

# Oracle® Solaris 11.2 네트워크 구성 요소의 구성 및 관리

ORACLE®

부품 번호: E53787  
2014년 7월

Copyright © 2011, 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 서비스에 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
<b>1 Oracle Solaris의 네트워크 관리 정보 .....</b>	<b>9</b>
Oracle Solaris 네트워크 프로토콜 스택 설명 .....	9
하드웨어 계층 .....	11
데이터 링크 계층 .....	11
네트워크 계층 .....	11
전송 계층 .....	12
응용 프로그램 계층 .....	12
Oracle Solaris 네트워크 프로토콜 스택 내 이름 지정 및 디렉토리 서비스 구성 .....	12
Oracle Solaris의 네트워크 장치 및 데이터 링크 이름 지정 .....	13
네트워크 구성 모드 정보 .....	13
고정적 모드 .....	13
반응적 모드 .....	14
프로파일 기반 네트워크 구성 정보 .....	14
Oracle Solaris 네트워크 관리 명령 .....	14
dladm 명령 .....	15
ipadm 명령 .....	15
route 명령 .....	16
netcfg 및 netadm 명령 .....	16
시스템 네트워크 재구성 명령 .....	16
네트워크의 클라이언트 시스템 구성에 필요한 정보 .....	17
Oracle Solaris 네트워크 관리에 대한 추가 정보 위치 .....	18
<b>2 Oracle Solaris의 데이터 링크 구성 관리 .....</b>	<b>19</b>
데이터 링크 구성 정보 .....	19
데이터 링크에 일반 이름 지정 .....	20
운영 체제의 일반 링크 이름 지정 방법 사용자 정의 .....	21
업그레이드된 시스템의 링크 이름 .....	22
데이터 링크 등록 정보 관리 .....	24

데이터 링크에 대한 일반 정보 표시 .....	24
시스템의 데이터 링크 표시 .....	25
데이터 링크의 물리적 속성 표시 .....	26
데이터 링크 삭제 .....	27
데이터 링크 이름 바꾸기 .....	27
데이터 링크에 대한 런타임 통계 가져오기 .....	27
데이터 링크 등록 정보 사용자 정의 .....	28
점보 프레임 지원 사용 .....	29
링크 속도 매개변수 수정 .....	29
데이터 링크에 STREAMS 모듈 설정 .....	30
데이터 링크 등록 정보에 대한 상태 정보 가져오기 .....	31
추가 dladm 구성 작업 .....	32
▼ 네트워크 장치 간에 IP 구성을 이동하는 방법 .....	33
▼ 동적 재구성을 사용하여 네트워크 인터페이스 카드를 교체하는 방법 .....	34
▼ SPARC: 각 인터페이스의 MAC 주소가 고유한지 확인하는 방법 .....	36
<b>3 Oracle Solaris에서 IP 인터페이스와 주소 구성 및 관리 .....</b>	<b>39</b>
ipadm 명령을 사용하여 네트워크 구성 관리 .....	39
IPv4 인터페이스 구성 .....	40
▼ IPv4 인터페이스를 구성하는 방법 .....	40
IPv6 인터페이스 구성 .....	45
▼ IPv6에 대해 시스템을 구성하는 방법 .....	45
IPv6 인터페이스에 대해 임시 주소 사용 .....	47
IPv6 토큰 구성 .....	49
서버에서 IPv6 사용 인터페이스 구성 .....	51
IPv4 네트워크에서 IPv6 네트워크로 마이그레이션 .....	52
경로 지정 구성 .....	53
경로 지정 테이블 및 경로 지정 유형 .....	53
지속(정적) 경로 만들기 .....	54
단일 인터페이스 시스템에 대한 경로 지정 사용 .....	57
IPv6 경로 지정 정보 .....	60
멀티홈 호스트 구성 .....	61
▼ 멀티홈 호스트를 만드는 방법 .....	62
멀티홈 호스트에 대칭 경로 지정 구현 .....	63
IP 인터페이스 등록 정보 및 주소 사용자 정의 .....	64
MTU 등록 정보 설정 .....	65
패킷 전달 사용 .....	65
IP 주소 등록 정보 사용자 정의 .....	66
IP 인터페이스 구성 사용 안함, 제거 및 수정 .....	67

IP 인터페이스 구성 제거 .....	67
IP 인터페이스 구성 사용 안함 .....	68
IP 인터페이스 구성 제거 또는 수정 .....	68
IP 인터페이스 및 주소 모니터링 .....	69
IP 인터페이스에 대한 일반 정보 가져오기 .....	70
IP 인터페이스에 대한 정보 가져오기 .....	70
IP 인터페이스 등록 정보에 대한 정보 가져오기 .....	71
IP 주소에 대한 정보 가져오기 .....	72
IP 주소 등록 정보에 대한 정보 가져오기 .....	73
<b>4 Oracle Solaris 클라이언트에서 이름 지정 및 디렉토리 서비스 관리 .....</b>	<b>75</b>
이름 지정 서비스 구성의 새로운 기능 .....	75
이름 지정 및 디렉토리 서비스 구성 개요 .....	76
name-service/switch SMF 서비스 정보 .....	77
로컬 파일 모드에 대한 시스템 구성 .....	78
▼ 로컬 파일 모드에 대한 시스템 구성 방법 .....	79
DNS 클라이언트 구성 .....	80
▼ DNS 클라이언트를 사용으로 설정하는 방법 .....	80
멀티캐스트 DNS 사용 .....	81
DNS에 대한 리소스 알림 .....	82
NIS 클라이언트 구성 .....	82
▼ 브로드캐스트 모드로 NIS 클라이언트를 구성하는 방법 .....	83
▼ 특정 NIS 서버를 사용하여 NIS 클라이언트를 구성하는 방법 .....	83
▼ NIS 클라이언트 서비스를 사용 안함으로 설정하는 방법 .....	84
LDAP 클라이언트 구성 .....	84
이름 지정 서비스 구성 가져오기 .....	85
SMF 이름 지정 서비스 구성 재설정 .....	85
<b>5 Oracle Solaris의 프로파일 기반 네트워크 구성 관리 정보 .....</b>	<b>87</b>
반응적 모드 정보 .....	87
프로파일 기반 네트워크 구성 정보 .....	88
프로파일 유형 설명 .....	89
시스템 정의 및 사용자 정의 프로파일 .....	92
프로파일 기반 네트워크 구성 사용 지침 .....	93
프로파일 활성화 정책 .....	93
프로파일 활성화 모드 .....	94
프로파일 기반 네트워크 구성 사용을 위한 보안 요구 사항 .....	95
프로파일 기반 네트워크 구성이 다른 Oracle Solaris 기능에서 작동하는 방식 .....	96

<b>6 Oracle Solaris의 프로파일 기반 네트워크 구성 관리</b> .....	97
프로파일 사용 및 사용 안함으로 설정 .....	97
프로파일 구성 .....	99
netcfg 대화식 모드로 작업 .....	99
netcfg 명령줄 모드로 작업 .....	101
netcfg 명령 파일 모드로 작업 .....	101
NCP 만들기 .....	102
NCP의 NCU 만들기 .....	103
위치 만들기 .....	106
ENM 만들기 .....	108
알려진 WLAN 만들기 .....	109
프로파일 관리 .....	111
프로파일에 대한 등록 정보 값 설정 .....	111
프로파일 구성에 대한 정보 가져오기 .....	113
walkprop 하위 명령을 사용하여 프로파일에 대한 등록 정보 값 설정 .....	116
프로파일에 대한 정보 표시 .....	118
프로파일 제거 .....	120
프로파일 구성 내보내기 .....	121
내보낸 프로파일 구성 복원 .....	123
데스크탑에서 네트워크 구성 관리 .....	123
<b>7 Oracle Solaris의 무선 네트워크 관리</b> .....	125
명령줄을 사용하여 무선 네트워크 관리 .....	125
▼ WiFi 네트워크에 연결하는 방법 .....	125
▼ WiFi 링크를 모니터하는 방법 .....	129
보안 WiFi 통신 설정 .....	130
▼ WEP 키를 지정하여 암호화된 WiFi 네트워크 연결을 설정하는 방법 .....	131
반응적 모드로 알려진 WLAN 관리 .....	132
데스크탑에서 무선 네트워크 관리 .....	132
▼ 무선 네트워크를 연결하는 방법 .....	133
데스크탑에서 즐겨 찾는 무선 네트워크 관리 .....	134
<b>색인</b> .....	137

## 이 설명서 사용

---

- **개요** - Oracle Solaris OS(운영 체제)에서 데이터 링크, IP 인터페이스 및 주소, 이름 지정 및 디렉토리 서비스, 반응적 프로파일, 무선 네트워크 등 다양한 네트워크 구성 요소를 구성하고 관리하는 방법에 대한 정보를 제공합니다.
- **대상** - 회사 데이터 센터에서 네트워크 구성을 관리하는 시스템 관리자를 대상으로 합니다.
- **필요한 지식** - 기본 및 전문적인 네트워크 관리 개념과 방법을 알고 있어야 합니다.

## 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 Solaris의 네트워크 관리 정보

---

이 장에서는 Oracle Solaris의 호스트 클라이언트 시스템에 대한 네트워크 구성을 이루는 다양한 구성 요소의 개요를 제공합니다. 이 설명서에 제공된 작업 및 예에서는 사용자가 설치 후 네트워크 구성을 수행하고 있다고 가정합니다. 설치 중 네트워크 구성에 대한 지침은 [“Oracle Solaris 11.2 시스템 설치”](#)를 참조하십시오.

새 네트워킹 기능에 대한 자세한 내용은 [“Oracle Solaris 11.2의 새로운 기능”](#)을 참조하십시오.

네트워크의 클라이언트 시스템을 구성하기 전에 필요한 계획 작업은 [“Oracle Solaris 11.2의 네트워크 배치 계획”](#)을 참조하십시오.

일반적으로 사용되는 네트워크 관리 명령에 대한 빠른 참조는 [“Oracle Solaris 11.2의 네트워크 관리 전략”](#)의 3 장, [“네트워크 관리 명령 참고 자료”](#)를 참조하십시오.

기존 TCP/IP 네트워크 관리에 대한 자세한 내용은 [“Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”](#)의 1 장, [“TCP/IP 네트워크 관리”](#)를 참조하십시오.

이 장의 내용:

- [“Oracle Solaris 네트워크 프로토콜 스택 설명”](#) [9]
- [“Oracle Solaris의 네트워크 장치 및 데이터 링크 이름 지정”](#) [13]
- [“네트워크 구성 모드 정보”](#) [13]
- [“프로파일 기반 네트워크 구성 정보”](#) [14]
- [“Oracle Solaris 네트워크 관리 명령”](#) [14]
- [“네트워크의 클라이언트 시스템 구성에 필요한 정보”](#) [17]
- [“Oracle Solaris 네트워크 관리에 대한 추가 정보 위치”](#) [18]

## Oracle Solaris 네트워크 프로토콜 스택 설명

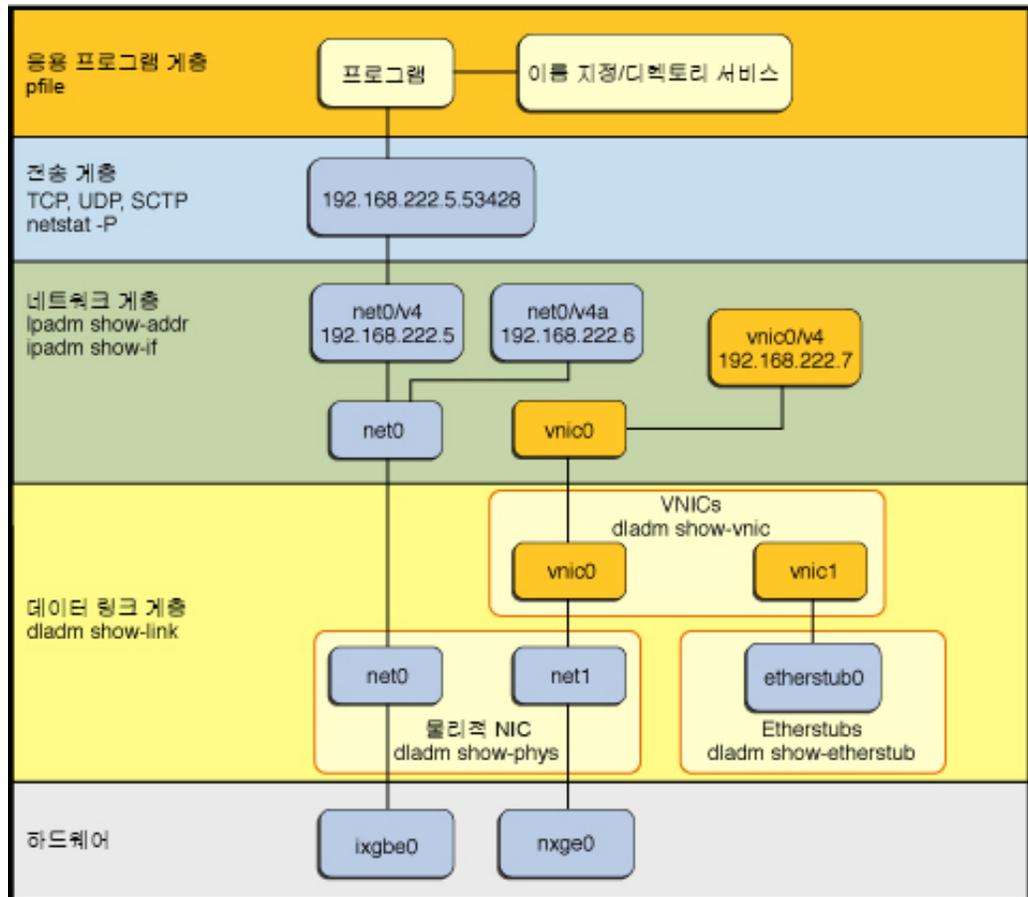
네트워크 인터페이스는 시스템과 네트워크 간의 연결을 제공합니다. 이러한 인터페이스는 데이터 링크를 통해 구성되며, 데이터 링크는 시스템의 하드웨어 장치 인스턴스에 따라 구성됩니다.

Oracle Solaris 10에는 장치와 데이터 링크, 인터페이스를 바인드하는 일대일 관계가 존재합니다. 즉, 네트워크 구성이 하드웨어 구성에 종속되며 네트워크 토폴로지에도 종속됩니다.

NIC(네트워크 인터페이스 카드) 교체와 같이 하드웨어 계층에서 변경이 구현되는 경우 시스템에서 인터페이스를 재구성해야 합니다.

하지만 이 Oracle Solaris 릴리스에서는 물리적 데이터 링크의 이름 지정이 더 이상 네트워크 장치에 연관된 기본 하드웨어와 관련되지 않습니다. 그림 1-1과 같이 해당 장치에는 기본적으로 net이라는 일반 이름과 시스템에서 장치의 물리적 위치를 반영하는 접미어가 지정됩니다. 이러한 구분에 따라 네트워크 계층의 네트워크 구성은 더 이상 하드웨어 계층의 네트워크 토폴로지 또는 칩셋에 구애 받지 않습니다.

그림 1-1 Oracle Solaris 11 네트워크 프로토콜 스택



이러한 구현으로 네트워크 관리가 다음과 같이 보다 유연해질 수 있습니다.

- 네트워크 구성은 하드웨어 계층에서 발생 가능한 변경 사항으로부터 분리됩니다. 링크와 인터페이스 구성은 기본 하드웨어를 제거하더라도 유지됩니다. 두 NIC가 서로 같은 유형이면 이러한 동일한 구성을 교체 NIC에 다시 적용할 수 있습니다.
- 네트워크 하드웨어 구성에서 네트워크 구성을 분리하면 데이터 링크 계층에 사용자 정의 링크 이름을 사용할 수 있습니다.
- 데이터 링크 계층의 추상화를 사용할 경우 VLAN(가상 근거리 통신망), VNIC(가상 네트워크 인터페이스 카드), 물리적 장치, 링크 통합 및 IP 터널과 같은 여러 네트워킹 추상화 또는 구성이 공통 관리 엔티티인 데이터 링크에 통합됩니다.

Oracle Solaris 10 네트워크 스택과 Oracle Solaris 11 네트워크 스택 간의 비교는 [“Oracle Solaris 10에서 Oracle Solaris 11.2로 전환”](#)의 [“Oracle Solaris 10 네트워크 프로토콜 스택과 Oracle Solaris 11 네트워크 프로토콜 스택 비교”](#)를 참조하십시오.

## 하드웨어 계층

네트워크 하드웨어 장치를 *NIC*(네트워크 인터페이스 카드) 또는 네트워크 어댑터라고도 부릅니다. NIC는 시스템 구입 시 이미 내장되어 제공될 수 있습니다. 별도로 NIC를 구입하여 시스템에 추가할 수도 있습니다. 특정 NIC는 카드에 단일 인터페이스만 포함됩니다. 다른 NIC 브랜드 제품의 경우 다양한 네트워크 작업을 수행하도록 구성할 수 있는 여러 개의 인터페이스가 포함되었을 수 있습니다.

## 데이터 링크 계층

데이터 링크 계층에서 네트워크 구성을 수행하려면 `dladm` 명령을 사용하십시오. 이 계층에서 수행할 수 있는 데이터 링크 구성의 몇 가지 유형은 다음과 같습니다.

- 물리적 링크의 기본적인 네트워크 구성
- VNIC 구성(물리적 링크를 통한 가상 링크)  
VNIC 구성의 경우 각 가상 링크에 고유의 MAC 주소가 있습니다.
- 링크 통합 구성  
안정성 및 성능을 위해 물리적 링크를 통해 통합이 구성됩니다.
- VLAN(가상 근거리 통신망) 지원을 위한 Etherstub
- VLAN 지원을 위한 브리지

다른 예는 [“Oracle Solaris 11.2의 네트워크 데이터 링크 관리”](#)를 참조하십시오.

## 네트워크 계층

네트워크 계층에서 네트워크 구성을 수행하려면 `ipadm` 명령을 사용하십시오. 이 계층에서 수행할 수 있는 IP 구성의 몇 가지 유형은 다음과 같습니다.

- 이름이 데이터 링크 이름과 일대일로 대응하는 IP 인터페이스 구성
- IPMP(IP Network Multipathing)
- VNI(가상 네트워크 인터페이스)
- 하나의 IP 인터페이스에 대한 다중 IP 주소
- 단일 IP 인터페이스에 대해 구성된 IPv4 및 IPv6 주소
- 주소 객체 이름을 사용하여 관리되는 IP 주소  
주소 객체 이름은 인터페이스 이름과 고유 문자열로 구성되며 시스템에서 구성된 IP 주소를 나타냅니다.

다른 예는 다음을 참조하십시오.

- [“Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”](#)
- [“Oracle Solaris 11.2의 네트워크 가상화 및 네트워크 리소스 관리”](#)
- [“Oracle Solaris 영역 만들기 및 사용”](#)

## 전송 계층

전송 계층의 명시적 구성은 필요한 경우가 거의 없습니다. Oracle Solaris에서 사용되는 응용 프로그램은 대개 적합한 전송 프로토콜과 해당하는 포트 번호를 자동으로 선택합니다. `netstat` 명령을 사용하여 활성 포트를 확인할 수 있습니다. [netstat\(1M\)](#)를 참조하십시오.

`ipadm` 명령을 사용하여 일부 전송 프로토콜 매개변수를 조정할 수 있습니다. 자세한 내용은 [“Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”](#)의 [“전송 계층 서비스 관리”](#) 및 [“Oracle Solaris 11.2 조정 가능 매개변수 참조 설명서”](#)를 참조하십시오.

## 응용 프로그램 계층

응용 프로그램은 소켓 `xti` 또는 `tli` API(응용 프로그래밍 인터페이스)를 통해 네트워크에 액세스합니다. 이러한 API를 사용하려면 연결을 시작할 때 클라이언트 응용 프로그램이 해당하는 서버의 IP 주소 및 전송 포트 번호를 제공해야 합니다. 일반적으로 서버는 IP 주소가 아닌 원격 호스트 또는 서비스 이름을 통해서만 알려줍니다. 응용 프로그램은 표준 라이브러리 서비스를 사용하여 네트워크 연결을 시도하기 전에 호스트 및 서비스 이름을 IP 주소로 변환합니다.

자세한 내용은 [“Oracle Solaris 11.2 Programming Interfaces Guide”](#)를 참조하십시오.

## Oracle Solaris 네트워크 프로토콜 스택 내 이름 지정 및 디렉토리 서비스 구성

클라이언트 시스템에는 IP 주소 조회를 수행하도록 구성된 네트워크의 이름 및 디렉토리 서비스가 있어야 합니다. 네트워크 환경에 따라 구성 프로세스는 시스템 부트 중 자동으로 발생

할 수도 있고 수동으로 수행되어야 할 수도 있습니다. 이 구성은 네트워크 스택의 응용 프로그램 계층에서 발생합니다.

클라이언트측 이름 지정 및 디렉토리 서비스 구성 옵션은 [4장. Oracle Solaris 클라이언트에서 이름 지정 및 디렉토리 서비스 관리](#)를 참조하십시오.

## Oracle Solaris의 네트워크 장치 및 데이터 링크 이름 지정

데이터 링크는 OSI(Open Systems Interconnection) 모델에서 두번째 계층(L2)의 링크 객체를 나타냅니다. 물리적 링크는 장치와 직접적으로 연결되며 장치 이름을 소유합니다. 장치 이름에는 드라이버 이름과 장치 인스턴스 번호가 포함됩니다. 인스턴스 번호는 시스템에서 해당 드라이버를 사용하는 NIC 수에 따라 0에서  $n$ 까지의 값을 가질 수 있습니다.

장치 인스턴스 이름은 시스템의 기본 하드웨어에 계속 종속됩니다. 하지만 하드웨어 계층과 소프트웨어 계층은 구분되므로 해당 장치를 기반으로 구성된 데이터 링크가 더 이상 유사한 방식으로 바인드되지 않습니다. 따라서 데이터 링크에는 해당 데이터 링크가 구성된 장치 이름이 아닌 다른 이름이 지정될 수 있습니다.

기본적으로 데이터 링크에는 net# 이름 지정 규약을 사용하는 일반 이름이 지정됩니다. 여기서 #은 장치 인스턴스 번호입니다. 인스턴스 번호는 시스템의 장치별로 증분됩니다(예: net0, net1, net2 등). 보다 자세한 개요는 [“데이터 링크 구성 정보” \[19\]](#)를 참조하십시오.

## 네트워크 구성 모드 정보

Oracle Solaris에서는 두 가지 네트워크 구성 모드(고정적 및 반응적)가 지원됩니다.

---

**참고** - 고정적 모드와 반응적 모드라는 용어는 시스템이 현재 네트워크 환경에서 변경 사항에 맞게 자동으로 조정을 수행할 수 있는지 여부 및 해당 모드로 사용될 때 정적 또는 고정 IP 주소를 구성할 수 있는지 여부를 나타냅니다.

---

### 고정적 모드

고정적 모드는 네트워크 상태 변경 여부에 관계없이 시스템에서 인스턴스화된 구성이 지속됨을 나타냅니다. 인터페이스 추가 등의 변경이 발생할 경우 시스템이 새 환경에 대응하도록 네트워크를 재구성해야 합니다. 고정적 모드를 사용하는 경우 항상 동일한 네트워크 구성 명령 세트를 사용하여 시스템이 구성됩니다. 회사 서버는 네트워크 환경이 비교적 안정적이므로 이 구성 모드를 사용하는 경우가 많습니다. 고정적 모드를 사용하는 경우 네트워크 구성의 다양한 요소를 관리하는 데 `dladm` 및 `ipadm` 명령을 사용합니다. [“Oracle Solaris 네트워크 관리 명령” \[14\]](#)을 참조하십시오.

## 반응적 모드

반면 반응적 모드는 현재 네트워크 상태에 따라 네트워크가 자동으로 구성되는 경우를 나타냅니다. 이 모드는 주로 랩탑 컴퓨터 및 노트북 PC(개인용 컴퓨터)에서 사용되며 네트워크 상태가 변경될 수 있는 경우 사용됩니다.

반응적 모드에서는 네트워크 데몬(nwamd)이 시스템의 네트워크 인터페이스 상태를 모니터링합니다. 네트워크 상태가 변경될 때마다 네트워크 데몬은 동적으로 네트워크 구성을 조정합니다. 예를 들어, 노트북 PC가 물리적으로 회사 네트워크에 연결되어 있을 수도 있고 연결되어 있지 않을 수도 있습니다. 물리적으로 연결된 경우 노트북의 무선 인터페이스를 사용 안함으로 설정할 가능성이 높습니다. 이더넷 케이블이 노트북에서 분리되는 경우 무선 인터페이스가 자동으로 사용으로 설정되도록 하고자 할 수도 있습니다. 또한 무선 네트워크로 전환할 때 시스템이 자동으로 IP 필터 설정을 조정하도록 할 수 있습니다. 반응적 모드에서는 네트워크 데몬이 자동으로 이러한 유형의 동적 구성 변경을 수행할 수 있습니다. 반대로 고정적 모드에서는 수동 재구성 단계를 통해서만 이러한 유형의 변경을 수행할 수 있습니다.

## 프로파일 기반 네트워크 구성 정보

프로파일 기반 네트워크 구성을 통해 각각 단일 프로파일로 식별되는 여러 가지 대체 구성을 정의할 수 있습니다. 이 단일 프로파일을 NCP(네트워크 구성 프로파일)라고 합니다. 예를 들어, 노트북 PC에 대해 정적 IP 주소와 DNS 서버 위치로 시스템을 구성하는 office라는 프로파일을 만들 수 있습니다. 대체 home 프로파일은 DHCP를 사용하여 이 정보를 획득할 수 있습니다. 단일 명령을 사용하면 몇 초 이내에 프로파일 간에 전환할 수 있습니다. 사용으로 설정할 수 있는 다양한 유형의 프로파일이 가능한 두 가지 네트워크 구성 모드(고정적 및 반응적)를 지원합니다. 기본 모드는 현재 시스템에서 활성 상태인 프로파일에 따라 결정됩니다.

Oracle Solaris 설치 중 시스템에서 프로파일을 활성화하는 방법에 대한 자세한 내용은 [“Oracle Solaris 10에서 Oracle Solaris 11.2로 전환”의 “설치 중 네트워크가 구성되는 방식”](#)을 참조하십시오.

현재 시스템에서 활성 상태인 프로파일을 모를 경우 `netadm list` 명령을 사용하여 이 정보를 표시하십시오. 자세한 내용은 [“프로파일 사용 및 사용 안함으로 설정” \[97\]](#)을 참조하십시오.

Oracle Solaris에서 지원되는 다양한 유형의 프로파일에 대한 전체 설명은 [5장. Oracle Solaris의 프로파일 기반 네트워크 구성 관리 정보](#)를 참조하십시오.

## Oracle Solaris 네트워크 관리 명령

네트워크 구성 관리에 사용되는 명령은 다음과 같습니다.

- dladm
- ipadm
- route
- netcfg
- netadm

## dladm 명령

Oracle Solaris 10에서 도입된 dladm은 데이터 링크 구성에 사용됩니다.

dladm 명령은 다음 유형의 네트워크 구성을 관리하는 데 사용됩니다.

- 물리적 인터페이스 - 이더넷, 무선 및 InfiniBand
- 가상 네트워킹 기능 - Etherstub, VNIC 및 IP 터널
- 전환 기능 - 링크 통합, VLAN 및 브리징 기술
- 장치 특성 - 속도, 이중화, 우선 순위 및 기능 협상

dladm 명령은 현재 시스템에서 활성 상태인 프로파일에 대해 지속 네트워크 구성을 만듭니다. 따라서 net0은 다른 프로파일에서 서로 다른 MTU 값을 가질 수 있습니다. 예를 들어, net0이라는 데이터 링크가 특정 MTU(최대 전송 단위) 1200으로 구성되면 이 MTU 값은 해당 프로파일의 net0에 대해서만 지속됩니다. 그런 후 dladm 명령을 사용하여 다른 프로파일을 활성화하고 해당 프로파일에 대해 다른 MTU 값을 설정하면 해당 프로파일에만 새 MTU 값이 적용됩니다. [2장. Oracle Solaris의 데이터 링크 구성 관리](#)를 참조하십시오.

또한 dladm 명령은 Oracle Solaris 10에서 프로토콜 등록 정보를 구성하는 데 사용되는 ndd 명령을 대체합니다. 계층 2 드라이버 등록 정보를 설정하는 도구로서 dladm 명령은 ndd 명령에 비해 다양한 이점을 제공합니다. [“Oracle Solaris 10에서 Oracle Solaris 11.2로 전환”의 “ndd 명령과 ipadm 명령 비교”](#)를 참조하십시오.

## ipadm 명령

ipadm 명령은 이 릴리스에서 IP 인터페이스 및 주소를 구성하는 ifconfig 명령을 대체합니다. ipadm 명령은 IP 인터페이스 관리에만 사용되므로 보다 효율적으로 IP 인터페이스 및 IP 주소를 관리합니다. 또한 ifconfig 명령과 달리 ipadm 명령은 지속 네트워크 구성을 구현합니다. [“Oracle Solaris 10에서 Oracle Solaris 11.2로 전환”의 “ifconfig 명령과 ipadm 명령 비교”](#)를 참조하십시오.

또한 ipadm 명령은 Oracle Solaris 10에서 프로토콜 등록 정보를 구성하는 데 사용되었던 ndd 명령을 대체합니다. 프로토콜 등록 정보를 설정하는 도구로서 ipadm 명령은 ndd 명령에 비해 여러 가지 이점을 제공합니다. [“Oracle Solaris 10에서 Oracle Solaris 11.2로 전환”의 “ndd 명령과 ipadm 명령 비교”](#)를 참조하십시오.

## route 명령

/etc/defaultrouter 파일은 Oracle Solaris 11에서 사용되지 않으므로 이 파일을 통해 더 이상 경로(기본값 또는 기타)를 관리할 수 없습니다. 대신 route 명령을 사용하여 수동으로 네트워크 경로 지정 테이블을 조작하십시오. route 명령은 활성 프로파일에 한해서만 경로를 조작합니다. 활성 프로파일이 변경되면 기본 경로와 기타 모든 경로가 대체될 수 있습니다. 하지만 시스템에서 프로파일을 전환하지 않는 경우 문제가 되지 않습니다.

자세한 내용은 [“지속\(정적\) 경로 만들기” \[54\]](#)를 참조하십시오.

## netcfg 및 netadm 명령

netcfg 및 netadm 명령은 다양한 유형의 프로파일을 관리하는 데 사용됩니다. 이러한 두 가지 명령으로 제공되는 대부분의 기능은 반응적 프로파일을 관리하는 데 사용됩니다. netcfg 명령은 회사 서버에서 거의 사용되지 않습니다. 해당 유형의 서버는 일반적으로 고정적 모드를 사용합니다.

netadm 명령은 프로파일을 사용 및 사용 안함으로 설정하고 프로파일과 해당 상태에 대한 정보를 표시하는 데 사용됩니다. [“프로파일 사용 및 사용 안함으로 설정” \[97\]](#) 및 [“프로파일 관리” \[111\]](#)를 참조하십시오.

---

**참고** - 일반적으로 netcfg 명령을 사용하여 반응적 프로파일의 등록 정보를 구성합니다. 하지만 프로파일이 현재 활성 상태인 경우 dladm 및 ipadm 명령을 사용하여 반응적 프로파일에 대한 지속 구성을 만들 수도 있습니다. 단, netcfg 명령을 통해서만 시스템의 유일한 고정적 프로파일인 DefaultFixed를 구성할 수 없습니다. 자세한 내용은 [netcfg\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

---

## 시스템 네트워크 재구성 명령

시스템 네트워크 재구성에 사용할 수 있는 다른 옵션은 SCI(시스템 구성 대화식) 도구라고도 하는 sysconfig 유틸리티입니다. SCI 도구는 새로 설치되거나 구성되지 않은 시스템의 구성을 지원하며, 텍스트 설치 중 새로 만들어진 비전역 영역에 대한 시스템 구성을 제공하도록 설계되었습니다. SCI 도구는 대화식 또는 비대화식으로 사용할 수 있습니다.

sysconfig 유틸리티를 사용하여 세 가지 작업(구성 해제, 구성 및 프로파일 만들기)을 수행할 수 있습니다. unconfigure 하위 명령은 전체 시스템의 구성을 해제하는 데 사용됩니다. 이 명령은 시스템을 구성 해제 상태로 유지합니다.

configure 하위 명령은 전체 시스템 또는 시스템 일부를 재구성하는 데 사용되며, 다음과 같은 여섯 가지 기능 그룹을 포함합니다.

- network

- location
- users
- identity
- support
- kdb\_layout

구성 해제된 그룹의 기본값에 대한 자세한 내용은 [“Oracle Solaris 11.2 시스템 설치”의 “기능 그룹 개요”](#)를 참조하십시오.

예를 들어, 다음과 같이 시스템의 기존 이름 지정 서비스를 재구성합니다.

```
# sysconfig configure -g network,naming_services
```

-g 옵션은 구성할 특정 기능 그룹을 지정하는 데 사용됩니다. 이 예에서는 시스템의 네트워크 구성 요소가 구성됩니다.

자세한 내용은 [sysconfig\(1M\)](#) 매뉴얼 페이지 및 [“Oracle Solaris 11.2 시스템 설치”의 6장, “Oracle Solaris 인스턴스 구성 해제 또는 재구성”](#)을 참조하십시오.

## 네트워크의 클라이언트 시스템 구성에 필요한 정보

Oracle Solaris를 설치하는 도중 또는 설치한 후 네트워크를 구성할 수 있습니다. 이 설명서에서는 설치 후 네트워크의 클라이언트 시스템을 구성하는 작업에 대해 설명합니다. 설치 중 네트워크 구성에 대한 지침은 [“Oracle Solaris 11.2 시스템 설치”](#)를 참조하십시오.

네트워크의 클라이언트 시스템을 구성하려면 다음 정보를 제공해야 합니다.

- 호스트 이름
- IP 주소
- 넷마스크

자세한 내용은 [“Oracle Solaris 11.2의 네트워크 배치 계획”의 “네트워크에 대한 IP 주소 지정 형식 결정”](#) 및 [“Oracle Solaris 11.2의 네트워크 배치 계획”의 “네트워크의 IP 번호 얻기”](#)를 참조하십시오.

네트워크가 서브넷으로 세분화된 경우 개별 넷마스크를 비롯하여 서브넷 번호와 각 서브넷의 시스템에 적용할 IP 주소 스키마가 있어야 합니다. [“Oracle Solaris 11.2의 네트워크 배치 계획”의 “네트워크에서 서브넷 사용”](#)을 참조하십시오.

- 시스템이 속한 도메인 이름

[“Oracle Solaris 11.2의 네트워크 배치 계획”의 “네트워크에서 이름 지정 엔티티 사용”](#)을 참조하십시오.

- 기본 라우터 주소

각 네트워크에 연결된 라우터가 하나뿐인 간단한 네트워크 토폴로지를 사용하는 경우 이 정보를 제공합니다. [“Oracle Solaris 11.2의 네트워크 배치 계획”의 “네트워크의 라우터 계획”](#)을 참조하십시오.

라우터가 RDISC(Router Discovery Server Protocol), RIP(Router Information Protocol) 등의 경로 지정 프로토콜을 실행하지 않는 경우에도 이 정보를 제공합니다. 라우터 및 Oracle Solaris에서 지원하는 경로 지정 프로토콜 목록에 대한 자세한 내용은 [“Oracle Solaris 11.2 시스템을 라우터 또는 로드 밸런서로 구성”](#)의 [“경로 지정 프로토콜”](#)을 참조하십시오.

네트워크의 클라이언트 시스템을 구성하는 경우 [“Oracle Solaris 11.2의 네트워크 배치 계획”](#)의 [“IPv4 자율 시스템 토폴로지”](#)의 정보를 참조하십시오.

각각의 해당 구성 요소 및 관련 작업에 대한 자세한 내용은 [“Oracle Solaris 11.2의 네트워크 배치 계획”](#)의 1 장, [“네트워크 배치 계획”](#)을 참조하십시오.

## Oracle Solaris 네트워크 관리에 대한 추가 정보 위치

이 설명서에서 제공되는 기본적인 구성 정보 외에 다음 참조를 통해 여러 유형의 네트워크 관리를 수행하는 것과 관련된 정보를 확인할 수 있습니다.

- 네트워크 프로토콜을 사용자 정의하고 전송 계층 서비스를 관리하려면 [“Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”](#)의 1 장, [“TCP/IP 네트워크 관리”](#)를 참조하십시오.
- 시스템을 라우터 또는 로드 밸런서로 구성하려면 [“Oracle Solaris 11.2 시스템을 라우터 또는 로드 밸런서로 구성”](#)을 참조하십시오.
- 링크 통합과 다양한 유형의 브리지를 비롯하여 고급 데이터 링크 및 IP 구성을 수행하려면 [“Oracle Solaris 11.2의 네트워크 데이터 링크 관리”](#)를 참조하십시오.
- IPMP 그룹 및 IP 터널을 구성하려면 [“Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”](#)를 참조하십시오.
- 네트워크에 대한 보안을 설정하려면 [“Oracle Solaris 11.2의 네트워크 보안”](#)을 참조하십시오.
- 네트워크 가상화 기능을 구현하려면 [“Oracle Solaris 11.2의 네트워크 가상화 및 네트워크 리소스 관리”](#)를 참조하십시오.

# ◆◆◆ 2 장

## Oracle Solaris의 데이터 링크 구성 관리

---

이 장에서는 네트워크 장치 및 데이터 링크에 대해 설명하며 고정적 모드를 사용할 때 기본 데이터 링크 구성을 관리하는 작업에 대해 다룹니다.

다음 정보는 네트워크 장치를 나타내는 링크 또는 물리적 링크 구성에만 적용됩니다. 구성할 수 있는 기타 계층 2(L2) 엔티티에 대한 자세한 내용은 “Oracle Solaris 11.2의 네트워크 데이터 링크 관리” 및 “Oracle Solaris 11.2의 네트워크 가상화 및 네트워크 리소스 관리”를 참조하십시오.

이 장의 내용:

- “데이터 링크 구성 정보” [19]
- “데이터 링크 등록 정보 관리” [24]
- “데이터 링크 등록 정보 사용자 정의” [28]
- “추가 dladm 구성 작업” [32]

### 데이터 링크 구성 정보

관리자가 데이터 링크를 기반으로 IP 인터페이스를 만듭니다. 각 데이터 링크는 OSI(Open Systems Interconnection) 모델에서 두번째 계층의 링크 객체를 나타냅니다. 데이터 링크는 물리적 네트워크 장치(물리적 링크라고 함), 물리적 데이터 링크 통합, VNIC(가상 네트워크 인터페이스 카드) 등 다양한 L2 엔티티를 나타낼 수 있습니다.

링크 이름은 연관된 링크 객체가 자동으로 만들어질 때 지정됩니다. 또는 사용자가 데이터 링크를 만들 때 명시적으로 링크 이름을 지정할 수도 있습니다. 물리적 링크(물리적 네트워크 장치와 연관됨)는 장치가 추가되거나 설치 후 Oracle Solaris 시스템이 처음 부트될 때 자동으로 만들어집니다. 이 Oracle Solaris 릴리스에서는 물리적 데이터 링크의 이름 지정이 더 이상 네트워크 장치에 연관된 기본 하드웨어와 관련되지 않습니다. 기본적으로 데이터 링크에는 net이 접두어로 사용되고 시스템에서 데이터 링크의 물리적 위치를 반영하는 번호가 접미어로 사용되는 이름이 지정됩니다. 예를 들어, 첫번째 내장 네트워크 장치 e1000g0에는 net0이라는 이름이 지정되며, 다음 e1000g1 장치에는 net1이라는 이름이 지정됩니다. 명시적으로 링크 통합 등을 만든 데이터 링크에 임의 이름을 지정할 수 있습니다. 원하는 경우 데이터 링크의 기본 지정된 netN 이름을 명시적으로 바꿀 수도 있습니다.

일반 또는 유연한 링크 이름은 네트워크 구성에서 다음 이점을 제공합니다.

- 단일 시스템 내에서 DR(동적 재구성)이 더 쉽습니다. 지정된 NIC의 네트워크 구성은 다른 NIC 교체 시에 상속될 수 있습니다.
- 영역 마이그레이션에 대한 네트워크 설정의 복잡성을 줄여줍니다. 대상 시스템의 링크와 마이그레이션 전 영역에 지정된 링크의 이름이 동일한 경우 마이그레이션된 시스템의 영역이 네트워크 구성을 보존합니다. 따라서 마이그레이션 후 영역에 대해 추가 네트워크 구성이 필요하지 않습니다.
- 일반 이름 지정 규약은 설치 중 사용되는 SC(시스템 구성) 매니페스트에서 지정된 네트워크 구성의 복잡성을 줄여줍니다. 주 네트워크 데이터 링크에는 일반적으로 모든 시스템에 대해 net0이라는 이름이 지정되므로 net0에 대한 구성을 지정하는 다중 시스템에 일반 SC 매니페스트를 사용할 수 있습니다.
- 데이터 링크 관리도 유연해졌습니다. 데이터 링크가 제공하는 특정 기능을 반영하도록 데이터 링크 이름 등을 추가로 사용자 정의할 수 있습니다.

다음 표에서는 링크를 통해 하드웨어(NIC), 장치 인스턴스, 링크 이름 및 인터페이스 간의 새로운 대응을 보여줍니다. 데이터 링크 이름은 OS에서 자동으로 제공됩니다.

하드웨어(NIC)	장치 인스턴스	링크의 지정된 이름	IP 인터페이스
e1000g	e1000g0	net0	net0
igb	ixgbe	net1	net1

이 표에 표시된 대로 장치 인스턴스 이름은 하드웨어 기반 이름으로 유지되지만 데이터 링크의 경우 설치 후 OS에 의해 이름이 바뀝니다.

데이터 링크와 일반 이름, 해당하는 장치 인스턴스 간의 매핑을 표시하려면 다음과 같이 `dladm show-phys` 명령을 사용하십시오.

```
# dladm show-phys
LINK    MEDIA      STATE    SPEED    DUPLEX    DEVICE
net2    Ethernet  up       1000    full     bge2
net0    Ethernet  up       1000    full     e1000g0
net3    Ethernet  up       1000    full     nge3
net1    Ethernet  up       1000    full     e1000g1
```

## 데이터 링크에 일반 이름 지정

- 물리적 네트워크 장치는 매체 유형에 따라 정렬됩니다. 이 경우 특정 유형이 다른 유형보다 높은 우선 순위를 갖습니다. 매체 유형은 다음과 같은 종속 우선 순위로 정렬됩니다.

