



Netra™ 240 서버 시스템 관리 설명서

Sun Microsystems, Inc.
www.sun.com

부품 번호 817-5015-11
2004년 7월, 개정판 A

이 책에 대한 의견 보내실 곳: <http://www.sun.com/hwdocs/feedback>

Copyright 2004 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 모든 권리는 저작권자의 소유입니다.

이 문서에 설명된 기술과 관련된 지적 재산권은 Sun Microsystems, Inc.에게 있습니다. 특히 <http://www.sun.com/patents>에 나열된 하나 이상의 미국 특허와 미국 및 기타 국가에서의 추가 특허 또는 특허 출원 중인 제품이 이러한 지적 재산권에 포함되며, 지적 재산권의 대상은 이에 국한되지 않습니다.

본 제품 또는 설명서는 사용, 복사, 배포 및 역컴파일을 제한하는 라이선스 하에서 배포됩니다. 본 제품 또는 설명서의 어떠한 부분도 Sun 및 Sun 소속 라이선스 부여자(있는 경우)의 사전 서면 승인 없이는 어떠한 형태나 수단으로도 복제할 수 없습니다.

글꼴 기술을 포함한 타사 소프트웨어는 저작권이 등록되었으며 Sun 공급업체로부터 라이선스를 취득한 것입니다.

본 제품의 일부는 Berkeley BSD 시스템에서 파생되어 나온 것일 수 있으며, 이에 관해서는 University of California로부터 라이선스를 취득했습니다. UNIX는 X/Open Company, Ltd.를 통해 독점적 라이선스를 취득한 미국 및 기타 국가의 등록 상표입니다.

Sun, Sun Microsystems, Sun 로고, AnswerBook2, docs.sun.com, OpenBoot, Netra, SunVTS, Sun Enterprise Authentication Mechanism, Solaris는 미국 및 기타 국가에서 Sun Microsystems, Inc.의 상표 또는 등록 상표입니다.

모든 SPARC 상표는 라이선스 하에서 사용되며 미국 및 기타 국가에서 SPARC International, Inc.의 상표 또는 등록 상표입니다. SPARC 상표가 있는 제품은 Sun Microsystems, Inc.가 개발한 구조를 기반으로 합니다.

OPEN LOOK과 Sun™ Graphical User Interface는 Sun Microsystems, Inc.가 사용자 및 라이선스 소유자를 위해 개발한 것입니다. Sun은 컴퓨터 업계에서 시각적 또는 그래픽 사용자 인터페이스 개념을 연구하고 개발해온 Xerox의 선구적 업적에 경의를 표합니다. Sun은 Xerox Graphical User Interface에 대한 Xerox의 비독점적 라이선스를 보유하고 있으며 이 라이선스는 OPEN LOOK GUI를 구현하거나 그 외의 경우 Sun의 서면 라이선스 계약을 준수하는 Sun의 라이선스 피부여자를 포괄합니다.

미국 정부 권리-상업용. 정부 사용자는 Sun Microsystems, Inc. 표준 라이선스 계약과 FAR의 해당 규정 및 부칙의 적용을 받습니다.

출판물은 "있는 그대로" 제공되며 본 제품의 시장성, 합목적성, 특허권 비침해에 대한 묵시적인 보증을 비롯한 모든 명시적, 묵시적인 조건, 표현, 보증에 대한 책임이 없습니다. 단, 이러한 권리가 법적으로 무효가 되는 경우는 예외로 합니다.



Adobe PostScript

목차

1. 문제 해결 도구	1
진단 도구 개요	2
시스템 프롬프트	3
Advanced Lights Out Manager	4
서버 상태 표시등	4
▼ 로케이터 LED 상태를 표시하는 방법	5
▼ 로케이터 LED 를 켜는 방법	5
▼ 로케이터 LED 를 끄는 방법	5
알람 상태 표시등	6
POST(전원 인가 후 자가 검사)	8
POST 제어 진단	8
▼ POST 진단 시작 방법	10
OpenBoot 명령	11
probe-scsi 및 probe-scsi-all 명령	11
probe-ide 명령	12
show-devs 명령	13
▼ OpenBoot 명령 실행 방법	14
OpenBoot 진단	14
▼ OpenBoot 진단 시작 방법	14
OpenBoot 진단 시험 제어	15

test 및 test-all 명령	16
OpenBoot 진단 오류 메시지	17
운영 체제 진단 도구	18
오류 메시지 및 시스템 메시지 로그 파일	18
Solaris 소프트웨어 시스템 정보 명령	18
prtconf 명령	19
prtdiag 명령	20
prtfru 명령	22
psrinfo 명령	23
showrev 명령	24
▼ Solaris 플랫폼 시스템 정보 명령의 실행 방법	25
최근 진단 시험 결과	25
▼ 최근의 시험 결과를 보는 방법	25
OpenBoot 구성 변수	26
▼ OpenBoot 구성 변수 확인 및 설정 방법	26
watch-net 및 watch-net-all 명령으로 네트워크 연결 확인	27
자동 시스템 복구	28
Auto-Boot 옵션	28
오류 처리 방식 요약	29
재설정 시나리오	30
▼ ASR 을 활성화하는 방법	30
▼ ASR 을 비활성화하는 방법	31
2. SunVTS 소프트웨어	33
SunVTS 소프트웨어 개요	33
SunVTS 테스트	34
SunVTS 소프트웨어 및 보안	35
▼ SunVTS 소프트웨어의 설치 여부 확인 방법	35
SunVTS 소프트웨어 설치	36

SunVTS 소프트웨어 설명서 보기 36

3. Advanced Lights Out Manager 37

Advanced Lights Out Manager 개요 37

ALOM 포트 38

admin 암호 설정 39

기본 ALOM 기능 39

▼ ALOM 프롬프트로 전환하는 방법 40

▼ 서버 콘솔 프롬프트로 전환하는 방법 40

▼ 다른 사용자로부터 콘솔 쓰기 권한 가져오기 40

자동 서버 재시작 41

환경 모니터링 및 제어 41

A. 알람 릴레이 출력 API 45

색인 51

그림

그림 1-1	시스템 프롬프트 흐름	3
그림 1-2	전면 패널의 표시등 위치	4

표

표 1-1	문제 해결 도구 요약	2
표 1-2	서버 상태 표시등(전면 및 후면)	4
표 1-3	알람 표시등 및 점점 알람 상태	6
표 1-4	OpenBoot 구성 변수	9
표 1-5	<code>test-args</code> OpenBoot 구성 변수의 키워드	16
표 1-6	Solaris 플랫폼 정보 표시 명령	25
표 2-1	SunVTS 소프트웨어 테스트	34
표 3-1	ALOM 이 감시하는 구성 요소	38
표 3-2	Netra 240 서버 인클로저 온도 임계값	42

머리말

*Netra 240 서버 시스템 관리 설명서*는 숙련된 시스템 관리자를 위한 설명서입니다. 이 설명서에서는 Netra™ 240 서버의 진단 도구와 각종 서버 관리 작업도 전체적으로 설명합니다.

이 설명서를 이용하려면 컴퓨터 네트워크 개념 및 용어에 대한 실무적인 지식과 Solaris™ 운영 체제(Solaris OS)에 대한 자세한 사전 지식이 있어야 합니다.

읽기 전에

이 설명서에서는 서버 설치 및 랙 장착 절차에 대해서는 다루지 않습니다. 이에 관해서는 *Netra 240 서버 설치 설명서*(부품 번호 817-4998-11)를 참조하십시오.

이 책에 나오는 절차를 따라 하기 전에, *Important Safety Information for Sun Hardware Systems*(부품 번호 816-7190-10)를 먼저 읽어 두어야 합니다.

UNIX 명령 사용

이 단원에서는 UNIX 명령 중 필요한 기본 명령만을 설명합니다.
예를 들면 다음과 같습니다.

이 설명서에는 시스템 종료, 시스템 부팅 및 장치 구성과 같은 기본 UNIX® 명령 및 절차에 대한 정보는 포함되어 있지 않습니다. 이와 관련된 내용은 다음을 참조하십시오.

- 시스템과 함께 제공되는 소프트웨어 설명서
- 다음 위치에 있는 Solaris™ 운영 환경 설명서

<http://docs.sun.com>

셸 프롬프트

셸	프롬프트
C 셸	시스템 이름%
C 셸 슈퍼유저	시스템 이름#
Bourne 셸 및 Korn 셸	\$
Bourne 셸 및 Korn 셸 슈퍼유저	#

표기 규칙

활자체 ⁱ	의미	예
AaBbCc123	컴퓨터 화면에 나타나는 명령, 파일 및 디렉토리의 이름	.login 파일을 편집합니다. 모든 파일 목록을 표시하려면 ls -a를 사용합니다. % You have mail.
AaBbCc123	컴퓨터 화면의 출력 내용과 대조되는 사용자의 입력 내용	% su Password:
AaBbCc123	문서 제목, 새 단어나 용어, 강조하는 단어 실제 이름이나 값으로 대체되는 명령행 변수	<i>사용 설명서</i> 6장을 참조하십시오. 이러한 것들을 <i>클래스</i> 옵션이라고 합니다. 이 작업은 <i>수퍼유저</i> 만 수행할 수 있습니다. 파일을 삭제하려면 rm <i>파일 이름</i> 을 입력하십시오.

i 사용 중인 브라우저의 설정이 이 설정과 다를 수도 있습니다.

관련 설명서

적용 분야	제목	부품 번호
설치 개요	<i>Netra 240 Server Quick Start Guide</i>	817-3904-xx
최신 제품 업데이트	<i>Netra 240 Server Release Notes</i>	817-3142-xx
준수 사항 및 안전	<i>Important Safety Information for Sun Hardware Systems</i>	816-7190-10
	<i>Netra 240 Server Safety and Compliance Manual</i>	817-5018-11
문서 웹 사이트의 위치	<i>Sun Netra 240 Server Documentation</i>	817-2697-10
설치	<i>Netra 240 서버 설치 설명서</i>	817-4998-11
LOM(Lights-Out Management)	<i>Netra 240 서버용 Sun ALOM(Advanced Lights Out Manager) 소프트웨어 사용 설명서</i>	817-5007-11
수리	<i>Netra 240 Server Service Manual</i>	817-2699-xx

Sun 설명서 얻기

다음을 통해서 한글화된 버전을 비롯하여 Sun에서 제공하는 다양한 설명서를 보거나 인쇄 또는 구입할 수 있습니다.

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

타사 웹 사이트

Sun은 이 설명서에서 언급한 타사 웹 사이트의 서비스 제공 여부에 대해서는 책임을 지지 않습니다. Sun은 타사 사이트나 리소스를 통해 제공되는 내용, 광고, 상품 및 기타 정보를 보증하거나 책임을 지지 않습니다. Sun은 타사 사이트나 리소스를 통해 제공되는 내용, 상품 또는 서비스의 사용으로 인해 발생한 실제적 또는 주장된 피해나 손실에 대해 책임을 지지 않습니다.

Sun 기술 지원

이 설명서에 나와 있지 않은 기술 문제가 이 제품과 관련하여 발생한 경우, 다음 사이트를 방문하십시오.

<http://www.sun.com/service/contacting>

고객 의견

Sun은 설명서의 개선을 위해 항상 노력하고 있으며, 고객의 의견 및 제안을 언제나 환영합니다. 다음 사이트에서 의견을 제출할 수 있습니다.

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

다음과 같이 설명서의 제목과 부품 번호를 함께 적어서 보내주십시오.

Netra 240 서버 시스템 관리 설명서, 부품 번호 817-5015-11

문제 해결 도구

이 장에서는 Netra 240 서버에서 사용 가능한 진단 도구를 설명합니다. 이 장은 다음 단원으로 구성되어 있습니다.

- 2페이지의 "진단 도구 개요"
- 3페이지의 "시스템 프롬프트"
- 4페이지의 "Advanced Lights Out Manager"
- 8페이지의 "POST(전원 인가 후 자가 검사)"
- 11페이지의 "OpenBoot 명령"
- 14페이지의 "OpenBoot 진단"
- 18페이지의 "운영 체제 진단 도구"
- 25페이지의 "최근 진단 시험 결과"
- 26페이지의 "OpenBoot 구성 변수"
- 28페이지의 "자동 시스템 복구"

진단 도구 개요

아래 표는 Netra 240 서버에서 사용할 수 있는 Sun 제공 진단 도구를 요약한 것입니다.

