaris Trusted Extensions 시스템에서 Oracle Solaris 영역 사용

Oracle Solaris Trusted Extensions 시스템에서의 영역 사용에 대한 자세한 내용은 [“Trusted Extensions 구성 및 관리”](#)의 13 장, [“Trusted Extensions에서 영역 관리”](#)를 참조하십시오. labeled 브랜드만 Oracle Solaris Trusted Extensions 시스템에서 부트될 수 있습니다.

Oracle Solaris Cluster 영역 클러스터

영역 클러스터는 Oracle Solaris Cluster 소프트웨어의 기능입니다. 영역 클러스터는 영역 클러스터 노드로 사용되는 비전역 영역의 그룹입니다. 영역 클러스터로 구성된 각 전역 클러스터 노드에 하나의 비전역 영역이 만들어집니다. 영역 클러스터 노드는 solaris 브랜드와 solaris10 브랜드 중 하나일 수 있으며 클러스터 속성을 사용합니다. 다른 브랜드 유형은 허용되지 않습니다. 영역에서 제공되는 격리를 사용하여 전역 클러스터에서와 같은 방식으로 영역 클러스터에서 지원되는 서비스를 실행할 수 있습니다. 자세한 내용은 [“Oracle Solaris Cluster 시스템 관리 설명서”](#)를 참조하십시오.

브랜드 영역 정보

기본적으로 시스템의 비전역 영역은 전역 영역과 동일한 운영 체제 소프트웨어를 실행합니다. Oracle Solaris 운영 체제의 브랜드 영역(BrandZ) 기능은 Oracle Solaris 영역을 단순히 확장한 것입니다. BrandZ 프레임워크는 전역 영역과는 다른 운영 환경을 포함하는 비전역 영역 브랜드 영역을 만드는 데 사용됩니다. 브랜드 영역은 Oracle Solaris 운영 체제에서 응용 프로그램을 실행하는 데 사용됩니다. BrandZ 프레임워크는 다양한 방식으로 Oracle Solaris 영역 기반구조를 확장합니다. 이러한 확장은 복잡할 수도 있고(예: 영역 내 다양한 운영 체제 환경을 실행할 수 있는 기능 제공) 간단할 수도 있습니다(예: 기본 영역 명령을 항상 시켜 새 기능 제공). 예를 들어 Oracle Solaris 10 영역은 Oracle Solaris 10 운영 체제를 에뮬레이트할 수 있는 브랜드 비전역 영역입니다. 전역 영역과 동일한 운영 체제를 공유하는 기본 영역조차 브랜드로 구성됩니다.

브랜드는 영역 안에 설치할 수 있는 운영 환경을 정의하며, 영역에 설치된 소프트웨어가 올바르게 작동하도록 영역 내에서 시스템의 작동 방식을 결정합니다. 또한, 영역의 브랜드는 응용 프로그램 시작 시 올바른 응용 프로그램 유형을 식별하는 데 사용됩니다. 모든 브랜드 영역 관리는 표준 영역 구조에 대한 확장을 통해 수행됩니다. 대부분의 관리 절차는 모든 영역에 대해 동일합니다.

정의된 파일 시스템과 권한 등 기본적으로 구성에 포함된 리소스는 브랜드 관련 문서에 설명되어 있습니다.

BrandZ는 다음과 같은 방식으로 영역 도구를 확장합니다.

- zonecfg 명령은 영역이 구성될 때 영역의 브랜드 유형을 설정하는 데 사용됩니다.
- zoneadm 명령은 영역의 브랜드 유형을 보고하고 영역을 관리하는 데 사용됩니다.

라벨이 사용으로 설정된 Oracle Solaris Trusted Extensions 시스템에서 브랜드 영역을 구성 및 설치할 수 있다고 해도, 부트되는 브랜드가 인증된 시스템 구성의 labeled 브랜드가 아니면 이 시스템 구성에서 브랜드 영역을 부트할 수 없습니다.

영역의 브랜드를 구성됨 상태로 변경할 수 있습니다. 브랜드 영역이 설치되고 나면 브랜드를 변경하거나 제거할 수 없습니다.



주의 - 기존 Oracle Solaris 10 시스템을 Oracle Solaris 11 릴리스를 실행하는 solaris10 브랜드 영역으로 마이그레이션하려는 경우 먼저 기존 영역을 대상 시스템으로 마이그레이션해야 합니다. solaris10 영역은 중첩되지 않으므로 시스템 마이그레이션 프로세스에서 기존 영역이 사용할 수 없는 상태로 표시됩니다. 자세한 내용은 “Oracle Solaris 10 영역 만들기 및 사용”의 3 장, “Oracle Solaris 10 native 비전역 영역을 Oracle Solaris 10 영역으로 마이그레이션”을 참조하십시오.

브랜드 영역에서 실행 중인 프로세스

브랜드 영역은 커널에 브랜드 영역에서 실행 중인 프로세스에만 적용되는 삽입점을 제공합니다.

- 이러한 지점은 `syscall` 경로, 프로세스 로딩 경로 및 스레드 생성 경로 등에서 발견됩니다.
- 이러한 각각의 지점에서 브랜드는 표준 Oracle Solaris 동작을 보완하거나 대체하도록 선택할 수 있습니다.

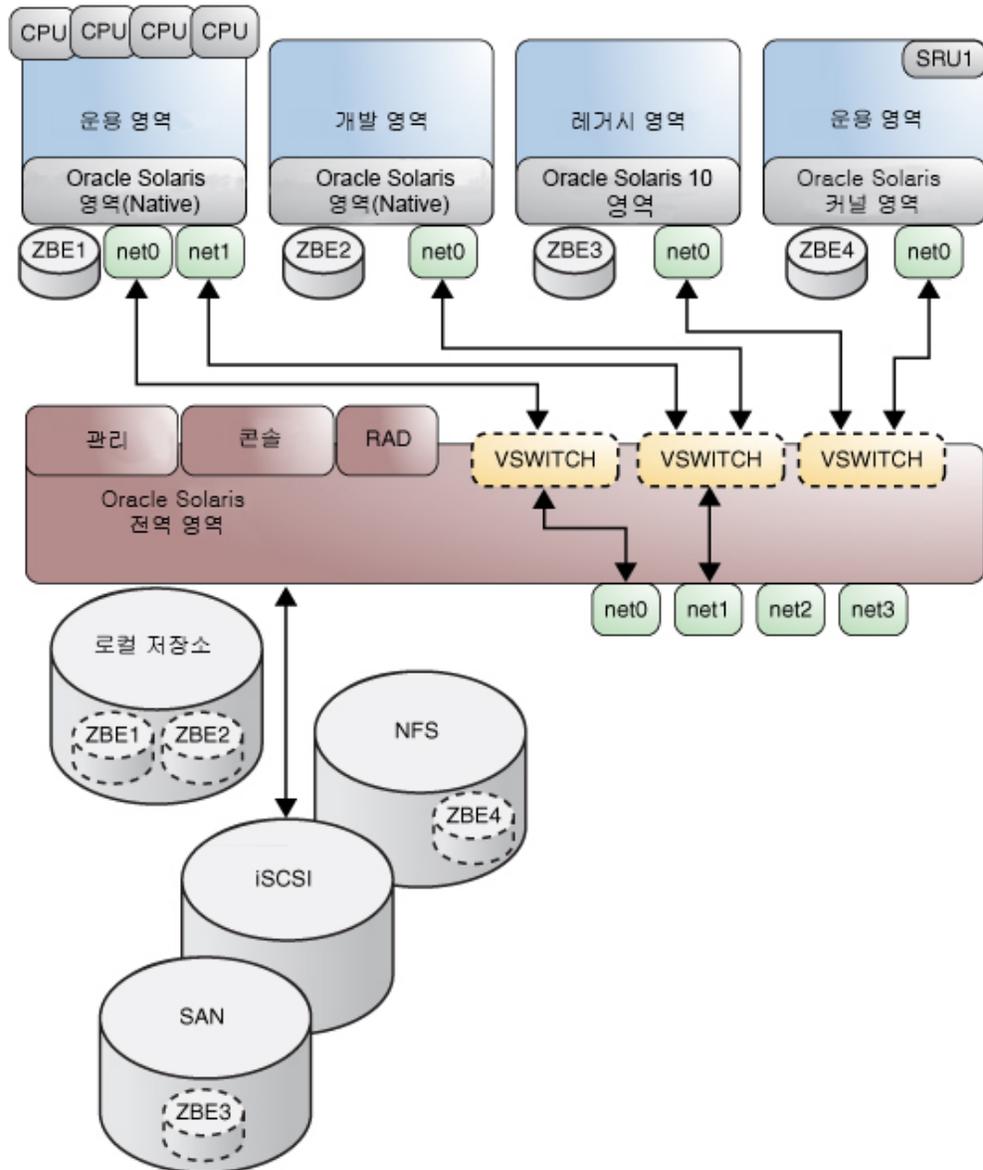
또한 브랜드는 `librtld_db`에 대한 플러그인 라이브러리를 제공할 수도 있습니다. 이 플러그인 라이브러리를 사용하면 `mdb(1)`에 설명된 디버거나 `dtrace(1M)`에 설명된 DTrace와 같은 Oracle Solaris 도구를 통해 브랜드 영역 내부에서 실행되는 프로세스의 기호 정보에 액세스할 수 있습니다.

영역은 정적으로 링크된 바이너리를 지원하지 않습니다.

영역을 사용해야 하는 경우

영역은 단일 서버에 여러 응용 프로그램을 통합하는 환경에 이상적입니다. 다양한 기기를 관리하는 데 드는 비용과 복잡성 문제로 인해 더 크고 확장성이 더 뛰어난 서버에 여러 응용 프로그램을 통합하는 것이 유리합니다.

그림 1-1 영역 서버 통합 예



영역을 사용하면 시스템에서 리소스를 보다 효율적으로 활용할 수 있습니다. 동적 리소스 재할당을 통해 필요에 따라 사용되지 않는 리소스를 이동할 수 있습니다. 결함 및 보안 격리란

잘못 작동하는 응용 프로그램에 전용 시스템 및 활용률이 낮은 시스템이 필요하지 않음을 의미합니다. 영역을 사용하면 이러한 응용 프로그램을 다른 응용 프로그램과 통합할 수 있습니다.

영역을 사용하면 전체 시스템 보안을 유지하면서 특정 관리 기능을 위임할 수 있습니다.

영역 작동 방식

비전역 영역은 하나의 상자로 생각할 수 있습니다. 하나 이상의 응용 프로그램이 시스템의 나머지와 상호 작용 없이 이 상자에서 실행될 수 있습니다. 영역은 소프트웨어에 의해 정의된 유연한 경계를 사용하여 소프트웨어 응용 프로그램이나 서비스를 격리합니다. 그러면 Oracle Solaris 운영 체제의 동일 인스턴스에서 실행 중인 응용 프로그램을 다른 응용 프로그램과 독립적으로 관리할 수 있습니다. 따라서 서로 다른 버전의 동일 응용 프로그램을 서로 다른 영역에서 실행하며 구성 요구 사항을 충족할 수 있습니다.

영역에 지정된 프로세스는 동일 영역에 지정된 다른 프로세스와 직접 통신하고 이러한 프로세스를 조작하며 모니터링할 수 있습니다. 이 프로세스는 시스템의 다른 영역에 지정된 프로세스와 함께 또는 영역에 지정되지 않은 프로세스와 함께 이러한 기능을 수행할 수 있습니다. 다양한 영역에 지정된 프로세스는 네트워크 API를 통해서만 통신할 수 있습니다.

영역에 고유의 배타적 IP 인스턴스가 지정되었는지 아니면 전역 영역과 IP 계층 구성 및 상태를 공유하는지에 따라 두 가지 방법으로 IP 네트워킹을 구성할 수 있습니다. 배타적 IP가 기본 유형입니다. 영역의 IP 유형에 대한 자세한 내용은 “[영역 네트워크 인터페이스](#)” [38]를 참조하십시오. 자세한 내용은 “[Oracle Solaris 영역 만들기 및 사용](#)”의 “[영역 구성 방법](#)”을 참조하십시오.

모든 Oracle Solaris 시스템에는 전역 영역이 있습니다. 전역 영역에는 두 가지 기능이 있습니다. 전역 영역은 시스템의 기본 영역이며 또한 시스템 전체의 관리 제어용으로 사용되는 영역입니다. 비전역 영역(간단히 영역이라고 함)이 없는 경우 전역 영역에서 실행되는 모든 프로세스는 전역 관리자 또는 영역 보안 프로파일을 사용하는 사용자가 만듭니다.

전역 영역은 비전역 영역을 구성, 설치, 관리 또는 제거할 수 있는 유일한 영역입니다. 오직 전역 영역만 시스템 하드웨어에서 부트할 수 있습니다. 물리적 장치, 공유 IP 영역의 경로 지정 또는 DR(동적 재구성)과 같은 시스템 기반구조의 관리는 물리적 시스템에서 실행되는 전역 영역에서만 가능합니다. 전역 영역에서 적절한 권한으로 실행되는 프로세스는 다른 영역에 연결된 객체에 액세스할 수 있습니다.

이 경우 전역 영역의 권한이 없는 프로세스는 비전역 영역의 권한이 부여된 프로세스에 허용되지 않는 작업을 수행할 수 있습니다. 예를 들면 전역 영역의 사용자는 시스템의 모든 프로세스에 대한 정보를 볼 수 있습니다. 이 기능이 사이트에 문제를 야기시키는 경우에는 전역 영역에 대한 액세스를 제한할 수 있습니다.

전역 영역을 비롯한 각각의 영역에는 영역 이름이 지정됩니다. 전역 영역에는 항상 `global` 이름이 할당됩니다. 각 영역에도 고유한 숫자 식별자가 지정되는데, 이 식별자는 영역이 부트될 때 시스템에서 지정됩니다. 전역 영역은 항상 ID 0에 매핑됩니다. 커널 영역에 `zlogin`한 경우

커널 영역은 가상 전역 영역이므로 ID 0을 갖는 것으로 보고됩니다. 영역 이름 및 숫자 ID는 “Oracle Solaris 영역 만들기 및 사용”의 “영역 구성 방법”에 설명되어 있습니다.

각 영역에는 영역 이름과 완전히 독립된 노드 이름도 제공됩니다. 노드 이름은 영역의 관리자가 지정합니다. 자세한 내용은 “Oracle Solaris 영역 만들기 및 사용”의 “비전역 영역 노드 이름”을 참조하십시오.

각 영역에는 전역 영역의 루트 디렉토리를 기준으로, 해당 루트 디렉토리에 대한 경로가 있습니다. 자세한 내용은 “zonecfg 명령 사용” [49]을 참조하십시오.

비전역 영역의 예약 클래스는 기본적으로 시스템의 예약 클래스로 설정됩니다. 영역의 예약 클래스를 설정하는 데 사용되는 방법에 대한 설명은 “예약 클래스” [34]를 참조하십시오.

블록 장치 다중 경로는 `scsi_vhci(7D)`를 통해 처리됩니다. 구성에 대해 선택한 `lu:storage` URI의 형식에 따라 구성 사용 방식이 결정됩니다. 다중 경로를 통한 `lu:URI` 사용에 대한 자세한 내용은 `suri(5)` 매뉴얼 페이지를 참조하십시오.

기능별 영역 요약

다음 표에는 전역 및 비전역 영역의 특성이 요약되어 있습니다.

영역 유형	특성
전역	<ul style="list-style-type: none"> ■ 시스템에서 ID 0을 지정합니다. ■ 부트 가능하며 시스템에서 실행 중인 Oracle Solaris의 단일 인스턴스를 제공합니다. ■ Oracle Solaris 시스템 소프트웨어 패키지의 전체 설치를 포함합니다. ■ 추가 소프트웨어 패키지 또는 패키지를 통해 설치되지 않은 추가 소프트웨어, 디렉토리, 파일 및 기타 데이터를 포함할 수 있습니다. ■ 전역 영역에 설치된 모든 소프트웨어 구성 요소에 대한 정보를 포함하는 완전하고 일관된 전체 제품 데이터베이스를 제공합니다. ■ 전역 영역 호스트 이름 및 파일 시스템 테이블 등 전역 영역에만 국한된 구성 정보를 보유합니다. ■ 모든 장치 및 모든 파일 시스템을 인식하는 유일한 영역입니다. ■ 비전역 영역 존재 여부와 구성을 알고 있는 유일한 영역입니다. ■ 비전역 영역을 구성, 설치, 관리 또는 제거할 수 있는 유일한 영역입니다.
비전역	<ul style="list-style-type: none"> ■ 영역이 부트될 때 시스템에서 영역 ID를 지정합니다. ■ 전역 영역에서 부트된 Oracle Solaris 커널 아래에서 작업을 공유합니다. ■ 전체 Oracle Solaris 운영 체제 소프트웨어 패키지 중 설치된 일부 소프트웨어를 포함합니다. ■ 추가로 설치된 소프트웨어 패키지를 포함할 수 있습니다. ■ 패키지를 통해 설치되지 않는 비전역 영역에 생성된 추가 소프트웨어, 디렉토리, 파일 및 기타 데이터를 포함할 수 있습니다. ■ 영역에 설치된 모든 소프트웨어 구성 요소에 대한 정보를 포함하는 완전하고 일관된 전체 제품 데이터베이스가 있습니다. ■ 다른 영역의 존재 여부를 인식하지 못합니다.

영역 유형	특성
	<ul style="list-style-type: none"> ■ 자체 영역을 비롯하여 다른 영역을 설치, 관리 또는 제거할 수 없습니다. ■ 비전역 영역 호스트 이름 및 파일 시스템 테이블 등 비전역 영역에만 국한된 구성 정보를 보유합니다. ■ 고유의 표준 시간대 설정이 있을 수 있습니다.

비전역 영역이 관리되는 방식

영역 관리자에게는 슈퍼 유저 권한 또는 이에 상응하는 관리 권한이 부여됩니다. 전역 영역에 로그인한 경우 전역 관리자는 시스템 전체를 모니터링하고 제어할 수 있습니다.

비전역 영역은 영역 관리자가 관리할 수 있습니다. 전역 관리자는 [“admin 리소스” \[32\]](#)에 설명된 대로 영역 관리자에게 필수 권한을 지정합니다. 영역 관리자의 권한은 특정 비전역 영역으로 제한됩니다.

비전역 영역이 생성되는 방식

AI(자동 설치) 클라이언트 설치의 일부로서 비전역 영역의 구성 및 설치를 지정할 수 있습니다. 자세한 내용은 [“Oracle Solaris 11.2 시스템 설치”](#)를 참조하십시오. Oracle Solaris 커널 영역은 주로 직접 설치 방식을 사용하여 만들 수 있습니다. 커널 영역 만들기 방식은 [“Oracle Solaris 커널 영역 만들기 및 사용”](#)의 [“커널 영역 설치”](#)에 설명되어 있습니다.

Oracle Solaris 시스템에서 영역을 만들려면 영역 관리자는 `zonecfg` 명령을 사용하여 영역의 가상 플랫폼과 응용 프로그램 환경에 대한 다양한 매개변수를 지정함으로써 영역을 구성할 수 있습니다. 그리고 나면 전역 관리자가 영역을 설치합니다. 전역 관리자는 영역 관리 명령인 `zoneadm`을 사용하여 패키지 레벨의 소프트웨어를 영역에 대해 설정된 파일 시스템 계층에 설치합니다. `zoneadm` 명령은 영역을 부트하는 데 사용됩니다. 그리고 나면 전역 관리자나 권한이 부여된 사용자가 `zlogin` 명령을 사용하여 설치된 영역에 로그인할 수 있습니다. RBAC(역할 기반 액세스 제어)를 사용 중인 경우 영역 관리자에게 `solaris.zone.manage/zoneName` 권한이 부여되어야 합니다.

영역 구성에 대한 자세한 내용은 [2장. 비전역 영역 구성 개요](#)를 참조하십시오. 영역 설치에 대한 자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”](#)의 [2 장, “비전역 영역, 설치, 종료, 정지 및 복제 정보”](#)를 참조하십시오. 영역 로그인에 대한 자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”](#)의 [4 장, “비전역 영역 로그인 정보”](#)를 참조하십시오.

Oracle Solaris 커널 영역을 구성하고 설치하려면 [“Oracle Solaris 커널 영역 만들기 및 사용”](#)을 참조하십시오.

비전역 영역 상태 모델

비전역 영역은 다음 7가지 상태 중 하나일 수 있습니다.