1. 이더넷
2. InfiniBand 장치
3. Ethernet over IB
4. WiFi

- 매체 유형에 따라 장치를 그룹화하고 정렬한 후에는 물리적 위치에 따라 추가로 정렬되고, 내장 장치가 주변 장치보다 우선시됩니다.
- 매체 유형 및 위치에 따라 우선 순위가 높은 장치에는 낮은 인스턴스 번호가 지정됩니다.

SPARC 기반 시스템에서는 OBP(OpenBoot PROM)에서 사용되는 netN 장치 별칭과 일치하도록 netN 이름이 지정됩니다. x86 기반 시스템에서는 내장 이더넷 장치를 식별하여 net0, net1 등을 지정하는 데 SMBIOS 데이터(사용 가능한 경우)가 사용됩니다. 해당 정보 소스와 함께 또는 해당 정보 소스와 무관하게, 하위 마더보드 또는 IO 보드의 장치, 호스트 브리지, PCIe 루트 컴플렉스, 버스, 장치 및 기능에는 다른 장치보다 앞선 순위가 지정되며 상위 마더보드, 호스트 브리지 등의 장치보다 낮은 net 인스턴스가 지정됩니다.

링크 이름, 장치 및 위치의 지정된 이름을 표시하려면 다음과 같이 `dladm show-phys` 명령에 `-L` 옵션을 사용하십시오.

```
# dladm show-phys -L
LINK          DEVICE          LOCATION
net0          e1000g0         MB
net1          e1000g1         MB
net2          e1000g2         MB
net3          e1000g3         MB
net4          ibp0            MB/RISER0/PCIE0/PORT1
net5          ibp1            MB/RISER0/PCIE0/PORT2
net6          eoib2           MB/RISER0/PCIE0/PORT1/cloud-nm2gw-2/1A-ETH-2
net7          eoib4           MB/RISER0/PCIE0/PORT2/cloud-nm2gw-2/1A-ETH-2
```

## 운영 체제의 일반 링크 이름 지정 방법 사용자 정의



주의 - 일반 링크 이름의 자동 지정 방법은 Oracle Solaris를 설치하기 전에 사용자 정의해야 합니다. 설치 후에는 기존 구성을 제거해야만 기본 링크 이름을 사용자 정의할 수 있습니다.

Oracle Solaris는 링크 이름을 지정할 때 접두어 `net`을 사용합니다. 하지만 `eth`와 같이 선호하는 사용자 정의 접두어를 사용할 수 있습니다. 일반 링크 이름의 자동 지정을 사용 안함으로 설정할 수도 있습니다.

자동 링크 이름 지정을 사용 안함으로 설정하거나 링크 이름의 접두어를 사용자 정의하려면 SC(시스템 구성) 매니페스트에서 다음 등록 정보를 설정합니다. SC 매니페스트는 Oracle Solaris의 AI(자동 설치 프로그램) 기능에서 사용됩니다.

```
<service name="network/datalink-management"
version="1" type="service">
<instance name="default enabled="true">
<property_group name='linkname-policy'
type='application'>
<propval name='phys-prefix' type='astring'
value='net' />
</property_group>
</instance>
</service
```

기본적으로 `phys-prefix` 등록 정보에 대한 값은 이전 출력에서 굵게 표시된 것처럼 `net`으로 설정되어 있습니다.

- 자동 이름 지정을 사용 안함으로 설정하려면 다음과 같이 `phys-prefix` 등록 정보에 대한 값을 빈 문자열로 설정하십시오.

```
<propval name='phys-prefix' type='astring' value='' />
```

자동 이름 지정을 사용 안함으로 설정할 경우 데이터 링크 이름은 연결된 하드웨어 드라이버를 기반으로 합니다(예: `bge0`, `e1000g0` 등).

- `net` 이외의 접두어를 사용하려면 `phys-prefix`의 값으로 새 접두어를 지정합니다(예: `eth`).

`phys-prefix` 등록 정보에 대해 제공된 값이 잘못된 경우 해당 값이 무시됩니다. 그런 후 연결된 하드웨어 드라이버에 따라 데이터 링크 이름이 지정됩니다(예: `bge0`, `e1000g0` 등). 유효한 링크 이름에 대한 규칙은 “[유효한 링크 이름 규칙](#)” [23]을 참조하십시오.

## 업그레이드된 시스템의 링크 이름

새로 설치된 시스템에서는 데이터 링크 이름이 자동으로 `net0`부터 `netN-1`까지로 지정됩니다. 여기서 `N`은 네트워크 장치의 총 개수를 나타냅니다.

반대로, 다른 Oracle Solaris 11 릴리스에서 업그레이드한 경우에는 업그레이드 전에 설정된 이름이 데이터 링크에 그대로 유지됩니다. 이러한 이름은 기본 하드웨어 기반 이름이거나 관리자가 업그레이드 전에 데이터 링크에 지정한 사용자 정의 이름입니다. 또한 업그레이드된 시스템에서 이후에 추가하는 새로운 네트워크 장치도 일반 이름을 지정하는 대신 기본 하드웨어 기반의 이름이 보존됩니다. 업그레이드된 시스템의 이러한 동작은 OS에서 지정한 일반 이름이 다른 하드웨어 기반 이름 또는 관리자가 업그레이드하기 전에 지정한 사용자 정의된 이름과 혼용되지 않도록 보장합니다.

하드웨어 기반 이름과 OS 제공 링크 이름은 모두 사용자가 사용하려는 다른 이름으로 바꿀 수 있습니다. 일반적으로 OS에서 지정된 기본 링크 이름으로도 시스템의 네트워크 구성을 충분히 만들 수 있습니다. 하지만 링크 이름을 변경하려면 먼저 다음 정보를 고려하십시오.

### 하드웨어 기반 링크 이름 바꾸기

시스템의 링크에 하드웨어 기반 이름이 포함된 경우 해당 링크를 최소한 일반 이름으로 바꿉니다. 하드웨어 기반 이름을 유지할 경우 나중에 이러한 물리적 장치를 제거하거나 교체할 때 혼동이 발생할 수 있습니다.

예를 들어, 장치 `bge0`와 연결된 `bge0`이라는 링크 이름을 유지할 수 있습니다. 모든 링크 구성은 링크 이름을 참조하여 수행됩니다. 그런 후 NIC `bge`를 NIC `e1000g0`로 교체할 수 있습니다. 이전 장치의 링크 구성을 새로운 NIC `e1000g0`에 다시 적용하려면 링크 이름 `bge0`을 `e1000g0`에 다시 지정해야 할 수 있습니다. 하드웨어 기반 링크 이름인 `bge0`을 다른 연결된

NIC인 e1000g0과 조합하면 혼동이 발생할 수 있습니다. 하드웨어 기반이 아닌 이름을 사용하면 연결된 장치와 링크를 보다 쉽게 구분할 수 있습니다.

## 링크 이름 변경에 대한 주의 사항

하드웨어 기반 링크 이름을 바꾸는 것이 모범 사례라도 링크 이름을 바꾸기 전에는 신중한 계획이 필요합니다. 장치의 링크 이름을 바꾸어도 새로운 이름이 기존의 모든 연결된 구성에 자동으로 전파되지 않습니다. 다음 예에서는 링크 이름을 바꿀 때의 위험 요소들을 보여줍니다.

- IP 필터 구성에서 일부 규칙은 특정 링크에 적용됩니다. 링크 이름을 변경할 때는 필터 규칙이 링크의 원래 이름을 계속해서 참조합니다. 따라서 링크 이름을 바꾼 후에는 이러한 규칙이 예상한 대로 작동하지 않습니다. 새로운 링크 이름을 사용하여 링크에 적용할 필터 규칙을 조정해야 합니다.
- 네트워크 구성 정보를 내보낼 수 있는 가능성을 고려해야 합니다. 앞에서 설명한 것처럼 OS에서 제공한 기본 net# 이름을 사용하면 영역을 마이그레이션하고 네트워크 구성을 다른 시스템으로 쉽게 내보낼 수 있습니다. 대상 시스템의 네트워크 장치 이름을 net0, net1 등과 같은 일반 이름으로 지정한 경우, 영역에 지정된 데이터 링크와 이름이 일치하는 데이터 링크의 네트워크 구성이 영역에 상속됩니다.

따라서 일반적으로는 데이터 링크 이름을 임의로 바꾸지 마십시오. 데이터 링크 이름을 바꿀 때는 링크에 연결된 모든 구성이 링크 이름 변경 후에도 계속 적용되는지 확인합니다.

링크 이름 바꾸기로 영향을 받을 수 있는 일부 구성은 다음과 같습니다.

- IP 필터 규칙
- ipadm 명령을 사용하여 지정된 IP 구성
- Oracle Solaris 11 영역
- autopush 구성

---

참고 - 링크 이름을 바꿀 때 autopush 구성은 변경할 필요가 없습니다. 하지만 링크 이름을 바꾼 후 링크별 autopush 등록 정보에서 구성이 어떻게 작동하는지 확인해야 합니다. 자세한 내용은 [데이터 링크에 STREAMS 모듈 설정](#)을 참조하십시오.

---

## 유효한 링크 이름 규칙

링크 이름을 지정할 때는 다음과 같은 규칙을 따릅니다.

- 링크 이름은 문자열과 PPA(물리적 연결 지점) 번호로 구성되어야 합니다.
- 링크 이름은 다음과 같은 제약 조건을 따라야 합니다.
  - 이름은 이상적으로 3~8자로 구성됩니다. 하지만 이름은 최대 16자까지 지정할 수 있습니다.

- 유효한 이름 문자는 영숫자(a-z, 0-9) 및 밑줄(\_)입니다.



주의 - 링크 이름에는 대문자를 사용하지 마십시오.

- 각 데이터 링크는 링크 이름을 한 번에 하나만 포함해야 합니다.
- 각 데이터 링크는 시스템 내에서 고유한 링크 이름을 포함해야 합니다.

참고 - 추가된 제한 사항으로, `lo0`을 유연한 링크 이름으로 사용할 수 없습니다. 이 이름은 IP 루프백 인터페이스를 식별하도록 예약되었습니다.

네트워크 설정 내에서 링크 기능은 링크 이름을 지정할 때 유용한 참조로 활용할 수 있습니다. 예를 들어, `netmgt0`은 네트워크 관리 전용의 링크일 수 있습니다. `Upstream2`는 ISP에 연결하는 링크일 수 있습니다. 일반적으로 혼동을 피하기 위해서는 알려진 장치의 이름을 링크에 지정하지 마십시오.

## 데이터 링크 등록 정보 관리

`dladm` 명령을 사용하여 공통 데이터 링크 등록 정보를 사용자 정의하는 경우 다음 이점이 제공됩니다.

- `dladm` 명령은 네트워크 드라이버 등록 정보 구성에 필요한 유일한 명령 인터페이스입니다. 이 명령은 `ndd` 명령과 `driver.conf` 파일 수정 사항의 조합을 사용하여 드라이버 등록 정보를 설정하던 이전 방식을 대체합니다.
- 설정된 등록 정보에 관계없이 다음과 같은 동일한 구문이 사용됩니다.

`dladm subcommand properties datalink`

- `dladm` 명령 사용은 드라이버의 공용 및 개인 등록 정보에 모두 적용됩니다.
- 특정 드라이버에서 `dladm` 명령을 사용하는 경우 비슷한 유형의 다른 NIC 네트워크 연결은 중단되지 않습니다. 따라서 데이터 링크 등록 정보를 동적으로 구성할 수 있습니다.
- 데이터 링크 구성 값은 `dladm` 저장소에 저장되며 시스템을 재부트한 후에도 지속됩니다.

## 데이터 링크에 대한 일반 정보 표시

옵션 없이 사용되는 경우 `dladm` 명령은 클래스, 상태, 기본 물리적 링크를 포함하여 시스템의 데이터 링크에 대한 일반 정보를 표시합니다.

```
# dladm
LINK      CLASS    MTU      STATE    OVER
net0      phys     1500     unknown --
net1      phys     1500     up       --
net2      phys     1500     unknown --
```

```
net3      phys      1500    unknown  --
net4      phys      1500    up        --
aggr0     aggr      1500    up        net1,net4
```

데이터 링크는 링크 통합, VLAN(가상 LAN) 및 VNIC(가상 NIC) 등의 물리적 링크가 아닌 다른 클래스일 수 있습니다. 이러한 다른 데이터 링크도 `dladm` 명령으로 표시되는 기본 정보에 포함됩니다. 예를 들어, 이전 출력에서 링크 통합(`aggr0`)은 물리적 데이터 링크 `net1` 및 `net4`를 통해 구성되었습니다.

링크 통합 및 VLAN에 대한 자세한 내용은 [“Oracle Solaris 11.2의 네트워크 데이터 링크 관리”](#)를 참조하십시오. VNIC에 대한 자세한 내용은 [“Oracle Solaris 11.2의 네트워크 가상화 및 네트워크 리소스 관리”](#)를 참조하십시오.

## 시스템의 데이터 링크 표시

`dladm show-link` 명령을 사용하여 시스템의 물리적 데이터 링크와 가상 데이터 링크를 모두 표시할 수 있습니다. 시스템에는 설치된 NIC만큼 많은 데이터 링크가 포함됩니다. 이 명령에 다양한 옵션을 사용하여 표시되는 정보를 사용자 정의할 수 있습니다.

추가 옵션 또는 인수 없이 사용되는 경우 `dladm show-link` 명령은 다음 정보를 표시합니다.

```
# dladm show-link
LINK          CLASS      MTU      STATE    OVER
net1          phys      1500     down    --
net3          phys      1500     unknown --
net0          phys      1500     up      --
net2          phys      1500     unknown --
net11         phys      1500     up      --
net5          phys      1500     up      --
net6          phys      1500     up      --
```

이전 출력에서 `STATE` 열은 가상 데이터 링크의 현재 상태를 보여줍니다. 상태는 `up`, `down` 또는 `unknown`일 수 있습니다. 가상 데이터 링크의 경우 NIC가 여러 VNIC로 분할될 때 내부에서 가상 스위치가 암시적으로 만들어집니다. 이와 같이 가상 스위치가 만들어지면 VNIC와 주 데이터 링크가 동일한 VLAN을 사용하는 경우 서로 통신할 수 있습니다. 물리적 데이터 링크가 외부 네트워크에 연결되지 않은 경우에도 통신이 가능합니다. 이 관계를 통해 데이터 링크의 가상 상태가 구성됩니다.

`-P` 옵션을 사용하여 데이터 링크에 대한 지속 구성 정보를 표시할 수 있습니다. 이 명령으로 제공되는 정보에 따라 추가 네트워크 구성을 계속할 수 있습니다. 예를 들어, 시스템의 NIC 개수를 확인하고 IP 인터페이스를 구성하는 데 사용할 데이터 링크를 선택할 수 있습니다. 명령을 입력하는 경우 표시되는 정보는 다음 예와 유사합니다.

```
# dladm show-link -P
LINK          CLASS      OVER
net0          phys      --
net1          phys      --
net2          phys      --
```

이전 예에서는 시스템에 있는 3개의 데이터 링크가 해당하는 물리적 NIC와 직접적으로 연관되어 있음을 보여줍니다. `phys` 클래스 아래에 데이터 링크로 구성된 집계 또는 가상 NIC와 같은 특수한 데이터 링크는 존재하지 않습니다.

## 데이터 링크의 물리적 속성 표시

`dladm show-phys` 명령을 사용하여 데이터 링크에 연관된 물리적 NIC와 관련된 시스템 데이터 링크에 대한 정보를 가져올 수 있습니다. 옵션 없이 사용되는 경우 이 명령은 다음 예와 유사한 정보를 표시합니다.

```
# dladm show-phys
LINK      MEDIA      STATE      SPEED      DUPLEX      DEVICE
net0      Ethernet   up         100Mb     full       e1000g0
net1      Ethernet   down       0Mb       --         nge0
net2      Ethernet   up         100Mb     full       bge0
net3      InfiniBand --         0Mb       --         ibd0
```

이전 출력에서는 다른 세부 정보 외에도 일반 링크 이름을 사용하는 데이터 링크가 연관된 물리적 NIC를 보여줍니다. 예를 들어, `net0`은 NIC `e1000g0`의 데이터 링크 이름입니다. 데이터 링크에 대해 설정된 플래그에 대한 정보를 표시하려면 `-p` 옵션을 사용하십시오. 예를 들어, `r`로 플래그 지정된 데이터 링크는 기본 NIC가 제거되었음을 의미합니다.

이전 출력에서 `STATE` 열은 물리적 데이터 링크의 현재 상태를 보여줍니다. 상태는 `up`, `down` 또는 `unknown`일 수 있습니다. 물리적 링크 상태는 물리적 장치가 외부 네트워크에 연결되었는지 여부를 식별합니다. 케이블이 연결되고 케이블의 다른 쪽 끝에 있는 포트의 상태가 `up`인 경우 연결된 것입니다.

`-L` 옵션도 사용할 수 있는 유용한 옵션 중 하나입니다. 이 옵션은 각 데이터 링크에 대한 물리적 위치를 표시합니다. 이 위치에 따라 `net0`, `net1` 등과 같은 데이터 링크의 인스턴스 번호가 결정됩니다.

```
# dladm show-phys -L
LINK      DEVICE      LOCATION
net0      bge0        MB
net2      ibp0        MB/RISER0/PCIE0/PORT1
net3      ibp1        MB/RISER0/PCIE0/PORT2
net4      eoib2       MB/RISER0/PCIE0/PORT1/cloud-nm2gw-2/1A-ETH-2
```

`-m` 옵션을 사용하여 시스템의 물리적 링크에 대한 MAC 주소를 표시할 수 있습니다.

```
# dladm show-phys -m
LINK      SLOT      ADDRESS      INUSE CLIENT
net0      primary  0:11:22:a9:ee:66  yes  net0
```

이 명령은 `ifconfig` 명령을 사용하는 것과 비슷합니다.

다음과 같이 시스템의 모든 링크(물리적 및 비물리적)에 대한 MAC 주소를 표시합니다.

```
# dladm show-linkprop -p mac-address
```

LINK	PROPERTY	PERM	VALUE	EFFECTIVE	DEFAULT	POSSIBLE
net0	mac-address	rw	0:11:22:a9:ee:66	0:11:22:a9:ee:66	0:11:22:a9:ee:66	0:11:22:a9:ee:66
--						

## 데이터 링크 삭제

`dladm delete-phys` 명령을 사용하여 시스템에서 데이터 링크를 제거할 수 있습니다.

데이터 링크 제거는 물리적 NIC 제거와 느슨한 형태로만 연결되어 있습니다. 예를 들어, 물리적 NIC가 시스템에서 제거되는 경우 [“Oracle Solaris 10에서 Oracle Solaris 11.2로 전환”](#)의 [“Oracle Solaris 10 네트워크 프로토콜 스택과 Oracle Solaris 11 네트워크 프로토콜 스택 비교”](#)에 설명된 대로 소프트웨어 계층이 더 이상 하드웨어 계층에 바인드되지 않으므로 해당 NIC와 연관된 데이터 링크 구성은 유지됩니다. 따라서 이 데이터 링크의 이름을 다른 NIC에 연결된 링크에 지정하여 다른 기본 물리적 NIC에서 이 데이터 링크 구성을 사용할 수 있습니다.

NIC를 교체하지 않고 분리하는 경우 해당 데이터 링크 구성이 더 이상 필요하지 않으면 다음과 같이 데이터 링크를 삭제할 수 있습니다.

```
# dladm delete-phys datalink
```

---

**작은 정보** - 데이터 링크의 NIC가 제거되었는지 여부를 확인하려면 `dladm show-phys -P` 명령을 사용하십시오. 출력에 제공되는 `FLAGS` 열의 `r` 플래그는 물리적 링크와 연관된 물리적 장치가 제거되었는지 여부를 나타냅니다.

---

## 데이터 링크 이름 바꾸기

`dladm rename-link` 명령을 사용하여 데이터 링크의 이름을 바꿀 수 있습니다. Oracle Solaris 시스템에서 OS는 모든 데이터 링크에 일반 이름을 자동으로 제공합니다. 일반 데이터 링크 이름에 대한 자세한 내용은 [“데이터 링크 구성 정보” \[19\]](#)를 참조하십시오.

기본적으로 이러한 일반 이름은 `net0`, `net1`, `net2` 등과 같이 `netn`의 이름 지정 형식을 사용합니다. 이러한 이름은 OS에서 관리되므로 사용자가 일반적인 관리 작업의 일부로 데이터 링크 이름을 바꿀 필요가 없습니다. 링크 이름 변경이 필요한 절차는 [네트워크 장치 간에 IP 구성을 이동하는 방법 \[33\]](#)을 참조하십시오.

## 데이터 링크에 대한 런타임 통계 가져오기

`dlstat` 명령을 사용하여 모든 유형의 데이터 링크에 대한 런타임 데이터 링크 통계를 가져올 수 있습니다. 다른 옵션 없이 단독으로 사용되는 경우 `dlstat`는 다음 출력과 같이 시스템에 있는 모든 데이터 링크에 대한 통계 정보를 표시합니다.

```
% dlstat
      LINK      IPKTS      RBYTES      OPKTS      OBYTES
net0    58.00K      9.52M      5.61K      1.91M
```

dlstat 명령 사용에 대한 자세한 내용은 “Oracle Solaris 11.2의 네트워크 가상화 및 네트워크 리소스 관리”의 8 장, “네트워크 트래픽 및 리소스 사용 모니터링”를 참조하십시오. dlstat(1M) 매뉴얼 페이지도 참조하십시오.

## 데이터 링크 등록 정보 사용자 정의

기본 데이터 링크 구성을 수행하는 것은 물론 dladm 명령을 사용하여 데이터 링크 등록 정보를 설정하고 네트워크 요구 사항에 따라 사용자 정의할 수도 있습니다.

다음과 같은 3개의 dladm 하위 명령은 데이터 링크 등록 정보를 관리하는 데 사용됩니다.

`dladm show-linkprop -p property datalink`      데이터 링크의 등록 정보 및 현재 값을 표시합니다. `-p property` 옵션을 사용하지 않으면 데이터 링크의 모든 등록 정보가 표시됩니다. 데이터 링크를 지정하지 않으면 모든 데이터 링크의 등록 정보가 모두 표시됩니다.

`dladm set-linkprop -p property=value datalink`      데이터 링크의 등록 정보에 값을 지정합니다.

`dladm reset-linkprop -p property datalink`      데이터 링크의 특정 등록 정보를 기본값으로 재설정합니다.

사용자 정의할 수 있는 데이터 링크 등록 정보는 특정 NIC 드라이버가 지원하는 등록 정보에 따라 달라집니다.

dladm 명령을 사용하여 구성할 수 있는 데이터 링크 등록 정보는 다음 두 가지 범주 중 하나에 속합니다.

- **공용 등록 정보** - 해당 등록 정보는 링크 속도, 이더넷의 자동 협상 또는 모든 데이터 링크 드라이버에 적용할 수 있는 MTU(최대 전송 단위) 크기 등 지정된 매체 유형의 드라이버에 적용할 수 있습니다.
- **개인 등록 정보** - 해당 등록 정보는 지정된 매체 유형에 대한 NIC 드라이버의 특정 하위 세트에 따라 고유합니다. 이러한 등록 정보는 드라이버와 연결된 하드웨어 또는 드라이버 구현 자체의 세부 정보(예: 디버깅 관련 조정 가능 항목)와 긴밀한 관계가 있기 때문에 이 부분에만 관련된 것일 수 있습니다.

링크 등록 정보에는 일반적으로 기본값이 사용됩니다. 하지만 특정 네트워킹 시나리오에서는 특정 등록 정보 값을 변경해야 할 수도 있습니다. 예를 들어, NIC가 자동 협상을 제대로 수행하지 않는 이전 스위치와 통신 중일 수 있습니다. 또는 스위치가 점보 프레임도 지원하지 않음

구성되었을 수도 있습니다. 또는 패킷 전송이나 패킷 수신을 규제하는 드라이버별 등록 정보를 특정 드라이버에 맞게 수정해야 할 수도 있습니다.

## 정보 프레임 지원 사용

MTU는 프로토콜이 시스템에서 전송할 수 있는 최대 패킷 크기를 정의합니다. 기본적으로 대부분의 NIC 드라이버는 MTU 크기를 1500으로 정의합니다. 하지만 정보 프레임이 네트워크를 순회하는 경우 기본값으로는 충분하지 않습니다. 정보 프레임을 지원하려면 MTU 크기가 적어도 9000 이상이어야 합니다.

**참고** - MTU 등록 정보는 데이터 링크와 IP 인터페이스에 공동됩니다. 따라서 데이터 링크에 대해 특정 MTU 값을 사용하고, 해당 링크를 통해 구성된 IP 인터페이스에 대해 다른 MTU 값을 사용할 수 있습니다. 데이터 링크 MTU 값은 IP 인터페이스의 MTU에 대해 설정할 수 있는 값에 영향을 끼칩니다. 데이터 링크 및 IP 인터페이스의 MTU 등록 정보를 구성할 때 이 동작이 끼치는 영향에 대한 자세한 내용은 [“MTU 등록 정보 설정” \[65\]](#)을 참조하십시오.

다음과 같이 MTU 크기의 기본값을 변경하십시오.

```
# dladm set-linkprop -p mtu=new-size datalink
```

MTU 크기를 변경한 후에는 데이터 링크에 대한 IP 인터페이스를 재구성할 수 있습니다.

다음 예에서는 정보 프레임에 대한 지원을 사용으로 설정하는 방법을 보여줍니다. 이 예에서는 데이터 링크를 통한 기존 IP 인터페이스 구성을 이미 제거했다고 가정합니다.

```
# dladm show-linkprop -p mtu net1
LINK      PROPERTY  PERM VALUE      EFFECTIVE  DEFAULT  POSSIBLE
net1      mtu       rw   1500           1500      1500     1500
```

```
# dladm set-linkprop -p mtu=9000 net1
# dladm show-link net1
LINK      CLASS     MTU      STATE  BRIDGE  OVER
web1     phys     9000    up     --      --
```

## 링크 속도 매개변수 수정

대부분의 네트워크 설정은 다양한 속도 기능의 시스템 조합으로 구성됩니다. 각 시스템은 네트워크의 다른 시스템에 각 시스템이 네트워크 트래픽을 전송 및 수신할 수 있는 속도 기능을 알립니다.

다음 두 가지 데이터 링크 등록 정보는 시스템이 알리는 속도 기능을 규제합니다.

- `adv_10gfdx_cap/en_10gfdx_cap`
- `adv_1000fdx_cap/en_1000fdx_cap`

- `adv_1000hdx_cap/en_1000hdx_cap`
- `adv_100fdx_cap/en_100fdx_cap`
- `adv_100hdx_cap/en_100hdx_cap`
- `adv_10fdx_cap/en_10fdx_cap`
- `adv_10hdx_cap/en_10hdx_cap`

각 링크 속도 기능은 알려진 속도(`adv_*_cap`)와 사용으로 설정된 알려진 속도(`en_*_cap`)의 등록 정보 쌍으로 참조됩니다. 또한 데이터 링크 속도 정보는 등록 정보 이름에서 `*fdx*` 및 `*hdx*`로 지정된 대로 전이중 기능과 반이중 기능 모두에 제공됩니다. 알려진 속도 등록 정보는 특정 데이터 링크 속도가 알려졌는지 여부를 나타내는 읽기 전용 등록 정보입니다. 특정 데이터 링크 속도를 알릴지 여부는 해당하는 `en_*_cap` 등록 정보를 설정하여 결정합니다.

기본적으로 데이터 링크의 모든 속도와 이중 기능이 알려집니다. 하지만 새로운 시스템이 이전 시스템과 통신하고 자동 협상이 사용 안함으로 설정되었거나 지원되지 않는 경우도 존재할 수 있습니다. 이러한 두 시스템 간의 통신을 사용으로 설정하기 위해서는 이전 시스템과 최신 시스템 간에 알려진 속도를 하위 값으로 변경해야 할 수 있습니다. 시스템의 기가비트 기능을 해제하고 느린 속도 기능만 알려야 할 수 있습니다. 이 경우 다음 명령을 입력하여 전이중 기능과 반이중 기능에 대한 기가비트 기능 알림을 해제합니다.

```
# dladm set-linkprop -p en_1000fdx_cap=0 datalink
# dladm set-linkprop -p en_1000hdx_cap=0 datalink
```

해당 등록 정보의 새 값을 표시하려면 다음과 같이 `dladm show-linkprop` 명령을 사용하십시오.

```
# dladm show-linkprop -p adv_10gfdx_cap datalink
# dladm show-linkprop -p adv_1000hdx_cap datalink
```

일반적으로 사용으로 설정된 지정된 속도 등록 정보와 해당하는 알려진 등록 정보의 값은 동일합니다. 하지만 NIC가 전원 관리 등의 고급 기능을 지원하는 경우 해당 기능이 호스트와 링크 파트너 간에 실제로 알려지는 비트에 제한을 설정할 수도 있습니다. 예를 들어, 전원 관리를 사용할 경우 `adv_*_cap` 등록 정보의 설정이 `en_*_cap` 등록 정보 설정의 일부일 뿐입니다.

## 데이터 링크에 STREAMS 모듈 설정

데이터 링크를 열 때 스트림에 최대 8개의 STREAMS 모듈이 푸시되도록 설정할 수 있습니다. 일반적으로 이러한 모듈은 VPN(가상 사설망) 및 방화벽과 같은 타사 네트워킹 소프트웨어에 사용됩니다. 네트워킹 소프트웨어에 대한 설명서는 소프트웨어 공급업체가 제공합니다.

특정 데이터 링크에 푸시할 모듈 목록은 `autopush` 등록 정보로 제어됩니다. `autopush` 등록 정보의 값은 `dladm set-linkprop` 명령을 사용하여 설정됩니다.

별도의 `autopush` 명령을 사용하여 드라이버별로 데이터 링크의 스트림에 모듈을 푸시할 수도 있습니다. 이 명령은 각 드라이버에 대해 설정되었고 푸시할 모듈을 명령에 알려 주는 구

성 파일을 사용합니다. 하지만 드라이버는 항상 NIC에 바인딩됩니다. 데이터 링크의 기본 NIC를 제거하면 링크의 autopush 등록 정보에 대한 정보도 손실됩니다.

따라서 이 용도로는 autopush 명령을 사용하는 것보다 dladm 명령을 사용하는 것이 더 적합합니다. 특정 데이터 링크에 대해 드라이버별 및 링크별 autopush 구성 유형이 있는 경우 dladm set-linkprop 명령으로 설정된 링크별 정보가 사용되며 드라이버별 정보는 무시됩니다.

데이터 링크가 열릴 때 STREAMS에 모듈을 푸시하려면 동일한 dladm set-linkprop 명령을 사용하여 autopush 등록 정보에 대한 모듈을 지정하십시오. 예를 들어, 다음과 같이 vpnmod 및 bufmod 모듈을 net0 링크 위에 푸시합니다.

```
# dladm set-linkprop -p autopush=vpnmod.bufmod net0
```

## 데이터 링크 등록 정보에 대한 상태 정보 가져오기

데이터 링크 등록 정보에 대한 정보를 가져오기 위해서는 다음 명령 중 하나를 사용할 수 있습니다.

- dladm show-linkprop -p *property datalink*
- dladm show-ether *datalink*

## 데이터 링크 등록 정보 표시

데이터 링크 등록 정보의 전체 목록을 표시하려면 다음 예에서와 같이 등록 정보를 지정하지 않은 상태로 명령을 입력하십시오.

```
# dladm show-linkprop net1
LINK      PROPERTY      PERM VALUE      EFFECTIVE      DEFAULT      POSSIBLE
net1      speed         r-  0          0             0            --
net1      autopush     rw  --         --            --            --
net1      zone         rw  --         --            --            --
net1      duplex      r- unknown    unknown       unknown      half,full
net1      state       r- up         up            up            up,down
net1      adv_autoneg_cap -- --         --            0            1,0
net1      mtu         rw  1500        1500         1500         1500
net1      flowctrl    -- --         --            no           no,tx,rx,bi,
                                         pfc,auto
net1      adv_10gfdx_cap r- --         --            0            1,0
net1      en_10gfdx_cap -- --         --            0            1,0
net1      adv_1000fdx_cap r- --         --            0            1,0
net1      en_1000fdx_cap -- --         --            0            1,0
net1      adv_1000hdx_cap r- --         --            0            1,0
net1      en_1000hdx_cap -- --         --            0            1,0
net1      adv_100fdx_cap r- --         --            0            1,0
net1      en_100fdx_cap -- --         --            0            1,0
net1      adv_100hdx_cap r- --         --            0            1,0
net1      en_100hdx_cap -- --         --            0            1,0
```

```
net1    adv_10fdx_cap  r-  --      --      0      1,0
net1    en_10fdx_cap  --  --      --      0      1,0
net1    adv_10hdx_cap  r-  --      --      0      1,0
net1    en_10hdx_cap  --  --      --      0      1,0
net1    maxbw         rw  --      --      --     --
net1    cpus         rw  --      --      --     --
```

## 이더넷 등록 정보 값 표시

dladm show-ether 명령에 옵션을 지정하지 않을 경우 데이터 링크에 대한 현재 이더넷 등록 정보 값만 표시됩니다. 기본적으로 제공되는 정보 이외의 추가 정보를 얻으려면 다음 예에서와 같이 -x 옵션을 사용하십시오.

```
# dladm show-ether -x net1
LINK    PTYPE      STATE     AUTO  SPEED-DUPLEX          PAUSE
net1    current    up        yes   1G-f                  both
--      capable    --        yes   1G-fh,100M-fh,10M-fh both
--      adv        --        yes   100M-fh,10M-fh       both
--      peeradv    --        yes   100M-f,10M-f         both
```

-x 옵션을 사용하면 이 명령은 지정한 링크의 내장 기능뿐 아니라 호스트와 링크 파트너 간에 현재 알려진 기능도 표시합니다.

이 예에서는 다음 정보가 표시됩니다.

- 현재 이더넷 장치 상태의 경우 링크가 전이중에서 초당 1기가비트 속도로 작동하고 있습니다. 자동 협상 기능이 사용으로 설정되었으며 호스트와 링크 파트너가 일시 중지 프레임 보내고 받을 수 있는 양방향 플로우 제어가 있습니다. 이 정보는 출력의 첫번째 행에 표시됩니다.
- 출력 예의 이후 행에는 데이터 링크 속도 기능 및 알려진 실제 데이터 링크 속도에 대한 정보뿐만 아니라 다음과 같은 피어 시스템의 정보가 표시됩니다.
  - 이더넷 장치의 기능이 나열됩니다. 협상 유형은 자동으로 설정할 수 있습니다. 또한 장치가 전이중 및 반이중에서 초당 1기가비트, 초당 100메가비트 및 초당 10메가비트 속도를 지원할 수 있습니다. 마찬가지로, 호스트와 링크 파트너 간에 일시 중지 프레임을 양방향으로 보내거나 받을 수 있습니다.
  - net1의 기능은 자동 협상, 속도-이중 및 일시 중지 프레임의 플로우 제어로 알려집니다.
  - 이와 유사하게, net1의 링크 또는 피어 파트너도 자동 협상, 속도-이중 및 일시 중지 프레임의 플로우 제어를 알립니다.

## 추가 dladm 구성 작업

이 절에서는 주 인터페이스 전환 및 DR(동적 재구성) 수행과 같이 dladm 명령을 사용하여 간소화된 추가 구성 절차에 대해 설명합니다.

## ▼ 네트워크 장치 간에 IP 구성을 이동하는 방법

특정 네트워크 장치와 연관된 IP 구성을 보존한 다음 해당 구성을 다른 네트워크 장치로 이동해야 하는 경우 다음 절차를 사용하십시오. 시스템에서 네트워크 카드를 제거하기 위한 선행 과정으로 또는 네트워크 케이블 연결을 변경할 때 이 절차를 수행할 수 있습니다.

이 절차에서는 net0(e1000g0) 장치와 연관된 IP 구성을 보존하고 nge0 장치에 해당 구성을 적용하는 방법에 대해 설명합니다(예로만 사용됨).

1. 관리자로 로그인합니다.
2. 시스템의 물리적 링크와 장치 간 매핑을 표시합니다.  
다음 예에서는 어떤 이유로든 e1000g0에 대한 IP 구성이 종료되어 구성을 nge0으로 이동해야 한다고 가정합니다.

```
# dladm show-phys
LINK  MEDIA  STATE  SPEED  DUPLEX  DEVICE
net0  Ethernet down   0      unknown e1000g0
net1  Ethernet down   0      unknown e1000g1
net2  Ethernet up    1000   full    nge0
net3  Ethernet down   0      unknown nge1
```

3. 지속 설정은 그대로 유지하면서 일시적으로 데이터 링크에 대한 IP 구성을 사용 안함으로 설정합니다.

```
# ipadm disable-if interface
```

예를 들어, 다음과 같이 net0에 대한 IP 구성을 사용 안함으로 설정합니다.

```
# ipadm disable-if net0
```

이 단계에서는 IP 구성을 다시 만들지 않고 데이터 링크의 이름을 바꿀 수 있습니다.

4. 데이터 링크의 이름을 바꿉니다.

예를 들어, 다음과 같이 net0 데이터 링크의 이름을 바꿀 수 있습니다.

```
# dladm rename-link net0 oldnet0
```

5. 주 장치가 되도록 지정된 데이터 링크에 주 링크 이름을 지정합니다.

```
# dladm rename-link new-link primary-link
```

예를 들어, 다음과 같이 net2 데이터 링크에 net0 링크 이름을 재지정합니다.

```
# dladm rename-link net2 net0
```

6. 새로운 이름 데이터 링크에 대한 IP 구성을 다시 사용으로 설정합니다. 예를 들면 다음과 같습니다.

```
# ipadm enable-if -t net0
```

#### 예 2-1 데이터 링크 인터페이스 제거

새로 설치하는 경우 시스템의 총 네트워크 장치 수에 따라 시스템에 있는 모든 데이터 링크에는 자동으로 이름 지정 규약을 사용하는 일반 이름(net0, net1, netN 등)이 지정됩니다. 설치 후 데이터 링크에 다른 이름을 지정할 수 있습니다. 다음 예에서는 인터페이스에 대해 처음에 제공된 IP 주소를 변경하는 방법을 보여줍니다. 먼저 기존 인터페이스를 제거하는 작업이 수행됩니다.

```
# ipadm delete-ip net0
# ipadm create-ip net0
# ipadm create-addr -T addrconf net0/new-add
```

자세한 내용은 [Chapter 2, Oracle Solaris의 데이터 링크 구성 관리](#)를 참조하십시오.

## ▼ 동적 재구성을 사용하여 네트워크 인터페이스 카드를 교체하는 방법

다음 절차는 DR(동적 재구성)을 지원하는 시스템에만 적용됩니다. 특히 DR이 완료된 후 구성 단계를 나타냅니다. 이제 DR 프로세스를 완료한 후 네트워크 링크를 재구성할 필요가 없습니다. 대신 제거된 NIC의 링크 구성만 교체 NIC에 전송합니다.

이 절차에서는 DR 자체를 수행하는 단계에 대해 다루지 않습니다. 해당 정보는 시스템 설명서를 참조하십시오.

DR 소개는 “[Oracle Solaris 11.2의 장치 관리](#)”의 2 장, “[동적으로 장치 구성](#)”을 참조하십시오.

시작하기 전에 먼저 다음 단계를 완료했는지 확인하십시오.

- 시스템이 DR을 지원하는지 확인합니다.
- 시스템의 DR에 대해 설명하는 해당 매뉴얼을 참조하십시오.

Oracle Sun 서버의 DR에 대한 현재 설명서를 찾으려면 <http://www.oracle.com/technetwork/indexes/documentation/index.html>에서 "dynamic reconfiguration"을 검색하십시오.

Oracle Solaris Cluster 환경에서 DR 수행에 대한 자세한 내용은 “[Oracle Solaris Cluster 시스템 관리 설명서](#)”를 참조하십시오.

1. 관리자로 로그인합니다.
2. (옵션) 시스템에서 데이터 링크의 물리적 속성 및 해당 위치에 대한 정보를 표시합니다.

```
# dladm show-phys -L
```

dladm show-phys -L 명령으로 표시되는 정보의 유형에 대한 자세한 내용은 [dladm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

3. 시스템 설명서에 설명된 대로 DR 프로세스를 수행합니다.
4. 교체 NIC를 설치한 후 다음 중 해당하는 단계를 진행합니다.
  - 이전 NIC와 동일한 슬롯에 교체 NIC를 삽입한 경우 5단계를 진행합니다.  
새 NIC가 이전 NIC에서 사용한 것과 동일한 위치를 사용하는 경우 새 NIC가 이전 NIC의 링크 이름과 구성을 상속합니다.
  - 다른 슬롯에 교체 NIC를 삽입했고 새 NIC가 제거되는 NIC의 데이터 링크 구성을 상속해야 하는 경우 다음과 같이 링크의 이름을 바꿉니다.

```
# dladm rename-link new-datalink old-datalink
```

*new-datalink*            이전 NIC가 제거된 위치와 다른 슬롯에 있는 교체 NIC의 데이터 링크를 참조합니다.

*old-datalink*            제거된 이전 NIC와 연관된 데이터 링크 이름을 나타냅니다.

---

참고 - 이 시나리오에서는 이전 NIC를 제거한 슬롯을 비워 두어야 합니다.

---

예를 들어, 슬롯 1의 NIC가 제거된 후 새 NIC가 슬롯 2에 삽입되어 있습니다. 이 때, 슬롯 1에는 삽입된 NIC가 없습니다. 슬롯 1의 데이터 링크가 *net0*이라고 가정하면 슬롯 2의 데이터 링크는 *net1*입니다. 다음과 같이 새 NIC의 데이터 링크가 이전 NIC의 데이터 링크 구성을 상속하도록 지정합니다.

```
# dladm rename-link net1 net0
```

5. 사용 가능한 상태가 되도록 새 NIC의 리소스를 사용으로 설정하여 DR 프로세스를 완료합니다.  
예를 들어, *cfgadm* 명령을 사용하여 NIC를 구성할 수 있습니다. 자세한 내용은 *cfgadm(1M)* 매뉴얼 페이지를 참조하십시오.
6. (옵션) 링크 정보를 표시합니다.  
*dladm show-phys* 명령 또는 *dladm show-link* 명령을 사용하여 데이터 링크 정보를 표시할 수 있습니다.

