

# Netra™ CT 900 서버 관리 및 참조 설명서

---

Sun Microsystems, Inc.  
[www.sun.com](http://www.sun.com)

부품 번호 820-0574-10  
2007년 1월, 개정판 A

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

Copyright 2007 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 모든 권리는 저작권자의 소유입니다.

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

본 제품 또는 설명서는 사용, 복사, 배포 및 역검과일을 제한하는 라이선스 하에서 배포됩니다. 본 제품 또는 설명서의 어떠한 부분도 Sun 및 해당 사용권자의 사전 서면 승인 없이는 형식이나 수단에 상관없이 재생이 불가능합니다.

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

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

Sun, Sun Microsystems, Sun 로고, Java, AnswerBook2, docs.sun.com, Netra 및 Solaris는 미국 및 기타 국가에서 Sun Microsystems, Inc.의 상표 또는 등록 상표입니다.

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

PICMG 및 PICMG 로고, AdvancedTCA 및 AdvancedTCA 로고는 PCI Industrial Computers Manufacturers Group의 등록 상표입니다.

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

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

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



재활용  
가능



Adobe PostScript

# 목차

---

머리말 xvii

## 1. 소개 1

Netra CT 900 서버 소프트웨어 1

선반 관리자 소개 5

ATCA의 지능형 플랫폼 관리 개요 5

선반 관리자 및 선반 관리 카드 7

선반 관리자 기능 7

선반 관리자 스위치오버 8

스위치오버 상세 정보 9

시스템 관리자 인터페이스 옵션 11

시스템 관리 작업 11

물리 주소-논리 슬롯 매핑 12

## 2. 시스템 구성 13

선반 관리 카드 액세스 14

U-Boot 설정 15

U-Boot 인터페이스 16

U-Boot 환경 변수 16

환경 변수에 값 할당 19

선반 관리자의 구성 환경 변수 20

선반 관리 카드 이더넷 포트 구성	21
첫 번째 이더넷 인터페이스 사용	21
첫 번째 네트워크 인터페이스에 추가 IP 주소 할당	21
RMCP 주소 전파	22
두 번째 이더넷 인터페이스 사용	22
중복 통신을 위한 이중 USB 네트워크 인터페이스 사용	23
기본 ShMM 네트워크 매개 변수 변경	24
▼ 기본 ShMM 네트워크 매개 변수를 변경하는 방법	24
선반 관리자 구성 파일 설정	28
상세 표시 수준 설명	43
날짜 및 시간 설정	44
시간 서버에서 날짜 및 시간 가져오기	45
선반 관리 카드에서 사용자 계정 설정	46
▼ RMCP 액세스를 위한 사용자 계정을 추가하는 방법	46
사용자 이름 제한	47
암호	47
선반 관리자에서 OpenHPI 구성	47
/etc/openhpi.conf 파일	47
▼ /etc/openhpi.conf 파일을 수정하는 방법	48
/etc/snmpd.conf 파일	49
액세스 제어	49
SNMPv3 구성	51
트랩 설정 및 정보 대상	52
▼ /etc/snmpd.conf 파일을 업데이트하는 방법	52
<b>3. 시스템 관리</b>	<b>53</b>
IPMI LAN 인터페이스	53
IPMI 명령	54

선반 관리자 명령줄 인터페이스	54
명령줄 인터페이스 시작	55
CLI 명령	56
시스템 모니터링	60
보드 및 IPMC 정보 표시	60
FRU 정보 표시	65
IPMI FRU 정보 레이아웃	65
환경 FRU	66
블레이드 FRU	66
예	66
선반 정보 표시	73
예	73
선반 관리자 다시 초기화	79
U-Boot 환경 다시 초기화	79
▼ U-Boot 환경 다시 초기화	79
파일 시스템 다시 초기화	80
로그인 암호 재설정	81
선반 관리 카드 재프로그래밍	82
안정적인 펌웨어 업그레이드 절차	82
플래시 분할	83
/var/upgrade 파일 시스템	84
안정적인 업그레이드 절차 상태 파일	85
안정적인 업그레이드 유틸리티	85
안정적인 업그레이드 유틸리티 사용 시나리오	91
안정적인 업그레이드 예	92
CPLD 프로그래밍	102
▼ ShMM에 대해 CPLD 이미지를 재프로그래밍하는 방법	102

노드 보드 콘솔에 연결 103

선반 관리자와 노드 보드 간의 콘솔 세션 설정 103

▼ 선반 관리자에서 콘솔 세션을 시작하는 방법 104

▼ 콘솔 세션을 종료하는 방법 105

수동으로 노드 보드 정상 종료 105

▼ 노드 보드를 종료하는 방법 105

## A. 선반 관리자 CLI 명령 109

activate 110

alarm 111

board 112

boardreset 114

busres 115

버스된 전자 키잉 관리 자원의 상태 표시 116

지정된 자원 해제 116

지정된 자원 잠금/잠금 해제 117

Bused Resource Control (Query) 명령 전송 118

자원의 소유자 설정 119

Bused Resource Control (Bus Free) 명령 전송 119

console 120

deactivate 121

debuglevel 122

exit | quit 123

fans 123

flashupdate 124

fru 125

frucontrol 127

frudata 128

frudatar 130

- frudataw 131
- fruinfo 132
- getfanlevel 134
- getfruledstate 134
- gethysteresis 137
- getipmbstate 138
- getlanconfig 139
  - auth\_support 141
  - auth\_enables 142
  - ip 143
  - ip\_source 143
  - mac 144
  - subnet\_mask 145
  - ipv4\_hdr\_param 145
  - pri\_rmcp\_port 146
  - sec\_rmcp\_port 146
  - arp\_control 147
  - arp\_interval 147
  - dft\_gw\_ip 148
  - dft\_gw\_mac 148
  - backup\_gw\_ip 149
  - backup\_gw\_mac 149
  - community 150
  - destination\_count 150
  - destination\_type 151
  - destination\_address 152
- getpefconfig 152
  - control 154

action\_control 155  
startup\_delay 156  
alert\_startup\_delay 156  
event\_filter\_count 157  
event\_filter 157  
event\_filter\_data1 158  
alert\_policy\_count 159  
alert\_policy 159  
system\_guid 160  
alert\_string\_count 161  
alert\_string\_key 161  
alert\_string 162  
oem\_filter\_count 162  
oem\_filter 163  
getsensoreventenable 164  
getthreshold | threshold 165  
help 167  
ipmc 169  
localaddress 171  
minfanlevel 172  
sel 173  
sensor 176  
sensordata 180  
sensorread 182  
session 184  
setextracted 185  
setfanlevel 185  
setfruledstate 186



- sethysteresis 188
- setipmbstate 188
- setlanconfig 189
  - auth\_enables 191
  - ip 192
  - subnet\_mask 192
  - ipv4\_hdr\_param 193
  - arp\_control 193
  - arp\_interval 194
  - dft\_gw\_ip 194
  - backup\_gw\_ip 195
  - community 195
  - destination\_type 196
  - destination\_address 196
- setlocked 197
- setpefconfig 198
  - control 199
  - action\_control 200
  - startup\_delay 201
  - alert\_startup\_delay 201
  - event\_filter 202
  - event\_filter\_data1 203
  - alert\_policy 204
  - system\_guid 204
  - alert\_string\_key 205
  - alert\_string 206
  - oem\_filter 206
- setsensoreventenable 207

setthreshold 209  
 shelf 211  
     선반 FRU 정보 표시 212  
     사용 가능한 최대 외부 전류 수정 217  
     최소 예상 작동 전압 수정 219  
     선반 관리자 제어 활성화 플래그 수정 221  
     선반 관리자 제어 비활성화 플래그 수정 226  
     최대 FRU 전원 용량 수정 228  
     다음 전원 켜기 전까지의 지연 229  
     FRU 활성화 준비 허용 수정 231  
     FRU 활성화 및 전원 설명자 순서 재정렬 232  
     선반 FRU 정보 새로 고침 234  
     선반 FRU 정보 저장 장치 업데이트 236  
 shelfaddress 236  
 shmstatus 237  
 showhost 238  
 showunhealthy 239  
 switchover 239  
 terminate 240  
 user 240  
     사용자 정보 표시 241  
     새 사용자 추가 242  
     사용자 삭제 243  
     사용자 활성화 및 비활성화 243  
     사용자 이름 수정 245  
     사용자의 암호 수정 246  
     지정된 사용자 및 지정된 채널의 채널 액세스 설정 수정 247  
 version 248

## **B. Sun OEM IPMI 명령 249**

Get Version 250

Set Boot Page 251

Get Boot Page 252

Set Front Panel Reset Button State 253

Get Front Panel Reset Button State 254

Set Ethernet Force Front Bit 255

Get Ethernet Force Front Bit 256

Get RTM Status 257



# 그림

---

그림 1-1	Netra CT 서버의 소프트웨어 및 하드웨어 인터페이스에 대한 논리적 표현	4
그림 1-2	ATCA 선반 예	6
그림 1-3	선반 관리자 스위치오버 신호	8
그림 2-1	선반 알람 패널 커넥터	14
그림 3-1	IPMI FRU 정보 레이아웃	66



## 표

---

표 1-1	시스템 관리자용 Netra CT 서버 소프트웨어	2
표 1-2	Netra CT 900 시스템 보드 액세스 방법	3
표 1-3	스위치오버를 지원하는 하드웨어 신호 및 인터페이스	9
표 1-4	물리 주소-논리 슬롯 매핑	12
표 2-1	기본 U-Boot 환경 변수	16
표 2-2	선반 관리자 구성 매개 변수	29
표 3-1	Sun OEM IPMI 명령	54
표 3-2	선반 관리자 CLI 명령 요약	56
표 3-3	플래시 분할 영역(16MB reliable_upgrade=y)	84
표 3-4	선반 관리자 CLI 콘솔 관련 명령	103
표 A-1	getlanconfig의 LAN 구성 매개 변수	139
표 A-2	PEF 구성 매개 변수	153
표 A-3	setlanconfig의 LAN 구성 매개 변수	190
표 A-4	setpefconf의 PEF 구성 매개 변수	198
표 A-5	shelf 명령의 매개 변수	212
표 B-1	Sun OEM IPMI 명령	249





# 머리말

---

Netra CT 900 서버 관리 및 참조 설명서에는 Netra™ CT 900 서버 시스템 관리자를 위한 구성 정보와 관리 정보가 포함되어 있습니다. 또한 선반 관리자 및 IPMI 명령 참조 정보도 제공합니다.

이 설명서는 UNIX® 명령 및 네트워크, PICMG® 3.x AdvancedTCA® 기본 사양, IPMI (Intelligent Platform Management Interface)에 대해 익숙한 사용자를 대상으로 합니다.

---

## 본 설명서의 구성

**1장**에서는 Netra CT 900 서버 소프트웨어에 대해 소개합니다.

**2장**에서는 시스템 구성에 대한 정보를 포함합니다.

**3장**에서는 시스템을 관리하는 방법에 대해 설명합니다.

**부록 A**에서는 각 선반 관리자 CLI(Command-Line Interface) 명령의 구문과 사용법에 대해 설명합니다.

**부록 B**에서는 Sun 고유의 OEM 정의 IPMI(Intelligent Platform Management Interface) 명령에 대해 설명합니다.

---

## UNIX 명령 사용

이 문서에는 시스템 종료, 시스템 부트, 장치 구성 등과 같은 UNIX 기본 명령과 절차에 대한 정보는 포함되어 있지 않을 수 있습니다. 이 정보에 대해서는 다음을 참조하십시오.

- 시스템과 함께 제공된 소프트웨어 설명서
- 다음 사이트에 있는 Solaris™ Operating System(Solaris OS) 설명서:  
<http://docs.sun.com>

---

## 셸 프롬프트

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

# 활자체 규약

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

\* 사용자가 사용하는 브라우저의 설정과 이 설정은 다를 수 있습니다.

## 관련 설명서

Netra CT 900 서버 설명서는 다음 표에 나열되어 있습니다. Important Safety Information for Sun Hardware Systems을 제외하고 여기에 나열된 모든 문서는 다음 사이트에서 온라인으로 볼 수 있습니다.

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

제목	부품 번호
Netra CT 900 Server Product Notes	819-1180
Netra CT 900 Server Overview	819-1174
Netra CT 900 Server Installation Guide	819-1175
Netra CT 900 Server Service Manual	819-1176
Netra CT 900 Server Switch Software Reference Manual	819-3774

제목	부품 번호
Netra CT 900 Server Safety and Compliance Manual	819-1179
Important Safety Information for Sun Hardware Systems (인쇄 버전만 해당)	816-7190-10

자세한 내용은 Solaris OS, OpenBoot™ PROM 펌웨어, Netra CP3010 보드, Netra CP3020 보드 및 Netra CP3060 보드 제품에 대한 설명서를 참조할 수 있습니다.

## 설명서, 지원 및 교육

Sun 기능	URL
설명서	<a href="http://www.sun.com/documentation/">http://www.sun.com/documentation/</a>
지원	<a href="http://www.sun.com/support/">http://www.sun.com/support/</a>
교육	<a href="http://www.sun.com/training/">http://www.sun.com/training/</a>

## Sun 기술 지원

본 제품과 관련하여 이 문서에 나와 있지 않은 기술 문제가 발생할 경우, 다음 URL을 참조하십시오.

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

## Sun은 여러분의 의견을 환영합니다.

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

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

아래와 같이 문서의 제목과 부품 번호를 함께 적어 보내주시기 바랍니다.

Netra CT 900 서버 관리 및 참조 설명서, 부품 번호 820-0574-10

# 1장

## 소개

---

이 장은 다음 절로 구성되어 있습니다.

- 1페이지의 "Netra CT 900 서버 소프트웨어"
- 5페이지의 "선반 관리자 소개"
- 11페이지의 "시스템 관리 작업"

---

## Netra CT 900 서버 소프트웨어

Netra CT 900 서버 소프트웨어는 다음을 포함합니다.

- 선반 관리자
- 운영 체제 및 응용 프로그램
- 펌웨어

---

주 - Advanced Telecom Computing Architecture®(ATCA)에서는 전자 통신 업계의 관행에 따라 선반이라는 용어를 채택했습니다. 일반적으로 새시라는 용어를 같은 의미로 사용해왔습니다.

---

소프트웨어에 대해서는 표 1-1에 설명되어 있고 그림 1-1에 하드웨어와 함께 논리적으로 표시되어 있습니다.

표 1-1 시스템 관리자용 Netra CT 서버 소프트웨어

범주	이름	설명
선반 관리	IPM Sentry 선반 관리자	선반 관리자 소프트웨어는 선반 관리 카드(ShMM)에서 실행되며 설치된 상태로 출하됩니다. 이 소프트웨어에서 제공하는 RMCP(Remote Management Control Protocol) 및 CLI를 통해 IPMI에 액세스하여 서버를 관리할 수 있습니다.
	명령줄 인터페이스(CLI)	CLI는 선반 관리자에 대한 내장 사용자 인터페이스입니다.
운영 체제 및 응용 프로그램	Solaris 운영 체제(Solaris OS)	Solaris OS는 Netra CP3010, Netra CP3020 및 CP3060 노드 보드와 같은 Sun 지원 ATCA 호환 노드 보드에서 실행됩니다. Solaris 10은 Netra 노드 보드에 옵션으로 사전 설치됩니다. Solaris 10 및 기타 Solaris OS 버전은 사용자가 직접 다운로드하여 설치할 수 있습니다.
	Monta Vista Carrier Grade Linux OS	또한 Netra CP3020은 Monta Vista Carrier Grade Linux OS를 실행할 수 있습니다.
펌웨어	OpenBoot PROM 펌웨어	부트를 제어하는 Sun 지원 노드 보드(예: Netra CP3010 보드)의 펌웨어이며 진단 기능을 포함합니다.
	U-Boot	POST(Power-On Self-Test)를 수행하고 선반 관리 카드 소프트웨어의 부트를 제어하는 선반 관리 카드의 펌웨어입니다.
	IPMC(Intelligent Platform Management Controller)	Netra CP3010 보드와 같은 Sun 지원 노드 보드에서 IPMI 제어기를 통해 통신할 수 있게 해주는 시스템 관리 제어기 펌웨어입니다.

Netra CT 900 서버에는 두 개의 선반 관리 카드(ShMM)가 있으며 특정 하드웨어 및 소프트웨어 이벤트가 발생할 경우 활성 선반 관리 카드에서 대기 선반 관리 카드로의 선반 관리 카드 페일오버를 제공합니다. 활성 선반 관리 카드는 미드프레인에 연결된 대부분의 구성 요소를 시스템 수준으로 구성하고 관리하는 데 사용됩니다. 대기(standby) 선반 관리 카드는 활성 선반 관리 카드에 대한 중복성 및 페일오버 기능을 제공합니다.

스위칭 패브릭 보드는 선반 관리 카드와 노드 보드를 내부적으로 연결하며 후면에 외부 연결을 위한 이더넷 포트가 있습니다.

Netra CP3010 노드 보드는 디스크와 같은 주변 장치를 지원 및 제공합니다. 또한 노드 보드는 사용자 응용 프로그램을 실행합니다. Netra CT 900 서버에서 각 노드 보드는 자체 운영 체제 복사본을 실행하므로 서버로 간주됩니다. 선반 관리 카드, 노드 보드, 스위칭 패브릭 보드 및 기타 시스템 FRU(Field-Replaceable Unit)가 하나의 시스템을 구성합니다.

주 - 별도로 명시되어 있지 않은 경우 노드 보드라는 용어는 Netra CP3010 보드와 같은 Sun 지원 ATCA CPU 보드를 나타냅니다.

PICMG® 3.x와 호환되는 타사 ATCA 노드 보드는 Netra CT 900 서버에서 사용될 수 있습니다. 이러한 보드에서는 Solaris OS를 실행할 필요가 없고 Netra CT 900 서버 시스템 관리 소프트웨어를 실행하지 않습니다. 따라서 Netra 노드 보드와 동일한 수준으로 관리될 수 없습니다.

표 1-2에는 다양한 보드에 액세스하는 방법이 요약되어 있습니다. 선반 관리 카드는 한 번에 22개의 세션(1개의 Tip 및 21개의 텔넷 연결)을 지원합니다.