표 1-1 문제 해결 도구 요약

진단 도구	유형	설명	접근성 및 가용성	원격 가능
ALOM	하드웨어 및 소프트웨어	환경 조건 모니터링, 기본적인 고장 분리 및 원격 콘솔 액세스 기능 제공.	대기 전원으로 운영 체제 없이 작동 가능.	원격 액세스 방식으로 설계됨.
LED	하드웨어	전체 시스템과 특정한 구성 요소의 상태 표시.	시스템 새시에 들어 있음. 전원이 켜져 있으면 언제든지 사용 가능.	로컬이지만 ALOM을 통해 볼 수 있음.
POST(전원 인가 후 자가 검사)	펌웨어	시스템의 핵심 구성 요소를 시험.	시스템 시작 시 자동으로 실행됨. 운영 체제가 실행 중이지 않을 때에도 사용 가능.	로컬이지만 ALOM을 통해 볼 수 있음.
OpenBoot 명령	펌웨어	각종 시스템 정보를 표시.	운영 체제가 실행 중이지 않을 때에도 사용 가능.	로컬이지만 ALOM을 통해 액세스할 수 있음.
OpenBoot 진단	펌웨어	주변 장치와 I/O 장치를 중심으로 시스템 구성 요소 시험.	자동 실행 또는 대화식 실행. 운영 체제가 실행 중이지 않을 때에도 사용 가능.	로컬이지만 ALOM을 통해 볼 수 있음.
Solaris 소프트웨어 명령	소프트웨어	각종 시스템 정보를 표시.	운영 체제 필요.	로컬이지만 ALOM을 통해 액세스할 수 있음.
SunVTS™ 소프트웨어	소프트웨어	여러 시험을 동시에 실행하여 시스템 진단 및 안정성 시험을 수행.	운영 체제 필요. 옵션 패키지.	네트워크를 통해 확인 및 제어 가능.

시스템 프롬프트

Netra 240 서버의 기본적인 서버 프롬프트는 다음과 같습니다.

- ok-OpenBoot PROM 프롬프트
- sc>-ALOM(Advanced Lights Out Manager) 프롬프트
- #-Solaris 소프트웨어 슈퍼유저(Bourne 및 Korn shell) 프롬프트

그림 1-1에는 이들 세 프롬프트의 관계와 프롬프트 전환 방법이 나와 있습니다.

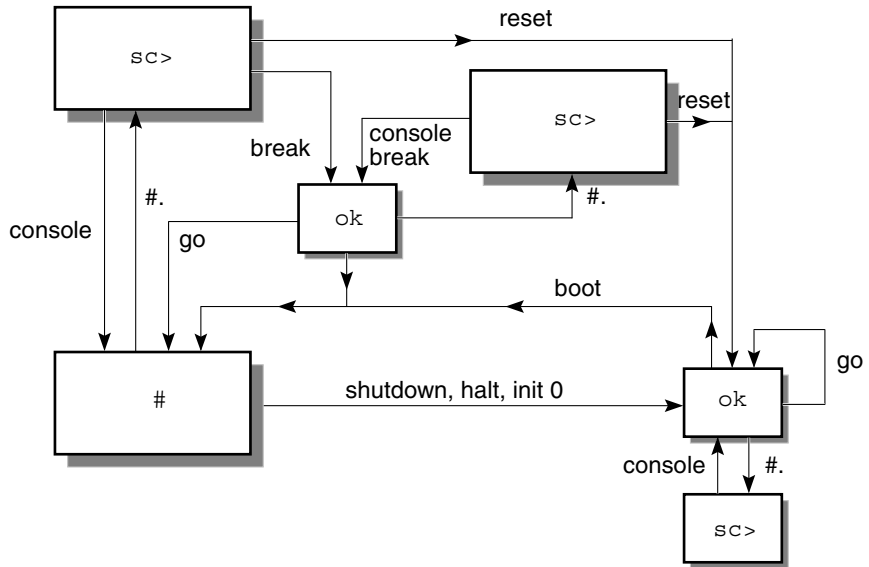


그림 1-1 시스템 프롬프트 흐름

다음 명령이 그림 1-1의 흐름도에 있습니다.

- ALOM 명령: console, reset, break
- 탈출 문자열: #.
- Solaris 소프트웨어 명령: shutdown, halt, init 0
- OpenBoot 명령: go, boot

Advanced Lights Out Manager

Netra 240 서버용 Sun™ Advanced Lights Out Manager(ALOM)는 몇 가지 LED 상태 표시등을 제공합니다. 여기서는 LED 상태의 의미와 켜고 끄는 방법을 자세히 설명합니다. ALOM에 관한 자세한 내용은 3장을 참조하십시오.

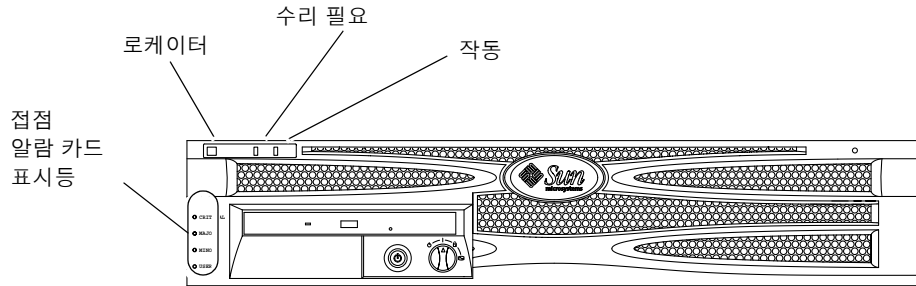


그림 1-2 전면 패널의 표시등 위치

서버 상태 표시등

서버에는 3개의 LED 상태 표시등이 있습니다. 이들은 전면 베젤(그림 1-2)에 있으며 후면 패널에서도 같은 자리에 위치합니다. 표 1-2에 표시등이 요약되어 있습니다.

표 1-2 서버 상태 표시등(전면 및 후면)

표시등	LED 색상	LED 상태	의미
작동	녹색	켜짐	서버가 켜져 있고 Solaris 운영 체제가 실행 중입니다.
		꺼짐	전원이 꺼져 있거나 Solaris OS가 실행되고 있지 않습니다.
수리 필요	노란색	켜짐	서버에서 문제를 감지했으며 서비스 요원의 점검이 필요합니다.
		꺼짐	서버에서 오류가 감지되지 않습니다.
로케이터	흰색	켜짐	setlocator 명령을 사용할 때에는 표시등이 계속 켜져서 랙의 다른 서버들과 이 서버를 구별해 줍니다.

시스템 콘솔이나 ALOM 명령행 인터페이스에서 로케이터 LED를 켜고 끌 수 있습니다.

▼ 로케이터 LED 상태를 표시하는 방법

- 다음 방법 중 하나를 사용합니다.
 - 슈퍼유저 권한으로 다음을 입력합니다.

```
# /usr/sbin/locator
```

- ALOM 명령행 인터페이스에서 다음을 입력합니다.

```
sc> showlocator
```

▼ 로케이터 LED를 켜는 방법

- 다음 방법 중 하나를 사용합니다.
 - 슈퍼유저 권한으로 다음을 입력합니다.

```
# /usr/sbin/locator -n
```

- ALOM 명령행 인터페이스에서 다음을 입력합니다.

```
sc> setlocator on
```

▼ 로케이터 LED를 끄는 방법

- 다음 방법 중 하나를 사용합니다.
 - 슈퍼유저 권한으로 다음을 입력합니다:

```
# /usr/sbin/locator -f
```

- ALOM 명령행 인터페이스에서 다음을 입력합니다.

```
sc> setlocator off
```

알람 상태 표시등

접점 알람 카드에는 ALOM이 지원하는 LED 상태 표시등 4개가 있습니다. 이들은 전면 베젤에 세로로 위치합니다(그림 1-2). 알람 표시등 및 접점 알람 상태는 표 1-3에 나와 있습니다. 알람 표시등에 대한 자세한 내용은 *Netra 240 서버용 Sun ALOM(Advanced Lights Out Manager) 소프트웨어 사용 설명서*(부품 번호 817-5007-11)를 참조하십시오. 알람 표시등을 제어하는 API에 대한 자세한 내용은 부록 A를 참조하십시오.

표 1-3 알람 표시등 및 접점 알람 상태

표시등 및 릴레이 레이블.	표시등 색상	응용 프로그램 또는 서버 상태	조건 또는 조치	시스템 표시등 상태	알람 표시등 상태	릴레이 NC ^{iv} 상태	릴레이 NO ^v 상태	의견	
위험 (Alarm0)	빨간색	서버 상태 (전원 켜짐/꺼짐 및 Solaris OS 작동/비작동)	전원 입력 없음.	꺼짐	꺼짐	단힘	열림	기본 상태.	
			시스템 전원 꺼짐.	꺼짐	꺼짐 ⁱⁱⁱ	단힘	열림	입력 전원 연결됨.	
			시스템 전원 켜짐; Solaris OS 완전히 로드되지 않음.	꺼짐	꺼짐 ⁱⁱⁱ	단힘	열림	일시적 상태.	
			Solaris OS 로드 완료.	켜짐	꺼짐	열림	단힘	정상 작동 상태.	
			감시 시간 초과.	꺼짐	켜짐	단힘	열림	일시적 상태, Solaris OS 재부트.	
			사용자가 Solaris OS 종료를 시작 ⁱ .	꺼짐	꺼짐 ⁱⁱⁱ	단힘	열림	일시적 상태.	
			입력 전원 차단.	꺼짐	꺼짐	단힘	열림	기본 상태.	
			사용자에 의한 시스템 전원 차단.	꺼짐	꺼짐 ⁱⁱⁱ	단힘	열림	일시적 상태.	
			응용 프로그램 상태	사용자가 비상 알람을 켜짐으로 설정 ⁱⁱ .	—	켜짐	단힘	열림	심각한 결함 감지.
			사용자가 비상 알람을 꺼짐으로 설정 ⁱⁱ .	—	꺼짐	열림	단힘	심각한 결함 해결.	
Major (Alarm1)	빨간색	응용 프로그램 상태	사용자가 Major 알람을 켜짐으로 설정 ⁱⁱ .	—	켜짐	열림	단힘	주요 결함 감지.	
			사용자가 Major 알람을 꺼짐으로 설정 ⁱⁱ .	—	꺼짐	단힘	열림	주요 결함 해결.	

표 1-3 알람 표시등 및 접점 알람 상태 (계속)

표시등 및 릴레이 레이블.	표시등 색상	응용 프로그램 또는 서버 상태	조건 또는 조치	시스템 표시등 상태	알람 표시등 상태	릴레이 NC ^{iv} 상태	릴레이 NO ^v 상태	의견
Minor (Alarm2)	황색	응용 프로그램 상태	사용자가 Minor 알람을 켜짐으로 설정 ⁱⁱ .	—	켜짐	열림	단힘	부차적 결합 감지.
			사용자가 Minor 알람을 꺼짐으로 설정 ⁱⁱ .	—	꺼짐	단힘	열림	부차적 결합 해결.
User (Alarm3)	황색	응용 프로그램 상태	사용자가 User 알람을 켜짐으로 설정 ⁱⁱ .	—	켜짐	열림	단힘	User 결합 감지.
			사용자가 User 알람을 꺼짐으로 설정 ⁱⁱ .	—	꺼짐	단힘	열림	User 결합 해결.

i 사용자는 `init0` 및 `init6` 등의 명령으로 시스템을 종료할 수 있습니다. 시스템 전원 차단은 여기에 포함되지 않습니다.

ii 사용자는 고장 상태를 파악하고 이를 바탕으로 Solaris 플랫폼 알람 API 또는 ALOM CLI에서 알람을 켤 수 있습니다. 알람 API에 대한 자세한 내용은 Appendix A를 참조하시고, ALOM CLI에 대한 자세한 내용은 *Netra 240 서버용 Sun ALOM(Advanced Lights Out Manager) 소프트웨어 사용 설명서*(부품 번호 817-5007-11)를 참조하십시오.

iii 이 알람 표시등 상태의 구현은 바뀔 수 있습니다.

iv NC 상태는 정상적으로 종료된(Normally Closed) 상태를 말합니다. 이는 정상적으로 종료된 상태인 릴레이 연결의 기본 모드를 나타냅니다.

v NO 상태는 정상적으로 열린 상태입니다. 이는 정상적으로 열린 상태인 릴레이 연결의 기본 모드를 나타냅니다.

사용자가 알람을 설정하면 항상 콘솔에는 메시지가 표시됩니다. 예를 들어, 비상 알람을 설정하면 다음 메시지가 콘솔에 표시됩니다.

```
SC Alert: CRITICAL ALARM 설정
```

비상 알람을 설정하더라도 해당 알람 표시등이 켜지지 않는 경우도 있습니다. 다음 릴리스에서 이를 개선할 것입니다 (표 1-3의 각주ⁱⁱⁱ 참조).