구성됨	영역의 구성이 완료되고 안정된 저장소로 연결됩니다. 하지만 영역의 응용 프로그램 환경에서 초기 부트 후에 지정해야 하는 요소는 아직 존재하지 않습니다.
불완전	설치 또는 제거 작업 중에 zoneadm은 대상 영역의 상태를 불완전으로 설정합니다. 작업이 완료되면 상태가 올바른 상태로 설정됩니다. zoneadm의 하위 명령인 mark를 사용하여 손상된 설치 영역을 불완전 상태로 표시할 수 있습니다. 불완전 상태의 영역이 zoneadm list - iv의 출력에 표시됩니다.
사용할 수 없음	영역이 설치되었지만 확인, 준비, 부트, 연결 또는 이동할 수 없음을 나타냅니다. 다음과 같은 경우에 영역이 사용할 수 없음 상태로 전환됩니다. <ul style="list-style-type: none"> ■ 영역의 저장소를 사용할 수 없고 svc:/system/zones:default가 시작될 때(예: 시스템이 부트하는 동안) ■ 영역의 저장소를 사용할 수 없는 경우 ■ 성공적인 아카이브 추출 후 아카이브 기반 설치가 실패하는 경우 ■ 영역의 소프트웨어가 전역 영역의 소프트웨어와 호환되지 않는 경우(예: 부적절한 -f(강제) 연결 후)
설치됨	영역의 구성이 시스템에서 인스턴스화됩니다. zoneadm 명령은 지정된 Oracle Solaris 시스템에서 구성을 정상적으로 사용할 수 있음을 확인하는 데 사용됩니다. 패키지는 영역의 루트 경로 아래에 설치됩니다. 이 상태에서는 영역에 연결된 가상 플랫폼이 없습니다.
준비	영역의 가상 플랫폼이 설정됩니다. 커널이 zsched 프로세스를 만들고, 네트워크 인터페이스가 설정되어 영역에서 사용할 수 있게 되며, 파일 시스템이 마운트되고 장치가 구성됩니다. 고유의 영역 ID는 시스템에서 지정합니다. 이 단계에서는 영역과 연결된 프로세스가 시작되지 않았습니다.
실행 중	영역 응용 프로그램 환경과 연결된 사용자 프로세스가 실행 중입니다. 응용 프로그램 환경과 연결된 첫번째 사용자 프로세스(init)가 만들어지자마자 영역이 실행 중 상태로 들어갑니다.
종료 및 작동 중지	이러한 상태는 영역이 정지되고 있는 동안 표시되는 중간 상태입니다. 하지만 어떤 이유로든 종료할 수 없는 영역은 이러한 상태 중 하나로 중지됩니다.

“Oracle Solaris 영역 만들기 및 사용”의 3 장, “비전역 영역 설치, 부트, 종료, 정지, 제거 및 복제” 및 zoneadm(1M) 매뉴얼 페이지에서는 zoneadm 명령을 사용하여 이러한 상태 간 전환을 시작하는 방법에 대해 설명합니다.

또한 Oracle Solaris 커널 영역은 3개의 보조 상태를 갖습니다. 이러한 상태는 현재 영역 상태에 대한 추가 정보와 함께 호스트에 알리는 데 사용됩니다.

일시 중지됨	기본 상태는 중지됨이고, 보조 상태는 일시 중지됨입니다.
디버깅	영역이 실행 중이지만, 네트워킹과 같은 외부 이벤트에 응답할 수 없습니다. <code>zlogin</code> 은 이 상태를 확인한 다음 <code>zlogin</code> 세션을 시작하기 전에 해당 상태가 해제될 때까지 기다립니다.
패닉	영역이 패닉 상태지만, 재부트될 때까지 영역이 외부 이벤트에 응답할 수 없습니다.

자세한 내용은 “Oracle Solaris 커널 영역 만들기 및 사용” 및 `solaris-kz(5)` 매뉴얼 페이지를 참조하십시오.

표 1-2 영역 상태에 영향을 주는 명령

현재 영역 상태	해당되는 명령
구성됨	<pre>zonecfg -z <i>zonename</i> verify zonecfg -z <i>zonename</i> commit zonecfg -z <i>zonename</i> delete zoneadm -z <i>zonename</i> attach zoneadm -z <i>zonename</i> verify zoneadm -z <i>zonename</i> install zoneadm -z <i>zonename</i> clone zoneadm -z <i>zonename</i> mark <i>incomplete</i> zoneadm -z <i>zonename</i> mark <i>unavailable</i></pre> <p><code>zonecfg</code> 명령을 사용하여 구성됨 상태에 있는 영역의 이름을 바꿀 수도 있습니다. <code>zoneadm</code> 명령을 사용하여 구성됨 또는 설치됨 상태에 있는 Oracle Solaris 영역 또는 Oracle Solaris 10 영역의 이름을 바꿀 수 있습니다.</p>
불완전	<pre>zoneadm -z <i>zonename</i> uninstall</pre>
사용할 수 없음	<pre>zoneadm -z <i>zonename</i> uninstall</pre> 은 지정된 시스템에서 영역을 제거합니다. <pre>zoneadm -z <i>zonename</i> attach</pre> <p><code>zonecfg -z <i>zonename</i></code>은 <code>zonepath</code> 및 설치됨 상태일 때 변경할 수 없는 다른 등록 정보나 리소스를 변경하는 데 사용할 수 있습니다.</p>
설치됨	<pre>zoneadm -z <i>zonename</i> ready(선택 사항) zoneadm -z <i>zonename</i> boot zoneadm -z <i>zonename</i> uninstall</pre> 은 시스템에서 지정된 영역의 구성을 제거합니다. <pre>zoneadm -z <i>zonename</i> move <i>path</i> zoneadm -z <i>zonename</i> detach</pre> <p><code>zonecfg -z <i>zonename</i></code>을 사용하여 <code>attr</code>, <code>bootargs</code>, <code>capped-memory</code>, <code>dataset</code>, <code>capped-cpu</code>, <code>dedicated-cpu</code>, <code>device</code>, <code>fs</code>, <code>ip-type</code>, <code>limitpriv</code>, <code>net</code>, <code>rctl</code></p>

현재 영역 상태	해당되는 명령
	<p>또는 <code>scheduling-class</code> 등록 정보를 추가하거나 제거할 수 있습니다. 영역 이름을 바꿀 수도 있습니다.</p> <p><code>zoneadm</code> 명령을 사용하여 구성됨 또는 설치됨 상태에 있는 Oracle Solaris 영역 또는 Oracle Solaris 10 영역의 이름을 바꿀 수 있습니다.</p> <p><code>zoneadm -z zonename mark incomplete</code></p> <p><code>zoneadm -z zonename mark unavailable</code></p>
준비	<p><code>zoneadm -z zonename boot</code></p> <p><code>zoneadm halt</code>를 실행하고 시스템을 재부트하면 준비 상태의 영역이 설치됨 상태로 복원됩니다.</p> <p><code>zonecfg -z zonename</code>을 사용하면 <code>attr</code>, <code>bootargs</code>, <code>capped-memory</code>, <code>dataset</code>, <code>capped-cpu</code>, <code>dedicated-cpu</code>, <code>device</code>, <code>fs</code>, <code>ip-type</code>, <code>limitpriv</code>, <code>net</code>, <code>rctl</code> 또는 <code>scheduling-class</code> 등록 정보를 추가하거나 제거할 수 있습니다.</p>
실행 중	<p><code>zlogin options zonename</code></p> <p><code>zoneadm -z zonename reboot</code></p> <p><code>zoneadm -z zonename halt</code>는 준비 상태의 영역을 설치됨 상태로 되돌립니다.</p> <p><code>zoneadm halt</code>를 실행하고 시스템을 재부트하면 실행 중 상태의 영역이 설치됨 상태로 복원됩니다.</p> <p><code>zoneadm -z shutdown</code>은 영역을 완전히 종료합니다.</p> <p><code>zonecfg -z zonename</code>을 사용하면 <code>attr</code>, <code>bootargs</code>, <code>capped-memory</code>, <code>dataset</code>, <code>capped-cpu</code>, <code>dedicated-cpu</code>, <code>device</code>, <code>fs</code>, <code>ip-type</code>, <code>limitpriv</code>, <code>net</code>, <code>rctl</code> 또는 <code>scheduling-class</code> 등록 정보를 추가하거나 제거할 수 있습니다. <code>zonepath</code> 리소스는 변경할 수 없습니다.</p>

참고 - `zonecfg`를 통해 변경된 매개변수는 실행 중 영역에 영향을 주지 않습니다. 변경 내용을 적용하려면 영역을 재부트해야 합니다.

비전역 영역 특성

영역은 필요한 거의 모든 레벨의 세분성에서 격리를 제공합니다. 영역에는 전용 CPU, 실제 장치 또는 실제 메모리의 부분이 필요하지 않습니다. 이들 리소스는 단일 도메인 또는 시스템에서 실행 중인 여러 영역에 걸쳐 다중화되거나 운영 체제에서 사용 가능한 리소스 관리 기능을 사용하여 각 영역마다 할당할 수 있습니다.

각 영역에서는 일련의 사용자 정의된 서비스가 제공될 수 있습니다. 기본 프로세스 격리를 강제로 적용하기 위해 동일한 영역에 존재하는 프로세스만 확인하거나 신호를 보낼 수 있습니다. 영역 간의 기본 통신은 각 영역에 IP 네트워크 연결을 제공함으로써 이루어집니다. 하나의 영역에서 실행되는 응용 프로그램은 다른 영역의 네트워크 트래픽을 볼 수 없습니다. 이러한 격리는 각각의 패킷 스트림이 동일한 실제 인터페이스를 통과하는 경우에도 유지됩니다.

각 영역에는 파일 시스템 계층의 일부분이 부여됩니다. 각 영역은 해당 파일 시스템 계층의 하위 트리로 제한되므로 특정 영역에서 실행되는 작업 부하가 다른 영역에서 실행되는 다른 작업 부하의 디스크 데이터에 액세스할 수 없습니다.

이름 지정 서비스에서 사용되는 파일은 영역의 자체 루프 파일 시스템 보기에 상주합니다. 따라서 서로 다른 영역의 이름 지정 서비스는 서로 격리되며 각각 다르게 구성될 수 있습니다.

리소스 관리 기능을 비전역 영역과 함께 사용

리소스 관리 기능을 사용하는 경우 리소스 관리 컨트롤의 경계를 영역의 경계와 맞춰야 합니다. 이 정렬을 통해 더욱 완벽한 가상 시스템 모델이 구축되며, 이러한 가상 시스템에서는 이름 공간, 보안 격리 및 리소스 사용이 모두 제어됩니다.

영역과 함께 다양한 리소스 관리 기능의 사용에 대한 특별 요구 사항은 이 설명서에서 이 기능을 설명하는 개별 절에서 설명합니다.

영역 관련 SMF 서비스

전역 영역의 영역 관련 SMF 서비스는 다음과 같습니다.

svc:/system/
zones:default autoboot=true인 각 영역을 시작합니다.

svc:/system/
zones-
install:default 필요한 경우 첫번째 부트 시 영역 설치를 수행합니다.

svc:/
application/
pkg/zones-
proxyd:default 패키징 시스템에서 영역에 시스템 저장소에 대한 액세스를 제공하는 데 사용됩니다.

svc:/
application/
pkg/system-
repository:default 영역 설치 및 pkg 작업 중 사용되는 pkg 데이터 및 메타 데이터를 캐시하는 캐싱 프록시 서버입니다. [pkg\(1\)](#) 및 [pkg\(5\)](#) 매뉴얼 페이지를 참조하십시오.

svc:/system/
zones-
monitoring:default zonestatd를 제어합니다.

svc:/application/pkg/zones-proxy-client:default 영역 프록시 클라이언트 SMF 서비스는 비전역 영역에서만 실행됩니다. 이 서비스는 패키징 시스템에서 영역에 시스템 저장소에 대한 액세스를 제공하는 데 사용됩니다.

비전역 영역 모니터링

현재 실행 중인 영역의 CPU, 메모리 및 리소스 제어 활용률에 대한 정보는 [“Oracle Solaris 영역 만들기 및 사용”](#)의 [“비전역 영역에서 zonestat 유틸리티 사용”](#)을 참조하십시오. zonestat 유틸리티는 배타적 IP 영역의 네트워크 대역폭 활용률도 보고합니다. 하나의 배타적 IP 영역에는 하나 이상의 전용 데이터 링크와 고유한 IP 관련 상태가 지정됩니다.

fsstat 유틸리티는 비전역 영역에 대한 파일 작업 통계를 보고하는 데 사용할 수 있습니다. [fsstat\(1M\)](#) 매뉴얼 페이지 및 [“Oracle Solaris 영역 만들기 및 사용”](#)의 [“fsstat 유틸리티를 사용하여 비전역 영역 모니터링”](#)을 참조하십시오.

비전역 영역에서 제공하는 기능

비전역 영역은 다음 기능을 제공합니다.

보안	<p>프로세스가 전역 영역이 아닌 다른 영역에 배치되고 나면 해당 프로세스와 이후 하위 프로세스에서 영역을 변경할 수 없습니다.</p> <p>네트워크 서비스는 영역에서 실행할 수 있습니다. 영역에서 네트워크 서비스를 실행하여 보안 위반 시 발생할 수 있는 손상을 제한할 수 있습니다. 영역 내에서 실행 중인 소프트웨어의 보안 결점을 악용하는 침입자는 해당 영역 내에서 가능한 특정 동작만 제한적으로 수행할 수 있습니다. 영역 내에서 사용할 수 있는 권한은 시스템 전체에서 사용할 수 있는 권한 중 일부입니다.</p>
격리	<p>여러 응용 프로그램이 서로 다른 트러스트 도메인에서 작동하거나, 전역 리소스에 대한 배타적 액세스를 필요로 하거나, 또는 전역 구성과 관련된 어려움이 있는 경우라도, 영역에서는 동일 시스템에 여러 응용 프로그램을 배포할 수 있습니다. 또한 응용 프로그램이 서로의 네트워크 트래픽, 파일 시스템 데이터 또는 프로세스 작업을 모니터링하거나 가로챌 수 없게 됩니다.</p>
네트워크 격리	<p>영역은 기본적으로 배타적 IP 유형으로 구성됩니다. 이 영역은 전역 영역으로부터 격리되고 IP 계층에서 서로 격리됩니다. 이러한 격리는 운영 및 보안상 유용합니다. 영역을 사용하면 고유한 LAN 또는 VLAN을 사용하는 다른 서브넷에서 통신해야 하는 응용 프로그램을 통합할 수 있습니다. 영역마다 고유한 IP 계층 보안 규칙을 정의할 수도 있습니다.</p>
가상화	<p>영역은 물리적 장치와 시스템의 기본 IP 주소와 호스트 이름 등 세부 정보를 응용 프로그램으로부터 숨길 수 있는 가상화 환경을 제공합니다. 동일한 응용 프로그램 환경이 서로 다른 물리적 시스템에서 유지될 수 있습니다. 가상화된 환경에서는 각 영역을 개별적으로 관리할 수 있습니다. 비전역 영역에서 영역 관리자가 수행하는 동작은 시스템의 나머지 부분에 영향을 미치지 않습니다.</p>

세분성	영역은 거의 모든 세분성 레벨에서 격리를 제공합니다. 자세한 내용은 “비전역 영역 특성” [22] 을 참조하십시오.
환경	영역은 보안 및 격리의 목적을 실현하는 데 필요한 경우를 제외하고는 응용 프로그램이 실행되는 환경을 변경하지 않습니다. 영역은 응용 프로그램을 이식해야 하는 새로운 API 또는 ABI를 제공하지 않습니다. 대신에, 영역은 약간의 제한과 함께 표준 Oracle Solaris 인터페이스 및 응용 프로그램 환경을 제공합니다. 이 제한은 기본적으로 권한이 부여된 작업을 수행하려 하는 응용 프로그램에 영향을 줍니다. 전역 영역의 응용 프로그램은 추가 영역의 구성 여부와 상관없이 수정되지 않은 상태로 실행됩니다.

이 Oracle Solaris 영역 릴리스 정보

이 절에서는 Oracle Solaris 커널 영역을 비롯한 Oracle Solaris 영역 기능에 대해 간략하게 설명합니다.

이 릴리스에서 기본 비전역 영역은 `solaris`이며, 이 설명서와 `solaris(5)` 매뉴얼 페이지에 설명되어 있습니다.

Oracle Solaris 릴리스와 시스템 구조를 확인하려면 다음을 입력합니다.

```
#uname -r -m
```

[virtinfo\(1M\)](#) 매뉴얼 페이지에 설명된 `virtinfo` 명령은 다음 정보를 얻는 데 사용됩니다.

- Oracle Solaris 가상화 기술에 대한 시스템 지원 확인
- Oracle Solaris가 실행 중인 가상 환경 유형(예: Oracle VM Server for SPARC) 감지

`solaris` 영역에서는 [brands\(5\)](#) 매뉴얼 페이지에 설명된 브랜드 영역 프레임워크를 사용하여 전역 영역과 동일한 소프트웨어로 설치된 영역을 실행합니다. 이 시스템 소프트웨어는 `solaris` 브랜드 비전역 영역 사용 시 전역 영역과 동기화되어야 합니다. 영역 내 시스템 소프트웨어 패키지는 IPS(이미지 패키징 시스템)를 사용하여 관리됩니다. IPS는 Oracle Solaris 11 릴리스의 패키징 시스템이므로, `solaris` 영역은 이 모델을 사용합니다.

Oracle Solaris 11 Express에서 만들어진 기본 `ipkg` 영역은 `solaris` 영역으로 매핑됩니다. [“ipkg 영역을 solaris 영역으로 변환에 대한 정보” \[11\]](#)를 참조하십시오.

AI(자동 설치) 매니페스트에 지정된 각각의 비전역 영역이 클라이언트 설치의 일부로 설치 및 구성됩니다. 비전역 영역은 전역 영역이 설치된 후에 처음 재부트 시 설치 및 구성됩니다. 시스템이 처음 부트할 때 영역 자체 어셈블리 SMF 서비스인 `svc:/system/zones-install:default`가 전역 영역 AI 매니페스트에 정의된 각각의 비전역 영역을 구성 및 설치합니다. 자세한 내용은 [“Oracle Solaris 11.2의 소프트웨어 추가 및 업데이트”](#)를 참조하십시오. 또한 설치된 Oracle Solaris 시스템에서 영역을 수동으로 구성 및 설치할 수도 있습니다.

패키지 업데이트를 위해 `--proxy` 옵션을 사용하여 이미지에서 지속성 프록시를 설정해야 합니다. 지속성 이미지 프록시 구성을 사용하지 않는 경우 `http_proxy` 및 `https_proxy` 환경 변수를 설정할 수 있습니다.

영역을 직렬이 아닌 병렬로 업데이트하도록 구성할 수 있습니다. 병렬 업데이트는 시스템의 모든 영역을 업데이트하는 데 필요한 시간을 상당히 단축합니다.

기본적으로 영역은 배타적 IP 유형으로 만들어집니다. 네트워크 구성을 지정하지 않은 경우 `anet` 리소스를 통해 영역 구성에 VNIC이 자동으로 포함됩니다. 자세한 내용은 [“영역 네트워크 인터페이스” \[38\]](#)를 참조하십시오.

영역의 `mac-address`를 가져오는 데 사용되는 `auto-mac-address`에 대한 자세한 내용은 [“리소스 유형 등록 정보” \[58\]](#)에서 `anet` 항목을 참조하십시오.

공유 저장소의 `solaris` 영역에는 `zonecfg rootzpool` 리소스가 있습니다. 영역은 전용 `zpool`로 캡슐화됩니다. 공유 저장소의 영역은 영역의 공유 저장소 리소스를 액세스하고 관리합니다. 커널 영역에는 `zpool` 또는 `rootzpool` 리소스가 없습니다. `solaris` 브랜드 영역은 영역 `device` 리소스, `zpool` 및 `rootzpool` 리소스에 대해 다음과 같은 공유 저장소를 사용할 수 있습니다.

- iSCSI
- FC LUN
- DAS

IPoIB(IP over InfiniBand) 데이터 링크를 지정하는 데 사용되는 등록 정보는 `zonecfg anet` 리소스에 사용할 수 있습니다. IPoIB는 `solaris` 및 `solaris10` 브랜드 영역에서 지원됩니다.

RDS(Reliable Datagram Socket) IPC 프로토콜은 배타적 IP 및 공유 IP 비전역 영역 모두에서 지원됩니다.

`fsstat` 유틸리티는 영역을 지원하도록 확장되었습니다. `fsstat` 유틸리티는 영역별 및 종합 통계를 제공합니다.

[“Oracle Solaris 영역 만들기 및 사용”](#)의 [“영역에서 NFS 서버 실행”](#) 절에 설명된 것과 같이, `solaris` 영역은 NFS 서버일 수 있습니다.

`dry-run`이라고도 하는 테스트 실행, `zoneadm attach -n`은 `zonecfg` 검증을 제공하지만 패키지 내용 검증은 수행하지 않습니다.