표 1-2 Netra CT 900 시스템 보드 액세스 방법

보드	액세스 방법
선반 알람 패널(SAP)	<p>전면 포트에는 다음과 같은 포트가 있습니다.</p> <ul style="list-style-type: none"> <li>• RJ-45 DTE 커넥터가 있는 직렬(RS-232) 포트 두 개. 직렬 1 포트는 기본 활성화 카드인 위쪽 선반 관리 카드(ShMM1)에 대한 콘솔 연결입니다. 직렬 2 포트는 기본 백업 카드인 아래쪽 선반 관리 카드(ShMM2)에 대한 콘솔 연결입니다.</li> <li>• Telco 알람 커넥터(DB-15)</li> </ul>
스위치 보드(슬롯 7 및 8)	<p>텔넷 연결을 위한 전면 패널의 여러 이더넷 포트</p> <p>주 - 후면 스위치오버 모듈(RTM)은 후면에서 이러한 포트에 액세스하는 데 필요합니다. 후면 또는 전면 포트 커넥터를 사용할 수 있지만 두 커넥터를 동시에 사용할 수는 없습니다. 케이블을 두 포트 모두에 연결하면 전면 포트만 활성화됩니다.</p>
노드 보드(Sun 지원 CPU 보드) (슬롯 1-6 및 9-14)	<p>Netra CP3010 보드의 전면 패널에는 다음과 같은 포트가 있습니다.</p> <ul style="list-style-type: none"> <li>• Tip 또는 ASCII 터미널 연결을 위한 직렬 포트(콘솔) 두 개</li> <li>• 텔넷 연결을 위한 이더넷 포트 두 개</li> <li>• 4X 직렬 연결 SCSI(SAS) 포트 커넥터 한 개</li> </ul> <p>주 - RTM은 후면에서 이러한 포트에 액세스하는 데 필요합니다. 후면 또는 전면 포트 커넥터를 사용할 수 있지만 두 커넥터를 동시에 사용할 수는 없습니다. 케이블을 두 포트 모두에 연결하면 전면 포트만 활성화됩니다.</p> <p>Netra CP3020 및 CP3060 보드의 전면 패널에는 다음과 같은 포트가 있습니다.</p> <ul style="list-style-type: none"> <li>• Tip 또는 ASCII 터미널 연결을 위한 직렬 포트(콘솔) 한 개</li> <li>• 텔넷 연결을 위한 이더넷 포트 두 개</li> </ul> <p>자세한 내용은 Netra ATCA CPU 보드 설명서에서 해당 노드 보드를 참조하십시오.</p>
타사 노드 보드(슬롯 1-6 및 9-14)	타사 보드에 따라 다릅니다.

하드웨어 인터페이스에는 선반 관리 카드, 노드 보드 및 스위칭 패브릭 보드에 대한 IPMI (Intelligent Platform Management Interface), 기본 인터페이스 및 확장 인터페이스, 네트 워크 인터페이스 등이 포함되어 있습니다.

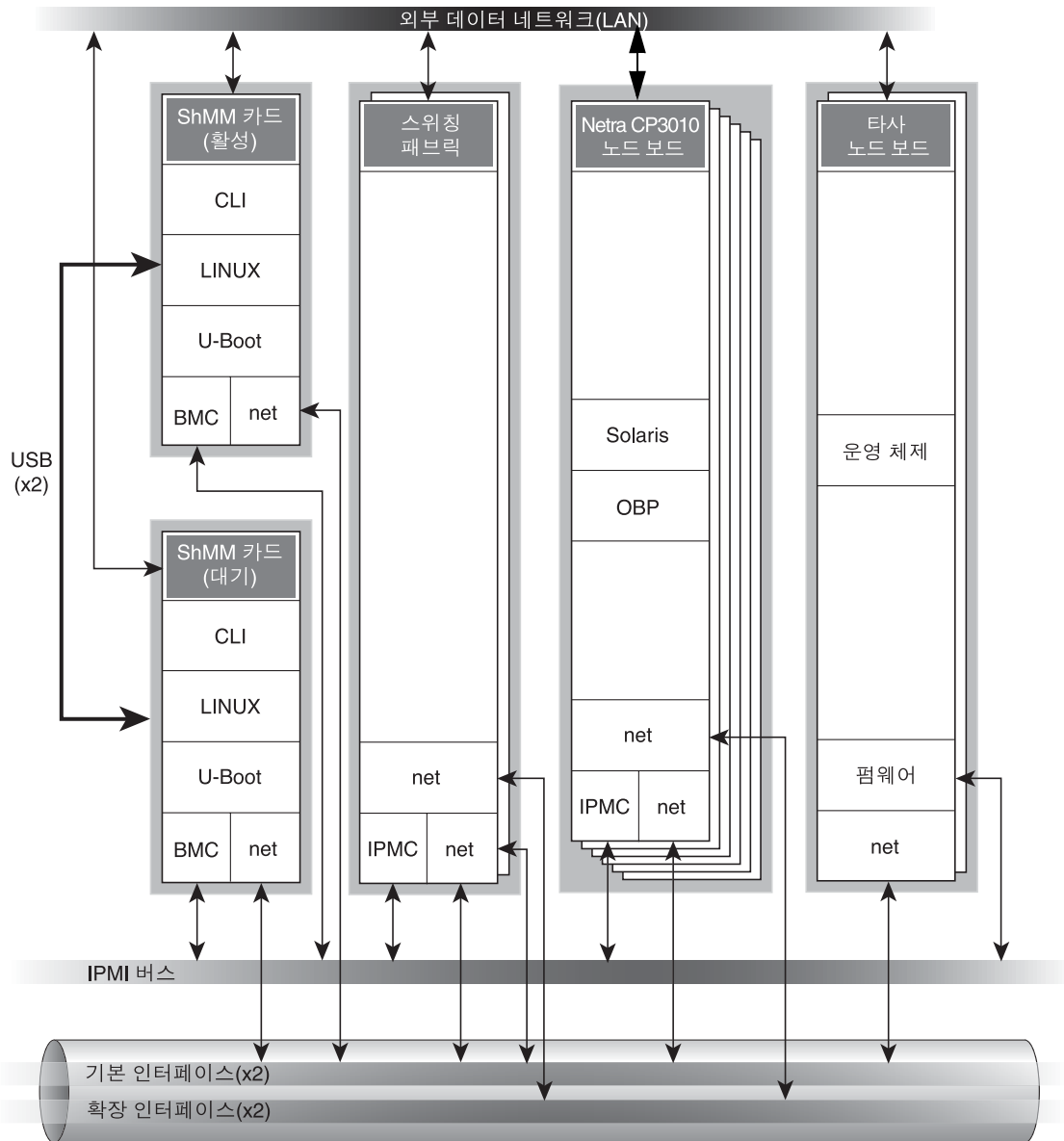


그림 1-1 Netra CT 서버의 소프트웨어 및 하드웨어 인터페이스에 대한 논리적 표현



---

## 선반 관리자 소개

선반 관리자는 ATCA 제품에 대한 선반 수준 관리 솔루션입니다. 선반 관리 카드에서는 ATCA 선반에서 선반 관리자를 실행하는 데 필요한 하드웨어를 제공합니다. 이 개요에서는 ATCA와 관련하여 사용되는 선반 관리 캐리어에 공통적인 선반 관리자 및 선반 관리 카드의 특징을 중점적으로 설명합니다.

## ATCA의 지능형 플랫폼 관리 개요

선반 관리자 및 선반 관리 카드는 ATCA와 같은 모듈형 플랫폼용으로 설계된 IPM (Intelligent Platform Management) 빌딩 블록이며, 동적 FRU 채우기 및 서비스 가용성 극대화에 중점을 두고 있습니다. IPMI 사양에는 이러한 플랫폼의 관리에 필요한 기본 정보가 포함되어 있지만 제대로 지원하려면 큰 폭으로 확장해야 합니다. ATCA 사양인 PICMG 3.0은 IPMI에 필요한 확장을 정의합니다.

그림 1-1에서는 ATCA 사양에 따라 식별되는 예제 ATCA 선반의 논리적 요소를 나타냅니다.

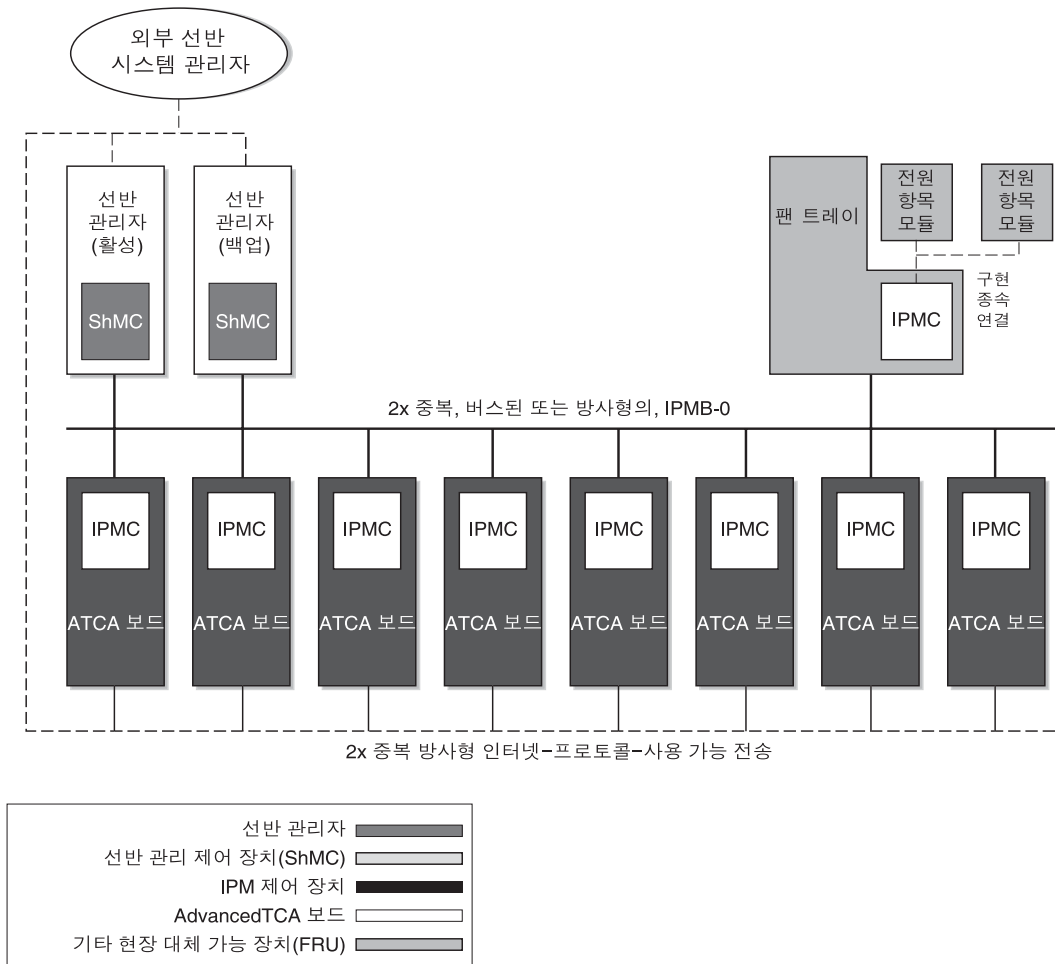


그림 1-2 ATCA 선반 예

전체 시스템 관리자(일반적으로 선반 외부)는 여러 선반의 활동을 조정할 수 있습니다. 일반적으로 시스템 관리자는 이더넷 또는 직렬 인터페이스를 통해 각 선반 관리자와 통신합니다.

그림 1-2에서는 보드, 선반, 시스템 등 세 가지 수준의 관리를 나타냅니다. 다음 절에서는 ATCA 호환 선반 관리자와 선반 관리 제어기(ShMC)를 구현하는 선반 관리자 소프트웨어 및 선반 관리 카드에 대해 설명합니다.

## 선반 관리자 및 선반 관리 카드

ATCA 선반 관리자 요구 사항을 충족하는 선반 관리자는 다음과 같은 두 가지 주 작업을 수행합니다.

- 전원, 냉각, 상호 연결 자원, 사용법 등 선반의 FRU 채우기 및 공통 인프라를 관리하고 추적합니다. 선반에서 이 관리 및 추적 기능은 주로 IPMB-0(Intelligent Platform Management Bus 0)에서 선반 관리자와 IPM 제어기 간의 상호 작용을 통해 수행됩니다.
- 전체 시스템 관리자는 시스템 관리자 인터페이스를 통해 관리와 추적을 결합할 수 있으며 이 작업은 일반적으로 이더넷을 통해 구현됩니다.

대부분의 선반 관리자 소프트웨어는 선반 전원 켜기/끄기, FRU 반입/반출 처리, 전원 할당 협상, 자원 상호 연결 등과 같은 일상 업무를 전담합니다. 또한 선반 관리자는 선반에서 예외가 발생할 경우 직접적인 조치를 취할 수 있습니다. 예를 들어 온도 예외가 발생할 경우 선반 관리자는 팬 수준을 높일 수 있으며 이 조치만으로 부족할 경우 FRU의 전원을 꺼서 선반의 열 부하를 낮출 수 있습니다.

## 선반 관리자 기능

선반 관리자 소프트웨어 기능은 다음과 같습니다.

- 선반에 적합한 캐리어 보드에 설치된 콤팩트 SO-DIMM 폼 팩터 모듈인 선반 관리 카드에서 실행됩니다.
- ATCA 사양을 준수합니다.
- ATCA 지정 이중 중복 IPMB(Intelligent Platform Management Bus)를 통해 선반 활동을 모니터링합니다.
- 선반의 지능형 FRU에서 게시한 이벤트를 수용하고 기록합니다(온도, 전압 등의 예외 반영). 또한 구성 가능한 IPMI 플랫폼 이벤트 필터를 기반으로 선반 외부에 경고를 게시합니다.
- 전체 관리 가시성을 유지하면서 FRU(현장 대체 가능 장치) 핫 스왑을 지원합니다.
- 선반 관리 구현 점점 알람 릴레이를 통해 표준 Telco 알람 인프라에 상호 연결합니다.
- 고가용성을 위한 중복 선반 관리자 인스턴스를 지원합니다.
- 주기적으로 저장하지 않을 경우 선반 관리 카드를 재설정하는 위치독 타이머를 통합합니다. 그러면 백업 선반 관리 카드(구성된 경우)로 자동 스위치오버됩니다.
- 타임스탬프 이벤트를 위한 배터리 기반 실시간 시계가 포함되어 있습니다.
- ATCA 및 CLI에 필요한 RMCP를 포함하여 이더넷을 통해 액세스 가능한 풍부한 선반 외부 인터페이스를 구현합니다.

**그림 1-3**에서는 한 번에 하나의 활성 인스턴스만 작동하는 방식으로 시스템 관리자에서 두 인스턴스에 액세스하는 방법을 나타냅니다. 마찬가지로 활성 인스턴스만 IPMB-0을 통해 선반의 IPM 제어기와 통신합니다. 두 인스턴스는 TCP/IP를 통해 서로 통신하며, 이 때 활성 인스턴스는 백업 인스턴스에 증분 상태 업데이트를 게시합니다. 따라서 필요할 경우 백업 인스턴스가 활성 역할을 신속하게 수행할 수 있습니다.

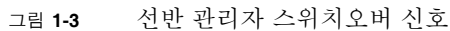


표 1-3에는 신호와 이에 대한 설명이 나열되어 있습니다.

표 1-3 스위치오버를 지원하는 하드웨어 신호 및 인터페이스

하드웨어	설명
USB 인터페이스	선반 관리 카드 사이의 기본 인터페이스이며 하드비트 및 상태 동기화 정보를 보내는 데 사용됩니다. 두 선반 관리 카드는 모두 특정 팬 트레이, 특정 슬롯의 노드 보드 등과 같은 동일한 FRU를 동일한 상태로 표시해야 합니다(예: 전원 켜짐).
#SWITCHOVER	백업 인스턴스가 필요에 따라 스위치오버를 수행할 수 있습니다.
#PRSNT	이 신호는 선반 관리 카드가 있다는 표시입니다.
#HEALTHY	이 신호는 하드웨어 및 소프트웨어를 포함하여 선반 관리 카드의 전체 상태를 나타냅니다.

## 스위치오버 상세 정보

활성 선반 관리자는 IPMB에 ShMC 장치(주소 20h)를 표시하고 IPMB 및 IPM 제어기를 관리하며 RMCP 및 기타 선반 외부 인터페이스를 통해 시스템 관리자와 상호 작용합니다. 또한 백업 선반 관리자와 TCP 연결을 열어 둔 상태로 유지합니다. 활성 선반 관리자는 관리 대상 객체의 모든 상태 변경 사항을 백업 선반 관리자에게 전달합니다.

백업 선반 관리자는 IPMB에 ShMC를 표시하지 않고 IPMB 및 IPM 제어기를 적극적으로 관리하지 않으며 선반 외부 인터페이스를 통해 시스템 관리자와 상호 작용하지 않습니다(아래 명시된 한 가지 경우만 예외). 대신 관리 대상 객체의 상태를 자체 메모리(회발성 및 비회발성)에서 유지 관리하고 활성 선반 관리자의 지시에 따라 상태를 업데이트합니다.

백업 선반 관리자는 스위치오버를 통해 활성 선반 관리자가 될 수 있습니다. 스위치오버의 두 가지 유형은 다음과 같이 정의됩니다.

- 협동 스위치오버 - 활성 선반 관리자와 백업 선반 관리자가 활성 선반 관리자에서 백업 선반 관리자로의 책임 스위치오버를 협상합니다. 이 모드는 활성 또는 백업 선반 관리자에서 실행된 CLI `switchover` 명령을 통해 지원됩니다.
- 강제 스위치오버 - 활성 선반 관리자가 더 이상 활성 또는 정상 상태가 아니라고 판단되면 백업 선반 관리자가 활성 선반 관리자의 책임을 강제로 넘겨 받습니다.