POST(전원 인가 후 자가 검사)

POST(Power-on self-test)는 시스템 일부가 고장났는지 진단할 수 있는 펌웨어 프로그램입니다. POST는 CPU 모듈, 마더보드, 메모리, 일부 온보드 I/O 장치를 포함하여 시스템의 핵심 기능을 검사합니다. 이 소프트웨어는 하드웨어 고장의 성질을 판단하는 데 유용한 메시지를 생성합니다. POST는 시스템이 부팅되지 않는 경우에도 실행할 수 있습니다.

POST는 대부분의 시스템 오류를 감지하며, 마더보드 OpenBoot PROM에 들어 있습니다. 두 가지 환경 변수를 설정하여 전원을 켤 때 POST가 실행되도록 OpenBoot 소프트웨어를 프로그래밍할 수 있습니다. diag-switch? 및 diag-level 플래그가 해당 변수입니다. 이들 두 변수는 시스템 구성 카드에 저장되어 있습니다.

POST는 시스템의 전원을 켜는 때, 그리고 다음 두 조건이 만족되는 경우에는 시스템 자동 재설정 이후에도 실행됩니다.

- diag-switch?가 true로 설정된 경우(기본값은 false)
- diag-level이 min, max 또는 menus로 설정된 경우(기본값은 min)
- post-trigger가 재설정의 종류에 맞는 경우(기본값은 power-on-reset)

diag-level이 min 또는 max로 설정된 경우, POST는 약식 시험이나 확장 시험을 각각 실시합니다.

diag-level이 menus로 설정된 경우, 시동 시 실행되는 모든 시험의 메뉴가 표시됩니다.

POST 진단 및 오류 메시지 보고서는 콘솔에 표시됩니다.

POST 제어 진단

OpenBoot 구성 변수를 설정하여 POST 진단(부팅 프로세스의 여러 측면)을 제어할 수 있습니다. 변경된 OpenBoot 구성 변수는 보통 컴퓨터를 다시 시작해야 적용됩니다.

표 1-4에는 이 중에서 가장 중요하고 유용한 변수들이 나와 있습니다. OpenBoot 구성 변수 변경 방법은 26페이지의 "OpenBoot 구성 변수 확인 및 설정 방법"에 나와 있습니다.

표 1-4 OpenBoot 구성 변수

OpenBoot 구성 변수	설명 및 키워드
auto-boot	<p>운영 체제의 자동 시작 여부를 결정합니다. 기본값은 true입니다.</p> <ul style="list-style-type: none"> • true—펌웨어 검사를 마쳤을 때 자동으로 운영 체제를 시작합니다. • false—boot를 입력할 때까지 시스템은 ok 프롬프트 상태로 있습니다.
diag-level	<p>실행되는 진단 단계 또는 유형을 결정합니다. 기본값은 min입니다.</p> <ul style="list-style-type: none"> • off—시험을 실행하지 않습니다. • min—기본 시험만 실행합니다. • max—장치에 따라 더욱 세밀한 시험을 실행합니다. • menus—POST 레벨에서 메뉴 방식 시험을 개별적으로 실행할 수 있습니다.
diag-script	<p>OpenBoot 진단 시험할 장치를 결정합니다. 기본값은 none입니다.</p> <ul style="list-style-type: none"> • none—장치를 검사하지 않습니다. • normal—자가 검사 기능이 있는 내장(중앙판 기준) 장치를 시험합니다. • all—자가 검사 기능이 있는 모든 장치를 시험합니다.
diag-switch?	<p>시스템을 진단 모드로 전환하거나 진단 모드에서 빠져 나옵니다. 기본값은 false입니다.</p> <ul style="list-style-type: none"> • true—진단 모드: POST 진단 및 OpenBoot 진단 시험을 실행합니다. • false—기본 모드: POST 또는 OpenBoot 진단 시험을 실행하지 않습니다.
post-trigger obdiag-trigger	<p>이 두 변수로 POST(또는 OpenBoot 진단 시험)를 실행시키는 재설정 이벤트의 클래스를 지정합니다. 이 변수에는 단일 키워드 뿐만 아니라 키워드 사이를 공백으로 구분하여 앞에서부터 3개의 키워드를 조합하여 지정할 수 있습니다. 자세한 내용은 26페이지의 "OpenBoot 구성 변수 확인 및 설정 방법"을 참조하십시오.</p> <ul style="list-style-type: none"> • error-reset—복구가 불가능한 특정 하드웨어 오류 상태로 인해 발생하는 재설정입니다. 오류 재설정은 대개 하드웨어가 시스템 상태 데이터를 손상시킬 때 일어납니다. 이러한 예로 CPU 및 시스템 감시 재설정, 치명적 오류 그리고 일부 CPU 재설정 이벤트를 들 수 있습니다 (기본값). • power-on-reset—전원/대기 버튼을 눌러 실행하는 재설정입니다(기본값). • user-reset—사용자 또는 운영 체제에 의해 시작되는 재설정입니다. • all-resets—모든 종류의 시스템 재설정입니다. • none—POST(또는 OpenBoot 진단 시험)를 실행하지 않습니다.
input-device	<p>콘솔 입력의 출처를 선택합니다. 기본값은 ttya입니다.</p> <ul style="list-style-type: none"> • ttya—내장 SERIAL MGT 포트. • ttyb—내장 범용 직렬 포트(10101). • keyboard—그래픽 단말기에 연결된 키보드로 입력.

표 1-4 OpenBoot 구성 변수 (계속)

OpenBoot 구성 변수	설명 및 키워드
output-device	진단 및 기타 콘솔 출력 내용이 표시되는 곳을 선택합니다. 기본값은 ttya입니다. <ul style="list-style-type: none"> • ttya—내장 SERIAL MGT 포트. • ttyb—내장 범용 직렬 포트(10101). • screen—그래픽 단말기에 연결된 화면으로 출력.ⁱ

i POST 메시지는 그래픽 단말기에 표시할 수 없습니다. POST 메시지는 output-device가 screen으로 설정된 경우에도 ttya로 출력됩니다.

주 - 이러한 변수는 OpenBoot 진단 시험 뿐만 아니라 POST 진단에도 영향을 줍니다.

POST 진단이 완료되면 POST는 POST가 실행한 모든 시험의 상태를 OpenBoot 펌웨어에 보고합니다. 그런 다음 OpenBoot 펌웨어 코드로 제어 기능이 다시 전환됩니다.

POST 진단에서 오류가 발견되지 않는에도 서버 시동이 안 되는 경우에는 OpenBoot 진단 시험을 실행하십시오.

▼ POST 진단 시작 방법

1. ok 프롬프트로 이동합니다.
2. 다음을 입력합니다.

```
ok setenv diag-switch? true
```

3. 다음을 입력합니다.

```
ok setenv diag-level x™
```

여기서 x™은 보려는 진단 정보의 양에 따라 min, max, menus 중 하나로 설정입니다.

4. 다음을 입력합니다.

```
ok reset-all
```

post-trigger가 user-reset으로 설정된 경우, 시스템이 POST 진단을 실행합니다. 상태 및 오류 메시지가 콘솔 창에 표시됩니다. POST는 오류가 감지되면 그 오류를 설명하는 오류 메시지를 표시합니다.

5. POST 실행이 완료되면 다음을 입력하여 diag-switch? 값을 false로 복원하십시오.

```
ok setenv diag-switch? false
```

diag-switch?를 false로 재설정하면 부팅 시간이 최소화됩니다.

OpenBoot 명령

OpenBoot 명령은 사용자가 ok 프롬프트에서 입력하는 명령입니다. 유용한 진단 정보를 표시하는 OpenBoot 명령은 다음과 같습니다.

- probe-scsi 및 probe-scsi-all
- probe-ide
- show-devs

probe-scsi 및 probe-scsi-all 명령

probe-scsi 및 probe-scsi-all 명령은 SCSI 장치의 문제를 진단합니다.



주의 - halt 명령 또는 Stop-A 키 시퀀스를 사용하여 ok 프롬프트로 전환한 경우, probe-scsi 또는 probe-scsi-all 명령을 실행하면 시스템이 정지될 수 있습니다.

probe-scsi 명령은 내장 SCSI 컨트롤러에 연결된 모든 SCSI 장치와 통신합니다. probe-scsi-all 명령은 또한 PCI 슬롯에 설치된 호스트 어댑터에 연결되어 있는 장치에도 액세스합니다.

probe-scsi 및 probe-scsi-all 명령은 연결되어 작동 중인 모든 SCSI 장치에 대해 루프 ID, 호스트 어댑터, 논리적 장치 번호, 고유 WWN(World Wide Name), 그리고 유형과 제조업체 등 장치 설명을 표시합니다.

다음은 probe-scsi 명령의 출력 예입니다.

코드 예제 1-1 probe-scsi 명령 출력

```
{1} ok probe-scsi
Target 0
  Unit 0   Disk      SEAGATE ST373307LSUN72G 0207
Target 1
  Unit 0   Disk      SEAGATE ST336607LSUN36G 0207
{1} ok
```

다음은 probe-scsi-all 명령의 출력 예입니다.

코드 예제 1-2 probe-scsi-all 명령 출력

```
{1} ok probe-scsi-all
/pci@1c,600000/scsi@2,1

/pci@1c,600000/scsi@2
Target 0
  Unit 0   Disk      SEAGATE ST373307LSUN72G 0207
Target 1
  Unit 0   Disk      SEAGATE ST336607LSUN36G 0207

{1} ok
```

probe-ide 명령

probe-ide 명령은 IDE 버스와 연결된 모든 IDE(Integrated Drive Electronics) 장치와 통신합니다. 이것은 DVD 드라이브와 같은 미디어 장치용 내부 시스템 버스입니다.



주의 - halt 명령 또는 Stop-A 키 시퀀스를 사용하여 ok 프롬프트로 전환한 경우, probe-ide 명령을 실행하면 시스템이 정지될 수 있습니다.

다음은 probe-ide 명령의 출력 예입니다.

코드 예제 1-3

probe-ide 명령 출력

```
{1} ok probe-ide
Device 0 ( Primary Master )
        Not Present

Device 1 ( Primary Slave )
        Not Present

Device 2 ( Secondary Master )
        Not Present

Device 3 ( Secondary Slave )
        Not Present

{1} ok
```

show-devs 명령

show-devs 명령은 펌웨어 장치 트리의 각 장치로 가는 하드웨어 장치 경로를 표시합니다. 코드 예제 1-4은 출력 예입니다.

코드 예제 1-4

show-devs 명령 출력

```
/pci@1d,700000
/pci@1c,600000
/pci@1e,600000
/pci@1f,700000
/memory-controller@1,0
/SUNW,UltraSPARC-IIIi@1,0
/memory-controller@0,0
/SUNW,UltraSPARC-IIIi@0,0
/virtual-memory
/memory@m0,0
/aliases
/options
/openprom
/chosen
/packages
/pci@1d,700000/network@2,1
/pci@1d,700000/network@2
/pci@1c,600000/scsi@2,1
/pci@1c,600000/scsi@2
/pci@1c,600000/scsi@2,1/tape
/pci@1c,600000/scsi@2,1/disk
```

```
/pci@1c,600000/scsi@2/tape
/pci@1c,600000/scsi@2/disk
/pci@1e,600000/ide@d
/pci@1e,600000/usb@a
/pci@1e,600000/pmu@d
/pci@1e,600000/isa@7
/pci@1e,600000/ide@d/cdrom
/pci@1e,600000/ide@d/disk
```

▼ OpenBoot 명령 실행 방법

1. 시스템을 중지하여 ok 프롬프트로 전환합니다.
시스템을 종료하기 전에 사용자에게 알립니다.
2. 콘솔 프롬프트에서 실행할 명령을 입력합니다.

OpenBoot 진단

OpenBoot 진단 코드는 POST 진단과 마찬가지로 펌웨어를 기반으로 하며 Boot PROM에 들어 있습니다.

▼ OpenBoot 진단 시작 방법

1. 유형:

```
ok setenv diag-switch? true
ok setenv auto-boot? false
ok reset-all
```

2. 유형:

```
ok obdiag
```

이 명령은 OpenBoot 진단 메뉴를 표시합니다.

```
ok obdiag

          o b d i a g
-----
| 1 i2c@0,320      | 2 ide@d      | 3 network@2    |
| 4 network@2,1   | 5 rtc@0,70   | 6 scsi@2       |
| 7 scsi@2,1      | 8 serial@0,2e8 | 9 serial@0,3f8 |
| 10 usb@a        | 11 usb@b     | 12 flashprom@2,0 |
|-----|-----|-----|
| Commands: test test-all except help what setenv versions printenvs exit |
```

주 - 서버에 PCI 카드가 설치되어 있다면 obdiag 메뉴에 추가의 시험 메뉴가 표시됩니다.