파일을 인수로 가져오는 모든 `zoneadm` 옵션에는 절대 경로가 필요합니다.

Oracle Solaris 10 영역은 Oracle Solaris 11에서 Oracle Solaris 10 환경을 제공합니다. Oracle Solaris 10 시스템 또는 영역을 Oracle Solaris 11 시스템의 `solaris10` 영역으로 마이그레이션할 수 있습니다. [“Oracle Solaris 10 영역 만들기 및 사용”](#)을 참조하십시오.

`zonep2vchk` 도구는 Oracle Solaris 11 시스템이나 Oracle Solaris 10 시스템을 Oracle Solaris 11 릴리스를 실행하는 시스템의 영역으로 마이그레이션하는 데 영향을 줄 수 있는

네트워킹 문제를 비롯한 문제를 식별합니다. zonep2vchk 도구는 마이그레이션이 시작되기 전에 소스 시스템에서 실행됩니다. 또한 이 도구는 대상 시스템에서 사용할 zonecfg 스크립트를 출력합니다. 스크립트는 소스 시스템 구성과 일치하는 영역을 만듭니다. 자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”의 7 장](#), [“영역 마이그레이션 및 zonep2vchk 도구 정보”](#)를 참조하십시오.

Oracle Solaris 10 릴리스의 native 영역과 solaris 영역은 다음과 같은 차이점이 있습니다.

- Oracle Solaris 11 시스템에서는 Oracle Solaris 10 시스템의 기본값인 native 브랜드 대신에 solaris 브랜드가 생성됩니다.
- solaris 영역은 전체 루트 형식뿐입니다.

Oracle Solaris 10에서 사용할 수 있는 native 영역의 최소 루트 형식은 SVR4 패키지 관리 시스템을 사용하며, IPS는 이 시스템을 사용하지 않습니다. 최소 루트 형식과 비슷한 읽기 전용 루트 영역 구성을 사용할 수 있습니다.

- 이 번 릴리스의 영역은 다음 부분에서 Oracle Solaris 10 릴리스와는 다른 소프트웨어 관리 관련 기능을 제공합니다.

- IPS와 SVR4 패키징
- 설치, 분리, 연결 및 물리-가상 기능
- 비전역 루트 영역은 ZFS™ 데이터 세트입니다.

전역 영역에 설치된 패키지는 더 이상 현재와 미래의 모든 영역에 설치되지 않습니다. 일반적으로 IPS 및 SVR4 패키징에 대해 전역 영역의 패키지 콘텐츠가 더 이상 각 영역의 패키지 콘텐츠를 좌우하지 않습니다.

- 비전역 영역은 부트 환경을 사용합니다. 영역은 ZFS BE(부트 환경)를 관리하기 위한 사용자 인터페이스 명령인 beadm과 통합됩니다.

전역 영역에서처럼 pkg 업데이트를 위해 영역 내부에서 beadm 명령이 지원됩니다. beadm 명령은 영역과 연결된 비활성 영역 BE를 삭제할 수 있습니다. [beadm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

- 영역을 설치하는 동안 모든 사용 가능 IPS 패키지 저장소에 액세스할 수 있어야 합니다. 자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”의 “구성된 영역 설치 방법”](#)을 참조하십시오.
- 영역 소프트웨어가 최소화되어 시작됩니다. 영역에 필요한 추가 패키지를 추가해야 합니다. 자세한 내용은 [“Oracle Solaris 11.2의 소프트웨어 추가 및 업데이트”](#)를 참조하십시오.

영역은 다음과 같은 Oracle Solaris 제품과 기능을 사용할 수 있습니다.

- Oracle Solaris ZFS 암호화
- 네트워크 가상화 및 QoS
- CIFS 및 NFS

다음 기능은 solaris-kz 브랜드 영역에서 구성할 수 없습니다.

- FC 서비스
- FCoE 서비스

라이브 영역 재구성

라이브 영역 재구성을 사용하여 재부트 없이 실행 중인 solaris 또는 solaris10 영역의 라이브 구성을 재구성하거나 보고할 수 있습니다. 변경은 임시로 또는 영구적으로 수행될 수 있습니다.

라이브 영역 재구성을 사용하여 solaris-kz 브랜드 영역에 대한 라이브 구성 정보를 보고할 수 있습니다.

자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”](#)을 참조하십시오.

◆◆◆ 2 장

비전역 영역 구성 개요

이 장에서는 비전역 영역 구성에 대해 소개합니다.

이 장에서 다루는 항목은 다음과 같습니다.

- “영역 내 리소스 정보” [29]
- “설치 전 구성 프로세스” [30]
- “영역 구성 요소” [31]
- “zonecfg 명령 사용” [49]
- “zonecfg 모드” [50]
- “영역 구성 데이터” [53]
- “명령줄 편집 라이브러리” [68]

영역 구성에 대해 알아본 후에는 “Oracle Solaris 영역 만들기 및 사용”의 1 장, “비전역 영역을 계획 및 구성하는 방법”으로 이동하여 시스템에 설치할 비전역 영역을 구성하십시오.

영역 내 리소스 정보

영역에서 제어할 수 있는 리소스에는 다음 항목이 포함됩니다.

- 시스템 리소스 분할에 사용되는 리소스 풀 또는 지정된 CPU
- 시스템 리소스에 대한 제약 조건 방식을 제공하는 리소스 제어
- 중요성에 따라 영역 간의 사용 가능한 CPU 리소스 할당을 제어할 수 있는 예약 클래스. 이 중요성은 각 영역에 지정하는 CPU 리소스의 할당 수로 표현됩니다.

영역 관리 시 권한 프로파일과 역할 사용

프로파일 및 역할에 대한 자세한 내용은 “Oracle Solaris 11 보안 지침”의 “응용 프로그램 보호 및 격리”를 참조하십시오.

zonecfg template 등록 정보

zonecfg template 등록 정보를 사용하여 다음과 같은 경우에 등록 정보를 변경할지 여부 및 변경하는 방식을 정의할 수 있습니다.

- 새 리소스 인스턴스가 구성에 추가된 경우
- 구성 복제 중 일부 등록 정보가 고유 값을 가져야 하는 경우. template 등록 정보의 토큰을 사용하여 고유 값을 제공할 수 있습니다.

표 2-1 zonecfg template 토큰

토큰	설명	사용법
%zonename	영역 이름입니다.	브랜드 메타 데이터 및 zonecfg에서 사용자 또는 템플릿 값의 입력으로 사용할 수 있습니다.
%network-id	네트워크 리소스 net 및 anet에 대한 고유 인스턴스 번호입니다. 이 번호는 전역 영역 내에서 net 및 anet 리소스에 대해 고유합니다.	브랜드 메타 데이터에서 id 등록 정보 net 및 anet 리소스에 대한 기본 속성으로 사용할 수 있습니다.
%resource-id	제공된 리소스 전역 범위, 전역 영역 내에서 net 및 anet을 제외한 모든 리소스에 대한 고유 인스턴스 번호입니다.	브랜드 메타 데이터에서 id 등록 정보에 대한 기본 속성으로 사용할 수 있습니다.
%id	리소스의 id 등록 정보 값인 고유 인스턴스 번호입니다.	zonecfg에서 사용자 또는 템플릿 값의 입력으로 사용할 수 있습니다. id 등록 정보를 지원하는 리소스 범위 내에 사용해야 합니다.
%	%와 같습니다.	브랜드 메타 데이터 및 zonecfg에서 사용자 입력으로 사용할 수 있습니다.

영역 RAD(원격 관리 데몬) 모듈 구성은 등록 정보 템플릿을 사용하여 변경 사항을 표시, 적용 또는 구현할 수 있는 체계적인 방법을 제공합니다. zonemgr(3RAD) 매뉴얼 페이지를 참조하십시오. 처음에는 rad-zonemgr 패키지를 시스템에 설치하지 않았지만 나중에 pkg install을 사용하여 설치한 경우 rad:local을 다시 시작해야 합니다. rad:remote도 실행 중인 경우 다시 시작하십시오. 다시 시작하려면 [svcadm\(1M\)](#)을 사용하십시오. RAD 데몬이 모듈을 로드했는지 확인하십시오.

설치 전 구성 프로세스

시스템에서 비전역 영역을 설치하고 사용하려면 먼저 영역을 구성해야 합니다.

zonecfg 명령은 구성을 생성하고 지정된 리소스와 등록 정보가 가설 시스템에서 유효한지 여부를 결정하는 데 사용됩니다. 지정된 구성에 대해 zonecfg를 통해 수행되는 검사에서는 다음 사항을 확인합니다.

- 영역 경로가 지정되었는지 여부를 확인합니다.
- 각 리소스에 대한 필요한 모든 등록 정보가 지정되었는지 확인합니다.
- 구성에 충돌이 없는지 확인합니다. 예를 들어 anet 리소스가 있을 경우 영역은 배타적 IP 유형이므로 공유 IP 영역이 될 수 없습니다. 또한 zonecfg 명령은 가칭된 데이터 세트가 장치와 충돌할 가능성이 있는 경우 경고를 표시합니다.

zonecfg 명령에 대한 자세한 내용은 [zonecfg\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

영역 구성 요소

이 절에서는 구성할 수 있는 필수 및 선택적 영역 구성 요소에 대해 설명합니다. 영역 이름과 영역 경로만 필요합니다. 추가 정보는 “[영역 구성 데이터](#)” [53]를 참조하십시오.

영역 이름 및 경로

영역 이름을 선택해야 합니다. 경로를 지정하지 않은 경우 zonepath의 기본값은 /system/zones/*zonename*입니다.

영역의 이름과 경로를 선택하는 경우 영역이 ZFS 데이터 세트에 있어야 합니다. ZFS 데이터 세트는 영역을 설치하거나 연결할 때 자동으로 생성됩니다. ZFS 데이터 세트를 만들 수 없을 경우 영역이 설치되거나 연결되지 않습니다. 영역 경로의 부모 디렉토리도 데이터 세트여야 합니다.

경로를 지정하지 않은 경우 zonepath의 기본값은 /system/zones/*zonename*입니다.

영역 자동 부트

autoboot 등록 정보 설정은 전역 영역이 부트될 때 영역이 자동으로 부트되는지 여부를 결정합니다. 영역 서비스인 svc:/system/zones:default도 사용으로 설정해야 합니다.

변경 불가능한 영역에 대한 file-mac-profile 등록 정보

solaris 영역에서는 file-mac-profile이 읽기 전용 루트를 사용하여 변경 불가능한 영역을 구성하는 데 사용됩니다.

자세한 내용은 “Oracle Solaris 영역 만들기 및 사용”의 12 장, “변경할 수 없는 영역 구성 및 관리”를 참조하십시오.

admin 리소스

admin 설정을 사용하면 영역 관리 권한 부여를 설정할 수 있습니다. 권한 부여를 정의하는 기본 방법은 zonecfg 명령을 사용하는 것입니다.

user	사용자 이름을 지정합니다.
auths	사용자 이름에 대한 권한을 지정합니다.
solaris.zone.login	RBAC를 사용 중인 경우 대화식 로그인을 위해 solaris.zone.login/ <i>zonename</i> 권한 부여가 필요합니다. 영역에서는 암호 인증이 수행됩니다.
solaris.zone.manage	RBAC가 사용 중인 경우 비대화식 로그인 시 또는 암호 인증을 생략하려면 solaris.zone.manage/ <i>zonename</i> 권한이 필요합니다.
solaris.zone.clonefrom	RBAC가 사용 중인 경우 다른 영역을 복사하는 하위 명령에 solaris.zone.clonefrom/ <i>source_zone</i> 권한이 필요합니다.

권한 부여에 대한 자세한 내용은 [auths\(1\)](#), [auth_attr\(4\)](#) 및 [user_attr\(4\)](#)를 참조하십시오.

dedicated-cpu 리소스

dedicated-cpu 리소스는 비전역 영역이 실행 중인 동안 시스템 프로세서의 하위 세트를 해당 비전역 영역 전용으로 지정합니다. 영역이 부트될 때 시스템은 영역이 실행 중인 동안 사용할 임시 풀을 동적으로 만듭니다.

zonecfg을 통해 지정한 상태에서는 풀 설정이 마이그레이션 중에 전파됩니다.

dedicated-cpu 리소스는 ncpus 및 importance(선택 사항)의 제한을 설정합니다.

ncpus	CPU 수를 지정하거나 2-4개 CPU 등 범위를 지정합니다. 동적 리소스 풀 동작을 위해서 범위를 지정하는 경우 다음 작업도 수행합니다.
-------	---

- importance 등록 정보를 설정합니다.
- poold 서비스를 사용으로 설정합니다. 지침은 “Oracle Solaris 11.2의 리소스 관리”의 “svcadm을 사용하여 동적 리소스 풀 서비스를 사용으로 설정하는 방법”을 참조하십시오.

importance CPU 범위를 사용하여 동작 동작을 수행하려면 importance 등록 정보도 설정합니다. importance 등록 정보는 선택 사항으로, 풀의 상대적 중요성을 정의합니다. 이 등록 정보는 ncpus에 대한 범위를 지정하고 poold를 통해 관리되는 동적 리소스 풀을 사용할 때만 필요합니다. poold를 실행하고 있지 않은 경우 importance가 무시됩니다. poold가 실행 중이고 importance가 설정되지 않은 경우 importance의 기본값은 1입니다. 자세한 내용은 “Oracle Solaris 11.2의 리소스 관리”의 “pool.importance 등록 정보 제약 조건”을 참조하십시오.

다음 등록 정보는 cpus, cores 및 sockets에 대해 dedicated-cpu 지속 리소스를 설정하는 데 사용됩니다.

cpus	특정 CPU를 영역에 영구 지정합니다.
cores	특정 코어를 영역에 영구 지정합니다.
sockets	지정된 수의 소켓을 영구 지정합니다.

참고 - capped-cpu 리소스와 dedicated-cpu 리소스는 호환되지 않습니다. cpu-shares rctl과 dedicated-cpu 리소스는 호환되지 않습니다.

참고 - 자동 크기 조절을 수행하고 사용 가능한 CPU 수에 맞게 자동으로 확대/축소를 수행하는 응용 프로그램에서는 capped-cpu 제한을 인식하지 못할 수 있습니다. 모든 CPU가 사용 가능한 상태로 표시되면 Oracle 데이터베이스 및 JVM(Java Virtual Machine)과 같은 응용 프로그램에서 확대/축소가 부정적인 영향을 받으며 성능이 저하될 수 있습니다. 응용 프로그램이 작동하지 않거나 사용할 수 없는 것으로 표시될 수 있습니다. 성능이 중요한 경우 capped-cpu와 함께 JVM을 사용하지 않아야 합니다. 영향을 받는 범주의 응용 프로그램은 dedicated-cpu 리소스를 사용할 수 있습니다.

solaris-kz만 해당: virtual-cpu 리소스

virtual-cpu 리소스를 사용하여 커널 영역 CPU 수를 설정할 수 있습니다. 커널 영역에만 사용되는 호스트 CPU는 ncpus 값으로 정의됩니다. 기본 커널 영역 구성에는 1개의 CPU가 있습니다. virtual-cpu 등록 정보를 추가하여 커널 영역에 CPU를 추가할 수 있습니다.

dedicated-cpu 리소스가 이미 정의된 경우 가상 플랫폼에 구성된 기본 가상 CPU 수는 dedicated-cpu 리소스에서 ncpus 범위의 하한값과 일치합니다. dedicated-cpu 리소스와 virtual-cpu 리소스를 둘 다 설정할 필요는 없습니다.

capped-cpu 리소스

capped-cpu 리소스는 프로젝트나 영역에서 사용할 수 있는 CPU 리소스 양에 대한 세분화된 절대적 제한을 제공합니다. CPU 상한값은 프로세서 세트와 함께 사용되는 경우 세트 내에서 CPU 사용을 제한합니다. capped-cpu 리소스에는 소수점 이하 자리 수가 두 개인 양수를 나타내는 단일 ncpus 등록 정보가 있습니다. 이 등록 정보는 CPU 단위에 해당합니다. 리소스는 범위를 허용하지 않습니다. 리소스는 십진수를 허용합니다. ncpus를 지정하는 경우 값 1은 CPU 100%를 의미합니다. 100%가 시스템에서 한 개의 CPU 전체를 의미하므로 값 1.25는 125%를 의미합니다.

참고 - capped-cpu 리소스와 dedicated-cpu 리소스는 호환되지 않습니다.

참고 - 자동 크기 조절을 수행하고 사용 가능한 CPU 수에 맞게 자동으로 확대/축소를 수행하는 응용 프로그램에서는 capped-cpu 제한을 인식하지 못할 수 있습니다. 모든 CPU가 사용 가능한 상태로 표시되면 Oracle 데이터베이스 및 JVM(Java Virtual Machine)과 같은 응용 프로그램에서 확대/축소가 부정적인 영향을 받으며 성능이 저하될 수 있습니다. 응용 프로그램이 작동하지 않거나 사용할 수 없는 것으로 표시될 수 있습니다. 성능이 중요한 경우 capped-cpu와 함께 JVM을 사용하지 않아야 합니다. 영향을 받는 범주의 응용 프로그램은 dedicated-cpu 리소스를 사용할 수 있습니다. [“dedicated-cpu 리소스” \[32\]](#)를 참조하십시오.

예약 클래스

FSS(Fair Share Scheduler)를 사용하여 중요성에 따라 사용 가능한 CPU 리소스 할당을 제어할 수 있습니다. 이 중요성은 각 영역에 지정하는 CPU 리소스의 할당 수로 표현됩니다. FSS를 사용하여 영역 간 CPU 리소스 할당을 관리하기를 원치 않더라도 영역 내에서 프로젝트에 대한 할당을 설정할 수 있도록 영역의 예약 클래스를 설정하여 FSS를 사용할 수 있습니다.

cpu-shares 등록 정보를 명시적으로 설정하는 경우 FSS(Fair Share Scheduler)가 해당 영역에 대한 예약 클래스로 사용됩니다. 하지만 여기서의 기본적인 FSS 사용 방법은 dispadmin 명령을 사용하여 FSS를 시스템 기본 예약 클래스로 설정하는 것입니다. 즉, 모든 영역이 시스템 CPU 리소스를 공정하게 할당받을 수 있습니다. cpu-shares가 영역에 대해 설정되지 않을 경우 영역에서는 시스템의 기본 예약 클래스를 사용합니다. 다음 작업은 영역에 대한 예약 클래스를 설정합니다.

- zonecfg에서 scheduling-class 등록 정보를 사용하여 해당 영역에 대한 예약 클래스를 설정할 수 있습니다.
- 리소스 풀 기능을 통해 영역에 대한 예약 클래스를 설정할 수 있습니다. pool.scheduler 등록 정보가 유효한 예약 클래스로 설정된 풀과 영역이 연결된 경우 영역에서 실행 중인 클래스가 기본적으로 해당 예약 클래스에서 실행됩니다. [“Oracle Solaris 11.2의 리소스 관리”](#)의 [“리소스 풀 소개”](#) 및 [“Oracle Solaris 11.2의 리소스 관리”](#)의 [“풀과 예약 클래스를 연결하는 방법”](#)을 참조하십시오.

- `cpu-shares rctl`이 설정되었으며 FSS가 다른 작업을 통해 영역의 예약 클래스로 설정되지 않은 경우 `zoneadmd`는 영역이 부트될 때 예약 클래스를 FSS로 설정합니다.
- 다른 작업을 통해 예약 클래스가 설정되지 않은 경우 영역은 시스템 기본 예약 클래스를 상속합니다.

`prionctl(1)` 매뉴얼 페이지에 설명된 `prionctl`을 사용하여 기본 예약 클래스와 재부트 없이 실행 중인 프로세스를 다른 예약 클래스로 이동할 수 있습니다.

물리적 메모리 제어 및 `capped-memory` 리소스

`capped-memory` 리소스는 `physical`, `swap` 및 `locked` 메모리에 대한 제한을 설정합니다. 각 제한은 선택 사항이지만 하나 이상의 제한을 설정해야 합니다. `capped-memory` 리소스를 사용하려면 `resource-cap` 패키지를 전역 영역에 설치해야 합니다. “[capped-cpu 리소스](#)” [34]를 참조하십시오.

- 전역 영역에서 `rcapd`를 사용하여 해당 영역에 대한 메모리를 제한하려는 경우 이 리소스의 값을 결정합니다. `capped-memory` 리소스의 `physical` 등록 정보는 해당 영역의 `max-rss` 값으로 사용되는 `rcapd`에 사용됩니다.
- `capped-memory` 리소스의 `swap` 등록 정보는 `zone.max-swap` 리소스 제어를 설정하는 기본 방법입니다.
- `capped-memory` 리소스의 `locked` 등록 정보는 `zone.max-locked-memory` 리소스 제어를 설정하는 기본 방법입니다.