백업 선반 관리자는 Remote Healthy 또는 Remote Presence 하위 수준 신호가 비활성화되면 활성 선반 관리자가 분리된 것으로 인식합니다. Remote Presence 신호는 피어 선반 관리자의 존재를 모니터링합니다. 이 신호가 비활성 상태가 되면 피어 선반 관리자를 호스트하는 보드가 선반에서 제거된 것입니다. Remote Healthy 신호는 초기화 중에 피어 선반 관리자에 의해 설정됩니다. 이 신호가 비활성화되면 원격 선반 관리자가 비정상 상태로 된 것이며 일반적으로 전원이 꺼졌거나 재설정된 상태를 나타냅니다.

백업 선반 관리자의 작업이 필요한 다른 상황으로는 선반 관리자 간의 TCP 연결이 끊어진 경우가 있습니다. 이러한 상황은 두 선반 관리자 간의 통신 링크가 끊어진 경우, 활성 선반 관리자의 선반 관리 프로세스가 자의 또는 타의에 의해 종료된 경우 또는 소프트웨어 예외가 발생한 경우에 발생합니다. TCP keepalive 옵션은 연결 시에 활성화되기 때문에 활성 선반 관리 카드가 꺼지거나 재설정되면 바로 닫힙니다.

선반 관리자가 종료될 경우 Remote Healthy 신호가 비활성화되기 전에 TCP 연결이 끊어질 수 있습니다. TCP 연결이 끊어진 원인을 확인하려면 백업 선반 관리자는 Remote Healthy 신호의 상태를 즉시 샘플링합니다. 아직 활성 상태이면 잠시 후에 다시 샘플링합니다. Remote Healthy 신호가 비활성화되면 백업 선반 관리자는 활성 선반 관리자가 사용 불능 상태인 것으로 간주하고 스위치오버를 시작합니다.

Remote Healthy 신호가 활성 상태로 유지되면 백업 선반 관리자는 선반 관리자 간의 통신 링크가 끊어진 것으로 간주합니다. 그럴 경우 스위치오버가 시작되지 않습니다. 대신 통신 링크가 복원될 때까지 백업 선반 관리자가 반복적으로 다시 초기화되어 활성 선반 관리자와의 연결을 설정하려고 시도합니다. 선반 관리 카드를 재부트한 후 선반 관리자를 자동으로 다시 시작하여 다시 초기화합니다. 선반 관리자의 특수 논리에 따라 피어 선반 관리자가 이미 활성 상태이면 시작할 때 활성 상태로 스위치오버하려고 시도하지 않습니다.

선반 관리자는 위치독 타이머를 사용하여 무한 루프 또는 기타 소프트웨어 버그로 인한 응답 불가 상태를 예방합니다. 활성 선반 관리자에서 위치독 타이머가 트리거되면 해당 선반 관리 카드가 재설정되어 백업 선반 관리 카드의 Remote Healthy 신호가 비활성화되므로 스위치오버가 트리거됩니다.

스위치오버 후 활성 선반 관리자가 다시 초기화되고 캐시된 상태 정보를 활성화한 다음 IPMB의 IPM 제어기에서 필요한 정보를 수집합니다. 그런 다음 이 활성 선반 관리자는 IPMB에 ShMC 장치(주소 20h)를 표시하고, 이전 활성 선반 관리자와 시스템 관리자 간의 RMCP 및 기타 선반 외부 상호 작용에 사용된 IP 주소를 사용합니다. RMCP 세션 정보는 활성 선반 관리자에서 백업 선반 관리자로 전파되기 때문에 RMCP 세션은 스위치오버 후에도 계속 유지됩니다. 따라서 RMCP를 사용하는 시스템 관리자는 스위치오버의 영향을 받지 않습니다.

스위치오버 후 이전 활성 선반 관리자가 중단되거나 백업 선반 관리자로 다시 초기화될 수 있습니다. 백업 선반 관리자로 다시 초기화하려면 이전 활성 선반 관리 카드에서 운영 체제를 다시 부트해야 합니다.

---

## 시스템 관리자 인터페이스 옵션

선반 관리자의 다른 주요 하위 시스템은 시스템 관리자 인터페이스를 구현합니다. 시스템 관리자는 운영 센터의 운영자를 비롯하여 소프트웨어를 포함하는 논리적 개념입니다. 선반 관리자는 선반과 관련한 유사한 유형의 정보와 제어에 대한 서로 다른 액세스 메커니즘을 제공하는 두 개의 시스템 관리자 인터페이스 옵션을 제공합니다.

- IPMI LAN(Local Area Network) 인터페이스
- 명령줄 인터페이스(CLI)

IPMI LAN 인터페이스는 독립적으로 구현되는 선반 제품 간의 상호 운용성을 극대화하는 데 사용됩니다. 이 인터페이스는 ATCA 사양에 필요하며 RMCP를 통해 선반 관리자와의 IPMI 메시지를 지원합니다. RMCP를 사용하여 선반과 통신하는 시스템 관리자는 모든 ATCA 호환 선반 관리자와 상호 작용할 수 있어야 합니다. 이 하위 수준의 인터페이스는 시스템 관리자가 선반 관리자를 프록시로 사용하여 선반의 IPMI 제어기에 IPMI 명령을 실행할 수 있는 기능을 포함하여 선반의 IPMI 기능에 대한 액세스를 제공합니다.

RMCP는 LAN을 통해 IPMI 제어기에 연결하는 표준 네트워크 인터페이스이며 IPMI 1.5 사양에 의해 정의됩니다.

CLI는 물리적 직렬 연결이나 텔넷 연결을 통해 선반 관리자에 실행될 수 있는 포괄적인 텍스트 명령을 제공합니다.

---

## 시스템 관리 작업

Netra CT 900 서버 시스템 관리는 일반적으로 설치, 구성 및 관리 작업을 포함합니다.

Solaris 사용자 계정 추가와 같은 Netra CT 900 서버의 Solaris OS 관리 작업은 노드 보드에 로그인하여 수행합니다. Netra CT 900 서버 관리는 선반 관리 카드에 로그인한 후 선반 관리 카드 CLI를 사용하여 수행됩니다. 구성 및 관리를 위해 선반 관리 카드를 Netra CT 900 서버의 단일 입력점으로 사용할 수 있습니다.

시스템 관리 작업에 대해서는 다음 장에서 설명합니다.

## 물리 주소-논리 슬롯 매핑

전면에서 Netra CT 900 서버를 보면 물리적 슬롯에 왼쪽에서 오른쪽으로 순차적인 번호가 매겨져 있습니다. 표 1-4에서는 물리-논리 슬롯 매핑 및 주소를 나타냅니다.

표 1-4 물리 주소-논리 슬롯 매핑

물리 슬롯	1	2	3	4	5	6	7	8	9	10	11	12	13	14	ShMM #1	ShMM #2
논리 슬롯	13	11	9	7	5	3	1	2	4	6	8	10	12	14	해당 없음	해당 없음
HW 주소 (16진수)	4D	4B	49	47	45	43	41	42	44	46	48	4A	4C	4E	8	9
IPMB 주소 (16진수)	9A	96	92	8E	8A	86	82	84	88	8C	90	94	98	9C	10	12



# 시스템 구성

---

이 장에서는 Netra CT 900 서버 및 Netra ATCA 노드 보드에 Solaris 운영 체제와 필수 패치를 이미 설치한 것으로 가정합니다.

Netra CT 900 시스템은 주로 활성 선반 관리 카드 CLI(명령줄 인터페이스)를 통해 구성합니다. 활성 선반 관리 카드 CLI를 사용하면 노드 보드, 스위칭 패브릭 보드, 선반 관리 카드, PEM(Power Entry Module), 팬 트레이 등을 시스템 수준으로 구성하고 관리할 수 있습니다. 선반 관리 카드 CLI 인터페이스는 로컬과 원격으로 모두 사용할 수 있습니다.

이 장은 다음 절로 구성되어 있습니다.

- 14페이지의 "선반 관리 카드 액세스"
- 15페이지의 "U-Boot 설정"
- 21페이지의 "선반 관리 카드 이더넷 포트 구성"
- 24페이지의 "기본 ShMM 네트워크 매개 변수 변경"
- 28페이지의 "선반 관리자 구성 파일 설정"
- 44페이지의 "날짜 및 시간 설정"
- 46페이지의 "선반 관리 카드에서 사용자 계정 설정"

## 선반 관리 카드 액세스

선반 관리자는 Linux 특수 구현 위에서 실행됩니다. Linux의 최하위 계층은 U-Boot 펌웨어 모니터입니다. 선반 관리 카드(ShMM)에 처음으로 액세스하는 경우 ASCII 터미널 또는 Tip 프로그램을 사용하여 직렬 포트(콘솔)를 통해 액세스해야 합니다. 각 선반 관리 카드는 한 번에 여러 세션(Tip 및 텔넷 연결)을 지원합니다. 활성 선반 관리 카드의 기본 TCP/IP 주소는 192.168.0.2입니다.

직렬 포트를 통해 선반 관리 카드에 연결하는 경우 SAP(선반 알람 패널)의 전면면에 있는 두 직렬 포트 중 하나에 직렬 터미널 또는 에뮬레이터를 연결합니다. 직렬 1 포트를 사용하여 기본 활성 선반 관리 카드인 위쪽 선반 관리 카드(ShMM1)에 연결합니다. 직렬 2 포트는 기본 백업 카드(ShMM2)에 연결합니다. [그림 2-1](#)은 위쪽 및 아래쪽 선반 관리 카드의 직렬 포트 위치를 나타냅니다. 터미널 또는 모뎀 설정을 115200, N, 8, 1로 설정해야 합니다.

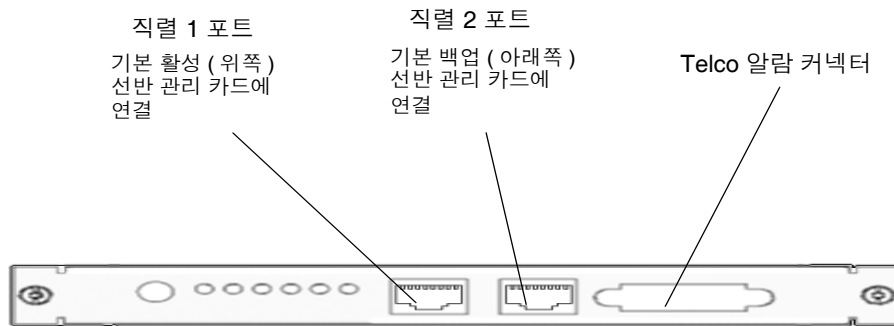


그림 2-1 선반 알람 패널 커넥터

선반 관리 카드에 처음 액세스하는 경우 root로 로그인하고 기본 암호 sunct900을 사용합니다. 이 계정은 전체 인증(권한)으로 설정됩니다. 이 계정은 삭제할 수 없습니다. 보안을 위해 Netra CT 900 서버를 작동하기 전에 이 계정의 암호를 변경해야 합니다.

다음과 같이 Linux passwd 명령을 사용하여 루트 암호를 변경합니다.

```
# passwd

Changing password for root
Enter the new password (minimum of 5, maximum of 8 characters)
Please use a combination of upper and lower case letters and
numbers.
Enter new password: xxxxxxxx
Re-enter new password: xxxxxxxx
Password changed.
#
```

로그온한 후 `clia shmstatus` 명령을 사용하여 활성 선반 관리 카드에 로그인되었는지 확인한 다음 계속합니다. 대기 선반 관리 카드에 로그인한 경우 `clia switchover` 명령을 사용하여 선반 관리 카드를 Active로 변경하거나, 종료한 후 활성 선반 관리 카드에 로그인할 수 있습니다. 자세한 내용은 [237페이지의 "shmstatus"](#) 및 [239페이지의 "switchover"](#)을 참조하십시오.

다음 절에서는 선반 관리 카드 이더넷 포트를 구성하고 선반 관리 카드 CLI를 사용하여 사용자 계정과 암호를 설정하는 방법을 설명합니다. 선반 관리 카드 CLI 사용에 대한 자세한 내용은 [3장](#)을 참조하십시오.

---

주 - 이 설명서에 사용된 선반 관리 카드라는 용어는 별도로 지정하지 않는 한 활성 또는 대기 선반 관리 카드를 나타냅니다. 이 설명서에서 두 카드의 프롬프트는 ShMM #으로 단축 표시됩니다.

---

## U-Boot 설정

선반 관리 카드(ShMM)의 전원을 켜고 다시 부트하면 하드웨어가 플래시 메모리에서 U-Boot 펌웨어를 실행하기 시작합니다. 펌웨어는 ShMM의 기본 초기화 과정을 수행합니다. 사용자가 Autoboot 기능을 명시적으로 비활성화하여 펌웨어가 유지 관리 사용자 명령 인터페이스로 스위치오버되는 경우가 아니면 Linux 커널 부트가 시작됩니다. Linux는 플래시에 있는 커널 및 root 파일 시스템 이미지로부터 부트됩니다. U-Boot는 커널 이미지를 RAM에 재배치하고 커널 매개 변수를 설정한 다음 제어를 커널 시작점으로 전달합니다.

## U-Boot 인터페이스

U-Boot는 ShMM의 직렬 포트를 통해 액세스할 수 있으며 해당 작업 환경에 맞게 구성되어야 합니다. ShMM의 전원을 켜면 다음과 같은 정보가 콘솔에 표시됩니다.

```
U-Boot 1.1.2 (Nov 11 2005 - 11:32:08)

CPU: Au1550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 08004610
DRAM: 128 MB
Flash: 64 MB
In: serial
Out: serial
Err: serial
Net: Au1X00 ETHERNET
Hit any key to stop autoboot: 0
#
```

#은 사용자 명령을 입력할 수 있는 프롬프트입니다.

## U-Boot 환경 변수

U-Boot에는 사용하기 전에 구성해야 하는 일련의 환경 변수가 포함되어 있습니다. [표 2-1](#)에서는 사용 가능한 기본 변수 세트에 대해 설명합니다.

표 2-1 기본 U-Boot 환경 변수

환경 변수	설명
addmisc	quiet, reliable_upgrade 및 console 설정을 bootargs에 추가합니다. 이 변수는 일반적으로 수정되지 않습니다.
baudrate	직렬 포트 변조 속도, 기본값=115200
bootargs	Linux 커널에 전달할 명령줄. 런타임에 확인되는 다른 U-Boot 환경 변수에 대한 참조를 포함할 수 있습니다. 기본값은 다음과 같습니다. root=/dev/ram rw console=ttyS0,115200 reliable_upgrade=y
bootcmd	auto-boot를 수행하기 위해 실행되는 U-Boot 명령
bootdelay	Autoboot 지연 값(초)
bootfile	net 및 nfs 부트 옵션에서 사용할 커널 이미지를 지정하는 매개 변수

표 2-1 기본 U-Boot 환경 변수(계속)

환경 변수	설명
console	커널 및 init 스크립트 콘솔 포트 및 변조 속도에 대한 설정입니다. 기본값은 console=ttyS0,115200입니다.
ethaddr	기본 칩 내장 이더넷 제어기의 MAC 주소입니다. 이 변수 값은 U-Boot에서 자동으로 설정됩니다. 이 주소는 커널 이더넷 드라이버에 전달됩니다.
ethladdr	보조 이더넷 제어기의 MAC 주소입니다. 이 변수 값은 U-Boot에서 자동으로 설정됩니다. 이 주소는 커널 이더넷 드라이버에 전달됩니다.
flash_reset	플래시 파일 시스템(/etc 및 /var)을 지우고 출하 시 기본값(y/n)으로 복원하도록 Linux에 지시합니다. 시스템 시작 스크립트는 플래시가 지워지면 이 변수를 다시 n으로 설정합니다. 기본값은 n입니다.
gateway	기본 게이트웨이 IP 주소입니다. 네트워크 인터페이스에 대한 라우팅을 자동으로 구성하도록 이 변수를 커널 명령줄의 일부로 전달할 수 있습니다.
hostname	네트워크 호스트 이름이며 기본값은 sentry입니다.
io_config	PSC 제어기를 이중 슬레이브 주소 구성(y/n)으로 구성할지 여부를 지정합니다. 기본 설정은 y입니다.
ipaddr	기본 칩 내장 이더넷 인터페이스에 사용되는 IP 주소입니다. 이 변수는 rc_ifconfig 변수가 y로 설정된 경우 ipdevice 변수에 의해 자동으로 지정되는 네트워크 인터페이스를 구성하는 데 사용됩니다. 시스템 시작 스크립트는 이 변수의 최소 유효 비트를 ShMM 캐리어에 대한 하드웨어 주소의 최소 유효 비트로 설정합니다. 즉, 하드웨어 주소가 짝수 값이면 IP 주소의 마지막 비트가 0으로 설정되고, 그렇지 않으면 1로 설정됩니다. 중복 ShMM에서 조정된 IP 주소 구성을 지원하도록 시작 스크립트 /etc/netconfig에서 이 설정이 지정됩니다. 이 기능을 비활성화하려면 간단히 /etc/readhwaddr 파일을 제거합니다.
ipladdr	보조 이더넷 인터페이스에 사용되는 IP 주소입니다. 해당 커널 네트워크 인터페이스를 자동으로 구성하도록 이 변수를 커널 명령줄의 일부로 전달할 수 있습니다.
ipdevice	ipaddr에 해당하는 장치이며 기본값은 eth0입니다.
ipldevice	ipladdr에 해당하는 장치이며 기본값은 eth1입니다.
kernel_start	플래시에 있는 커널 이미지의 절대 시작 주소입니다. 이 변수는 부트스트랩 중에 U-Boot에 의해 자동으로 설정됩니다.
logging	메시지 로그 파일을 ram에 유지할지 flash에 유지할지를 지정합니다. 기본값은 ram이며 권장 옵션입니다.
net	TFTP 서버로부터 커널 및 .frs 이미지를 부트할 때 bootcmd 대신 이 변수를 사용할 수 있습니다.
netmask	네트워크 넷마스크, 기본값=255.255.255.0.
password_reset	출하 시 기본 암호(root 사용자의 경우 빈 암호)를 복원하도록 Linux에 지시합니다. 기본값은 n입니다.