3. 다음을 입력합니다.

```
obdiag> test n
```

여기서 *n*은 시험을 실행할 횟수를 나타냅니다.

해당 시험에 대한 설명을 보려면, obdiag> 프롬프트에서 다음을 입력합니다.

```
obdiag> help
```

OpenBoot 진단 시험 제어

POST(표 1-4 참조)를 제어하는데 사용되는 대부분의 OpenBoot 구성 변수는 OpenBoot Diagnostics 검사에도 영향을 미칩니다.

- OpenBoot 진단 시험의 수준을 제어하려면 `diag-level` 변수를 사용합니다.
- 시험의 실행 방식을 사용자 정의하려면 `test-args`를 사용합니다.

기본적으로 `test-args`는 빈 문자열을 포함하도록 설정됩니다. 표 1-5에 지정되어 있는 키워드를 한 개 이상 사용하여 `test-args`를 수정할 수 있습니다.

표 1-5 test-args OpenBoot 구성 변수의 키워드

키워드	설명
<code>bist</code>	외부 장치와 주변 장치에 내장된 자가 검사(BIST)를 호출.
<code>debug</code>	모든 디버그 메시지를 표시.
<code>iopath</code>	버스 및 상호 연결 무결성을 시험.
<code>loopback</code>	장치의 외부 루프백 경로를 시험.
<code>media</code>	외부 장치와 주변 장치 매체의 접근성을 시험.
<code>restore</code>	선행 시험 실행이 실패한 경우 장치를 원래의 상태로 복원.
<code>silent</code>	각 시험의 상태는 표시하지 않고 오류만 표시.
<code>subtests</code>	호출된 기본 시험 및 각 하위 시험을 표시.
<code>verbose</code>	모든 시험에 대해 자세한 상태 메시지 표시.
<code>callers=<i>n</i></code>	오류가 발생할 경우 그 이전의 <i>N</i> 개의 호출자를 역추적하여 표시합니다. <code>callers=0</code> —오류가 발생하기 전까지의 모든 호출자를 역추적하여 표시합니다.
<code>errors=<i>n</i></code>	<i>N</i> 개의 오류가 발생할 때까지 시험을 계속 실행합니다. <code>errors=0</code> —검사를 중지하지 않고 모든 오류 보고 사항을 표시합니다.

OpenBoot 진단 시험을 사용자 정의하려면 다음 예와 같이 `test-args`를 쉘표로 구분된 키워드 목록으로 설정하면 됩니다.

```
ok setenv test-args debug,loopback,media
```

test 및 test-all 명령

또한 `ok` 프롬프트에서 OpenBoot 진단 시험을 직접 실행할 수도 있습니다. 이렇게 하려면 `test` 명령 다음에 시험할 장치(장치 세트)의 전체 하드웨어 경로를 입력하십시오. 예를 들면 다음과 같습니다.

```
ok test /pci@x,y/SUNW,qlc@2
```

개별 시험을 사용자 정의하려면 `test-args`를 다음과 같이 사용하십시오.

```
ok test /usb@1,3:test-args={verbose,debug}
```


이 명령은
test-args OpenBoot 구성 변수의 값을 변경하지 않으며 현재 시험에만 적용됩니다.
test-all 명령을 사용하면 장치 트리의 모든 장치를 시험할 수 있습니다.

```
ok test-all
```

test-all에 경로 인수를 지정할 경우 지정된 장치와 그 하위 장치만 시험합니다. 다음 예는 USB 버스와 USB 버스에 연결된 자가 검사 기능을 가진 모든 장치를 시험하는 명령입니다.

```
ok test-all /pci@9,700000/usb@1,3
```

OpenBoot 진단 오류 메시지

OpenBoot 진단 오류 결과는 문제에 대한 간략한 요약, 문제가 있는 하드웨어, 실패한 하위 시험 그리고 기타 진단 정보를 포함하는 표 형식으로 보고됩니다. 코드 예제 1-5는 OpenBoot 진단 오류 메시지의 예입니다.

코드 예제 1-5 OpenBoot Diagnostics 오류 메시지

```
Testing /pci@1e,600000/isa@7/flashprom@2,0

ERROR: FLASHPROM CRC-32 is incorrect
SUMMARY : Obs=0x729f6392 Exp=0x3d6cdf53 XOR=0x4ff3bcc1 Addr=0xfeebbffc
DEVICE  : /pci@1e,600000/isa@7/flashprom@2,0
SUBTEST : selftest:crc-subtest
MACHINE : Netra 240
SERIAL#  : 52965531
DATE    : 03/05/2003 01:33:59 GMT
CONTROLS: diag-level=max test-args=

ERROR: /pci@1e,600000/isa@7/flashprom@2,0 selftest failed, return code = 1
Selftest at /pci@1e,600000/ide@7/flashprom@2,0 (errors=1) .....
failed
Pass:1 (of 1) Errors:1 (of 1) Tests Failed:1 Elapsed Time: 0:0:0:27
```

운영 체제 진단 도구

OpenBoot 진단 시험을 통과한 뒤, 시스템은 Solaris OS 부팅을 시도합니다. 서버가 다중 사용자 모드로 실행되고 있을 때에는 소프트웨어 기반의 진단 도구와 SunVTS 소프트웨어를 사용할 수 있습니다. 이러한 도구로 서버를 감시하고, 시스템을 시험하여 오류를 찾아낼 수 있습니다.

주 - auto-boot? OpenBoot 구성 변수를 false로 설정한 경우, 펌웨어 기반의 시험이 완료되어도 운영 체제는 부팅되지 *않습니다*.

위에서 언급한 도구 외에도 오류 메시지 및 시스템 메시지 로그 파일과 Solaris 시스템 정보 명령을 이용할 수 있습니다.

오류 메시지 및 시스템 메시지 로그 파일

오류 및 기타 시스템 메시지는 `/var/adm/messages` 파일에 저장됩니다. 메시지는 운영 체제, 환경 제어 하위 시스템 및 여러 소프트웨어 응용 프로그램 등 다양한 출처로부터 이 파일에 기록됩니다.

Solaris 소프트웨어 시스템 정보 명령

다음 Solaris 소프트웨어 시스템 정보 명령은 Netra 240 서버의 상태를 확인하는 데 이용되는 데이터를 표시합니다.

- prtconf
- prtdiag
- prtfru
- psrinfo
- showrev

이 항목에서는 위의 명령을 통해 제공되는 내용에 대해 설명합니다. 이 명령들에 대한 자세한 정보는 해당 설명 페이지를 참조하십시오.

prtconf 명령

prtconf 명령은 Solaris 소프트웨어 장치 트리를 표시합니다. 이 트리에는 OpenBoot 펌웨어가 감시하는 모든 장치는 물론 운영 체제 소프트웨어에서만 인식하는 추가 장치(예: 개인 디스크)가 포함되어 있습니다. 또한 prtconf의 출력에는 시스템 메모리의 총 크기가 포함됩니다. 코드 예제 1-6은 prtconf 출력의 일부를 발췌한 것입니다.

코드 예제 1-6 prtconf 명령 출력

```
# prtconf

System Configuration: Sun Microsystems sun4u
Memory size: 5120 Megabytes
System Peripherals (Software Nodes):

SUNW,Netra-240
  packages (driver not attached)
    SUNW,builtin-drivers (driver not attached)
    deblocker (driver not attached)
    disk-label (driver not attached)
    terminal-emulator (driver not attached)
    dropins (driver not attached)
    kbd-translator (driver not attached)
    obp-tftp (driver not attached)
    SUNW,i2c-ram-device (driver not attached)
    SUNW,fru-device (driver not attached)
    ufs-file-system (driver not attached)
  chosen (driver not attached)
  openprom (driver not attached)
    client-services (driver not attached)
  options, instance #0
  aliases (driver not attached)
  memory (driver not attached)
  virtual-memory (driver not attached)
  SUNW,UltraSPARC-IIIi (driver not attached)
  memory-controller, instance #0
  SUNW,UltraSPARC-IIIi (driver not attached)
  memory-controller, instance #1
  pci, instance #0
```

prtconf 명령에 -p 옵션을 붙여 실행하면 OpenBoot과 비슷하게 출력됩니다. show-devs 명령. 시스템 펌웨어가 컴파일한 장치만 출력됩니다.

prtdiag 명령

prtdiag 명령은 시스템 구성 부품의 상태를 요약한 진단 정보 표를 표시합니다. prtdiag 명령의 표시 형식은 Solaris 운영 체제의 버전에 따라 다릅니다. 다음 코드 예제는 Solaris 소프트웨어를 실행하는 Netra 240 서버에서 prtdiag가 만들어낸 출력의 일부를 발췌한 것입니다.

코드 예제 1-7

prtdiag 명령 출력

```
# prtdiag
System Configuration: Sun Microsystems sun4u Netra 240
System clock frequency: 160 MHz
Memory size: 2GB

===== CPUs =====
      CPU  Freq      E$      CPU      CPU      Temperature      Fan
      CPU  Freq      Size      Impl.  Mask      Die      Ambient      Speed  Unit
-----
      MB/P0 1280 MHz  1MB      US-IIIi  2.3      -      -
      MB/P1 1280 MHz  1MB      US-IIIi  2.3      -      -

===== IO Devices =====
      Bus  Freq
      Brd  Type  MHz  Slot      Name      Model
-----
      0  pci   66      2  network-pci14e4,1648.108e.16+
      0  pci   66      2  network-pci14e4,1648.108e.16+
      0  pci   66      2  scsi-pci1000,21.1000.1000.1 +
      0  pci   66      2  scsi-pci1000,21.1000.1000.1 +
      0  pci   66      2  network-pci14e4,1648.108e.16+
      0  pci   66      2  network-pci14e4,1648.108e.16+
      0  pci   33      7  isa/serial-su16550 (serial)
      0  pci   33      7  isa/serial-su16550 (serial)
      0  pci   33      7  isa/rmc-comm-rmc_comm (seria+
      0  pci   33      13  ide-pci10b9,5229.c4 (ide)

===== Memory Configuration =====
Segment Table:
-----
Base Address      Size      Interleave Factor  Contains
-----
0x0              1GB      1                  GroupID 0
0x100000000      1GB      1                  GroupID 0
```

코드 예제 1-7 prtdiag 명령 출력 (계속)

```

Memory Module Groups:
-----
ControllerID  GroupID  Labels
-----
0              0        MB/P0/B0/D0,MB/P0/B0/D1
Memory Module Groups:
-----
ControllerID  GroupID  Labels
-----
1              0        MB/P1/B0/D0,MB/P1/B0/D1
    
```

코드 예제 1-7의 정보 외에도, prtdiag 명령에 **verbose** 옵션(-v)을 붙여 실행하면 전면 패널의 상태, 디스크 상태, 팬 상태, 전원 공급 장치, 하드웨어 개정판 그리고 시스템 온도에 대한 내용이 보고됩니다(코드 예제 1-8 참조).

코드 예제 1-8 prtdiag Verbose 옵션 출력

```

-----
Location      Sensor          Temperature  Lo LoWarn HiWarn  Hi Status
-----
MB            T_ENC           22C         -7C   -5C    55C   58C okay
MB/P0        T_CORE          57C         -     -     110C  115C okay
MB/P1        T_CORE          54C         -     -     110C  115C okay
PS0          FF_OT           -           -     -     -     -   okay
PS1          FF_OT           -           -     -     -     -   okay
    
```

과열 상태인 경우 prtdiag 명령은 상태 열에 오류를 보고합니다(코드 예제 1-9 참조).