**예 2-2** 새 네트워크 카드를 설치하여 동적 재구성 수행

다음 예에서는 링크 이름이 *net0*인 *bge* 카드가 *e1000g* 카드로 교체되는 방법을 보여줍니다. *e1000g*가 시스템에 연결된 후 *net0*의 링크 구성이 *bge*에서 *e1000g*로 전송됩니다.

```
# dladm show-phys -L
LINK    DEVICE    LOCATION
net0    bge0      MB
net1    ibp0      MB/RISER0/PCIE0/PORT1
net2    ibp1      MB/RISER0/PCIE0/PORT2
```

```
net3      eoib2      MB/RISER0/PCIE0/PORT1/cloud-nm2gw-2/1A-ETH-2
```

cfgadm 명령을 사용하여 bge 카드를 제거한 후 해당 위치에 e1000g 카드를 설치하는 등의 DR 관련 단계를 수행합니다. 카드가 설치된 후 e1000g0의 데이터 링크는 자동으로 net0이라는 이름을 사용하며 링크 구성을 상속합니다.

```
# dladm show-phys -L
LINK      DEVICE      LOCATION
net0      e1000g0     MB
net1      ibp0        MB/RISER0/PCIE0/PORT1
net2      ibp1        MB/RISER0/PCIE0/PORT2
net3      eoib2       MB/RISER0/PCIE0/PORT1/cloud-nm2gw-2/1A-ETH-2
```

```
# dladm show-link
LINK      CLASS      MTU      STATE      OVER
net0      phys       9600     up         ---
net1      phys       1500     down      ---
net2      phys       1500     down      --
net3      phys       1500     down      ---
```

## ▼ SPARC: 각 인터페이스의 MAC 주소가 고유한지 확인하는 방법

모든 SPARC 기반 시스템에는 시스템 차원의 MAC 주소가 있으며, 기본적으로 모든 인터페이스가 이 주소를 사용합니다. 하지만 일부 응용 프로그램에서는 호스트의 각 인터페이스가 고유한 MAC 주소를 가져야 합니다. 링크 통합 및 IPMP(IP Network Multipathing)와 같은 특정 유형의 인터페이스 구성에서는 인터페이스가 고유한 MAC 주소를 포함해야 합니다.

EEPROM 매개변수 local-mac-address?는 SPARC 기반 시스템의 모든 인터페이스가 시스템 차원의 MAC 주소를 사용하는지 아니면 고유한 MAC 주소를 사용하는지 확인합니다. 다음 절차에서는 eeprom 명령을 사용하여 local-mac-address? 매개변수의 현재 값을 확인하고 필요한 경우 변경하는 방법에 대해 설명합니다.

1. 관리자로 로그인합니다.
2. 시스템의 모든 인터페이스가 현재 시스템 차원의 MAC 주소를 사용하는지 여부를 확인합니다.

```
# eeprom local-mac-address?
local-mac-address?=false
```

이전 출력에서 local-mac-address?=false 설정은 모든 인터페이스가 시스템 차원의 MAC 주소를 사용함을 나타냅니다. local-mac-address?=false 설정의 값을 local-mac-address?=true로 변경해야만 인터페이스가 IPMP 그룹 등에 속할 수 있습니다.

---

참고 - 링크 통합을 구성할 때도 이 항목을 변경해야 합니다.

---

3. 다음과 같이 `local-mac-address?` 설정의 값을 변경합니다.

```
# eeprom local-mac-address?=true
```

시스템을 재부트하면 출하 시 설치된 MAC 주소를 사용하는 인터페이스가 시스템 차원의 MAC 주소가 아닌 이러한 출하 시 설정을 사용합니다. 출하 시 설치된 MAC 주소가 없는 인터페이스는 계속해서 시스템 차원의 MAC 주소를 사용합니다.

4. 시스템에 있는 모든 인터페이스의 MAC 주소를 확인합니다.

여러 인터페이스가 동일한 MAC 주소를 가진 경우를 찾습니다. 다음 예에서는 두 인터페이스가 시스템 차원의 MAC 주소인 `8:0:20:0:0:1`을 사용합니다.

```
# dladm show-linkprop -p mac-address
LINK      PROPERTY      PERM VALUE      EFFECTIVE      DEFAULT      POSSIBLE
net0      mac-address   rw  0:14:4f:f9:b1:a9 0:14:4f:f9:b1:a9 0:14:4f:f9:b1:a9 --
net3      mac-address   rw  0:14:4f:fb:9a:d4 0:14:4f:fb:9a:d4 0:14:4f:fb:9a:d4 --
net2      mac-address   rw  0:14:4f:f9:c:d 0:14:4f:f9:c:d 0:14:4f:f9:c:d --
net1      mac-address   rw  0:14:4f:fa:ea:42 0:14:4f:fa:ea:42 0:14:4f:fa:ea:42 --
```

5. (옵션) 필요한 경우 모든 인터페이스가 고유한 MAC 주소를 갖도록 나머지 인터페이스를 수동으로 구성합니다.

```
# dladm set-linkprop -p mac-address=mac-address interface
```

---

참고 - 이 단계는 두 개 이상의 네트워크 인터페이스가 동일한 MAC 주소를 사용하는 경우에만 필요합니다.

---

이전 예에서는 로컬에서 관리되는 MAC 주소로 `net0` 및 `net1`을 구성해야 합니다. 예를 들어, 로컬에서 관리되는 MAC 주소 `06:05:04:03:02`를 사용하여 `net0`을 재구성하려면 다음 명령을 입력합니다.

```
# dladm set-linkprop -p mac-address=06:05:04:03:02 net0
```

6. 시스템을 재부트합니다.



# ◆◆◆ 3 장 3

## Oracle Solaris에서 IP 인터페이스와 주소 구성 및 관리

---

이 장에서는 IPv4 및 IPv6 주소 지정을 구현할 네트워크를 구성하는 방법에 대해 설명합니다. 이 장에서 설명되는 대부분의 작업은 IPv4 및 IPv6 사용 네트워크에 모두 적용됩니다. IPv4 또는 IPv6 네트워크와만 관련된 절차는 해당 내용이 명시됩니다.

네트워크를 구성하기 전에 “Oracle Solaris 11.2의 네트워크 배치 계획”에서 설명된 IP 관련 계획 작업을 검토하십시오.

전역 패킷 전달 및 전송 계층 서비스와 같은 기타 TCP/IP 등록 정보를 관리하는 것과 관련된 자세한 내용은 “Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”의 “전송 계층 서비스 관리”를 참조하십시오.

이 장의 내용:

- “ipadm 명령을 사용하여 네트워크 구성 관리” [39]
- “IPv4 인터페이스 구성” [40]
- “IPv6 인터페이스 구성” [45]
- “IPv4 네트워크에서 IPv6 네트워크로 마이그레이션” [52]
- “경로 지정 구성” [53]
- “멀티홈 호스트 구성” [61]
- “IP 인터페이스 등록 정보 및 주소 사용자 정의” [64]
- “IP 주소 등록 정보 사용자 정의” [66]
- “IP 인터페이스 구성 사용 안함, 제거 및 수정” [67]
- “IP 인터페이스 및 주소 모니터링” [69]

### ipadm 명령을 사용하여 네트워크 구성 관리

ipadm 명령은 궁극적으로 IP 인터페이스 구성을 위한 기본 수단으로서 ifconfig 명령을 대체하도록 도입되었습니다.

또한 ipadm 명령은 다음과 같은 TCP/IP 프로토콜의 등록 정보를 구성하는 ndd 명령을 대체합니다.

- IP

- ARP(주소 결정 프로토콜)
- SCTP(흐름 제어 전송 프로토콜)
- ICMP(인터넷 제어 메시징 프로토콜)
- TCP 및 UDP(사용자 데이터그램 프로토콜)와 같은 상위 계층 프로토콜

인터페이스 구성 도구로서 ipadm 명령은 ifconfig 명령을 통해 다음 이점을 제공합니다.

- ifconfig 명령이 제공하는 구조보다 효과적인 객체 지향 하위 명령 구조를 제공합니다. 이 변경으로 인해 궁극적으로 네트워크 구성 절차를 더 쉽게 이해할 수 있게 됩니다.
- ifconfig 명령과 달리 네트워크 구성 변경 사항이 지속되도록 할 수 있습니다.
- 스크립팅에 유용할 수 있는 구문 분석 가능 출력 옵션을 지원합니다.

프로토콜 등록 정보를 설정하는 도구로서 ipadm 명령은 ndd 명령에 비해 다음과 같은 이점을 제공합니다.

- 등록 정보에 대해 ndd 명령보다 포괄적인 정보(예: 등록 정보의 현재 값 및 기본값과 가능한 값 범위)를 제공합니다.
- 등록 정보 값을 영구적(또는 일시적)으로 설정합니다. ndd 명령은 등록 정보 값을 일시적으로만 설정합니다.
- 스크립팅에 유용할 수 있는 구문 분석 가능 출력 옵션을 지원합니다.

ipadm 명령과 ifconfig 및 ndd 명령을 비교하려면 [“Oracle Solaris 10에서 Oracle Solaris 11.2로 전환”](#)의 [“네트워크 관리 명령 변경 사항”](#)을 참조하십시오.

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

## IPv4 인터페이스 구성

다음 절차 및 예에서는 IPv4 주소를 사용하는 네트워크의 구성 방법에 대해 설명합니다.

### ▼ IPv4 인터페이스를 구성하는 방법

시작하기 전에 구성을 올바른 프로파일에 적용 중인지 확인하기 위해서는 시스템에서 활성 상태인 NCP를 확인하십시오. [예 6-6. “고정적 모드와 반응적 모드 간 전환”](#)을 참조하십시오.

1. 관리자로 로그인합니다.
2. 인터페이스를 만듭니다.

```
# ipadm create-interface-class interface
```

*interface-class*            만들 수 있는 세 가지 인터페이스 클래스 중 하나를 나타냅니다.

- IP 인터페이스

이 인터페이스 클래스는 네트워크 구성을 수행할 때 만드는 가장 일반적인 클래스입니다. 이 인터페이스 클래스를 만들려면 `create-ip` 하위 명령을 사용합니다.

- STREAMS 가상 네트워크 인터페이스(VNI 인터페이스)

이 인터페이스 클래스를 만들려면 `create-vni` 하위 명령을 사용합니다.

Oracle Solaris 11.2부터 VNI 인터페이스 이름을 보다 임의적으로 지정할 수 있습니다. 이전에는 VNI 인터페이스 이름의 접두어에 "vni"가 포함되어야 했습니다(예: vni0). 이 요구 사항은 더 이상 적용되지 않습니다. VNI 장치 및 인터페이스에 대한 자세한 내용은 [vni\(7d\)](#) 및 [ipadm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

- IPMP 인터페이스

이 인터페이스 클래스는 IPMP 그룹을 구성할 때 사용됩니다. 이 인터페이스 클래스를 만들려면 `create-ipmp` 하위 명령을 사용합니다. IPMP 그룹에 대한 자세한 내용은 “Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”의 2 장, “IPMP 관리 정보”를 참조하십시오.

*interface*

인터페이스의 이름을 나타냅니다. 이 이름은 인터페이스를 만들 데이터 링크의 이름과 같습니다. 시스템에 있는 데이터 링크를 표시하려면 `dladm show-link` 명령을 사용합니다.

### 3. 다음 명령 중 하나를 사용하여 유효한 IP 주소로 IP 인터페이스를 구성합니다.

- 정적 IP 주소를 구성합니다.

```
# ipadm create-addr -a address [interface | addrobj]
```

`-a address`                    인터페이스에 구성할 IP 주소를 지정합니다.

---

**참고** - 터널 구성을 위해서는 일반적으로 터널 인터페이스에 대한 두 주소(로컬 주소 및 원격 주소)가 필요합니다. 로컬 및 원격 주소와 터널 구성에 대한 자세한 내용은 “Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”의 5 장, “IP 터널 관리”를 참조하십시오.

---

숫자 IP 주소의 경우 CIDR(Classless Inter-Domain Routing) 표기법을 사용합니다. CIDR 표기법을 사용하지 않을 경우 `svc:/system/name-service/switch:default` 넷마스크 데이터베이스 검색 순서를 사용하거나 클래스 주소 의미 체계를 사용하여 넷마스크가 결정됩니다.

선택적으로 숫자 IP 주소 대신 호스트 이름을 지정할 수 있습니다. `/etc/hosts` 파일에서 해당 호스트 이름에 대해 숫자 IP 주소가 정의된 경우 호스트 이름을 사용해도 유효합니다. 파일에 숫자 IP 주소가 정의되지 않은 경우 `name-service/switch` 서비스에서 `host에`

대해 지정된 분석기 순서를 사용하여 숫자 값을 고유하게 가져옵니다. 지정된 호스트 이름에 대한 항목이 여러 개 있으면 오류가 생성됩니다.

**참고** - 부트 프로세스 중 이름 지정 서비스가 온라인으로 전환되기 전에 IP 주소가 구성됩니다. 따라서 네트워크 구성에서 사용된 호스트 이름이 `/etc/hosts` 파일에 정의되어 있는지 확인해야 합니다.

`[interface | addrobj]` Oracle Solaris에서 각 주소는 해당하는 주소 객체로 식별되며 명령에 `addrobj`로 표시됩니다. 주소에 대한 이후 구성을 위해서는 실제 IP 주소 대신 주소 객체를 참조합니다. 예를 들어, `ipadm show-addr addrobj` 또는 `ipadm delete-addr addrobj`를 입력합니다. 주소 객체 이름이 자동으로 생성되도록 하려면 `interface`에 대한 인터페이스 이름만 지정합니다. 수동으로 주소 객체 이름을 지정하려면 주소 객체 이름을 직접 제공합니다.

- 인터페이스 이름을 지정하는 경우 주소 객체 이름이 자동으로 `interface/address-family` 형식으로 바뀝니다. `Address family`는 IPv4 주소의 경우 `v4`이고 IPv6 주소의 경우 `v6`입니다. 자동으로 생성된 주소 객체 이름을 사용하여 인터페이스에서 여러 주소가 구성된 경우 주소 객체 이름이 고유하도록 주소 객체 이름에 알파벳이 추가됩니다. 예를 들어, `net0/v4`, `net0/v4a`, `net0/v4b`, `net0/v6`, `net0/v6a` 등입니다.
- `addrobj`에 대한 주소 객체 이름을 수동으로 지정하는 경우에는 `interface/user-specified-string` 형식을 사용해야 합니다. `User-specified-string`은 영문자로 시작하고 최대 길이가 32자인 영숫자의 문자열을 나타냅니다. 예를 들어, `net0/static`, `net0/static1`, `net1/private` 등으로 주소 객체 이름을 지정할 수 있습니다.

- 비정적 주소를 구성합니다.

```
# ipadm create-addr -T address-type [interface | addrobj]
```

여기서 `address-type`은 `dhcp` 또는 `addrconf`입니다. `addrconf` 인수는 자동으로 생성된 IPv6 주소를 나타냅니다.

`interface` 및 `addrobj` 옵션에 대한 자세한 설명은 이전 정적 주소 생성 설명을 참조하십시오.

#### 4. (옵션) 새로 구성된 IP 인터페이스에 대한 정보를 표시합니다.

확인하려는 정보에 따라 다음 명령을 사용할 수 있습니다.

```
# ipadm interface
```

하위 명령을 지정하지 않을 경우 시스템의 모든 인터페이스에 대한 정보가 표시됩니다.

```
# ipadm show-if interface
```

*interface*를 지정하지 않을 경우 시스템의 모든 인터페이스에 대한 정보가 표시됩니다.

```
# ipadm show-addr interface|addrobj
```

*interface* 또는 *addrobj*를 지정하지 않을 경우 모든 주소 객체에 대한 정보가 표시됩니다.

`ipadm show-*` 하위 명령의 출력 결과에 대한 자세한 내용은 “[IP 인터페이스 및 주소 모니터링](#)” [69]을 참조하십시오.

5. 호스트 이름을 사용하는 정적 IP 주소를 구성하는 경우 `/etc/hosts` 파일에 IP 주소에 대한 항목을 추가합니다.

이 파일의 항목은 IP 주소와 해당 호스트 이름으로 구성됩니다.

---

참고 - DHCP 주소를 구성하는 경우 `/etc/hosts` 파일을 업데이트할 필요가 없습니다.

---

6. 기본 경로를 정의합니다.

```
# route -p add default address
```

`netstat -r` 명령으로 경로 지정 테이블의 내용을 확인할 수 있습니다.

경로 관리에 대한 자세한 내용은 [route\(1M\)](#) 및 “[지속\(정적\) 경로 만들기](#)” [54]를 참조하십시오.

### 예 3-1 정적 IP 주소를 사용하여 IPv4 인터페이스 구성

다음 예에서는 정적 IP 주소를 사용하여 인터페이스를 구성하는 방법에 대해 설명합니다. 예에서는 `dladm` 및 `ipadm` 명령이 반응적 NCP를 수정하지 않도록 시스템에서 `DefaultFixed` NCP를 사용으로 설정하는 작업부터 시작합니다. 이로 인해 사용자 환경에 따라 수행하는 수동 네트워크 구성이 무효화될 수 있습니다.

```
# netadm enable -p ncp DefaultFixed
```

```
# dladm show-phys
```

LINK	MEDIA	STATE	SPEED	DUPLEX	DEVICE
net3	Ethernet	up	100Mb	full	bge3

```
# dladm show-link
```

LINK	CLASS	MTU	STATE	OVER	---
net3	phys	1500	up	--	--

```
# ipadm create-ip net3
```

```
# ipadm create-addr -a 192.168.84.3/24 net3
net3/v4
```

```
# ipadm
```

NAME	CLASS/TYPE	STATE	UNDER	ADDR
lo0	loopback	ok	--	--

```

lo0/v4    static    ok    --    127.0.0.1/8
lo0/v6    static    ok    --    ::1/128
net3      ip         ok    --    --
net3/v4   static    ok    --    192.168.84.3/24

```

```
# vi /etc/hosts
```

```
# Internet host table
# 127.0.0.1    localhost
10.0.0.14    myhost
192.168.84.3 sales1
```

```
# route -p add default 192.168.84.1
```

```
# netstat -r
```

```
Routing Table: IPv4
```

Destination	Gateway	Flags	Ref	Use	Interface
default	192.168.84.1	UG	2	10466	
192.168.84.0	192.168.84.3	U	3	1810	net0
localhost	localhost	UH	2	12	lo0

```
Routing Table: IPv6
```

Destination/Mask	Gateway	Flags	Ref	Use	If
solaris	solaris	UH	2	156	lo0

/etc/hosts 파일에서 sales1이 이미 정의된 경우 다음 주소를 지정할 때 해당 호스트 이름을 사용할 수 있습니다.

```
# ipadm create-addr -a sales1 net3
net3/v4
```

### 예 3-2 DHCP 서버에서 IP 주소를 수신하도록 네트워크 인터페이스 구성

다음 예에서는 IP 인터페이스가 DHCP 서버에서 해당 주소를 수신하도록 구성됩니다. 일반적으로 DHCP는 기본 경로도 설치합니다. 따라서 이 예에는 수동으로 기본 경로를 추가하는 단계가 포함됩니다.

```
# dladm show-phys
```

LINK	MEDIA	STATE	SPEED	DUPLEX	DEVICE
net3	Ethernet	up	100Mb	full	bge3

```
# dladm show-link
```

LINK	CLASS	MTU	STATE	OVER
net3	phys	1500	up	--

```
# ipadm create-ip net3
```

```
# ipadm create-addr -T dhcp net3
net3v4
```

```
# ipadm
```

NAME	CLASS/TYPE	STATE	UNDER	ADDR
lo0	loopback	ok	--	--
lo0/v4	static	ok	--	127.0.0.1/8

```
net3      ip          ok      --      --
net3/v4   dhcp         ok      --      10.0.1.13/24
```

## IPv6 인터페이스 구성

네트워크에서 IPv6 주소 지정을 사용하는 초기 단계로 시스템의 IP 인터페이스에서 IPv6을 구성해야 합니다. 설치 프로세스 중 하나 이상의 시스템 인터페이스에서 IPv6을 사용으로 설정할 수 있습니다.

설치 중 IPv6 지원을 사용으로 설정하는 경우 설치가 완료되면 다음과 같은 IPv6 관련 파일 및 테이블이 생성됩니다.

- name-service/switch SMF 서비스는 IPv6 주소를 사용하여 조회가 가능하도록 수정되었습니다.
- IPv6 주소 선택 정책 테이블이 생성됩니다. 이 테이블은 IPv6 지원 인터페이스를 통한 전송에 사용할 IP 주소 형식의 우선 순위를 정합니다.

### ▼ IPv6에 대해 시스템을 구성하는 방법

다음 절차는 Oracle Solaris 설치 후 추가된 인터페이스에 대해 IPv6을 사용으로 설정하는 방법에 대해 설명합니다. IPv6 노드로 사용될 모든 시스템의 인터페이스에서 IPv6을 사용으로 설정하여 IPv6 구성 프로세스를 시작합니다. 일반적인 IPv6 배치에서는 자동 구성을 사용하여 IP 인터페이스를 구성합니다. autoconf IP 주소는 링크 로컬 주소를 지정하며 서브넷에서 사용되고 있는 접두어 및 라우터를 검색합니다. 그런 다음 IPv6 네트워크의 기능을 기반으로 노드의 구성을 호스트, 서버 또는 라우터로 조정할 수 있습니다. autoconf에 대해 설정되는 인터페이스는 DHCPv6 주소 정보도 자동으로 요청합니다. 자동 구성 또는 DHCPv6 없이 정적 IPv6 주소만 사용으로 설정하려면 ipadm 명령에 적합한 옵션을 사용하여 동적으로 지정된 다른 주소를 추가하지 않은 상태로 인터페이스에 링크 로컬 주소를 만듭니다. 예는 [“IPv4 네트워크에서 IPv6 네트워크로 마이그레이션” \[52\]](#)을 참조하십시오.

참고 - 인터페이스가 현재 IPv6 접두어를 알리는 라우터와 동일한 링크에 있는 경우, 자동 구성된 주소의 일부로 해당 사이트의 접두어를 얻습니다. 자세한 내용은 [“Oracle Solaris 11.2 시스템을 라우터 또는 로드 밸런서로 구성”](#)의 [“IPv6 지원 라우터를 구성하는 방법”](#)을 참조하십시오.

1. (옵션) ipadm create-ip 명령에 적합한 옵션을 사용하여 IP 인터페이스를 구성합니다.

```
# ipadm create-ip interface
```

예를 들어, 다음과 같이 net0에 대한 IP 인터페이스를 구성합니다.

```
# ipadm create-ip net0
```

인터페이스가 IPv4와 함께 사용하도록 이미 구성된 경우 이 단계가 필요하지 않습니다. IP 인터페이스 구성에 대한 일반 지침은 [IPv4 인터페이스를 구성하는 방법](#)을 참조하십시오.

## 2. IP 주소를 지정합니다.

**참고** - IP 주소를 지정할 경우 올바른 옵션을 사용하여 IPv6 주소를 지정해야 합니다.

```
# ipadm create-addr -T addrconf interface
```

주소를 더 추가하려면 다음 구문을 사용합니다.

```
# ipadm create-addr -a ipv6-address interface
```

## 3. (옵션) 정적 IPv6 기본 경로를 만듭니다.

```
# /usr/sbin/route -p add -inet6 default ipv6-address
```

**참고** - 자동 구성의 일부로 `in.ndpd`는 검색된 대로 기본 경로를 추가합니다. 이로 인해 수동으로 구성된 기본 경로를 비롯하여 여러 기본 경로가 사용 가능하도록 설정될 수 있습니다. 시스템에서는 사용 가능한 모든 경로에 따라 자동으로 기본 경로를 선택합니다. 따라서 수동으로 구성된 기본 경로가 항상 사용되는 것은 아닙니다.

## 4. (옵션) 노드의 인터페이스 변수에 대한 매개변수를 정의하는 `/etc/inet/ndpd.conf` 파일을 만듭니다.

호스트의 인터페이스에 대해 임시 주소를 만들어야 하는 경우 [“IPv6 인터페이스에 대해 임시 주소 사용” \[47\]](#)을 참조하십시오. `/etc/inet/ndpd.conf`에 대한 자세한 내용은 [ndpd.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오.

## 5. (옵션) 다음과 같이 IP 인터페이스의 상태와 해당 IPv6 구성을 표시합니다.

```
# ipadm show-addr
```

### 예 3-3 설치 후 IPv6 인터페이스 사용

다음 예에서는 `net0` 인터페이스에서 IPv6을 사용으로 설정하는 방법을 보여줍니다. 시작하기 전에 시스템에 구성된 모든 인터페이스의 상태를 확인합니다.

```
# ipadm show-addr
ADDROBJ  TYPE      STATE  ADDR
lo0/v4   static   ok     127.0.0.1/8
net0/v4  static   ok     172.16.27.74/24
```

이전 출력에서와 같이 현재 `net0` 인터페이스만 이 시스템에 대해 구성되어 있습니다. `net0` 인터페이스가 아직 구성되지 않은 경우 `ipadm create-ip net0` 명령을 사용하여 인터페이스를 실행합니다.

그러면 다음과 같이 IPv6이 이 인터페이스에 대해 사용으로 설정됩니다.

```
# ipadm create-addr -T addrconf net0
# ipadm create-addr -a 2001:db8:3c4d:15::203/64 net0

# ipadm show-addr
ADDROBJ      TYPE      STATE  ADDR
lo0/v4       static    ok     127.0.0.1/8
net0/v4       static    ok     172.16.27.74/24
net0/v6       addrconf  ok     fe80::203:baff:fe13:14e1/10
lo0/v6       static    ok     ::1/128
net0/v6a     static    ok     2001:db8:3c4d:15::203/64

# route -p add -inet6 default fe80::203:baff:fe13:14e1
```

- 다음 순서
- IPv6 노드를 라우터로 구성하는 방법에 대한 자세한 내용은 “Oracle Solaris 11.2 시스템을 라우터 또는 로드 밸런서로 구성”의 “IPv6 라우터 구성”을 참조하십시오.
  - 노드를 서버로 조정하는 방법에 대한 자세한 내용은 “서버에서 IPv6 사용 인터페이스 구성” [51]을 참조하십시오.

## IPv6 인터페이스에 대해 임시 주소 사용

IPv6 임시 주소에는 인터페이스의 MAC 주소 대신 무작위로 생성된 64비트 숫자가 인터페이스 ID로 포함됩니다. 익명으로 유지하려는 IPv6 노드의 인터페이스에 대해 임시 주소를 사용할 수 있습니다. 예를 들어 공개 웹 서버에 액세스해야 하는 호스트의 인터페이스에 대해 임시 주소를 사용할 수 있습니다. 임시 주소는 IPv6 프라이버시의 향상된 기능을 구현합니다. 이러한 향상된 기능은 “Privacy Extensions for Stateless Address Autoconfiguration in IPv6” (<http://www.rfc-editor.org/rfc/rfc3041.txt>)에서 제공하는 RFC 3041에 설명되어 있습니다.

필요한 경우 `/etc/inet/ndpd.conf` 파일에서 하나 이상의 인터페이스에 대해 임시 주소를 사용으로 설정할 수 있습니다. 그러나 자동 구성된 표준 IPv6 주소와 달리, 임시 주소는 64비트 서브넷 접두어와 무작위로 작성된 64비트 숫자로 구성됩니다. 이 무작위 숫자가 IPv6 주소의 인터페이스 ID 세그먼트가 됩니다. 임시 주소를 사용할 경우 링크 로컬 주소가 인터페이스 ID로 생성되지 않습니다.

임시 주소에는 기본 선호 수명(1일)이 지정됩니다. 임시 주소 생성을 사용으로 설정하는 경우 `/etc/inet/ndpd.conf` 파일에서 다음 변수를 구성할 수도 있습니다.

<i>valid lifetime</i> TmpValidLifetime	호스트에서 주소가 삭제된 후 임시 주소가 존재하는 시간 범위입니다.
<i>preferred lifetime</i> TmpPreferredLifetime	임시 주소가 제거되기 전의 경과 시간입니다. 이 시간 범위는 유효 수명보다 짧아야 합니다.
<i>address regeneration</i>	선호 수명이 만료되기 이전 기간으로, 이 기간 동안 호스트에서 임시 주소를 새로 생성해야 합니다.

임시 주소의 기간은 다음과 같이 표시됩니다.

<i>n</i>	<i>n</i> 은 초 수입니다(기본값).
<i>n h</i>	<i>n</i> 은 시간(h) 수입니다.
<i>n d</i>	<i>n</i> 은 일(d) 수입니다.

## ▼ 임시 IPv6 주소를 구성하는 방법

- 필요한 경우 호스트의 인터페이스에서 IPv6을 사용으로 설정합니다.  
IPv6에 대해 시스템을 구성하는 방법 [45]을 참조하십시오.
- `/etc/inet/ndpd.conf` 파일을 편집하여 임시 주소 생성을 설정합니다.
  - 호스트의 모든 인터페이스에서 임시 주소를 구성하려면 `/etc/inet/ndpd.conf` 파일에 다음 행을 추가합니다.
 

```
ifdefault TmpAddrsEnabled true
```
  - 특정 인터페이스에 대해 임시 주소를 구성하려면 `/etc/inet/ndpd.conf` 파일에 다음 행을 추가합니다.
 

```
if interface TmpAddrsEnabled true
```
- (옵션) 임시 주소의 유효 수명을 지정합니다.
 

```
ifdefault TmpValidLifetime duration
```

이 구문은 호스트에 있는 모든 인터페이스의 유효 수명을 지정합니다. *duration*의 값은 초, 시간 또는 일 단위여야 합니다. 기본 유효 수명은 7일입니다. `TmpValidLifetime`을 `if interface` 키워드와 함께 사용하여 특정 인터페이스의 임시 주소에 대한 유효 수명을 지정할 수도 있습니다.
- (옵션) 임시 주소의 선호 수명을 지정합니다. 이 기간이 경과하면 주소가 제거됩니다.
 

```
if interface TmpPreferredLifetime duration
```

이 구문은 특정 인터페이스의 임시 주소에 대한 선호 수명을 지정합니다. 기본 선호 수명은 1일입니다. `TmpPreferredLifetime`을 `ifdefault` 키워드와 함께 사용하여 호스트의 모든 인터페이스에서 임시 주소에 대한 선호 수명을 지정할 수도 있습니다.

---

참고 - 기본 주소 선택은 제거된 IPv6 주소에 낮은 우선 순위를 지정합니다. IPv6 임시 주소가 제거된 경우 기본 주소 선택은 사용 가능한 주소를 패킷의 소스 주소로 선택합니다. 사용 가능한 주소는 자동으로 생성된 IPv6 주소 또는 인터페이스의 IPv4 주소일 수 있습니다. 기본 주소 선택에 대한 자세한 내용은 “Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”의 “기본 주소 선택 관리”를 참조하십시오.

---
- (옵션) 주소가 제거되기 전에 제공되는 선행 시간을 지정합니다. 이 시간 동안 호스트에서 임시 주소를 새로 생성해야 합니다.

```
ifdefault TmpRegenAdvance duration
```

이 구문은 호스트에 있는 모든 인터페이스의 임시 주소가 제거되기 전에 제공되는 선행 시간을 지정합니다. 기본값은 5초입니다.

6. 다음과 같이 `in.ndpd` 데몬의 구성을 변경합니다.

```
# pkill -HUP in.ndpd
# /usr/lib/inet/in.ndpd
```

7. `ipadm show-addr` 명령을 실행하여 임시 주소가 만들어졌는지 확인합니다. 예 3-4. “임시 주소가 사용으로 설정된 `ipadm show-addr` 명령 출력 표시”를 참조하십시오.

명령 출력에서 임시 주소의 CURRENT 필드에 t 플래그가 표시됩니다.

예 3-4 임시 주소가 사용으로 설정된 `ipadm show-addr` 명령 출력 표시

다음 예에서는 임시 주소가 만들어진 후 `ipadm show-addr` 명령의 출력을 보여줍니다. IPv6 관련 정보만 샘플 출력에 포함되어 있습니다.

```
# ipadm show-addr -o all
ADDROBJ  TYPE      STATE CURRENT PERSISTENT ADDR
lo0/v6   static   ok    U----   ---      ::1/128
net0/v6  addrconf ok    U----   ---      fe80::a00:20ff:feb9:4c54/10
net0/v6a static   ok    U----   ---      2001:db8:3c4d:15:a00:20ff:feb9:4c54/64
net0/?   addrconf ok    U--t-   ---      2001:db8:3c4d:15:7c37:e7d1:fc9c:d2cb/64
```

주소 객체 `net0/?`의 경우 t 플래그가 CURRENT 필드 아래에 설정되어 해당하는 주소에 임시 인터페이스 ID가 있음을 나타냅니다.

- 참조
- IPv6 주소에 대한 이름 서비스 지원을 설정하려면 4장. [Oracle Solaris 클라이언트에서 이름 지정 및 디렉토리 서비스 관리를 참조하십시오.](#)
  - 서버에 대해 IPv6 주소를 구성하려면 [사용자 지정 IPv6 토큰을 구성하는 방법 \[50\]](#)을 참조하십시오.
  - IPv6 노드의 작업을 모니터하려면 “[Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리](#)”의 1 장, “[TCP/IP 네트워크 관리](#)”를 참조하십시오.

## IPv6 토큰 구성

IPv6 주소의 64비트 인터페이스 ID를 토큰이라고도 합니다. 주소 자동 구성 중 토큰은 인터페이스의 MAC 주소와 연관됩니다. 대부분의 경우 비경로 지정 노드(IPv6 호스트와 서버)는 자동 구성된 토큰을 사용해야 합니다.

그러나 시스템 유지 관리의 일부로 인터페이스가 자주 교체되는 서버의 경우 자동 구성된 토큰을 사용하면 문제가 발생할 수 있습니다. 인터페이스 카드가 변경되면 MAC 주소도 변경됩니다. 그 결과 정적 IP 주소에 의존하는 서버에서 문제가 발생할 수 있습니다. DNS(Domain Name System) 또는 NIS(Network Information System)와 같은 다양한 네

트위크 기반구조 부분이 서버의 인터페이스에 대해 특정 IPv6 주소를 저장했을 수도 있습니다.

주소 변경 문제를 방지하려면 IPv6 주소에서 인터페이스 ID로 사용할 토큰을 수동으로 구성하면 됩니다. 토큰을 만들려면 IPv6 주소의 인터페이스 ID 부분을 차지할 64비트 이하의 16 진수를 지정하십시오. 이후 주소 자동 구성 중 Neighbor Discovery는 인터페이스의 MAC 주소를 기반으로 하는 인터페이스 ID를 만들지 않습니다. 대신 수동으로 생성된 토큰이 인터페이스 ID가 됩니다. 이 토큰은 카드가 교체된 후에도 계속 인터페이스에 지정되어 있습니다.

---

참고 - 사용자 지정 토큰과 임시 주소의 차이점은 임시 주소는 사용자가 명시적으로 만드는 것이 아니라 무작위로 생성된다는 점입니다.

---

## ▼ 사용자 지정 IPv6 토큰을 구성하는 방법

다음 절차는 인터페이스가 자주 교체되는 서버에 특히 유용합니다. IPv6 노드에서 사용자 지정 토큰을 구성하려는 경우에도 다음 단계를 수행할 수 있습니다.

1. 토큰을 사용하여 구성할 인터페이스가 존재하며 해당 인터페이스에 IPv6 주소가 구성되지 않았는지 확인합니다.

```
# ipadm show-if
IFNAME CLASS STATE ACTIVE OVER
lo0 loopback ok yes ---
net0 ip ok yes ---

# ipadm show-addr
ADDROBJ TYPE STATE ADDR
lo0/v4 static ok 127.0.0.1/8
```

이전 출력은 네트워크 인터페이스 net0이 구성된 IPv6 주소 없이 존재함을 보여줍니다.

2. 다음 형식을 따르는 노드 인터페이스에 대한 토큰으로 사용할 64비트 16진수를 하나 이상 만듭니다.

```
XXXX:XXXX:XXXX:XXXX
```

3. 사용자 지정 인터페이스 ID(토큰)를 사용할 각 인터페이스를 구성합니다.

```
# ipadm create-addr -T addrconf -i interface-ID interface
```

예를 들어, 다음과 같이 토큰을 사용하여 인터페이스 net0을 구성합니다.

```
# ipadm create-addr -T addrconf -i ::1a:2b:3c:4d/64 net0
```

---

참고 - 토큰을 사용하여 주소 객체가 생성되면 더 이상 토큰을 수정할 수 없습니다.

---

4. 변경 사항으로 IPv6 데몬을 업데이트합니다.

```
# pkill -HUP in.ndpd
```

**예 3-5 IPv6 인터페이스에서 사용자 지정 토큰 구성**

다음 예에서는 IPv6 주소 및 토큰을 사용하여 net0을 구성하는 방법을 보여줍니다.

```
# ipadm show-if
IFNAME CLASS STATE ACTIVE OVER
lo0 loopback ok yes ---
net0 ip ok yes ---

# ipadm show-addr
ADDROBJ TYPE STATE ADDR
lo0/v4 static ok 127.0.0.1/8

# ipadm create-addr -T addrconf -i ::1a:2b:3c:4d/64 net0
# pkill -HUP in.ndpd
# ipadm show-addr
ADDROBJ TYPE STATE ADDR
lo0/v6 static ok ::1/128
net0/v6 addrconf ok fe80::1a:2b:3c:4d/10
net0/v6a addrconf ok 2002:a08:39f0:1:1a:2b:3c:4d/64
```

토큰이 구성되면 주소 객체 net0/v6에 링크 로컬 주소와 인터페이스 ID에 대해 구성된 1a:2b:3c:4d 주소가 생깁니다. net0/v6이 만들어진 후에는 더 이상 이 인터페이스에 대해 토큰을 수정할 수 없습니다.

- 참조**
- 서버의 IPv6 주소로 이름 서비스를 업데이트하려면 [4장. Oracle Solaris 클라이언트에서 이름 지정 및 디렉토리 서비스 관리](#)를 참조하십시오.
  - 서버 성능을 모니터링하려면 “[Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리](#)”의 1 장, “[TCP/IP 네트워크 관리](#)”를 참조하십시오.

## 서버에서 IPv6 사용 인터페이스 구성

서버 인터페이스에서 IPv6을 사용으로 설정했으므로 서버에서 IPv6 주소를 계획할 때는 몇 가지 사항을 결정해야 합니다. 이러한 결정 사항은 인터페이스 IPv6 주소의 인터페이스 ID(토큰)를 구성하는 데 사용할 전략에 영향을 끼칩니다.

### ▼ 서버 인터페이스에서 IPv6을 사용으로 설정하는 방법

다음 절차에서는 네트워크의 서버에서 IPv6을 사용으로 설정하는 방법에 대해 설명합니다. IPv6 구현 방식에 따라 몇 가지 단계는 다를 수 있습니다.

1. **서버의 IP 인터페이스에서 IPv6을 사용으로 설정합니다.**  
단계별 지침은 “[IPv6 인터페이스 구성](#)” [45]을 참조하십시오.

2. 서버와 동일한 링크에 있는 라우터에서 IPv6 서브넷 접두어가 구성되었는지 확인합니다. 자세한 내용은 [“Oracle Solaris 11.2 시스템을 라우터 또는 로드 밸런서로 구성”의 “IPv6 라우터 구성”](#)을 참조하십시오.
3. 서버의 IPv6 사용 인터페이스에 인터페이스 ID를 지정하기 위한 다음 전략 중 하나를 선택합니다.
 

기본적으로 IPv6 주소 자동 구성은 IPv6 주소의 인터페이스 ID 부분을 만들 때 인터페이스의 MAC 주소를 사용합니다. 인터페이스의 IPv6 주소가 잘 알려진 주소일 경우 한 인터페이스를 다른 인터페이스로 교체하면 문제가 발생할 수 있습니다. 새 인터페이스의 MAC 주소는 다릅니다. 주소 자동 구성 중 토큰은 새 인터페이스 ID가 생성됩니다.

  - 바꾸지 않을 IPv6 사용 인터페이스의 경우 자동 구성된 IPv6 주소를 사용합니다.
  - 로컬 네트워크 외부에 익명으로 표시되어야 하는 IPv6 사용 인터페이스의 경우 무작위로 생성된 토큰을 인터페이스 ID로 사용하는 것이 좋습니다. 자세한 내용은 [임시 IPv6 주소를 구성하는 방법 \[48\]](#)을 참조하십시오.
  - 정기적으로 교체하려는 IPv6 사용 인터페이스의 경우 정적 구성을 사용할 수도 있고 인터페이스 ID에 대한 토큰을 만들 수도 있습니다. 자세한 내용은 [사용자 지정 IPv6 토큰을 구성하는 방법 \[50\]](#)을 참조하십시오.

## IPv4 네트워크에서 IPv6 네트워크로 마이그레이션

참고 - IPv4 네트워크에서 IPv6 네트워크로 마이그레이션하기 전에 대부분의 마이그레이션 계획에는 오랫동안(가능한 경우 무기한) IPv4와 IPv6을 함께 실행하는 작업이 수반됨을 인지해야 합니다.

IPv4 네트워크에서 IPv6 네트워크로 마이그레이션하기 전에 [“Oracle Solaris 11.2의 네트워크 배치 계획”의 2 장, “IPv6 주소 사용 계획”](#)의 정보를 검토하여 추가 작업을 수행해야 할지 여부를 결정하십시오.

IPv4 네트워크에서 IPv6 네트워크로 마이그레이션하는 기본적인 단계를 수행할 때는 먼저 기존 IPv4 DHCP 및 정적 IP 주소를 모두 제거한 다음 필요에 따라 여러 개의 새 IPv6 주소를 재구성하는 작업을 수행합니다. 새 IPv6 인터페이스가 현재 IPv6 접두어를 알리는 라우터와 동일한 링크에 있는 경우 인터페이스가 링크 접두어를 얻습니다. 자세한 내용은 [“Oracle Solaris 11.2 시스템을 라우터 또는 로드 밸런서로 구성”의 “IPv6 라우터 구성”](#)을 참조하십시오.

예 3-6 IPv6 주소로 IPv4 주소 마이그레이션

다음 예에서는 기존 IPv4 주소를 IPv6 주소로 마이그레이션하는 방법을 보여줍니다. 프로세스는 기존 IPv4 DHCP 및 정적 IP 주소를 모두 삭제하는 작업부터 시작합니다.

```
# ipadm show-addr net0/
ADDROBJ  TYPE    STATE  ADDR
lo0/v4   static  ok     127.0.0.1/8
net0/v4  static  ok     172.16.27.74/24
# ipadm delete-addr net0/v4
```

지침은 [“IP 인터페이스 구성 제거 또는 수정” \[68\]](#)을 참조하십시오.

다음으로 ipadm create addr 명령에 적합한 옵션 및 인수를 사용하여 새 IPv6 주소가 만들어집니다.

예를 들어, 다음과 같이 링크 로컬 및 addrconf IPv6 주소를 만들 수 있습니다.

```
# ipadm create-addr -T addrconf -p stateless=yes,stateful=yes net0/v6a
```

다음과 같이 DHCPv6 및 addrconf 주소 없이 정적 IPv6 주소를 만듭니다.

```
# ipadm create-addr -T addrconf -p stateless=no,stateful=no net0/v6a
# ipadm create-addr -T static -a a::b/64 net0/v6b
```

다음과 같이 정적 IPv6 주소를 만듭니다.

```
# ipadm create-addr -T static -a a::b/64 net0/v6b
```

ipadm show-addr 명령을 사용하여 새 IPv6 구성을 표시합니다.