표 2-1 기본 U-Boot 환경 변수(계속)

환경 변수	설명
post_normal	각 부트 시 실행되는 POST 테스트 목록을 결정합니다. 이 변수를 설정하지 않으면 컴파일 시 기본 설정이 사용됩니다. 이 변수 값에 나열되는 테스트 이름은 공백 문자로 구분됩니다.
post_poweron	전원 투입 시 재설정 이후에만 실행되는 POST 테스트 목록을 결정합니다(각 부트와 비교). 이 변수를 설정하지 않으면 컴파일 시 기본 설정이 사용됩니다. 이 변수 값에 나열되는 테스트 이름은 공백 문자로 구분됩니다.
quiet	부트 시 직렬 콘솔에 진행 메시지를 인쇄하지 않도록 커널에 지시합니다. 기본값은 quiet=quiet입니다.
ramargs	ramdisk로부터 마운트할 root 파일 시스템에 적절하게 bootargs 변수의 커널 명령줄을 설정합니다.
ramdisk	net 및 nfs 부트 옵션에서 사용할 .rfs 이미지를 지정합니다.
ramsize	시스템 메모리 크기(바이트)입니다. 기본 설정은 구성 시간 구성 블록의 SDRAM 구성 인코딩으로부터 계산됩니다.
rc_ifconfig	shelfman 파일에서 주소를 가져오는 대신 /etc/rc 스크립트가 IP 주소를 설정하도록 합니다. 기본값은 n(shelfman이 IP 주소를 설정하도록 허용)입니다.
rc2	호출될 보조 RC 스크립트를 지정합니다. 이 스크립트는 캐리어 관련 시작 스크립트입니다. 기본값은 /etc/rc.acb3이거나 지정된 대상 플랫폼에 해당하는 스크립트입니다.
reliable_upgrade	ShMM-500에서 안정적인 소프트웨어 업그레이드 절차를 활성화할지 지정합니다(y/n). 기본 설정은 y입니다. 현재는 이 변수를 n으로 설정할 수 없습니다. 이 변수를 n으로 설정할 경우 다음에 ShMM을 부트하면 오류 메시지가 표시되고 시스템이 중지됩니다.
rfs_start	플래시에 있는 root 파일 시스템 이미지의 절대 시작 주소입니다. 이 변수는 부트스트랩 중에 U-Boot에 의해 자동으로 설정됩니다.
rmcpaddr	RMCP 서비스의 기본 IP 주소입니다.
serverip	TFTP 서버의 IP 주소입니다.
start_rc2_daemons	부트 후에 snmpd/boa 및 shelfman 데몬을 시작하거나 시작하지 않도록 보조 시작 스크립트에 지시합니다. 기본값은 y입니다.
time_server	런타임에 동기화할 시간 서버입니다. 이 변수를 지정하지 않으면 시스템 시작 시 하드웨어 시계로부터 시간이 추출됩니다. 주: 이 변수를 지정한 경우 적절하게 동기화하려면 ip1device 변수를 usb0으로 설정해야 합니다.
timezone	CCCn 형식의 지역 시간대입니다. 여기서 n은 GMT(그리니치 표준시)로부터의 오프셋이며 음수가 사용될 수도 있습니다. CCC는 시간대를 나타냅니다. 기본값은 UTC0입니다.

## 환경 변수에 값 할당

환경 변수에 값을 할당하려면 다음 형식을 사용합니다.

```
setenv variable_name new_value
```

예:

```
# setenv bootdelay 1
```

모든 환경 변수를 적절하게 설정한 경우 ShMM의 전원을 끈 후에도 변수가 설정된 상태로 유지되도록 해당 변수를 플래시에 다시 저장해야 합니다. `saveenv` 명령은 이러한 용도로 사용됩니다.

```
# saveenv
```

동일한 용도의 Linux 유틸리티로 `setenv` 기능을 사용할 수도 있습니다. 셸 프롬프트에 U-Boot 변수를 표시하려면 추가 `getenv` 유틸리티를 사용합니다.

## 선반 관리자의 구성 환경 변수

U-Boot를 처음으로 시작하면 다음과 같이 기본 환경 변수가 정의되어 있습니다.

```
bootcmd=run setup_bootargs; bootm BFB00000 BFC40000
bootdelay=3
baudrate=115200
ethaddr= 00:00:1a:18:xx:yy
ethladdr= 00:00:1a:18:xx:zz
serverip=192.168.0.7
netmask=255.255.0.0
hostname=sentry
gateway=192.168.0.1
ipdevice=eth0
ipladdr=192.168.1.3
ipldevice=eth1
rc2=/etc/rc.acb3
ipaddr=192.168.0.2
start_rc2_daemons=y
flash_reset=n
password_reset=n
logging=ram
rc_ifconfig=n
bootfile=sentry.mips.kernel
ramdisk=sentry.mips.rfsnet=tftp 80400000 $(bootfile); tftp
80800000 $(ramdisk); bootm 80400000 80800000
rmcpaddr=192.168.1.15
timezone=EST
bootargs=root=/dev/ram rw console=ttyS0,115200 reliable_upgrade=y
```

이러한 환경 변수 중 일부는 ShMM에서 사용되는 네트워크 컨텍스트에 적절한 값으로 다시 구성해야 합니다.



## 선반 관리 카드 이더넷 포트 구성

각 선반 관리 카드는 중복 스위치 카드에 연결되는 두 개의 이더넷 포트를 사용합니다. RMCP가 ATCA에 필요한 유일한 선반 외부 인터페이스이기 때문에 선반의 외부 이더넷 포트를 RMCP 포트라고 합니다. 다른 선반의 외부 인터페이스(텔넷)도 이 포트를 통해 액세스할 수 있습니다.

연결한 후 모든 권한이 있는 사용자 계정을 사용하여 선반 관리 카드에 로그인해야 합니다. CLI 명령을 사용하여 포트를 구성한 다음 선반 관리 카드를 다시 부트하여 변경 사항을 적용합니다.

### 첫 번째 이더넷 인터페이스 사용

RMCP 이더넷 포트는 사이트 네트워크에 직접 연결되므로 해당 네트워크에 적절하게 IP 주소를 설정해야 합니다. 예를 들어, 사이트에서 192.168.0.x IP 주소 범위를 사용하는 경우 RMCP 이더넷 포트를 해당 범위 내의 고유한 IP 주소(예: 192.168.0.2)로 설정해야 합니다. 중복 ShMM 설정의 경우 RMCP 이더넷 포트에서 하나의 ShMM (활성 ShMM)에 대해서만 RMCP IP 주소가 활성화됩니다. 백업 ShMM은 RMCP 이더넷 포트에 동일한 IP 주소를 할당하지만, 해당 ShMM이 활성 역할을 하는 경우에만 주소가 활성화됩니다. RMCP IP 주소는 이러한 방법으로 폐일으며 상황에서 가용성을 유지합니다.

### 첫 번째 네트워크 인터페이스에 추가 IP 주소 할당

기본 구성에서는 선반 관리자가 시작되고 RMCP IP 주소가 할당된 다음에 첫 번째 네트워크 인터페이스에 IP 주소가 할당됩니다. 또한 ShMM은 네트워크를 통해 액세스할 수 없습니다. 하지만 RMCP 네트워크 인터페이스에 IP 주소를 할당하면 유용한 경우도 있으며 운영 체제가 부트되면 바로 네트워크를 통해 ShMM에 액세스할 수 있습니다. 이 경우 선반 관리자가 시작될 때 RMCP IP 주소를 대체하지 않고 원래 할당된 IP 주소와 공존하도록 할 수도 있습니다.

이 구성을 사용하려면 첫 번째 네트워크 어댑터 자체(eth0)가 아니라 첫 번째 별칭(eth0:1)에 RMCP IP 주소를 할당하도록 선반 관리자에 지시해야 합니다. 그러면 운영 체제를 시작하는 동안 초기 IP 주소가 네트워크 어댑터 자체(eth0)에 할당됩니다. 이 초기 할당은 초기화 스크립트 /etc/rc에서 다음과 같은 방법으로 수행됩니다.

1. 다음과 같이 U-Boot 변수 rc\_ifconfig를 활성화합니다.

```
setenv rc_ifconfig y
```

2. U-Boot 변수 `ipaddr`에 원래 IP 주소를 할당합니다. 예를 들면 다음과 같습니다.

```
setenv ipaddr 192.168.1.240
```

3. 선반 관리자 구성 파일 `/etc/shelfman.conf`에서 `RMCP_NET_ADAPTER` 값을 `eth0:1`로 변경합니다. 예를 들면 다음과 같습니다.

```
RMCP_NET_ADAPTER = eth0:1
```

중복 구성의 경우 두 ShMM에서 U-Boot 변수 `ipaddr`가 동일한 값을 가질 수 있습니다. 두 중복 ShMM 각각에 실제로 할당된 초기 IP 주소는 `ipaddr`의 값을 기반으로 하며 ShMM의 하드웨어 주소에 따라 수정됩니다. IP 주소의 최소 유효 비트는 하드웨어 주소의 최소 유효 비트로 설정됩니다. 위 예에서 하드웨어 주소가 짝수인 ShMM의 IP 주소는 192.168.1.240이고 하드웨어 주소가 홀수인 ShMM의 IP 주소는 192.168.1.241입니다. `/etc/readhwaddr` 파일을 제거하여 이 IP 주소 수정을 해제할 수 있습니다.

## RMCP 주소 전파

또한 선반 관리자의 이 선택적 기능을 사용하면 IP 주소가 최소 유효 비트 내의 RMCP IP 주소와 다른 외부 네트워크에서 백업 ShMM을 사용할 수 있습니다. 백업 ShMM의 넷마스크와 기본 게이트웨이는 활성 ShMM과 동일합니다. 예를 들어, RMCP IP 주소가 192.168.0.2 일 경우 백업 ShMM의 IP 주소는 192.168.0.3이고 넷마스크와 기본 게이트웨이는 동일합니다. 이 기능을 활성화하려면 선반 관리자 구성 파일(`/etc/shelfman.conf`)에서 선반 관리자 구성 매개 변수 `PROPAGATE_RMCP_ADDRESS`를 `TRUE`로 정의해야 합니다.

## 두 번째 이더넷 인터페이스 사용

두 번째 네트워크 인터페이스는 선반 관리자를 ATCA 네트워크 허브 보드 중 하나에 연결합니다. 이중 USB 기반 네트워크 인터페이스는 중복 선반 관리자 간 통신에 사용됩니다.

## 중복 통신을 위한 이중 USB 네트워크 인터페이스 사용

ShMM에서 두 개의 추가 네트워크 인터페이스는 두 USB 연결을 통해 구현됩니다. 이 구성에서 두 추가 네트워크 인터페이스는 두 개의 중복 선반 관리자를 연결합니다. 각 인터페이스의 이름은 usb0 및 usb1입니다. 인터페이스 usb0은 항상 존재하지만, 인터페이스 usb1은 인터페이스 usb0이 피어 선반 관리자(피어 선반 관리자를 물리적으로 설치하여 실행 중인 경우)에서 활성 상태인 경우에만 존재합니다. 두 인터페이스는 상호 연결됩니다. 첫 번째 선반 관리자의 usb0이 두 번째 선반 관리자의 usb1에 연결되며, 그 반대의 경우도 마찬가지입니다.

선반 관리자는 USB 네트워크 인터페이스를 사용한 중복 선반 관리자 간 통신을 지원합니다. 이 기능을 사용하려면 선반 관리자 구성 파일 /etc/shelfman.conf에서 두 중복 네트워크 어댑터를 다음과 같이 정의해야 합니다.

```
REDUNDANCY_NET_ADAPTER = "usb0"
REDUNDANCY_NET_ADAPTER2 = "usb1"
```

추가 고려 사항은 중복 네트워크 인터페이스의 서브넷 마스크 정의에 따라 다릅니다. 기존에는 중복 네트워크 어댑터가 하나만 사용될 경우 /etc/shelfman.conf에 지정된 중복 IP 주소로부터 두 개의 다른 IP 주소가 파생됩니다. 두 주소는 중복 연결의 두 끝점에 할당되며 최소 유효 비트만 다릅니다.

두 개의 중복 네트워크 어댑터를 사용하는 경우에는 각 끝점(중복 선반 관리자별로 두 개의 끝점)에 하나씩 네 개의 서로 다른 IP 주소가 사용됩니다. 올바르게 작동하려면 동일한 선반 관리자의 두 끝점(usb0 및 usb1)이 서로 다른 논리 네트워크에 속하고, 한 선반 관리자의 usb0과 다른 선반 관리자의 usb1이 동일한 논리 네트워크에 속해야 합니다. 이러한 고려 사항에 따라 /etc/shelfman.conf에 지정된 중복 IP 주소에서 서브넷 마스크의 최소 유효 비트를 토글하여 두 개의 추가 IP 주소를 파생합니다. 따라서 서브넷 마스크는 지정된 중복 IP 주소 클래스의 기본값보다 더 엄격해야 합니다. 서브넷 마스크를 지정하지 않으면 기본적으로 255.255.255.128로 설정됩니다. 이 값은 중복을 위해 USB 네트워크 인터페이스를 사용하는 경우에 /etc/shelfman.conf에서 이 매개 변수에 대해 권장되는 값입니다.

다음은 USB 네트워크 인터페이스에 대한 IP 주소를 파생하는 예입니다.

/etc/shelfman.conf에 다음과 같이 정의되어 있다고 가정합니다.

```
REDUNDANCY_IP_ADDRESS = 192.168.1.2
REDUNDANCY_NETMASK = 255.255.255.128
```

작수 하드웨어 주소를 가진 ShMM에서는 IP 주소가 다음과 같이 할당됩니다.

```
usb0: 192.168.1.2(변경 없음)
usb1: 192.168.1.130(넷마스크의 최소 유효 비트 토글)
```

홀수 하드웨어 주소를 가진 ShMM에서는 IP 주소가 다음과 같이 할당됩니다.

```
usb0: 192.168.1.131(IP 주소의 최소 유효 비트 및 넷마스크의 최소 유효 비트 토글)
usb1: 192.168.1.3(IP 주소의 최소 유효 비트 토글)
```

## 기본 ShMM 네트워크 매개 변수 변경

ShMM을 특정 네트워크 환경에서 작동하도록 구성하려면 다음 네트워크 매개 변수를 변경해야 합니다.

- RMCP IP 주소
- RMCP 게이트웨이 주소
- RMCP 넷마스크

RMCP 네트워크 매개 변수를 변경하는 작업은 여러 단계에 걸친 절차입니다. U-Boot 네트워크 환경 변수를 업데이트한 다음 선반 관리자 CLI를 사용하여 부트된 활성 ShMM 모듈 네트워크 설정을 업데이트해야 합니다.

### ▼ 기본 ShMM 네트워크 매개 변수를 변경하는 방법

#### 1. ShMM 모듈에 직렬 포트 콘솔을 연결합니다.

이 변조는 일반적으로 115200보드, N/8/1입니다. ShMM 캐리어를 다시 부트하고 스페이스바를 눌러 자동 부트 프로세스를 중단합니다. 다음과 같이 표시됩니다.

```
U-Boot 1.1.2 (Nov 11 2005 - 11:32:08)

CPU: Aul550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 00 00 00 00 00 00 00 00 00 03 03 03
DRAM: 128 MB
Flash: 64 MB
In: serial
Out: serial
Err: serial
Net: AulX00 ETHERNET
Hit any key to stop autoboot: 0
ShMM #
```

#### 2. 현재 네트워크 설정을 반향합니다.

```
ShMM # printenv rmcpaddr netmask gateway
rmcpaddr=192.168.0.44
netmask=255.255.255.0
gateway=192.168.0.1
ShMM #
```

3. 설정을 변경하고 비휘발성 저장소에 완결합니다.

```
ShMM # setenv rmcpaddr 10.1.1.10
ShMM # setenv netmask 255.255.0.0
ShMM # setenv gateway 10.1.1.1
ShMM # saveenv
Un-Protected 1 sectors
Erasing sector 0 ... Erasing sector at 0x 800000
ok.
Saving Environment to EEPROM...done.
ShMM #
```

4. ShMM을 전체 작업 상태로 부트하고 root 사용자로 로그인합니다.

```
ShMM # reset

U-Boot 1.1.2 (Nov 11 2005 - 11:32:08)

CPU: Au1550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 00 00 00 00 00 00 00 00 03 03 03
DRAM: 128 MB
Flash: 64 MB
In: serial
Out: serial
Err: serial
Net: Au1X00 ETHERNET
Hit any key to stop autoboot: 0
## Booting image at bfb00000 ...
   Image Name: MIPS Linux-2.4.26
   Created: 2005-05-07 17:35:21 UTC
   Image Type: MIPS Linux Kernel Image (gzip compressed)
   Data Size: 843144 Bytes = 823.4 kB
   Load Address: 80100000
   Entry Point: 802bc040
   Verifying Checksum ... OK
   Uncompressing Kernel Image ... OK
## Loading Ramdisk Image at bfc40000 ...
   Image Name: sentry RFS Ramdisk Image
...
...
sentry login: root

BusyBox v0.60.5 (2005.05.07-17:27+0000) Built-in shell (msh)
#
```

## 5. ShMM을 시작합니다.

---

주 – U-Boot 펌웨어에서 변경된 설정이 Linux 환경에 반드시 전파되는 것은 아닙니다. 그 이유는 페일오버 상황을 관리하려면 선반 관리자가 자체 네트워크 구성 데이터 복사본을 유지해야 하기 때문입니다.

---

선반 관리자를 처음으로 부트했거나 부트하기 전에 플래시 장치를 출하 시 기본값으로 재설정 한 경우, 선반 관리자는 U-Boot에서 제공하는 네트워크 설정을 사용하여 이 네트워크 컨텍스트를 설정하므로, U-Boot에서 변경한 내용이 앞으로 전파됩니다.

그렇지 않으면 다음 단계를 수행하여 선반 관리자 컨텍스트에서 네트워크 설정을 구성해야 합니다.