코드 예제 1-9 prtdiag 과열 상태 표시 출력

```

-----
Location      Sensor          Temperature  Lo LoWarn HiWarn  Hi Status
-----
MB            T_ENC           22C         -7C   -5C    55C   58C okay
MB/P0        T_CORE          118C        -     -     110C  115C failed
MB/P1        T_CORE          112C        -     -     110C  115C warning
PS0          FF_OT           -           -     -     -     -   okay
PS1          FF_OT           -           -     -     -     -   okay
    
```

또한 특정 구성 요소에 장애가 발생한 경우에도 prtdiag 명령은 해당 상태 열에 오류를 보고합니다(코드 예제 1-10 참조).

```

팬 속도 :
-----
Location          Sensor           Status   Speed
-----
MB/P0/F0          RS               failed   0 rpm
MB/P0/F1          RS               okay     3994 rpm
F2                RS               okay     2896 rpm
PS0               FF_FAN          okay
F3                RS               okay     2576 rpm
PS1               FF_FAN          okay
-----

```

prtfriu 명령

Netra 240 서버는 시스템에 있는 모든 FRU의 계층적 목록과 다양한 FRU에 대한 세부 정보를 유지 관리합니다.

prtfriu 명령은 이 계층적 목록과 여러 FRU의 SEEPROM(serial electrically-erasable PROM) 장치에 저장된 데이터를 표시합니다. 코드 예제 1-11는 prtfriu -l 명령으로 생성한 계층적 FRU 목록의 일부입니다.

```

# prtfriu -l
/frutree
/frutree/chassis (fru)
/frutree/chassis/MB?Label=MB
/frutree/chassis/MB?Label=MB/system-board (container)
/frutree/chassis/MB?Label=MB/system-board/SC?Label=SC
/frutree/chassis/MB?Label=MB/system-board/SC?Label=SC/sc (fru)
/frutree/chassis/MB?Label=MB/system-board/BAT?Label=BAT
/frutree/chassis/MB?Label=MB/system-board/BAT?Label=BAT/battery (fru)
/frutree/chassis/MB?Label=MB/system-board/P0?Label=P0
/frutree/chassis/MB?Label=MB/system-board/P0?Label=P0/cpu (fru)
/frutree/chassis/MB?Label=MB/system-board/P0?Label=P0/cpu/F0?Label=F0
/frutree/chassis/MB?Label=MB/system-board/P0?Label=P0/cpu/F0?Label=F0/fan-unit
(fru)
/frutree/chassis/MB?Label=MB/system-board/P0?Label=P0/cpu/F1?Label=F1
/frutree/chassis/MB?Label=MB/system-board/P0?Label=P0/cpu/F1?Label=F1/fan-unit
(fru) .....

```

코드 예제 1-12은 prtfru -c 명령으로 생성한 SEEPROM 데이터의 일부입니다. 이 출력에는 컨테이너와 그 데이터만 표시되고 FRU 트리 구조는 나타나지 않습니다.

코드 예제 1-12 prtfru -c 명령 출력

```
# prtfru -c
/frutree/chassis/MB?Label=MB/system-board (container)
  SEGMENT: SD
    /ManR
    /ManR/UNIX_Stamp32: Mon Dec  2 19:47:38 PST 2002
    /ManR/Fru_Description: FRUID, INSTR, M'BD, 2X1.28GHZ, CPU
    /ManR/Manufacture_Loc: Hsinchu, Taiwan
    /ManR/Sun_Part_No: 3753120
    /ManR/Sun_Serial_No: 000615
    /ManR/Vendor_Name: Mitac International
    /ManR/Initial_HW_Dash_Level: 02
    /ManR/Initial_HW_Rev_Level: 0E
    /ManR/Fru_Shortname: MOTHERBOARD
    /SpecPartNo: 885-0076-11
/frutree/chassis/MB?Label=MB/system-board/P0?Label=
P0/cpu/B0?Label=B0/bank/D0?Label=D0/mem-module (container)
/frutree/chassis/MB?Label=MB/system-board/P0?Label=
P0/cpu/B0?Label=B0/bank/D1?Label=D1/mem-module (container).....
```

prtfru 명령으로 표시되는 데이터는 FRU의 종류에 따라 다릅니다. 일반적으로 다음과 같은 정보가 표시됩니다.

- FRU 설명
- 제조업체 이름 및 위치
- 부품 번호 및 일련 번호
- 하드웨어 개정 단계

psrinfo 명령

psrinfo 명령은 각 CPU가 온라인에 소개된 날짜와 시간을 알려줍니다. 이 명령에 verbose(-v) 옵션을 붙여 실행하면 클럭 속도 등 CPU에 대한 추가 정보가 표시됩니다. 코드 예제 1-13은 psrinfo -v 명령의 출력 예입니다.

코드 예제 1-13 psrinfo -v 명령 출력

```
# psrinfo -v
Status of processor 0 as of: 07/28/2003 14:43:29
  Processor has been on-line since 21.07.03 18:43:37.
  The sparcv9 processor operates at 1280 MHz,
    and has a sparcv9 floating point processor.
Status of processor 1 as of: 07/28/2003 14:43:29
  Processor has been on-line since 21.07.03 18:43:36.
  The sparcv9 processor operates at 1280 MHz,
    and has a sparcv9 floating point processor.
```

showrev 명령

showrev 명령은 현재 하드웨어와 소프트웨어의 개정판 정보를 표시합니다. 코드 예제 1-14는 showrev 명령의 출력 예입니다.

코드 예제 1-14 showrev 명령 출력

```
# showrev
Hostname: vsp78-36
Hostid: 8328c87b
Release: 5.8
Kernel architecture: sun4u
Application architecture: sparc
Hardware provider: Sun_Microsystems
Domain: vsplab.SFBay.Sun.COM
Kernel version: SunOS 5.8 Generic 108528-18 November 2002
```

showrev 명령에 -p 옵션을 붙여 실행하면 설치된 패치가 표시됩니다. 코드 예제 1-15는 showrev -p 명령의 출력 예를 일부 발췌한 것입니다.

코드 예제 1-15 showrev -p 명령 출력

```
Patch: 109729-01 Obsoletes: Requires: Incompatibles: Packages: SUNWcsu
Patch: 109783-01 Obsoletes: Requires: Incompatibles: Packages: SUNWcsu
Patch: 109807-01 Obsoletes: Requires: Incompatibles: Packages: SUNWcsu
Patch: 109809-01 Obsoletes: Requires: Incompatibles: Packages: SUNWcsu
Patch: 110905-01 Obsoletes: Requires: Incompatibles: Packages: SUNWcsu
Patch: 110910-01 Obsoletes: Requires: Incompatibles: Packages: SUNWcsu
Patch: 110914-01 Obsoletes: Requires: Incompatibles: Packages: SUNWcsu
Patch: 108964-04 Obsoletes: Requires: Incompatibles: Packages: SUNWcsr
```


▼ Solaris 플랫폼 시스템 정보 명령의 실행 방법

- 명령 프롬프트에서 표시할 시스템 정보의 유형에 해당하는 명령을 입력합니다.

자세한 내용은 18페이지의 "Solaris 소프트웨어 시스템 정보 명령"을 참조하십시오. 명령 요약은 표 1-6을 참조하십시오.

표 1-6 Solaris 플랫폼 정보 표시 명령

명령	표시되는 정보	입력할 내용	참고
prtconf	시스템 구성 정보	/usr/sbin/prtconf	—
prtdiag	진단 및 구성 정보	/usr/platform/sun4u/sbin/prtdiag	추가 세부사항을 보려면 -v 옵션을 사용합니다.
prtfru	FRU 계층 구조 및 SEEPROM 메모리의 내용	/usr/sbin/prtfru	계층을 표시하려면 -l 옵션을 사용합니다. SEEPROM 데이터를 표시하려면 -c 옵션을 사용합니다.
psrinfo	CPU가 작동되기 시작한 날짜 및 시간, 프로세서 클럭 속도	/usr/sbin/psrinfo	클럭 속도 및 기타 데이터를 얻으려면 -v 옵션을 사용합니다.
showrev	하드웨어 및 소프트웨어 개정 정보	/usr/bin/showrev	소프트웨어 패치를 표시하려면 -p 옵션을 사용합니다.

최근 진단 시험 결과

전원을 켜다 켜 때마다, 최근의 POST 및 OpenBoot 진단 시험 결과의 요약이 저장됩니다.

▼ 최근의 시험 결과를 보는 방법

1. ok 프롬프트로 이동합니다.
2. 다음 중 하나를 수행합니다.
 - 가장 최근의 POST 결과를 보려면 다음을 입력합니다.

```
ok show-post-results
```

- 가장 최근의 OpenBoot 진단 시험 결과를 보려면 다음을 입력합니다.

```
ok show-obdiag-results
```

이 명령은 시스템에 설치되어 있는 하드웨어 구성 요소 목록을 표시하며, 구성 요소에는 시험 통과 결과 및 통과하지 못한 경우는 실패한 POST 또는 OpenBoot 진단 시험이 함께 표시됩니다.

OpenBoot 구성 변수

IDPROM에 저장된 스위치 및 진단 구성 변수는 POST 진단 및 OpenBoot 진단 시험의 실행 방식과 시기를 결정하는 데 사용됩니다. 여기서는 OpenBoot 구성 변수를 사용하고 수정하는 방법을 설명합니다. 중요한 OpenBoot 구성 변수의 목록은 표 1-4를 참조하십시오.

변경한 OpenBoot 구성 변수는 대개 다음 번 부팅부터 적용됩니다.

▼ OpenBoot 구성 변수 확인 및 설정 방법

- ok 프롬프트가 나타나도록 서버를 중지시킵니다.
 - 모든 OpenBoot 구성 변수의 현재 값을 표시하려면 `printenv` 명령을 사용하십시오. 다음은 `printenv` 명령의 출력 내용 예제입니다.

Variable Name	Value	Default Value
diag-level	min	min
diag-switch?	false	false

- OpenBoot 구성 변수 값을 설정하거나 변경하려면 `setenv` 명령을 사용하십시오.

```
ok setenv diag-level max
diag-level =          max
```

- 여러 개의 키워드를 수용하는 OpenBoot 구성 변수를 설정하려면 키워드를 공백으로 구분하십시오.

watch-net 및 watch-net-all 명령으로 네트워크 연결 확인

watch-net 진단 시험은 기본 네트워크 인터페이스의 이더넷 패킷을 감시합니다. watch-net-all 진단 시험은 시스템 보드에 연결되어 있는 기본 네트워크 인터페이스와 그 밖의 모든 추가된 네트워크 인터페이스의 이더넷 패킷을 감시합니다. 시스템에서 제대로 수신한 패킷은 마침표(.)로 표시됩니다. 프레임링 오류와 주기적 중복 검사(cyclic redundancy check) 오류가 발생한 경우는 X와 함께 이와 관련된 오류 설명이 표시됩니다.

- watch-net 진단 시험을 시작하려면 ok 프롬프트에서 watch-net 명령을 입력합니다(코드 예제 1-16 참조).

코드 예제 1-16 watch-net 진단 출력 메시지

```
{0} ok watch-net
Internal loopback test -- succeeded.
Link is -- up
Looking for Ethernet Packets.
ë.í is a Good Packet. ëXí is a Bad Packet.
Type any key to stop.....
```

- watch-net-all 진단 시험을 수행하려면 ok 프롬프트에서 watch-net-all을 입력합니다(코드 예제 1-17 참조).

코드 예제 1-17 watch-net-all 진단 출력 메시지

```
{0} ok watch-net-all
/pci@1f,0/pci@1,1/network@c,1
Internal loopback test -- succeeded.
Link is -- up
Looking for Ethernet Packets.
ë.í Is a Good Packet. ëXí is a Bad Packet.
Type any key to stop.
```

자동 시스템 복구

주 - 자동 시스템 복구(ASR)는 Netra™ 240 서버가 지원하는 자동 서버 재시작과는 다른 기능입니다. 자동 서버 재시작에 대한 정보는 3장을 참조하십시오.

자동 시스템 복구(ASR) 기능은 자가 검사 기능 및 오류가 있는 하드웨어 구성 요소를 감지하여 이에 대한 구성을 해제하는 자동 구성 기능으로 구성되어 있습니다. ASR 기능을 사용하면 몇 가지 치명적이지 않은 하드웨어 오류 또는 결함이 발생해도 서버의 작동 재개가 가능합니다.

ASR로 감시하는 구성 요소 중 서버 작동에 핵심적이지 않은 구성 요소에 오류 또는 결함이 발생할 경우, 서버는 자동으로 재부팅됩니다. 이렇게 함으로써 오류가 있는 하드웨어 구성 요소로 인해 시스템 전체가 다운되거나 시스템이 계속 다운되는 것을 방지할 수 있습니다.

시스템의 전원을 켜는 과정에서 오류가 감지될 경우, 오류가 있는 해당 구성 요소가 비활성화됩니다. 이 구성 요소가 없어도 시스템 작동이 가능하다면 부팅 절차가 계속됩니다.

이렇게 불완전한 부팅 기능을 지원하기 위해 OpenBoot 펌웨어는 장치 트리 노드에 적절한 상태 속성을 만들어 줌으로써 1275 클라이언트 인터페이스(장치 트리 이용)에서 장치를 *고장* 또는 *사용 안함*으로 표시합니다. Solaris OS는 이런 표시가 있는 하위 시스템의 드라이버는 활성화하지 않습니다.

오류가 있는 구성 요소가 전기적으로 유훈 상태(예: 임의적인 버스 오류 또는 신호 잡음이 발생되지 않음)이면, 시스템은 자동으로 재부팅되어 서비스가 호출되는 동안 작동을 재개합니다.