참고 - 응용 프로그램은 일반적으로 많은 양의 메모리를 잠그지 않지만, 영역의 응용 프로그램이 메모리를 잠그는 것으로 알려진 경우에는 잠김 메모리를 설정할 수 있습니다. 영역 신뢰가 문제가 되는 경우 잠긴 메모리 상한값을 시스템의 물리적 메모리 중 10% 또는 영역 물리적 메모리 상한값의 10%로 설정하는 것을 고려해 볼 수도 있습니다.

자세한 내용은 “[Oracle Solaris 11.2의 리소스 관리](#)”의 10 장, “[리소스 상한값 지원 데몬을 사용한 물리적 메모리 제어 정보](#)”, “[Oracle Solaris 11.2의 리소스 관리](#)”의 11 장, “[리소스 상한값 지원 데몬 관리\(작업\)](#)” 및 “[Oracle Solaris 영역 만들기 및 사용](#)”의 “[영역 구성 방법](#)”을 참조하십시오. 영역에 대한 리소스 상한값을 일시적으로 설정하려면 “[Oracle Solaris 11.2의 리소스 관리](#)”의 “[영역에 대한 임시 리소스 상한값을 지정하는 방법](#)”을 참조하십시오.

`solaris` 및 `solaris10`만 해당: `rootzpool` 리소스

`zonecfg` 유틸리티의 선택적 `rootzpool` 리소스는 `solaris` 및 `solaris10` 브랜드 영역의 영역 설치를 위한 전용 `zpool`을 만드는 데 사용됩니다. 영역 루트 `zpool`은 하나 이상의 URI(Universal Resource Identifier)로 정의된 공유 저장 장치에서 호스트될 수 있습니다. 필수 `storage` 등록 정보는 영역의 루트 `zfs` 파일 시스템을 포함할 저장소 객체 URI를 식별합니다. 지정된 영역에 대해 `rootzpool`을 하나만 정의할 수 있습니다. 영역이 부트될 때 영역의 저장소가 자동으로 구성됩니다.

영역 설치 또는 영역 연결 작업 중 해당하는 zpool을 자동으로 만들거나 가져옵니다. rootzpool 및 zpool 리소스의 경우 영역이 설치되는 즉시 자동으로 zpool 미러를 만들 수 있습니다. 자세한 내용은 “[Oracle Solaris 영역 만들기 및 사용](#)”의 14 장, “[공유 저장소에서 Oracle Solaris 영역 시작하기](#)”를 참조하십시오.

영역을 제거하거나 분리하면 다음 작업이 수행됩니다.

- 해당 zpool이 자동으로 내보내지거나 삭제됩니다.
- 저장소 리소스가 자동으로 구성 해제됩니다.

미리 만든 zpool을 영역 설치에 다시 사용하려면 시스템에서 zpool을 내보내야 합니다.

영역 프레임워크에서는 다음 URI 유형을 지원합니다.

■ dev

로컬 장치 경로 URI

형식:

```
dev:local-path-under-/dev
dev://absolute-path-with-dev
dev:absolute-path-with-dev
```

예:

```
dev:dsk/c7t0d0s0
dev:///dev/dsk/c7t0d0s0
dev:/dev/dsk/c7t0d0s0
dev:chassis/SYS/HD1/disk
```

■ lu(논리 장치)

FC(광섬유 채널) 및 SAS(직렬 연결 SCSI)

형식:

```
lu:luname.naa.ID
lu:luname.eui.ID
lu:initiator.naa.ID,target.naa.ID,luname.naa.ID
lu:initiator.naa.ID,target.naa.ID,luname.eui.ID
```

예:

```
lu:luname.naa.5000c5000288fa25
lu:luname.eui.0021280001cf80f6
lu:initiator.naa.2100001d38089fb0,target.naa.2100001d38089fb0,luname.naa.5000c5000288fa25
lu:initiator.naa.2100001d38089fb0,target.naa.2100001d38089fb0,luname.eui.0021280001cf80f6
```

■ iscsi

iSCSI URI

형식:

```
iscsi:///lunname.naa.ID
iscsi:///lunname.eui.ID
iscsi://host[:port]/lunname.naa.ID
iscsi://host[:port]/lunname.eui.ID
iscsi:///target.IQN,lun.LUN
iscsi://host[:port]/target.IQN,lun.LUN
```

예:

```
iscsi:///lunname.eui.0021280001cf80f6
iscsi:///lunname.naa.600144f03d70c80000004ea57da10001
iscsi://[::1]/lunname.naa.600144f03d70c80000004ea57da10001
iscsi://127.0.0.1/lunname.naa.600144f03d70c80000004ea57da10001
iscsi://hostname:1234/lunname.eui.0021280001cf80f6
iscsi://hostname:3260/lunname.naa.600144f03d70c80000004ea57da10001
```

```
iscsi://127.0.0.1/target.iqn.com.sun:02:d0f2d311-f703,lun.0
iscsi:///target.iqn.com.sun:02:d0f2d311-f703,lun.6
iscsi://[::1]:1234/target.iqn.com.sun:02:d0f2d311-f703,lun.2
iscsi://hostname:1234/target.iqn.com.sun:4db41b76-e3d7-cd2f-bf2d-9abef784d76c,lun.0
```

suriadm 도구는 저장소 URI를 기반으로 공유 객체를 관리하는 데 사용됩니다. ID, NAA(Name Address Authority) 및 기존 저장소 객체의 URI 얻기에 대한 자세한 내용은 [suriadm\(1M\)](#) 및 [suri\(5\)](#) 매뉴얼 페이지를 참조하십시오.

시스템에서는 연관된 영역에 대해 새로 만들거나 가져온 rootzpool의 이름을 지정합니다. 지정된 이름의 형식은 `zonename _rpool`입니다.

storage 등록 정보는 rootzpool 리소스 범위 내에서 다음 명령을 사용하여 관리됩니다.

- add storage *URI* 문자열
- remove storage *URI* 문자열

zpool 리소스 자동 추가

zonecfg 유틸리티에 선택적 zpool 리소스를 구성하여 zpool을 비전역 영역에 위임할 수 있습니다. 영역이 부트될 때 영역의 zpool이 자동으로 구성됩니다.

영역 설치 또는 영역 연결 작업 중 해당하는 zpool을 자동으로 만들거나 가져옵니다.

영역을 제거하거나 분리하면 다음 작업이 수행됩니다.

- 해당 zpool이 자동으로 내보내지거나 삭제됩니다.
- 저장소 리소스가 자동으로 구성 해제됩니다.

필수 storage 등록 정보는 이 리소스와 연관된 저장소 객체 URI를 식별합니다.

storage 등록 정보는 zpool 리소스 범위 내에서 다음 설정을 사용하여 관리됩니다.

- add storage *URI* 문자열
- remove storage *URI* 문자열

zpool 리소스에는 name 등록 정보가 필수입니다. 이 등록 정보는 영역에 위임된 zpool의 이름에서 사용됩니다. ZFS 파일 시스템 name 구성 요소에 슬래시(/)를 포함할 수 없습니다.

새로 만들거나 가져온 zpool에 지정된 이름은 name 등록 정보의 값입니다. 이 이름은 비전역 영역 내부에서 표시되는 zpool 이름입니다. 새로 만들거나 가져온 zpool에 지정된 이름의 형식은 *zonename_name* (전역 영역에서 표시된 경우)입니다.

참고 - 저장소 객체에 기존 파티션, zpool 또는 UFS 파일 시스템이 포함되어 있으면 영역 설치가 실패합니다. 자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”의 “구성된 영역 설치 방법”](#)에 설명된 4단계를 참조하십시오.

영역 네트워크 인터페이스

zoncfg 유틸리티에서 네트워크 연결을 제공하기 위해 구성된 영역 네트워크 인터페이스는 영역이 부트될 때 자동으로 설정되어 영역에 배치됩니다.

IP(인터넷 프로토콜) 계층은 네트워크의 패킷을 받고 전달합니다. 이 계층에는 IP 경로 지정, ARP(Address Resolution Protocol), IPsec(IP 보안 구조) 및 IP 필터가 포함됩니다.

비전역 영역에 사용할 수 있는 IP 유형에는 공유 IP와 배타적 IP라는 두 가지 유형이 있습니다. 배타적 IP는 기본 IP 유형입니다. 공유 IP 영역은 전역 영역과 네트워크 인터페이스를 공유합니다. 공유 IP 영역을 사용하려면 ipadm 유틸리티를 통해 전역 영역의 구성을 수행해야 합니다. 배타적 IP 영역에는 전용 네트워크 인터페이스가 있어야 합니다. anet 리소스를 사용하여 배타적 IP 영역이 구성된 경우 전용 VNIC가 자동으로 만들어지고 해당 영역에 지정됩니다. 자동화된 anet 리소스를 사용하면 전역 영역에서 데이터 링크를 만들어 구성하고 데이터 링크를 비전역 영역에 지정할 필요가 없습니다. anet 리소스를 사용하여 다음을 수행합니다.

- 전역 영역 관리자가 비전역 영역에 지정된 데이터 링크에 대한 특정 이름을 선택할 수 있습니다.
- 여러 영역에서 동일한 이름의 데이터 링크를 사용할 수 있습니다.

이전 버전과의 호환성을 위해 미리 구성된 데이터 링크를 비전역 영역에 지정할 수 있습니다.

유형별 IP 기능에 대한 자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”의 “배타적 IP 비전역 영역의 네트워킹”](#) 및 [“Oracle Solaris 영역 만들기 및 사용”의 “공유 IP 비전역 영역의 네트워킹”](#)을 참조하십시오.

참고 - [“Oracle Solaris 11.2의 네트워크 보안”](#)에 설명된 링크 보호는 영역을 실행하는 시스템에서 사용할 수 있습니다. 이 기능은 전역 영역에서 구성됩니다.

데이터 링크 정보

데이터 링크는 OSI 프로토콜 스택의 계층 2에 있는 물리적 인터페이스로, 시스템에서 STREAMS DLPI(v2) 인터페이스로 표시됩니다. 이러한 인터페이스는 TCP/IP 등 프로토콜 스택에서 연결될 수 있습니다. 데이터 링크는 NIC(네트워크 인터페이스 카드)과 같은 물리적 인터페이스라고도 합니다. 데이터 링크는 `zonecfg` (1M)를 사용하여 구성된 `physical` 등록 정보입니다. `physical` 등록 정보는 VNIC일 수 있습니다.

기본적으로 Oracle Solaris 11에서 물리적 네트워크 장치 이름에는 장치 드라이버 이름(예: `nxge0`) 대신 일반 이름(예: `net0`)이 사용됩니다.

solaris 영역에서의 IPoB(IP over Infiniband) 사용에 대한 자세한 내용은 “[리소스 유형 등록 정보](#)” [58]에서 `anet` 설명을 참조하십시오.

탄력적 가상 스위치 및 영역 정보

`evs` 및 `vport` 등록 정보 세트를 사용하여 EVS(탄력적 가상 스위치)에 연결되는 `anet` 리소스의 경우, 해당 `anet` 리소스의 등록 정보가 `evs` 및 `vport` 쌍에 캡슐화됩니다. EVS `anet` 리소스에 대한 다음 등록 정보는 변경할 수 없습니다.

- `mac-address`
- `mtu`
- `maxbw`
- `priority`
- `allowed-address`
- `vlan-id`
- `defrouter`
- `lower-link`

EVS `anet` 리소스에 대해 설정할 수 있는 등록 정보는 다음과 같습니다.

- `linkname`
- `evs`
- `vport`
- `configure-allowed-address`

`tenant` 리소스도 설정해야 합니다. 테넌트는 이름 공간 관리에 사용됩니다. `tenant` 내에 정의된 EVS 리소스는 테넌트 이름 공간 외부에 표시되지 않습니다.

`evszone`이라는 영역에 대한 다음 입력은 `tenantA`라는 테넌트에 대해 `tenant` 리소스를 설정합니다. `zonecfg anet` 리소스 등록 정보는 `anet` 리소스를 포함하는 영역에 대해 VNIC를 만드는데, 이 리소스는 `evsa`라는 EVS 및 `vport0`이라는 VPort에 연결됩니다.

```
zonecfg:evszone> set tenant=tenantA
```

```
zonecfg:evszone> add anet

zonecfg:evszone> set evs=EVSA

zonecfg:evszone> set vport=vport0
```

자세한 내용은 “Oracle Solaris 11.2의 네트워크 가상화 및 네트워크 리소스 관리”의 5 장, “탄력적 가상 스위치 정보”를 참조하십시오.

공유 IP 비전역 영역

공유 IP 영역은 전역 영역의 기존 IP 인터페이스를 사용합니다. 영역에는 전용 IP 주소가 하나 이상 있어야 합니다. 공유 IP 영역은 IP 계층 구성 및 상태를 전역 영역과 공유합니다. 다음 두 가지 조건이 모두 참이면 영역은 공유 IP 인스턴스를 사용해야 합니다.

- 비전역 영역은 전역 및 비전역 영역이 동일 서브넷에 있는지 여부와 상관없이 전역 영역에서 사용되는 동일한 데이터 링크를 사용합니다.
- 배타적 IP 영역이 제공하는 다른 기능은 필요하지 않습니다.

공유 IP 영역에는 zonecfg 영역의 net 리소스를 사용하여 하나 이상의 IP 주소가 지정됩니다. 전역 영역에서 데이터 링크 이름도 구성해야 합니다.

zonecfg net 리소스에서 address 및 physical 등록 정보를 설정해야 합니다. defrouter 등록 정보는 선택 사항입니다.

전역 영역의 공유 IP 네트워킹 구성을 사용하려면 자동 네트워크 구성이 아닌 ipadm를 사용해야 합니다. ipadm을 통해 네트워킹 구성이 수행되는지 여부를 확인하려면 다음 명령을 실행합니다. 표시되는 응답이 DefaultFixed여야 합니다.

```
# svcprop -p netcfg/active_ncp svc:/network/physical:default
DefaultFixed
```

공유 IP 영역에 지정된 IP 주소는 논리 네트워크 인터페이스와 연결됩니다.

전역 영역에서 ipadm 명령을 사용하여 실행 중인 영역에서 논리 인터페이스를 지정하거나 제거할 수 있습니다.

인터페이스를 추가하려면 다음 명령을 사용합니다.

```
global# ipadm set-addrprop -p zone=my-zone net0/addr1
```

인터페이스를 제거하려면 다음 명령 중 하나를 사용합니다.

```
global# ipadm set-addrprop -p zone=global net0/addr
```

또는

```
global# ipadm reset-addrprop -p zone net0/addr1
```

자세한 내용은 “Oracle Solaris 영역 만들기 및 사용”의 “공유 IP 네트워크 인터페이스”를 참조하십시오.

배타적 IP 비전역 영역

배타적 IP는 비전역 영역을 위한 기본 네트워킹 구성입니다.

하나의 배타적 IP 영역에는 하나 이상의 전용 데이터 링크와 고유한 IP 관련 상태가 지정됩니다.

배타적 IP 영역에서 다음 기능을 사용할 수 있습니다.

- DHCPv4 및 IPv6 Stateless 주소 자동 구성
- IP 필터, NAT(네트워크 주소 변환) 기능 포함
- IPMP(IP Network Multipathing)
- IP 경로 지정
- TCP/UDP/SCTP 및 IP/ARP 레벨 조정 가능 매개변수 설정용 ipadm
- IPsec 보안 연결을 위한 인증된 키 관련 자료 프로비저닝을 자동화하는 IKE(Internet Key Exchange)와 IPsec(IP 보안)

배타적 IP 영역을 구성하는 방법에는 두 가지가 있습니다.

- zonecfg 유틸리티의 anet 리소스 사용을 사용하여 영역이 부트될 때 영역의 임시 VNIC를 자동으로 생성하고 영역이 정지할 때 삭제합니다.
- zonecfg 유틸리티의 net 리소스를 사용하여 전역 영역에서 데이터 링크를 미리 구성하고 배타적 IP 영역에 지정합니다. net 리소스의 physical 등록 정보를 사용하여 데이터 링크가 지정됩니다. physical 등록 정보는 VNIC일 수 있습니다. net 리소스의 address 등록 정보는 설정되지 않습니다.

기본적으로 배타적 IP 영역은 연결된 인터페이스의 IP 주소를 구성하고 사용할 수 있습니다. 또는 allowed-address 등록 정보를 사용하여 심표로 구분된 IP 주소 목록을 지정할 수 있습니다. 배타적 IP 영역은 allowed-address 목록에 없는 IP 주소를 사용할 수 없습니다. 또한, allowed-address 목록의 모든 주소는 영역이 부트될 때 배타적 IP 영역에 대해 자동으로 영구 구성됩니다. 이 인터페이스 구성을 원치 않는 경우에는 configure-allowed-address 등록 정보를 false로 설정해야 합니다. 기본값은 true입니다.

지정된 데이터 링크는 snoop 명령을 사용으로 설정합니다.

dladm 명령을 show-linkprop 하위 명령과 함께 사용하여 실행 중인 배타적 IP 영역에 데이터 링크의 할당을 표시할 수 있습니다. dladm 명령을 set-linkprop 하위 명령과 함께 사용하여 실행 중인 영역에 추가 데이터 링크를 지정할 수 있습니다. 사용 예는 “Oracle Solaris 영역 만들기 및 사용”의 “배타적 IP 비전역 영역에서 데이터 링크 관리”를 참조하십시오.

고유한 데이터 링크 세트가 지정된 실행 중인 배타적 IP 영역 내부에서는 `ipadm` 명령을 사용하여 IP를 구성합니다. 여기에는 논리적 인터페이스를 추가하거나 제거할 수 있는 기능이 포함됩니다. `sysconfig(1M)` 매뉴얼 페이지에 설명된 `sysconfig` 인터페이스를 사용하여 전역 영역에서와 똑같이 영역 내 IP 구성을 설정할 수 있습니다.

배타적 IP 영역의 IP 구성은 `zlogin` 명령을 사용하여 전역 영역에서만 볼 수 있습니다.

```
global# zlogin zone1 ipadm show-addr
ADDROBJ          TYPE      STATE      ADDR
lo0/v4           static   ok         127.0.0.1/8
nge0/v4          dhcp     ok         10.134.62.47/24
lo0/v6           static   ok         ::1/128
nge0/_a          addrconf ok         fe80::2e0:81ff:fe5d:c630/10
```

비전역 영역에서 Reliable Datagram Socket 지원

RDS(Reliable Datagram Socket) IPC 프로토콜은 배타적 IP 및 공유 IP 비전역 영역 모두에서 지원됩니다. RDSv3 드라이버는 SMF 서비스 `rds`로 사용 가능합니다. 기본적으로 이 서비스는 설치 후 사용 안함으로 설정됩니다. 적절한 권한이 부여된 영역 관리자가 주어진 비전역 영역 내에서 이 서비스를 사용으로 설정할 수 있습니다. `zlogin` 후 `rds`를 실행할 각 영역에서 사용으로 설정할 수 있습니다.

예 2-1 비전역 영역에서 `rds` 서비스를 사용으로 설정하는 방법

1. 배타적 IP 또는 공유 IP 영역에서 RDSv3 서비스를 사용으로 설정하려면 `zlogin`하고 `svcadm enable` 명령을 실행합니다.

```
# svcadm enable rds
```

2. `rds`가 사용으로 설정되어 있는지 확인합니다.

```
# svcs rds
STATE      STIME      FMRI
online     22:50:53  svc:/system/rds:default
```

자세한 내용은 `svcadm(1M)` 매뉴얼 페이지를 참조하십시오.

공유 IP 비전역 영역과 배타적 IP 비전역 영역의 보안 차이점

공유 IP 영역에서는 슈퍼 유저를 비롯하여 영역의 응용 프로그램이 `zonecfg` 유틸리티를 통해 영역에 지정된 주소와는 다른 소스 IP 주소와 함께 패킷을 전송할 수 없습니다. 이 영역 유형에는 임의의 데이터 링크(계층 2) 패킷을 보내고 받을 수 있는 권한이 없습니다.

배타적 IP 영역의 경우 대신에 `zonecfg`는 지정된 전체 데이터 링크를 영역에 부여합니다. 결과적으로 배타적 IP 영역에서는 슈퍼 유저나 필요한 권한 프로파일을 가진 사용자가 전역 영역에서처럼 이러한 데이터 링크를 통해 스푸핑된 패킷을 보낼 수 있습니다. `allowed-address`

등록 정보를 설정하여 IP 주소 스푸핑을 사용 안함으로 설정할 수 있습니다. anet 리소스의 경우 mac-nospoof 및 dhcp-nospoof 등 추가 보호는 link-protection 등록 정보를 설정하여 사용으로 설정할 수 있습니다.