## 6. 자신이 활성 선반 관리자인지 확인합니다.

중복 인터페이스를 통해 네트워크 구성 변경으로 백업을 업데이트하는 cpld 명령을 사용하여 활성 선반 관리 카드를 변경해야 합니다. 활성 ShMM이 아닌 경우 다른 ShMM 장치에 연결하여 [7단계](#)를 계속합니다.

```
# cpld
CPLD word: E806
    0002h - Local Healthy
    0004h - Switchover Request Local
    0800h - Hot Swap Latch Open
    2000h - Active
    4000h - Interrupt Status
    8000h - Reboot Was Caused By Watchdog
#
```

## 7. 현재 IP 설정 가져오기

```
# clia getlanconfig 1

IPM Sentry Shelf Manager Command Line Interpreter

Authentication Type Support: 0x15 ( None MD5 Straight Password/Key )
Authentication Type Enables:
    Callback level: 0x00
    User level: 0x15 ( "None" "MD5" "Straight Password/Key" )
    Operator level: 0x15 ( "None" "MD5" "Straight Password/Key" )
    Administrator level: 0x15 ( "None" "MD5" "Straight Password/Key" )
    OEM level: 0x00
IP Address: 206.25.139.28
IP Address Source: Static Address (Manually Configured) (0x01)
MAC Address: 00:50:c2:22:50:30
Subnet Mask: 0.0.0.0
IPv4 Header Parameters: 0x40:0x40:0x10
Primary RMCP Port Number: 0x026f
Secondary RMCP Port Number: 0x0298
BMC-generated ARP Control: 0x02
    Enable BMC-generated ARP Response
Gratuitous ARP Interval: 2.0 seconds
Default Gateway Address: 206.25.139.3
Default Gateway MAC Address: 00:00:00:00:00:00
Backup Gateway Address: 0.0.0.0
Backup Gateway MAC Address: N/A
Community String: "public"
Number of Destinations: 16
Destination Type:
    N/A
Destination Address:
    N/A
#
```

8. 다음과 같이 IP 설정을 변경합니다.

```
# clia setlanconfig 1 ip 10.1.1.10

IPM Sentry Shelf Manager Command Line Interpreter
IP set successfully

# clia setlanconfig 1 subnet_mask 255.255.0.0

IPM Sentry Shelf Manager Command Line Interpreter
Subnet Mask set successfully

# clia setlanconfig 1 dft_gw_ip 10.1.1.1

IPM Sentry Shelf Manager Command Line Interpreter
Default Gateway Address set successfully
#
```

## 선반 관리자 구성 파일 설정

선반 관리자 구성 파일(shelfman.conf)은 /etc 디렉토리에 있습니다. 파일의 각 행은 구성 매개 변수에 대한 할당을 나타내는 주석 행(#으로 시작) 또는 이름 = 값 쌍입니다. 이름과 값은 등호(=)로 구분됩니다.

구성 매개 변수 이름은 대소문자를 구분하지 않습니다. 각 구성 매개 변수의 유형은 부울, 숫자, 문자열, IP 주소 중 하나입니다.

값 형식은 다음과 같이 구성 매개 변수 유형과 일치합니다.

부울	부울은 문자열 FALSE 또는 TRUE로 표시하거나 숫자 0 또는 1로 표시할 수 있습니다.
숫자	전체(부호 사용 가능) 숫자 값이며, 16진수 표기법 "0x..."도 지원됩니다.
문자열	따옴표를 사용하거나 사용하지 않은 문자열입니다(큰 따옴표 "" 사용). 따옴표를 사용한 문자열에는 공백이 포함될 수 있으며, 따옴표를 사용하지 않은 문자열은 첫 번째 공백 위치에서 종료됩니다. 최대 문자열 크기는 문자열 기반 구성 매개 변수마다 별도로 지정됩니다.
IP 주소	10진수-점(xxx.xxx.xxx.xxx) 표기법으로 된 IP 주소입니다.

\$envvar 표기법을 사용하여 환경 변수 값을 구성 매개 변수 값으로 지정할 수 있습니다. 그럴 경우 구성 파일을 읽을 때 envvar 변수 값이 대체됩니다. 예를 들면 다음과 같습니다.

```
DEFAULT_RMCP_IP_ADDRESS = $IPADDR
```



선반 관리자를 처음으로 실행한 이후에는 IP 주소가 IPMI LAN 구성 매개 변수로 저장됩니다. LAN 구성 매개 변수는 RMCP 또는 CLI 선반 외부 인터페이스를 통해 액세스하거나 수정될 수 있으며 선반 관리자를 다시 시작할 때 shelfman 구성 파일보다 우선 순위가 높습니다. 따라서 이러한 인터페이스를 통해 LAN IP 주소와 게이트웨이를 수정한 내용이 지속됩니다. 선반 FRU 정보의 선반 관리자 IP 연결 레코드에 IP 주소가 포함되어 있으면, 해당 IP 주소가 모든 다른 선반 외부 또는 RMCP IP 주소 설정보다 우선합니다. 선반 관리자 구성 파일 및 IPMI LAN 구성 매개 변수를 통해 주소를 제어할 수 있도록 선반 FRU 정보에서 이 주소를 지정하지 않거나 주소를 0.0.0.0으로 설정하는 것이 좋습니다.

현재 지원되는 구성 매개 변수는 다음과 같습니다.

표 2-2 선반 관리자 구성 매개 변수

이름	유형	기본값	설명
2_X_SYSTEM	부울	없음	이 매개 변수를 지정하면 현재 시스템을 AdvancedTCA로 명시적으로 지정합니다 (FALSE인 경우). 이 매개 변수를 지정하지 않으면(TRUE) 시스템 유형이 자동으로 선택됩니다. 시스템 유형에 대한 잘못된 하드웨어 감지 알고리즘을 대체해야 하는 경우가 아니면 이 매개 변수를 지정하지 않는 것이 좋습니다.
ALARM_CUTOFF_TIMEOUT	숫자	600초(5분)	초 단위 알람 중지 시간 초과(경과하면 알람 중지가 비활성화되는 시간)
ALLOW_CLEARING_CRITICAL_ALARM	부울	FALSE	TRUE로 설정한 경우 CLI 명령 <code>clia alarm clear</code> 를 실행하여 위험 알람 조건을 지울 수 있습니다.
ALTERNATE_CONTROLLER	부울	TRUE	주소 = ShMM 하드웨어 주소를 사용하여 선반 관리자에서 대체 제어기를 사용합니다.
AUTO_SEND_MESSAGE	부울	TRUE	선반 관리자가 아닌 IPMB 주소로 보낸 RMCP 요청을 해당 주소에 직접 전달되는 Send Message 요청으로 자동으로 변환합니다.
CARRIER	문자열(16)	PPS	ShMM이 설치되는 특정 캐리어 보드의 이름입니다.
CARRIER_OPTIONS	문자열(256)	" "	캐리어 관련 옵션이며 지원되는 캐리어마다 별도로 정의됩니다.
CONSOLE_LOGGING_ENABLED	부울	FALSE	선반 관리자가 시작된 콘솔에 로그 메시지를 출력합니다.
COOLING_IGNORE_LOCAL_CONTROL	부울	FALSE	팬 장치에서 로컬 제어 기능을 사용하지 않습니다. 선반 관리자가 팬 수준을 명시적으로 관리합니다.

표 2-2 선반 관리자 구성 매개 변수(계속)

이름	유형	기본값	설명
COOLING_POLL_TIMEOUT	숫자	30초	냉각 모니터링 및 관리 스레드에 대한 연속 호출 간 최대 시간(초)입니다.
CTCA_FRU_RESET_TIMEOUT	숫자	해당 없음	해당 없음
CTCA_HEALTHY_TIMEOUT	숫자	해당 없음	해당 없음
CTCA_INITIAL_FAN_LEVEL	숫자	해당 없음	해당 없음
DEFAULT_GATEWAY_IP_ADDRESS	IP 주소	없음	채널 1의 IPMI LAN 구성 매개 변수에 해당 매개 변수가 0.0.0.0으로 설정되어 있는 경우에 선반 외부(RMCP 기반) 통신을 위해 게이트웨이에 사용되는 기본 IP 주소입니다. LAN 구성 매개 변수에 0이 아닌 게이트웨이 IP 주소가 제공되는 경우에는 선반 관리자 구성 파일에 제공된 값은 무시됩니다.
DEFAULT_RMCP_IP_ADDRESS	IP 주소	없음	선반 외부(RMCP 기반) 통신에 사용되는 기본 IP 주소이며 선반 관리자의 중복 인스턴스 간에 스위치오버됩니다. 이 IP 주소는 채널 1의 IPMI LAN 구성 매개 변수 및 선반 FRU 정보의 선반 관리자 IP 연결 레코드에 해당 매개 변수가 0.0.0.0으로 설정되어 있는 경우에만 사용됩니다. LAN 구성 매개 변수 및/또는 선반 FRU 정보에 0이 아닌 IP 주소가 제공된 경우 선반 관리자 구성 파일에 제공된 값은 무시됩니다.
DEVICE_POLL_TIMEOUT	숫자	10초	선반 관리자가 Get Device ID 명령으로 실행하는 IPMB 장치의 연속 폴링 간 간격(초)입니다.
EXIT_IF_NO_SHELF_FRU	부울	FALSE	TRUE이면 선반 FRU가 없을 경우 선반 관리자가 종료됩니다(ShMM 재설정 가능).
INITIAL_FAN_LEVEL	숫자	5	선반 관리자가 팬 트레이에 적용하는 초기 팬 수준입니다. 일반적으로 팬 수준 값은 0-15 범위 내에 있습니다. 0이 가장 느리고 15가 가장 빠른 팬 속도입니다.
IPMB_ADDRESS	숫자	0	선반 관리자의 IPMB 주소이며 하드웨어 주소를 대체합니다. 값이 0이면 선반 관리자가 하드웨어에서 하드웨어 주소를 읽은 다음 IPMB 주소를 하드웨어 주소 * 2로 설정합니다.
IPMB_RETRIES	숫자	3	요청에 대한 응답이 없는 경우 마지막으로 중단하기 전에 IPMB 요청을 다시 보내려고 시도하는 횟수입니다.

표 2-2 선반 관리자 구성 매개 변수(계속)

이름	유형	기본값	설명
IPMB_RETRY_TIMEOUT	숫자	4초	IPMB 요청을 보낸 이후 요청을 재시도하기 전에 선반 관리자가 응답을 기다리는 시간입니다.
LOCAL_SHELF_FRU	부울	TRUE	/var/nvdata/shelf_fru_info 파일에서 가져온 선반 FRU 정보를 사용하는 선반 관리자에서 로컬 FRU 1을 만듭니다.
M7_TIMEOUT	숫자	-1(초)	FRU가 M7 상태로 유지되는 최대 시간(초)입니다. 이 시간이 만료되면 FRU가 자동으로 M0으로 스위치오버됩니다. -1(기본값)은 계속을 의미합니다. 이 매개 변수를 0으로 설정하면 FRU가 M7 상태로 스위치오버되지 않습니다.
MAX_ALERT_POLICIES	숫자	64	사용 가능한 최대 PEF 경고 정책 수입입니다.
MAX_ALERT_STRINGS	숫자	64	사용 가능한 최대 PEF 경고 문자열 수입입니다.
MAX_DEFERRED_ALERTS	숫자	32	미해결 PEF 경고 최대 수입입니다.
MAX_EVENT_FILTERS	숫자	64	사용 가능한 최대 PEF 이벤트 필터 수입입니다.
MAX_EVENT_SUBSCRIBERS	숫자	64	선반 관리자로부터 이벤트 알림을 수신하기 위해 동시에 가입할 수 있는 최대 엔티티 수입입니다.
MAX_EVENT_SUBSCRIBER_IDLE_TIME	숫자	60초	이벤트가 도착한 시간부터 가입자가 선반 관리자에서 이 이벤트를 검색하는 시간까지 이벤트 가입자에게 허용되는 최대 시간 초과(초)입니다. 이 시간 초과가 초과되면 해당 가입자는 사용 불능 상태로 간주되어 자동으로 등록 취소됩니다.
MAX_PENDING_EVENT_NOTIFICATIONS	숫자	1024	활성 가입자별 최대 미해결 이벤트 알림 수입입니다.
MAX_PENDING_IPMB_REQUESTS	숫자	64	응답을 대기하는 보류 중 IPMB 요청의 최대 수입입니다.
MAX_SEL_ENTRIES	숫자	1024	시스템 이벤트 로그(SEL)의 최대 항목 수입입니다.
MAX_SESSIONS	숫자	32	최대 동시 IPMI 세션 수입입니다.
MAX_USERS	숫자	32	최대 IPMI 사용자 수입입니다.
MIN_FAN_LEVEL	숫자	1	최소 팬 수준입니다. 팬 수준을 자동으로 제어할 경우 냉각 관리 코드는 팬 수준을 이 값 이하로 줄일 수 없습니다.
MIN_SHELF_FRUS	숫자	2	선반 관리자가 성공적으로 시작하기 위해 선반에서 감지해야 하는 최대 선반 FRU 수입입니다.

표 2-2 선반 관리자 구성 매개 변수(계속)

이름	유형	기본값	설명
PHYSICAL_SENSORS	부울	TRUE	ADM1026 및 LM75 칩에서 호스트하는 물리적 센서를 기반으로 IPMI 센서를 만듭니다.
POWER_UNLISTED_FRUS	부울	TRUE	선반 FRU 정보의 전원 관리 테이블에 나열되지 않은 FRU를 활성화하고 전원을 켤 수 있습니다.
PROPAGATE_RMCP_ADDRESS	부울	FALSE	TRUE이면 활성 선반 관리자는 RMCP IP 주소를 백업 선반 관리자에 전파합니다. 백업 선반 관리자는 해당 IP 주소를 사용하여 RMCP_NET_ADAPTER 변수에 지정된 네트워크 인터페이스를 구성합니다. 이때 최소 유효 비트는 반전됩니다.
REDUNDANCY_ENABLED	부울	TRUE	선반 관리자를 중복 모드에서 실행합니다.
REDUNDANCY_NET_ADAPTER	문자열(16)	usb0	선반 관리자의 중복 인스턴스 간 통신에 사용되는 네트워크 어댑터의 이름입니다.
REDUNDANCY_NET_ADAPTER2	문자열(16)	usb1	선반 관리자의 중복 인스턴스 간 통신에 사용되는 두 번째 네트워크 어댑터의 이름입니다(이중 USB 네트워크 인터페이스가 이 용도로 사용되는 경우).
REDUNDANCY_NETMASK	숫자	0	중복 IP 주소에 할당할 넷마스크입니다. 기본적으로(0인 경우) 넷마스크는 IP 주소의 클래스로부터 자동으로 결정됩니다.
REDUNDANCY_PORT	숫자	1040	선반 관리자의 중복 인스턴스 간 상호 작용에 사용되는 TCP 포트입니다.
REDUNDANT_IP_ADDRESS	IP 주소	없음	중복 통신에 사용되는 IP 주소입니다. 이 주소는 실제로 최소 유효 비트만큼만 차이나는 IP 주소 쌍을 지정합니다. 이 주소 쌍은 하드웨어 주소에 따라 중복 선반 관리자에 할당됩니다.
RESERVATION_RETRIES	숫자	10	선반 관리자가 Reserve Device SDR 명령을 재시도하는 최대 횟수입니다.
RMCP_NET_ADAPTER	문자열(16)	eth0	RMCP 기반 통신에 사용되는 네트워크 어댑터의 이름입니다.
RMCP_NET_ADAPTER2	문자열(16)	없음	하드웨어에서 상호 연결 링크를 지원하는 경우 RMCP 기반 통신에 사용되는 대체 네트워크 어댑터의 이름입니다.
SDR_READ_RETRIES	숫자	3	선반 관리자가 Read Device SDR 명령을 재시도하는 최대 횟수입니다.

표 2-2 선반 관리자 구성 매개 변수(계속)

이름	유형	기본값	설명
SEL_HIGH_WATERMARK	숫자	0	SEL 자동 제거를 제어하는 알고리즘에 대한 최대 워터마크입니다. SEL에서 사용 가능한 항목의 실제 비율이 이 값 아래로 떨어지거나 SEL 오버플로가 발생하면 선반 관리자는 SEL에서 이전 레코드를 제거하는 스레드를 시작하여 수명을 줄입니다.
SEL_LOW_WATERMARK	숫자	0	SEL 자동 제거를 제어하는 알고리즘에 대한 최소 워터마크입니다. SEL에서 이전 레코드를 제거하는 스레드가 시작되면 SEL의 항목 비율이 이 값 아래로 떨어질 때까지 레코드를 제거합니다.
SHELF_FRU_IN_EEPROM	부울	TRUE	TRUE이면 백플레인의 EEPROM에서 캐리어 특정 방식으로 선반 FRU 정보를 검색하고, FALSE이면 플래시 파일 시스템의 파일에서 선반 FRU 정보를 가져옵니다.
SHELF_FRU_TIMEOUT	숫자	5초	초기화 중에 선반 관리자가 선반 FRU 정보 장치가 검색될 때까지 대기하는 시간 간격입니다.
SHORT_SEND_MSG_RESPONSE	부울	TRUE	선반 관리자가 제공하는 메시지 전송 응답 유형을 결정합니다. TRUE이면 PICMG 3.0 ECR이 필요하고, FALSE이면 이전 버전의 선반 관리자와 호환됩니다.
SWITCHOVER_TIMEOUT_ON_BROKEN_LINK	숫자	10초	이 매개 변수는 선반 관리자와 시스템 관리자 간의 물리적 네트워크 링크(RMCP 링크)가 끊어진 경우에 선반 관리자가 스위치오버를 시작하는 시간과 스위치오버를 시작할지 여부에 영향을 미칩니다. 이 매개 변수에 지정된 시간(초) 이상 링크를 끊어진 상태로 두면 스위치오버가 수행되고, 이 시간 초과 기간 중에 링크가 복원되면 스위치오버가 수행되지 않습니다. 이 매개 변수의 값이 -1이면 끊어진 RMCP 링크에 대해 자동 스위치오버가 수행되지 않습니다.
SYSLOG_LOGGING_ENABLED	부울	TRUE	시스템 로그에 대한 로그 메시지를 출력합니다.
TASKLET_RETRIES	숫자	3	각 선반 관리자 작업(활성화, 비활성화, 정보 가져오기 등)을 최종적으로 중단하기 전에 재시도하는 횟수입니다.