고장나거나 비활성인 장치를 새 장치로 교체하면 OpenBoot 펌웨어가 재시동 시 자동으로 장치 상태를 수정합니다.

주 - ASR 기능을 이용하려면 사용자가 직접 설정해야 합니다(30페이지의 "ASR을 활성화하는 방법" 참조).

Auto-Boot 옵션

auto-boot? 설정은 재설정 후에 펌웨어가 운영 체제를 자동 부팅하도록 할지 여부를 지정합니다. 기본 설정은 true입니다.

auto-boot-on-error? 설정은 하위 시스템 장애가 감지되었을 때 시스템이 저하된 부팅을 시도하도록 할지 여부를 지정합니다. 불완전 부팅이 자동으로 실행되게 하려면 auto-boot? 및 auto-boot-on-error? 설정을 모두 true로 설정해야 합니다.

- 스위치를 설정하려면 다음을 입력합니다.

```
ok setenv auto-boot? true
ok setenv auto-boot-on-error? true
```

주 - auto-boot-on-error?의 기본 설정은 false입니다. 따라서 이 설정을 true로 변경하기 전까지 시스템은 불완전 부팅을 시도하지 않습니다. 또 불완전 부팅 기능을 설정했다라도 복구 불가능한 치명적인 오류가 발생한 경우에는 부팅을 시도하지 않습니다. 치명적이고 복구 불가능한 오류의 예는 29페이지의 "오류 처리 방식 요약"을 참조하십시오.

오류 처리 방식 요약

시스템 전원을 켜는 과정 중에 발생하는 오류의 처리 방식에는 다음 3가지 방식이 있습니다.

- POST 또는 OpenBoot 진단에서 오류가 발견되지 않은 경우, auto-boot?가 true이면 시스템은 부팅을 시도합니다.
- POST 또는 OpenBoot 진단에서 치명적이지 않은 오류만 발견된 경우, auto-boot?가 true이고 auto-boot-on-error?가 true이면 시스템은 부팅을 시도합니다.

주 - POST 또는 OpenBoot 진단에서 일반 부팅 장치와 연관된 비치명적 오류가 발견된 경우, OpenBoot 펌웨어는 고장난 장치를 자동으로 구성 해제하고 boot-device 구성 변수에 지정된 다음 순번의 부팅 장치 사용을 시도합니다.

- POST 또는 OpenBoot 진단에서 치명적인 오류가 감지된 경우, auto-boot? 또는 auto-boot-on-error?의 설정에 관계 없이 시스템은 부팅되지 않습니다. 복구 불가능한 치명적인 오류에는 다음과 같은 것들이 있습니다.
 - 모든 CPU에 오류가 있는 경우
 - 모든 논리 메모리 뱅크에 오류가 있는 경우
 - 플래시 RAM의 주기적 중복 검사(CRC)가 실패한 경우
 - 현장 교체 가능한 장치(FRU) PROM의 구성 데이터가 치명적으로 손상된 경우
 - 주문형 집적 회로(ASIC)에 치명적 오류가 발생한 경우

재설정 시나리오

세 가지 OpenBoot 구성 변수인 `diag-switch?`, `obdiag-trigger`, `post-trigger`는 시스템 재설정 이벤트가 발생할 때 펌웨어 진단의 실행 여부를 제어합니다.

`diag-switch?`를 `true`로 설정하지 않으면 표준 시스템 재설정 프로토콜은 POST 및 OpenBoot 진단을 수행하지 않습니다. 이 변수의 기본 설정은 `false`입니다. ASR은 펌웨어 진단을 통해 오류가 있는 장치를 감지하므로 ASR을 실행하려면 `diag-switch?`를 반드시 `true`로 설정해야 합니다. 자세한 지침은 30페이지의 "ASR을 활성화하는 방법"을 참조하십시오.

펌웨어 진단을 자동으로 작동시키는 재설정 이벤트(있는 경우)를 제어하려면 `obdiag-trigger` 및 `post-trigger`를 사용하십시오. 이러한 변수와 그 사용 방법에 대한 자세한 설명은 8페이지의 "POST 제어 진단" 및 15페이지의 "OpenBoot 진단 시험 제어"를 참조하십시오.

▼ ASR을 활성화하는 방법

1. 시스템 `ok` 프롬프트에서 다음을 입력합니다.

```
ok setenv diag-switch? true
ok setenv auto-boot? true
ok setenv auto-boot-on-error? true
```

2. `obdiag-trigger` 변수를 `power-on-reset`, `error-reset` 또는 `user-reset`으로 설정합니다. 예를 들어, 다음을 입력합니다.

```
ok setenv obdiag-trigger user-reset
```

3. 다음을 입력합니다.

```
ok reset-all
```

OpenBoot 구성 변수 `auto-boot?`가 `true`(기본값)로 설정되어 있으면, 매개변수 변경 사항이 시스템에 영구적으로 저장되고 시스템은 자동 부팅됩니다.

주 - 전면 패널의 전원/대기 단추를 눌러 시스템의 전원을 켜도 매개 변수의 변경 사항이 저장됩니다.

▼ ASR을 비활성화하는 방법

1. 시스템 ok 프롬프트에서 다음을 입력합니다.

```
ok setenv diag-switch? false
```

2. 다음을 입력합니다.

```
ok reset-all
```

매개 변수의 변경 사항이 시스템에 영구적으로 저장됩니다.

주 - 전면 패널의 전원/대기 단추를 눌러 시스템의 전원을 켜다 켜도 매개 변수의 변경 사항이 저장됩니다.

SunVTS 소프트웨어

이 장에서는 SunVTS에 대해 설명합니다. 이 장에서는 다음 항목을 다룹니다.

- 33페이지의 "SunVTS 소프트웨어 개요"
- 34페이지의 "SunVTS 테스트"
- 35페이지의 "SunVTS 소프트웨어 및 보안"
- 36페이지의 "SunVTS 소프트웨어 설치"
- 36페이지의 "SunVTS 소프트웨어 설명서 보기"

SunVTS 소프트웨어 개요

Netra 240 서버는 SunVTS 5.1 패치 세트 5(PS5) 소프트웨어와 그 호환 버전을 지원합니다.

SunVTS 소프트웨어(Sun Validation Test Suite)는 하드웨어 컨트롤러, 장치, 플랫폼의 구성 및 기능을 확인할 때 사용하는 온라인 진단 도구로서, Solaris OS에서 작동되며 다음과 같은 인터페이스를 가집니다.

- 명령행 인터페이스(CLI)
- 직렬(tty) 인터페이스

SunVTS 소프트웨어 스위트는 시스템과 주변 장치의 스트레스 테스트를 실행합니다. 네트워크를 통해서 SunVTS 소프트웨어 세션을 보고 제어할 수 있습니다. 원격 시스템에서 테스트 세션의 진행 상황을 보고 테스트 옵션을 변경할 수 있으며, 네트워크 상의 다른 시스템에 대한 테스트 기능도 모두 제어할 수 있습니다.

SunVTS 소프트웨어는 다음 3가지 테스트 모드로 실행할 수 있습니다.

- *Connection mode*(연결 모드)에서는 장치 컨트롤러의 존재 여부를 확인합니다. 보통 몇 분밖에 걸리지 않으며 시스템 연결의 "위생 검사"를 실행하는 좋은 방법입니다.
- *Functional mode*(기능 모드)에서는 선택한 특정 하위 시스템만 테스트합니다. 이것이 기본 모드입니다.

- *Auto Config mode*(자동 구성 모드)에서는 자동으로 모든 하위 시스템을 감지하고 다음 방법 중 하나로 이들을 테스트합니다.
 - *Confidence testing*(간략 테스트)에서는 모든 하위 시스템을 한 번씩 테스트하고 테스트를 종료합니다. 일반적인 구성의 시스템이라면 약 1 ~ 2시간이 소요됩니다.
 - *Comprehensive testing*(전체 테스트)에서는 모든 하위 시스템을 최고 24시간 동안 반복하여 테스트합니다.

SunVTS 소프트웨어는 동시에 여러 테스트를 수행하면서 시스템 자원을 많이 소모할 수 있기 때문에, 업무용 시스템에서 사용하는 경우에는 주의를 기울여야 합니다. SunVTS 소프트웨어의 **Comprehensive** 테스트 모드를 사용하여 시스템 안정성 테스트를 수행할 경우, 같은 시스템에서 동시에 여러 프로그램을 실행하지 마십시오.

SunVTS 소프트웨어로 테스트를 실행하려면 서버의 운영 체제가 **Solaris**여야 합니다. SunVTS 소프트웨어 패키지는 선택 사항이므로 사용자의 시스템에 설치되어 있지 않을 수도 있습니다. 자세한 지침은 35페이지의 "SunVTS 소프트웨어의 설치 여부 확인 방법"을 참조하십시오.

SunVTS 테스트

SunVTS 소프트웨어가 있으면 원격 연결된 서버에서도 테스트 세션을 보고 제어할 수 있습니다. 표 2-1에 몇 가지 이용 가능한 테스트를 정리했습니다.

표 2-1 SunVTS 소프트웨어 테스트

SunVTS 소프트웨어 테스트	설명
cputest	CPU를 테스트합니다.
disktest	로컬 디스크 드라이브를 테스트합니다.
dvdtest	DVD-ROM 드라이브를 테스트합니다.
n240atest	FRU ID, LED, 알람 릴레이의 알람 카드를 테스트합니다.
fputest	부동 소수점 처리 장치를 테스트합니다.
nettest	시스템 보드 상의 이더넷 하드웨어 및 모든 선택적 PCI 카드의 네트워킹 하드웨어를 테스트합니다.
netlbttest	이더넷 어댑터의 패킷 송수신 여부를 확인하기 위한 루프백을 테스트합니다.
pmem	물리적 메모리(읽기 전용)를 테스트합니다.
sutest	서버의 온보드 직렬 포트를 테스트합니다.
vmem	가상 메모리(스왑 파티션 및 물리적 메모리의 혼합)를 테스트합니다.

SunVTS 소프트웨어 테스트	설명
env6test	환경 제어 장치를 테스트합니다.
ssptest	ALOM 하드웨어 장치를 테스트합니다.
i2c2test	I ² C 장치의 올바른 작동 여부를 테스트합니다.

SunVTS 소프트웨어 및 보안

SunVTS 소프트웨어 설치 과정에서 사용자는 기본 보안 모드와 Sun Enterprise Authentication Mechanism(SEAM) 보안 중에서 하나를 선택해야 합니다. 기본 보안은 SunVTS 소프트웨어 설치 디렉토리에 들어 있는 로컬 보안 파일을 사용하여 SunVTS 소프트웨어에 대한 사용자, 그룹, 호스트의 사용 권한을 제한합니다. SEAM 보안은 표준 네트워크 인증 프로토콜인 Kerberos를 기반으로 하며, 네트워크 트랜잭션을 위한 보안 사용자 인증, 데이터 무결성 및 비밀 유지 기능이 있습니다.

SEAM 보안을 이용하려면 네트워크 환경에 SEAM 클라이언트와 서버 소프트웨어를 설치하고 Solaris 및 SunVTS 소프트웨어를 올바르게 구성해야 합니다. SEAM 보안을 사용하지 않을 계획이라면 SunVTS 소프트웨어 설치 중에 SEAM 옵션을 선택하지 마십시오.

설치 중에 보안 옵션을 잘못 선택하거나 선택한 보안 옵션을 잘못 설정할 경우 SunVTS 소프트웨어 테스트가 실행되지 않을 수 있습니다. 자세한 내용은 *SunVTS User's Guide* 및 SEAM 소프트웨어와 함께 제공된 지침을 참조하십시오.

▼ SunVTS 소프트웨어의 설치 여부 확인 방법

- 다음을 입력합니다.

```
# pkginfo -l SUNWvts
```

- SunVTS 소프트웨어가 설치되어 있다면 패키지 정보가 표시됩니다.
- SunVTS 소프트웨어가 없으면 다음과 같은 오류 메시지가 표시됩니다.

```
ERROR: information for "SUNWvts" was not found
```

SunVTS 소프트웨어 설치

SunVTS 소프트웨어는 기본적으로 Netra 240 서버에 설치되어 있지 않습니다. 그러나 Solaris OS 부록 CD에서 이 프로그램을 사용할 수 있으며, 다음 웹 사이트에서 최신 버전을 다운로드할 수 있습니다.

<http://www.sun.com/oem/products/vts/>

주 - Netra 240 서버는 SunVTS 5.1 패치 세트 5(PS5) 소프트웨어와 그 호환 버전을 지원합니다.