이 예에 포함되지 않은 추가 IPv6 구성 단계(필수 및 선택 사항)는 [“IPv6 인터페이스 구성” \[45\]](#)을 참조하십시오.

## 경로 지정 구성

이 절은 다음 항목으로 구성되어 있습니다.

- [“경로 지정 테이블 및 경로 지정 유형” \[53\]](#)
- [“지속\(정적\) 경로 만들기” \[54\]](#)
- [“단일 인터페이스 시스템에 대한 경로 지정 사용” \[57\]](#)

## 경로 지정 테이블 및 경로 지정 유형

모든 라우터와 호스트는 경로 지정 테이블을 유지 관리합니다. 예를 들어, 다음 경로 지정 테이블에는 시스템의 로컬 기본 네트워크를 비롯하여 시스템에서 인식한 네트워크의 IP 주소가 나열됩니다. 알려진 각 네트워크에 대한 게이트웨이 시스템의 IP 주소도 나열됩니다. 게이트웨이는 송신 패킷을 수신하여 로컬 네트워크 외부의 한 홉으로 전달할 수 있습니다.

Routing Table: IPv4

Destination	Gateway	Flags	Ref	Use	Interface
default	172.20.1.10	UG	1	532	net0
224.0.0.0	10.0.5.100	U	1	0	net1
10.0.0.0	10.0.5.100	U	1	0	net1
127.0.0.1	127.0.0.1	UH	1	57	lo0

Oracle Solaris 시스템에서는 두 가지 유형(정적 및 동적)의 경로 지정을 구성할 수 있습니다. 단일 시스템에서 경로 지정 유형 중 하나 또는 두 가지 모두를 구성할 수 있습니다. 동적 경로 지정을 구현하는 시스템은 IPv4 네트워크용 RIP(Routing Information Protocol) 및 IPv6 네트워크용 RIPng(RIP next generation)과 같은 경로 지정 프로토콜을 사용하여 네트워크 트래픽을 경로 지정하고 테이블의 경로 지정 정보를 업데이트합니다. 정적 경로 지정을 사용하는 경우 route 명령을 사용하여 수동으로 정보를 유지 관리합니다. 자세한 내용은 [route\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

로컬 네트워크 또는 AS(자율 시스템)에 대한 경로 지정을 구성할 때는 특정 라우터 및 호스트에서 지원할 경로 지정 유형을 고려하십시오. 다음 표에서는 다양한 경로 지정 유형과 각 경로 지정 유형이 최적으로 적용되는 네트워킹 시나리오를 보여줍니다.

경로 지정 유형	최적 사례
정적	작은 규모의 네트워크, 기본 라우터에서 경로를 가져오는 호스트 및 다음 홉에서 하나 또는 두 개의 라우터에 대해서만 인식해야 할 기본 라우터
동적	호스트가 여러 개인 로컬 네트워크의 라우터 및 큰 자율 시스템의 호스트를 포함하여 보다 큰 규모의 인터넷 네트워크. 거의 모든 네트워크의 시스템에 동적 경로 지정을 선택하는 것이 좋습니다.
정적과 동적 결합	정적으로 경로 지정된 네트워크와 동적으로 경로 지정된 네트워크를 연결하는 라우터, 내부 자율 시스템을 외부 네트워크와 연결하는 경계 라우터. 시스템에서 정적 경로 지정과 동적 경로 지정을 결합하여 사용하는 것이 일반적입니다.

“Oracle Solaris 11.2의 네트워크 배치 계획”의 “IPv4 자율 시스템 토폴로지”에서 설명되는 토폴로지는 정적 경로 지정과 동적 경로 지정을 결합합니다.

참고 - 시스템에서는 동일한 대상에 대한 두 경로를 통해 자동으로 로드 균형 조정 또는 페일오버를 수행하지 않습니다. 해당 기능이 필요하다면 IPMP를 사용하십시오. 자세한 내용은 “Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”의 2 장, “IPMP 관리 정보”를 참조하십시오.

## 지속(정적) 경로 만들기

route 명령을 사용하여 수동으로 네트워크 경로 지정 테이블을 조작할 수 있습니다. 재부트 후에도 변경 사항이 지속되도록 하려면 -p 옵션을 사용하십시오. /etc/defaultrouter 파일은 Oracle Solaris 11에서 사용되지 않으므로 이 파일을 통해 더 이상 경로(기본값 또는 기타)를 관리할 수 없습니다. route 명령을 사용하는 것이 시스템 재부트 후에도 수동으로 경로가 지속되도록 할 수 있는 유일한 방법입니다.

**참고** - route 명령은 활성 프로파일에 한해서만 경로를 조작합니다. 활성 프로파일이 변경되면 기본 경로와 기타 모든 경로가 바뀔 수 있습니다. 하지만 시스템에서 항상 동일한 프로파일을 사용하는 경우 문제가 되지 않습니다.

지속적으로 경로를 추가할 때는 추가하는 경로가 지속 구성에 없는지 신중히 확인해야 합니다. 해당 경로가 지속 구성에 이미 있을 경우 지속 경로를 업데이트하지 않은 상태로 네트워크 경로 지정 테이블이 변경될 수 있습니다. 예를 들어, 시스템의 기본 경로가 시스템의 주 인터페이스에 매핑(Oracle Solaris 설치 후 자주 발생함)된다고 가정합니다. 나중에 시스템의 주 인터페이스를 다른 인터페이스로 변경하는 경우 시스템의 기본 경로도 지속적으로 업데이트되어야 합니다. 새 경로를 추가하기 전에 지속 경로 구성을 삭제하는 것이 가장 좋습니다. 자세한 내용은 [“Oracle Solaris 11.2의 네트워크 관리 문제 해결”의 “지속 경로를 추가할 때 문제 해결”](#)을 참조하십시오.

지속 경로를 만들고 표시하는 것과 관련된 추가 정보는 다음과 같습니다.

- 지속적으로 경로를 추가하려면 route 명령에 -p 옵션을 사용합니다.

```
# route -p add default ip-address
```

이 방법을 사용하여 만든 경로에 대해 route -p show 명령을 사용하여 지속 정적 경로를 모두 표시할 수 있습니다.

```
# route -p show
```

- netstat 명령에 다음 옵션을 사용하여 시스템의 현재 활성 경로를 표시합니다.

```
# netstat -rn
```

[netstat\(1M\)](#) 및 [route\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

자세한 내용은 [netstat\(1M\)](#) 및 [route\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

반응적 모드를 사용할 때 기본 경로를 만들고 표시하는 것과 관련된 자세한 내용은 [5장. Oracle Solaris의 프로파일 기반 네트워크 구성 관리 정보](#)를 참조하십시오.

## ▼ 경로 지정 테이블에 정적 경로 지정을 추가하는 방법

1. 일반 사용자 계정으로 경로 지정 테이블의 현재 상태를 확인합니다.

```
% netstat -rn
```

출력은 다음과 유사합니다.

```
Routing Table: IPv4
  Destination      Gateway           Flags Ref    Use    Interface
  -----
192.168.5.125     192.168.5.10     U      1      5879    net0
224.0.0.0         198.168.5.10     U      1        0      net0
default          192.168.5.10     UG     1     91908
```

```

127.0.0.1          127.0.0.1          UH    1      811302    lo0

Routing Table: IPv6
Destination/Mask    Gateway                Flags Ref  Use  If
-----
::1                  ::1                    UH     2      0  lo0
    
```

2. 관리자로 로그인합니다.
3. (옵션) 경로 지정 테이블의 기존 항목을 비웁니다.

```
# route flush
```

4. 지속 경로를 추가합니다.

```
# route -p add -net network-address -gateway gateway-address
```

-p                    시스템 재부트 후에도 지속되는 경로를 만듭니다. 경로가 현재 세션에 대해서만 지속되도록 하려면 -p 옵션을 사용하지 마십시오.

-net *network-address*    경로가 *network-address*에 지정된 주소를 사용하는 네트워크로 이동하도록 지정합니다.

-gateway *gateway-address*    지정된 경로에 대한 게이트웨이 시스템의 IP 주소가 *gateway-address*임을 나타냅니다.

**예 3-7** 경로 지정 테이블에 정적 경로 지정 추가

다음 예에서는 라우터(Router 2)에 정적 경로를 추가하는 방법을 보여줍니다. 정적 경로 지정은 AS의 경계 라우터 10.0.5.150에 필요합니다. 이 특정 설정에 대한 그림은 [그림 3-1](#), “IPv4 라우터가 여러 개인 자율 시스템”을 참조하십시오.

다음과 같이 Router 2의 경로 지정 테이블을 확인합니다.

```

# netstat -rn
Routing Table: IPv4
Destination          Gateway                Flags Ref  Use  Interface
-----
default              172.20.1.10           UG     1    249  ce0
224.0.0.0            172.20.1.10           U      1     0  ce0
10.0.5.0              10.0.5.20             U      1     78  bge0
127.0.0.1            127.0.0.1             UH     1     57  lo0

Routing Table: IPv6
Destination/Mask    Gateway                Flags Ref  Use  If
-----
::1                  ::1                    UH     2      0  lo0
    
```

경로 지정 테이블은 Router 2가 인식하는 두 개의 경로가 있음을 나타냅니다. 기본 경로는 Router 2의 172.20.1.10 인터페이스를 게이트웨이로 사용합니다. 두번째 경로 10.0.5.0은

Router 2에서 실행되고 있는 `in.routed` 데몬을 통해 검색되었습니다. 이 경로에 대한 게이트웨이는 IP 주소가 `10.0.5.20`인 Router 1입니다.

다음과 같이 게이트웨이가 경계 라우터인 `10.0.5.0` 네트워크에 두번째 경로를 추가합니다.

```
# route -p add -net 10.0.5.0/24 -gateway 10.0.5.150
add net 10.0.5.0: gateway 10.0.5.150
```

그러면 경로 지정 테이블에 IP 주소가 `10.0.5.150`인 경계 라우터에 대한 경로가 포함됩니다.

```
# netstat -rn
Routing Table: IPv4
Destination          Gateway              Flags Ref  Use  Interface
-----
default              172.20.1.10         UG     1   249  ce0
224.0.0.0            172.20.1.10         U      1     0  ce0
10.0.5.0             10.0.5.20           U      1     78  bge0
10.0.5.0             10.0.5.150          U      1    375  bge0
127.0.0.1           127.0.0.1           UH     1     57  lo0

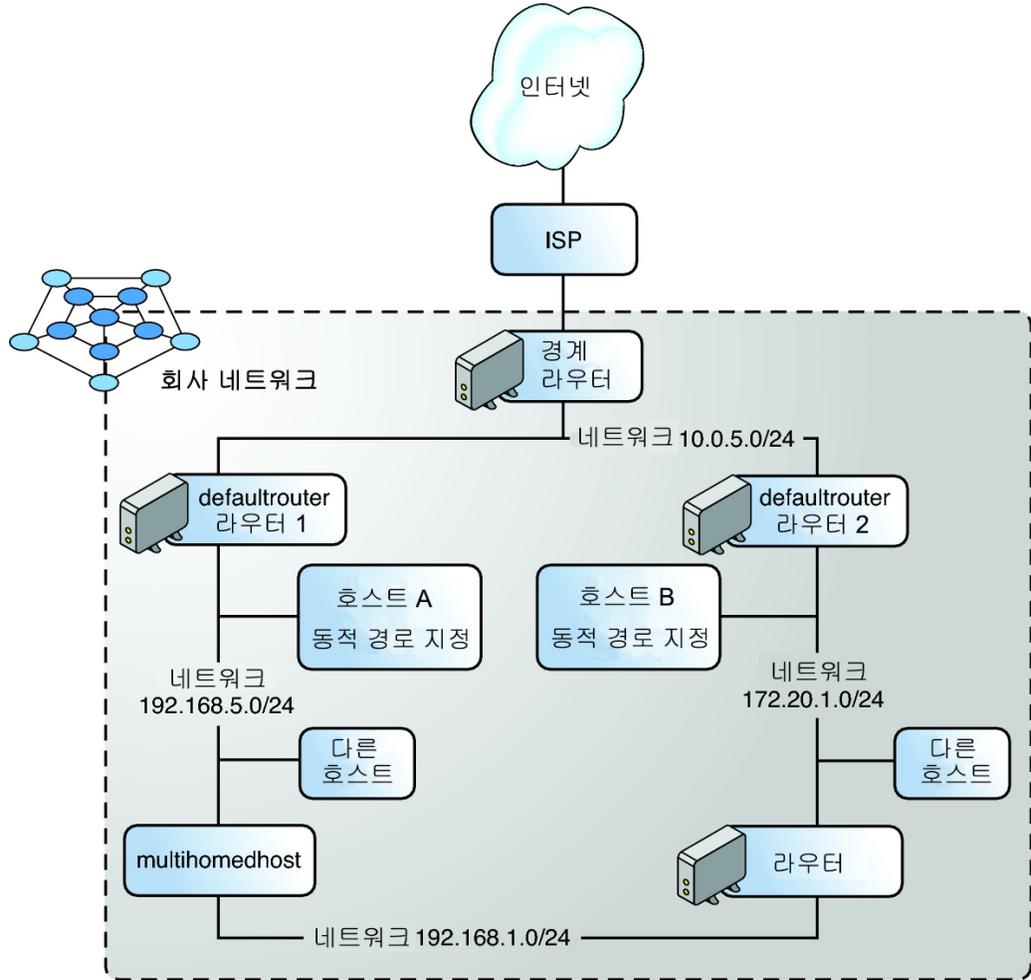
Routing Table: IPv6
Destination/Mask     Gateway              Flags Ref  Use  If
-----
::1                  ::1                  UH     2     0  lo0
```

## 단일 인터페이스 시스템에 대한 경로 지정 사용

정적 또는 동적 경로 지정으로 단일 인터페이스 시스템을 구성할 수 있습니다. 정적 경로 지정을 사용하는 경우 호스트는 경로 지정 정보에 기본 라우터의 서비스를 사용해야 합니다. 경로 지정 프로토콜을 사용하는 동적 경로 지정을 사용으로 설정하면 가장 간편하게 시스템에서 경로 지정을 관리할 수 있습니다.

일반적으로 라우터와 네트워크가 여러 개인 사이트에서는 네트워크 토폴로지를 단일 경로 지정 도메인 또는 AS(자율 시스템)로 관리합니다. 이 절의 절차 및 예는 다음 그림을 기반으로 합니다. 이 그림에서 AS는 세 개의 로컬 네트워크(`10.0.5.0`, `172.20.1.0` 및 `192.168.5.0`)로 구분되어 있습니다. 네트워크는 여러 유형의 라우터(경계 라우터, 기본 라우터, 패킷 전달 라우터)와 클라이언트 시스템으로 구성됩니다. 클라이언트 시스템에는 멀티홈 시스템과 단일 인터페이스 시스템이 포함됩니다. 해당 구성 요소 각각에 대한 자세한 내용은 [“Oracle Solaris 11.2의 네트워크 배치 계획”](#)의 [“IPv4 자율 시스템 토폴로지”](#)를 참조하십시오.

그림 3-1 IPv4 라우터가 여러 개인 자율 시스템



### ▼ 단일 인터페이스 시스템에서 동적 경로 지정을 사용하여 설정하는 방법

다음 절차에서는 시스템의 IP 인터페이스를 이미 구성했다고 가정합니다. 네트워크의 라우터 계획에 대한 자세한 내용은 “Oracle Solaris 11.2의 네트워크 배치 계획”의 “IPv4 자율 시스템 토폴로지”를 참조하십시오.

1. 관리자로 로그인합니다.

2. 시스템이 속한 네트워크에 대한 IP 주소로 시스템의 IP 인터페이스 중 하나를 구성합니다.  
지침은 [IPv4 인터페이스를 구성하는 방법 \[40\]](#)을 참조하십시오.
3. 정의된 지속 경로를 시스템에서 모두 삭제합니다.  
정적으로 정의된 기본 경로가 있을 경우 시스템에서 시스템 부트 중 동적 경로 지정을 사용으로 설정하기 못하므로 이 단계를 수행하는 것입니다.
  - a. 다음과 같이 정의된 지속 기본 경로를 모두 확인합니다.  

```
# route -p show
```
  - b. 정의된 지속 경로를 개별적으로 삭제합니다. 예를 들면 다음과 같습니다.  

```
# route -p delete -net default -gateway 172.20.1.10
```
4. 패킷 전달이 사용 안함으로 설정되었는지 확인합니다.  

```
# routeadm -d ipv4-forwarding -u
```
5. 시스템에서 IPv4 경로 지정을 사용으로 설정합니다.  

```
# routeadm -e ipv4-routing -u
```

예 3-8 단일 인터페이스 시스템에서 동적 경로 지정 실행

다음 예에서는 [그림 3-1](#), “IPv4 라우터가 여러 개인 자율 시스템”에 표시된 네트워크 192.168.5.0에서 단일 인터페이스 시스템 hosta에 대한 동적 경로 지정을 구성하는 방법을 보여줍니다. 시스템에서는 Router 1을 기본 라우터로 사용합니다. 이 예에서는 시스템의 IP 인터페이스를 이미 구성한 것으로 간주합니다.

먼저 관리자 권한으로 hosta에 로그인합니다. 그런 다음 정의된 지속 경로를 시스템에서 모두 제거합니다.

```
# route -p show
persistent: route add default 172.20.1.10

# route -p delete default 172.20.1.10
delete net default: gateway 172.20.1.10
delete persistent net default: gateway 172.20.1.10

# routeadm
```

Configuration Option	Current Configuration	Current System State
IPv4 routing	disabled	disabled
IPv6 routing	disabled	disabled
IPv4 forwarding	disabled	disabled
IPv6 forwarding	disabled	disabled

```

Routing services "route:default ripng:default"

Routing daemons:

STATE FMRI
disabled svc:/network/routing/ripng:default
online svc:/network/routing/ndp:default
disabled svc:/network/routing/rdisc:default
disabled svc:/network/routing/legacy-routing:ipv4
disabled svc:/network/routing/legacy-routing:ipv6
disabled svc:/network/routing/route:default

# routeadm -d ipv4-forwarding -u
# routeadm -e ipv4-routing -u
# routeadm
Configuration Current Current
Option Configuration System State
-----
IPv4 routing enabled enabled
IPv6 routing disabled disabled
IPv4 forwarding disabled disabled
IPv6 forwarding disabled disabled

Routing services "route:default ripng:default"

Routing daemons:

STATE FMRI
disabled svc:/network/routing/ripng:default
online svc:/network/routing/ndp:default
disabled svc:/network/routing/rdisc:default
disabled svc:/network/routing/legacy-routing:ipv4
disabled svc:/network/routing/legacy-routing:ipv6
online svc:/network/routing/route:default

```

## IPv6 경로 지정 정보

CIDR에 의거하여 IPv6 경로 지정은 IPv4 경로 지정과 거의 동일합니다. 주소가 32비트 IPv4 주소 대신 128비트 IPv6 주소라는 점만 다릅니다. 매우 간단한 확장을 통해 OSPF(Open Shortest Path First), RIP, IDR(P(Inter-domain Routing Protocol), IS-IS(Intermediate System to Intermediate System) 등 모든 IPv4 경로 지정 알고리즘을 사용하여 IPv6 경로를 지정할 수 있습니다.

또한 IPv6에는 다음과 같이 강력한 새로운 경로 지정 기능을 지원하는 단순 경로 지정 확장도 포함되어 있습니다.

- 정책, 성능 및 비용 등을 기준으로 하는 공급자 선택
- 호스트 이동성, 현재 위치로 경로 지정
- 자동 주소 재지정, 새 주소로 경로 지정

새로운 경로 지정 기능은 IPv6 경로 지정 옵션을 사용하는 IPv6 주소의 순서를 만들어 이용할 수 있습니다. IPv6 소스는 경로 지정 옵션을 사용하여 패킷 대상으로 이동하는 중에 방문할 하나 이상의 중간 노드 또는 토폴로지 그룹을 나열할 수 있습니다. 이 기능은 IPv4의 느슨한 소스 및 레코드 경로 옵션과 매우 유사합니다.

주소 시퀀스가 일반적인 방식으로 작동하도록 하려면 대부분의 경우 IPv6 호스트에서 호스트가 수신하는 패킷의 경로를 역순으로 설정해야 합니다. IPv6 인증 헤더를 사용하여 패킷이 성공적으로 인증되어야 합니다. 패킷에 주소 순서가 포함되어 있어야 패킷이 원래 전송자에게 반환됩니다. 이 기술은 IPv6 호스트 구현에서 소스 경로의 처리 및 전환이 강제로 지원되도록 합니다. 소스 경로의 처리 및 전환을 통해 공급자는 새로운 IPv6 기능(예: 공급자 선택 및 확장 주소)을 구현하는 호스트를 사용할 수 있습니다.

## 멀티홈 호스트 구성

Oracle Solaris에서는 인터페이스가 두 개 이상인 시스템을 멀티홈 호스트로 간주합니다. 멀티홈 호스트의 인터페이스는 다른 물리적 네트워크 또는 동일한 물리적 네트워크의 서로 다른 서브넷에 연결됩니다. 멀티홈 호스트를 만드는 단계별 지침은 [멀티홈 호스트를 만드는 방법 \[62\]](#)을 참조하십시오.

여러 인터페이스가 동일한 서브넷에 연결되는 시스템에서는 먼저 인터페이스를 하나의 IPMP 그룹으로 구성해야 합니다. 그렇지 않으면 시스템이 멀티홈 호스트가 될 수 없습니다. IPMP에 대한 자세한 내용은 [“Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”의 2 장, “IPMP 관리 정보”](#)를 참조하십시오.

멀티홈 호스트는 IP 패킷을 전달하지 않지만 경로 지정 프로토콜을 실행하도록 멀티홈 호스트를 구성할 수 있습니다. 일반적으로 다음 유형의 시스템을 멀티홈 호스트로 구성합니다.

- 대규모 사용자 풀에서 파일을 공유하기 위해 NFS 서버, 특히 큰 데이터 센터로 작동하는 서버를 두 개 이상의 네트워크에 연결할 수 있습니다. 이러한 서버는 경로 지정 테이블을 유지 관리할 필요가 없습니다.
- NFS 서버와 마찬가지로 데이터베이스 서버는 대규모 사용자 풀에 리소스를 제공할 네트워크 인터페이스를 여러 개 포함할 수 있습니다.
- 방화벽 게이트웨이는 회사 네트워크와 공용 네트워크(예: 인터넷) 간의 연결을 제공하는 시스템입니다. 관리자는 방화벽을 보안 조치로 설정합니다. 방화벽으로 구성된 호스트는 호스트의 인터페이스에 연결된 네트워크 간에 패킷을 전달하지 않습니다. 단, 이 경우에도 호스트는 권한이 부여된 사용자에게 표준 TCP/IP 서비스(예: ssh)를 제공할 수 있습니다.

---

**참고** - 멀티홈 호스트의 인터페이스에 여러 유형의 방화벽이 있을 경우 의도치 않게 호스트의 패킷이 중단되지 않도록 주의해야 합니다. 이 문제는 특히 Stateful 방화벽에서 발생할 수 있습니다. 한 가지 해결 방법은 Stateless 방화벽을 구성하는 것입니다. 방화벽에 대한 자세한 내용은 [“Oracle Solaris 11.2에서 시스템 및 연결된 장치의 보안”의 “방화벽 시스템”](#) 또는 타사 방화벽 설명서를 참조하십시오.

---

## ▼ 멀티홈 호스트를 만드는 방법

1. 관리자로 로그인합니다.
2. 설치 프로세스의 일부로 구성되지 않은 각 추가 네트워크 인터페이스를 구성합니다. [IPv4 인터페이스를 구성하는 방법 \[40\]](#)을 참조하십시오.
3. 패킷 전달을 사용으로 설정할 경우 이 서비스를 사용 안함으로 설정합니다.

```
# routeadm -p ipv4-forwarding
persistent=enabled default=disabled current=enabled
# routeadm -d ipv4-forwarding -u
# routeadm -p ipv4-forwarding
persistent=disabled default=disabled current=disabled
```

4. (옵션) 멀티홈 호스트에 대한 동적 경로 지정을 설정합니다.

```
# routeadm -e ipv4-routing -u
# routeadm -p ipv4-routing
persistent=enabled default=enabled current=enabled
```

### 예 3-9 멀티홈 호스트 구성

다음 예에서는 “Oracle Solaris 11.2의 네트워크 배치 계획”의 “IPv4 자율 시스템 토폴로지”의 그림에 설명된 대로 멀티홈 호스트를 구성하는 방법을 보여줍니다. 이 예에서는 시스템의 호스트 이름이 `hostc`입니다. 이 호스트에는 두 개의 인터페이스가 있으며 모두 192.168.5.0 네트워크에 연결됩니다.

시작하려면 시스템 인터페이스의 상태를 표시합니다.

```
# dladm show-link
LINK      CLASS      MTU      STATE    BRIDGE    OVER
net0      phys       1500     up       --        --
net1      phys       1500     up       --        --

# ipadm show-addr
ADDROBJ   TYPE      STATE      ADDR
lo0/v4    static    ok         127.0.0.1/8
net0/v4    static    ok         192.168.5.82/24
```

`dladm show-link` 명령이 `hostc`에 두 개의 데이터 링크가 있는 것으로 보고합니다. 하지만 `net0`만 IP 주소로 구성되었습니다. `hostc`를 멀티홈 호스트로 구성하려면 동일한 192.168.5.0 네트워크의 IP 주소로 `net1`을 구성합니다. `net1`의 기본 물리적 NIC가 네트워크에 물리적으로 연결되었는지 확인합니다.

```
# ipadm create-ip net1
# ipadm create-addr static -a 192.168.5.85/24 net1
# ipadm show-addr
ADDROBJ   TYPE      STATE      ADDR
lo0/v4    static    ok         127.0.0.1/8
```

```
net0/v4      static ok      192.168.5.82/24
net1/v4      static ok      192.168.5.85/24
```

그런 후 다음과 같이 net1 인터페이스를 /etc/hosts 파일에 추가합니다.

```
# vi /etc/inet/hosts
127.0.0.1      localhost
192.168.5.82   hostc #primary network interface for host3
192.168.5.85   hostc-2 #second interface
```

다음과 같이 이 서비스가 hostc에서 실행 중인 경우 패킷 전달을 해제합니다.

```
# routeadm -p ipv4-forwarding
persistent=enabled default=disabled current=enabled

# routeadm
Configuration Current Current
Option Configuration System State
-----
IPv4 routing enabled enabled
IPv6 routing disabled disabled
IPv4 forwarding disabled disabled
IPv6 forwarding disabled disabled

Routing services "route:default ripng:default"
```

Routing daemons:

STATE	FMRI
disabled	svc:/network/routing/ripng:default
online	svc:/network/routing/ndp:default
disabled	svc:/network/routing/rdisc:default
disabled	svc:/network/routing/legacy-routing:ipv4
disabled	svc:/network/routing/legacy-routing:ipv6
online	svc:/network/routing/route:default

routeadm 명령이 in.routed 데몬을 통한 동적 경로 지정이 현재 사용으로 설정된 것으로 보고합니다.

## 멀티홉 호스트에 다중 경로 지정 구현

기본적으로 여러 인터페이스가 있는 시스템(멀티홉 호스트라고도 함)은 경로 지정 테이블에서 트래픽 대상까지 가장 긴 일치 경로를 기반으로 네트워크 트래픽의 경로를 지정합니다. 대상까지의 길이가 같은 경로가 여러 개 있을 경우 Oracle Solaris는 ECMP(Equal-Cost Multi-Path) 알고리즘을 적용하여 트래픽을 해당 경로에 분산시킵니다.

이런 방식의 트래픽 분산이 적합하지 않은 경우도 있습니다. 예를 들어, 패킷의 IP 소스 주소와 동일한 서브넷에 없는 멀티홉 호스트의 인터페이스를 통해 IP 패킷이 전송될 수 있습니다.

또한 송신 패킷이 특정 수신 요청(예: ICMP 에코 요청)에 대한 응답인 경우 요청과 응답이 동일한 인터페이스를 순회할 수 없습니다. 이 유형의 트래픽 경로 지정 구성을 비대칭 경로 지정이라고 합니다. ISP(인터넷 서비스 공급자)가 RFC 3704 (<http://www.rfc-editor.org/rfc/bcp/bcp84.txt>)에 설명된 대로 진입 필터링을 구현하는 경우 비대칭 경로 지정 구성으로 인해 ISP가 송신 패킷을 삭제할 수도 있습니다.

RFC 3704는 인터넷에서 DoS(서비스 거부) 공격을 제한하기 위한 것입니다. 이 의도를 준수하려면 네트워크에서 대칭 경로 지정을 구성해야 합니다. IP `hostmodel` 등록 정보를 사용하면 이 요구 사항을 충족시킬 수 있습니다. 이 등록 정보는 멀티홈 호스트를 통해 수신 또는 전송된 IP 패킷의 동작을 제어합니다.

`hostmodel` 등록 정보는 세 가지 가능한 값 중 하나를 가질 수 있습니다.

<code>strong</code>	RFC 1122에 정의된 강력한 ES(엔드 시스템) 모델에 해당합니다. 이 값은 대칭 경로 지정을 구현합니다.
<code>weak</code>	RFC 1122에 정의된 약한 ES 모델에 해당합니다. 이 값에서는 멀티홈 호스트가 비대칭 경로 지정을 사용합니다.
<code>src-priority</code>	기본 경로를 사용하여 패킷 경로 지정을 구성합니다. 경로 지정 테이블에 대상 경로가 여러 개 있는 경우 기본 경로는 송신 패킷의 IP 소스 주소가 구성된 인터페이스를 사용하는 경로입니다. 해당 경로가 없는 경우 송신 패킷은 패킷의 IP 대상에 대한 가장 긴 일치 경로를 사용합니다.

예를 들어, 다음과 같이 멀티홈 호스트에서 IP 패킷의 대칭 경로 지정을 구현합니다.

```
# ipadm set-prop -p hostmodel=strong ipv4
# ipadm set-prop -p hostmodel=strong ipv6
# ipadm show-prop -p hostmodel ip
PROTO PROPERTY PERM CURRENT PERSISTENT DEFAULT POSSIBLE
ipv6 hostmodel rw strong -- weak strong,
src-priority,
weak
ipv4 hostmodel rw strong -- weak strong,
src-priority,
weak
```

## IP 인터페이스 등록 정보 및 주소 사용자 정의

IP 인터페이스 등록 정보 관리에 사용되는 세 가지 `ipadm` 하위 명령이 있습니다.

- `show-ifprop -p property interface` - IP 인터페이스의 등록 정보 및 현재 값을 표시합니다. `-p property` 옵션을 사용하지 않으면 IP 인터페이스의 모든 등록 정보가 나열됩니다. IP 인터페이스를 지정하지 않으면 모든 IP 인터페이스의 등록 정보가 모두 나열됩니다.
- `set-ifprop -p property=value interface` - IP 인터페이스의 등록 정보에 값을 지정합니다.

- `reset-ifprop -p property interface` - 특정 등록 정보를 기본값으로 재설정합니다.

데이터 링크처럼 IP 인터페이스에도 특정 네트워크 환경에 맞게 사용자 정의할 수 있는 등록 정보가 있습니다. 인터페이스마다 IPv4 및 IPv6에 대해 하나씩 두 개의 등록 정보 세트가 존재합니다.

## MTU 등록 정보 설정

MTU 등록 정보를 비롯한 일부 등록 정보는 데이터 링크와 IP 인터페이스에 공통됩니다. 따라서 데이터 링크의 MTU 값과 해당 링크를 통해 구성된 인터페이스의 MTU 값을 다르게 지정할 수 있습니다. 해당 IP 인터페이스를 순회하는 IPv4 및 IPv6 패킷에 적용되는 다른 MTU 값을 지정할 수도 있습니다.

IP 인터페이스에 대한 MTU 등록 정보를 설정할 때는 다음과 같은 중요 사항을 고려하십시오.

- IP 인터페이스의 MTU 설정 값은 데이터 링크의 MTU 설정 값보다 클 수 없습니다. 이러한 경우 `ipadm` 명령이 오류 메시지를 표시합니다.
- IP 인터페이스의 MTU 값이 데이터 링크의 MTU 값과 다른 경우 IP 패킷이 IP 인터페이스의 MTU 값으로 제한됩니다. 예를 들어, 데이터 링크의 MTU 값이 9000바이트이며 IP 인터페이스의 MTU 값이 1500바이트인 경우 IP 패킷이 1500바이트로 제한됩니다. 하지만 기본 계층 2 프로토콜을 사용 중인 다른 계층 3 프로토콜이 패킷을 최대 9000바이트까지 전송할 수 있습니다.

데이터 링크의 MTU 설정이 IP 인터페이스의 MTU 설정에 어떤 방식으로 영향을 끼치는지에 대한 정보를 비롯하여 데이터 링크 등록 정보를 사용자 정의하는 것과 관련된 지침은 “[데이터 링크 등록 정보 사용자 정의](#)” [28]를 참조하십시오.

## 패킷 전달 사용

네트워크에서 호스트는 다른 호스트 시스템으로 전송된 데이터 패킷을 받을 수 있습니다. 수신 로컬 시스템에서 패킷 전달을 사용으로 설정하면 해당 시스템이 데이터 패킷을 대상 호스트로 전달할 수 있습니다. 이 프로세스는 *IP* 전달이라고 하며 기본적으로 Oracle Solaris에서 사용 안함으로 설정되어 있습니다.

패킷 전달은 IP 인터페이스와 TCP/IP 프로토콜 모두에 설정할 수 있는 등록 정보로 관리됩니다. 패킷 전달 방식을 선택하려는 경우 IP 인터페이스에서 패킷 전달을 사용으로 설정할 수 있습니다. 예를 들어, NIC가 여러 개인 시스템에서 일부 NIC는 외부 네트워크에 연결되고 일부 NIC는 사설망에 연결된다고 가정합니다. 따라서 모든 인터페이스가 아니라 일부 인터페이스에서만 패킷 전달을 사용으로 설정합니다.

또한 TCP/IP 프로토콜의 등록 정보를 설정하여 시스템에서 패킷 전달을 전역으로 사용으로 설정할 수도 있습니다. 자세한 내용은 “[Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리](#)”의 “[전역적으로 패킷 전달 사용](#)”을 참조하십시오.

**참고** - IP 인터페이스 또는 프로토콜의 forwarding 등록 정보는 상호 배타적이지 않습니다. 인터페이스 및 프로토콜에 대한 등록 정보를 동시에 설정할 수 있습니다. 예를 들어, 프로토콜에서 전역적으로 패킷 전달을 사용으로 설정한 후 시스템에서 각 IP 인터페이스에 대한 패킷 전달을 사용자 정의할 수 있습니다. 따라서 전역적으로 사용으로 설정되었더라도 시스템에 대해 패킷 전달을 선택적으로 적용할 수 있습니다.

예를 들어, 다음과 같이 IP 인터페이스에서 패킷 전달을 사용으로 설정합니다.

```
# ipadm set-ifprop -p forwarding=on -m protocol-version interface
```

여기서 *protocol-version*은 IPv4 또는 IPv6입니다. IPv4 및 IPv6 패킷에 대해 별도로 명령을 입력해야 합니다.

다음 예에서는 시스템에서 IPv4 패킷 전달만 사용으로 설정하는 방법을 보여줍니다.

```
# ipadm show-ifprop -p forwarding net0
IFNAME  PROPERTY  PROTO  PERM  CURRENT  PERSISTENT  DEFAULT  POSSIBLE
net0    forwarding  ipv4   rw    off      off         off      on,off
net0    forwarding  ipv6   rw    off      --         off      on,off

# ipadm set-ifprop -p forwarding=on -m ipv4 net0
# ipadm show-ifprop net0
IFNAME  PROPERTY  PROTO  PERM  CURRENT  PERSISTENT  DEFAULT  POSSIBLE
...
net0    forwarding  ipv4   rw    on       on         off      on,off
...
```

## IP 주소 등록 정보 사용자 정의

ipadm 명령을 통해 IP 주소 관련 등록 정보를 관리할 수 있습니다.

IP 주소 등록 정보를 사용자 정의하여 다음과 같은 네트워크 구성 매개변수를 관리할 수 있습니다.

- 넷마스크 길이
- IP 주소를 아웃바운드 패킷의 소스 주소로 사용할 수 있는지 여부
- 주소가 전역 또는 비전역 영역에 속하는지 여부
- 주소가 개인 주소인지 여부

IP 주소 등록 정보 관련 작업을 수행할 때는 다음과 같은 ipadm 하위 명령을 사용하십시오.

- show-addrprop -p *property addrobj* - 사용되는 옵션에 따라 주소 등록 정보를 표시합니다.

모든 IP 주소의 등록 정보를 표시하려면 등록 정보 또는 주소 객체를 지정하지 마십시오. 모든 IP 주소에 대해 단일 등록 정보의 값을 표시하려면 해당 등록 정보만 지정합니다. 특정 주소 객체의 모든 등록 정보를 표시하려면 주소 객체만 지정합니다.

- `set-addrprop -p property=value addrobj` - 주소 등록 정보에 값을 지정합니다. 주소 등록 정보는 한 번에 하나만 설정할 수 있습니다.
- `reset-addrprop -p property addrobj` - 주소 등록 정보에 대한 모든 기본값을 복원합니다.

**참고** - 특정 인터페이스의 IP 주소를 변경하려면 `set-addressprop` 하위 명령을 사용하지 마십시오. 대신 주소 객체를 삭제하고 새 IP 주소를 사용하여 새 항목을 만듭니다. [“IP 인터페이스 구성 제거 또는 수정” \[68\]](#)을 참조하십시오.

한 가지 예제로, IP 주소의 넷마스크를 변경한다고 가정해보십시오. IP 주소는 IP 인터페이스 `net3`에 구성되어 있으며 주소 객체 이름 `net3/v4`로 식별됩니다. 다음 예에서는 넷마스크를 수정하는 방법을 보여줍니다.

```
# ipadm show-addr
ADDROBJ    TYPE      STATE     ADDR
lo0/?      static    ok        127.0.0.1/8
net3/v4     static    ok        192.168.84.3/24

# ipadm show-addrprop -p prefixlen net3/v4
ADDROBJ  PROPERTY  PERM  CURRENT  PERSISTENT  DEFAULT  POSSIBLE
net3/v4  prefixlen rw     24       24          24       1-30,32

# ipadm set-addrprop -p prefixlen=8 net3/v4
# ipadm show-addrprop -p prefixlen net3/v4
ADDROBJ  PROPERTY  PERM  CURRENT  PERSISTENT  DEFAULT  POSSIBLE
net3/v4  prefixlen rw     8        24        24        24       1-30,32
```

## IP 인터페이스 구성 사용 안함, 제거 및 수정

이 절은 다음 항목으로 구성되어 있습니다.

- [“IP 인터페이스 구성 제거” \[67\]](#)
- [“IP 인터페이스 구성 사용 안함” \[68\]](#)
- [“IP 인터페이스 구성 제거 또는 수정” \[68\]](#)

## IP 인터페이스 구성 제거

`delete-ip` 하위 명령을 사용하여 구성된 IP 인터페이스를 제거할 수 있습니다. 이 명령은 특정 데이터 링크 구성 작업을 수행할 때 특히 중요합니다. 예를 들어, 데이터 링크 이름 바꾸기는 해당 데이터 링크를 통해 구성된 IP 인터페이스가 있을 경우 실패합니다. 데이터 링크 이름 바꾸기를 시도하기 전에 `ipadm delete-ip` 명령을 사용하여 기존 IP 구성을 제거해야 합니다. 다음 예에서는 이 작업을 수행하는 데 사용하는 명령을 보여줍니다.

```
# ipadm delete-ip interface
```

```
# dladm rename-link old-name new-name
# ipadm create-ip interface
# ipadm create-address parameters
```

추가 정보는 “데이터 링크 이름 바꾸기” [27]를 참조하십시오. 데이터 링크 이름이 바뀐 후 IP 인터페이스를 재구성하려면 [IPv4 인터페이스를 구성하는 방법 \[40\]](#)을 참조하십시오.

## IP 인터페이스 구성 사용 안함

기본적으로 IP 인터페이스는 연결되며 ipadm create-ip 명령을 사용하여 인터페이스를 만들 때 활성 구성에 포함됩니다. 인터페이스에 첫번째 주소가 만들어질 때 인터페이스에 UP 플래그가 지정됩니다.

구성을 삭제하지 않고 활성 구성에서 인터페이스를 제거하려면 다음과 같이 disable-if 하위 명령을 사용하십시오. 이 하위 명령은 커널에서 인터페이스의 연결을 취소합니다.

```
# ipadm disable-if -t interface
```

다음과 같이 IP 인터페이스가 작동하고 플래그가 UP으로 표시되도록 합니다.

```
# ipadm enable-if -t interface
```

---

작은 정보 - IP 인터페이스의 현재 상태를 표시하려면 “[IP 인터페이스에 대한 정보 가져오기](#)” [70]를 참조하십시오.

---

## IP 인터페이스 구성 제거 또는 수정

ipadm delete-addr 명령은 IP 인터페이스에서 특정 주소 구성을 삭제합니다. 이 명령은 시스템에서 IP 주소를 제거하거나 인터페이스에서 구성된 IP 주소를 변경하는 작업의 일부로 IP 주소를 제거하려는 경우 유용합니다. 인터페이스에서 구성된 IP 주소를 변경하려면 먼저 원래 주소 구성을 제거한 후 새 주소 구성을 지정해야 합니다. [기존 IP 주소를 수정하는 방법 \[69\]](#)을 참조하십시오.

인터페이스에 대한 IP 주소를 만드는 지침은 [IPv4 인터페이스를 구성하는 방법 \[40\]](#)을 참조하십시오.

---

참고 - 하나의 인터페이스는 여러 IP 주소를 가질 수 있습니다. 각 주소는 주소 객체로 식별됩니다. 올바른 주소를 제거하는지 확인하려면 주소 객체를 알아야 합니다. ipadm show-addr 명령을 사용하여 인터페이스에 구성된 IP 주소를 표시할 수 있습니다. 주소 객체에 대한 설명은 [IPv4 인터페이스를 구성하는 방법 \[40\]](#)을 참조하십시오. IP 주소 표시에 대한 자세한 내용은 “[IP 주소에 대한 정보 가져오기](#)” [72]를 참조하십시오.

---

## ▼ 기존 IP 주소를 수정하는 방법

다음 절차에서는 시스템의 기존 IP 주소를 재구성하는 단계에 대해 설명합니다.

1. 관리자로 로그인합니다.
2. 재구성할 IP 주소를 나타내는 주소 객체를 삭제합니다.

```
# ipadm delete-addr addrobj
```

3. 동일한 주소 객체 이름을 사용하여 새 IP 주소를 지정합니다.

```
# ipadm create-addr -a IP-address addrobj
```

시스템에 다른 인터페이스를 추가하려면 [IPv4 인터페이스를 구성하는 방법](#)을 참조하십시오.