표 2-2 선반 관리자 구성 매개 변수(계속)

이름	유형	기본값	설명
VERBOSITY	숫자	7	선반 관리자 상세 표시 수준입니다.
VERIFY_SHELF_FRU_CHECKSUM	부울	TRUE	선반 FRU 정보 레코드에서 체크섬 확인을 활성화합니다. FALSE로 설정하면 선반 관리자가 체크섬을 무시합니다.
WATCHDOG_ENABLED	부울	TRUE	CPLD에서 지원하는 하드웨어 워치독 타이머를 사용합니다.

기본적으로 ShMM이 처음으로 실행되면 구성 파일 변수가 자동으로 사용됩니다. 기본 구성 파일은 U-Boot가 설정한 다음과 같은 몇 가지 환경 변수를 가져옵니다.

\$CARRIER_OPTIONS	기본 캐리어 특정 옵션
\$IPADDR	기본 RMCP IP 주소
\$IPDEVICE	기본 RMCP 네트워크 어댑터
\$IPIADDR	기본 중복 IP 주소
\$IPIDEVICE	기본 중복 네트워크 어댑터
\$GATEWAY	RMCP 통신에 사용되는 기본 게이트웨이

필요한 경우 선반 관리자를 출하 시 기본 매개 변수 값으로 재설정할 수 있습니다. 코드 예 2-1에서는 기본 구성 파일의 복사본을 나타냅니다.

코드 예 2-1 기본 shelfman.conf 파일

```
# /etc/shelfman.conf
#
# This is the PPS Shelf Manager configuration file.
# Copyright (c) 2005 Pigeon Point Systems.
# All rights reserved.
#

# CARRIER: This parameter is the name of the carrier-specific module to use.
# Default is PPS.
CARRIER = $CARRIER

# CARRIER_OPTIONS: This parameter specifies the carrier-specific options.
# Default is an empty string.
CARRIER_OPTIONS = $CARRIER_OPTIONS

# ALTERNATE_CONTROLLER: This parameter of boolean type specifies whether to
# use the alternate controller on the Shelf Manager with the address
# equal to the ShM hardware address. Default is TRUE.
#
```

```
ALTERNATE_CONTROLLER = TRUE

# ALLOW_CLEARING_CRITICAL_ALARM: This parameter of boolean type enables the
# ability to clear the critical alarm condition without the alarm cutoff
# button. Default is FALSE.
#
ALLOW_CLEARING_CRITICAL_ALARM = FALSE

# ALARM_CUTOFF_TIMEOUT: This parameter specifies the time interval in
# seconds for the Shelf Manager to hold the Alarm Cutoff state. Default
# interval is 600 seconds.
#
ALARM_CUTOFF_TIMEOUT = 600

# COOLING_IGNORE_LOCAL_CONTROL: This parameter of boolean type specifies
# whether the Shelf Manager should use local control capabilities on fan
# devices i.e. whether the Shelf Manager should explicitly manage fan
# levels or not. Default is FALSE.
#
COOLING_IGNORE_LOCAL_CONTROL = FALSE

# COOLING_POLL_TIMEOUT: This parameter specifies the maximum time (in
# interval is between subsequent invocations of the cooling monitoring and
# management facility. Default is 30 seconds.
#
COOLING_POLL_TIMEOUT = 30

# DEVICE_POLL_TIMEOUT: This parameter specifies the time (in seconds)
# between subsequent polls of the IPMB-0 devices by the Shelf Manager via
# sending the "Get Device ID" command to them. Default is 10 seconds.
#
DEVICE_POLL_TIMEOUT = 10

# IPMB_ADDRESS: This parameter defines the IPMB address of the Shelf
# Manager's slot. This parameter overrides the hardware address. The default
# value of 0 forces the Shelf Manager to use the hardware address and set its
# IPMB address to hardware address * 2.
#
IPMB_ADDRESS = 0

# IPMB_RETRIES: This parameter is the number of attempts to re-send an IPMB
# request before finally giving up, if no response is received to this
# request. Default is 3.
#
IPMB_RETRIES = 3
```

```
# IPMB_RETRY_TIMEOUT: This parameter is the amount of time (in seconds) the
# Shelf Manager waits for a response after sending an IPMB request, before
# retrying it. Default is 4 seconds.
#
IPMB_RETRY_TIMEOUT = 4

# M7_TIMEOUT: This parameter specifies the maximum time interval (in
# seconds for a FRU to stay in M7 state. After the expiration of this time
# the FRU automatically transitions into the M0 state. Default is -1 which
# means "forever". Setting this parameter to 0 completely prevents FRUs from
# going into the M7 state.
#
M7_TIMEOUT = -1

# MAX_ALERT_POLICIES: This parameter specifies the number of available
entries
# in the PEF Alert Policy table. Default is 64.
#
MAX_ALERT_POLICIES = 64

# MAX_ALERT_STRINGS: This parameter specifies the number of available
entries
# in the PEF Alert String table. Default is 64.
#
MAX_ALERT_STRINGS = 64

# MAX_DEFERRED_ALERTS: This parameter sets the maximum number of outstanding
# PEF alerts. Default is 32.
#
MAX_DEFERRED_ALERTS = 32

# MAX_EVENT_FILTERS: This parameter specifies the number of available
entries
# in the PEF Event Filter table.
#
MAX_EVENT_FILTERS = 64

# MAX_OEM_FILTERS: This parameter specifies the number of available entries
# in the PEF OEM Event Filter table. Default is 16.
#
MAX_OEM_FILTERS = 16

# MAX_PENDING_IPMB_REQUESTS: The parameter sets the maximum number of
# pending IPMB requests awaiting response. Default is 192.
```



```
#
MAX_PENDING_IPMB_REQUESTS = 192

# MAX_SEL_ENTRIES: The parameter defines the SEL capacity in records.
# Default is 1024.
#
MAX_SEL_ENTRIES = 1024

# SEL_HIGH_WATERMARK: This parameter is the "high watermark" for the
algorithm
# algorithm that controls automatic SEL purging. The purging process will
# start when the actual percentage of free entries in SEL falls below this
# value or the SEL is full. During the purge the oldest SEL records are
# removed according their timestamp. Default is 10 percent i.e. start
# purging when SEL is full.
#
SEL_HIGH_WATERMARK = 10

# SEL_LOW_WATERMARK: This parameter is the "low watermark" for the algorithm
# that controls automatic SEL purging. When the SEL purging thread starts
# it removes records one by one until the percentage of remaining occupied
# entries in the SEL falls below this value. Default is 50 percent.
#
SEL_LOW_WATERMARK = 50

# MAX_SESSIONS: This parameter specifies the maximum number of simultaneous
# IPMI sessions. Default 32.
#
MAX_SESSIONS = 32

# MAX_USERS: This parameter specifies the maximum number of IPMI users.
# Default is 32.
#
MAX_USERS = 32

# INITIAL_FAN_LEVEL: This parameter specifies the initial fan level that the
# Shelf Manager applies to fan trays. Usually fan levels values are in
# 0..15 range where 0 is the slowest, and 15 is the fastest possible fan
# speed. This parameter has an alias CTCA_INITIAL_FAN_LEVEL for CompactPCI
# systems. Default is 5.
#
INITIAL_FAN_LEVEL = 5

# MIN_FAN_LEVEL: This parameter specifies the minimal fan level that can be
# set by the Cooling Management. Default is 0.
```

```
#
MIN_FAN_LEVEL = 1

# PHYSICAL_SENSORS: This parameter of boolean type specifies whether the
# Shelf Manager should create IPMI sensors based on physical sensors hosted
# by ADM1026 and LM75. Default is TRUE.
#
PHYSICAL_SENSORS = TRUE

# POWER_UNLISTED_FRUS: This parameter of boolean type specifies whether the
# Shelf Manager should power up and activate FRU devices that are not listed
# in the Power Management table of the Shelf FRU Information. Default is
# TRUE.
#
POWER_UNLISTED_FRUS = TRUE

# AUTO_SEND_MESSAGE: This parameter of boolean type specifies whether to
# auto-convert RMCP requests targeting a non-ShM IPMB address into "Send
# Message" requests directed to that address. Default is TRUE.
#
AUTO_SEND_MESSAGE = TRUE

# SHORT_SEND_MSG_RESPONSE: This parameter of boolean type determines the
# type of response on the Send Message command provided by the Shelf
# Manager:required by the PICMG 3.0 R1.0 ECN-001 if TRUE or compatible with
# previous versions of the Shelf Manager if FALSE. Default is TRUE.
#
SHORT_SEND_MSG_RESPONSE = TRUE

# SDR_READ_RETRIES: This parameter sets the number of times the Shelf
# Manager retries the "Read Device SDR" command. Default is 3.
#
SDR_READ_RETRIES = 3

# RESERVATION_RETRIES: This parameter specifies the number of times the
# Shelf Manager retries the "Reserve Device SDR" command. Default is 10.
#
RESERVATION_RETRIES = 10

# TASKLET_RETRIES: This parameter specifies the number of times each Shelf
# Manager tasklet (activation, deactivation, getting information) is
# retried before finally giving up. The default is 3.
#
TASKLET_RETRIES = 3
```

```
# SHELF_FRU_IN_EEPROM: This parameter of boolean type tells the Shelf
# Manager if it should use EEPROMs as the Shelf FRU Info storage. If set
# to FALSE the "/var/nvdata/shelf_fru_info" file contents are used. Default
# is TRUE.
#
SHELF_FRU_IN_EEPROM = TRUE

# LOCAL_SHELF_FRU: This parameter of boolean type specifies whether the
# Shelf Manager should create a local FRU#1 that will expose the Shelf FRU
# Info (obtained from the "/var/nvdata/shelf_fru_info" file). If the Shelf
# FRU Info is acquired from EEPROM as a result of the SHELF_FRU_IN_EEPROM
# set to TRUE then this parameter ignored. Default is TRUE.
#
LOCAL_SHELF_FRU = TRUE

# SHELF_FRU_TIMEOUT: This parameter specifies the time interval (in seconds)
# during which the Shelf Manager detects and reads the Shelf FRU Information
# source devices at initial startup. Default is 15 seconds.
#
SHELF_FRU_TIMEOUT = 15

# MIN_SHELF_FRUS: This parameter specifies the minimum number of valid and
# equal Shelf FRU Information instances that must be found to determine the
# true Shelf FRU Information. Default is 2.
#
MIN_SHELF_FRUS = 2

# EXIT_IF_NO_SHELF_FRU: This parameter of boolean type tells the Shelf
# Manager if it should exit if no valid Shelf FRU Information data is found.
# Default is FALSE.
#
EXIT_IF_NO_SHELF_FRU = FALSE

# VERIFY_SHELF_FRU_CHECKSUM: This parameter boolean type specifies whether
# the Shelf FRU Information record checksums should be validated. The
# default is TRUE.
#
VERIFY_SHELF_FRU_CHECKSUM = TRUE

# WATCHDOG_ENABLED: This parameter of boolean type tells the Shelf Manager
# whether it should use the hardware watchdog timer supported by the CPLD or
# not. The default is TRUE.
#
WATCHDOG_ENABLED = TRUE
```

```
# REDUNDANCY_ENABLED: This parameter of boolean type tells Shelf Manager if
# it should run in redundant mode or not. Default is TRUE.
#
REDUNDANCY_ENABLED = TRUE

# REDUNDANCY_PORT: The parameter specifies the TCP port number used for
# inter-host communications by redundant instances of the Shelf Manager.
# Default is 1040.
#
REDUNDANCY_PORT = 1040

# REDUNDANCY_NET_ADAPTER: This parameter specifies the name of network
# adapter used for communication between redundant ShMMs. Default is eth0 if
# it does not conflict with RMCP_NET_ADAPTER.
#
REDUNDANCY_NET_ADAPTER = $IP1DEVICE

# REDUNDANCY_NET_ADAPTER2: This parameter specifies the name of the second
# network adapter used for communication between redundant ShMMs (if USB
# interface is used for redundancy). By default, this parameter is not
# defined.
#REDUNDANCY_NET_ADAPTER2 = "usb1"

# REDUNDANT_IP_ADDRESS: This parameter specifies the IP address for network
# adapter used for redundant communications. This address actually provides
# a pair of IP addresses that differ in the least significant bit. They are
# assigned to redundant ShMs according to their hardware addresses, so they
# are equal on both ShMs. This parameter has no default value and must
# always be set.
#
REDUNDANT_IP_ADDRESS = $IP1ADDR

# REDUNDANCY_NETMASK: This parameter sets the network mask for the network
# adapter used for redundancy communications. Default is 255.255.255.0
#
# REDUNDANCY_NETMASK = 255.255.255.0

# RMCP_NET_ADAPTER: This parameter specifies the name of network adapter
# used for RMCP-based communications. Default is eth0:1 if it does not
# conflict with REDUNDANCY_NET_ADAPTER.
#
RMCP_NET_ADAPTER = $IPDEVICE

# RMCP_NET_ADAPTER2: This parameter specifies the alternate name of network
# adapter used for RMCP-based communications, if cross-connect links are
```

```
# supported by hardware. Undefined by default.
#
#RMCP_NET_ADAPTER2 = "eth1"

# DEFAULT_RMCP_IP_ADDRESS: This parameter specifies the default IP address
# for network adapter used for RMCP communications. It is switched over
# between redundant instances of the Shelf Manager. This address is only
# used if no IP address is set in the LAN Configuration Parameters for
# channel # 1. Default is the REDUNDANT_IP_ADDRESS parameter value.
#
DEFAULT_RMCP_IP_ADDRESS = $RMCPADDR

# PROPAGATE_RMCP_ADDRESS: This parameter specifies whether the RMCP IP
# address should be propagated to the backup Shelf Manager. If set, the
# backup Shelf Manager configures its network interface specified by
# RMCP_NET_ADAPTER using given IP address with the least significant bit
# inverted. Default is FALSE.
#
PROPAGATE_RMCP_ADDRESS = FALSE

# DEFAULT_RMCP_NETMASK: This parameter specifies the network mask for
# network adapter used for RMCP communications. Default is 255.255.255.0
#
DEFAULT_RMCP_NETMASK = 255.255.255.0

# DEFAULT_GATEWAY_IP_ADDRESS: This parameter specifies the default gateway
# IP address used for RMCP-based communications. It should be equal for the
# redundant instances of the Shelf Manager. This address is only used if no
# gateway address is set in the LAN Configuration Parameters for channel 1.
# Default is no gateway.
#
DEFAULT_GATEWAY_IP_ADDRESS = $GATEWAY

# SWITCHOVER_TIMEOUT_ON_BROKEN_LINK: This parameter sets the number of
# seconds to wait before switchover if the RMCP link is down, i.e. system
# manager is inaccessible from the shelf manager. A zero value of this
# parameter leads to an immediate switchover on RMCP link fault detection.
# With a -1 value, no automatic switchovers on RMCP link faults will occur.
# The default value is 10 second.
#
SWITCHOVER_TIMEOUT_ON_BROKEN_LINK = 10

# CONSOLE_LOGGING_ENABLED: This parameter of boolean type enables or
# disables log messages output to the console from which the Shelf Manager
# was started. Default is FALSE.
```

```

#
CONSOLE_LOGGING_ENABLED = FALSE

# SYSLOG_LOGGING_ENABLED: This parameter of boolean type enables or disables
# logging messages to the syslog facility. Default is TRUE.
#
SYSLOG_LOGGING_ENABLED = TRUE

# VERBOSITY: This parameter sets the Shelf Manager verbosity level. This
# value is actually a bitmask with each bit enabling a corresponding class
# of output messages. The current bit layout has 8 classes:
#
#      Errors:                0x01
#      Warnings:              0x02
#      Information:           0x04
#      Verbose Info:          0x08
#      Debug Trace Messages: 0x10 (not recommended)
#      Verbose Debug Trace:   0x20 (not recommended)
#      Demo Messages:         0x40 (not recommended)
#      Locks Information:     0x80 (not recommended)
# The default verbosity level is 7 i.e. errors, warnings and information.
#
VERBOSITY = 7

### PICMG 2.x specific settings

# 2_X_SYSTEM: If configured, this parameter explicitly specifies the current
# system as CompactPCI (if TRUE) or AdvancedTCA (if FALSE). If not specified
# the choice of the system type is made automatically. It is not recommended
# to specify this parameter, unless it is necessary to override an incorrect
# hardware detection algorithm for the system type. Default is FALSE.
#
# 2_X_SYSTEM = FALSE

# CTCA_FRU_RESET_TIMEOUT: This parameter specifies the time interval in
# msecs which is used to holds the BD_SEL# line low in order to reset a
# CompactPCI board. Default is 500 milliseconds.
#
# CTCA_FRU_RESET_TIMEOUT = 500

# CTCA_HEALTHY_TIMEOUT: This parameter specifies the time interval in
# seconds during which the Shelf Manager waits for the HEALTHY# signal to
# appear after powering on a CompactPCI board. If the board HEALTHY# signal
# is not detected within the specified time, the Shelf Manager will
# deactivate this board. Default is 0 which means endless waiting.

```

```
#
# CTCA_HEALTHY_TIMEOUT = 0
#
### Notification settings

# MAX_EVENT_SUBSCRIBERS: The parameter defines the maximum number of
# entities
# that can simultaneously subscribe to receive event notifications
# from the Shelf Manager.
#
MAX_EVENT_SUBSCRIBERS = 64

# MAX_PENDING_EVENT_NOTIFICATIONS: The parameter defines the maximum number
# of outstanding event notifications for each active subscriber.
#
MAX_PENDING_EVENT_NOTIFICATIONS = 1024

# MAX_EVENT_SUBSCRIBER_IDLE_TIME: This parameter defines the maximum timeout
# for an event subscriber, in seconds, between the moment when an event
# arrives and the moment when the subscriber retrieves this event from the
# Shelf Manager. If this timeout is exceed, the subscriber is considered
# dead and is automatically unregistered.
#
MAX_EVENT_SUBSCRIBER_IDLE_TIME = 60
```

## 상세 표시 수준 설명

상세 표시 수준을 사용하면 구성 매개 변수 `CONSOLE_LOGGING_ENABLED` 및 `SYSLOG_LOGGING_ENABLED`가 설정된 방법에 따라 콘솔 또는 Syslog에 추가 출력을 보낼 수 있습니다. `VERBOSITY` 구성 매개 변수는 16진수 비트 마스크이며, 각 비트가 특정 유형의 메시지 출력을 활성화합니다.