SunVTS 소프트웨어 사용에 대한 자세한 내용은 사용 중인 Solaris 소프트웨어 버전에 해당하는 SunVTS 설명서를 참조하십시오. 또한 위 웹 사이트에서 SunVTS 소프트웨어에 대한 추가 설명과 설치 지침을 볼 수 있습니다.

SunVTS 소프트웨어 설명서 보기

SunVTS 소프트웨어에 대한 여러 설명서는 모든 Solaris 미디어 키트에 포함된 소프트웨어 부록 CD에 들어 있으며, <http://docs.sun.com>에서도 볼 수 있습니다.

또한, 다음과 같은 SunVTS 소프트웨어 설명서도 참조할 수 있습니다.

- *SunVTS User's Guide*에는 SunVTS 진단 소프트웨어의 설치, 구성 및 실행 방법이 나와 있습니다.
- *SunVTS Quick Reference Card*에서는 SunVTS 인터페이스의 사용 방법을 다룹니다.
- *SunVTS Test Reference Manual*에는 SunVTS 테스트가 각각 자세히 설명되어 있습니다.

Advanced Lights Out Manager

이 장에서는 Sun™ ALOM(Advanced Lights Out Manager) 소프트웨어를 소개합니다. 이 장에서는 다음 항목을 다룹니다.

- 37페이지의 "Advanced Lights Out Manager 개요"
- 38페이지의 "ALOM 포트"
- 39페이지의 "admin 암호 설정"
- 39페이지의 "기본 ALOM 기능"
- 41페이지의 "자동 서버 재시작"
- 41페이지의 "환경 모니터링 및 제어"

Advanced Lights Out Manager 개요

Netra 240 서버는 Sun Advanced Lights Out Manager가 설치된 상태로 판매됩니다. 시스템 콘솔은 기본적으로 ALOM으로 지정되어 있으며, 시스템을 시작하면 서버 콘솔 정보가 표시됩니다.

ALOM은 직렬 연결(SERIAL MGT 포트 사용) 또는 이더넷 연결(NET MGT 포트 사용)을 통해 서버를 감시하고 제어하는 소프트웨어입니다. 이더넷 연결 구성 방법에 대해서는 *Netra 240 서버용 Sun ALOM(Advanced Lights Out Manager) 소프트웨어 사용 설명서 (817-5007-11)*를 참조하십시오.

주 - SERIAL MGT라고 표시된 ALOM 직렬 포트는 서버 관리에만 이용됩니다. 일반 용도의 직렬 포트가 필요한 경우에는 10101이라고 표시된 직렬 포트를 사용하십시오.

하드웨어 고장이나 서버 또는 ALOM과 관련된 각종 이벤트를 전자 우편으로 통지하도록 ALOM을 구성할 수 있습니다.

ALOM 회로는 서버의 대기 전원을 사용하기 때문에 다음과 같은 장점을 가집니다.

- ALOM은 서버의 전원을 연결하는 순간부터 전원 케이블을 뽑을 때까지 계속 작동합니다.
- 서버의 운영 체제가 오프라인으로 전환되어도 ALOM 펌웨어 및 소프트웨어는 계속 작동합니다.

표 3-1은 ALOM이 감시하는 구성 요소와 소프트웨어가 제공하는 각 구성 요소에 대한 정보를 나열합니다.

표 3-1 ALOM이 감시하는 구성 요소

구성 요소	제공되는 정보
하드 드라이브	설치 여부 및 상태
시스템 및 CPU 팬	속도 및 상태
CPU	설치 여부, 온도, 열과 관련된 각종 경고 또는 오류 상태
전원 공급 장치	설치 여부 및 상태
시스템 온도	주변 온도, 열과 관련된 각종 경고 또는 오류 상태
서버 전면 패널	로터리 스위치의 위치 및 LED 상태
전압	상태 및 임계값
SCSI 및 USB 회로 차단기	상태
접점 릴레이 알람	상태

ALOM 포트

기본 관리 포트에는 SERIAL MGT라는 표시가 있습니다. 이 포트에는 RJ-45 커넥터를 사용하며 서버 관리 전용으로, 외부 콘솔과의 연결은 ASCII 방식으로만 가능합니다. 이 포트는 서버를 처음으로 작동시킬 때 사용합니다.

10101이라고 표시된 또 하나의 포트는 일반적인 직렬 데이터 전송에 이용됩니다. 이 포트는 DB-9 커넥터를 사용합니다. 핀 배치에 관한 내용은 *Netra 240 서버 설치 설명서*(부품 번호 817-4998-11)를 참조하십시오.

이 서버에는 또 NET MGT라고 표시된 10Base-T 이더넷 관리 도메인 인터페이스가 하나 있습니다. 이 포트를 사용하려면 ALOM을 구성해야 합니다. 자세한 내용은 *Netra 240 서버용 Sun ALOM(Advanced Lights Out Manager) 소프트웨어 사용 설명서*(부품 번호 817-5007-11)를 참조하십시오.

admin 암호 설정

전원을 켜면 ALOM 소프트웨어가 시작되고 `sc>` 프롬프트가 나타납니다. 여기서는 사용자 권한이 필요 없는 명령을 실행할 수 있습니다. 이 명령의 사용 방법은 *Netra 240 서버용 Sun ALOM(Advanced Lights Out Manager) 소프트웨어 사용 설명서*(부품 번호 817-5007-11)를 참조하십시오. 사용자 권한이 있어야만 가능한 명령을 실행하려고 하면, `admin` 암호를 설정하라는 메시지가 표시됩니다.

- **프롬프트가 나타나면 admin 암호를 설정합니다.**

암호는 다음 조건에 부합해야 합니다.

- 2자 이상의 영문자 포함
- 1자 이상의 숫자 또는 특수 문자 포함
- 총 6자 이상 8자 이하

암호를 설정하면 `admin`에게 모든 권한이 부여되어 모든 ALOM CLI 명령을 실행할 수 있습니다. 이후로 ALOM을 시작하는 사용자에게는 관리자 암호로 로그인하라는 메시지가 표시됩니다.

기본 ALOM 기능

여기서는 ALOM의 기본 기능 몇 가지를 소개합니다. 전체적인 설명은 *Netra 240 서버용 Sun ALOM(Advanced Lights Out Manager) 소프트웨어 사용 설명서*(부품 번호 817-5007-11) 및 *Netra 240 서버 릴리스 노트* (817-3142-xx)를 참조하십시오.

▼ ALOM 프롬프트로 전환하는 방법

- 명령 프롬프트에서 #. 키를 순서대로 입력합니다.

```
# #.
```

주 - ALOM 프롬프트로 전환하면 admin이라는 사용자 ID로 로그인됩니다. 39페이지의 "admin 암호 설정"를 참조하십시오.

▼ 서버 콘솔 프롬프트로 전환하는 방법

- 다음을 입력합니다.

```
sc> console
```

동시에 여러 명의 ALOM 사용자가 서버 콘솔에 연결할 수 있지만, 콘솔 입력이 허용되는 것은 그 중 한 명뿐입니다.

읽기 권한을 가진 다른 사용자가 로그인되어 있는 경우에 console 명령을 실행하면 다음 메시지가 표시됩니다.

```
sc> Console session already in use. [view mode]
```

▼ 다른 사용자로부터 콘솔 쓰기 권한 가져오기

- 다음을 입력합니다.

```
sc> console -f
```

자동 서버 재시작

주 - 자동 시스템 복구(Automatic System Recovery) 기능은 Netra 240 서버에도 있는 자동 서버 재시작(Automatic Server Restart) 기능과 다릅니다.

자동 서버 재시작 기능은 ALOM의 구성 요소로서, 작동 중인 Solaris 운영 체제를 감시하고 기본적으로 파일 시스템을 동기화하며 서버가 중지되었을 때 이를 재시작합니다.

ALOM은 감시(watchdog) 프로세스를 사용하여 오직 커널만 감시합니다. 프로세스가 중지되어도 커널은 그대로 실행 중이라면 ALOM은 서버를 재시작하지 않습니다. 감시 확인 간격 및 감시 시간 초과 등 ALOM의 감시 프로세스 매개 변수는 사용자가 설정할 수 없습니다.

커널이 중지되고 감시 시간을 초과한 경우, ALOM은 해당 이벤트를 보고 및 기록하고 다음 세 가지 조치 중 하나를 실시합니다. 이 조치는 사용자가 설정 가능합니다.

- **xir**-서버가 파일 시스템과 동기화한 뒤에 재시작하도록 하는 기본 조치입니다. 시스템이 중지되면 ALOM은 15분 후에 강제 재설정으로 되돌아갑니다.
- **Reset**-강제 재설정(하드 리셋)을 말하며, 시스템은 신속하게 복구되지만 작동 중지에 관한 진단 정보가 저장되지 않기 때문에 가장 심각한 손상으로 이어질 수 있습니다
- **None**-감시 프로세스의 시간 초과가 보고된 후 시스템을 영원히 작동 중지 상태로 내버려둡니다.

자세한 내용은 *Netra 240 서버용 Sun ALOM(Advanced Lights Out Manager) 소프트웨어 사용 설명서*(부품 번호 817-5007-11)의 `sys_autorestart` 부분을 참조하십시오.

자동 시스템 복구(ASR, Automatic System Recovery)의 실행 지침은 1장을 참조하십시오.

환경 모니터링 및 제어

Netra 240 서버의 환경 모니터링 하위 시스템은 다음과 같은 상황에서 서버와 그 구성 요소를 보호합니다.

- 과열
- 시스템 환기 부족
- 누락되거나 잘못 구성된 구성 요소로 작동
- 전원 공급 장치 고장
- 내부 하드웨어 장애

모니터링 및 제어 기능은 ALOM 펌웨어가 담당합니다. 따라서 시스템이 중단되거나 부팅할 수 없는 경우에도 모니터링 기능은 계속 작동합니다. 또한 ALOM 펌웨어를 통해 시스템을 모니터링하면 CPU 및 메모리 자원을 운영 체제와 응용 프로그램에 추가로 할당할 수 있습니다.

환경 모니터링 하위 시스템은 업계 표준인 I²C 버스를 사용합니다. I²C 버스는 단순한 2선 직렬 버스로서 시스템 전체의 온도 감지기, 팬, 전원 공급 장치, 상태 LED, 전면 패널의 시스템 제어 로터리 스위치 등을 감시하고 제어하는 데 사용됩니다.

서버에는 세 개의 온도 감지기가 있어서 서버 주변의 기온과 두 CPU의 다이 온도를 감시합니다. 모니터링 하위 시스템은 각 감지기를 확인하여 기록된 온도를 바탕으로 모든 과열 또는 이상 저온 상태를 보고하고 조치를 취합니다. 또 다른 I²C 감지기는 구성 요소의 존재 여부와 장애를 감지합니다.

하드웨어와 소프트웨어 모두 장치 내부 온도가 사전에 지정한 "안전 작동" 범위를 벗어나지 않도록 합니다. 저온 경고 임계값 이하 또는 고온 경고 임계값 이상의 온도가 감지기에 감지되면, 모니터링 하위 시스템 소프트웨어는 전면 및 후면 패널에 있는 시스템 수리 필요 LED에 불이 켜지게 합니다. 이상 온도가 지속되어 저온/고온 정상 종료(소프트 종료) 임계값에 도달하면 시스템은 정상적으로 종료됩니다. 만일 저온/고온 강제 종료(하드 종료) 임계값에 도달하면 시스템은 강제 종료를 시작합니다.

오류 및 경고 메시지가 시스템 콘솔에 표시되고 /var/adm/messages 파일에 기록되며, 시스템이 자동으로 종료된 이후에도 수리 필요 LED가 계속 켜져 있어 문제 진단에 도움이 됩니다.

시스템 콘솔로 전송하고 /var/adm/messages 파일에 기록하는 메시지의 유형은 sc_clieventlevel 및 sys_eventlevel이라는 ALOM 사용자 변수를 어떻게 설정했는가에 따라 달라집니다. 이러한 변수의 설정 방법은 *Netra 240 서버용 Sun ALOM(Advanced Lights Out Manager) 소프트웨어 사용 설명서(817-5007-11)*를 참조하십시오.

표 3-2 Netra 240 서버 인클로저 온도 임계값

온도 임계값	온도	서버 조치
저온, 강제 종료	-11° C	서버가 시스템 강제 종료를 시작합니다.
저온, 정상 종료	-9° C	서버가 시스템을 정상 종료합니다.
저온 경고	-7° C	서버가 전면과 후면 패널에 있는 시스템 수리 필요 LED 표시등을 켭니다.
고온 경고	57° C	서버가 전면과 후면 패널에 있는 시스템 수리 필요 LED 표시등을 켭니다.
고온, 정상 종료	60° C	서버가 시스템을 정상 종료합니다.
고온, 강제 종료	63° C	서버가 시스템 강제 종료를 시작합니다.