4. (옵션) 필요한 경우 다음과 같이 시스템의 호스트 이름을 수정합니다.

```
# hostname new-hostname
```

5. (옵션) 서브넷 마스크가 변경된 경우 서브넷 항목을 수정합니다.
6. (옵션) 서브넷 주소가 변경된 경우 기본 라우터의 IP 주소를 변경합니다.  
지침은 ["지속\(정적\) 경로 만들기" \[54\]](#)를 참조하십시오.
7. 변경 사항을 적용하려면 시스템을 재부트합니다.

## IP 인터페이스 및 주소 모니터링

ipadm 명령을 사용하면 IP 인터페이스 및 해당 등록 정보를 모니터링하고 정보를 가져올 수 있습니다. 이 명령을 단독으로 사용하면 시스템에서 IP 인터페이스에 대한 일반 정보가 표시됩니다. 하지만 다음과 같은 명령 구문을 사용하여 표시하려는 정보를 제한하기 위한 다양한 하위 명령을 사용할 수도 있습니다.

```
ipadm show-* other-arguments interface
```

- 인터페이스 정보만 가져오려면 show-if 하위 명령을 사용합니다.
- 주소 정보만 가져오려면 show-addr 하위 명령을 사용합니다.
- 인터페이스 등록 정보에 대한 정보를 가져오려면 show-ifprop 하위 명령을 사용합니다.
- 주소 등록 정보에 대한 정보를 가져오려면 show-addrprop 하위 명령을 사용합니다.

ipadm show-\* 명령으로 표시되는 모든 필드에 대한 설명은 [ipadm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## IP 인터페이스에 대한 일반 정보 가져오기

ipadm 명령은 시스템의 인터페이스에 대한 포괄적인 개요를 제공합니다. 하위 명령을 사용하지 않고 명령을 사용하면 모든 시스템의 IP 인터페이스에 대한 기본 정보가 표시됩니다. 예를 들면 다음과 같습니다.

```
# ipadm
NAME          CLASS/TYPE STATE   UNDER ADDR
lo0           loopback  ok      --    --
lo0/v4        static    ok      --    127.0.0.1/8
lo0/v6        static    ok      --    ::1/128
net0          ip        ok      --    --
net0/v4       static    ok      --    10.132.146.233/23
net0/v4       dhcp     ok      --    10.132.146.234/23
ipmp0         ipmp     degraded --    --
ipmp0/v6      static    ok      --    2001:db8:1:2::4c08/128
net1          ip        failed  ipmp0 --
net1/v6       addrconf ok      --    fe80::124:4fff:fe58:1831/10
net2          ip        ok      ipmp0 --
net2/v6       addrconf ok      --    fe80::214:4fff:fe58:1832/10
iptun0        ip        ok      --    --
iptun0/v4     static    ok      --    172.16.111.5->172.16.223.75
iptun0/v6     static    ok      --    fe80::10:5->fe80::223:75
iptun0/v6a    static    ok      --    2001:db8:1a0:7::10:5->2001:db8:7a82:64::223:75
```

이전 출력은 다음 정보를 표시합니다.

- IP 인터페이스
- 각 인터페이스의 클래스
- 각 인터페이스의 상태
- 인터페이스 상태: "독립형" IP 인터페이스 또는 다른 유형의 인터페이스 구성을 위한 기본 인터페이스. 이 예에서는 net1 및 net2가 UNDER 열에 표시된 것처럼 ipmp0의 기본 인터페이스입니다.
- 인터페이스와 연관된 주소 객체. 주소 객체는 특정 IP 주소를 식별합니다. 이러한 주소 객체는 인터페이스 이름과 구분하기 위해 NAME 제목 아래에 나열되고 들여써집니다.
- CLASS/TYPE 제목 아래에 들여써지고 static, dhcp 등일 수 있는 IP 주소 유형
- ADDRESS 열 아래에 나열된 실제 주소

## IP 인터페이스에 대한 정보 가져오기

IP 인터페이스에 대한 자세한 내용을 보려면 ipadm show-if *interface* 명령을 사용하십시오. 인터페이스를 지정하지 않으면 시스템의 모든 인터페이스에 대한 정보가 제공됩니다.

이 명령 출력의 필드는 다음 정보를 나타냅니다.

IFNAME	정보가 표시되는 인터페이스를 나타냅니다.
CLASS	다음 네 가지 중 하나일 수 있는 인터페이스 클래스를 나타냅니다. <ul style="list-style-type: none"> <li>■ ip는 IP 인터페이스를 나타냅니다.</li> <li>■ ipmp는 IPMP 인터페이스를 나타냅니다.</li> <li>■ vni는 가상 인터페이스를 나타냅니다.</li> <li>■ loopback은 자동으로 생성되는 루프백 인터페이스를 나타냅니다. 루프백 인터페이스를 제외하고 나머지 세 인터페이스 클래스는 수동으로 만들 수 있습니다.</li> </ul>
STATE	ok, offline, failed, down, disabled 중 하나일 수 있는 인터페이스 상태를 나타냅니다. <p>failed 상태는 IPMP 그룹에 적용되며 작동 중지되고 트래픽을 호스트할 수 없는 데이터 링크 또는 IP 인터페이스를 나타낼 수 있습니다. IP 인터페이스가 IPMP 그룹에 속하는 경우 IPMP 인터페이스가 그룹의 다른 활성 IP 인터페이스를 사용하여 계속 트래픽을 받고 보낼 수 있습니다.</p> <p>down 상태는 관리자가 오프라인 상태로 전환한 IP 인터페이스를 나타냅니다.</p> <p>disable 상태는 ipadm disable-if 명령을 사용하여 연결 취소된 IP 인터페이스를 나타냅니다.</p>
ACTIVE	인터페이스가 트래픽을 호스트하는 데 사용되는지 여부를 나타내며 yes 또는 no로 설정됩니다.
OVER	인터페이스의 IPMP 클래스에만 적용되며 IPMP 인터페이스 또는 그룹을 구성하는 기본 인터페이스를 나타냅니다.

다음은 명령이 표시하는 정보의 예입니다.

```
# ipadm show-if
IFNAME      CLASS      STATE      ACTIVE     OVER
lo0         loopback   ok         yes        --
net0        ip         ok         yes        --
net1        ip         ok         yes        --
tun0        ip         ok         yes        --
```

## IP 인터페이스 등록 정보에 대한 정보 가져오기

ipadm show-ifprop *interface* 명령을 사용하여 IP 인터페이스의 등록 정보에 대한 정보를 가져올 수 있습니다. 등록 정보 또는 인터페이스를 지정하지 않으면 시스템에서 모든 IP 인터페이스의 모든 등록 정보에 대한 정보가 표시됩니다.

명령 출력 결과의 필드는 다음을 나타냅니다.

IFNAME	정보가 표시되는 IP 인터페이스를 나타냅니다.
PROPERTY	인터페이스의 등록 정보를 나타냅니다. 한 인터페이스에 등록 정보가 여러 개 있을 수 있습니다.
PROTO	등록 정보가 적용되고 IPv4 또는 IPv6일 수 있는 프로토콜을 나타냅니다.
PERM	지정된 등록 정보에 대해 허용된 권한을 나타내며 읽기 전용, 쓰기 전용 또는 둘 다일 수 있습니다.
CURRENT	활성 구성에서 등록 정보의 현재 값을 나타냅니다.
PERSISTENT	시스템을 재부트할 때 재적용되는 등록 정보의 값을 나타냅니다.
DEFAULT	지정한 등록 정보의 기본값을 나타냅니다.
POSSIBLE	지정한 등록 정보에 지정될 수 있는 값의 목록을 나타냅니다. 숫자 값의 경우 허용되는 값의 범위가 표시됩니다.

**참고** - 정보가 요청되는 등록 정보를 인터페이스가 지원하지 않는 경우와 같이 필드 값을 알 수 없는 경우 값이 물음표(?)로 표시됩니다.

다음 예에서는 `show-ifprop` 하위 명령이 표시하는 정보의 유형을 보여줍니다.

```
# ipadm show-ifprop -p mtu net1
IFNAME PROPERTY PROTO PERM CURRENT PERSISTENT DEFAULT POSSIBLE
net1 mtu ipv4 rw 1500 -- 1500 68-1500
net1 mtu ipv6 rw 1500 -- 1500 1280-1500
```

## IP 주소에 대한 정보 가져오기

IP 주소에 대한 자세한 내용을 보려면 `ipadm show-addr interface` 명령을 사용하십시오. 인터페이스를 지정하지 않으면 시스템에 있는 모든 IP 주소에 대한 정보가 표시됩니다.

명령 출력 결과의 필드는 다음을 나타냅니다.

ADDROBJ	IP 주소가 나열되는 주소 객체를 지정합니다.
TYPE	IP 주소가 <code>static</code> , <code>dhcp</code> 또는 <code>addrconf</code> 인지 나타냅니다. <code>addrconf</code> 값은 Stateless 또는 Stateful 주소 구성을 사용하여 주소를 가져왔음을 나타냅니다.
STATE	활성 구성에서 주소 객체의 상태에 대해 설명합니다. 해당 값의 전체 목록은 <a href="#">ipadm(1M)</a> 매뉴얼 페이지를 참조하십시오.

ADDR 인터페이스에 구성되는 IP 주소를 지정합니다. 주소는 IPv4 또는 IPv6 일 수 있습니다. 터널 인터페이스는 로컬 및 원격 주소를 모두 표시합니다.

터널에 대한 자세한 내용은 “Oracle Solaris 11.2의 TCP/IP 네트워크, IPMP 및 IP 터널 관리”의 5 장, “IP 터널 관리”를 참조하십시오.

다음은 show-addr 하위 명령이 제공하는 정보의 예입니다.

```
# ipadm show-addr
ADDROBJ      TYPE      STATE     ADDR
lo0/v4       static    ok        127.0.0.1/8
net0/v4       static    ok        192.168.84.3/24
tun0/v4       static    ok        172.16.134.1-->172.16.134.2
```

이 명령에 인터페이스를 지정하고 인터페이스에 여러 주소가 포함된 경우 다음과 유사한 정보가 표시됩니다.

```
# ipadm show-addr net0
ADDROBJ      TYPE      STATE     ADDR
net0/v4       static    ok        192.168.84.3/24
net0/v4a      static    ok        10.0.1.1/24
net0/v4bc     static    ok        172.16.10.1
```

*interface/?*로 표시되는 주소 객체는 libipadm API를 사용하지 않는 응용 프로그램이 인터페이스에 주소를 구성했음을 나타냅니다. 해당 응용 프로그램은 ipadm 명령의 제어를 받지 않습니다. 이 명령을 사용하려면 주소 객체 이름이 *interface/user-defined-string* 형식을 사용해야 합니다. IP 주소 지정 예는 IPv4 인터페이스를 구성하는 방법 [40]을 참조하십시오.

## IP 주소 등록 정보에 대한 정보 가져오기

IP 주소 등록 정보에 대한 자세한 내용을 보려면 ipadm show-addrprop *addrobj* 명령을 사용하십시오. 모든 등록 정보를 나열하려면 *addrobj* 옵션을 생략하십시오. 모든 IP 주소에 대한 단일 등록 정보를 나열하려면 해당 등록 정보만 지정하십시오. 특정 주소의 모든 등록 정보를 표시하려면 *addrobj* 옵션만 지정하십시오.

명령 출력 결과의 필드는 다음을 나타냅니다.

ADDROBJ	등록 정보가 나열되는 주소 객체를 나타냅니다.
PROPERTY	주소 객체의 등록 정보를 나타냅니다. 한 주소 객체에 등록 정보가 여러 개 있을 수 있습니다.
PERM	지정된 등록 정보에 대해 허용된 권한을 나타내며 읽기 전용, 쓰기 전용 또는 둘 다일 수 있습니다.
CURRENT	현재 구성에서 등록 정보의 실제 값을 나타냅니다.

- PERSISTENT            시스템을 재부트할 때 재적용되는 등록 정보의 값을 나타냅니다.
- DEFAULT             지정한 등록 정보의 기본값을 나타냅니다.
- POSSIBLE            지정한 등록 정보에 지정될 수 있는 값의 목록을 나타냅니다. 숫자 값의 경우 허용되는 값의 범위가 표시됩니다.

다음은 show-addrprop 하위 명령이 표시하는 정보 유형의 예입니다.

```
# ipadm show-addrprop net1/v4
ADDROBJ  PROPERTY  PERM  CURRENT      PERSISTENT  DEFAULT      POSSIBLE
net1/v4  broadcast r-     192.168.84.255 --          192.168.84.255 --
net1/v4  deprecated rw     off      --          off         on,off
net1/v4  prefixlen rw     24      24          24         1-30,32
net1/v4  private  rw     off      --          off         on,off
net1/v4  transmit rw     on      --          on         on,off
net1/v4  zone     rw     global   --          global      --
```

# ◆◆◆ 4 장 4

## Oracle Solaris 클라이언트에서 이름 지정 및 디렉토리 서비스 관리

---

이 장에서는 Oracle Solaris 호스트 클라이언트 시스템에 대한 이름 지정 서비스를 구성하는 방법을 설명합니다. 이름 지정 및 디렉토리 서비스와 서버측 관리에 대한 전체 개요는 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)를 참조하십시오.

이름 지정 및 디렉토리 서비스 구성 문제 해결에 대한 자세한 내용은 [“Oracle Solaris 11.2의 네트워크 관리 문제 해결”](#)의 3 장, [“이름 지정 서비스 문제 해결”](#)을 참조하십시오.

이 장의 내용:

- [“이름 지정 서비스 구성의 새로운 기능”](#) [75]
- [“이름 지정 및 디렉토리 서비스 구성 개요”](#) [76]
- [“로컬 파일 모드에 대한 시스템 구성”](#) [78]
- [“DNS 클라이언트 구성”](#) [80]
- [“NIS 클라이언트 구성”](#) [82]
- [“LDAP 클라이언트 구성”](#) [84]

---

참고 - 별도의 언급이 없는 한 이 장에서 설명되는 작업은 IPv4 네트워크와 IPv6 네트워크에 모두 적용됩니다.

---

### 이름 지정 서비스 구성의 새로운 기능

새로 추가되거나 변경된 기능은 다음과 같습니다.

- **SMF로 이름 지정 서비스 및 시스템 구성 마이그레이션** - 이 릴리스에서 이름 지정 서비스는 SMF(서비스 관리 기능)를 통해 관리됩니다. 특정 파일(예: `/etc/nsswitch.conf` 및 `/etc/resolv.conf`)을 수정하여 이름 지정 서비스를 구성하던 이전 동작은 더 이상 작동하지 않습니다. 이 Oracle Solaris 릴리스에서는 이전 Oracle Solaris 릴리스와의 호환성을 위해서만 레거시 구성 파일이 보존됩니다. 해당 파일의 콘텐츠는 특정 이름 지정 서비스에 속하는 SMF 서비스를 통해 생성됩니다.

네트워크 구성이 없는 경우 이름 지정 서비스는 기본적으로 `nis files`가 아닌 `files only` 동작으로 설정됩니다. `svc:/system/name-service/cache` SMF 서비스는 항상 사용

으로 설정해야 합니다. 또한 SMF 명령을 사용하여 해당 서비스에 대한 구성을 변경하는 경우 서비스를 사용으로 설정하거나 새로 고치거나 두 작업을 모두 수행해야만 변경 사항이 적용됩니다. [svccfg\(1M\)](#) 및 [svcadm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

- **resolv.conf 오류 검사 기능** - SMF로 이름 지정 서비스를 마이그레이션하기 전에는 resolv.conf 파일 구성에 있던 오류가 자동으로 처리되었으므로 경고가 표시되지 않아 오류가 발견되지 않았습니다. 그 결과 resolv.conf 파일이 구성된 방식에 따라 동작하지 않았습니다. Oracle Solaris 11에서는 오류 상태가 제대로 보고되도록 SMF 템플릿을 통해 기본적인 오류 검사가 수행됩니다. [resolv.conf\(4\)](#) 매뉴얼 페이지를 참조하십시오.
- **DNS(Domain Name System) 서버 설정** - DNS 서버를 설정하는 프로세스가 변경되었습니다. 자세한 지침은 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)의 [“DNS 관리\(작업\)”](#)를 참조하십시오.

## 이름 지정 및 디렉토리 서비스 구성 개요

이름 지정 서비스는 저장된 정보(예: 호스트 이름 및 주소, 사용자 이름, 암호, 액세스 권한 등)를 조회합니다. 이 정보는 사용자가 호스트에 로그인하고, 리소스에 액세스하고, 권한이 부여될 수 있도록 하기 위해 제공됩니다. 이름 지정 서비스 정보는 파일, 맵 또는 다양한 형식의 데이터베이스 파일에 저장할 수 있습니다. 이러한 정보 저장소는 로컬 시스템에 있거나 중앙 네트워크 기반 저장소 또는 데이터베이스에 있을 수 있습니다. 중앙 이름 지정 서비스가 없으면 각 호스트가 이 정보의 고유한 복사본을 유지 관리해야 합니다. 모든 데이터를 중앙에 배치하면 관리가 더 쉬워집니다. 이름 지정 서비스는 모든 컴퓨팅 네트워크의 기본 요소입니다.

지원되는 이름 지정 및 디렉토리 서비스는 다음과 같습니다.

- **DNS(Domain Name System)**

DNS는 TCP/IP 네트워크에 구현된 계층적 분산 데이터베이스입니다. 주로 인터넷 호스트 이름의 IP 주소와 IP 주소의 호스트 이름을 조회하는 데 사용됩니다. 데이터는 네트워크에 분산되고 오른쪽에서 왼쪽으로 읽는 마침표로 구분된 이름을 사용하여 찾습니다. DNS는 메일 교환 경로 지정 정보, 위치 데이터, 사용 가능한 서비스 등 다른 인터넷 관련 호스트 정보를 저장하는 데도 사용됩니다. 서비스의 계층적 특성은 로컬 도메인의 로컬 관리를 가능하게 하는 동시에 인터넷과 인트라넷 중 하나 또는 모두에 연결된 다른 도메인의 국제 서비스 범위를 제공합니다. 자세한 내용은 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)의 [“DNS 이름 지정 서비스에 대한 설명”](#)을 참조하십시오.

DNS 프로토콜에 대한 두 가지 확장은 svc:network/dns/multicast 서비스에서 관리됩니다. mDNS(멀티캐스트 DNS)는 기존 DNS 서버가 설치되지 않은 소규모 네트워크에서 DNS를 구현합니다. DNS-SD(DNS 서비스 검색)는 멀티캐스트 DNS를 확장하여 간단한 서비스 검색(네트워크 검색) 기능도 제공합니다. [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)의 [“멀티캐스트 DNS 및 서비스 검색에 대한 설명”](#)을 참조하십시오.

- **NIS(Network Information System)**

NIS(이 설명서에서는 "niss"로 발음됨)는 DNS와 독립적으로 개발되었습니다. NIS는 다양한 네트워크 정보에 대한 중앙집중 제어를 제공하여 네트워크 관리를 더 용이하게 하는데 주력합니다. NIS는 네트워크, 시스템 이름 및 주소, 사용자, 네트워크 서비스 등에 대한 정보를 저장합니다. 이 네트워크 정보 모음을 *NIS* 이름 공간이라고 합니다. 자세한 내용은 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)의 [“NIS 이름 지정 서비스에 대한 설명”](#)을 참조하십시오.

#### ■ LDAP(Lightweight Directory Access Protocol)

LDAP은 분산 이름 지정 및 기타 디렉토리 서비스를 위해 디렉토리 서버에 액세스하는 데 사용되는 보안 네트워크 프로토콜입니다. 이 표준 기반 프로토콜은 계층적 데이터베이스 구조를 지원합니다. 동일한 프로토콜을 사용하여 UNIX 및 다중 플랫폼 환경에서 이름 지정 서비스를 제공할 수 있습니다. Oracle Solaris는 Oracle Directory Server Enterprise Edition(이전에는 Sun Java System Directory Server라고 함) 및 기타 LDAP 디렉토리 서버와 함께 LDAP을 지원합니다. 자세한 내용은 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)의 [“LDAP 이름 지정 서비스에 대한 설명”](#)을 참조하십시오.

Oracle Solaris에서의 이름 지정 서비스 지원(서버측 및 클라이언트측)에 대한 전체 개요는 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)의 1 장, [“이름 지정 및 디렉토리 서비스 정보”](#) 및 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: LDAP”](#)을 참조하십시오.

## name-service/switch SMF 서비스 정보

name-service/switch SMF 서비스는 네트워크 정보의 각 유형에 사용할 이름 정보 서비스 또는 소스를 지정하는 데 사용할 수 있는 구성 가능한 선택 서비스입니다.

이름 서비스 스위치는 다음 인터페이스를 호출하는 클라이언트 응용 프로그램에서 사용됩니다.

- gethostbyname
- getpwuid
- getpwnam
- getaddrinfo

name-service/switch SMF 서비스는 각 네트워크 데이터베이스에 사용되어야 할 이름 지정 서비스를 정의합니다. 이전에는 이 정보가 `/etc/nsswitch.conf` 파일에 저장되었습니다. 이 파일이 계속 존재하기는 하지만, 해당 SMF 서비스에서 적합한 등록 정보를 변경하여 파일에 포함된 구성 설정을 수정해야 합니다.

해당 등록 정보는 다음과 같이 표시할 수 있습니다.

```
$ svccfg -s name-service/switch listprop config
config                application
config/default        astring              files
config/value_authorization astring              solaris.smf.value.name-service.switch
```

```

config/password      astring      "files ldap"
config/group         astring      "files ldap"
config/host          astring      "files dns"
config/automount     astring      "files ldap"

```

config/default 등록 정보는 검색할 기본 소스를 지정합니다. 특정 데이터베이스에 고유의 등록 정보가 설정되지 않은 경우 기본 소스가 사용됩니다. 이전 예에서는 password, group, host 및 automount를 제외한 모든 데이터베이스가 로컬 파일을 소스를 사용합니다. 기본 소스 이외의 다른 소스가 필요한 경우 특정 데이터베이스에 대한 등록 정보가 만들어집니다. 이 예에서는 password, groups 및 automount가 먼저 로컬 파일에서 검색된 후 LDAP에서 검색됩니다. 호스트 조회는 먼저 로컬 파일에서 검색된 후 DNS에서 검색됩니다.

시스템에서 사용으로 설정된 이름 지정 서비스를 변경하는 경우 올바른 이름 지정 서비스가 사용되도록 name-service/switch SMF 서비스의 적합한 등록 정보를 업데이트해야 합니다. 예를 들어, name-service/switch가 이전 예와 유사하게 구성되었으며 대신 LDAP을 사용 안 함으로 설정하고 NIS를 사용으로 설정했다고 가정합니다.

- 이 경우 파일 및 NIS를 사용하도록 name-service/switch 서비스의 다음 등록 정보를 설정해야 합니다.
- config/password
- config/group
- config/automount

해당 등록 정보를 올바르게 설정하려면 다음 명령을 입력합니다.

```

# svccfg -s name-service/switch setprop config/password = astring: "'files nis'"
# svccfg -s name-service/switch setprop config/group = astring: "'files nis'"
# svccfg -s name-service/switch setprop config/automountconfig/password = astring:
"'files nis'"
# svccfg -s name-service/switch:default refresh

```

자세한 내용은 “Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”의 2 장, “이름 서비스 스위치 정보” 및 “Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”의 “이름 서비스 스위치 구성”을 참조하십시오.

## 로컬 파일 모드에 대한 시스템 구성

로컬 파일 모드로 실행하는 경우 시스템은 로컬 디렉토리에 있는 파일에서 모든 TCP/IP 구성 정보를 가져옵니다. 네트워크 클라이언트 모드에서는 원격 네트워크 구성 서버가 네트워크의 모든 시스템에 구성 정보를 제공합니다.

일반적으로 네트워크의 다음 서버는 로컬 파일 모드로 실행됩니다.

- 네트워크 구성 서버
- NFS 서버
- NIS, LDAP 또는 DNS 서비스를 제공하는 이름 서버

- 메일 서버
- 라우터

클라이언트는 지정된 네트워크에서 네트워크 클라이언트 모드 또는 로컬 파일 모드로 실행될 수 있으므로 구성된 여러 시스템과 해당 모드를 함께 사용할 수 있습니다.

## ▼ 로컬 파일 모드에 대한 시스템 구성 방법

1. 관리자로 로그인합니다.
2. 지정된 IP 주소로 시스템의 IP 인터페이스를 구성합니다.  
IPv4 인터페이스를 구성하는 방법을 참조하십시오.
3. 호스트 이름이 올바르게 설정되었는지 확인합니다.

```
# hostname
```

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

4. `/etc/inet/hosts`의 항목이 최신인지 확인합니다.  
Oracle Solaris가 주 네트워크 인터페이스, 루프백 주소 및 설치 중 구성된 추가 인터페이스 (해당하는 경우)에 대한 항목을 만듭니다. 항목이 최신이 아닌 경우 설치 후 시스템에 추가된 네트워크 인터페이스에 대한 IP 주소 및 해당 이름을 추가합니다.
5. `nis/domain` SMF 서비스의 등록 정보로 시스템의 정규화된 도메인을 지정합니다.  
예를 들어, 다음과 같이 `deserts.worldwide.com`을 `nis/domain` SMF 서비스의 `domainname` 등록 정보에 대한 값으로 지정합니다.

```
# domainname domainname
```

이 단계에서는 변경 사항이 지속되도록 합니다.

6. 경로 지정 정보를 추가합니다.

---

참고 - DHCP 서비스를 사용 중인 경우 이 단계를 건너뛸 수 있습니다.

---

지침은 “[경로 지정 구성](#)” [53]을 참조하십시오.

7. 넷마스크 정보(해당하는 경우)를 추가합니다.

---

참고 - DHCP 서비스를 사용 중인 경우 이 단계를 건너뛸 수 있습니다.

---

- a. `/etc/inet/netmasks` 파일에 네트워크 번호 및 넷마스크를 입력합니다.

항목을 만들려면 *network-number netmask* 형식을 사용합니다. 예를 들어, 클래스 C 네트워크 번호 192.168.83을 지정하려면 다음 정보를 입력합니다.

```
192.168.83.0    255.255.255.0
```

CIDR 주소의 경우 네트워크 접두어를 동등한 점으로 구분된 십진수 표현으로 변환합니다. 예를 들어, CIDR 네트워크 접두어 192.168.3.0/22를 표현하려면 다음 정보를 입력합니다.

```
192.168.3.0    255.255.252.0
```

- b. 로컬 파일만 검색되도록 *name-service/switch* 등록 정보에서 넷마스크에 대한 조회 소스를 변경한 다음 인스턴스를 새로 고칩니다.

```
# svccfg -s name-service/switch setprop config/netmask = astring: "files"
# svccfg -s name-service/switch:default refresh
```

8. 시스템을 재부트합니다.

## DNS 클라이언트 구성

DNS는 두 부분(응답을 제공하는 서비스와 서비스를 질의하는 클라이언트)으로 구성됩니다. Oracle Solaris에서 기본 DNS 서비스는 ISC(Internet Systems Consortium)의 BIND(Berkeley Internet Name Domain) 및 연관된 *named* 서버 데몬에 의해 제공됩니다. DNS 클라이언트는 유틸리티 및 라이브러리 컬렉션으로 구성됩니다.

추가 작업 관련 정보는 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”의 “DNS 관리\(작업\)”](#)를 참조하십시오.

### ▼ DNS 클라이언트를 사용으로 설정하는 방법

1. 관리자로 로그인합니다.
2. 검색할 도메인과 DNS 이름 서버에 대한 IP 주소를 나열한 다음 SMF 저장소를 업데이트합니다. 예를 들면 다음과 같습니다.

```
# svccfg -s network/dns/client
svc:/network/dns/client> setprop config/search = astring: ("example.com"
"sales.example.com")
svc:/network/dns/client> setprop config/nameserver = net_address: (192.168.1.10
192.168.1.11)
svc:/network/dns/client> select network/dns/client:default
svc:/network/dns/client:default> refresh
svc:/network/dns/client:default> quit
```

변경 사항을 적용하려면 서비스를 새로 고쳐야 합니다.

3. DNS를 사용하도록 이름 서비스 스위치 정보를 업데이트합니다.  
첫번째 명령은 SMF 저장소의 DNS 구성 정보를 업데이트합니다.

```
# svccfg -s system/name-service/switch
svc:/system/name-service/switch> setprop config/host = astring: "files dns"
svc:/system/name-service/switch> select system/name-service/switch:default
svc:/system/name-service/switch:default> refresh
svc:/system/name-service/switch:default> quit
```

4. DNS 클라이언트를 실행하는 데 필요한 서비스를 시작합니다.

```
# svcadm enable network/dns/client
# svcadm enable system/name-service/switch
```

5. 다음 명령 중 하나를 사용하거나 모두 사용하여 DNS 클라이언트가 사용으로 설정되었는지 확인합니다.

```
# dig knownserver.example.com
# getent hosts knownserver.example.com
```

dig 명령은 단독으로 사용될 때 DNS 클라이언트가 사용으로 설정되었는지 확인합니다.  
getent hosts 명령은 DNS 클라이언트의 /etc/nsswitch.conf 파일 사용을 확인합니다.

#### 예 4-1 클라이언트에 대한 여러 DNS 옵션 동시 설정

다음 예에서는 여러 /etc/resolv.conf 옵션을 설정하는 방법을 보여줍니다.

```
# svccg
svc:> select /network/dns/client
svc:/network/dns/client> setprop config/options = "ndots:2 retrans:3 retry:1"
svc:/network/dns/client> listprop config/options
config/options astring      ndots:2 retrans:3 retry:1
svc:/network/dns/client> exit
# svcadm refresh dns/client
# grep options /etc/resolv.conf
options ndots:2 retrans:3 retry:1
```

## 멀티캐스트 DNS 사용

mDNS(멀티캐스트 DNS) 및 DNS 서비스 검색이 작동하려면 mDNS에 참여할 모든 시스템에 mDNS를 배치해야 합니다. mDNS 서비스는 시스템에 제공되는 서비스의 가용성을 알리는 데 사용됩니다.

mDNS를 사용으로 설정하기 전에 소프트웨어 패키지가 시스템에 설치되어 있어야 합니다. 필요한 경우 다음과 같이 패키지를 설치하십시오.

```
# pkg install pkg:/service/network/dns/mdns
```

mDNS를 사용으로 설정하는 프로세스의 일부로 먼저 이름 서비스 스위치 정보를 업데이트해야 합니다. 로컬 호스트를 확인할 수 있으려면 다음과 같이 mdns를 소스로 포함하도록 name-service/switch SMF 서비스의 config/host 등록 정보를 변경해야 합니다.

```
# /usr/sbin/svccfg -s svc:/system/name-service/switch
svc:/system/name-service/switch> setprop config/host = astring: "files dns mdns"
svc:/system/name-service/switch> select system/name-service/switch:default
svc:/system/name-service/switch:default> refresh
svc:/system/name-service/switch> quit
```

다음과 같이 mDNS SMF 서비스를 사용으로 설정하십시오.

```
# svcadm enable svc:/network/dns/multicast:default
```

이 방법으로 mDNS를 사용으로 설정할 경우 업그레이드 및 재부트 후에도 변경 사항이 지속됩니다. 자세한 내용은 [svcadm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## DNS에 대한 리소스 알림

dns-sd 명령을 통해 ping 또는 traceroute 명령을 사용하는 방법과 유사하게 서비스를 찾아 보고 검색할 수 있습니다. 명령줄 인수와 출력 형식은 지속적으로 변경될 수 있으며 이로 인해 셸 스크립트에서 명령을 호출할 때 예측이 불가능하고 위험한 상황이 발생할 수 있으므로 dns-sd 명령은 주로 대화식으로 사용됩니다. 또한 DNS-SD(DNS 서비스 검색)의 비동기 특성상, 스크립트 지향 프로그래밍이 쉽게 적용되지 않습니다.

예는 “Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”의 “DNS에 대한 리소스 알림”을 참조하십시오. [dns-sd\(1M\)](#) 매뉴얼 페이지도 참조하십시오.

## NIS 클라이언트 구성

NIS는 네트워크 객체 및 리소스를 식별하고 찾는 데 사용되는 분산 이름 지정 서비스입니다. NIS는 전송 프로토콜 및 매체 독립적 방식으로 네트워크 전체 정보를 저장하고 검색하는 통합적 방법을 제공합니다. NIS는 DNS와 독립적으로 개발되었으며, 주력하는 부분은 약간 다릅니다. DNS는 숫자 IP 주소 대신 시스템 이름을 사용하여 통신을 더 간소화하는 데 주력하고, NIS는 다양한 네트워크 정보에 대한 중앙집중 제어를 제공하여 네트워크 관리를 더 용이하게 하는 데 주력합니다.

NIS를 실행하여 다양한 서버(마스터 및 슬레이브) 간에 관리 데이터베이스(맵)를 분산할 수 있습니다. 안정적인 자동 방식으로 중앙 위치에서 이러한 데이터베이스를 업데이트하여 모든 클라이언트가 전체 네트워크에서 일관된 방식으로 동일한 이름 지정 서비스 정보를 공유하도록 할 수 있습니다. 자세한 개요는 “Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”의 5 장, “NIS(네트워크 정보 서비스) 정보”를 참조하십시오.

시스템에서 NIS 클라이언트 서비스를 제공하는 두 개의 SMF 서비스가 있습니다. svcadm 명령을 사용하여 해당 서비스를 사용 및 사용 안함으로 설정하고 다시 시작하며 새로 고칠 수 있습니다. 다음과 같이 시스템에서 NIS 서비스의 상태를 표시하십시오.

```
# svcs \*nis\*
STATE          STIME      FMRI
online         Oct_09    svc:/network/nis/domain:default
online         Oct_09    svc:/network/nis/client:default
```

추가 작업 관련 정보는 [“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)의 6 장, [“NIS\(네트워크 정보 서비스\) 설정 및 구성”](#)을 참조하십시오.

## ▼ 브로드캐스트 모드로 NIS 클라이언트를 구성하는 방법

브로드캐스트 모드를 사용하면 가장 간단하게 NIS 클라이언트를 설정할 수 있습니다. nis/client SMF 서비스를 시작하면 서비스에서 ypbind 명령을 실행하고, 이 명령은 로컬 서브넷에서 NIS 서버를 검색합니다. 서브넷이 발견되면 ypbind 명령은 바인드를 수행합니다. 이 검색을 브로드캐스팅이라고 합니다. 클라이언트의 로컬 서브넷에 NIS 서버가 없으면 ypbind 명령이 바인드를 수행하지 못하며 클라이언트 시스템이 NIS 서비스에서 이름 공간 데이터를 가져올 수 없습니다. [특정 NIS 서버를 사용하여 NIS 클라이언트를 구성하는 방법 \[83\]](#)을 참조하십시오.

1. 관리자로 로그인합니다.
2. NIS 도메인 이름을 설정합니다.

```
# domainname example.com
```

3. 필요한 경우 이름 서비스 스위치를 변경합니다.  
[“Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: DNS 및 NIS”](#)의 [“이름 서비스 스위치 구성”](#)을 참조하십시오.

4. NIS 클라이언트 서비스를 시작합니다.

```
# svcadm enable network/nis/domain
# svcadm enable network/nis/client
```

## ▼ 특정 NIS 서버를 사용하여 NIS 클라이언트를 구성하는 방법

시작하기 전에 다음 절차를 수행하려면 3단계에서 지정된 호스트 이름(서버)을 DNS로 확인할 수 있어야 합니다. DNS를 사용하지 않거나 IP 주소 대신 호스트 이름을 입력하는 경우 클라이언트의 /etc/hosts 파일에 각 NIS 서버에 대한 적합한 항목을 추가해야 합니다. 자세한 내용은 [ypinit\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

1. 관리자로 로그인합니다.

2. NIS 도메인을 설정합니다.

```
# domainname example.com  
# svcadm enable network/nis/domain
```

3. 클라이언트 구성 스크립트를 실행합니다.

```
# ypinit -c
```

클라이언트가 이름 지정 서비스 정보를 가져오는 NIS 서버의 이름을 지정하라는 메시지가 표시됩니다. 마스터 서버와 원하는 개수만큼 슬레이브 서버를 나열할 수 있습니다. 나열하는 서버는 도메인의 모든 위치에 배치될 수 있습니다. 먼저 네트워크 측면에서 시스템에 가장 가까운 서버를 나열한 다음 네트워크에서 더 먼 위치에 있는 서버를 나열하는 것이 좋습니다.

4. NIS 클라이언트를 사용으로 설정합니다.

```
# svcadm enable network/nis/client
```

## ▼ NIS 클라이언트 서비스를 사용 안함으로 설정하는 방법

1. 관리자로 로그인합니다.

2. 다음과 같이 NIS 클라이언트 서비스를 중지합니다.

```
# svcadm disable network/nis/domain  
# svcadm disable network/nis/client
```

## LDAP 클라이언트 구성

Oracle Solaris 클라이언트가 LDAP을 이름 지정 서비스로 사용하려면 다음 요구 사항을 충족해야 합니다.

- LDAP 서버가 클라이언트의 도메인 이름을 제공해야 합니다.
- 이름 서비스 스위치가 필수 서비스의 LDAP을 가리켜야 합니다.
- 동작을 정의하는 모든 특정 매개변수를 사용하여 클라이언트를 구성해야 합니다.
- `ldap_cachemgr`이 클라이언트에서 실행되고 있어야 합니다.
- 클라이언트가 구성된 서버가 하나 이상 작동되어 실행 중이어야 합니다.

`ldapclient` 명령은 서버 시작을 제외하고 이전에 나열된 모든 작업을 수행하므로 LDAP 클라이언트 설정에 중요합니다.

네트워크 구성에 고정적 모드를 사용하는 경우 클라이언트 시스템에서 LDAP을 설정하는 가장 간편한 방법은 `DefaultFixed` 프로파일을 사용하여 설정하고 지속 네트워크 구성을 수

행하는 것입니다. 그런 다음 `ldapclient` 명령을 사용하여 프로파일 또는 수동 설정을 통해 LDAP 설정을 완료할 수 있습니다. 자세한 내용은 [ldapclient\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

LDAP에 대한 자세한 개요는 “[Oracle Solaris 11.2의 이름 지정 및 디렉토리 서비스 작업: LDAP](#)”을 참조하십시오.

## 이름 지정 서비스 구성 가져오기

`nscfg` 명령은 `name-service` 스위치 구성 요소에 대한 레거시 파일 구성을 SMF 저장소로 전송합니다. Oracle Solaris 11로 업그레이드하는 경우 시스템의 이름 지정 서비스 구성이 자동으로 SMF로 마이그레이션됩니다. 하지만 필요한 경우 `nscfg` 명령을 사용하여 수동으로 이 구성을 SMF로 마이그레이션할 수 있습니다.

다음 명령은 레거시 파일을 가져온 다음 구성을 변환하여 SMF에 푸시합니다.

```
# /usr/sbin/nscfg import -f FMRI
```

`nscfg` 명령을 사용하면 가장 간단하게 DNS 구성에 기존 `resolv.conf` 파일의 정보를 채울 수 있습니다. 다음 예에서는 `nscfg` 명령이 `/etc/resolv.conf` 파일의 정보를 읽어 변환한 다음 `svc:/network/dns/client` SMF 서비스에 정보를 저장합니다.

```
# cp resolv.conf /etc/resolv.conf
# /usr/sbin/nscfg import -f dns/client
# svcadm enable dns/client
```

시스템의 이름 지정 서비스를 변경하는 경우 다음 예에서와 같이 이름 서비스 스위치 정보도 적절히 수정해야 하며 이름 서비스 캐시에서 사용되지 않는 정보를 비우는 것이 좋습니다.

```
# cp /etc/nsswitch.dns /etc/nsswitch.conf
# /usr/sbin/nscfg import -f name-service/switch
# svcadm refresh name-service/switch
# svcadm refresh name-service/cache
```

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

## SMF 이름 지정 서비스 구성 재설정

다음과 같이 SMF 이름 지정 서비스의 구성 등록 정보를 다시 `files only` 모드로 재설정할 수 있습니다.

```
# /usr/sbin/nscfg unconfig FMRI
# svcadm refresh name-service/switch
```

예를 들어, 다음과 같이 “[이름 지정 서비스 구성 가져오기](#)” [85]에서 SMF 구성에 적용한 변경 사항을 되돌립니다.

```
# svcadm disable dns/client
# /usr/sbin/nscfg unconfig dns/client
# /usr/sbin/nscfg unconfig name-service/switch
# svcadm refresh name-service/switch
# svcadm refresh name-service/cache
```

# ◆◆◆ 5 장

## Oracle Solaris의 프로파일 기반 네트워크 구성 관리 정보

---

이 장에서는 시스템에서 반응적 모드를 사용할 때 주로 사용되는 프로파일 기반 네트워크 구성의 개요를 제공합니다. 반응적 모드는 Oracle Solaris 시스템에서 네트워크 구성을 관리할 수 있도록 다양한 유형의 프로파일을 지원합니다. 이 모드는 노트북 PC용으로, 그리고 네트워크 상태가 자주 변경되는 경우 주로 사용됩니다. 반응적 모드에서 네트워크 데몬(nwamd)은 시스템의 상태를 모니터링하고, 상태가 변경되는 경우 동적으로 네트워크 구성을 조정합니다.

고정적 모드 사용 및 회사 환경에서 주로 사용되는 네트워크 관리 모델에 대한 자세한 내용은 [2장. Oracle Solaris의 데이터 링크 구성 관리](#) 및 [3장. Oracle Solaris에서 IP 인터페이스와 주소 구성 및 관리](#)를 참조하십시오.

네트워크 프로파일 구성 및 관리에 대한 단계별 지침은 [6장. Oracle Solaris의 프로파일 기반 네트워크 구성 관리](#)를 참조하십시오.

이 장의 내용:

- “반응적 모드 정보” [87]
- “프로파일 기반 네트워크 구성 정보” [88]
- “프로파일 기반 네트워크 구성 사용 지침” [93]
- “프로파일 기반 네트워크 구성 사용을 위한 보안 요구 사항” [95]
- “프로파일 기반 네트워크 구성이 다른 Oracle Solaris 기능에서 작동하는 방식” [96]

### 반응적 모드 정보

반응적 모드에서는 자동으로 시스템이 네트워크 상태 변경 사항에 대응하고 시스템의 현재 네트워크 구성을 조정하므로 수동 재구성이 필요하지 않습니다. 예를 들어, 유선 네트워크 인터페이스가 분리되거나 새 무선 네트워크가 사용 가능한 상태가 되거나 물리적 위치를 변경하는 경우 이에 따라 시스템이 네트워크 구성에 대응합니다.