0x01	오류 메시지
0x02	경고 메시지
0x04	정보 메시지
0x08	자세한 정보 메시지
0x10	추적 메시지
0x20	자세한 추적 메시지
0x40	초기화 중에 IPM 제어기에 전송되는 중요 명령에 대해 표시되는 메시지
0x80	내부 잠금 및 잠금 해제에 대한 자세한 메시지

기본 디버그 수준은 7이며, 오류, 경고 및 정보 메시지의 출력이 허용됩니다.

## 날짜 및 시간 설정

시스템을 처음 실행하면 시계가 설정되어 있지 않으므로 초기화해야 합니다. 처음에는 시계가 1970년 1월 1일로 설정되어 있습니다. 직렬 콘솔을 통해 날짜에 액세스할 수 있습니다.

```
# date
Thu Jan 1 03:16:30 UTC 1970
```

날짜를 변경하려면 `date` 응용 프로그램을 사용하여 올바른 날짜를 입력합니다. `date` 명령의 형식은 `MMDDHHMMSSYYYY`입니다. 여기서 각 문자열은 다음을 의미합니다.

MM	월
DD	일
HH	시(24시간 표기법 사용)
MM	분
SS	초
YYYY	연도

예:

```
# date 04291628002003
Tue Apr 29 16:28:00 UTC 2003
```

날짜가 지속되게 하려면 `hwclock` 응용 프로그램을 사용하여 날짜를 저장해야 합니다.

```
# hwclock -systohc
```

경우에 따라 다음 오류 메시지가 표시될 수도 있습니다.

```
mktime: cannot convert RTC time to UNIX time
```

이 오류는 원래의 날짜가 초기화되지 않은 상태이기 때문에 발생되며 무시할 수 있습니다.



## 시간 서버에서 날짜 및 시간 가져오기

선반 관리 카드에 RTC(실시간 시계) 배터리가 없는 경우 시스템을 시작하는 중에 시간 서버에서 시스템 날짜와 시간을 가져온 다음 주기적으로 동기화할 수 있습니다. 선택된 시간 서버는 `rdate` 유틸리티에서 요구하는 대로 RFC 868 over TCP를 지원해야 합니다. 이 기능을 활성화하려면 U-Boot 변수 `time_server`를 정의하고 선택적으로 추가 변수 `timezone`을 정의해야 합니다.

`time_server` 변수에는 선반 관리자가 시작된 후에 시스템 시간을 쿼리할 시간 서버의 IP 주소가 포함되어 있습니다. 이 변수는 Linux 수준에 환경 변수 `TIMESERVER`로 전파됩니다. 이 변수를 설정하면 시작 스크립트 `/etc/netconfig`가 스크립트 `/etc/timesync`를 데몬으로 시작합니다. 이 스크립트는 무한 반복적으로 실행되어 300초의 기본 간격으로 시간 서버에 쿼리합니다. 이 간격을 변경하려면 스크립트 `/etc/timesync`를 편집하여 `INTERVAL` 변수 값을 변경합니다.

---

주 - `time_server` 변수를 지정한 경우 적절하게 동기화하려면 `ip1device` 변수를 `usb0`으로 설정해야 합니다.

---

`timezone` 변수에는 현재 시간대의 이름 다음에 GMT(그리니치 표준시)로부터의 오프셋이 표시됩니다. 오프셋은 그리니치의 서쪽에 위치한 시간대의 경우 양수이고 동쪽에 위치한 시간대의 경우 음수입니다. 이 변수는 Linux 수준에 환경 변수 `TZ`로 전파됩니다. 이 변수의 기본값은 `UTC0`입니다. 즉, 그리니치 시간과 일치하는 UTC(협정 세계 표준시)입니다.

시간 서버에서 전송되는 시간은 GMT입니다. 선반 관리자의 시간대가 설정되어 있지 않거나 잘못 설정된 경우, 시간 서버에서 가져온 시간이 올바르게 해석되지 않습니다. 시간대의 세 문자 이름은 선반 관리자에서 사용되지 않지만, Linux 시간대를 설정하기 위해 전파됩니다. (예를 들어, 시간대 이름 `XXX0`을 사용하는 경우 `date` 명령은 `Thu Sep 9 21:24:24 XXX 2004`와 같은 출력을 생성합니다.) 일광 절약 시간은 지원되지 않습니다.

다음은 미국 동부 표준시에 대한 `timezone` 정의 예입니다.

```
timezone = EST5
```

여기서 숫자 5는 시간대가 GMT로부터 5시간 서쪽임을 지정합니다. 임의의 세 문자로 EST를 대체할 수 있습니다. 이 문자는 예를 들어, Linux `date` 명령 출력에서 시간대를 식별하는 데 사용됩니다.

## 선반 관리 카드에서 사용자 계정 설정

RMCP 액세스를 위한 사용자 계정은 선반 관리자의 CLI를 사용하여 설정됩니다. 사용자 정보를 활성 선반 관리 카드에 입력하면 해당 정보가 대기 선반 관리 카드에서 즉시 미러되거나 공유됩니다. 선반 관리 카드는 암호가 있는 32개의 계정을 지원합니다.

### ▼ RMCP 액세스를 위한 사용자 계정을 추가하는 방법

1. 활성 선반 관리 카드에 로그인합니다.
2. 다음과 같이 사용자를 추가합니다.

```
# clia user add userid user-name channel-access-flags privilege-level password
```

여기서 변수 매개 변수의 의미는 다음과 같습니다.

*userid* - 유효한 사용자 ID

*user-name* - 사용자 이름(최대 16자)

*channel-access-flag* - SetUserInfo 명령의 첫 번째 바이트(비트 4, 5 및 6만 의미 있음)

- 비트 6 - IPMI 메시징 활성화
- 비트 5 - 링크 인증 활성화
- 비트 4 - 콜백으로 제한

*privilege-level* - 사용자 권한 수준

*password* - 사용자 암호(알림 없이 16자로 잘림)

다음 예에서는 이름이 root이고, 암호가 PICMG guru이고, 관리자 권한 수준을 가진 user 9를 추가하는 방법을 나타냅니다.

```
# clia user add 9 "root" 0x40 4 "PICMG guru"
IPM Sentry Shelf Manager Command Line Interpreter
User 9 added successfully
#
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
9: "root"
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
#
```

권한 및 clia user 명령에 대한 자세한 내용은 [240페이지](#)의 "user"를 참조하십시오.

## 사용자 이름 제한

사용자 이름 필드의 최대 길이는 16자입니다. 2자 이상의 소문자 영문자를 포함해야 하며 첫 번째 문자는 영문자여야 합니다.

`username`에 사용할 수 있는 문자는 다음과 같습니다.

- 영문자
- 숫자
- 마침표(.)
- 밑줄(\_)
- 하이픈(-)

## 암호

암호의 최대 길이는 16자이고 16자를 초과하는 문자는 잘립니다.

---

## 선반 관리자에서 OpenHPI 구성

선반 관리자는 SA Forum HPI(하드웨어 플랫폼 인터페이스)의 개방형 소스 구현인 OpenHPI를 지원합니다. HPI는 일반적으로 새시 및 랙 기반 서버에 대해 컴퓨터 하드웨어 관리를 위한 인터페이스를 제공합니다. SNMP MIB를 사용하여 OpenHPI SNMP 하위 에이전트를 통해 HPI에 액세스할 수 있습니다.

다음 두 구성 파일은 시스템 관리자의 주의가 필요한 파일입니다.

- `/etc/openhpi.conf` - OpenHPI 구성 파일
- `/etc/snmpd.conf` - SNMP 하위 에이전트 구성 파일

### `/etc/openhpi.conf` 파일

ShMM에 올바른 IP 주소를 제공하려면 OpenHPI 구성 파일인 `/etc/openhpi.conf`를 업데이트해야 합니다. 구성 파일을 업데이트한 후 변경 사항을 구현하도록 ShMM을 재설정해야 합니다.

## ▼ /etc/openhpi.conf 파일을 수정하는 방법

1. /etc/openhpi.conf 파일을 편집하고 libipmdirect addr 매개 변수 값을 **ShMM**의 IP 주소로 변경합니다.

/etc/openhpi.conf 샘플 파일이 표시됩니다.

```
OPENHPI_THREADED = "YES"
OPENHPI_UID_MAP = "/var/bin/uip_map"
plugin libipmdirect
handler libipmdirect {
    entity_root = "{SYSTEM_CHASSIS, 1}"
    name = "lan"
    addr = " "
    port = "623"
    auth_type = "{none}"
    auth_level = "admin"
    username = "openhpi"
    password = "openhpi"
    MaxOutstanding = "1"
    ActConnectionTimeout = "5000"
    logflags = " "
    logfile = "log"
    logfile_max = "10"
}
```

2. 프롬프트에서 reboot 명령을 실행하여 **ShMM**을 다시 부트합니다.  
예를 들면 다음과 같습니다.

```
# reboot
```

## /etc/snmpd.conf 파일

SNMP 하위 에이전트 구성 파일인 `/etc/snmpd.conf`는 SNMP 하위 에이전트의 작동 방법을 정의하고 액세스 제어 및 트랩 설정에 대한 지시문이 포함되어 있습니다. 액세스 제어, SNMPv3 구성 및 트랩 설정에 대한 정보는 아래의 해당 절을 참조하십시오.

### 액세스 제어

SNMP 하위 에이전트는 RFC 2575에 정의된 것처럼 VACM(View-Based Access Control Model)을 지원합니다. 따라서 구성 파일에서 다음 키워드를 인식합니다.

- `com2sec`
- `group`
- `access`
- `view`

또한 다음과 같은 몇 가지 사용이 간편한 래퍼 지시문을 인식합니다.

- `rocommunity`
- `rwcommunity`
- `rouser`
- `rwuser`

이 절에서는 다양한 액세스 유형과 수준을 적용하도록 `snmpd` 프로그램을 구성하는 방법을 정의합니다.

```
rouser user [noauth|auth|priv] [OID]
rwuser user [noauth|auth|priv] [OID]
```

VACM 액세스 구성 테이블에서 SNMPv3 USM 사용자를 만듭니다. `group`, `access` 및 `view` 지시문을 결합하여 사용하는 것이 더 효과적이고 강력하지만, 이러한 래퍼 지시문은 훨씬 간단합니다.

사용자가 사용해야 하는 최소 인증 및 프라이버시 수준은 첫 번째 토큰(기본값: `auth`)에 의해 지정됩니다. `OID` 매개 변수는 사용자의 액세스 권한을 지정된 `OID` 아래에 있는 모든 항목으로 제한합니다.

```
rocommunity community [source] [OID]
rwcommunity community [source] [OID]
```

에이전트에 액세스하는 데 사용될 수 있는 읽기 전용 및 읽기/쓰기 커뮤니티를 만듭니다. 이 커뮤니티는 더 복잡하고 강력한 `com2sec`, `group`, `access` 및 `view` 지시문 행 주위의 빠른 래퍼입니다. 이 커뮤니티는 그룹이 생성되지 않아 테이블이 더 커질 수 있으므로 그만큼 효율적이지는 않습니다. 이러한 지시문은 복잡한 환경에서는 사용하지 않는 것이 좋습니다. 사용자의 환경이 상대적으로 단순하거나 성능에 미치는 좋지 않은 영향을 감안할 수 있다면 이 지시문을 사용하십시오.

`source` 토큰의 형식에 대해서는 아래 `com2sec` 지시문 절을 참조하십시오. `OID` 토큰은 해당 커뮤니티의 액세스 권한을 지정된 `OID` 아래의 모든 항목으로 제한합니다.

*com2sec name source community*

*source/community* 쌍에서 보안 *name*으로의 매핑을 지정합니다. *source*는 호스트 이름, 서브넷 또는 단어 *default*입니다. 서브넷을 IP/마스크 또는 IP/비트로 지정할 수 있습니다. 들어오는 패킷과 일치하는 첫 번째 *source/community* 결합이 선택됩니다.

*group name model security*

*securitymodel/securityname*에서 *group*으로의 매핑을 정의합니다. *model*은 *v1*, *v2c*, *usm* 중 하나입니다.

*access name context model level prefix read write notify*

*group/security* 및 *model/security* 수준에서 *view*로 매핑합니다. *model*은 *any*, *v1*, *v2c*, *usm* 중 하나입니다. *level*은 *noauth*, *auth*, *priv* 중 하나입니다. *prefix*는 들어오는 PDU(*exact* 또는 *prefix*)의 컨텍스트를 기준으로 *context*를 일치시키는 방법을 지정합니다. *read*, *write* 및 *notify*는 해당 액세스에 대해 사용할 보기를 지정합니다. *v1* 또는 *v2c* 액세스의 경우 *level*은 *noauth*이고 *context*는 비어 있습니다.

*view name type subtree [mask]*

명명된 보기를 정의합니다. *type*은 *included* 또는 *excluded*입니다. *mask*는 마침표(.) 또는 콜론(:)으로 구분되는 16진수 옥텟 목록입니다. 마스크는 지정되지 않은 경우 기본값 *ff*를 사용합니다. *mask*를 사용하면 테이블의 한 행에 대한 액세스를 상대적으로 간단하게 제어할 수 있습니다. 예를 들어, ISP는 각 고객에게 해당 고객 고유 인터페이스에 대한 액세스 권한을 부여할 수 있습니다.

```
view cust1 included interfaces.ifTable.ifEntry.ifIndex.1 ff.a0
view cust2 included interfaces.ifTable.ifEntry.ifIndex.2 ff.a0

# interfaces.ifTable.ifEntry.ifIndex.1 == .1.3.6.1.2.1.2.2.1.1.1
# ff.a0 == 11111111.10100000
```

이러한 항목은 행 색인을 포함하며 사용자가 행의 필드를 변경할 수 있습니다.

다음은 VACM 예입니다.

```
# sec.name source community
com2sec local localhost private
com2sec mynet 10.10.10.0/24 public
com2sec public default public

# sec.model sec.name
group mygroup v1 mynet
group mygroup v2c mynet
group mygroup usm mynet
group local v1 local
group local v2c local
group local usm local
group public v1 public
group public v2c public
group public usm public

# incl/excl subtree mask
view all included .1 80
view system included system fe
view mib2 included .iso.org.dod.internet.mgmt.mib-2 fc

# context sec.model sec.level prefix read write notify
access mygroup "" any noauth exact mib2 none none
access public "" any noauth exact system none none
access local "" any noauth exact all all all
```

## SNMPv3 구성

`engineID string`

SNMPv3 메시지에 응답하려면 engineID를 사용하여 하위 에이전트를 구성해야 합니다. 이 구성 파일 행에서 engineID는 *string*으로부터 구성됩니다. engineID의 기본값은 시스템의 호스트 이름에 대한 첫 번째 IP 주소로 구성됩니다.

`createUser username (MD5|SHA) authpassphrase [DES] [privpassphrase]`

MD5 및 SHA는 사용할 인증 유형입니다. SHA를 사용하려면 OpenSSL이 설치된 패키지를 구축해야 합니다. DES는 현재 지원되는 유일한 프라이버시 프로토콜입니다. *privpassphrase*를 지정하지 않으면 *authpassphrase*와 동일한 것으로 간주됩니다.

---

주 - 만든 사용자를 위에서 설명한 VACM 액세스 제어 테이블에 추가하지 않으면 해당 사용자는 소용이 없습니다.

---

---

주 - 최소 암호 구문 길이는 8자입니다.

---

## 트랩 설정 및 정보 대상

`trapcommunity string`

트랩을 전송할 때 사용할 기본 커뮤니티 *string*을 정의합니다. 이 명령은 이 커뮤니티 문자열과 함께 사용되는 세 명령 중에서 가장 먼저 사용해야 합니다.

```
trapsink host [community [port]]
trap2sink host [community [port]]
informsink host [community [port]]
```

트랩을 수신할 호스트를 정의합니다. 또는 `informsink`를 사용하여 정보 알림을 수신할 호스트를 정의합니다. 데몬은 시작될 때 콜드 시동(Cold Start) 트랩을 전송합니다. 활성화하면 인증 실패 시에도 트랩을 전송합니다. 여러 `trapsink`, `trap2sink` 및 `informsink` 행을 지정하여 여러 대상을 지정할 수 있습니다. `trap2sink`를 사용하여 SNMPv2 트랩을 전송하고 `informsink`를 사용하여 알림을 전송합니다. *community*를 지정하지 않으면 앞의 `trapcommunity` 지시문의 문자열이 사용됩니다. *port*를 지정하지 않으면 알려진 SNMP 트랩 포트(162)가 사용됩니다.

`trapsess [snmpcmdargs] host`

모든 버전의 SNMP를 사용하여 모든 유형의 트랩 대상을 지정할 수 있도록 해주는 보다 일반적인 트랩 구성 토큰입니다. 이렇게 하려면 v2c 또는 v3의 버전 번호도 함께 지정해야 합니다.

## ▼ /etc/snmpd.conf 파일을 업데이트하는 방법

1. 추가할 `/etc/snmpd.conf` 파일을 편집하고 필요에 따라 지시문을 변경하거나 삭제합니다.
2. 프롬프트에서 `reboot` 명령을 실행하여 **ShMM**을 다시 부트합니다.



# 시스템 관리

---

선반 관리 카드 명령줄 인터페이스(CLI)를 사용하거나 RMCP 인터페이스를 사용하는 이더넷을 통해 시스템을 관리할 수 있습니다.

이 장은 다음 절로 구성되어 있습니다.