모니터링 하위 시스템에는 네 가지 시스템 환기 장치의 장애를 감지하는 기능도 있습니다. 환기 장치가 고장나면 모니터링 하위 시스템이 이를 감지하여 시스템 콘솔에 오류 메시지를 전송하고, 그 메시지를 `/var/adm/messages` 파일에 기록하며, 수리 필요 LED에 불이 켜지게 합니다.

전원 하위 시스템도 비슷한 방식으로 모니터링됩니다. 모니터링 하위 시스템은 가끔씩 전원 공급 상태를 확인하여 각 공급 장치의 존재 여부 및 출력과 입력을 알려줍니다.

전원 공급 장치의 문제가 감지되면 오류 메시지가 시스템 콘솔로 전송되고 `/var/adm/messages` 파일에 기록됩니다. 또한 각 전원 공급 장치의 LED가 켜져 고장임을 나타냅니다. 수리 필요 LED가 켜져 시스템 장애를 표시하고, ALOM 콘솔 알람에 전원 공급 장치의 고장이 기록됩니다.

전원 하위 시스템과 팬 속도의 경고 임계값을 표시하는 ALOM 명령은 `showenvironment`입니다. 이 명령의 사용 방법은 *Netra 240 서버용 Sun ALOM(Advanced Lights Out Manager) 소프트웨어 사용 설명서*(부품 번호 817-5007-11)를 참조하십시오.

알람 릴레이 출력 API

이 부록에는 알람의 상태를 get/set하는 방법을 보여주는 샘플 프로그램(코드 예제 A-1)이 나와 있습니다. 이 응용 프로그램은 LOMIOCALSTATE ioctl 명령으로 알람의 상태를 알아보고 LOMIOCALCTL ioctl 명령으로 각각의 알람을 설정합니다. 알람 표시등에 관한 자세한 내용은 *Netra 240 Server Service Manual(817-2699-xx)*을 참조하십시오.

코드 예제 A-1 알람 상태의 get/set 작업을 위한 예제 프로그램

```
#include <sys/types.h>
#include <string.h>
#include <stdlib.h>
#include <sys/unistd.h>
#include <fcntl.h>
#include "lom_io.h"

#define ALARM_INVALID    -1
#define LOM_DEVICE      "/dev/lom"

static void usage();
static void get_alarm(const char *alarm);
static int set_alarm(const char *alarm, const char *alarmval);
static int parse_alarm(const char *alarm);
static int lom_ioctl(int ioc, char *buf);
static char *get_alarmval(int state);
static void get_alarmvals();

main(int argc, char *argv[])
{
    if (argc < 3) {
        usage();
        if (argc == 1)
            get_alarmvals();
        exit (1);
    }
}
```

```

#include <sys/types.h>
}

if (strcmp(argv[1], "get") == 0) {
    if (argc != 3) {
        usage();
        exit (1);
    }

    get_alarm(argv[2]);
}
else
if (strcmp(argv[1], "set") == 0) {
    if (argc != 4) {
        usage();
        exit (1);
    }
    set_alarm(argv[2], argv[3]);
} else {
    usage();
    exit (1);
}
}

static void
usage()
{
    printf("사용법: alarm [get|set] [crit|major|minor|user] [on|off]\n");
}

static void
get_alarm(const char *alarm)
{
    ts_aldata_t    ald;
    int altype = parse_alarm(alarm);
    char *val;

    if (altype == ALARM_INVALID) {
        usage();
        exit (1);
    }

    ald.alarm_no = altype;
    ald.alarm_state = ALARM_OFF;

    lom_ioctl(LOMIOCALSTATE, (char *)&ald);
}

```

```

#include <sys/types.h>
if ((ald.alarm_state != ALARM_OFF) &&
    (ald.alarm_state != ALARM_ON)) {
    printf("유효하지 않은 값: %d\n", ald.alarm_state);
    exit (1);
}

printf("알람 %s = %s\n", alarm, get_alarmval(ald.alarm_state));
}

static int
set_alarm(const char *alarm, const char *alarmstate)
{
    ts_aldata_t    ald;
    int alarmval = ALARM_OFF, altype = parse_alarm(alarm);

    if (altype == ALARM_INVALID) {
        usage();
        exit (1);
    }

    if (strcmp(alarmstate, "on") == 0)
        alarmval = ALARM_ON;
    else
        if (strcmp(alarmstate, "off") == 0)
            alarmval = ALARM_OFF;
        else {
            usage();
            exit (1);
        }

    ald.alarm_no = altype;
    ald.alarm_state = alarmval;

    if (lom_ioctl(LOMIOCALCTL, (char *)&ald) != 0) {
        printf("%s 알람을 %s로 설정하는 데 실패하였습니다.\n", alarm, alarmstate);
        return (1);
    } else {
        printf("%s 알람을 %s로 설정했습니다.\n", alarm, alarmstate);
        return (1);
    }
}

static int
parse_alarm(const char *alarm)
{

```

```
#include <sys/types.h>
int altype;

if (strcmp(alarm, "crit") == 0)
    altype = ALARM_CRITICAL;
else
if (strcmp(alarm, "major") == 0)
    altype = ALARM_MAJOR;
else
if (strcmp(alarm, "minor") == 0)
    altype = ALARM_MINOR;
else
if (strcmp(alarm, "user") == 0)
    altype = ALARM_USER;
else {
    printf("유효하지 않은 알람 값: %s\n", alarm);
    altype = ALARM_INVALID;
}

return (altype);
}

static int
lom_ioctl(int ioc, char *buf)
{
    int fd, ret;

    fd = open(LOM_DEVICE, O_RDWR);

    if (fd == -1) {
        printf("장치 열기 오류: %s\n", LOM_DEVICE);
        exit (1);
    }

    ret = ioctl(fd, ioc, (void *)buf);

    close (fd);

    return (ret);
}

static char *
get_alarmval(int state)
{
    if (state == ALARM_OFF)
```


코드 예제 A-1

알람 상태의 get/set 작업을 위한 예제 프로그램 (계속)

```
#include <sys/types.h>
    return ("off");
else
    if (state == ALARM_ON)
        return ("on");
    else
        return (NULL);
}
static void
get_alarmvals()
{
    get_alarm("crit");
    get_alarm("major");
    get_alarm("minor");
    get_alarm("user");
}
```


색인

A

Advanced Lights Out Manager

ALOM 참조

ALOM

LED 상태 표시등 4

개요 37

기본 기능 39

암호 설정 39

자동 서버 재시작 41

진단 도구 2

포트 38

환경 모니터링 하위 시스템 41

ASR 28

auto-boot? 변수 9

B

BIST, 내장 자가 검사 참조

C

CPU

정보 표시 23

클럭 속도 23

D

diag-level 변수 9, 15

diag-script 변수 9

diag-switch? 변수 8, 9

F

FRU 22 - 23

I

I²C 감지기 42

I²C 버스 42

IDE 버스 12

IDE(Integrated Drive Electronics), IDE 버스 참조

input-device 변수 9

L

LED, 진단 도구 2

M

major, 알람 표시등 6

message

POST, 오류 8

오류 해석 17

minor, 알람 표시등 7

O

- obdiag-trigger 변수 9
- OpenBoot PROM 매개변수, diag-level 변수 8
- OpenBoot 구성 변수
 - 설명 9
 - 키워드 9
- OpenBoot 명령
 - probe-ide 12
 - probe-scsi 및 probe-scsi-all 11
 - run 14
 - show-devs 13
 - 진단 도구 2
- OpenBoot 진단 14
 - 시작 14
 - 시험 제어 15
 - 진단 도구 2
- OpenBoot 진단 시험
 - ok 프롬프트 16
 - test 명령 16
 - test-all 명령 17
 - 오류 메시지, 해석 17
 - 하드웨어 장치 경로 16
- output-device 변수 10

P

- POST
 - 오류 메시지 8
 - 진단 도구 2
 - 진단 시작 10
 - 진단, 제어 8
- POST(전원 인가 후 자가 검사)
 - POST 참조
- post-trigger 변수 9
- probe-ide 명령(OpenBoot) 12
- probe-scsi 및 probe-scsi-all 명령(OpenBoot) 11
- prtconf 명령, Solaris 19
- prtdiag 명령, Solaris 20
- prtfru 명령, Solaris 22
- psrinfo 명령, Solaris 23

S

- SCSI 장치, 문제 진단 11
- SEAM 35
- show-devs 명령, OpenBoot 13
- showrev 명령, Solaris 24
- Solaris OS
 - SunVTS 34
 - 장치 트리 19
- Solaris 명령
 - prtconf 19
 - prtdiag 20
 - prtfru 22
 - psrinfo 23
 - showrev 24
 - 진단 도구 2
- Sun Enterprise Authentication Mechanism, SEAM
 - 참조
- Sun Validation Test Suite
 - SunVTS 참조
- SunVTS 33 - 36
 - SEAM 보안 35
 - Solaris OS 34
 - 개요 33
 - 기본 보안 35
 - 선택적 소프트웨어 패키지 34
 - 설명서 36
 - 설치 여부 확인 35
 - 소프트웨어, 테스트 모드 33
 - 이용 가능한 테스트 34
 - 인터페이스 33
 - 장착 36
 - 진단 도구 2
 - 호환 버전 33, 36

T

- test 명령(OpenBoot 진단 시험) 16
- test-all 명령(OpenBoot 진단 시험) 17
- test-args 변수 16
 - 시험 사용자 정의 15
 - 키워드 16

U

USB 장치, OpenBoot 진단 자가 시험 17
user, 알람 표시등 7

W

watch-net 27
WWN(World Wide Name)(probe-scsi) 12

ㄱ

개정, 하드웨어 및 소프트웨어, showrev로 표시 24
과열 상태 21
과열, 모니터링 하위 시스템 42

ㄴ

내장 자가 검사, test-args 변수 16
논리적 장치 번호(probe-scsi) 12

ㄹ

로그 파일 18
 시스템 메시지 18
 오류 메시지 18
로케이터 LED 4
 on 5
 꺼짐 5
 상태 5
루프 ID(probe-scsi) 12
릴레이 상태
 정상 열림(NO) 7
 정상 종료(NC) 7

ㅁ

모니터링 하위 시스템
 과열 42
 저온 42
문제 해결 도구 2

ㅂ

비상, 알람 표시등 6

ㅅ

서버 상태 표시등, 전면 및 후면 4
서버 프롬프트
 ALOM(Advanced Lights Out Manager) 프롬프트 3
 OpenBoot 프롬프트 3
 Solaris 소프트웨어 수퍼유저 프롬프트 3
소프트웨어 개정판, showrev로 표시 24
수리 필요 LED 42
시스템 구성 카드 8
시스템 메모리, 크기 결정 19
시스템 상태 LED
 LED 참조
 환경 오류 표시기 43

ㅇ

알람
 릴레이 출력 API 45 - 49
 상태 보기(get) 45 - 49
 상태 설정(set) 45 - 49
알람 모드
 알람 상태 6
 알람 표시등 6
알람 상태, 접점 6
알람 표시등 6
 major 6
 minor 7
 user 7
 위험 6
오류 메시지
 OpenBoot 진단 시험 17
 OpenBoot 진단, 해석 17
 로그 파일 42
 전원 관련 43
온도 감지기 42

ㄹ

- 자동 서버 재시작 41
- 자동 시스템 복구는 ASR을 참조하십시오. 28
- 장치 경로, 하드웨어 13, 16
- 장치 트리, Solaris 소프트웨어, 표시 19
- 재설정 이벤트, 종류 9
- 저온, 모니터링 하위 시스템 42
- 전원 공급 장치, 고장 모니터링 43
- 정상 열림(NO), 릴레이 상태 7
- 정상 종료(NC), 릴레이 상태 7
- 중양 처리 장치, CPU 참조
- 진단
 - OpenBoot 14
 - POST 8
 - Solaris OS 18
 - SunVTS 33
- 진단 도구
 - ALOM 2
 - LED 2
 - OpenBoot 명령 2
 - OpenBoot 진단 2
 - POST 2
 - Solaris 소프트웨어 명령 2
 - SunVTS 2
- 진단 시험, 실행 안함 9

ㄴ

- 클럭 속도, CPU 23

ㄷ

- 패치, 설치된, showrev 24
- 프로세서 속도, 표시 23

ㄹ

- 하드웨어 개정판, showrev로 표시 24
- 하드웨어 장치 경로 13, 16
- 호스트 어댑터(probe-scsi) 12
- 환경 모니터링 하위 시스템 41