이동성에 중점을 두는 Oracle Solaris의 반응적 네트워크 구성 정책을 사용하여 여러 네트워크 이벤트에 대한 응답으로 또는 사용자 요청 시 시스템의 네트워크 구성을 동적으로 변경할 수 있습니다. 이 유형의 네트워크 구성은 노트북 PC용으로 가장 적합하며, 네트워크 상태가

자주 변경되는 경우 유용합니다. 반응적 모드를 사용할 때 시스템의 기본 이더넷 및 WiFi 구성이 자동으로 수행됩니다. 시작 시 자동으로 시스템이 유선 또는 무선 네트워크에 연결하고 데스크탑의 현재 활성화 네트워크 연결 상태에 대한 알림을 표시합니다. 특정 프로파일이 사용으로 설정되는 조건을 결정할 등록 정보로 반응적 프로파일을 구성할 수 있습니다. 해당 등록 정보를 사용하면 필요에 따라 네트워크 관리 데몬 `nwamd`를 통해 동적으로 시스템에 프로파일 구성을 적용할 수 있습니다.

일반적으로 반응적 네트워크 구성 변경 사항은 다음 이벤트 및 작업으로 트리거됩니다.

- 이더넷 케이블 연결 또는 연결 해제
- WLAN(무선 LAN) 카드 연결 또는 연결 해제
- 유선 인터페이스 또는 무선 인터페이스를 사용할 수 있는 경우 시스템 부트
- 유선 인터페이스 또는 무선 인터페이스를 사용할 수 있는 경우 일시 중지에서 계속(지원되는 경우)
- DHCP 임대 획득 또는 손실

두 가지 명령을 사용하여 반응적 모드로 네트워크 구성을 관리할 수 있습니다. `netcfg` 명령은 프로파일에 대한 네트워크 구성을 변경하는 데 사용되며, `netadm` 명령은 프로파일에 대한 정보를 표시하고 프로파일을 사용 및 사용 안함으로 설정하는 데 사용됩니다. 전체 설명은 `netcfg(1M)` 및 `netadm(1M)` 매뉴얼 페이지를 참조하십시오. 작업 관련 정보는 [6장. Oracle Solaris의 프로파일 기반 네트워크 구성 관리](#)를 참조하십시오.

반면 고정적 모드는 반응적 모드와 반대입니다. 고정적 모드를 사용 중인 경우 네트워크 데몬이 시스템에서 특정 네트워크 구성을 인스턴스화하지만 다양한 네트워크 상태에 따라 해당 구성을 자동으로 조정하지는 않습니다. 고정적 모드에 대한 자세한 내용은 [“네트워크 구성 모드 정보” \[13\]](#)를 참조하십시오.

지원되는 모든 네트워크 관리 명령 및 사용 시기에 대한 자세한 내용은 [“Oracle Solaris 네트워크 관리 명령” \[14\]](#)을 참조하십시오.

## 프로파일 기반 네트워크 구성 정보

Oracle Solaris는 미리 정해진 일련의 시스템 정의 프로파일을 제공하며, 특정 네트워킹 요구 사항을 충족시키기 위해 지정한 활성화 조건과 등록 정보로 다양한 유형의 사용자 정의 반응적 프로파일을 만들 수 있는 기능을 제공합니다. 사용자 정의 프로파일을 사용하면 시스템에서 데이터 링크 및 IP 주소의 기본 구성을 간소화하고 이름 지정 서비스, IP 필터, IPsec(IP 보안) 구성 등과 같이 보다 복잡한 시스템 차원의 네트워크 구성을 정의할 수 있습니다.

지원되는 프로파일 유형은 다음과 같습니다.

- **NCP(네트워크 구성 프로파일)** - NCP는 네트워크 데이터 링크 및 IP 인터페이스의 구성을 지정하는 데 사용되는 주요 프로파일 유형입니다. NCP는 시스템에서 특정 NCP가 활성화된 경우 네트워크 구성 방법을 지정하는 등록 정보 값으로 구성됩니다. NCP는 반응적 또는 고정적일 수 있습니다. 반응적 NCP는 여러 개 구성할 수 있지만, 고정적 NCP는 Oracle Solaris에서 `DefaultFixed` 하나만 지원됩니다.

- **NCU(네트워크 구성 단위)** - NCP를 정의하는 개별 구성 정보(등록 정보)가 NCU 내에서 지정됩니다. 각 NCU는 물리적 링크 또는 인터페이스를 나타낼 수 있으며 해당 링크나 인터페이스의 구성을 지정하는 등록 정보를 포함합니다.
- **위치 프로파일** - 위치 프로파일(위치라고도 함)은 시스템 차원의 네트워크 구성(예: 이름 지정 서비스, 도메인, IP 필터 구성 및 IPsec 구성)을 지정합니다.
- **ENM(외부 네트워크 수정자)** - ENM은 시스템의 주 네트워크 구성과 무관하게 네트워크 구성을 만들어야 하는 응용 프로그램(예: VPN 응용 프로그램)을 관리하는 프로파일입니다.
- **알려진 WLAN** - 알려진 WLAN은 시스템에서 검색한 무선 네트워크에 대한 정보를 저장하는 프로파일입니다.

## 프로파일 유형 설명

다음은 Oracle Solaris 릴리스에서 지원되는 각 프로파일 유형에 대한 자세한 설명입니다.

### NCP에 대한 설명

NCP는 시스템 특정 네트워크 구성(예: 데이터 링크, IP 인터페이스 및 주소)을 정의합니다. 각 NCP에 포함되는 다양한 NCU(네트워크 구성 단위)는 여러 네트워크 링크와 인터페이스를 구성하는 방법(예: 표시할 인터페이스, 인터페이스를 표시할 조건) 및 인터페이스의 IP 주소를 가져오는 방법을 지정합니다.

Automatic NCP는 현재 시스템에 있는 모든 네트워크 링크와 인터페이스를 나타냅니다. 네트워크 장치를 추가하거나 제거하면 Automatic NCP의 콘텐츠가 변경됩니다. Automatic NCP는 DHCP 주소 자동 구성을 활용하는 프로파일에 대한 액세스를 제공하여 시스템의 IP 주소를 가져올 수 있도록 합니다. 이 NCP는 무선 링크보다 유선 링크를 선호하는 링크 선택 정책을 구현합니다. 대체 IP 구성 정책 또는 대체 링크 선택 정책을 지정해야 하는 경우 시스템에서 다른 NCP를 만들어야 합니다. Automatic NCP는 삭제할 수 없습니다. 이 NCP를 복사하고 복사본을 변경할 수 있습니다. [예 6-7. “Automatic NCP를 복제하여 NCP 만들기”](#)을 참조하십시오.

### NCU에 대한 설명

NCU에는 NCP를 정의하는 등록 정보 값이 포함됩니다. NCU는 시스템에 있는 개별 물리적 링크와 인터페이스를 나타냅니다. 사용자 정의 NCP를 구성하는 프로세스에는 각 링크와 인터페이스를 구성하는 방법 및 조건을 지정하는 NCU를 만드는 작업이 포함됩니다.

다음 두 가지 NCU 유형이 있습니다.

- **링크 NCU** - 물리적 장치, 즉 OSI(Open Systems Interconnection) 모델의 계층 2 엔티티를 나타냅니다.
- **인터페이스 NCU** - IP 인터페이스(계층 3 엔티티)를 나타냅니다.

링크 NCU는 다음과 같은 데이터 링크 계층 클래스를 나타냅니다.

- 통합
- 브리지
- Etherstub
- EoIB(Ethernet over IB)
- 물리적 링크(이더넷 또는 WiFi)
- 터널
- VXLAN(가상 확장 가능 LAN)
- VLAN(가상 LAN)
- VNIC(가상 네트워크 인터페이스 카드)

인터페이스 NCU는 다음과 같은 IP 계층 클래스를 나타냅니다.

- IP 인터페이스
- IPMP 인터페이스
- VNI 인터페이스

다양한 객체 유형에 설정할 수 있는 등록 정보에 대한 자세한 내용은 [netcfg\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## 위치 프로파일에 대한 설명

위치 프로파일(간단히 위치라고도 함)은 해당 위치가 활성화 상태인 경우 시스템 차원의 네트워크 구성을 지정하기 위해 함께 적용되는 이름 지정 서비스 및 방화벽 설정과 같은 네트워크 구성 정보로 구성됩니다. 위치가 반드시 물리적 위치와 일치하지는 않기 때문에 각 네트워킹 요구 사항을 충족하는 위치 프로파일을 여러 개 설정할 수 있습니다. 예를 들어, 한 위치는 회사 인트라넷에 연결된 경우에 사용할 수 있습니다. 다른 위치는 사무실에 있는 무선 액세스 포인트를 통해 공용 인터넷에 연결된 경우에 사용할 수 있습니다.

기본적으로 시스템에서 미리 정의한 위치 3개가 있습니다.

### ■ DefaultFixed

DefaultFixed NCP가 활성화 상태인 경우 항상 DefaultFixed 위치가 사용으로 설정됩니다. netcfg 명령을 사용하여 직접 DefaultFixed 위치를 수정할 수 없습니다. DefaultFixed NCP를 사용으로 설정하는 과정에서 이 위치가 사용으로 설정되는 경우 위치 설정이 반영되도록 관련 SMF(서비스 관리 기능) 등록 정보가 업데이트됩니다. 시스템이 종료되거나 다른 위치가 사용으로 설정되는 경우 DefaultFixed 위치 구성 과정에서 관련 SMF 등록 정보가 저장됩니다.

### ■ Automatic

Automatic 위치는 사용 가능한 네트워크가 있지만 대체하는 다른 위치가 없을 경우 활성화됩니다. netcfg 명령을 사용하여 Automatic 위치를 수정할 수 있습니다.

참고 - Automatic 위치를 Automatic NCP와 혼동해서는 안됩니다. Automatic 위치는 시스템의 초기 네트워크 구성이 수행된 후 시스템 차원의 네트워크 등록 정보를 정의합니다. Automatic NCP는 시스템의 링크 및 인터페이스 네트워크 구성을 지정합니다.

#### ■ NoNet

NoNet 위치에는 특정 활성화 조건이 있습니다. 지정된 IP 주소를 가진 로컬 인터페이스가 없는 경우 시스템에서 이 위치를 독립형 시스템에 적용합니다. netcfg 명령을 사용하여 NoNet 위치를 수정할 수 있습니다.

사용자 정의 위치는 시스템 정의 위치와 동일합니다. 단, 사용자 정의 위치는 지정된 사용자 정의 값으로 구성되지만 시스템 정의 위치에는 미리 설정된 값이 있습니다.

## ENM에 대한 설명

ENM을 통해 응용 프로그램 또는 스크립트가 NCP 및 위치 프로파일에서 지정된 구성과 무관하게 네트워크 구성을 수행할 시기를 지정할 수 있습니다. ENM은 사용 또는 사용 안함으로 설정될 경우 네트워크 구성을 직접 수정하는 서비스나 응용 프로그램으로 정의될 수도 있습니다. ENM을 사용 또는 사용 안함으로 설정해야 하는 조건을 지정할 수 있습니다. ENM을 수동으로 사용 또는 사용 안함으로 설정할 수도 있습니다. 지정된 한 시점에 각 프로파일 유형 중 하나만 시스템에서 활성 상태일 수 있는 NCP 또는 위치 프로파일과 달리, ENM은 동시에 여러 개가 시스템에서 활성 상태일 수 있습니다. 지정된 한 시점에 활성 상태인 ENM이 동시에 시스템에서 사용으로 설정되는 NCP 또는 위치 프로파일에 반드시 종속되는 것은 아닙니다.

ENM을 만들 수 있는 여러 외부 응용 프로그램과 서비스가 있지만 그 중 가장 많이 사용되는 것은 VPN 응용 프로그램입니다. 시스템에서 VPN을 설치 및 구성한 후 지정한 조건에 따라 VPN을 자동으로 활성화 및 비활성화하는 ENM을 만들 수 있습니다.

참고 - 반응적 네트워크 구성 모드는 시스템에서 네트워크 구성을 직접 수정할 수 있는 외부 응용 프로그램을 자동으로 감지할 수 없습니다. VPN 응용 프로그램이나 모든 외부 응용 프로그램 또는 서비스의 활성화나 비활성화를 관리하려면 먼저 응용 프로그램을 설치하고 CLI(명령줄 인터페이스) 또는 네트워크 관리 GUI를 사용하여 응용 프로그램에 대한 ENM을 만들어야 합니다.

ENM에서 수행되는 네트워크 구성에 대한 지속 정보는 NCP 또는 위치 프로파일 정보의 저장 방식과 정확히 동일하게 저장되거나 추적되지 않습니다. 하지만 시스템에서는 외부적으로 시작된 네트워크 구성을 확인한 다음 ENM의 시스템 구성 변경 사항을 기반으로 활성화할 위치를 재평가한 다음 해당 위치를 활성화할 수 있습니다. 예를 들어, 특정 IP 주소가 사용 중일 때 조건부로 활성화되는 위치로 전환합니다. 언제든지 svc:/network/physical:default 서비스를 다시 시작하면 활성 NCP에서 지정된 네트워크 구성이 복구됩니다. ENM도 다시 시작되어 프로세스의 네트워크 구성을 해제하고 다시 만들 수 있습니다.

## 알려진 WLAN에 대한 설명

알려진 WLAN은 시스템에 알려진 무선 네트워크를 관리하는 데 사용되는 프로파일입니다. 시스템에서는 이와 같이 알려진 무선 네트워크의 전역 목록을 유지 관리합니다. 이 정보는 사용 가능한 무선 네트워크에 연결을 시도하는 순서를 결정하는 데 사용됩니다. 알려진 WLAN 목록에 있는 무선 네트워크를 사용할 수 있는 경우 시스템에서 자동으로 해당 네트워크에 연결합니다. 알려진 무선 네트워크를 두 개 이상 사용할 수 있는 경우 시스템에서는 우선 순위가 가장 높은(가장 낮은 번호) 무선 네트워크에 연결을 시도합니다. 연결하는 새 무선 네트워크가 알려진 WLAN 목록의 맨 위에 자동으로 추가되고 우선 순위가 가장 높은 현재 무선 네트워크가 됩니다.

기본적으로 이전에 연결한 WLAN보다 최근에 연결한 WLAN이 선호됩니다. 알려진 WLAN은 동일한 우선 순위를 공유할 수 없습니다. 기존 WLAN과 동일한 우선 순위 값을 가진 새 WLAN을 목록에 추가하면 기존 항목이 더 낮은 우선 순위 값으로 이동됩니다. 그 뒤에 목록에서 다른 모든 WLAN의 우선 순위 값이 동적으로 더 낮은 우선 순위 값으로 이동됩니다.

하나의 키 이름을 알려진 WLAN과 연관시킬 수도 있습니다. 키 이름을 사용하면 `dladm create-secobj` 명령으로 고유한 키를 만들 수 있습니다. 그런 다음 WLAN keyname 등록 정보에 보안 객체 이름을 추가하여 이 키를 WLAN과 연관시킬 수 있습니다. 자세한 내용은 [dladm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

명령줄에서의 WLAN 관리에 대한 자세한 내용은 [“반응적 모드로 알려진 WLAN 관리” \[132\]](#)를 참조하십시오.

## 시스템 정의 및 사용자 정의 프로파일

Automatic NCP는 시스템에 있는 각 물리적 링크에 대한 링크 NCU 한 개와 인터페이스 NCU 한 개로 구성된 시스템 정의 프로파일입니다. 이 NCP의 NCU 활성화 정책은 무선 링크보다 연결된 유선 링크를 선호하고 각 사용 링크에서 IPv4와 IPv6을 모두 연결합니다. DHCP는 IPv4 주소를 가져오는 데 사용됩니다. Stateless 자동 구성 및 DHCP는 IPv6 주소를 가져오는 데 사용됩니다. Automatic NCP는 새 링크를 시스템에 삽입하거나 시스템에서 제거할 때 동적으로 변경됩니다. 삽입 또는 제거된 링크에 해당하는 NCU도 동시에 모두 추가되거나 제거됩니다. 이 프로파일은 `nwamd` 데몬에 의해 자동으로 업데이트됩니다.

활성 상태인 경우 Automatic NCP는 다음과 같은 기본 정책을 구현합니다.

- DHCP를 통해 사용 가능한(연결된) 이더넷 인터페이스를 모두 구성합니다.
- 이더넷 인터페이스가 연결되어 있지 않거나 IP 주소를 얻을 수 없는 경우 무선 인터페이스를 사용으로 설정하고 알려진 WLAN 목록에서 사용 가능한 최상의 WLAN에 자동으로 연결합니다. [“알려진 WLAN에 대한 설명” \[92\]](#)을 참조하십시오.
- 하나 이상의 IP4 주소를 얻을 때까지 NoNet 위치를 활성 상태로 유지합니다. [“위치 프로파일에 대한 설명” \[90\]](#)을 참조하십시오. 이 위치는 IP 주소 획득과 관련된 데이터(DHCP 및 IPv6 autoconf 메시지)만 전달하는 엄격한 IP 필터 규칙 세트를 제공합니다. 활성화 조건을 제외하고 NoNet 위치의 모든 등록 정보를 수정할 수 있습니다.

- 시스템 인터페이스 중 하나에 IP 주소가 하나 이상 지정된 경우 Automatic 위치를 활성화합니다. 이 위치에는 IP 필터 또는 IPsec 규칙이 없습니다. 위치는 DHCP 서버로부터 얻은 DNS(Domain Name System) 구성 데이터를 적용합니다. NoNet 위치와 마찬가지로, 활성화 조건을 제외하고 Automatic 위치의 모든 등록 정보를 수정할 수 있습니다.

선택적으로 사용자 정의 NCP를 구성할 수 있습니다. 명시적으로 NCU를 지정된 NCP에 추가하고 NCP에서 제거해야 합니다. 현재 시스템에 있는 링크에 대한 상관 관계가 없는 NCU를 만들 수도 있습니다. 또한 사용자 정의 NCP에 대한 정책을 결정할 수 있습니다. 예를 들어, 지정된 시간에 여러 링크와 인터페이스가 시스템에서 사용으로 설정될 수 있게 하고 NCU와 정적 IP 주소 간에 다른 종속성 관계를 지정할 수 있습니다.

## 프로파일 기반 네트워크 구성 사용 지침

프로파일 기반 네트워크 구성에는 다음 지침이 적용됩니다.

- 지정된 시간에 시스템에서 하나의 NCP(네트워크 구성 프로파일) 및 하나의 위치 프로파일만 활성화 상태일 수 있습니다. 시스템의 다른 기존 NCP는 모두 작동하지 않습니다.
- 활성화 NCP는 반응적 또는 고정적(DefaultFixed)일 수 있습니다. 반응적 NCP를 사용할 경우 시스템에서 네트워크 구성을 모니터링하여 시스템의 네트워크 환경 변경 사항에 대응합니다. DefaultFixed NCP(시스템의 유일한 고정적 프로파일)를 사용할 경우 네트워크 구성이 인스턴스화되지만 모니터링되지 않습니다.
- NCP의 여러 등록 정보 값으로 구성된 정책에 따라 프로파일의 시스템 네트워크 구성 관리 방식이 제어됩니다.
- NCP 등록 정보에 대한 변경 사항은 새 등록 정보 값으로 즉시 구현된 후 해당 프로파일이 활성화 상태인 경우 항상 네트워크 구성을 관리하기 위한 프로파일의 정책에 포함됩니다.
- 시스템에서 반응적 모드가 사용되고 있을 경우 네트워크 구성을 관리하는 활성화 NCP는 Automatic NCP 또는 사용자가 만든 사용자 정의 반응적 NCP입니다. 반응적 NCP가 활성화 상태인 경우 네트워크 구성을 관리할 때는 netcfg 및 netadm 명령을 사용합니다.

시스템에서 고정적 모드가 사용되고 있을 경우 네트워크 구성을 관리하는 활성화 NCP는 항상 DefaultFixed입니다. 이 NCP가 활성화 상태인 경우 네트워크 구성을 관리할 때는 dtadm 및 ipadm 명령을 사용합니다. 자세한 내용은 [2장. Oracle Solaris의 데이터 링크 구성 관리](#) 및 [3장. Oracle Solaris에서 IP 인터페이스와 주소 구성 및 관리](#)를 참조하십시오.

## 프로파일 활성화 정책

프로파일 활성화 모드는 수동, 자동 또는 조건부입니다. 반응적 프로파일이 활성화 상태인 경우 네트워크 환경이 변경되면 시스템에서 네트워크 구성을 재평가한 다음 현재 상태를 기반으로 활성화할 가장 적합한 반응적 NCP 및 위치를 "추측"합니다. 네트워크 상태 변경으로는 이더넷 케이블 연결 또는 분리, DHCP 임대 획득 또는 손실, 새 무선 네트워크 감지가 있습니다. 시스템에는 항상 하나의 NCP 및 위치가 있어야 합니다.

반응적 네트워크 구성 모드를 사용하여 반응적 NCP에 대한 활성화 정책을 지정할 수 있습니다. 이 정책에 따라 NCU를 사용으로 설정할 시기가 결정됩니다. 각 위치 프로파일에는 활성화 기준을 정의하는 등록 정보도 포함되어 있습니다.

NCU, 위치 및 ENM에는 모두 activation-mode 등록 정보가 있습니다. 각 프로파일 유형에 허용되는 값은 서로 다릅니다. 각 프로파일 유형이 사용으로 설정되는 조건과 마찬가지로 activation-mode 등록 정보의 검증 방식도 각 프로파일 유형에 따라 달라집니다.

**참고** - NCU에 대한 activation-mode 등록 정보는 manual 또는 prioritized로 설정될 수 있습니다. 위치 프로파일의 activation-mode 등록 정보는 manual, conditional-any, conditional-all 또는 system으로 설정될 수 있습니다.

NCP 활성화 정책은 각 NCU에 대해 지정할 수 있는 등록 정보 및 조건을 사용하여 적용됩니다. 지정할 수 있는 정책의 예로 "무선 연결 대신 유선 연결 선호" 또는 "한 번에 하나씩 인터페이스 활성화"가 있습니다. NCP를 사용으로 설정하는 방식 및 시기는 각 NCU 유형에 대해 설정된 등록 정보에서 정의됩니다. 활성화 조건에 대한 자세한 내용은 [netcfg\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

**참고** - 인터페이스 NCU는 기본 링크 NCU와 연관됩니다. 연결된 링크 NCU를 사용으로 설정하면 각 인터페이스 NCU도 활성화됩니다. 기본 링크 NCU에 대한 종속성은 제거할 수 없습니다. 연관된 링크 NCU를 사용으로 설정하지 않고 인터페이스 NCU를 사용으로 설정하면 해당 인터페이스의 기본 링크 NCU를 사용으로 설정할 때까지 인터페이스 NCU가 실제로 활성화되지 않습니다.

## 프로파일 활성화 모드

사용자 정의 NCU, 위치 프로파일 및 ENM에는 모두 activation-mode 등록 정보가 있습니다. activation-mode 등록 정보는 프로파일을 만들거나 수정할 때 netcfg 명령을 사용하여 설정됩니다. NCP에는 activation-mode 등록 정보가 없습니다. 모든 NCP는 수동으로 사용으로 설정됩니다.

다음 표에서는 여러 프로파일 유형에 대한 activation-mode 등록 정보의 가능한 값을 설명합니다.

**표 5-1** activation-mode 등록 정보 값

프로파일 유형	activation-mode 값
NCU	manual 또는 prioritized
위치	manual, conditional-any, conditional-all 또는 system
ENM	manual, conditional-any 또는 conditional-all

프로파일 사용 및 사용 안함 설정에 대한 자세한 내용은 [“프로파일 사용 및 사용 안함으로 설정” \[97\]](#)을 참조하십시오.

## 프로파일 기반 네트워크 구성 사용을 위한 보안 요구 사항

netcfgd 데몬은 모든 네트워크 구성 정보가 저장되는 저장소를 제어합니다. netcfg 명령, 네트워크 관리 GUI 및 nwamd 데몬은 각각 저장소에 액세스하기 위해 netcfgd 데몬에 요청을 보냅니다.

현재 네트워크 구성 구현에서는 다음 권한을 사용하여 특정 작업을 수행합니다.

- `solaris.network.autoconf.read` - netcfgd 데몬에 의해 확인된 네트워크 프로파일 데이터를 읽을 수 있도록 합니다.
- `solaris.network.autoconf.write` - netcfgd 데몬에 의해 확인된 네트워크 프로파일 데이터를 쓸 수 있도록 합니다.
- `solaris.network.autoconf.select` - nwamd 데몬에 의해 확인된 새 구성 데이터를 적용할 수 있도록 합니다.
- `solaris.network.autoconf.wlan` - 알려진 WLAN 구성 데이터를 쓸 수 있도록 합니다.

이러한 권한은 `auth_attr` 데이터베이스에 등록됩니다. [auth\\_attr\(4\)](#) 매뉴얼 페이지를 참조하십시오.

`solaris.network.autoconf.read` 권한 부여는 기본적으로 모든 사용자에게 지정되는 Basic Solaris User 권한 프로파일에 포함됩니다. 따라서 이 권한이 부여된 모든 사용자는 현재 네트워크 상태 및 모든 네트워크 프로파일 콘텐츠를 볼 수 있습니다.

Network Autoconf User 및 Network Autoconf Admin이라는 추가적인 권한 프로파일 두 개가 제공됩니다. Network Autoconf User 프로파일에는 `read`, `select` 및 `wlan` 권한 부여가 있습니다. Network Autoconf Admin 프로파일은 `write` 권한을 추가합니다. Network Autoconf User 프로파일이 Console User 프로파일에 지정됩니다. 콘솔에 로그인한 누구든지 기본적으로 프로파일을 확인하고 사용 및 사용 안함으로 설정할 수 있습니다. Console User 프로파일은 `solaris.network.autoconf.write` 권한 부여에 지정되지 않기 때문에 이 권한이 부여된 사용자가 NCP, NCU, 위치 또는 ENM을 만들거나 수정할 수 없습니다. 하지만 Console User 프로파일은 WLAN을 확인, 만들기 및 수정할 수 있습니다.

netcfg 및 netadm 명령을 사용하여 Basic Solaris User 권한 프로파일을 보유한 모든 사용자가 네트워크 프로파일을 볼 수 있습니다. 이 프로파일은 기본적으로 모든 사용자에게 지정됩니다.

netadm 명령을 사용하여 Network Autoconf User 또는 Console User 프로파일을 보유한 모든 사용자가 프로파일을 사용으로 설정할 수도 있습니다. Console User 프로파일은 `/dev/console`에서 시스템에 로그인한 사용자에게 자동으로 지정됩니다.

netcfg 명령을 사용하여 네트워크 프로파일을 수정하려면 `solaris.network.autoconf.write` 권한 또는 Network Autoconf Admin 프로파일이 필요합니다.

예를 들어, 다음과 같이 Console User 권한 프로파일과 연관된 권한을 확인합니다.

```
$ profiles -p "Console User" info
```

```
name=Console User
desc=Manage System as the Console User
auths=solaris.system.shutdown,solaris.device.cdrw,solaris.devinde.mount.removable,
solaris.smf.manage.vbiosd,solaris.smf.value.vbiosd
profiles=Suspend To RAM,Suspend To Disk,Brightness,CPU
Power Management,Network Autoconf User,Desktop Removable Media User
help=RtConsUser.html
```

## 프로파일 기반 네트워크 구성이 다른 Oracle Solaris 기능에서 작동하는 방식

다음과 같이 반응적 네트워크 구성 모드는 다른 Oracle Solaris 네트워킹 기술에서 작동합니다.

- 가상 시스템: Oracle VM Server for SPARC(이전 Logical Domains) 및 Oracle VM VirtualBox  
반응적 프로파일은 Oracle Solaris 호스트와 게스트에서 모두 작동합니다. 하지만 반응적 네트워크 구성 모드는 시스템의 다른 가상 시스템을 방해하지 않고 지정된 가상 시스템에 속하는 인터페이스만 관리합니다.
- Oracle Solaris 영역 및 스택 인스턴스  
반응적 프로파일은 전역 영역이나 배타적 스택 비전역 영역에서 작동합니다. 하지만 공유 스택 영역에 대한 네트워크 구성은 항상 전역 영역에서 관리되므로 공유 스택 영역에 대해서는 반응적 프로파일을 구성할 수 없습니다.
- DR(동적 재구성)  
시스템의 네트워크 구성은 DR(동적 재구성) 기능 및 핫 플러그 기능을 제공하는 시스템에서만 이러한 기능을 지원합니다. 해당 기능을 사용하면 현재 활성 NCP(반응적 NCP 또는 DefaultFixed NCP)에 관계없이 장치를 추가하거나 제거할 수 있습니다. 하지만 시스템의 동작은 현재 활성 NCP의 유형에 따라 달라집니다.  
Automatic NCP 또는 다른 반응적 NCP가 활성 상태이며 장치가 연결된 경우 자동으로 NCP가 새로 추가된 장치에 대한 IP 구성을 만듭니다. 반응적 프로파일이 현재 활성 상태인 동안 시스템에서 장치가 제거되면 장치에 대한 IP 인터페이스의 구성이 해제됩니다. DefaultFixed NCP가 시스템에서 활성 상태인 경우 장치를 추가한 후 명시적으로 IP 인터페이스를 구성해야 합니다. 또한 장치를 제거하기 전에 명시적으로 IP 구성을 제거해야 합니다.  
장치 동적 구성에 대한 자세한 내용은 “[Oracle Solaris 11.2의 장치 관리](#)”를 참조하십시오. 고정적 프로파일을 사용 중인 경우 동적 재구성을 수행하는 방법은 [동적 재구성을 사용하여 네트워크 인터페이스 카드를 교체하는 방법](#)을 참조하십시오.

# ◆◆◆ 6 장

## Oracle Solaris의 프로파일 기반 네트워크 구성 관리

---

이 장에서는 다양한 유형의 프로파일을 사용하여 네트워크 구성을 관리하는 방법에 대해 설명합니다. 프로파일 기반 네트워크 구성 및 반응적 모드에 대한 개요는 [5장. Oracle Solaris의 프로파일 기반 네트워크 구성 관리 정보](#)를 참조하십시오.

고정적 모드를 사용할 때 데이터 링크 및 IP 인터페이스를 구성하는 것과 관련된 자세한 내용은 [2장. Oracle Solaris의 데이터 링크 구성 관리](#) 및 [3장. Oracle Solaris에서 IP 인터페이스와 주소 구성 및 관리](#)를 참조하십시오.

이 장의 내용:

- “프로파일 사용 및 사용 안함으로 설정” [97]
- “프로파일 구성” [99]
- “프로파일 관리” [111]
- “데스크탑에서 네트워크 구성 관리” [123]

### 프로파일 사용 및 사용 안함으로 설정

netadm 명령을 사용하면 프로파일 유형에 관계없이 또는 프로파일이 고정적인지 아니면 반응적인지에 관계없이 모든 프로파일을 사용 및 사용 안함으로 설정할 수 있습니다. 기본 명령 구문은 다음과 같습니다.

```
# netadm enable [ -p profile-type ] [ -c ncu-class ] profile-name
```

예를 들어, 다음과 같이 시스템 정의 Automatic NCP를 사용으로 설정합니다.

```
# netadm enable -p ncp Automatic
```

프로파일을 사용 및 사용 안함으로 설정하는 것과 관련된 배경 정보는 [“프로파일 기반 네트워크 구성 사용 지침” \[93\]](#)을 참조하십시오.

다양한 유형의 프로파일을 사용 및 사용 안함으로 설정할 때는 다음과 같은 추가 지침을 참조하십시오.

- **NCP** - 항상 시스템에 한 개의 활성 NCP와 한 개의 활성 위치 프로파일이 반드시 있어야 합니다. 활성 NCP는 다른 NCP를 명시적으로 사용으로 설정할 때까지 활성 상태로 유지

됩니다. 다른 NCP를 사용으로 설정하면 현재 활성화 NCP가 암시적으로 사용 안함으로 설정됩니다. 시스템에서 현재 활성화 상태인 NCP를 명시적으로 사용 안함으로 설정할 수는 없습니다. [“NCP에 대한 설명” \[89\]](#)을 참조하십시오.

DefaultFixed NCP를 사용으로 설정하여 고정적 모드로 전환할 때 자동으로 DefaultFixed 위치도 사용으로 설정되므로 변경할 수 없습니다.

Automatic NCP를 사용으로 설정하는 경우 활성화 정책이 현재 네트워크 상태를 기반으로 적합한 해당 위치를 선택한 다음 이 위치를 사용으로 설정합니다.

- **NCU** - NCU에 대한 활성화 모드가 manual로 설정된 경우 현재 활성화 NCP의 일부인 개별 NCU를 수동으로 사용 및 사용 안함으로 설정할 수 있습니다. 링크 또는 NCU 클래스가 지정되지 않으면 두 NCU가 사용 또는 사용 안함으로 설정됩니다. [“NCU에 대한 설명” \[89\]](#)을 참조하십시오.
- **위치** - 기본적으로 시스템에서 사용으로 설정할 최적의 위치 프로파일을 선택합니다. 시스템에서는 활성화 모드가 system 또는 conditional인 위치 세트에서 위치를 선택합니다. 하지만 사용자는 언제든지 위치의 활성화 모드에 관계없이 위치를 수동으로 사용으로 설정하여 시스템의 선택을 대체할 수 있습니다. 위치를 수동으로 사용으로 설정하면 활성화 위치가 자동으로 변경되지 않습니다. 위치 자동 선택은 사용 안함으로 설정되어 있습니다. 시스템에 의한 조건부 위치 선택을 복원하려면 수동으로 사용으로 설정된 위치를 명시적으로 사용 안함으로 설정해야 합니다. [“위치 프로파일에 대한 설명” \[90\]](#)을 참조하십시오.
- **ENM** - 해당 프로파일에는 manual 또는 conditional 활성화 모드가 사용될 수 있습니다. activation-mode 등록 정보를 conditional로 설정하면 지정된 조건에 따라 ENM이 사용 또는 사용 안함으로 설정됩니다. 활성화 모드를 manual로 설정하면 netadm 명령을 사용하여 ENM을 사용 또는 사용 안함으로 설정할 수 있습니다. ENM 활성화에 대한 제약 조건은 없습니다. 항상 0개 이상의 ENM이 시스템에서 활성화 상태일 수 있습니다. ENM을 사용 또는 사용 안함으로 설정해도 현재 활성화 상태인 다른 ENM에는 영향을 주지 않습니다. [“프로파일 활성화 모드” \[94\]](#)를 참조하십시오.

**예 6-1** NCP 사용

다음 예에서는 myncp라는 사용자 정의 NCP가 사용으로 설정됩니다.

```
$ netadm enable -p ncp myncp
Enabling ncp 'myncp'
```

**예 6-2** 위치 프로파일 사용

다음 예에서는 office라는 위치 프로파일이 사용으로 설정됩니다.

```
$ netadm enable -p loc office
Enabling loc 'office'
```

프로파일 이름을 지정할 때 netadm 명령은 대소문자를 구분하지 않습니다.

**예 6-3** 링크 NCU 사용 안함으로 설정

다음 예에서는 net1이라는 링크 NCU가 사용 안함으로 설정됩니다.

```
$ netadm disable -p ncu -c phys net1
```

**예 6-4** 위치 사용 안함

다음 예에서는 office라는 위치가 사용 안함으로 설정됩니다.

```
$ netadm disable -p loc office
Disabling loc 'office'
```

**예 6-5** ENM 사용 및 사용 안함

다음 예에서는 ENM test-enm1이 사용으로 설정되고 다른 ENM test-enm2가 사용 안함으로 설정됩니다.

```
$ netadm enable -p enm test-enm1
Enabling enm 'test-enm1'
$ netadm disable -p enm test-enm2
Disabling enm 'test-enm2'
```

**예 6-6** 고정적 모드와 반응적 모드 간 전환

다음 예에서는 시스템 정의 DefaultFixed NCP를 사용으로 설정하여 고정적 모드로 전환하는 방법을 보여줍니다.

```
$ netadm enable -p ncp DefaultFixed
Enabling ncp 'DefaultFixed'
```

## 프로파일 구성

다음 방법으로 netcfg 명령을 사용하여 반응적 프로파일을 구성할 수 있습니다.

- 대화식
- 명령줄 모드
- 명령 파일 모드

### netcfg 대화식 모드로 작업

대화식으로 사용되는 경우 netcfg 명령에 범위라는 개념이 사용됩니다. 대화식으로 명령을 사용하는 경우 지정된 한 시점의 해당 범위는 프로파일 유형과 수행 중인 특정 작업에 따라

달라집니다. 다음 예에서와 같이 터미널 창에 netcfg 명령을 단독으로 입력하면 전역 범위에서 프롬프트가 표시됩니다.

```
$ netcfg
netcfg>
```

프로파일을 만들거나 선택하려면 먼저 netcfg 대화식 세션을 시작해야 합니다.

전역 범위 프롬프트에서 select 또는 create 하위 명령을 사용하여 최상위 레벨 프로파일인 다음 프로파일 유형을 확인, 수정 또는 생성할 수 있습니다.

- NCP
- 위치
- ENM
- 알려진 WLAN

프로파일을 만들거나 선택한 후 netcfg 대화식 프롬프트의 구문은 다음 예와 유사하게 표시됩니다.

```
netcfg:object-type:object-name>
```

대화식 모드로 netcfg 명령을 사용하여 다음 작업을 수행할 수 있습니다.

- 프로파일을 만듭니다.
- 프로파일을 선택하고 수정합니다.
- 프로파일에 대한 모든 필수 정보가 설정되고 유효한지 확인합니다.
- 새 프로파일에 대한 변경 사항을 커밋합니다.
- 변경 사항을 지속 저장소에 커밋하지 않고 현재 프로파일 구성을 취소합니다.
- 프로파일에 대한 변경 사항을 되돌립니다.

위치 프로파일 및 ENM의 경우 netcfg 대화식 모드로 작업하는 동안 최상위 레벨 프로파일을 선택하거나 만들면 다음 예에서와 같이 프로파일 범위에서 명령 프롬프트가 표시됩니다.

```
netcfg> select loc test-loc
netcfg:loc:test-loc>
```

NCP를 선택하면 NCP 범위에서 명령 프롬프트가 표시됩니다. 이 범위에서 NCU가 선택되고 만들어집니다. NCU를 선택하거나 만들면 선택된 NCP에 대한 프로파일 범위로 세션이 이동됩니다. 이 범위에서 현재 선택된 프로파일과 연결된 모든 등록 정보를 보고 설정할 수 있습니다.

다음 예에서는 office NCP가 선택되어 대화식 세션이 NCP에 대한 NCP 범위로 이동되고 여기서 NCU가 선택됩니다. 이 작업을 통해 선택된 NCU에 대한 프로파일 범위가 생성됩니다. 이 범위에서 NCU의 등록 정보를 보거나 설정할 수 있습니다.

```
$ netcfg
netcfg> select ncp office
netcfg:ncp:office> select ncu phys net2
netcfg:ncp:office:ncu:net2>
```

지정된 모든 범위에서 명령 프롬프트는 현재 선택한 프로파일을 나타냅니다. 이 범위에서 프로파일을 변경하는 경우 변경 사항을 커밋할 수 있습니다. 그러면 변경 사항이 지속 저장소

에 저장됩니다. 범위를 종료할 때 암시적으로 변경 사항이 커밋됩니다. 변경 사항을 커밋하지 않으려는 경우 해당 프로파일에 대해 이전에 커밋된 상태로 되돌릴 수 있습니다. 그러면 해당 레벨에서 프로파일에 대한 모든 변경 사항이 되돌려집니다. `revert` 및 `cancel` 하위 명령은 유사하게 작동합니다.

지침은 “NCP 만들기” [102]를 참조하십시오.

---

참고 - `walkprop` 하위 명령은 개별적으로 프로파일과 연관된 각 등록 정보를 표시하는 데 사용됩니다. 이 명령은 대화식 모드로 사용되는 경우에만 의미가 있습니다.

---

## netcfg 명령줄 모드로 작업

명령줄 모드에서는 선택한 프로파일이나 등록 정보에 영향을 주는 `netcfg` 하위 명령을 선택한 프로파일이나 등록 정보가 있는 특정 범위에서 수행해야 합니다. 명령줄 모드에서의 프로파일 선택은 대화식 모드에서의 선택과 동일합니다. 명령줄 모드의 경우 모든 하위 명령이 명령줄에 기록된다는 점만 다릅니다. 예를 들어, NCP의 콘텐츠를 나열하려면 다음 예에서와 같이 먼저 NCP를 선택한 다음 `list` 하위 명령을 사용하여 해당 NCP에 대한 NCU를 표시합니다.

```
$ netcfg "select ncp myncp; list"
ncp:myncp
management-type reactive
NCUs:
phys net0
phys net1
ip net0
ip net1
```

이 모드로 `netcfg` 명령을 사용할 때는 다음 지침을 참조하십시오.

- 각 하위 명령을 세미콜론으로 구분합니다.
- NCP 범위로 이동하려면 전역 범위에서 `select` 하위 명령을 지정합니다.
- 해당 범위 내 등록 정보를 나열하려면 NCP 범위에서 `list` 하위 명령을 지정합니다.
- 셸이 세미콜론을 해석하지 않도록 하려면 곧은 따옴표를 사용합니다.

---

참고 - 명령줄 모드에서는 전체 명령을 한 행에 입력해야 합니다. 명령줄 모드로 `netcfg` 명령을 사용하여 선택한 프로파일을 변경하는 경우 명령이 실행될 때 변경 사항이 지속 저장소에 커밋됩니다.

명령줄 모드로 `walkprop` 하위 명령을 제외한 모든 `netcfg` 하위 명령을 사용할 수 있습니다.

---

## netcfg 명령 파일 모드로 작업

명령 파일 모드에서는 파일에서 프로파일 구성 정보 및 명령이 추출됩니다. 파일의 명령은 대화식 모드로 사용되는 명령 및 `export` 하위 명령이 제공하는 명령과 동일합니다. `-f` 옵션이

포함된 export 하위 명령은 파일을 생성하는 데 사용됩니다. 예를 들어, 다음 명령은 현재 프로파일 구성을 파일로 내보냅니다.

```
$ netcfg export -f /tmp/nwam.config
```

파일에서 프로파일 구성을 가져오려면 다음 명령을 입력하십시오.

```
$ netcfg -f /tmp/nwam.config
```

대화식으로 export 하위 명령을 사용할 수도 있습니다. 자세한 내용은 “[프로파일 구성 내보내기](#)” [121]를 참조하십시오.

## NCP 만들기

NCP는 시스템의 네트워크 구성을 정의합니다. 고정적 NCP가 아닌 반응적 NCP만 만들 수 있습니다.