공유 IP 및 배타적 IP 비전역 영역 동시 사용

공유 IP 영역은 항상 IP 계층을 전역 영역과 공유하며 배타적 IP 영역은 항상 IP 계층의 고유 인스턴스를 가지고 있습니다. 공유 IP 영역과 배타적 IP 영역 둘 다 동일한 시스템에서 사용할 수 있습니다.

영역에서 마운트된 파일 시스템

각 영역에는 기본적으로 ZFS 데이터 세트가 위임됩니다. 이 기본 위임 데이터 세트는 기본 전역 영역 데이터 세트 레이아웃의 데이터 세트 레이아웃을 대신합니다. .../rpool/ROOT라고 하는 데이터 세트에는 부트 환경이 포함되어 있습니다. 이 데이터 세트는 직접 조작하면 안 됩니다. 존재해야 하는 rpool 데이터 세트는 기본적으로 ... /rpool에서 마운트됩니다. .../rpool/export 및 .../rpool/export/home 데이터 세트는 /export 및 /export/home에서 마운트됩니다. 이러한 비전역 영역 데이터 세트는 해당 전역 영역 데이터 세트와 동일하게 사용되며 동일한 방식으로 관리할 수 있습니다. 영역 관리자는 .../rpool, .../rpool/export 및 .../rpool/export/home 데이터 세트 내에서 추가 데이터 세트를 만들 수 있습니다.

영역의 rpool/ROOT 파일 시스템에서 시작되는 계층 내에서 파일 시스템을 만들거나 삭제하거나 이름을 바꾸기 위해 **zfs(1M)** 매뉴얼 페이지에 설명된 zfs 명령을 사용하면 안 됩니다. zfs 명령은 canmount, mountpoint, sharesmb, zoned, com.oracle.*:*, com.sun:* 및 org.opensolaris.*:* 외의 등록 정보를 설정하는 데 사용할 수 있습니다.

일반적으로 영역에 마운트된 파일 시스템에는 다음 항목이 포함됩니다.

- 가상 플랫폼이 초기화될 때 마운트된 파일 시스템 세트
- 응용 프로그램 환경 자체 내에서 마운트된 파일 시스템 세트

예를 들어 이러한 세트에는 다음과 같은 파일 시스템이 포함될 수 있습니다.

- canmount 등록 정보의 값이 yes인 legacy 또는 none 이외의 mountpoint를 사용하는 ZFS 파일 시스템
- 영역의 /etc/vfstab 파일에 지정된 파일 시스템
- AutoFS 및 AutoFS 트리거 마운트 autofs 등록 정보는 **sharectl(1M)**에 설명된 sharectl을 사용하여 설정됩니다.
- 영역 관리자가 명시적으로 수행하는 마운트

실행 중인 영역 내 파일 시스템 마운트 권한도 zonecfg fs-allowed 등록 정보를 통해 정의됩니다. 이 등록 정보는 zonecfg add fs 또는 add dataset 리소스를 사용하여 영역에 마운트된 파일 시스템에는 적용되지 않습니다. 기본적으로 영역의 기본 위임 데이터 세트

내 파일 시스템, `hsfs` 파일 시스템 및 네트워크 파일 시스템(예: NFS)의 마운트만 영역 내에서 허용됩니다.



주의 - 특정 제한은 응용 프로그램 환경 내에서 수행되는 기본값과 다른 마운트에만 적용됩니다. 이러한 제한으로 인해 영역 관리자는 시스템의 나머지 부분에 대한 서비스를 거부하지 못합니다. 그렇지 않으면 다른 영역에 부정적인 영향을 미치게 됩니다.

영역 내 특정 파일 시스템 마운트와 관련하여 보안 제한이 있습니다. 영역에서 다른 파일 시스템이 마운트될 경우 특정 동작을 나타냅니다. 자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”](#)의 [“파일 시스템 및 비전역 영역”](#)을 참조하십시오.

데이터 세트에 대한 자세한 내용은 [datasets\(5\)](#) 매뉴얼 페이지를 참조하십시오. BE에 대한 자세한 내용은 [“Oracle Solaris 11.2 부트 환경 만들기 및 관리”](#)를 참조하십시오.

파일 시스템 마운트 및 업데이트

[pkg\(5\)](#) 매뉴얼 페이지에 설명된 대로 영역 시스템 이미지의 일부인 파일, 심볼릭 링크 또는 디렉토리를 숨기는 방식으로 파일 시스템을 마운트할 수 없습니다. 예를 들어 `/usr/local`에 콘텐츠를 제공하는 패키지가 설치되지 않은 경우 파일 시스템을 `/usr/local`에 마운트할 수 있습니다. 그러나 레거시 SVR4 패키지를 비롯한 패키지가 `/usr/local`로 시작되는 경로에 파일, 디렉토리 또는 심볼릭 링크를 제공하는 경우 파일 시스템을 `/usr/local`에 마운트할 수 없습니다. 파일 시스템을 `/mnt`에 임시로 마운트할 수 있습니다.

`/export`가 영역의 `rpool/export` 데이터 세트나 다른 위임된 데이터 세트에서 제공되는 경우 영역에서 파일 시스템이 마운트되는 순서로 인해 `fs` 리소스가 파일 시스템을 `/export/filesys`에 마운트할 수 없습니다.

영역의 호스트 ID

전역 영역의 `hostid`와는 다른 비전역 영역에 대한 `hostid` 등록 정보를 설정할 수 있습니다. 예를 들면 다른 시스템의 영역으로 마이그레이션된 시스템의 경우에 이와 같이 설정할 수 있습니다. 이제 영역 내 응용 프로그램은 원래의 `hostid`에 따라 다를 수 있습니다. 자세한 내용은 [“리소스 유형과 등록 정보” \[53\]](#)를 참조하십시오.

비전역 영역의 `/dev` 파일 시스템

`zonecfg` 명령은 규칙 일치 시스템을 사용하여 특정 영역에 표시되어야 할 장치를 지정합니다. 규칙 중 하나와 일치하는 장치가 영역의 `/dev` 파일 시스템에 포함됩니다. 자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”](#)의 [“영역 구성 방법”](#)을 참조하십시오.

비전역 영역의 이동식 lofi 장치

비전역 영역에서는 CD-ROM 장치처럼 작동하는 이동식 루프백 파일 lofi 장치를 구성할 수 있습니다. 장치가 매핑되는 파일을 변경하고 동일한 파일을 읽기 전용 모드로 사용할 여러 lofi 장치를 만들 수 있습니다. 이러한 유형의 lofi 장치는 lofiadm 명령을 -r 옵션과 함께 사용하여 만듭니다. 만들 때 파일 이름은 필요하지 않습니다. 이동식 lofi 장치의 수명 주기 동안 파일을 빈 장치와 연결하거나 비어 있지 않은 장치와의 연결을 해제할 수 있습니다. 파일을 동시에 여러 이동식 lofi 장치와 안전하게 연결할 수 있습니다. 이동식 lofi 장치는 읽기 전용입니다. 일반 읽기 쓰기 lofi 장치나 이동식 lofi 장치에 매핑된 파일을 다시 매핑할 수 없습니다. 잠재적 lofi 장치 수는 zone.max-lofi 리소스 제어로 제한되며, 이 리소스 제어는 전역 영역에서 zonecfg(1M)를 사용하여 설정할 수 있습니다.

생성된 후에는 이동식 lofi 장치는 읽기 전용입니다. 이동식 lofi 장치에 쓰기 작업이 수행되면 lofi 드라이버에서 오류를 반환합니다.

lofiadm 명령은 이동식 lofi 장치를 나열하는 데도 사용됩니다.

예 2-2 연결된 파일이 있는 이동식 lofi 장치 만들기

```
# lofiadm -r /path/to/file
/dev/lofi/1
```

예 2-3 빈 이동식 lofi 장치 만들기

```
# lofiadm -r
/dev/lofi/2
```

예 2-4 이동식 lofi 장치에 파일 삽입

```
# lofiadm -r /path/to/file /dev/lofi/1
/dev/lofi/1
```

자세한 내용은 [lofiadm\(1M\)](#), [zonecfg\(1M\)](#) 및 [lofi\(7D\)](#) 매뉴얼 페이지를 참조하십시오. [표 2-2](#). “영역 전체 리소스 제어”도 참조하십시오.

비전역 영역의 디스크 포맷 지원

zonecfg 도구를 통해 디스크 분할 및 uscsi 명령 사용을 가능하도록 설정합니다. 관련 예는 Resource Type Properties의 “[리소스 유형 등록 정보](#)” [58]를 참조하십시오. uscsi 명령에 대한 자세한 내용은 [uscsi\(7i\)](#)를 참조하십시오.

- 위임은 solaris 영역에 대해서만 지원됩니다.
- 디스크는 표시된 것처럼 -D 옵션과 prtconf 명령을 함께 사용하여 sd 대상을 사용해야 합니다. [prtconf\(1M\)](#)를 참조하십시오.

저장소 URI를 사용하는 커널 영역 장치 리소스

다음과 같은 지원이 제공됩니다.

- solaris-kz는 장치 리소스에서 bootpri 및 id 등록 정보를 지원합니다.
 - bootpri는 영역 루트 풀에 포함될 디스크에 대해서만 설정합니다. 영역 루트 풀에 포함되지 않는 디스크에 대해 bootpri를 설정할 경우 디스크 데이터가 손상될 수 있습니다.
 - id는 커널 영역에서 디스크 인스턴스를 제어합니다. 예를 들어 id=5는 디스크가 영역에서 c1d5임을 의미합니다.
- 부트 가능한 solaris-kz 디스크에 만들어진 루트 zpool은 설치 중 전역 영역으로 가져올 수 있습니다. 이때 루트 zpool은 zpool 명령을 사용하여 표시됩니다. 자세한 내용은 [zpool\(1M\)](#)을 참조하십시오.

구성 가능한 권한

영역이 부트될 때 구성에 기본 안전 권한 세트가 포함됩니다. 영역의 권한 부여된 프로세스는 시스템의 다른 비전역 영역 또는 전역 영역의 프로세스에 영향을 주지 않으므로 이러한 권한은 안전한 것으로 간주됩니다. zonecfg 명령을 사용하여 다음을 수행할 수 있습니다.

- 전역 리소스를 제어할 수 있게 됨에 따라 이러한 변경으로 인해 한 영역의 프로세스가 다른 영역의 프로세스에 영향을 줄 수 있음을 고려하여 기본 권한 세트에 추가합니다.
- 이러한 권한을 실행해야 하는 경우 이러한 변경으로 인해 일부 프로세스가 올바르게 작동하지 못할 수도 있다는 점을 고려하여 기본 권한 세트에서 제거합니다.

참고 - 영역의 기본 권한 세트에서 제거할 수 없는 권한이 있는 반면, 이러한 세트에 추가할 수 없는 권한도 있습니다.

자세한 내용은 “Oracle Solaris 영역 만들기 및 사용”의 “비전역 영역의 권한”, “Oracle Solaris 영역 만들기 및 사용”의 “영역 구성 방법” 및 [privileges\(5\)](#)를 참조하십시오.

리소스 풀 연결

“Oracle Solaris 11.2의 리소스 관리”의 13 장, “리소스 풀 만들기 및 관리 작업”의 설명에 따라 시스템에서 리소스 풀을 구성한 경우 pool 등록 정보를 사용하여 영역을 구성할 때 리소스 풀 중 하나를 영역에 연결할 수 있습니다.

dedicated-cpu 리소스를 사용하여 비전역 영역이 실행 중인 동안 시스템 프로세서의 하위 세트를 비전용 영역 전용으로 지정할 수 있습니다. dedicated-cpu 등록 정보를 사용하여 CPU, 코어 및 소켓을 영역에 지정할 수 있습니다. 영역이 실행 중인 동안 시스템에서 사용할 임시 풀을 동적으로 만듭니다. zonecfg를 통해 지정한 상태에서는 풀 설정이 마이그레이션 중에

전파됩니다. Oracle Solaris 커널 영역을 구성하는 경우 `virtual-cpu` 리소스도 참조하십시오.

`pool` 등록 정보를 사용하여 같은 풀을 공유하는 영역을 여러 개 구성할 수 있습니다.

참고 - `pool` 등록 정보를 통해 설정된 지속성 풀을 사용하는 영역 구성은 `dedicated-cpu` 리소스를 통해 구성된 임시 풀과 호환되지 않습니다. 이러한 두 가지 등록 정보 중 하나만 설정할 수 있습니다.

영역 전체 리소스 제어 설정

영역 관리자 또는 해당 권한을 가진 사용자는 영역에 대해 권한 부여된 영역 전체 리소스 제어를 설정할 수 있습니다. 영역 전체 리소스 제어는 영역 내에서 모든 프로세스 항목의 총 리소스 사용을 제한합니다.

이러한 제한은 `zonecfg` 명령을 사용하여 전역 및 비전역 영역 모두에 대해 지정됩니다. [“Oracle Solaris 영역 만들기 및 사용”의 “영역 구성 방법”](#)을 참조하십시오.

영역 전체 리소스 제어를 설정하기 위한 보다 간단한 기본적인 방법은 `rctl` 리소스(예: `cpu-cap`) 대신에 `capped-cpu`와 같은 등록 정보 이름 또는 리소스를 사용하는 것입니다.

`zone.cpu-cap` 리소스 제어는 영역에 사용될 수 있는 CPU 리소스의 양에 대한 절대적 제한을 설정합니다. 값 100은 한 CPU의 100%를 설정함을 의미합니다. 100%는 CPU 상한값을 사용할 때 시스템의 CPU 하나 전체에 해당하므로 125 값은 125%입니다.

참고 - `capped-cpu` 리소스를 설정할 때 단위로 십진수를 사용할 수 있습니다. 이 값은 `zone.cpu-cap` 리소스 제어와 상관되지만 설정이 100분의 1로 낮춰집니다. 설정 1은 리소스 제어의 설정 100에 해당합니다.

`zone.cpu-shares` 리소스 제어는 영역에 대해 FSS(Fair Share Scheduler) CPU 할당에 대한 제한을 설정합니다. 먼저 CPU 할당이 영역에 할당되고 나서 `project.cpu-shares` 항목에 지정된 대로 영역 내 여러 프로젝트에 세분됩니다. 자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”의 “영역이 설치된 Oracle Solaris 시스템에서 Fair Share Scheduler 사용”](#)을 참조하십시오. 이 컨트롤의 전역 등록 정보 이름은 `cpu-shares`입니다.

`zone.max-locked-memory` 리소스 제어는 영역에 사용할 수 있는 잠긴 물리적 메모리의 양을 제한합니다. 영역 내 프로젝트 전체에 대해 잠긴 메모리 리소스 할당은 `project.max-locked-memory` 리소스 제어를 사용하여 제어할 수 있습니다. 자세한 내용은 [“Oracle Solaris 11.2의 리소스 관리”의 “사용 가능한 리소스 제어”](#)를 참조하십시오.

`zone.max-lofi` 리소스 제어는 영역에 만들 수 있는 잠재적 `lofi` 장치 수를 제한합니다.

`zone.max-lwps` 리소스 제어는 한 영역의 너무 많은 LWP가 다른 영역에 영향을 주지 않도록 함으로써 리소스 격리를 향상시킵니다. 영역 내 프로젝트 간의 LWP 리소스 할당은

project.max-lwps 리소스 제어를 사용하여 제어할 수 있습니다. 자세한 내용은 [“Oracle Solaris 11.2의 리소스 관리”의 “사용 가능한 리소스 제어”](#)를 참조하십시오. 이 컨트롤의 전역 등록 정보 이름은 max-lwps입니다.

zone.max-processes 리소스 제어는 한 영역에서 너무 많은 프로세스 테이블 슬롯을 사용하여 다른 영역에 영향을 주지 못하게 함으로써 리소스 격리를 향상시킵니다. 영역 내 프로젝트 간의 프로세스 테이블 슬롯 리소스 할당은 [“Oracle Solaris 11.2의 리소스 관리”의 “사용 가능한 리소스 제어”](#)에 설명된 project.max-processes 리소스 제어를 사용하여 설정할 수 있습니다. 이 컨트롤의 전역 등록 정보 이름은 max-processes입니다. zone.max-processes 리소스 제어는 zone.max-lwps 리소스 제어를 포함할 수도 있습니다. zone.max-processes가 설정되고 zone.max-lwps가 설정되지 않은 경우 영역이 부트될 때 zone.max-lwps가 암시적으로 zone.max-processes 값의 10배로 설정됩니다. 일반 프로세스와 좀비 프로세스는 모두 프로세스 테이블 슬롯을 사용하기 때문에 max-processes 컨트롤은 좀비 프로세스가 프로세스 테이블을 소모하지 못하도록 방지합니다. 좀비 프로세스에는 정의상 LWP가 없으므로 max-lwps가 이 가능성으로부터 보호할 수 없습니다.

zone.max-msg-ids, zone.max-sem-ids, zone.max-shm-ids 및 zone.max-shm-memory 리소스 제어는 영역 내 모든 프로세스에 사용되는 시스템 V 리소스를 제한하는 데 사용됩니다. 영역 내 프로젝트 간의 시스템 V 리소스 할당은 이러한 리소스 제어의 프로젝트 버전을 사용하여 제어할 수 있습니다. 이러한 컨트롤의 전역 등록 정보 이름은 max-msg-ids, max-sem-ids, max-shm-ids 및 max-shm-memory입니다.

zone.max-swap 리소스 제어는 영역 내 tmpfs 마운트와 사용자 프로세스 주소 공간 매핑에 사용되는 스왑을 제한합니다. prstat -z의 출력에 SWAP 열이 표시됩니다. 보고된 스왑은 영역의 프로세스 및 tmpfs 마운트에서 사용된 총 스왑입니다. 이 값은 zone.max-swap 설정을 선택하는 데 사용할 수 있는 각 영역에 예약된 스왑을 모니터링하는 데 도움이 됩니다.

표 2-2 영역 전체 리소스 제어

제어 이름	전역 등록 정보 이름	설명	기본 단위	사용된 값
zone.cpu-cap		이 영역의 CPU 리소스 양에 대한 절대 제한	수량(CPU 수), 백분율로 표현됨 참고 - capped-cpu 리소스로 설정하는 경우 단위로 십진 수를 사용할 수 있습니다.	
zone.cpu-shares	cpu-shares	이 영역에 대한 FSS(Fair Share Scheduler) CPU 할당 수	수량(할당)	
zone.max-locked-memory		영역에 사용할 수 있는 전체 물리적 잠긴 메모리의 양입니다. priv_proc_lock_memory가 영역에 지정	크기(바이트)	capped-memory의 locked 등록 정보

제어 이름	전역 등록 정보 이름	설명	기본 단위	사용된 값
		된 경우 이 리소스 제어를 설정하여 해당 영역이 모든 메모리를 잠그는 것을 방지할 수 있습니다.		
zone.max-lofi	max-lofi	영역에서 생성할 수 있는 잠재적 lofi 장치 수에 대한 제한	수량(lofi 장치 수)	
zone.max-lwps	max-lwps	이 영역에 동시에 사용할 수 있는 최대 LWP 수	수량(LWP)	
zone.max-msg-ids	max-msg-ids	이 영역에 허용되는 최대 메시지 대기열 ID 수	수량(메시지 대기열 ID)	
zone.max-processes	max-processes	이 영역에 동시에 사용할 수 있는 최대 프로세스 테이블 슬롯 수	수량(프로세스 테이블 슬롯)	
zone.max-sem-ids	max-sem-ids	이 영역에 허용되는 최대 세마포 ID 수	수량(세마포 ID)	
zone.max-shm-ids	max-shm-ids	이 영역에 허용되는 공유 메모리 ID 수	수량(공유 메모리 ID)	
zone.max-shm-memory	max-shm-memory	이 영역에 허용되는 시스템 V 공유 메모리의 총량	크기(바이트)	
zone.max-swap		이 영역에 대한 사용자 프로세스 주소 공간 매핑 및 tmpfs 마운트에 사용할 수 있는 총 스왑 크기입니다.	크기(바이트)	capped-memory의 swap 등록 정보

prctl 명령을 사용하여 프로세스 실행을 위해 이러한 제한을 지정할 수 있습니다. 예는 [“Oracle Solaris 영역 만들기 및 사용”](#)의 [“prctl 명령을 사용하여 전역 영역에서 FSS 할당을 설정하는 방법”](#)에 나와 있습니다. prctl 명령을 통해 지정된 제한은 지속적이지 않습니다. 시스템을 재부트할 때까지만 제한이 적용됩니다.