대화식 모드로 NCP를 만들려면 먼저 대화식 세션을 시작합니다. 그런 후 다음 예에서와 같이 create 하위 명령을 사용하여 새 NCP를 만듭니다.

```
$ netcfg
netcfg> create ncp myncp
netcfg:ncp:myncp>
```

### 예 6-7 Automatic NCP를 복제하여 NCP 만들기

DefaultFixed NCP 이외의 다른 기존 NCP를 복제하여 NCP를 만들 수 있습니다. 그런 다음 NCP 등록 정보를 수정하여 새 구성 매개변수를 지정할 수 있습니다. 다음 예에서는 시스템 정의 Automatic NCP를 복제하여 newncp라는 새 NCP를 만듭니다.

```
$ netcfg
netcfg> create -t Automatic ncp newncp
netcfg:ncp:newncp> list
ncp:newncp
  management-type reactive
NCUs:
  phys net0
  phys net1
  ip net0
  ip net1
```

management-type 등록 정보는 읽기 전용 등록 정보로, 항상 reactive로 설정됩니다. 이전 예에서는 list 하위 명령이 새로 복사된 NCP(newncp)의 콘텐츠를 표시하는 데 사용됩니다.

또한 다음 예에서와 같이 netcfg 명령줄 모드를 사용하여 직접 Automatic NCP를 복제할 수 있습니다. 대화식 방법과 명령줄 방법은 Automatic NCP에서 새로 만들어진 NCP로 기존 NCU를 복사합니다.

```
$ netcfg create -t Automatic ncp newncp
$ netcfg list ncp newncp
```

```
ncp:newncp
management-type reactive
NCUs:
phys net0
phys net1
ip net0
ip net1
```

## NCP의 NCU 만들기

NCP는 실질적으로 NCP에 대한 네트워크 구성을 정의하는 등록 정보를 통해 구성된 일련의 NCU로 이루어진 컨테이너입니다. 모든 NCP에 링크와 인터페이스 NCU가 모두 포함됩니다.

링크 NCU는 NCP에 대한 링크 구성과 링크 선택 정책을 모두 지정합니다. 인터페이스 NCU는 NCP에 대한 인터페이스 구성 정책을 지정합니다. IP 연결이 필요한 경우 링크 및 인터페이스 NCU가 모두 필요합니다. `netcfg` 명령을 사용하거나 네트워크 관리 GUI를 사용하여 명시적으로 NCU를 추가 또는 제거해야 합니다. 네트워크 관리 GUI를 사용하여 NCU를 추가 및 제거하는 것과 관련된 자세한 내용은 [“데스크탑에서 네트워크 구성 관리” \[123\]](#)를 참조하십시오.

대화식 또는 명령줄 모드로 `netcfg` 명령을 사용하여 NCU를 만듭니다. NCU를 만드는 과정에는 여러 작업이 포함되므로 NCU 및 모든 등록 정보를 만드는 한 행의 명령을 생성하는 대신 대화식으로 NCU를 만드는 것이 더 쉽고 효율적입니다. NCU는 처음에 NCP를 만들 때나 그 이후에 생성될 수 있습니다. NCU를 만들거나 수정하는 프로세스에는 일반 NCU 등록 정보를 설정하는 작업과 구체적으로 각 NCU 유형에 적용되는 등록 정보를 설정하는 작업이 포함됩니다.

대화식으로 NCU를 만드는 경우 `netcfg` 명령은 NCU에 대한 각 관련 등록 정보를 검토하고 기본값이 있는 경우 기본값과 가능한 값을 모두 표시합니다. 예를 들어, 인터페이스 NCU의 `ipv4-addrsrc` 등록 정보에 대한 `dhcp`를 지정하는 경우 `ipv4-addr` 등록 정보에 대한 값을 지정하라는 메시지가 표시되지 않습니다. 이 등록 정보는 정적 IP 주소 구성 용도로만 사용되기 때문입니다. 대화식 프롬프트에서 각 등록 정보에 대한 대체 값을 지정할 수 있습니다. 값을 지정하지 않고 Return 키를 누르면 기본값이 재적용되거나 등록 정보가 빈 상태로 유지됩니다(기본값이 없을 경우).

NCU를 만들거나 수정할 때 지정할 수 있는 여러 가지 NCU 등록 정보가 있습니다. 모든 NCU 유형에 적용되는 등록 정보도 있고, 링크 NCU 또는 인터페이스 NCU에 적용되는 등록 정보도 있습니다. 해당 등록 정보를 지정할 때 적용할 수 있는 규칙 및 조건을 비롯하여 모든 NCU 등록 정보에 대한 전체 설명은 [netcfg\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

### ▼ 대화식으로 NCP에 대한 NCU를 만드는 방법

다음 절차에서는 기존 NCP를 선택한 다음 대화식으로 NCP에 대한 NCU를 만드는 방법을 설명합니다.

---

**참고** - 초기 프로파일을 만드는 동안 수행되는 "검토" 프로세스는 NCP를 수정할 때 생성 중 선택된 항목을 고려하여 적용 가능한 등록 정보에 대해서만 확인 메시지가 표시되도록 합니다.

---

1. **netcfg 대화식 세션을 시작합니다.**

```
$ netcfg
netcfg>
```

2. **기존 NCP를 선택합니다.**

다음 예에서는 NCP myncp가 선택됩니다.

```
netcfg> select ncp myncp
netcfg:ncp:myncp>
```

NCP를 선택하면 자동으로 해당 NCP의 범위로 이동됩니다. 위치, ENM 또는 WLAN 객체의 경우 명령 프롬프트에서 해당 프로파일에 대한 프로파일 범위로 이동됩니다.

3. **NCP의 링크 및 인터페이스 NCU를 만듭니다.**

다음 예에서는 링크 NCU가 만들어집니다.

```
netcfg:ncp:myncp> create ncu phys net0
Created ncu `net0'. Walking properties ...
activation-mode (manual) [manual|prioritized]>
mac-address>
autopush>
mtu> 1600
```

여기서 ncu는 객체 유형이며, phys는 NCU의 클래스이고, net0은 객체 이름입니다.

NCU를 만들면 객체 범위로 이동되고 객체의 기본 등록 정보가 검토됩니다.

이 예에서는 다음 등록 정보가 지정됩니다.

- Return 키를 누르면 activation-mode 등록 정보(기본값: manual)가 수락됩니다.
- mac-address 및 autopush 등록 정보는 공백으로 유지됩니다.
- mtu 등록 정보 값은 1600으로 설정됩니다.

다음 예에서는 인터페이스 NCU를 만드는 방법을 보여줍니다.

```
netcfg:ncp:myncp> create ncu ip net0
Created ncu `net0'. Walking properties ...
ip-version (ipv4,ipv6) [ipv4|ipv6]> ipv4
ipv4-addrsrc (dhcp) [dhcp|static]> dhcp
ipv4-default-route>
```

여기서 ncu는 객체 유형이며, ip는 객체 클래스이고, net0은 객체 이름입니다.

NCU를 만들면 객체 범위로 이동되고 객체의 기본 등록 정보가 검토됩니다.

이 예에서는 다음 등록 정보가 지정됩니다.

- ip-version 등록 정보는 ipv4로 설정됩니다.
- ipv4-addrsrc 등록 정보는 dhcp로 설정됩니다.

NCU를 만드는 동안 class 옵션을 사용하여 두 가지 NCU 유형을 구별합니다.

#### 4. (옵션) 다음과 같이 구성이 올바른지 확인합니다.

```
netcfg:ncp:myncp:ncu:net0> verify
All properties verified
```

verify 하위 명령이 구성을 확인하고 필수 값이 누락되었는지 여부를 알립니다.

#### 5. 만들어진 각 NCU를 저장합니다.

- commit 하위 명령을 사용합니다.

```
netcfg:ncp:myncp:ncu:net0> commit
Committed changes
netcfg:ncp:myncp:ncu:net0>
```

commit 하위 명령은 암시적으로 등록 정보를 확인합니다.

- end 하위 명령을 사용합니다.

```
netcfg:ncp:myncp:ncu:net0> end
Committed changes
netcfg:ncp:myncp>
```

end 하위 명령은 암시적으로 변경 사항을 커밋합니다.

이 경우 NCP에 NCU를 추가하는 작업이 완료되면 end 하위 명령이 NCP 범위로 세션을 이동합니다.

대화식 모드에서는 커밋할 때까지 변경 사항이 지속 저장소에 저장되지 않습니다. commit 하위 명령을 사용하면 전체 프로파일이 커밋됩니다. 지속 저장소의 일관성을 유지하기 위해 commit 작업에는 확인 단계도 포함됩니다. 확인이 실패하면 commit도 실패합니다. 암시적 커밋이 실패할 경우 현재 변경 사항을 커밋하지 않고 대화식 세션을 끝내거나 종료하는 옵션이 제공됩니다. 또는 현재 범위를 유지하고 프로파일 변경을 계속할 수 있습니다.

---

참고 - 변경 사항을 취소하려면 cancel 또는 revert 하위 명령을 사용합니다.

cancel 하위 명령은 현재 변경 사항을 지속 저장소에 커밋하지 않고 현재 프로파일 구성을 끝낸 다음 대화식 세션을 한 레벨 위의 범위로 이동합니다. revert 하위 명령은 변경 사항을 실행 취소하고 이전 구성을 다시 읽습니다. revert 하위 명령을 사용하는 경우 대화식 세션이 동일한 범위로 유지됩니다.

---

#### 6. 완료되면 대화식 세션을 종료합니다.

```
netcfg:ncp:myncp> exit
```

exit 하위 명령은 end 하위 명령과 유사하지만 대화식 세션을 종료합니다.

## 위치 만들기

위치 프로파일에는 기본 링크 및 IP 연결과 직접 관련이 없는 네트워크 구성 값을 정의하는 등록 정보가 포함됩니다. 일부 예에는 필요한 경우 함께 적용되는 IP 필터 설정과 이름 지정 서비스가 포함되어 있습니다. 시스템에서는 항상 정확히 한 개의 위치 프로파일과 한 개의 NCP가 활성 상태여야 합니다.

대화식 또는 명령줄 모드로 netcfg 명령을 사용하여 위치 프로파일을 만듭니다. 위치 프로파일을 만들 때는 해당 특정 위치가 사용으로 설정되는 경우 특정 구성 매개변수를 정의하는 값을 지정하여 위치의 등록 정보를 설정합니다. 위치 등록 정보는 그룹별로 분류되며, 여기서 그룹은 구성 기본 설정의 특정 클래스를 나타냅니다.

또한 위치 등록 정보는 저장소에 저장됩니다. 특정 위치가 사용으로 설정되면 해당 등록 정보가 자동으로 실행 중인 시스템에 적용됩니다. 위치 프로파일을 만들고 수정하는 과정에서는 특정 위치가 사용으로 설정되는 시기를 정의하는 등록 정보를 설정합니다. 구성 프로세스에서 제공되는 등록 정보는 이전에 설정된 등록 정보 값을 기반으로 합니다.

해당 등록 정보를 지정할 때 적용할 수 있는 규칙, 조건 및 종속성을 비롯하여 모든 위치 프로파일 등록 정보에 대한 전체 설명은 [netcfg\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

### ▼ 대화식으로 위치를 만드는 방법

다음 절차에서는 위치 프로파일을 만드는 방법에 대해 설명합니다. 초기 프로파일을 만드는 동안 수행되는 "검토" 프로세스는 이전에 입력된 값을 고려하여 적용 가능한 등록 정보에 대해서만 확인 메시지를 표시합니다.

#### 1. netcfg 대화식 세션을 시작합니다.

```
$ netcfg
netcfg>
```

#### 2. 위치 프로파일을 만듭니다.

다음 예에서는 office라는 위치가 만들어집니다.

```
netcfg> create loc office
Created loc 'office'. Walking properties ...
activation-mode (manual) [manual|conditional-any|conditional-all]> conditional-any
conditions> ncu ip:net0 is active
nameservices (dns) [dns|files|nis|ldap]>
nameservices-config-file ("/etc/nsswitch.dns")>
dns-nameservice-configsrc (dhcp) [manual|dhcp]>
nfsv4-domain>
```

```

ipfilter-config-file> /export/home/test/wifi.ipf.conf
ipfilter-v6-config-file>
ipnat-config-file>
ippool-config-file>
ike-config-file> /etc/inet/ike/ikev1.config
ikev2-config-file>
ipsecpolicy-config-file>

```

위치를 만들면 자동으로 해당 위치의 프로파일 범위로 이동됩니다.

이 예에서는 다음 등록 정보가 설정되었습니다.

- activation-mode 등록 정보가 conditional-any로 설정되었습니다. 이로 인해 활성화 조건을 사용으로 설정한 명령 프롬프트가 지정되었습니다.
- 위치에 대한 활성화 조건이 ncu ip:net0 is active로 지정되었습니다.
- ipfilter-config-file 등록 정보에 대해 /export/home/test/wifi.ipf.conf 파일이 지정되었습니다.
- ike-config-file 등록 정보에 대해 /etc/inet/ike/ikev1.config 파일이 지정되었습니다.
- Return 키를 눌러 나머지 등록 정보에 대한 기본값이 수락되었습니다.

### 3. (옵션) 다음과 같이 list 하위 명령을 사용하여 프로파일 구성을 표시합니다.

```

netcfg:loc:office> list
loc:office
      activation-mode          conditional-any
      conditions                "ncu ip:net0 is active"
      enabled                   false
      nameservices              dns
      nameservices-config-file  "/etc/nsswitch.dns"
      dns-nameservice-configsrc dhcp
      ipfilter-config-file      "/export/home/test/wifi.ipf.conf"
      ike-config-file           "/etc/inet/ike/ikev1.config"

```

### 4. 프로파일 구성이 올바른지 확인합니다.

```

netcfg:loc:office> verify
All properties verified

```

verify 하위 명령이 구성을 확인하고 필수 값이 누락되었는지 여부를 알립니다.

### 5. 구성 확인이 완료되면 위치를 저장합니다.

```

netcfg:loc:office> commit
Committed changes

```

또는 end 하위 명령을 사용하여 세션을 끝낼 수 있습니다. 이 경우 프로파일 구성도 저장되며 세션이 전역 범위로 이동됩니다.

```

netcfg:loc:office> end
Committed changes
netcfg>

```

6. 대화식 세션을 종료합니다.

```
netcfg> exit
```

## ENM 만들기

ENM을 통해 응용 프로그램 또는 스크립트(예: VPN 응용 프로그램)가 NCP 및 위치 프로파일에서 지정된 구성과 무관하게 네트워크 구성을 수행할 시기를 지정할 수 있습니다. ENM에 대한 자세한 내용은 “ENM에 대한 설명” [91]을 참조하십시오.

---

**참고** - ENM을 만들 수 있는 응용 프로그램은 자동으로 인식되지 않습니다. `netcfg` 명령을 사용하여 관련 ENM을 만들기 전에 먼저 시스템에서 해당 응용 프로그램을 설치하고 구성해야 합니다.

---

ENM을 만들 때 지정할 수 있는 등록 정보에 대한 자세한 내용은 `netcfg(1M)` 매뉴얼 페이지를 참조하십시오.

### ▼ 대화식으로 ENM을 만드는 방법

1. `netcfg` 대화식 세션을 시작합니다.

```
$ netcfg
netcfg>
```

2. ENM을 만듭니다.

```
netcfg> create enm test-enm
Created enm 'test-enm'. Walking properties ...
activation-mode (manual) [manual|conditional-any|conditional-all]>
fmri> svc:/application/test-enm:default
start>
stop>
netcfg:enm:test-enm>
```

ENM을 만들면 자동으로 ENM의 프로파일 범위로 이동되고 해당 등록 정보가 각각 표시됩니다.

이 예에서는 `test-enm` ENM에 대해 다음 등록 정보가 지정됩니다.

- Return 키를 누르면 `manual`로 설정된 `activation-mode` 등록 정보가 수락됩니다. 이 값은 `manual`로 설정되어 있으므로 `conditions` 등록 정보를 설정에 사용할 수 없습니다.
- `fmri` 등록 정보가 `svc:/application/test-enm:default`로 설정됩니다.
- 이 ENM에 대해서는 `start` 및 `stop` 등록 정보가 설정되지 않습니다.

3. (옵션) 프로파일 구성을 표시합니다.

```
netcfg:enm:test-enm> list
enm:test-enm
  activation-mode manual
  enabled false
  fmri "svc:/application/test-enm:default"
```

#### 4. 프로파일 구성이 올바른지 확인합니다.

```
netcfg:enm:test-enm> verify
All properties verified
```

verify 하위 명령이 구성을 확인하고 필수 값이 누락되었는지 여부를 알립니다.

#### 5. ENM을 저장합니다.

```
netcfg:enm:test-enm> commit
Committed changes
netcfg>
```

commit 하위 명령이 구성을 확인하고 저장합니다.

또는 end 하위 명령을 사용하여 세션을 끝낼 수 있습니다. 이 경우 프로파일 구성도 저장됩니다.

```
netcfg:enm:test-enm> end
Committed changes
```

#### 6. 대화식 세션을 종료합니다.

```
netcfg> exit
```

## 알려진 WLAN 만들기

알려진 WLAN은 시스템에 알려진 무선 네트워크에 대한 정보를 저장하는 프로파일입니다. NCP는 시스템이 연결된 각 무선 네트워크가 제공하는 구성 정보를 기반으로 무선 인터페이스를 자동으로 구성할 수 있습니다. 자세한 내용은 [“알려진 WLAN에 대한 설명” \[92\]](#)을 참조하십시오.

WLAN을 만들거나 수정할 때 지정할 수 있는 등록 정보에 대한 자세한 내용은 [netcfg\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

### ▼ 대화식으로 알려진 WLAN을 만드는 방법

#### 1. netcfg 대화식 세션을 시작합니다.

```
$ netcfg
netcfg>
```

## 2. 알려진 WLAN을 만듭니다.

다음 예에서는 ESSID라는 무선 네트워크에 연결되는 알려진 WLAN을 만듭니다. 알려진 WLAN의 이름은 무선 네트워크 이름 또는 해당 ESSID와 동일해야 합니다.

```
netcfg> create wlan mywifi
Created wlan 'mywifi'. Walking properties ...
priority (0)> 100
bssids>
keyname> mywifi-key
keyslot>
security-mode [none|wep|wpa]> wpa
netcfg:wlan:mywifi>
```

WLAN을 만들면 자동으로 WLAN의 프로파일 범위로 이동되고 해당 등록 정보가 각각 표시됩니다.

이 예에서는 mywifi 알려진 WLAN에 대해 다음 등록 정보가 지정됩니다.

- priority 등록 정보 값이 기본값인 0에서 100으로 변경됩니다.
- keyname 등록 정보가 mywifi-key로 설정되고 이 무선 네트워크에 대한 보안 객체의 이름을 지정합니다. 자세한 내용은 [“보안 WiFi 통신 설정” \[130\]](#)을 참조하십시오.
- security-mode 등록 정보가 wpa로 설정됩니다. 이 등록 정보는 해당 무선 네트워크에 사용되는 암호화의 유형을 지정합니다.
- keyslot 및 bssid 등록 정보 값이 공백으로 유지됩니다.

## 3. (옵션) 프로파일 구성을 표시합니다.

```
netcfg:wlan:mywifi> list
known wlan:mywifi
priority 100
keyname "mywifi-key"
security-mode wpa
netcfg:wlan:mywifi>
```

## 4. 프로파일 구성이 올바른지 확인합니다.

```
netcfg:wlan:mywifi> verify
All properties verified
```

verify 하위 명령이 구성을 확인하고 필수 값이 누락되었는지 여부를 알립니다.

## 5. 알려진 WLAN을 저장합니다.

```
netcfg:wlan:mywifi> commit
Committed changes
```

commit 하위 명령이 구성을 확인하고 저장합니다.

또는 end 하위 명령을 사용하여 세션을 끝낼 수 있습니다. 이 경우 프로파일 구성도 저장되며 세션이 전역 범위로 이동됩니다.

```
netcfg:wlan:mywifi> end
```

```
Committed changes
netcfg>
```

## 6. 대화식 세션을 종료합니다.

```
netcfg> exit
```

## 프로파일 관리

이 절은 다음 항목으로 구성되어 있습니다.

- “프로파일에 대한 등록 정보 값 설정” [111]
- “프로파일 구성에 대한 정보 가져오기” [113]
- “walkprop 하위 명령을 사용하여 프로파일에 대한 등록 정보 값 설정” [116]
- “프로파일에 대한 정보 표시” [118]
- “프로파일 제거” [120]
- “프로파일 구성 내보내기” [121]
- “내보낸 프로파일 구성 복원” [123]

## 프로파일에 대한 등록 정보 값 설정

반응적 프로파일에 대한 등록 정보 값을 설정 또는 수정하려면 set 하위 명령을 사용합니다. 이 하위 명령은 대화식 또는 명령줄 모드로 사용할 수 있습니다. 명령줄 모드에서 등록 정보 값을 설정하거나 변경하면 변경 사항이 지속 저장소에 즉시 커밋됩니다.

---

**참고** - DefaultFixed NCP 또는 DefaultFixed 위치 프로파일은 set 하위 명령을 통해 수정할 수 없습니다. DefaultFixed NCP가 활성화 상태인 경우 항상 dladm 및 ipadm 명령을 사용하여 구성을 변경하십시오. DefaultFixed 위치가 활성화 상태인 경우 svccfg 및 svcadm 명령을 사용하여 직접 관련 SMF 등록 정보를 변경하십시오. [svccfg\(1M\)](#) 및 [svcadm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

---

set 하위 명령의 구문은 다음과 같습니다.

```
netcfg> set prop-name=value1[,value2,...]
```

### ▼ 대화식으로 프로파일 등록 정보 값을 설정하는 방법

다음 절차에서는 대화식으로 위치 프로파일에 대한 등록 정보 값을 설정하는 방법을 설명합니다. 대화식으로 등록 정보 값을 설정하는 경우 먼저 현재 범위에서 프로파일을 선택해야 합니다. 이 경우 대화식 세션이 해당 프로파일의 범위로 이동됩니다. 선택한 프로파일이 지속

저장소에서 메모리로 로드됩니다. 그러면 이 범위에서 프로파일의 등록 정보를 수정할 수 있습니다.

다음 절차에서는 대화식으로 test-loc 위치의 ipfilter-config-file 등록 정보를 설정하는 방법을 보여줍니다(예로만 사용됨).

1. **netcfg 대화식 세션을 시작합니다.**

```
$ netcfg
netcfg>
```

2. **수정할 프로파일 또는 구성 객체를 선택합니다.**

```
netcfg> select loc test-loc
netcfg:loc:test-loc>
```

3. **등록 정보 값을 설정합니다.**

다음 예에서는 ipfilter-config-file 등록 정보가 설정됩니다.

```
netcfg:loc:test-loc> set ipfilter-config-file = /path/to/ipf-file
```

4. **(옵션) 구성 정보를 나열합니다.**

```
netcfg:loc:test-loc> list
loc:test-loc
activation-mode    manual
enabled            false
nameservices      dns
dns-nameservice-configsrc  dhcp
nameservices-config-file  "/etc/nsswitch.dns"
ipfilter-config-file  "/path/to/ipf-file"
```

5. **세션을 끝냅니다.**

```
netcfg:loc:test-loc> end
Committed changes
netcfg>
```

end 하위 명령이 세션을 저장하고 전역 범위로 이동합니다.

6. **대화식 세션을 종료합니다.**

```
netcfg> exit
```

**예 6-8** 명령줄 모드로 프로파일에 대한 등록 정보 값 설정

대화식으로 ipfilter-config-file 등록 정보를 설정하는 방법을 보여주는 이전 예를 다음과 같이 명령줄 모드로 수행할 수도 있습니다.

```
$ netcfg "select loc test-loc; set ipfilter-config-file = /path/to/ipf-file"
```

간단한 작업만 수행하면 되는 경우 명령줄 모드가 가장 적합합니다. 하지만 명령줄 모드를 사용하여 보다 복잡한 작업을 수행할 수도 있습니다. 명령줄 모드에서 신중하게 적합한 하위 명령을 지정하면 됩니다. 대화식 예와 마찬가지로 명령줄 모드에서도 먼저 위치를 선택하여 해당 프로파일 범위로 이동해야 합니다. 그런 다음 set 하위 명령을 지정하여 개별 등록 정보 값을 설정할 수 있습니다.

명령줄 모드를 사용할 때 지정된 등록 정보에 대해 동시에 여러 값을 설정할 수 있습니다. 이 방식으로 여러 값을 설정하는 경우 각 값을 쉼표(,)로 구분해야 합니다. 지정한 등록 정보의 개별 값에도 쉼표가 포함되어 있는 경우 등록 정보 값의 일부인 쉼표 앞에 백슬래시(\)를 추가해야 합니다. 단일 값만 포함된 등록 정보 내의 쉼표는 분리자로 해석되지 않으므로 앞에 백슬래시를 추가할 필요가 없습니다.

예를 들어, 다음과 같이 myncp NCP에서 net0 NCU에 IPv4와 IPv6을 모두 사용하도록 ip-version 등록 정보를 설정합니다.

```
$ netcfg "select ncp myncp; select ncu ip net0; set ip-version=ipv4,ipv6"
```

## 프로파일 구성에 대한 정보 가져오기

대화식 또는 명령줄 모드로 list 하위 명령을 사용하여 프로파일, 등록 정보-값 쌍 및 현재 또는 지정된 범위에 있는 리소스를 모두 나열할 수 있습니다.

전역 범위에서 list 하위 명령은 다음 예에서와 같이 시스템의 시스템 정의 및 사용자 정의 프로파일을 모두 나열합니다.

```
$ netcfg list
netcfg list
NCPs:
  DefaultFixed
  Automatic
  myncp
Locations:
  Automatic
  NoNet
  DefaultFixed
  office
ENMs:
  test-enm
WLANS:
  mywifi
```

---

참고 - 대화식 모드로 전역 범위에서 list 하위 명령을 사용하는 경우에도 동일한 정보가 나열됩니다.

---

list 하위 명령은 각 프로파일의 상태를 나열하지 않습니다. 프로파일 및 해당 상태에 대한 정보를 표시하려면 netadm 명령과 list 하위 명령을 함께 사용하십시오. 자세한 내용은 “[프로파일의 현재 상태 표시](#)” [119]를 참조하십시오.

## 개별 프로파일에 대한 등록 정보 값 나열

프로파일 범위에서 `list` 하위 명령은 지정된 프로파일에 대한 등록 정보 값을 모두 나열합니다. 구문은 다음과 같습니다.

```
netcfg> list [ object-type [ class ] object-name ]
```

### ▼ 대화식으로 프로파일에 대한 등록 정보 값을 모두 나열하는 방법

다음 절차에서는 대화식으로 `list` 하위 명령을 사용하여 프로파일에 대한 등록 정보 값을 모두 나열하는 방법을 설명합니다. 다음 절차의 예에서는 IP NCP 이름 `net0`에 대한 구성 정보를 나열하는 방법을 보여줍니다. 각 프로파일에 대해 나열되는 값은 프로파일에 따라 다릅니다.

1. **netcfg** 대화식 세션을 시작합니다.

```
$ netcfg
netcfg>
```

2. **NCP**를 선택합니다.

```
netcfg> select ncp myncp
netcfg:ncp:myncp>
```

3. 다음 방법 중 하나로 IP NCU에 대한 구성을 나열합니다.

#### ■ 전역 범위에서 구성을 나열합니다.

```
netcfg:ncp:myncp> list ncu ip net0
ncu:net0
      type                interface
      class               ip
      parent              "myncp"
      enabled              false
      ip-version           ipv4,ipv6
      ipv4-addrsrc         dhcp
      ipv6-addrsrc         dhcp,autoconf
netcfg:ncp:myncp>
```

#### ■ 프로파일 범위에서 구성을 나열합니다.

```
netcfg:ncp:myncp> select ncu ip net0
netcfg:ncp:myncp:ncu:net0> list
ncu:net0
      type                interface
      class               ip
      parent              "myncp"
      enabled              false
```

```

ip-version          ipv4, ipv6
ipv4-addrsrc       dhcp
ipv6-addrsrc       dhcp, autoconf
netcfg:ncp:myncp:ncu:net0>

```

#### 4. 대화식 세션을 종료합니다.

```
netcfg:ncp:myncp:ncu:net0> exit
```

#### 예 6-9 명령줄 모드로 등록 정보 값 나열

대화식으로 등록 정보 값을 나열하는 방법을 보여주는 이전 예를 명령줄 모드로도 수행할 수 있습니다. 사용하는 모드에 관계없이 출력은 동일합니다.

예를 들어, 다음과 같이 NCP 범위에서 net0 IP NCU의 등록 정보를 나열합니다.

```
$ netcfg "select ncp myncp; list ncu ip net0"
```

프로파일 범위에서 net0 IP NCU의 등록 정보를 나열하려면 다음 명령을 사용합니다.

```
$ netcfg "select ncp myncp; select ncu ip net0; list"
```

## 프로파일에 대한 특정 등록 정보 값 가져오기

get 하위 명령을 사용하여 프로파일에 대한 특정 등록 정보 값을 가져올 수 있습니다. 이 하위 명령은 대화식 또는 명령줄 모드로 사용할 수 있습니다. 명령 구문은 다음과 같습니다.

```
netcfg> get [ -V ] prop-name
```

### ▼ 프로파일에 대한 특정 등록 정보 값을 가져오는 방법

다음 절차에서는 대화식으로 get 하위 명령을 사용하여 프로파일에 대한 특정 등록 정보 값을 가져오는 방법을 설명합니다. 다음 절차의 예에서는 IP NCU의 ip-version 등록 정보를 가져오는 방법을 보여줍니다.

#### 1. netcfg 대화식 세션을 시작합니다.

```
$ netcfg
netcfg>
```

#### 2. NCP를 선택한 다음 IP NCU를 선택합니다. 예를 들면 다음과 같습니다.

```
netcfg> select ncp myncp
netcfg:ncp:myncp> select ncu ip net0
```

```
netcfg:ncp:myncp:ncu:net0>
```

3. 다음 명령 중 하나를 사용하여 특정 등록 정보 값을 가져옵니다.

- 다음과 같이 **get** 하위 명령을 사용하여 등록 정보 이름 및 등록 정보 값을 표시합니다.

```
netcfg:ncp:myncp:ncu:net0> get ip-version
ip-version ipv4,ipv6
```

- 등록 정보 이름을 표시하지 않고 등록 정보 값만 가져오려면 다음과 같이 **get** 하위 명령에 **-v** 옵션을 사용합니다.

```
netcfg:ncp:myncp:ncu:net0> get -V ip-version
ipv4,ipv6
```

4. 대화식 세션을 종료합니다.

```
netcfg:ncp:myncp:ncu:net0> exit
```

예 6-10 명령줄 모드로 프로파일에 대한 특정 등록 정보 값 가져오기

명령줄 모드로도 이전 대화식 예를 수행할 수 있습니다. 사용하는 모드에 관계없이 출력은 동일합니다.

예를 들어, 다음과 같이 명령줄 모드로 IP NCU에 대한 `ip-version` 등록 정보의 값을 가져옵니다.

```
$ netcfg "select ncp myncp; select ncu ip net0; get ip-version"
ip-version ipv4,ipv6
```

다음 예에서는 `get` 하위 명령에 `-v` 옵션을 사용하여 특정 등록 정보 값을 가져오는 방법을 보여줍니다. 이 방법은 등록 정보 이름의 구문을 분석하지 않아도 되는 스크립트에 유용합니다.

```
$ netcfg "select ncp myncp; select ncu ip net0; get -V ip-version"
ipv4,ipv6
```

## walkprop 하위 명령을 사용하여 프로파일에 대한 등록 정보 값 설정

`walkprop` 하위 명령을 사용하여 대화식으로 프로파일에 대한 개별 등록 정보 값을 확인하고 변경할 수 있습니다. 대화식 세션을 시작한 후 `walkprop` 하위 명령을 입력하면 프로파일의 등록 정보마다 한 번에 하나씩 이름 및 현재 값을 표시할 수 있습니다. 다양한 등록 정보를 확인하면서 필요에 따라 현재 값 또는 기본값을 설정하거나 변경할 수 있습니다.

---

참고 - `walkprop` 하위 명령은 대화식 모드로 사용되는 경우에만 의미가 있습니다.

---

## ▼ 지정된 프로파일에 대한 등록 정보 값을 "검토" 및 설정하는 방법

다음 절차에서는 walkprop 하위 명령을 사용하여 대화식으로 지정된 프로파일에 대한 등록 정보 값을 확인하고 변경하는 방법을 설명합니다. 다음 예에서와 같이 walkprop 하위 명령을 사용하여 프로파일의 등록 정보를 설정할 때는 set 하위 명령을 사용하지 않아도 됩니다.

1. netcfg 대화식 세션을 시작합니다.

```
$ netcfg
netcfg>
```

2. 등록 정보를 확인 및 변경할 프로파일을 선택합니다.

다음 예에서는 test-loc라는 위치가 선택됩니다.

```
netcfg> select loc test-loc
netcfg:loc:test-loc>
```

3. walkprop 하위 명령을 입력하여 검토를 시작합니다.

다음 예에서는 walkprop 하위 명령이 실행된 후 표시되는 첫번째 등록 정보가 activation-mode 등록 정보입니다. 등록 정보 기본값을 살펴봅니다. 이 예에서는 현재 manual(괄호 안에 표시됨)로 설정되어 있습니다.

```
netcfg:loc:test-loc> walkprop
activation-mode (manual) [manual|conditional-any|conditional-all]>
```

4. 등록 정보 값을 수정하려면 대화식 프롬프트에서 새 값을 입력한 다음 Return 키를 누릅니다.

예를 들어, 다음과 같이 위치에 대한 activation-mode 등록 정보를 manual에서 conditional-all로 변경합니다.

```
netcfg:loc:test-loc> walkprop
activation-mode (manual) [manual|conditional-any|conditional-all]> conditional-all
```

Return 키를 누르면 현재 설정이 저장되고 다음 등록 정보가 표시됩니다.

5. 프로파일에 대한 모든 등록 정보가 표시될 때까지 검토 프로세스를 반복하면서 필요에 따라 4단계의 지침대로 수정합니다.

```
netcfg:loc:test-loc> walkprop
activation-mode (manual) [manual|conditional-any|conditional-all]> conditional-all
conditions> advertised-domain is example.com
nameservices (dns) [dns|files|nis|ldap]>
nameservices-config-file ("/etc/nsswitch.dns")>
dns-nameservice-configsrc (dhcp) [manual|dhcp]>
nfsv4-domain>
ipfilter-config-file>
ipfilter-v6-config-file>
ipnat-config-file>
ippool-config-file>
ike-config-file>
```

```
ikev2-config-file>
ipsecpolicy-config-file>
```

등록 정보를 변경하지 않은 상태로 Return 키를 누르면 기존 기본값이 유지되고 다음 등록 정보로 이동됩니다.

---

참고 - [대화식으로 위치를 만드는 방법 \[106\]](#)에 설명된 대로 지정된 프로파일에 대한 관련 등록 정보만 표시됩니다.

---

## 6. 프로파일에 대한 현재 기본 등록 정보 값을 나열합니다. 예를 들면 다음과 같습니다.

```
netcfg:loc:test-loc> list
loc:test-loc
activation-mode    conditional-all
conditions        "advertised-domain is example.com"
enabled           false
nameservices      dns
nameservices-config-file  "/etc/nsswitch.dns"
dns-nameservice-configsrc  dhcp
```

이전 출력의 activation-mode 등록 정보가 conditional-all로 설정되었음을 알 수 있습니다.

## 7. 대화식 세션을 종료하여 변경 사항을 커밋합니다.

```
netcfg:loc:test-loc> exit
Committed changes
```

# 프로파일에 대한 정보 표시

list 하위 명령과 함께 netadm 명령을 사용하여 프로파일별 현재 상태를 비롯하여 시스템에 있는 모든 파일 또는 원하는 파일에 대한 정보를 표시할 수 있습니다.

list 하위 명령의 구문은 다음과 같습니다.

```
$ netadm list [ -p object-type ] [ -c ncu-class ] [ object-name ]
```

다음과 같이 시스템의 모든 프로파일과 프로파일별 현재 상태를 표시합니다.

```
$ netadm list
TYPE  PROFILE  STATE
ncp   DefaultFixed  disabled
ncp   Automatic    online
ncu:phys  net0    online
ncu:phys  net1    offline
ncu:ip    net0    online
ncu:ip    net1    offline
loc    Automatic    online
loc    NoNet       offline
loc    DefaultFixed  offline
```

```
enm      test-enm      disabled
```

`list` 하위 명령은 사용으로 설정된 NCP 및 해당 특정 NCP를 구성하는 모든 NCU를 표시합니다.

## 프로파일의 현재 상태 표시

`list` 하위 명령에 프로파일 유형 및 NCU 클래스를 포함시켜 특정 프로파일을 식별할 수 있습니다. 프로파일 유형만 제공하면 해당 유형의 모든 프로파일이 표시됩니다. 프로파일을 이름으로 지정하면 해당 프로파일의 현재 상태가 표시됩니다. 프로파일 이름이 고유하지 않은 경우 해당 이름을 가진 모든 프로파일이 나열됩니다.

**예 6-11** 지정한 프로파일의 현재 상태 표시

다음 예에서는 시스템에서 이름이 `Automatic`인 모든 프로파일의 현재 상태를 나열합니다.

```
$ netadm list Automatic
TYPE      PROFILE      STATE
ncp       Automatic    online
ncu:ip    net1         offline
ncu:phys  net1         offline
ncu:ip    net0         online
ncu:phys  net0         online
loc       Automatic    online
```

다음 예에서는 `list` 하위 명령에 `-p` 옵션을 사용하여 현재 시스템에 있는 모든 위치를 표시합니다. 이 특정 예에서 `Automatic` 위치는 `online`(사용)입니다.

```
$ netadm list -p loc
TYPE PROFILE STATE
loc DefaultFixed offline
loc NoNet offline
loc Automatic online
```

다음 예에서는 `list` 하위 명령에 `-c` 옵션을 사용하여 활성 NCP의 인터페이스 NCU를 모두 표시합니다.

```
$ netadm list -c ip
TYPE      PROFILE      STATE
ncu:ip    net0         online
ncu:ip    net1         offline
```

## 프로파일의 보조 상태 값 표시

프로파일의 보조 상태는 지정된 프로파일이 `online` 또는 `offline`(사용 또는 사용 안함)인 이유에 대해 설명합니다. 보조 상태 값을 나열하려면 다음과 같이 `list` 하위 명령에 `-x` 옵션을 사용합니다.

```
$ netadm list -x
TYPE          PROFILE          STATE          AUXILIARY STATE
ncp           DefaultFixed    disabled       disabled by administrator
ncp           Automatic        online         active
ncu:phys      net0             online         interface/link is up
ncu:phys      net1             offline        interface/link is down
ncu:ip        net0             online         interface/link is up
ncu:ip        net1             offline        conditions for activation are unmet
loc           Automatic        offline        conditions for activation are unmet
loc           NoNet            offline        conditions for activation are unmet
enm           test-enm        disabled       disabled by administrator
loc           DefaultFixed    offline        conditions for activation are unmet
```

프로파일 유형에 따라 보조 상태 값이 달라집니다. 보조 상태에 대한 자세한 내용은 [nwmad\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## 프로파일 제거

`destroy` 하위 명령으로 사용자 정의 프로파일 및 구성 객체(예: NCU)를 제거할 수 있습니다. `Automatic` 및 `DefaultFixed` NCP와 `Automatic`, `NoNet` 및 `DefaultFixed` 위치를 포함하는 시스템 정의 프로파일은 제거할 수 없습니다. 현재 활성화 프로파일도 제거할 수 없습니다. 먼저 프로파일을 사용 안함으로 설정한 다음 제거해야 합니다.

`destroy` 하위 명령의 구문은 다음과 같습니다.

```
netcfg> destroy [ -a | object-type [ class ] object-name ]
```

`-a` 옵션은 현재 활성화 사용자 정의 프로파일을 제외한 모든 사용자 정의 프로파일을 시스템에서 제거합니다.

### ▼ 대화식으로 프로파일을 제거하는 방법

다음 절차에서는 대화식으로 사용자 정의 프로파일을 제거하는 방법에 대해 설명합니다. 이 절차에서는 `myncp` 사용자 정의 NCP에서 IP NCU를 제거하는 방법을 보여줍니다(예로만 사용됨).

1. **netcfg** 대화식 세션을 시작합니다.

```
$ netcfg
netcfg>
```

2. **프로파일을 선택합니다.**

예를 들어, `myncp` NCP를 선택하려면 다음 명령을 입력합니다.

```
netcfg> select ncp myncp
netcfg:ncp:myncp>
```

### 3. 프로파일 또는 구성 객체를 제거합니다.

다음 예에서는 net1 IP NCU가 myncp NCP에서 제거됩니다.

```
netcfg:ncp:myncp> destroy ncu ip net1
netcfg:ncp:myncp>
```

### 4. 대화식 세션을 종료합니다. 예를 들면 다음과 같습니다.

```
netcfg:ncp:myncp> exit
```

#### 예 6-12 명령줄 모드로 프로파일 제거

대화식으로 프로파일을 제거하는 방법을 보여주는 이전 예를 다음과 같이 명령줄 모드로도 수행할 수 있습니다.

```
$ netcfg "select ncp myncp; destroy ncu ip net1"
```

## 프로파일 구성 내보내기

export 하위 명령을 사용하여 프로파일 구성을 저장할 수 있습니다. 동일한 네트워크 구성이 필요한 여러 서버를 유지 관리해야 하는 경우 프로파일 내보내기가 유용할 수 있습니다. export 하위 명령은 대화식 또는 명령줄 모드로 사용할 수 있습니다. 프로파일을 내보내면 netcfg 명령이 해석할 수 있는 일련의 하위 명령으로 출력이 표시됩니다. 해당 하위 명령은 대화식 또는 명령줄 모드로 입력하는 명령과 유사합니다.

---

**참고** - 일부 구성의 경우 export 기능 사용이 제한됩니다. 처음에 netcfg 명령을 사용하여 만들어진 구성 객체만 내보낼 수 있습니다. dladm 또는 ipadm 명령을 사용하여 만들어진 구성 객체(예: 링크 통합 또는 IPMP 그룹)는 내보낼 수 없습니다. DefaultFixed NCP 및 DefaultFixed 위치도 내보낼 수 없습니다.

---

export 하위 명령의 구문은 다음과 같습니다.

```
netcfg> export [ -d ] [ -f output-file ] [ object-type [ class ] object-name ]
```

---

**참고** - export 하위 명령의 -d 및 -f 옵션은 서로 독립적으로 사용할 수 있습니다. -f 옵션은 현재 또는 지정된 범위의 현재 구성을 지정된 파일에 인쇄합니다. -d 옵션은 출력의 첫번째 행으로 destroy -a 명령을 추가합니다.