영역에 대한 설명 포함

attr 리소스 유형을 사용하여 영역에 대한 설명을 추가할 수 있습니다. 자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”](#)의 [“영역 구성 방법”](#)을 참조하십시오.

zonecfg 명령 사용

zonecfg(1M) 매뉴얼 페이지에 설명되어 있는 zonecfg 명령은 비전역 영역을 구성하는 데 사용됩니다.

zonecfg 명령을 사용하여 전역 영역에 대한 리소스 관리 설정을 영구적으로 지정할 수도 있습니다. 예를 들면, dedicated-cpu 리소스를 사용하여 전용 CPU를 사용하도록 전역 영역을 구성하는 데 이 명령을 사용할 수 있습니다.

zonecfg 명령은 대화식 모드, 명령줄 모드 또는 명령 파일 모드로 사용할 수 있습니다. 이 명령을 사용하여 다음 작업을 수행할 수 있습니다.

- 영역 구성 만들기 또는 삭제(완전 삭제)
- 특정 구성에 리소스 추가
- 구성에 추가된 리소스에 대한 등록 정보 설정
- 특정 구성에서 리소스 제거
- 구성 쿼리 또는 확인
- 구성으로 완결
- 이전 구성으로 되돌리기
- 영역 이름 바꾸기
- zonecfg 세션 종료

zonecfg 프롬프트는 다음과 같은 형태입니다.

```
zonecfg:zonename>
```

파일 시스템과 같이 특정 리소스 유형을 구성할 때 해당 리소스 유형도 프롬프트에 포함됩니다.

```
zonecfg:zonename:fs>
```

이 장에서 설명한 다양한 zonecfg 구성 요소를 사용하는 방법을 보여 주는 절차를 비롯한 자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”의 1 장, “비전역 영역을 계획 및 구성하는 방법”](#)을 참조하십시오.

zonecfg 모드

범위의 개념은 사용자 인터페이스용으로 사용됩니다. 범위는 전역 범위이거나 리소스 특정 범위입니다. 기본 범위는 전역입니다.

전역 범위에서 add 하위 명령과 select 하위 명령은 특정 리소스를 선택하는 데 사용됩니다. 그러면 범위가 해당 리소스 유형으로 변경됩니다.

- add 하위 명령의 경우 리소스 지정을 완료하기 위해 end 또는 cancel 하위 명령이 사용됩니다.
- select 하위 명령의 경우 리소스 수정을 완료하는 데 end 또는 cancel 하위 명령이 사용됩니다.

그러면 범위가 전역으로 되돌려집니다.

add, remove 및 set과 같은 특정 하위 명령은 각 범위마다 의미가 다릅니다.

zonecfg 대화식 모드

대화식 모드에서는 다음 하위 명령이 지원됩니다. 하위 명령과 함께 사용되는 의미와 옵션에 대한 자세한 내용은 zonecfg(1M) 매뉴얼 페이지를 참조하십시오. 하위 명령으로 인해 완전히 삭제되거나 작업이 손실될 수 있는 경우 진행하기 전에 시스템에서 사용자 확인을 요청합니다. -f(강제) 옵션을 사용하여 이 확인을 생략할 수 있습니다.

help	일반 도움말을 인쇄하거나 지정된 리소스에 대한 도움말을 표시합니다. zonecfg:my-zone:capped-cpu> help
create	다음 목적 중 하나로 지정된 새 영역에 대한 메모리 내 구성을 구성하기 시작합니다. <ul style="list-style-type: none"> ■ Oracle Solaris 기본 설정을 새 구성에 적용하기 위해. 이 방법은 기본값입니다. ■ -t <i>template</i> 옵션을 사용하여 지정된 템플릿과 동일한 구성을 생성하기 위해. 영역 이름은 템플릿 이름을 새 영역 이름으로 변경됩니다. ■ -f 옵션을 사용하여 기존 구성을 덮어쓰기 위해. ■ -b 옵션을 사용하여 아무것도 설정되는 빈 구성을 생성하기 위해.
export	구성을 표준 출력이나 명령 파일에 사용할 수 있는 형태의 지정된 출력 파일로 인쇄합니다.
add	전역 범위에서 지정된 리소스 유형을 구성에 추가합니다. 리소스 범위에서 지정된 값과 지정된 이름의 등록 정보를 추가합니다. 자세한 내용은 “Oracle Solaris 영역 만들기 및 사용”의 “영역 구성 방법” 및 zonecfg(1M) 매뉴얼 페이지를 참조하십시오.
set	지정된 등록 정보 이름을 지정된 등록 정보 값으로 설정합니다. zonepath와 같은 일부 등록 정보는 전역이지만, 그 밖의 등록 정보는 리소스에 따라 달라집니다. 따라서 이 명령은 전역 및 리소스 범위 모두에서 사용할 수 있습니다.
select	전역 범위에서만 사용할 수 있습니다. 수정에 대한 지정된 등록 정보 이름-등록 정보 값 쌍 기준과 일치하는 지정된 유형의 리소스를 선택합니다. 범위가 해당 리소스 유형으로 변경됩니다. 리소스를 고유하게 식별할 충분한 수의 등록 정보 이름-값 쌍을 지정해야 합니다.
clear	선택적 설정의 값을 지웁니다. 필수 설정은 지울 수 없습니다. 하지만, 일부 설정은 새 값을 지정하여 변경할 수 있습니다. 등록 정보에 대해 clear 명령을 사용하면 등록 정보의 기본값으로 값이 지워집니다.

remove	전역 범위에서 지정된 리소스 유형을 제거합니다. 리소스를 고유하게 식별할 충분한 개수의 등록 정보 이름-값 쌍을 지정해야 합니다. 등록 정보 이름-값 쌍을 지정하지 않은 경우 모든 인스턴스가 제거됩니다. 두 개 이상 존재할 경우 -f 옵션을 사용하지 않는 한 확인이 필요합니다. 리소스 범위에서 현재 리소스로부터 지정된 등록 정보 이름-등록 정보 값을 제거합니다.
end	리소스 범위에서만 사용할 수 있습니다. 리소스 지정을 종료합니다. 그러면 zonecfg 명령이 현재 리소스가 완전히 지정되었음을 확인합니다. <ul style="list-style-type: none"> ■ 리소스가 완전히 지정된 경우 메모리 내 구성에 추가되고 범위가 전역으로 되돌려집니다. ■ 지정이 완전하지 않으면 시스템에서 수행해야 할 작업을 설명하는 오류 메시지가 표시됩니다.
cancel	리소스 범위에서만 사용할 수 있습니다. 리소스 지정을 종료하고 범위를 전역으로 재설정합니다. 부분적으로 지정된 리소스는 유지되지 않습니다.
delete	지정된 구성을 완전히 삭제합니다. 구성을 메모리와 안정된 저장소에서 모두 삭제합니다. -f(강제) 옵션을 delete와 함께 사용해야 합니다.



주의 - 이 작업은 즉시 수행됩니다. 완결이 필요하지 않으며 삭제된 영역을 되돌릴 수 없습니다.

info	전역 리소스 등록 정보 zonepath, autoboot 및 pool 또는 현재 구성에 대한 정보를 표시합니다. 리소스 유형이 지정된 경우 해당 유형의 리소스에 대해서만 정보를 표시합니다. 리소스 범위에서 이 하위 명령은 추가되거나 수정되는 리소스에만 적용됩니다.
verify	현재 구성의 정확성을 확인합니다. 모든 리소스에 모든 필수 등록 정보가 지정되었는지 확인합니다. rootzpool 리소스 그룹 및 해당 등록 정보의 구문을 확인합니다. URI로 지정된 저장소의 접근성은 확인되지 않습니다.
commit	메모리의 현재 구성을 안정된 저장소로 완결합니다. 메모리 내 구성이 완료될 때까지 revert 하위 명령을 사용하여 변경을 제거할 수 있습니다. zoneadm에 사용될 구성을 완결해야 합니다. zonecfg 세션을 완료할 때 이 작업이 자동으로 시도됩니다. 올바른 구성만 완결할 수 있으므로 완결 작업이 자동으로 확인을 수행합니다.
revert	구성을 마지막 완결 상태로 되돌립니다.
exit	zonecfg 세션을 종료합니다. exit를 -f(강제) 옵션과 함께 사용할 수 있습니다.

commit은 필요할 경우 자동으로 시도됩니다. EOF 문자를 사용하여 세션을 종료할 수도 있습니다.

zonecfg 명령 파일 모드

명령 파일 모드에서는 파일에서 입력을 받습니다. 이 파일을 생성하는 데 zonecfg Interactive Mode에 설명된 “zonecfg 대화식 모드” [51] 하위 명령이 사용됩니다. 구성을 표준 출력으로 인쇄하거나 -f 옵션을 사용하여 출력 파일을 지정할 수 있습니다.

영역 구성 데이터

영역 구성 데이터는 리소스와 등록 정보라는 두 종류의 엔티티로 구성됩니다. 각 리소스마다 유형이 지정되어 있으며 각 리소스에는 하나 이상의 등록 정보 세트가 있을 수도 있습니다. 등록 정보마다 이름과 값이 있습니다. 등록 정보 세트는 리소스 유형에 따라 달라집니다.

필수 등록 정보는 zonename과 zonepath뿐입니다.

리소스 유형과 등록 정보

다음은 각 리소스와 등록 정보 유형에 대한 설명입니다.

zonename	<p>영역 이름입니다. 영역 이름에는 다음 규칙이 적용됩니다.</p> <ul style="list-style-type: none"> ■ 각 영역마다 고유한 이름이 지정되어야 합니다. ■ 영역 이름은 대/소문자를 구분합니다. ■ 영역 이름은 영숫자 문자로 시작해야 합니다. <p>이름에는 영숫자 문자, 밑줄(_), 하이픈(-) 및 마침표(.)를 사용할 수 있습니다.</p> <ul style="list-style-type: none"> ■ 이름은 63자를 초과할 수 없습니다. ■ 이름 global은 전역 영역용으로 예약되었습니다. ■ sys로 시작하는 이름은 예약되어 있으므로 사용할 수 없습니다.
zonepath	<p>zonepath 등록 정보는 영역이 설치될 경로를 지정합니다. 각 영역에는 전역 영역의 루트 디렉토리를 기준으로, 해당 루트 디렉토리에 대한 경로가 있습니다. 설치 시 전역 영역 디렉토리의 표시 여부를 제한해야 합니다. 영역 경로는 root와 700 모드를 함께 사용하여 소유해야 합니다. 영역 경로는 존재하지 않을 경우 설치 시 자동으로 만들어집니다. 권한이 올바르지 않으면 자동으로 수정됩니다.</p> <p>비전역 영역의 루트 경로는 한 레벨 낮습니다. 영역의 루트 디렉토리는 전역 영역의 루트 디렉토리(/)와 동일한 소유권과 권한이 부여됩니다.</p>

다. 영역 디렉토리는 root와 755 모드를 함께 사용하여 소유해야 합니다. 이 계층을 사용하면 전역 영역의 권한이 없는 사용자가 비전역 영역의 파일 시스템을 탐색할 수 없습니다.

영역이 ZFS 데이터 세트에 있어야 합니다. ZFS 데이터 세트는 영역이 설치되거나 연결될 때 자동으로 만들어집니다. ZFS 데이터 세트를 만들 수 없을 경우 영역이 설치되거나 연결되지 않습니다.

경로	설명
/zones/my-zone	zonecfg zonepath
/zones/my-zone/root	영역 루트

자세한 내용은 [“Oracle Solaris 영역 만들기 및 사용”의 “파일 시스템 탐색”](#)을 참조하십시오.

zonecfg template 등록 정보에서 zonepath의 기본값은 /system/zones/*zonename*입니다.

참고 - zoneadm 명령의 move 하위 명령을 사용하여 새로운 전체 zonepath를 지정하면 영역을 동일한 시스템의 다른 위치로 이동할 수 있습니다. 지침은 [“Oracle Solaris 영역 만들기 및 사용”의 “비전역 영역 이동”](#)을 참조하십시오.

autoboot	<p>이 등록 정보가 true로 설정되면 전역 영역이 부트될 때 해당 영역이 자동으로 부트됩니다. 기본적으로는 false로 설정됩니다. 영역 서비스 svc:/system/zones:default를 사용 안함으로 설정하면 이 등록 정보의 설정과 상관없이 영역이 자동으로 부트됩니다. svcadm(1M) 매뉴얼 페이지에 설명된 svcadm 명령을 사용하여 영역 서비스를 사용으로 설정할 수 있습니다.</p> <pre>global# svcadm enable zones</pre> <p>pkg 업데이트 중 이 설정에 대한 자세한 내용은 “Oracle Solaris 영역 만들기 및 사용”의 “영역 패키징 개요”를 참조하십시오.</p>
bootargs	<p>이 등록 정보는 영역의 부트 인수를 설정하는 데 사용됩니다. 부트 인수는 reboot, zoneadm boot 또는 zoneadm reboot 명령에 의해 대체되지 않을 경우에 적용됩니다. <i>Zone Boot Arguments</i>를 참조하십시오.</p>
limitpriv	<p>이 등록 정보는 기본값 이외의 권한 마스크를 지정하는 데 사용됩니다. “Oracle Solaris 영역 만들기 및 사용”의 “비전역 영역의 권한”을 참조하십시오.</p> <p>선행 priv_를 지정하든 지정하지 않든 권한 이름을 지정하면 권한이 추가됩니다. 이름 앞에 대시(-) 또는 느낌표(!)를 붙이면 권한이 제외됩니다. 권한 값은 쉼표로 구분되며 따옴표(") 안에 지정됩니다.</p>

`priv_str_to_set(3C)`에 설명된 바와 같이 `none`, `all` 및 `basic`의 특수 권한 세트가 일반적인 정의로 확장됩니다. 전역 영역에서 영역 구성을 수행하므로 특수 권한 세트 `zone`를 사용할 수 없습니다. 일반적인 사용은 특정 권한을 추가하거나 제거하여 기본 권한 세트를 변경하는 것이므로 특수 세트 `default`가 기본 권한 세트로 매핑됩니다. `default`가 `limitpriv` 등록 정보의 시작 부분에 나타나면 기본 세트로 확장됩니다. 다음 항목은 영역에서 `dtrace_proc` 및 `dtrace_user` 권한만 필요로 하는 DTrace 프로그램을 사용할 수 있는 기능을 추가합니다.

```
global# zonecfg -z userzone
zonecfg:userzone> set limitpriv="default,dtrace_proc,dtrace_user"
```

영역의 권한 세트에 허용되지 않은 권한만 들어 있거나 필요한 권한이 없거나, 알 수 없는 권한이 포함된 경우 영역을 확인, 준비 또는 부트하려는 시도가 실패하며 오류 메시지가 표시됩니다.

scheduling-class	이 등록 정보는 영역의 예약 클래스를 설정합니다. 자세한 내용과 팁은 “예약 클래스” [34] 를 참조하십시오.
ip-type	이 등록 정보는 모든 비전역 영역에 대해 설정해야 합니다. “배타적 IP 비전역 영역” [41] , “공유 IP 비전역 영역” [40] 및 “Oracle Solaris 영역 만들기 및 사용” 의 “영역 구성 방법” 을 참조하십시오.
dedicated-cpu	이 리소스는 영역이 실행 중인 동안 시스템 프로세서의 하위 세트를 해당 영역 전용으로 사용합니다. <code>dedicated-cpu</code> 리소스는 <code>ncpus</code> 및 <code>importance</code> , <code>ncores</code> , <code>cores</code> , <code>sockets</code> (선택 사항)에 대한 제한을 제공합니다. 자세한 내용은 “dedicated-cpu 리소스” [32] 를 참조하십시오.
solaris-kz만 해당: virtual-cpu	이 <code>solaris-kz</code> 리소스는 영역이 실행 중인 동안 시스템 프로세서의 하위 세트를 해당 영역 전용으로 사용합니다. <code>virtual-cpu</code> 리소스는 <code>ncpus</code> 에 대한 제한을 제공합니다. 자세한 내용은 “solaris-kz만 해당: virtual-cpu 리소스” [33] 를 참조하십시오.
capped-cpu	이 리소스는 영역이 실행 중인 동안 사용될 수 있는 CPU 리소스 양에 대한 제한을 설정합니다. <code>capped-cpu</code> 리소스는 <code>ncpus</code> 에 대한 제한을 제공합니다. 자세한 내용은 “capped-cpu 리소스” [34] 를 참조하십시오.
capped-memory	이 리소스는 영역에 대한 메모리를 제한할 때 사용되는 등록 정보를 그룹화합니다. <code>capped-memory</code> 리소스는 <code>physical</code> , <code>swap</code> 및 <code>locked</code> 메모리에 대한 제한을 제공합니다. 이러한 등록 정보 중 하나 이상을 지정해야 합니다. <code>capped-memory</code> 리소스를 사용하려면 <code>service/resource-cap</code> 패키지를 전역 영역에 설치해야 합니다.
anet	<code>anet</code> 리소스는 영역이 부트될 때 배타적 IP 영역의 임시 VNIC 인터페이스를 자동으로 만들고 영역이 정지할 때 해당 VNIC를 삭제합니다.

net	<p>net 리소스는 전역 영역의 기존 네트워크 인터페이스를 비전역 영역에 지정합니다. 네트워크 인터페이스 리소스는 인터페이스 이름입니다. 각 영역에는 영역이 설치된 상태에서 준비 상태로 전환될 때 설정되는 네트워크 인터페이스가 있을 수 있습니다.</p>
dataset	<p>데이터 세트는 파일 시스템, 볼륨 또는 스냅샷을 지칭하는 일반 용어입니다. ZFS™ 데이터 세트 리소스를 추가하면 저장소 관리를 비전역 영역으로 위임할 수 있습니다. 위임된 데이터 세트가 파일 시스템인 경우 영역 관리자가 데이터 세트 내 파일 시스템을 생성 및 완전 삭제할 수 있으며 데이터 세트의 등록 정보를 수정할 수 있습니다. 영역 관리자는 스냅샷, 자식 파일 시스템 및 볼륨, 그리고 그 후손의 복제본을 만들 수 있습니다. 위임된 데이터 세트가 볼륨인 경우 영역 관리자가 등록 정보를 설정하고 스냅샷을 만들 수 있습니다. 영역 관리자는 영역에 지정된 데이터 세트에 설정되어 있는 최상위 할당량을 초과하거나 영역에 추가되지 않은 데이터 세트에 영향을 줄 수 없습니다. 데이터 세트가 비전역 영역으로 위임된 후에는 zoned 등록 정보가 자동으로 설정됩니다. zoned 파일 시스템은 영역 관리자가 마운트 지점을 허용할 수 없는 값으로 설정해야 할 수 있으므로 전역 영역에서 마운트할 수 없습니다. 다음과 같은 방식으로 ZFS 데이터 세트를 영역에 추가할 수 있습니다.</p> <ul style="list-style-type: none"> ■ lofs 마운트된 파일 시스템으로(전역 영역과의 공간 공유가 유일한 목표인 경우) ■ 위임된 데이터 세트로 <p>zonecfg template 등록 정보를 사용할 때 rootzpool 리소스를 지정하지 않은 경우 기본 영역 경로 데이터 세트는 <i>rootpool/VARSHARE/zones/zonename</i>입니다. 데이터 세트는 마운트 지점이 <i>/system/zones</i>인 svc-zones 서비스를 통해 만들어집니다. 나머지 등록 정보는 <i>rootpool/VARSHARE/zones/</i>에서 상속됩니다.</p> <p>“Oracle Solaris 11.2의 ZFS 파일 시스템 관리”의 9 장, “Oracle Solaris ZFS 고급 주제”, “Oracle Solaris 영역 만들기 및 사용”의 “파일 시스템 및 비전역 영역” 및 datasets(5) 매뉴얼 페이지를 참조하십시오.</p> <p>데이터 세트 문제에 대한 자세한 내용은 “Oracle Solaris 영역 만들기 및 사용”의 13 장, “그 밖의 기타 Oracle Solaris 영역 문제 해결”도 참조하십시오.</p>
fs	<p>각 영역에는 영역이 설치된 상태에서 준비 상태로 전환될 때 마운트되는 다양한 파일 시스템이 있을 수 있습니다. 파일 시스템 리소스는 파일 시스템 마운트 지점의 경로를 지정합니다. 영역에서 파일 시스템 사용에 대한 자세한 내용은 “Oracle Solaris 영역 만들기 및 사용”의 “파일 시스템 및 비전역 영역”을 참조하십시오.</p>

참고 - fs 리소스를 통해 비전역 영역에서 UFS 파일 시스템을 사용하려면 설치 후에 또는 SI 매니페스트 스크립트를 통해 `system/file-system/ufs` 패키지를 영역 안에 설치해야 합니다.

`quota(1M)`에 설명된 `quota` 명령을 사용하여 fs 리소스를 통해 추가된 UFS 파일 시스템의 할당량 정보를 검색할 수 없습니다.

<code>fs-allowed</code>	<p>이 등록 정보를 설정하면 영역 관리자는 자신이 생성했거나 NFS를 사용하여 가져온 해당 유형의 파일 시스템을 마운트하고 해당 파일 시스템을 관리할 수 있습니다. <code>fs-allowed</code> 등록 정보를 통해 실행 중인 영역 내 파일 시스템 마운트 권한도 제한됩니다. 기본적으로 영역 내에서는 <code>hsfs</code> 파일 시스템 및 네트워크 파일 시스템(예: NFS)의 마운트만 허용됩니다.</p> <p>이 등록 정보는 영역 안에 위임된 ZVOL 장치나 블록 장치와 함께 사용할 수도 있습니다.</p> <p><code>fs-allowed</code> 등록 정보는 <code>ufs,pcfs</code>와 같은 영역 내에서 마운트할 수 있는 심표로 구분된 추가 파일 시스템 목록을 받습니다.</p> <pre>zonecfg:my-zone> set fs-allowed=ufs,pcfs</pre> <p>이 등록 정보는 <code>add fs</code> 또는 <code>add dataset</code> 등록 정보를 통해 전역 영역에서 관리되는 영역 마운트에만 적용됩니다.</p> <p>보안 고려 사항은 “Oracle Solaris 영역 만들기 및 사용”의 “파일 시스템 및 비전역 영역” 및 “Oracle Solaris 영역 만들기 및 사용”의 “비전역 영역에서 장치 사용”을 참조하십시오.</p>
<code>device</code>	<p><code>zonefigdevice</code> 리소스는 가상 디스크를 비전역 영역의 플랫폼에 추가하는 데 사용됩니다. 장치 리소스는 장치와 일치하는 지정자입니다. 각 영역에는 영역이 설치된 상태에서 준비 상태로 전환될 때 구성해야 하는 장치가 있을 수 있습니다.</p>

참고 - `device` 리소스를 통해 비전역 영역에서 UFS 파일 시스템을 사용하려면 `system/file-system/ufs` 패키지를 설치 후에 또는 SI 매니페스트 스크립트를 통해 영역 안에 설치해야 합니다.

<code>pool</code>	<p>이 등록 정보는 시스템의 리소스 풀과 영역을 연결하는 데 사용됩니다. 여러 영역이 풀 하나의 리소스를 공유할 수 있습니다. “dedicated-cpu 리소스” [32]를 참조하십시오.</p>
<code>rctl</code>	<p><code>rctl</code> 리소스는 영역 전체 리소스 제어용으로 사용됩니다. 컨트롤은 영역이 설치된 상태에서 준비 상태로 전환될 때 사용으로 설정됩니다. 자세한 내용은 “영역 전체 리소스 제어 설정” [47]을 참조하십시오.</p>

참고 - rctl 리소스 대신 zonefig의 set *global_property_name* 하위 명령을 사용하여 영역 전체 제어를 구성하려면 “Oracle Solaris 영역 만들기 및 사용”의 “영역 구성 방법”을 참조하십시오.

attr 이 일반 속성은 사용자 설명이나 다른 부속 시스템용으로 사용할 수 있습니다. attr의 name 등록 정보는 영숫자 문자로 시작되어야 합니다. name 등록 정보에는 영숫자 문자, 하이픈(-) 및 마침표(.)가 포함될 수 있습니다. zone.로 시작하는 속성 이름은 시스템용으로 예약되어 있습니다.

리소스 유형 등록 정보

리소스에도 구성할 등록 정보가 있습니다. 다음 등록 정보는 표시된 리소스 유형과 연결됩니다.

admin 지정된 영역에 대한 해당 사용자의 권한과 사용자 이름을 정의합니다.

```
zonecfg:my-zone> add admin
zonecfg:my-zone:admin> set user=zadmin
zonecfg:my-zone:admin> set auths=login,manage
zonecfg:my-zone:admin> end
```

다음 값은 auths 등록 정보에 사용할 수 있습니다.

- login(solaris.zone.login)
- manage(solaris.zone.manage)
- clone(solaris.zone.clonefrom)

이러한 auths 등록 정보를 사용하면 영역을 생성할 수 없습니다. 이 기능은 영역 보안 프로파일에 포함되어 있습니다.

solaris 및 solaris10만 해당:
rootzpool

storage

영역 설치를 위한 전용 ZFS zpool을 제공할 저장소 객체 URI를 식별합니다. storage에 대한 URI 및 허용되는 값은 “solaris 및 solaris10만 해당:rootzpool 리소스” [35]를 참조하십시오. 영역 설치 중 zpool을 자동으로 만들거나 미리 만든 zpool을 가져옵니다. 이름 *my-zone_rpool*이 지정됩니다.

```
zonecfg:my-zone> add rootzpool
zonecfg:my-zone:rootzpool> add storage dev:dsk/c4t1d0
zonecfg:my-zone:rootzpool> end
```

미러링된 구성을 만드는 경우 storage 등록 정보를 추가할 수 있습니다.

```
add storage dev:dsk/c4t1d0
```

solaris 및 solaris10만 해당: zpool	<pre>add storage dev:dsk/c4t3d0</pre> <p>영역에 대해 rootzpool 리소스를 하나만 구성할 수 있습니다.</p> <p>storage, name</p> <p>하나 이상의 저장소 객체 URI를 정의하여 zpool을 영역에 위임합니다. storage 등록 정보에 대한 URI 및 허용되는 값은 “solaris 및 solaris10만 해당:rootzpool 리소스” [35]를 참조하십시오. name 등록 정보에 허용되는 값은 zpool(1M) 매뉴얼 페이지에 정의되어 있습니다.</p> <p>이 예에서는 zpool 저장소 리소스가 영역에 위임됩니다. 설치 중 zpool을 자동으로 만들거나 이전에 만든 zpool을 가져옵니다. zpool 이름은 my-zone_pool1입니다.</p> <pre>zonecfg:my-zone> add zpool zonecfg:my-zone:zpool> set name=pool1 zonecfg:my-zone:zpool> add storage dev:dsk/c4t2d0 zonecfg:my-zone:zpool> add storage dev:dsk/c4t4d0 zonecfg:my-zone:zpool> end</pre> <p>영역 구성에 하나 이상의 zpool 리소스를 포함할 수 있습니다.</p>
dedicated-cpu	<p>ncpus, importance, cores, cpus, sockets</p> <p>CPU 수와 풀의 상대적 중요성(옵션)을 지정합니다. 다음 예는 my-zone 영역에서 사용할 CPU 범위를 지정합니다. importance도 설정되어 있습니다.</p> <pre>zonecfg:my-zone> add dedicated-cpu zonecfg:my-zone:dedicated-cpu> set ncpus=1-3 zonecfg:my-zone:dedicated-cpu> set importance=2 zonecfg:my-zone:dedicated-cpu> end</pre> <p>코어 0, 1, 2 및 3을 my-zone 영역에 영구적으로 지정하십시오. 다음 dedicated-cpu 예에서는 cores가 사용되지만, cpus=, cores= 및 sockets=를 모두 사용할 수 있습니다.</p> <pre>zonecfg:my-zone> add dedicated-cpu zonecfg:my-zone:dedicated-cpu> set cores=0-3 zonecfg:my-zone:dedicated-cpu> end</pre>
virtual-cpu	<p>ncpus</p> <p>CPU 수를 지정합니다. 다음 예에서는 my-zone 영역에 대해 3개의 CPU를 지정합니다.</p> <pre>zonecfg:my-zone> add virtual-cpu zonecfg:my-zone:dedicated-cpu> set ncpus=3 zonecfg:my-zone:dedicated-cpu> end</pre>
capped-cpu	ncpus

CPU 수를 지정합니다. 다음 예에서는 my-zone 영역에 대해 CPU 상한 값을 3.5개 CPU로 지정합니다.

```
zonecfg:my-zone> add capped-cpu
zonecfg:my-zone:capped-cpu> set ncpus=3.5
zonecfg:my-zone:capped-cpu> end
```

capped-memory

physical, swap, locked

my-zone 영역에 대한 메모리 제한을 지정합니다. 각 제한은 선택 사항이지만 하나 이상의 제한을 설정해야 합니다.

```
zonecfg:my-zone> add capped-memory
zonecfg:my-zone:capped-memory> set physical=50m
zonecfg:my-zone:capped-memory> set swap=100m
zonecfg:my-zone:capped-memory> set locked=30m
zonecfg:my-zone:capped-memory> end
```

capped-memory 리소스를 사용하려면 resource-cap 패키지를 전역 영역에 설치해야 합니다.

fs

dir, special, raw, type, options

fs 리소스 매개변수는 파일 시스템을 마운트하는 방법과 위치를 결정하는 값을 제공합니다. fs 매개변수는 다음과 같이 정의됩니다.

- dir 파일 시스템의 마운트 지점을 지정합니다.
- special 전역 영역에서 마운트할 블록 특수 장치 이름 또는 디렉토리를 지정합니다.
- raw 파일 시스템(ZFS에는 해당되지 않음)을 마운트하기 전에 fsck을 실행할 원시 장치를 지정합니다.
- type 파일 시스템 유형을 지정합니다.
- options mount 명령을 사용하여 찾은 것과 유사한 마운트 옵션을 지정합니다.

다음 예의 코드는 전역 영역 내 pool1/fs1 이름의 데이터 세트가, 구성되는 영역에서 /shared/fs1로 마운트되도록 지정합니다. 사용할 파일 시스템 유형은 ZFS입니다.

```
zonecfg:my-zone> add fs
zonecfg:my-zone:fs> set dir=/shared/fs1
zonecfg:my-zone:fs> set special=pool1/fs1
zonecfg:my-zone:fs> set type=zfs
zonecfg:my-zone:fs> end
```

매개변수에 대한 자세한 내용은 “Oracle Solaris 영역 만들기 및 사용”의 “o nosuid 옵션”, “Oracle Solaris 영역 만들기 및 사용”의 “보안 제

한 및 파일 시스템 동작” 및 `fsck(1M)` and `mount(1M)` 매뉴얼 페이지를 참조하십시오. 또한 섹션 1M 매뉴얼 페이지는 특정 파일 시스템에 고유한 마운트 옵션에도 사용할 수 있습니다. 이러한 매뉴얼 페이지의 이름은 `mount_filesystem` 형태로 되어 있습니다.

참고 - `quota(1M)`에 설명된 `quota` 명령을 사용하여 이 리소스를 통해 추가된 UFS 파일 시스템의 할당량 정보를 검색할 수 없습니다.

dataset name,
alias

name

다음 예의 코드는 데이터 세트 `sales`가 비전역 영역에서 표시되고 마운트되며 전역 영역에서는 더 이상 표시되지 않도록 지정합니다.

```
zonecfg:my-zone> add dataset
zonecfg:my-zone> set name=tank/sales
zonecfg:my-zone> end
```

위임된 데이터 세트에는 다음 예에 표시된 것처럼 기본값이 아닌 별칭이 있을 수 있습니다. 데이터 세트 별칭에는 슬래시(/)를 사용할 수 없습니다.

```
zonecfg:my-zone> add dataset
zonecfg:my-zone:dataset> set name=tank/sales
zonecfg:my-zone:dataset> set alias=data
zonecfg:my-zone:dataset> end
```

기본 별칭으로 되돌리려면 `clear alias`를 사용합니다.

```
zonecfg:my-zone> clear alias
```

anet

linkname, lower-link, allowed-address, auto-mac-address, configure-allowed-address, defrouter, linkmode(IPoIB), mac-address(비 IPoIB), mac-slot(비 IPoIB), mac-prefix(비 IPoIB), mtu, maxbw, pkey(IPoIB), priority, vlan-id(비 IPoIB), rxfanout, rxrings, txrings, link-protection, allowed-dhcp-cids

solaris만 해당: zonecfg에서 IPoIB 데이터 링크에 대해 다음 anet 등록 정보를 설정하지 마십시오.

- mac-address
- mac-prefix
- mac-slot
- vlan-id

zonecfg에서 비 IPoIB 데이터 링크에 대해 다음 anet 등록 정보를 설정하지 마십시오.

- linkmode
- pkey

EVS anet 리소스에 대해 다음 등록 정보만 설정하십시오.

- linkname
- evs
- vport
- configure-allowed-address

anet 리소스는 영역이 부트될 때 자동 VNIC 인터페이스 또는 IPoIB 인터페이스를 생성하고 영역이 정지할 때 VNIC 또는 IPoIB 인터페이스를 삭제합니다. solaris-kz 브랜드는 IPoIB를 지원하지 않습니다. 리소스 등록 정보는 zonecfg 명령을 통해 관리됩니다. 사용할 수 있는 등록 정보에 대한 전체 텍스트는 [zonecfg\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

lower-link	<p>생성해야 할 기본 링크를 지정합니다. auto로 설정한 경우 영역이 부트될 때마다 zoneadmd 데몬이 VNIC이 생성될 링크를 자동으로 선택합니다. VNIC을 만들 수 있는 링크를 anet 리소스에 대한 lower-link로 지정할 수 있습니다.</p> <p>부트 중 VNIC을 자동으로 만들기 위해 데이터 링크를 선택할 때 모든 IPoIB 링크를 건너뛸니다.</p>
linkname	<p>자동으로 생성된 VNIC 인터페이스 또는 IPoIB 인터페이스의 이름을 지정합니다. solaris-kz는 IPoIB를 지원하지 않습니다.</p>
mac-address(IPoIB에 해당되지 않음)	<p>지정된 값이나 키워드에 따라 VNIC MAC 주소를 설정합니다. 키워드가 아닌 값은 단일 유형 변환 MAC 주소로 해석됩니다. 지원되는 키워드는 zonecfg(1M) 매뉴얼 페이지를 참조하십시오. 임의 MAC 주소를 선택한 경우 영역 부트, 영역 분리 및 연결 작업 간에 생성된 주소가 유지됩니다. 기본 정책 auto-mac-address를 사용하는 경우 Oracle Solaris 영역에서 무작위 mac-address를 가져올 수 있습니다.</p>
pkey(IPoIB에만 해당)	<p>IPoIB 데이터 링크 인터페이스를 만들기 위해 사용할 분할 영역 키를 설정합니다. 필수 등록 정보입니다. 지정된 pkey는 0x 접두어가 있는지 여부와 상관없이 항상 16진수로 처리됩니다.</p>
linkmode(IPoIB에만 해당)	<p>데이터 링크 인터페이스에 대한 linkmode를 설정합니다. 기본값은 cm입니다. 유효한 값은 다음과 같습니다.</p>

cm(기본값)	연결됨 모드. 이 모드에서는 기본 65520바이트의 MTU를 사용하고 최대 65535바이트의 MTU를 지원합니다.
ud	신뢰할 수 없는 데이터그램 모드. 원격 노드에 대해 연결됨 모드를 사용할 수 없는 경우 대신 신뢰할 수 없는 데이터그램 모드가 자동으로 사용됩니다. 이 모드에서는 기본 2044바이트의 MTU를 사용하고 최대 4092바이트의 MTU를 지원합니다.
allowed-address	배타적 IP 영역의 IP 주소를 구성하고 배타적 IP 영역에 사용할 수 있는 구성 가능한 IP 주소 세트를 제한합니다. 여러 주소를 지정하려면 쉼표로 구분된 IP 주소를 사용합니다.
defrouter	비전역 영역과 전역 영역이 개별 네트워크에 상주하는 경우 defrouter 등록 정보를 사용하여 기본 경로를 설정할 수 있습니다. defrouter 등록 정보가 설정된 모든 영역은 전역 영역에 대해 구성되지 않은 서브넷에 있어야 합니다.
<p>zonecfg 명령이 SYSdefault 템플릿을 사용하여 영역을 만들 때 다른 IP 리소스를 설정하지 않은 경우 다음 등록 정보를 가진 anet 리소스가 영역 구성에 자동으로 포함됩니다. linkname은 물리적 이더넷 링크를 통해 자동으로 만들어지고 netN, net 0 형태의 첫번째 사용 가능한 이름으로 설정됩니다. 기본값을 변경하려면 zonecfg 명령을 사용합니다.</p> <p>기본 정책 auto를 사용하는 경우 적합한 mac-address가 지정됩니다.</p>	
Oracle Solaris 영역	무작위 mac-address
Oracle Solaris 커널 영역	무작위 mac-address
커널 영역 아래의 Oracle Solaris 영역	출하 시 mac-address

Oracle VM Server 출하 시 mac-address
for SPARC 게스트 도
메인

Oracle VM Server 출하 시 mac-address
for SPARC 게스트 도
메인에서 실행 중인
Oracle Solaris 커널
영역

기본 정책은 물리적 이더넷 링크(예: net0)를 통해 자동 VNIC를 만들고 MAC 주소를 VNIC에 지정합니다. 선택적 lower-link 등록 정보는 기본 링크인 vnic1로 설정됩니다. 자동 VNIC은 이 링크를 통해 만들어야 합니다. 링크 이름, 기본 물리적 링크, MAC 주소, 대역폭 제한과 같은 VNIC 등록 정보와 그 밖의 VNIC 등록 정보는 zonecfg 명령을 사용하여 지정할 수 있습니다. ip-type=exclusive도 지정해야 합니다.

```
zonecfg:my-zone> set ip-type=exclusive
zonecfg:my-zone> add anet
zonecfg:my-zone:anet> set linkname=net0
zonecfg:my-zone:anet> set lower-link=auto
zonecfg:my-zone:anet> set mac-address=random
zonecfg:my-zone:anet> set link-protection=mac-nospoof
zonecfg:my-zone:anet> end
```

다음 예에서는 IB 분할 영역 키 0xffff를 사용하는 물리적 링크 net5를 통해 IPoIB 데이터 링크 인터페이스로 구성된 solaris 브랜드 영역을 보여줍니다.

```
zonecfg:my-zone> set ip-type=exclusive
zonecfg:my-zone:anet> add anet
zonecfg:my-zone:anet> set linkname=ib0
zonecfg:my-zone:anet> set lower-link=net5
zonecfg:my-zone:anet> set pkey=0xffff
zonecfg:my-zone:anet> end
```

등록 정보에 대한 자세한 내용은 [zonecfg\(1M\)](#) 매뉴얼 페이지를 참조하십시오. 링크 등록 정보에 대한 자세한 내용은 [dladm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

net

address, allowed-addressphysical , defrouter

참고 - 공유 IP 영역의 경우 IP 주소와 물리적 장치를 둘 다 지정해야 합니다. 또는 기본 라우터를 설정할 수도 있습니다.

배타적 IP 영역의 경우에는 물리적 인터페이스만 지정하면 됩니다.

- `allowed-address` 등록 정보는 배타적 IP 영역에 사용될 수 있는 구성 가능한 IP 주소 세트를 제한합니다.
- 비전역 영역과 전역 영역이 개별 네트워크에 상주하는 경우 `defrouter` 등록 정보를 사용하여 기본 경로를 설정할 수 있습니다.
- `defrouter` 등록 정보가 설정된 모든 영역은 전역 영역에 대해 구성되지 않은 서브넷에 있어야 합니다.
- 기본 라우터를 사용하는 영역의 트래픽은 대상 영역으로 들어가기 전에 라우터를 거칩니다.

다양한 서브넷에 공유 IP 영역이 존재하는 경우 전역 영역에서 데이터 링크를 구성하지 마십시오.

공유 IP 영역에 대한 다음 예에서는 IP 주소가 192.168.0.1인 영역에 물리적 인터페이스인 `nge0`이 추가됩니다. 시스템에서 네트워크 인터페이스를 나열하려면 다음을 입력합니다.

```
global# ipadm show-if -po ifname,class,active,persistent
lo0:loopback:yes:46--
nge0:ip:yes:----
```

루프백 행이 아닌 각각의 출력 행에는 네트워크 인터페이스의 이름이 있습니다. 설명에 `loopback`을 포함하는 행은 카드에 적용되지 않습니다. 46 지속성 플래그는 인터페이스가 전역 영역에서 지속적으로 구성됨을 나타냅니다. `yes` 활성 값은 인터페이스가 현재 구성되어 있음을 나타내며, `ip`의 클래스 값은 `nge0`이 비루프백 인터페이스임을 나타냅니다. 해당 영역에 대한 기본 경로는 10.0.0.1로 설정됩니다. 필요에 따라 `defrouter` 등록 정보를 설정할 수 있습니다. `ip-type=shared`는 필수입니다.

```
zonecfg:my-zone> set ip-type=shared
zonecfg:my-zone> add net
zonecfg:my-zone:net> set physical=vnic1
zonecfg:my-zone:net> set address=192.168.0.1
zonecfg:my-zone:net> set defrouter=10.0.0.1
zonecfg:my-zone:net> end
```

배타적 IP 영역에 대한 다음 예에서는 VNIC가 VLAN인 물리적 인터페이스에 사용됩니다. 사용할 수 있는 데이터 링크를 확인하려면 `dladm show-link`를 사용합니다. `allowed-address` 등록 정보는 영역에서 사용할 수 있는 IP 주소를 제한합니다. `defrouter` 등록 정보는 기본 경로를 설정하는 데 사용됩니다. `ip-type=exclusive`도 지정해야 합니다.

```
zonecfg:my-zone> set ip-type=exclusive
```

```

zonecfg:my-zone> add net
zonecfg:myzone:net> set allowed-address=10.1.1.32/24
zonecfg:my-zone:net> set physical=vnic1
zonecfg:myzone:net> set defrouter=10.1.1.1
zonecfg:my-zone:net> end

```

물리적 장치 유형만 add net 단계에서 지정됩니다. physical 등록 정보는 VNIC일 수 있습니다.