---

#### 예 6-13 대화식으로 프로파일 구성 내보내기

다음 예에서는 대화식으로 export 하위 명령을 사용하여 화면에 프로파일 구성을 표시하는 방법을 보여줍니다.

```
$ netcfg
netcfg> export
create ncp "myncp"
create ncu ip "net0"
set ip-version=ipv4
set ipv4-addrsrc=dhcp
set ipv6-addrsrc=dhcp,autoconf
end
create ncu phys "net0"
set activation-mode>manual
set mtu=5000
end
end
create loc "test-loc"
set activation-mode=conditional-all
set conditions="system-domain is example.com"
set nameservices=dns
set nameservices-config-file="/etc/nsswitch.dns"
set dns-nameservice-configsrc=dhcp
end
create enm "test-enm"
set activation-mode=conditional-all
set conditions="ip-address is-not-in-range 10.2.3.4"
set fmri="svc:/application/test-enm:default"
end
create wlan "mywifi"
set priority=100
set keyname="mywifi-key"
set security-mode=wpa
end
```

명령줄 모드로 다음 명령을 입력합니다.

```
$ netcfg export
```

다음 예에서와 같이 export 하위 명령에 -d 옵션을 사용하여 간략하게 나타내기 위해 잘린 netcfg export 출력의 첫번째 행으로 destroy -a 명령을 추가할 수 있습니다.

```
$ netcfg
netcfg> export -d
destroy -a
create ncp "myncp"
create ncu ip "net0"
set ip-version=ipv4
set ipv4-addrsrc=dhcp
.
.
.
```

명령줄 모드로 다음 명령을 입력합니다.

```
$ netcfg export -d
```

**예 6-14** 파일에 프로파일 구성 내보내기

다음 예에서는 export 하위 명령에 -f 옵션을 사용하여 myncp NCP에 대한 구성 정보를 파일에 기록합니다. 다음 예에서 -f 옵션은 myncp2라는 새 파일에 출력을 기록합니다. -d 옵션은 netcfg export 출력의 첫번째 행으로 destroy -a 명령을 추가하는 데 사용됩니다.

다음과 같이 대화식으로 파일에 프로파일 구성을 내보냅니다.

```
$ netcfg
netcfg> export -d -f myncp2
```

다음과 같이 명령줄 모드로 동일한 작업을 수행합니다.

```
$ netcfg export -d -f myncp2
```

잘린 다음 예에서는 프로파일 구성을 표시하는 방법을 보여줍니다.

```
$ cat myncp2
destroy -a
create ncp "myncp"
create ncu ip "net0"
.
.
.
```

## 내보낸 프로파일 구성 복원

export 하위 명령으로 생성된 프로파일 구성을 시스템으로 다시 가져올 수 있습니다. 내보낸 파일에는 netcfg 명령이 해석할 수 있는 일련의 하위 명령이 포함됩니다. export 하위 명령은 프로파일 구성을 복원하거나 다른 시스템에 프로파일 구성을 복사해야 하는 경우 유용합니다. netcfg 명령줄 모드를 통해서만 프로파일 구성을 복원할 수 있습니다.

다음 명령을 사용하여 내보낸 구성을 복원할 수 있습니다.

```
$ netcfg -f file
```

예를 들어, 다음과 같이 [예 6-14. “파일에 프로파일 구성 내보내기”](#)에서 내보낸 파일을 복원합니다.

```
$ netcfg -f myncp2
Configuration read.
```

## 데스크탑에서 네트워크 구성 관리

네트워크 관리 GUI(이전의 NWAM)를 사용하여 데스크탑에서 네트워크 구성을 관리할 수 있습니다. 이 도구는 netcfg 및 netadm 명령을 사용하는 것과 비슷합니다. GUI를 사용하여

유선 또는 무선 네트워크에 연결하고, 유선 또는 무선 연결을 새로 구성하고, 위치 프로파일을 만들고, 프로파일을 활성화 또는 비활성화할 수 있습니다. 데스크탑에서 네트워크 구성을 관리하는 방법은 노트북 PC 사용자에게 가장 적합하며, 홈 오피스에서 직장의 무선 네트워크로 전환하거나 출장을 떠날 때와 같이 네트워크 상태가 자주 변경되는 경우 유용합니다.

데스크탑에서의 무선 네트워크 관리에 대한 작업 관련 정보는 [“데스크탑에서 무선 네트워크 관리” \[132\]](#)를 참조하십시오.

데스크탑에서 네트워크 구성을 관리할 때는 다음과 같은 일반 지침과 최적 방법을 따르십시오.

- 데스크탑에서 네트워크 구성을 관리할 때 가장 간단한 솔루션은 시스템 정의 Automatic NCP를 사용으로 설정하는 것입니다. 집에서 이 NCP를 사용하여 무선 네트워크에 연결할 수 있습니다.
- 유선 연결을 사용하려면 이더넷 케이블을 연결합니다. 기본 Automatic NCP를 다른 NCP로 전환하지 마십시오. 기존 네트워크 구성을 변경할 필요 없이 네트워크 연결이 무선 네트워크 연결에서 유선 네트워크 연결로 자동으로 전환됩니다.
- 사무실에서 동일한 규칙이 적용됩니다. 네트워크에 연결된 이더넷 케이블이 없고 Automatic NCP가 사용으로 설정된 경우 반응적 네트워킹이 사용되고 무선 네트워크 연결이 자동으로 설정됩니다.
- DefaultFixed NCP가 현재 활성화 상태인 경우 상태만 볼 수 있습니다. 이 NCP가 활성화 상태인 경우 네트워크를 구성하려면 `dladm` 및 `ipadm` 명령을 사용해야 합니다. [2장. Oracle Solaris의 데이터 링크 구성 관리](#) 및 [3장. Oracle Solaris에서 IP 인터페이스와 주소 구성 및 관리](#)를 참조하십시오.
- 집과 사무실 시나리오를 모두 염두에 두고 처음에 무선 네트워크를 선택하고 즐겨찾는 무선 네트워크 목록에 저장해야 합니다(아직 수행하지 않은 경우).

네트워크 관리 GUI를 사용하거나 `netadm select-wifi` 명령을 실행하여 무선 네트워크를 선택합니다. [“반응적 모드로 알려진 WLAN 관리” \[132\]](#)를 참조하십시오.

- 현재 네트워크 연결 상태를 보려면 바탕 화면에 있는 Network Status(네트워크 상태) 알림 아이콘 위로 마우스를 가져가거나 아이콘을 누르면 됩니다. Network Status(네트워크 상태) 알림 아이콘에는 GUI로 네트워크 구성을 만들고 관리하기 위한 컨텍스트 메뉴도 있습니다.

Network Status(네트워크 상태) 알림 아이콘이 데스크탑에 보이지 않으면 System(시스템) -> Administration(관리) -> Network(네트워크)를 선택하여 시작하십시오. 명령줄에서 GUI를 시작하려면 `nwam-manager` 명령을 실행합니다. 자세한 내용은 JDS/GNOME 매뉴얼 페이지 모음에서 `nwam-manager(1M)` 매뉴얼 페이지를 참조하십시오.

- IP 관련 구성은 Network Preferences(네트워크 기본 설정) 대화 상자의 Network Profile(네트워크 프로파일) 섹션에서 관리됩니다. 바탕 화면에 있는 Network Status(네트워크 상태) 알림 아이콘을 누르거나 Network Status(네트워크 상태) 알림 아이콘의 컨텍스트 메뉴에서 Network Preferences(네트워크 기본 설정) 옵션을 선택하여 Network Preferences(네트워크 기본 설정) 대화 상자에 액세스합니다.

# ◆◆◆ 7 장

## Oracle Solaris의 무선 네트워크 관리

---

이 장에는 Oracle Solaris 릴리스에서 무선 네트워크를 관리하는 작업이 포함되어 있습니다.

IEEE 802.11 사양은 LAN(근거리 통신망)에 대한 무선 통신을 정의합니다. 이러한 사양과 사양에서 설명하는 네트워크를 총체적으로 *WiFi*라고 합니다. 이 용어는 Wi-Fi Alliance 무역 그룹의 상표입니다. WiFi 네트워크는 공급자와 잠재 클라이언트가 모두 쉽게 구성할 수 있습니다. 따라서 점차 인기가 증가하고 있으며 전세계에서 일반적으로 사용되고 있습니다. WiFi 네트워크는 휴대폰, TV 및 라디오와 동일한 전파 기술을 사용합니다.

---

참고 - WiFi 서버 또는 액세스 포인트를 구성하는 기능은 Oracle Solaris에 포함되어 있지 않습니다.

---

이 장의 내용:

- [“명령줄을 사용하여 무선 네트워크 관리” \[125\]](#)
- [“보안 WiFi 통신 설정” \[130\]](#)
- [“반응적 모드로 알려진 WLAN 관리” \[132\]](#)
- [“데스크탑에서 무선 네트워크 관리” \[132\]](#)

## 명령줄을 사용하여 무선 네트워크 관리

이 절에는 다음 작업이 설명되어 있습니다.

- [WiFi 네트워크에 연결하는 방법 \[125\]](#)
- [WiFi 링크를 모니터하는 방법 \[129\]](#)

### ▼ WiFi 네트워크에 연결하는 방법

시작하기 전에 다음 단계를 수행하여 WiFi 네트워크에 노트북 PC를 연결하십시오.

1. 관리자로 로그인합니다.
2. 시스템에 있는 데이터 링크의 물리적 속성을 표시합니다.

```
# dladm show-phys
LINK          MEDIA          STATE  SPEED  DUPLEX  DEVICE
net0          Ethernet      up     1500   full    ath0
net1          Ethernet      up     1500   full    e1000g0
```

이전 예에서 출력은 두 가지 링크를 사용할 수 있음을 나타냅니다. 장치 ath0을 통한 net0 링크는 WiFi 통신을 지원합니다. e1000g0 링크를 사용하면 유선 네트워크에 시스템을 연결할 수 있습니다.

### 3. WiFi 인터페이스를 구성합니다.

- a. WiFi를 지원하는 인터페이스를 만듭니다. 예를 들면 다음과 같습니다.

```
# ipadm create-ip net0
```

- b. 링크가 연결되었는지 확인합니다.

```
# ipadm show-if
IFNAME      CLASS      STATE     ACTIVE    OVER
lo0         loopback  ok        yes       --
net0        ip         ok        yes       --
```

### 4. 사용 가능한 네트워크를 확인합니다.

```
# dladm scan-wifi
LINK      ESSID      BSSID/IBSSID  SEC    STRENGTH  MODE  SPEED
net0      ofc        00:0e:38:49:01:d0  none  good      g     54Mb
net0      home      00:0e:38:49:02:f0  none  very weak g     54Mb
net0      linksys   00:0d:ed:a5:47:e0  none  very good g     54Mb
```

scan-wifi 명령은 현재 위치에서 사용 가능한 WiFi 네트워크에 대한 정보를 표시합니다. 출력에는 다음 정보가 포함됩니다.

LINK	WiFi 연결에서 사용할 링크 이름을 나타냅니다.
ESSID	Extended Service Set ID를 나타냅니다. ESSID는 특정 무선 네트워크의 관리자가 임의로 이름을 지정할 수 있는 WiFi 네트워크의 이름입니다.
BSSID/IBSSID	특정 ESSID에 대한 고유 식별자인 BSSID(Basic Service Set ID)를 나타냅니다. BSSID는 네트워크에 특정 ESSID를 제공하는 주변 액세스 포인트의 48비트 MAC 주소입니다.
SEC	무선 네트워크 액세스에 필요한 보안 유형을 나타냅니다. 값은 none, WEP 및 WPA입니다. 자세한 내용은 <a href="#">“보안 WiFi 통신 설정” [130]</a> 을 참조하십시오.
STRENGTH	해당 위치에서 사용할 수 있는 WiFi 네트워크의 라디오 신호 강도를 나타냅니다.

MODE                    네트워크에서 실행하는 802.11 프로토콜의 버전을 나타냅니다. 모드는 a, b 및 g 또는 이러한 모드의 조합입니다.

SPEED                    특정 네트워크의 속도(초당 메가비트)를 나타냅니다.

5. 다음 방법 중 하나로 WiFi 네트워크에 연결합니다.

- 신호가 가장 강한 비보안 WiFi 네트워크에 연결합니다.

```
# dladm connect-wifi
```

- ESSID를 지정하여 비보안 네트워크에 연결합니다.

```
# dladm connect-wifi -e ESSID
```

dladm connect-wifi 명령 사용에 대한 자세한 내용은 “보안 WiFi 통신 설정” [130] 및 dladm(1M) 매뉴얼 페이지를 참조하십시오.

6. 다음과 같이 시스템이 연결된 WiFi 네트워크의 상태를 확인합니다.

```
# dladm show-wifi
LINK        STATUS        ESSID        SEC        STRENGTH    MODE    SPEED
net0        connected    ofc        none        very good    g        36Mb
```

이전 출력은 시스템이 ofc 네트워크에 연결되어 있음을 나타냅니다. 이 절차 중 4단계의 scan-wifi 출력에서는 사용 가능한 네트워크 중 ofc의 신호가 가장 강한 것으로 확인되었습니다. 다른 무선 네트워크를 명시적으로 지정하지 않는 경우 dladm connect-wifi 명령은 신호가 가장 강한 WiFi 네트워크를 자동으로 선택합니다.

7. 다음 방법 중 하나로 인터페이스에 대한 IP 주소를 구성합니다.

- DHCP 서버에서 IP 주소를 받습니다.

```
# ipadm create-addr -T dhcp interface
```

WiFi 네트워크가 DHCP를 지원하지 않는 경우 다음 메시지가 표시됩니다.

```
ipadm: interface: interface does not exist or cannot be managed using DHCP
```

- 정적 IP 주소를 구성합니다.

```
# ipadm create-addr -a address interface
```

시스템에 전용 IP 주소가 있는 경우 이 옵션을 사용합니다.

8. 다음 방법 중 하나로 WiFi 네트워크를 통해 인터넷에 액세스합니다.

- 액세스 포인트가 무료 서비스를 제공하는 경우 선택한 브라우저나 응용 프로그램을 실행할 수 있습니다.

- 액세스 포인트가 요금을 지불해야 하는 상용 WiFi 네트워크에 있는 경우 해당 위치에서 제공되는 지침을 따릅니다.

일반적으로 이 옵션에 대한 키 및 지불 방법을 제공해야 합니다.

9. 다음 방법 중 하나로 세션을 끝냅니다.

- WiFi 세션을 종료하지만 시스템이 계속 실행되도록 합니다.

```
# dladm disconnect-wifi
```

- 현재 여러 세션이 실행되고 있는 경우 특정 WiFi 세션을 종료합니다.

```
# dladm disconnect-wifi link
```

여기서 *link*는 세션에 사용 중인 인터페이스를 나타냅니다.

- WiFi 세션이 실행되는 동안 시스템을 정상적으로 종료합니다.

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

시스템을 종료하기 전에 명시적으로 WiFi 세션의 연결을 해제하지 않아도 됩니다.

예 7-1 특정 WiFi 네트워크에 연결

다음 예에서는 Oracle Solaris 시스템을 무선 네트워크에 연결하기 위해 수행하는 서로 다른 단계를 결합합니다. 이 예제에서는 또한 OS가 무선 네트워크를 임의로 선택하도록 허용하는 대신 시스템이 특정 선호 무선 네트워크에 강제로 연결하도록 지정하는 방법을 보여줍니다. 다음 예에서는 정적 IP 주소 10.192.16.3/24가 노트북 PC용으로 지정되었다고 가정합니다.

```
# dladm show-phys
LINK          MEDIA          STATE  SPEED  DUPLEX  DEVICE
net0          Ethernet      up     1500   full    ath0
net1          Ethernet      up     1500   full    e1000g0

# ipadm create-ip net0
# ipadm show-if net0
IFNAME      CLASS      STATE  ACTIVE  OVER
lo0         loopback  ok     yes     --
net0       ip         ok     yes     --

# dladm scan-wifi
LINK      ESSID      BSSID/IBSSID  SEC  STRENGTH  MODE  SPEED
net0     wifi-a     00:0e:38:49:01:d0  none  weak      g     54Mb
net0     wifi-b     00:0e:38:49:02:f0  none  very weak g     54Mb
net0     ofc-net    00:0d:ed:a5:47:e0  wep   very good g     54Mb
net0     citinet    00:40:96:2a:56:b5  none  good      b     11Mb

# dladm connect-wifi -e citinet
```

```
# dladm show-wifi
LINK      STATUS      ESSID      SEC      STRENGTH  MODE  SPEED
net0      connected   citinet    none     good      g     11Mb

# ipadm create-addr -a 10.192.16.3/24 net0
ipadm: net0/v4
# ipadm show-addr net0
ADDROBJ   TYPE      STATE      ADDR
net0/v4   static   ok         10.192.16.3/24
```

브라우저나 다른 응용 프로그램을 시작하여 WiFi 네트워크를 통해 작업을 시작합니다.

```
# firefox
```

세션을 종료하되 PC가 계속 실행되도록 합니다.

```
# dladm disconnect-wifi
# dladm show-wifi
LINK      STATUS      ESSID      SEC      STRENGTH  MODE  SPEED
net0      disconnected --         --         --         --     --
```

show-wifi 출력 결과에서 net0 링크가 WiFi 네트워크에서 연결이 끊어졌음을 확인합니다.

## ▼ WiFi 링크를 모니터링하는 방법

다음 절차에서는 표준 네트워킹 도구를 통해 WiFi 링크 상태를 모니터링하는 방법과 선택된 링크 등록 정보를 변경하는 방법에 대해 설명합니다.

1. 관리자로 로그인합니다.
2. [WiFi 네트워크에 연결하는 방법 \[125\]](#)에 설명된 대로 WiFi 네트워크에 연결합니다.
3. 시스템의 데이터 링크 등록 정보를 확인합니다.

```
# dladm show-linkprop link
```
4. 링크에 고정 속도를 설정합니다.



주의 - Oracle Solaris는 WiFi 연결에 대한 최적 속도를 자동으로 선택합니다. 링크의 초기 속도를 수정하면 성능이 저하되거나 특정 WiFi 연결이 설정되지 않을 수 있습니다.

다음 예에서와 같이 show-linkprop 하위 명령의 출력에 나열되는 가능한 값 중 하나로 링크 속도를 수정할 수 있습니다.

```
# dladm set-linkprop -p speed=value link
```

5. 링크를 통해 패킷 플로우를 확인합니다.

```
# netstat -I net0 -i 5
```

input net0		output		input (Total)			output		
packets	errs	packets	errs	colls	packets	errs	packets	errs	colls
317	0	106	0	0	2905	0	571	0	0
14	0	0	0	0	20	0	0	0	0
7	0	0	0	0	16	0	1	0	0
5	0	0	0	0	9	0	0	0	0
304	0	10	0	0	631	0	316	0	0
338	0	9	0	0	722	0	381	0	0
294	0	7	0	0	670	0	371	0	0
306	0	5	0	0	649	0	338	0	0
289	0	5	0	0	597	0	301	0	0

예 7-2 링크 속도 설정

다음 예에서는 WiFi 네트워크에 연결한 후 링크 속도를 설정하는 방법을 보여줍니다.

```
# dladm show-linkprop -p speed net0
LINK      PROPERTY      PERM VALUE      EFFECTIVE  DEFAULT  POSSIBLE
net0      speed         r- 25          0          0
1,2,5,6,9,11,12,18,24,36,48,54
# dladm set-linkprop -p speed=36 net0

# dladm show-linkprop -p speed net0
LINK      PROPERTY      PERM VALUE      EFFECTIVE  DEFAULT  POSSIBLE
net0      speed         r- 36          0          0
1,2,5,6,9,11,12,18,24,36,48,54
```

## 보안 WiFi 통신 설정

전파 기술을 통해 WiFi 네트워크를 쉽게 사용할 수 있으며 사용자가 무료로 액세스할 수 있는 경우도 많습니다. 그 결과, WiFi 네트워크 연결이 안전하지 않을 수 있습니다.

다음 유형의 WiFi 연결이 더 안전합니다.

- 액세스가 제한된 개인 WiFi 네트워크에 연결
 

회사나 대학에서 설정한 내부 네트워크와 같은 개인 네트워크는 올바른 보안 챌린지를 제공할 수 있는 사용자만 네트워크에 액세스할 수 있도록 제한합니다. 잠재적 사용자는 연결 시퀀스 도중 키를 제공하거나 보안 VPN(가설 사설망) 응용 프로그램을 통해 네트워크에 로그인해야 합니다.
- WiFi 네트워크에 대한 연결 암호화
 

보안 키를 사용하여 시스템과 WiFi 네트워크 간의 통신을 암호화할 수 있습니다. WiFi 네트워크에 대한 액세스 포인트는 보안 키 생성 기능이 있는 홈 또는 사무실 라우터여야 합니다. 시스템과 라우터는 보안 연결을 만들기 전에 키를 설정하고 공유합니다.

dladm 명령은 액세스 포인트를 통한 연결 암호화를 위해 WEP(Wired Equivalent Privacy) 또는 WPA(Wi-Fi Protected Access) 키를 사용할 수 있습니다. WEP 프로토콜은 무선 연결에 대한 IEEE 802.11 사양에서 정의됩니다. WPA 프로토콜은 무선 연결에 대한 IEEE

802.11i 사양에서 정의됩니다. Oracle Solaris는 WPA 표준의 버전 1 및 2를 지원합니다. WEP 및 WPA와 관련된 `dladm` 명령 옵션에 대한 자세한 내용은 [dladm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

## ▼ WEP 키를 지정하여 암호화된 WiFi 네트워크 연결을 설정하는 방법

다음 절차에서는 홈 시스템과 홈 라우터 간에 보안 통신을 설정하는 방법을 설명합니다. 많은 유무선 홈 라우터에는 보안 키를 생성할 수 있는 암호화 기능이 있습니다.

시작하기 전에 홈 무선 네트워크에 연결하는 경우 라우터가 구성되고 WEP 키가 생성되었는지 확인하십시오. 키 구성을 생성 및 저장하는 방법은 라우터 제조업체의 설명서를 참조하십시오.

1. 관리자로 로그인합니다.
2. 다음과 같이 WEP 키가 포함된 보안 객체를 만듭니다.

```
# dladm create-secobj -c wep keyname
```

여기서 *keyname*은 키에 지정할 이름을 나타냅니다.

3. 보안 객체에 WEP 키의 값을 제공합니다.  
`create-secobj` 하위 명령이 키의 값을 요청하는 스크립트를 실행합니다.

```
provide value for keyname: 5-or-13-byte key
confirm value for keyname: Retype key
```

이 값은 라우터에서 생성된 키입니다. 스크립트는 ASCII나 16진수 형식의 5바이트 또는 13바이트 문자열을 키 값으로 수락합니다.

4. 방금 만든 키의 콘텐츠를 확인합니다.

```
# dladm show-secobj
OBJECT          CLASS
keyname         wep
```

여기서 *keyname*은 보안 객체의 이름입니다.

5. WiFi 네트워크에 대해 암호화된 연결을 설정합니다.

```
# dladm connect-wifi -e network -k keyname interface
```

6. 연결이 안전한지 확인합니다.

```
# dladm show-wifi
LINK    STATUS    ESSID    SEC    STRENGTH  MODE  SPEED
net0    connected  wifi-1  wep    good      g     11Mb
```

이전 출력에서 SEC 열 아래에 있는 wep 값은 연결에 대한 WEP 암호화가 적용되었음을 나타냅니다.

#### 예 7-3 WEP 키를 사용하여 암호화된 WiFi 통신 설정

다음 예에서는 아래 작업을 이미 완료했다고 가정합니다.

- 라우터 제조업체의 설명서에 따라 WEP 키를 만들었습니다.
- 시스템에 보안 객체를 만드는 데 사용할 수 있도록 키를 저장했습니다.

다음과 같이 보안 객체가 만들어집니다.

```
# dladm create-secobj -c wep mykey
provide value for mykey: *****
confirm value for mkey: *****
```

라우터에서 생성된 WEP 키를 제공할 때 입력한 값이 별표로 표시됩니다.

다음 명령은 보안 객체 mykey를 사용하여 WiFi 네트워크 citinet에 대한 암호화된 연결을 설정합니다.

```
# dladm show-secobj
OBJECT          CLASS
mykey           wep
# dladm connect-wifi -e citinet -k mykey net0
```

다음 명령은 WEP 암호화를 통해 citinet 무선 네트워크에 연결되어 있는지 확인합니다.

```
# dladm show-wifi
LINK      STATUS      ESSID      SEC      STRENGTH  MODE  SPEED
net0     connected  citinet    wep      good      g     36Mb
```

## 반응적 모드로 알려진 WLAN 관리

반응적 프로파일을 사용 중인 경우 시스템에서 자동으로 알려진 WLAN에 연결하도록 할 수 있습니다. WLAN에 연결되는 경우 해당 WLAN에 대한 정보가 known-wlan 유형의 네트워크 프로파일에 보존됩니다. 또한 netcfg 명령을 사용하여 known-wlan 프로파일을 수동으로 만들 수 있습니다. 알려진 WLAN에 대한 개요 정보는 [“알려진 WLAN에 대한 설명” \[92\]](#)을 참조하고, 작업 관련 정보는 [“알려진 WLAN 만들기” \[109\]](#)를 참조하십시오.

## 데스크탑에서 무선 네트워크 관리

기본적으로 무선 네트워크 연결이 사용으로 설정된 경우 반응적 네트워킹 데몬(nwamd)은 연결이 나열된 우선 순위 순서에 따라 확인 없이 즐겨찾기 목록에서 사용 가능한 네트워크에 연결하려고 시도합니다. 즐겨 찾는 네트워크를 사용할 수 없는 경우 Wireless Chooser(무선

선택기) 대화 상자가 열립니다. 이 대화 상자에서 연결할 무선 네트워크를 선택할 수 있습니다.

네트워크 관리 GUI의 Network Preferences(네트워크 기본 설정) 대화 상자에 있는 Connection Properties(연결 등록 정보) 뷰의 Wireless(무선) 탭에서 무선 네트워크 연결 방식을 수정할 수도 있습니다. 필요한 경우 바탕 화면에 있는 Network Status(네트워크 상태) 알림 아이콘을 마우스 오른쪽 버튼으로 눌러 수동으로 다른 무선 네트워크에 연결할 수도 있습니다.

**작은 정보** - Network Preferences(네트워크 기본 설정) 대화 상자를 통해 선택한 네트워크에 대한 Connection Properties(연결 등록 정보) 뷰에 액세스할 수 있습니다. 이 대화 상자에는 Show(표시)라는 드롭다운 목록이 있습니다. 이 목록을 사용하여 지정된 네트워크의 뷰를 전환할 수 있습니다. 각 뷰에는 수행할 수 있는 여러 작업 및 해당 뷰와 관련된 선택한 네트워크에 대한 정보가 있습니다.

다음 뷰는 시스템에 있는 각 네트워크 프로파일의 모든 네트워크 연결에 대해 존재합니다.

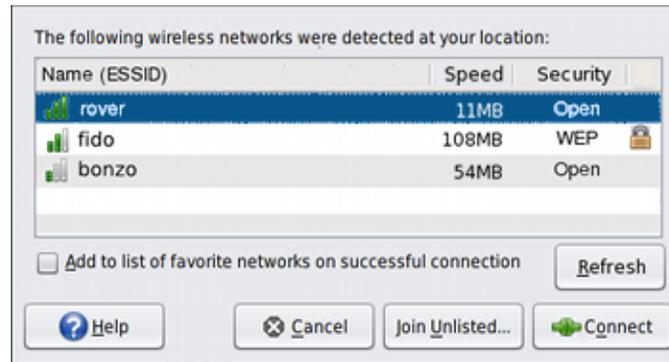
- Connection Status(연결 상태)
- Network Profile(네트워크 프로파일)
- Connection Properties(연결 등록 정보)

프로파일 기반 네트워크 구성에 대한 자세한 내용은 [5장. Oracle Solaris의 프로파일 기반 네트워크 구성 관리 정보](#) 및 GUI 온라인 도움말을 참조하십시오.

## ▼ 무선 네트워크를 연결하는 방법

Network Status(네트워크 상태) 알림 아이콘을 마우스 오른쪽 버튼으로 누르면 사용할 수 있는 Join Wireless Network(무선 네트워크 연결) 옵션을 선택하여 무선 네트워크를 연결합니다. Wireless Chooser(무선 선택기) 대화 상자에는 연결에 사용 가능한 무선 네트워크 목록이 표시됩니다.

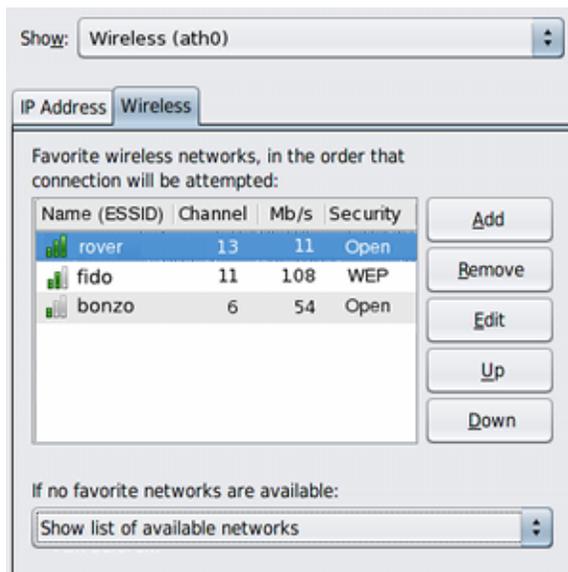
1. 다른 무선 네트워크에 수동으로 연결하려면 다음 작업 중 하나를 수행합니다.
  - Network Status(네트워크 상태) 알림 아이콘을 마우스 오른쪽 버튼으로 누르면 나타나는 메뉴에서 사용 가능한 무선 네트워크를 선택합니다.
  - Network Status(네트워크 상태) 알림 아이콘 메뉴에서 Join unlisted wireless network(나열되지 않은 무선 네트워크 연결) 옵션을 선택합니다.  
나열되지 않은 무선 네트워크는 네트워크 이름을 브로드캐스트하지 않지만 아직 연결에 사용할 수 있도록 구성된 것입니다.
  - Wireless Chooser(무선 선택기) 대화 상자에서 사용 가능한 무선 네트워크를 선택합니다. 이 대화 상자는 사용 가능한 무선 네트워크 중 연결할 네트워크를 선택할 수 있을 때 자동으로 표시됩니다.



2. Join Wireless Network(무선 네트워크 연결) 대화 상자가 열리면 선택한 무선 네트워크에 대해 필요한 정보를 모두 제공합니다.

## 데스크탑에서 즐겨 찾는 무선 네트워크 관리

기본적으로 무선 네트워크에 처음 연결하면 Join Wireless Network(무선 네트워크 연결) 대화 상자에 Add to List of Favorite Networks on Successful Connection(연결되면 즐겨 찾는 네트워크 목록에 추가)라는 확인란이 표시됩니다.



연결에 성공할 경우 즐겨찾기 목록에 무선 네트워크를 추가하려면 이 확인란을 선택합니다. 즐겨찾기 목록에 네트워크를 추가하지 않으려면 확인란의 선택을 해제합니다. 이 확인란은 기본적으로 선택됩니다.

현재 사용할 수 없거나 현재 네트워크 이름을 즐겨찾기 목록에 브로드캐스트하지 않는 무선 네트워크를 추가하려면 Connection Properties(연결 등록 정보) 뷰의 Wireless(무선) 탭으로 이동한 다음 Add(추가) 버튼을 누릅니다. 네트워크를 추가하려면 해당 네트워크 이름, 보안 유형 및 보안 키를 알고 있어야 합니다.



## 색인

---

### 번호와 기호

- activation-mode 등록 정보, 94
  - 서로 다른 프로파일 유형에 대한 등록 정보 값, 94
- ARP(주소 결정 프로토콜), 40
- AS(자율 시스템) 살펴볼 내용 네트워크 토폴로지
- autopush 등록 정보, 30
- BSSID, 126
- cfgadm 명령, 35
- CIDR 표기법, 41
- DefaultFixed 위치, 111
- DHCP, 42
- dladm 명령, 14, 24
  - delete-phys , 27
  - rename-link , 27
  - reset-linkprop , 28
  - scan-wifi, 126
  - set-linkprop , 28
  - show-ether, 31, 32
  - show-link , 25
  - show-linkprop, 31, 129
  - show-phys , 26
  - show-wifi, 127
- dlstat 명령, 27
- DR(동적 재구성)
  - NIC 교체, 34
- ECMP, 63
- ENM
  - 대화식으로 ENM 프로파일 만들기, 108
- ESSID, 126
- /etc/hosts 파일, 41
- /etc/inet/ndpd.conf 파일
  - 임시 주소 구성, 48
- ICMP, 40
- ifconfig 명령, 39
- IP 인터페이스
  - IP 주소, 72, 73
  - IP 주소 변경, 68
  - IP 주소 제거, 68
  - IP 주소 지정, 41
  - MAC 주소 고유성 확인, 36
  - 구성, 43
  - 모니터링, 69, 70
  - 사용 안함 및 사용, 68
  - 인터페이스 구성 삭제, 67
  - 인터페이스 등록 정보, 71
  - 주 인터페이스 변경, 67
  - 주소 등록 정보, 66
  - 패킷 전달 사용, 65
  - 표시
    - IP 주소, 72
    - 인터페이스, 70
    - 인터페이스 등록 정보, 71
    - 일반 정보, 42, 70
    - 주소 등록 정보, 73
- IP 주소
  - DHCP, 42
  - IPv4 및 IPv6, 41
  - 등록 정보, 66, 73
  - 로컬 및 원격, 41
  - 모니터링, 69
  - 정적, 41
  - 제거, 68
  - 패킷 전달, 65
- IP 터널, 41
  - 로컬 및 원격 주소, 41
- ipadm 명령, 14
  - create-addr, 41
  - create-ip, 40
  - delete-addr, 68
  - delete-ip, 67
  - disable-if, 68
  - set-addrprop, 66

- show-addr, 72
- show-addrprop, 66, 73
- show-if, 70
- show-ifprop, 71
- 멀티홈 호스트, 62
- IPv6
  - 경로 지정, 60
  - 사용, 서버에서, 51
  - 임시 주소 구성, 47
- MAC 주소
  - 고유성 확인, 36
- MTU, 29
- name-service/switch 서비스, 41
- NCP
  - NCU와 함께 NCP 만들기
  - NCU 만들기, 103
- NCU
  - 만들기, 103
- ndpd.conf 파일
  - 임시 주소 구성, 48
- netadm 명령, 14
- netcfg 명령, 14
  - walkprop 하위 명령, 116
  - 대화식 모드, 99
- netstat 명령
  - WiFi 링크를 통해 패킷 플로우 확인, 129
- NIC(네트워크 인터페이스 카드)
  - 교체, DR 사용, 34
- nis/tdomain SMF 서비스
  - 로컬 파일 모드 구성, 79
- route 명령, 14, 43
- SCTP, 40
- STREAMS 모듈
  - 데이터 링크, 30
- UDP, 40
- VLAN(가상 근거리 통신망), 24
- VNIC(가상 네트워크 카드), 24
- VPN(가상 사설망), 30
- walkprop 하위 명령
  - 등록 정보 값 확인 및 변경, 116
- WiFi
  - BSSID(Basic Service Set ID), 126
  - ESSID(Extended Service Set ID), 126
  - IEEE 802.11 사양, 125
  - WiFi 구성 예, 128
  - WiFi 네트워크에 연결, 125
  - 링크 모니터, 129
  - 보안 WiFi 링크, 130
  - 암호화된 통신 예, 132
  - 연결 암호화, 131
  - 예, 링크 속도 설정, 130
  - 정의, 125
  - 패킷 플로우 확인, 129
- ㄱ
  - 경로 지정
    - IPv6, 60
    - 단일 인터페이스 호스트, 57
    - 정적 구성, 57
  - 경로 지정 테이블
    - 수동 구성, 55
  - 구성
    - 인터페이스를 수동으로, IPv6에 대해, 45
- ㄴ
  - 나열
    - 모든 프로파일, 113
    - 시스템의 프로파일 정보, 113
    - 특정 등록 정보의 값, 115
    - 특정 프로파일에 대한 모든 등록 정보 값, 114
  - 네트워크 구성
    - IPv6 사용 멀티홈 호스트, 45
  - 네트워크 구성 권한 부여
    - 보안, 95
  - 네트워크 구성 도구
    - dladm 명령, 24
  - 네트워크 구성 명령, 14
  - 네트워크 토폴로지
    - 자율 시스템, 58
  - 네트워크 프로파일
    - ENM, 89
    - NCU, 89
    - 내보낸 프로파일 복원, 123
    - 만들기
      - 위치 프로파일, 106
    - 시스템의 프로파일 정보 나열, 113
    - 알려진 WLAN, 89
    - 위치 프로파일, 89
    - 프로파일 구성 내보내기 및 복원, 121

- 프로파일 만들기 및 관리, 87
  - 프로파일 제거, 120
  - 프로파일에 대한 등록 정보 값 설정 및 변경, 111
  - 네트워크 프로파일 구성
    - 만들기
      - 위치 프로파일, 106
      - 시스템의 프로파일 정보 나열, 113
      - 프로파일 구성 내보내기 및 복원, 121
      - 프로파일에 대한 등록 정보 값 설정 및 변경, 111
    - 네트워크 프로파일 만들기
      - ENM 만들기, 108
- ㄷ**
- 대칭 경로 지정, 63
  - 대화식 모드 netcfg 명령, 99
  - 대화식으로 단일 등록 정보 값 가져오기, 115
  - 대화식으로 등록 정보 값 설정, 111
  - 대화식으로 만들기
    - ENM 프로파일, 108
    - NCU와 함께 NCP, 103
    - 알려진 WLAN 프로파일, 109
  - 데이터 링크
    - autopush 등록 정보, 30
    - MTU 크기 변경, 29
    - STREAMS 모듈, 30
    - VLAN, 24
    - VNIC, 24
    - 등록 정보 설정, 28
    - 런타임 통계 가져오기, 27
    - 링크 속도, 29
    - 링크 이름, 22
    - 링크 통합, 24
    - 물리적 링크, 25
    - 사용자 정의 이름 사용 규칙, 23
    - 알려진 속도 및 사용으로 설정된 속도, 29
    - 이더넷 매개변수 값, 31
    - 이름 바꾸기, 27
    - 일반 이름, 27
    - 자동 협상, 28
    - 제거, 27
    - 표시
      - 네트워크 드라이버 등록 정보, 31, 32
      - 링크, 25
      - 링크 등록 정보, 31
      - 물리적 속성, 26
      - 시스템에서의 물리적 위치, 26
      - 일반 정보, 24
- 도메인 이름
  - nis/domain SMF 서비스, 79
- 동적 경로 지정
  - 최적 사례, 54
- 등록 정보
  - activation-mode 등록 정보, 94
  - walkprop 하위 명령을 사용하여 등록 정보 값 변경, 116
  - 대화식으로 단일 등록 정보 값 가져오기 및 나열, 115
  - 대화식으로 등록 정보 값 설정, 111
  - 특정 등록 정보의 값 가져오기, 115
  - 특정 프로파일에 대한 모든 등록 정보 값 나열, 114
  - 프로파일에 대한 등록 정보 값 설정 및 변경, 111
- ㄹ**
- 런타임 통계
    - 데이터 링크
      - dlstat, 27
  - 로컬 주소, 41
  - 링크 로컬 주소
    - 수동 구성, 토큰 사용, 51
  - 링크 속도, 29
  - 링크 이름, 27
    - 일반, 20
  - 링크 통합, 24
- ㄴ**
- 멀티홈 호스트, 63
    - IPv6에 대해 사용, 45
    - 정의, 61
  - 무선 인터페이스, 125
- ㄷ**
- 반응적 네트워크 구성
    - 프로파일 구성 내보내기, 123
    - 프로파일 제거, 120
  - 반응적 네트워크 구성 모드
    - 네트워크 프로파일 및 유형, 87

- 반응적 프로파일
  - 프로파일 활성화 정책, 93
- 반응적 프로파일 만들기 및 관리, 87
- 반이중, 30
- 변경
  - walkprop 하위 명령을 사용하여 등록 정보 값, 116
  - 프로파일에 대한 등록 정보 값, 111
- 보안 고려 사항
  - WiFi, 130
- 보안 및 권한 부여, 95
- 보조 상태 값
  - 표시, 119
  
- ㅅ
  - 새로운 기능
    - IPv6의 임시 주소, 47
    - 링크 로컬 주소 수동 구성, 50
  - 서버, IPv6
    - IPv6 사용, 51
  
- ㅇ
  - 알려진 WLAN, 109
  - 알려진 WLAN 프로파일
    - 대화식으로 만들기, 109
- 예
  - 대화식으로 등록 정보 값 설정, 112
  - 프로파일 구성 내보내기, 121
    - 명령 파일 모드, 123
  - 프로파일 사용, 98
  - 프로파일 사용 안함, 99
- 원격 주소, 41
- 위치 프로파일
  - 대화식으로 만들기, 106
  - 위치 프로파일 만들기, 106
- 이더넷 매개변수, 31
- 인터페이스
  - 구성
    - 수동으로, IPv6에 대해, 45
    - 임시 주소, 47
- 인터페이스 ID
  - 수동으로 구성된 토큰 사용, 51
- 임시 주소, IPv6
  - 구성, 48
  - 정의, 47
  
- ㅈ
  - 자동 협상, 28
  - 전원 관리, 30
  - 전이중, 30
  - 점보 프레임
    - 지원 사용, 29
  - 정적 경로 지정
    - 구성 예, 56
    - 정적 경로 추가, 55
    - 최적 사례, 54
    - 호스트에서 수동 구성, 57
  - 주 인터페이스, 전환, 27, 32, 67
  - 주소
    - 임시, IPv6, 47
    - 주소 객체, 42
  
- ㅊ
  - 최상위 레벨 프로파일, 100
  
- ㅋ
  - 특정 등록 정보의 값 가져오기, 115
  
- ㅌ
  - 패킷 전달
    - 인터페이스, 65
  - 표시
    - 보조 상태 값, 119
  - 프로파일
    - 네트워크 프로파일 유형, 87
    - 프로파일 구성 내보내기 및 복원, 121
    - 프로파일 만들기
      - ENM을 만드는 프로파일
        - ENM 프로파일, 108
      - 대화식으로 ENM 만들기, 108
      - 대화식으로 만들기
        - NCU와 함께 NCP, 103
        - 알려진 WLAN 프로파일, 109

만들기

NCP에 대한 NCU, 103

WLAN 만들기, 109

위치 프로파일, 106

프로파일 복원, 123

프로파일 사용, 98

프로파일 사용 안함, 99

프로파일 제거, 120

프로파일 활성화 정책, 93

프로파일에 대한 등록 정보 값 설정 및 변경, 111

대화식으로 등록 정보 값 설정, 112

명령줄 모드로 등록 정보 값 설정, 112

ㅎ

호스트

멀티홈

구성, 61

임시 IPv6 주소, 47