참고 - Oracle™ Solaris 운영 체제에서는 모든 이더넷 유형의 인터페이스를 지원하므로 이러한 인터페이스의 데이터 링크를 dladm 명령으로 관리할 수 있습니다.

device

match, allow-partition, allow-raw-io

일치시킬 장치 이름은 일치시켜야 할 패턴이거나 절대 경로일 수 있습니다. allow-partition 및 allow-raw-io 둘 다 true 또는 false로 설정할 수 있습니다. 기본값은 false입니다. allow-partition은 분할을 사용으로 설정합니다. allow-raw-io는 uscsi를 사용으로 설정합니다. 이러한 리소스에 대한 자세한 내용은 [zonecfg\(1M\)](#)을 참조하십시오.

solaris-kz 영역에 대한 device:match 리소스 등록 정보에 지정할 수 있는 제한 사항은 다음과 같습니다.

- LUN당 한 개의 리소스만 허용됩니다.
- 슬라이스 및 분할 영역은 지원되지 않습니다.
- 원시 디스크 장치에 대해서만 지원이 제공됩니다.
- 지원되는 장치 경로는 lofi, ramdisk, dsk 및 zvols입니다.

다음 예에서는 디스크 장치에 대한 uscsi 작업이 solaris 영역 구성에 포함됩니다.

```

zonecfg:my-zone> add device
zonecfg:my-zone:device> set match=/dev/*dsk/cXtYdZ*
zonecfg:my-zone:device> set allow-raw-io=true
zonecfg:my-zone:device> end

```

Veritas 볼륨 관리자 장치는 add device를 사용하여 비전역 영역에 위임됩니다.

다음 예에서는 저장 장치가 solaris-kz 영역에 추가됩니다.

```

zonecfg:my-zone> add device
zonecfg:my-zone:device> set storage=iscsi:///
lunname.naa.600144f03d70c8000004ea57da1001
zonecfg:my-zone:device> set bootpri=0
zonecfg:my-zone:device> end

```



주의 - 장치를 추가하기 전에 제한 사항 및 보안 문제에 대해 “Oracle Solaris 영역 만들기 및 사용”의 “비전역 영역에서 장치 사용”, “Oracle Solaris 영역 만들기 및 사용”의 “비전역 영역에서 실행 중인 응용 프로그램” 및 “Oracle Solaris 영역 만들기 및 사용”의 “비전역 영역의 권한”을 참조하십시오.

`rctl` name, value
다음과 같은 영역 전체 리소스 제어를 사용할 수 있습니다.

- `zone.cpu-cap`
- `zone.cpu-shares`(기본: `cpu-shares`)
- `zone.max-locked-memory`
- `zone.max-lofi`
- `zone.max-lwps`(기본: `max-lwps`)
- `zone.max-msg-ids`(기본: `max-msg-ids`)
- `zone.max-processes`(기본: `max-processes`)
- `zone.max-sem-ids`(기본: `max-sem-ids`)
- `zone.max-shm-ids`(기본: `max-shm-ids`)
- `zone.max-shm-memory`(기본: `max-shm-memory`)
- `zone.max-swap`

영역 전체 리소스 제어를 설정하기 위한 보다 간단한 기본적인 방법은 “Oracle Solaris 영역 만들기 및 사용”의 “영역 구성 방법”에 표시된 것과 같이, `rctl` 리소스 대신 등록 정보 이름을 사용하는 것입니다. `add rctl`을 사용하여 영역 내 영역 전체 리소스 제어 항목을 구성한 경우 형식이 `project` 데이터베이스의 리소스 제어 항목과는 다릅니다. 영역 구성에서 `rctl` 리소스 유형은 세 개의 이름/값 쌍으로 구성됩니다. 이름은 `priv`, `limit` 및 `action`입니다. 각 이름은 단순 값을 받습니다.

```
zonecfg:my-zone> add rctl
zonecfg:my-zone:rctl> set name=zone.cpu-shares
zonecfg:my-zone:rctl> add value
(priv=privileged,limit=10,action=none)
zonecfg:my-zone:rctl> end

zonecfg:my-zone> add rctl
zonecfg:my-zone:rctl> set name=zone.max-lwps
zonecfg:my-zone:rctl> add value
(priv=privileged,limit=100,action=deny)
zonecfg:my-zone:rctl> end
```

리소스 제어 및 속성에 대한 일반 정보는 “Oracle Solaris 11.2의 리소스 관리”의 6 장, “리소스 제어 정보” 및 “Oracle Solaris 영역 만들기 및 사용”의 “비전역 영역에서 사용되는 리소스 제어”를 참조하십시오.

`attr` name, type, value

다음 예에서는 영역에 대한 설명이 추가됩니다.

```
zonecfg:my-zone> add attr
zonecfg:my-zone:attr> set name=comment
zonecfg:my-zone:attr> set type=string
zonecfg:my-zone:attr> set value="Production zone"
zonecfg:my-zone:attr> end
```

export 하위 명령을 사용하여 영역 구성을 표준 출력으로 인쇄할 수 있습니다. 구성은 명령 파일에서 사용할 수 있는 형태로 저장됩니다.

명령줄 편집 라이브러리

Tecla 명령줄 편집 라이브러리는 zonecfg 명령에 사용하기 위한 것입니다. 이 라이브러리는 명령줄 내역 및 편집 지원 방식을 제공합니다.

자세한 내용은 [tecla\(5\)](#) 매뉴얼 페이지를 참조하십시오.

용어집

동적 재구성	SPARC 기반 시스템에서 시스템이 실행 중인 동안 하드웨어를 다시 구성하는 기능입니다. DR이라고도 합니다.
리소스 관리	응용 프로그램의 시스템 리소스 사용을 제어할 수 있는 기능입니다.
리소스 분할 영역	리소스의 배타적 부분을 말합니다. 리소스의 모든 분할 영역의 합은 실행 중인 단일 Solaris 인스턴스에서 사용할 수 있는 리소스의 총 크기입니다.
리소스 소비자	기본적으로 Solaris 프로세스를 의미합니다. 프로젝트와 작업 등의 프로세스 모델 엔티티는 통합 리소스 사용량 측면에서 리소스 사용량을 다루는 방법을 제공합니다.
리소스 제어	리소스 사용량을 프로세스별, 작업별 또는 프로젝트별로 제한합니다.
리소스 풀	시스템 리소스를 분할하는 데 사용되는 구성 방식입니다. 리소스 풀은 분할할 수 있는 리소스 그룹 간의 연관을 나타냅니다.
메모리 상한값 적용 임계치	rcapd(Resource Capping Daemon)에서 상한값 적용을 트리거하는 시스템의 물리적 메모리 사용량 백분율입니다.
상한값	시스템 리소스 사용량에 대한 제한입니다.
이름 지정 서비스 데이터베이스	이 설명서의 프로젝트 및 작업(개요) 단원에 나오는 LDAP 컨테이너와 NIS 맵에 대한 참조입니다.
작업	리소스 관리 시 시간 경과에 따른 일련의 작업을 나타내는 프로세스 집합을 말합니다. 각 작업은 하나의 프로젝트와 연관됩니다.
작업 세트 크기	작업 세트의 크기입니다. 작업 세트는 처리 주기 중에 프로젝트 작업 부하에서 활발하게 사용하는 페이지 세트를 말합니다.
잠긴 메모리	페이징할 수 없는 메모리입니다.
전체 루트 영역	필요한 모든 시스템 소프트웨어와 추가 패키지가 영역의 개인 파일 시스템에 설치되는 비전역 영역 유형입니다.
확장 계정	Solaris 운영 체제에서 작업이나 프로세스를 기반으로 리소스 사용량을 기록할 수 있는 유연한 방법입니다.

auxiliary zone state(보조 영역 상태)	추가 상태 정보를 전역 영역에 전달하는 데 사용됩니다. zone state(영역 상태) 도 참조하십시오.
brand(브랜드)	BrandZ 기능의 인스턴스로, 응용 프로그램 실행에 사용되는 고유하지 않은 운영 환경을 포함하는 비전역 영역을 제공합니다.
branded zone(브랜드 영역)	비전역 영역에서 고유하지 않은 응용 프로그램을 실행할 격리된 환경입니다.
capping(상한 값 설정)	시스템 리소스 사용량에 대한 제한을 설정하는 프로세스입니다.
CMT resource(CMT 리소스)	CPUS, 코드 및 소켓을 가리킵니다.
CPU	영역 컨텍스트에서 하드웨어 스레드를 가리킵니다.
data-link(데이터 링크)	OSI 프로토콜 스택의 계층 2에 있는 인터페이스로, 시스템에서 STREAMS DLPI(v2) 인터페이스로 표현됩니다. 이 인터페이스는 TCP/IP 등 프로토콜 스택에서 연결될 수 있습니다. Oracle Solaris 10 영역 컨텍스트에서는 데이터 링크가 물리적 인터페이스, 통합 또는 VLAN 태그된 인터페이스입니다. 데이터 링크는 예를 들어 NIC 또는 VNIC를 참조할 때 물리적 인터페이스로 참조할 수도 있습니다.
default pool(기본 풀)	풀을 사용할 수 있는 경우 시스템에서 만든 풀입니다. 리소스 풀 을 참조하십시오.
default processor set(기본 프로세서 세트)	풀을 사용할 수 있는 경우 시스템에서 만든 프로세서 세트입니다. processor set(프로세서 세트) 를 참조하십시오.
disjoint(분리)	세트의 구성원이 겹치지 않고 중복되지 않는 세트 유형입니다.
dynamic configuration(동적 구성)	특정 시점에 지정된 시스템에 대한 리소스 풀 프레임워크 내에서 리소스 처리에 대한 정보입
Fair Share Scheduler	할당 수를 기반으로 CPU 시간을 할당할 수 있는 예약 클래스로 FSS라고도 합니다. 할당은 프로젝트에 할당되는 시스템의 CPU 리소스 양을 정의합니다.
FSS	Fair Share Scheduler 를 참조하십시오.
global administrator(전리자나 해당 권한이 부여된 사용자 역할 관리자)	루트 사용자 또는 루트 역할을 담당하는 관리자입니다. 전역 영역에 로그인한 경우 전역 관리 권한이 부여된 사용자는 시스템 전체를 모니터링하고 제어할 수 있습니다.

	zone administrator(영역 관리자) 를 참조하십시오.
global scope(전역 범위)	시스템의 모든 리소스 제어에 대한 리소스 제어 값을 적용하는 동작입니다.
global zone(전역 영역)	모든 Oracle Solaris 시스템에 포함된 영역입니다. 비전역 영역이 사용 중인 경우 전역 영역은 시스템의 기본 영역이자 시스템 전반 관리 제어에 사용되는 영역입니다. non-global zone(비전역 영역) 을 참조하십시오.
heap(힙)	프로세스에서 할당하는 스택래치 메모리입니다.
local scope(로컬 범위)	프로세스에 대해 수행되는 로컬 동작으로, 제어값을 초과하려고 시도합니다.
non-global zone administrator(비전역 영역 관리자)	zone administrator(영역 관리자) 를 참조하십시오.
non-global zone(비전역 영역)	Oracle Solaris 운영 체제의 단일 인스턴스 내에서 생성된 가상 운영 체제 환경입니다. Oracle Solaris 영역 소프트웨어 분할 기술은 운영 체제 서비스를 가상화하는 데 사용됩니다.
Oracle Solaris 10 Zone(Oracle Solaris 10 영역)	Oracle Solaris 11 릴리스를 실행하는 시스템의 solaris10 브랜드 영역에서 실행되는 Solaris 10 응용 프로그램에 완벽한 런타임 환경을 제공하는 소프트웨어 분할 기술입니다.
Oracle Solaris Kernel Zone(Oracle Solaris 커널 영역)	영역 내에서 완전한 커널 및 사용자 환경을 제공하며 호스트와 영역 간 커널 분리도 향상시켜 주는 소프트웨어 분할 기술입니다.
Oracle Solaris Zone(Oracle Solaris 영역)	운영 체제 시스템 서비스를 가상화하는 데 사용되는 소프트웨어 분할 기술로, 응용 프로그램을 실행하기 위한 격리된 보안 환경을 제공합니다.
pool daemon(풀 데몬)	동적 리소스 할당이 필요할 때 활성화되는 poold 시스템 데몬입니다.

pool(풀)	리소스 풀을 참조하십시오.
processor set(프로세서 세트)	CPU 분리 그룹입니다. 각 프로세서 세트에는 0개 이상의 프로세서가 포함될 수 있습니다. 프로세서 세트 하나는 리소스 풀 구성에서 리소스 요소로 표현됩니다. pset이라고도 합니다. disjoint(분리) 도 참조하십시오.
project(프로젝트)	관련 작업에 대한 네트워크 차원의 관리 식별자입니다.
rcapd(resource capping daemon)	리소스 한도가 정의된 프로젝트에서 실행하는 프로세스의 물리적 메모리의 사용량을 조정하는 데몬입니다.
read-only zone(읽기 전용 영역)	읽기 전용 루트로 구성된 변경할 수 없는 영역입니다.
resident set size(상주 메모리 페이지 세트 크기)	상주하는 메모리 페이지 세트의 크기입니다. 상주 메모리 페이지 세트란 물리적 메모리에 상주하는 페이지 세트를 말합니다.
resource set(리소스 세트)	프로세스 바인드가 가능한 리소스입니다. 대체로 특정 분할 형태를 제공하는 커널 부속 시스템에서 생성된 객체를 나타내는 데 사용됩니다. 리소스 세트의 예로는 예약 클래스 및 프로세서 세트 등이 있습니다.
resource(리소스)	응용 프로그램 동작을 변경할 목적으로 조작할 수 있는 컴퓨팅 시스템 요소입니다.
RSS	resident set size(상주 메모리 페이지 세트 크기) 를 참조하십시오.
scanner(스캐너)	자주 사용되지 않은 페이지를 식별하는 커널 스레드입니다. 사용 가능한 메모리의 양이 적은 상태에서는 스캐너가 최근에 사용되지 않은 페이지를 재생 이용합니다.
static pools configuration(정적 풀 구성)	관리자가 리소스 풀 기능과 관련하여 시스템을 구성하고자 하는 방법을 나타냅니다.
workload(작업 부하)	응용 프로그램 또는 응용 프로그램 그룹의 모든 프로세스를 총칭하는 용어입니다.
WSS	작업 세트 크기 를 참조하십시오.
zone administrator(영역 관리자)	영역 관리자의 권한은 비전역 영역으로 제한됩니다. global administrator(전역 관리자) 를 참조하십시오.
zone state(영역 상태)	비전역 영역의 상태입니다. 영역 상태는 구성됨, 완전하지 않음, 설치됨, 준비, 사용할 수 없음, 실행 중 또는 종료 중 상태 중 하나입니다.

색인

번호와 기호

- allowed-addresses
 - 배타적 IP 영역, 41
- autoboot, 31
- bootargs 등록 정보, 54
- BrandZ, 14
- capped-cpu 리소스, 34, 55
- capped-memory, 55
- capped-memory 리소스, 35
- dedicated-cpu 리소스, 32, 55
- defrouter, 65
 - 배타적 IP 영역, 41
- DHCP
 - 배타적 IP 영역, 41
- dtrace_proc, 54
- dtrace_user, 54
- EVS
 - 영역 사용, 39
- FSS(Fair Share Scheduler), 34
- hostid, 44
- IP 경로 지정
 - 배타적 IP 영역, 41
- IP 필터
 - 배타적 IP 영역, 41
- ip-type 등록 정보, 55
- ipkg 영역
 - solaris에 매핑, 25
 - 변환, 11
- IPMP
 - 배타적 IP 영역, 41
- IPoB, 64
- limitpriv 등록 정보, 54
- linkmode, 62
- lofi 장치
 - 이동식, 45
- net 리소스
 - 공유 IP 영역, 40
 - 배타적 IP 영역, 41
- Oracle Solaris 영역, 12
- Oracle Solaris 커널 영역, 12
- Oracle Solaris Cluster
 - 영역 클러스터, 14
- pkey, 62, 64
- pkg 업데이트 시 autoboot 사용 안함, 31
- pool 등록 정보, 57
- RDS(Reliable Datagram Socket), 42
- rootzpool 리소스
 - solaris 브랜드, 35
- scheduling-class 등록 정보, 55
- SMF 서비스
 - 비전역 영역, 23
 - 전역 영역, 23
- solaris, 12
- solaris 비전역 영역
 - Oracle Solaris, 25
- virtual-cpu 리소스, 33, 55
- ZFS
 - 데이터 세트, 56
- zone
 - Oracle Solaris 제한 사항 및 기능, 25
 - zone.cpu-cap 리소스 제어, 47
 - zone.cpu-shares 리소스 제어, 47
 - zone.max-locked-memory 리소스 제어, 47
 - zone.max-lofi 리소스 제어, 47
 - zone.max-lwps 리소스 제어, 47
 - zone.max-msg-ids 리소스 제어, 48
 - zone.max-processes 리소스 제어, 48
 - zone.max-sem-ids 리소스 제어, 48
 - zone.max-shm-ids 리소스 제어, 48
 - zone.max-shm-memory 리소스 제어, 48
 - zone.max-swap 리소스 제어, 48
- zonecfg

admin 권한 부여, 32
 template, 30
 모드, 50
 범위, 50
 범위, 리소스 특정, 50
 범위, 전역, 50
 엔티티, 53
 임시 풀, 32
 작업, 30
 전역 영역, 49
 하위 명령, 51
 zpool 리소스, 37

ㄱ
 공유 IP 영역, 40
 구성 가능한 권한, 영역, 46
 기능
 배타적 IP 영역, 41
 기본값 아님
 영역, 14

ㄷ
 데이터 링크, 39
 디스크 형식 지원
 영역, 45

ㄹ
 라이브 영역 재구성, 28
 리소스 제어
 영역 전체, 47

ㅁ
 물리적 메모리 상한값, 35

ㅂ
 배타적 IP 영역, 41
 변경 불가능한 영역
 읽기 전용 영역, 11

브랜드, 12, 12
 브랜드 영역, 14
 실행 중인 프로세스, 15
 비전역 영역, 17
 비전역 영역 관리자, 17

ㅅ
 스왑 공간 캡, 35

ㅇ
 영역
 anet, 55, 61
 bootargs 등록 정보, 54
 capped-cpu, 55
 capped-memory, 35, 55
 dedicated-cpu, 55
 ip-type, 55
 IPoIB(solaris만 해당), 61
 limitpriv, 54
 net, 56
 pool, 57
 rootzpool, 58
 scheduling-class, 55
 virtual-cpu, 55
 공유 IP, 40
 구성, 49
 구성 가능한 권한, 46
 구성 개요, 30
 권한, 역할, 프로파일, 29
 기능, 24
 기본값 아님, 14
 데이터 세트, 56
 등록 정보 유형, 53
 디스크 형식 지원, 45
 라이브 재구성, 28
 리소스 유형, 53
 리소스 유형 등록 정보, 58
 리소스 제어, 47
 만들기, 19
 모니터링, 24
 배타적 IP, 41
 브랜드, 14
 상태, 19

- 상태 모델, 19
- 영역 전체 리소스 제어, 53
- 유형별 특성, 18
- 정의, 9
- 영역 관리자, 19
- 영역 이름, 17
- 영역 전체 리소스 제어, 47
- 영역 admin 권한 부여, 32
- 영역 ID, 17
- 응용 프로그램 및 capped-cpu, 34
- 이동식 lofi 장치, 45
- 읽기 전용 영역
 - file-mac-profile, 31
- 읽기 전용 영역 루트, 31
- 임시 풀, 32

ㅈ

- 잠긴 메모리 상한값, 35
- 장치 리소스
 - 저장소 URI 사용, 46
- 전역 관리자, 17, 19
- 전역 영역, 17