- 54페이지의 "선반 관리자 명령줄 인터페이스"
- 60페이지의 "시스템 모니터링"
- 79페이지의 "선반 관리자 다시 초기화"
- 82페이지의 "선반 관리 카드 재프로그래밍"

---

## IPMI LAN 인터페이스

IPMI LAN 인터페이스는 ATCA 사양에 필요하며 RMCP(Remote Management Control Protocol)를 통해 선반 관리자와의 IPMI 메시징을 지원합니다. RMCP를 사용하여 선반과 통신하는 시스템 관리자는 모든 ATCA 호환 선반 관리자와 상호 작용할 수 있습니다. 이 하위 수준의 인터페이스는 시스템 관리자가 선반 관리자를 프록시로 사용하여 선반의 IPM 제어기에 IPMI 명령을 실행할 수 있는 기능을 포함하여 선반의 IPMI 기능에 대한 액세스를 제공합니다.

## IPMI 명령

표준 IPMI 명령은 PICMG 3.0, ATCA 사양에 설명되어 있습니다. 이 사양에서는 사용자 정의 OEM IPMI 명령도 제공합니다. Sun Microsystems는 설계한 ATCA 보드에 고유한 일련의 명령을 제공했습니다. 이러한 명령은 [표 3-1](#)의 목록과 [부록 B](#)의 설명을 참조하십시오.

표 3-1 Sun OEM IPMI 명령

명령	Opcode	구문
Get Version	0x80	#GET_VERSION
Set Boot Page <sup>†</sup>	0x81	#SET_BOOT_PAGE
Get Boot Page <sup>†</sup>	0x82	#GET_BOOT_PAGE
Set Front panel reset button state	0x83	#SET_FP_RESET_BUTTON
Get Front panel reset button state	0x84	#GET_FP_RESET_BUTTON_STATE
Set Ethernet Force Front bit	0x85	#SET_ETH_FORCE_FRONT
Get Ethernet Force Front bit	0x86	#GET_ETH_FORCE_FRONT
Get RTM status	0x88	#GET_RTM_PRESENCE

<sup>†</sup> Sun Netra™ CP3010 노드 보드에만 유효

## 선반 관리자 명령줄 인터페이스

선반 관리자 명령줄 인터페이스(CLI)를 사용하면 텍스트 명령을 통해 선반의 지능형 관리 제어기, 보드 및 선반 관리자 자체와 통신할 수 있습니다. CLI는 직접 액세스하거나 상위 수준 관리 응용 프로그램 또는 스크립트를 통해 액세스할 수 있는 IPMI 기반 명령 집합입니다. 관리자는 텔넷 연결 또는 선반 관리 카드 직렬 포트를 통해 CLI에 액세스할 수 있습니다. 운영자는 CLI를 사용하여 현재 FRU 채우기, 현재 센서 값, 임계값 설정, 최신 이벤트, 전체 선반 상태 등과 같은 선반의 현재 상태 정보에 액세스할 수 있습니다.

## 명령줄 인터페이스 시작

CLI를 사용하려면 먼저 선반 관리 카드의 Linux 운영 체제에 로그인합니다. 로그인한 후 특정 매개 변수를 사용하여 명령줄에서 실행 파일 `clia`를 실행합니다. 첫 번째 매개 변수는 명령 동사입니다. `clia` 실행 파일은 선반 관리 카드에서 실행 중인 Linux가 유지 관리하는 가상 루트 파일 시스템에 있습니다. `clia` 실행 파일은 주 선반 관리자 소프트웨어 프로세스에 연결하여 명령 정보를 전달한 다음 결과를 검색합니다. CLI를 시작하려면 먼저 선반 관리자를 실행해야 합니다.

예를 들면 다음과 같습니다.

```
# clia ipmc

IPM Sentry Shelf Manager Command Line Interpreter

20: Entity: (d0, 0) Maximum FRU device ID: 20
    PICMG Version 2.0
    Hot Swap State: M4, Previous: M3, Last State Change Cause: Normal State
    Change (0)
#
```

매개 변수 없이 `clia`를 시작하면 대화식 모드로 스위치오버됩니다. 이 모드에서 프로그램은 터미널에 프롬프트를 반복적으로 실행하고 매개 변수와 함께 사용자 입력을 다음 명령으로 받아 해당 명령을 실행한 다음 그 결과를 터미널에 표시합니다. 이 과정은 사용자가 `exit` 또는 `quit` 명령을 입력할 때까지 계속됩니다. 예를 들면 다음과 같습니다.

```
# clia

IPM Sentry Shelf Manager Command Line Interpreter

CLI> ipmc 20

20: Entity: (d0, 0) Maximum FRU device ID: 20
    PICMG Version 2.0
    Hot Swap State: M4, Previous: M3, Last State Change Cause: Normal State
    Change (0)

CLI> exit
#
```

# CLI 명령

CLI는 다음 명령을 구현합니다. 명령에 대한 자세한 내용은 [109페이지의 부록 A "선반 관리자 CLI 명령"](#)에 사전순으로 나열된 각 명령의 세부절을 참조하십시오.

표 3-2 선반 관리자 CLI 명령 요약

명령	매개 변수	설명
activate	IPMB 주소 FRU 장치 ID	지정된 FRU를 활성화합니다.
alarm	알람 유형	Telco 알람을 활성화하거나 지웁니다.
board	슬롯 번호(선택 사항)	보드에 대한 정보를 표시합니다.
boardreset	슬롯 번호	지정된 ATCA 보드를 재설정합니다.
busres	하위 명령, 매개 변수 사용	버스된 전자 키잉 관리 자원에 대한 지정된 작업을 수행합니다.
console	슬롯 번호	지정된 슬롯에 있는 노드 보드에서 콘솔 세션을 엽니다.
deactivate	IPMB 주소 FRU 장치 ID	지정된 FRU를 비활성화합니다.
debuglevel	새 디버그 수준(선택 사항)	선반 관리자에 대한 현재 디버그 수준을 가져오거나 새 디버그 수준을 설정합니다.
exit/quit		대화식 모드에서 해석기를 종료합니다.
fans	IPMB 주소(선택 사항) FRU 장치 ID(선택 사항)	팬에 대한 정보를 표시합니다.
flashupdate	서버 IP 주소 펌웨어 이미지에 대한 경로 이름	지정된 서버와 경로 이름에서 Netra CP3060 노드 보드에 대한 시스템 펌웨어를 다운로드하고 업데이트합니다.
fru	IPMB 주소(선택 사항) FRU 장치 ID(선택 사항) FRU 유형(선택 사항)	선반에 있는 단일 FRU 또는 FRU 그룹에 대한 정보를 표시합니다. FRU는 유형별 또는 부모 IPM 제어기별로 선택됩니다.
frucontrol	IPMB 주소 FRU 장치 ID 옵션	FRU 제어 명령을 특정 FRU로 보냅니다.
frudata	IPMB 주소(선택 사항) FRU 장치 ID(선택 사항) 블록/바이트 오프셋(선택 사항) 데이터(선택 사항)	지정된 FRU의 FRU 정보에 대한 원시 액세스를 제공합니다.

표 3-2 선반 관리자 CLI 명령 요약(계속)

명령	매개 변수	설명
frudatar	IPMB 주소 FRU 장치 ID 파일 이름	지정된 FRU의 FRU 데이터 영역을 읽고 데이터를 지정된 파일에 저장합니다.
frudataw	IPMB 주소 FRU 장치 ID 파일 이름	지정된 파일의 FRU 데이터를 지정된 FRU의 FRU 데이터 영역에 씁니다.
fruinfo	IPMB 주소 FRU 장치 ID	사용자에게 친숙한 FRU 정보 출력을 제공합니다.
getfanlevel	IPMB 주소(선택 사항) FRU 장치 ID(선택 사항)	지정된 FRU에서 제어하는 팬의 현재 수준을 표시합니다.
getfruledstate	IPMB 주소(선택 사항) FRU 장치 ID(선택 사항) LED ID 또는 ALL(선택 사항)	FRU LED 상태를 표시합니다.
gethysteresis	IPMB 주소(선택 사항) 센서 이름(선택 사항) 센서 번호(선택 사항)	지정된 센서의 양수 및 음수 자기 이력 현상을 모두 표시합니다.
getipmbstate	IPMB 주소 IPMB 링크 번호(선택 사항)	대상 주소에 있는 IPMB-0의 현재 상태를 표시합니다. 링크 번호가 지정되어 있고 대상 IPMC가 IPMB 허브인 경우 해당 링크에 대한 정보가 표시됩니다.
getlanconfig	채널 번호 매개 변수 이름 또는 번호(선택 사항) 선택기 설정(선택 사항)	특정 채널에 대한 LAN 구성 매개 변수를 가져와서 표시합니다.
getpefconfig	매개 변수 이름 또는 번호(선택 사항) 선택기 설정(선택 사항)	PEF 구성 매개 변수를 가져와서 표시합니다.
getsensoreventenable	IPMB 주소(선택 사항) 센서 이름(선택 사항) 센서 번호(선택 사항)	지정된 센서의 지원되는 이벤트에 대한 현재 센서 이벤트 마스크 값을 표시합니다.
getthreshold, threshold	IPMB 주소(선택 사항) 센서 이름(선택 사항) 센서 번호(선택 사항)	특정 센서에 대한 임계값 정보를 표시합니다.
help		지원되는 명령 목록을 표시합니다.
ipmc	IPMB 주소(선택 사항)	선반에 있는 하나 또는 모든 IPM 제어기에 대한 정보를 표시합니다.
localaddress		현재 선반 관리자의 IPMB 주소를 검색합니다.

표 3-2 선반 관리자 CLI 명령 요약(계속)

명령	매개 변수	설명
minfanlevel	팬 수준(선택 사항)	최소 팬 수준을 표시하거나 설정합니다.
sel	IPMB 주소(선택 사항) 항목 수(선택 사항)	대상 IPM 제어기에 유지 관리되는 시스템 이벤트 로그에서 다양한 최신 항목을 표시합니다.
sensor	IPMB 주소(선택 사항) 센서 이름(선택 사항) 센서 번호(선택 사항)	단일 센서 또는 센서 그룹에 대한 정보를 표시합니다. 센서는 IPM 제어기 주소, 번호 또는 이름별로 선택됩니다.
sensordata	IPMB 주소(선택 사항) 센서 이름(선택 사항) 센서 번호(선택 사항)	특정 센서에 대한 값 정보를 표시합니다.
sensorread	IPMB 주소 센서 번호	특정 센서에 대한 원시 값 정보를 표시합니다(센서를 설명하는 센서 데이터 레코드 무시).
session		활성 RMCP 세션에 대한 정보를 표시합니다.
setextracted	IPMB 주소 FRU 장치 ID	지정된 FRU가 선반에서 물리적으로 추출되었음을 선반 관리자에게 알립니다.
setfanlevel	IPMB 주소 FRU 장치 ID 수준	지정된 FRU에서 제어하는 팬에 대한 새 수준을 설정합니다.
setfruledstate	IPMB 주소 FRU 장치 ID LED ID 또는 ALL LED 작동 LED 색상(선택 사항)	지정된 FRU에 대한 특정 LED 또는 모든 LED의 상태를 설정합니다.
sethysteresis	IPMB 주소 센서 이름 또는 센서 번호 설정할 히스테리시스(pos 또는 neg) 히스테리시스 값	지정된 센서에 대한 새 히스테리시스 값을 설정합니다.
setipmbstate	IPMB 주소 IPMB 버스 이름(A 또는 B) IPMB 링크 번호(선택 사항) 수행할 작업	대상 IPM 제어기에서 IPMB-A, IPMB-B 또는 특정 IPMB 링크를 비활성화/활성화합니다.
setlanconfig	채널 매개 변수 이름 또는 번호 추가 매개 변수	지정된 채널에 대한 LAN 구성 매개 변수 값을 설정합니다.

표 3-2 선반 관리자 CLI 명령 요약(계속)

명령	매개 변수	설명
setlocked	IPMB 주소 FRU 장치 ID 상태	지정된 FRU에 대한 잠금 비트를 지정된 상태(0 - 잠금 해제, 1 - 잠금)로 설정합니다.
setpefconfig	매개 변수 이름 또는 번호 선택기 설정(선택 사항) 매개 변수 값	새 PEF 구성 매개 변수 값을 설정합니다.
setpowerlevel	IPMB 주소 FRU 장치 ID 전원 수준 또는 OFF Copy	지정된 FRU의 전원 수준을 설정하고 FRU에 대한 전원을 끄고 원하는 수준을 현재 수준에 복사합니다.
setsensoreventenable	IPMB 주소 센서 이름 센서 번호 전역 플래그 어설션 이벤트 마스크(선택 사항) 디어설션 이벤트 마스크(선택 사항)	특정 센서에 대한 이벤트 사용 마스크를 변경합니다.
setthreshold	IPMB 주소 센서 이름 센서 번호 임계값 유형 임계값	특정 센서에 대한 특정 임계값 (upper/lower, critical/non-critical/non-recoverable)을 변경합니다.
shelf	하위 명령, 매개 변수 사용	선반에 대한 일반 정보를 표시합니다. 다양한 하위 명령을 사용하여 선반 속성을 설정하고 특정 영역에 대한 추가 정보를 가져올 수 있습니다.
shelfaddress	선반 주소 문자열(선택 사항)	선반 FRU 정보에서 주소 테이블의 선반 주소 필드를 가져오거나 설정합니다.
shmstatus		선반 관리자 활성화/백업 상태를 표시합니다.
showhost	슬롯 번호	Netra CP3060 보드의 펌웨어에 대한 버전 정보를 표시합니다.
showunhealthy		선반의 비정상 구성 요소를 표시합니다.
switchover		백업 선반 관리자로의 스위치오버를 시작합니다.

표 3-2 선반 관리자 CLI 명령 요약(계속)

명령	매개 변수	설명
terminate		선반 관리 카드를 다시 부트하지 않고 선반 관리자를 종료합니다.
user	하위 명령, 매개 변수 사용	선반 관리자에 있는 RMCP 사용자 계정에 대한 정보를 표시하고 사용자 계정을 추가, 삭제 및 수정할 수 있는 간단한 방법을 제공합니다.
version		선반 관리자 버전 정보를 표시합니다.

대부분의 정보 명령은 제공되는 정보 양에 따라 단순 모드와 세부 정보 표시 모드를 지원합니다. 단순 모드가 기본(표준) 모드입니다. 세부 정보 표시 모드를 선택하려면 명령 줄에서 명령 바로 뒤, 그리고 위치 인수 앞에 `-v` 옵션을 사용합니다.

## 시스템 모니터링

선반 관리자 CLI는 시스템을 모니터링하여 시스템 상태를 표시할 수 있는 많은 명령을 제공합니다. 이 절에서는 시스템을 모니터링하는 다양한 방법을 설명합니다. 자세한 내용은 [56페이지의 "CLI 명령"](#) 또는 [109페이지의 부록 A "선반 관리자 CLI 명령"](#)을 참조하십시오.

## 보드 및 IPMC 정보 표시

보드 정보는 ATCA 슬롯에 할당된 IPMB 주소 범위의 각 IPM 제어기에 대한 정보와 해당 제어기에 의해 제어되는 추가 FRU에 대한 정보를 포함합니다. 보드에 자체 IPM 제어기가 있는 PICMG 3.0 시스템의 경우 IPMB 주소 범위는 82h-A0h입니다.

다음은 사용된 명령과 해당 출력을 나타내는 예입니다.

- [서버의 모든 보드에 대한 표준 정보 표시](#)
- [보드에 대한 자세한 정보 표시](#)
- [보드에 있는 센서 나열](#)
- [보드의 센서에서 데이터 표시](#)
- [서버의 모든 IPMC 나열](#)
- [특정 IPM 제어기에 대한 정보 표시](#)
- [IPM 제어기에 대한 자세한 정보 표시](#)



- 서버의 모든 보드에 대한 표준 정보 표시

이 예에서는 물리 슬롯 1과 14에 있는 보드만 표시됩니다.

```
# clia board
IPM Sentry Shelf Manager Command Line Interpreter
Physical Slot # 1
82: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
82: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
Physical Slot # 14
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
#
```

- 보드에 대한 자세한 정보 표시

이 예에서는 물리 슬롯 14에 있는 보드에 대한 자세한 정보를 나타냅니다.

```
# clia board -v 14
Physical Slot # 14
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID: 0x00, Revision: 0, Firmware: 1.01, IPMI ver 1.5
    Manufacturer ID: 00315a (PICMG), Product ID: 0000, Auxiliary
Rev: 01ac1014
    Device ID String: "IPM Sentry 6"
    Global Initialization: 0x0, Power State Notification: 0x0,
Device Capabilities: 0x29
    Controller provides Device SDRs
    Supported features: 0x29
```

```

        "Sensor Device" "FRU Inventory Device" "IPMB Event
Generator"
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
    Site Type: 0x00, Site Number: 14
    Current Power Level: 0x01, Maximum Power Level: 0x01, Current
Power Consumption: 20.0 Watts
#

```

● 보드에 있는 센서 나열

이 예에서는 IPMB 주소 92에 있는 보드의 센서 목록을 표시합니다.

```

# clia sensor 92
IPM Sentry Shelf Manager Command Line Interpreter

92: LUN: 0, Sensor # 0 ("Hot Swap")
    Type: Discrete (0x6f), "Hot Swap" (0xf0)
    Belongs to entity: (0xa0, 96) [FRU # 0]
92: LUN: 0, Sensor # 1 ("IPMB Physical")
    Type: Discrete (0x6f), "IPMB Link" (0xf1)
    Belongs to entity: (0xa0, 96) [FRU # 0]

92: LUN: 0, Sensor # 255 ("BMC Watchdog")
    Type: Discrete (0x6f), "Watchdog 2" (0x23)
    Belongs to entity: (0x3, 96) [FRU # 0]

92: LUN: 0, Sensor # 255 ("+12.0V")
    Type: Threshold (0x01), "Voltage" (0x02)
    Belongs to entity: (0x14, 96) [FRU # 0]

92: LUN: 0, Sensor # 255 ("+5.0V")
    Type: Threshold (0x01), "Voltage" (0x02)
    Belongs to entity: (0x14, 96) [FRU # 0]

92: LUN: 0, Sensor # 255 ("+3.3V")
    Type: Threshold (0x01), "Voltage" (0x02)
    Belongs to entity: (0x14, 96) [FRU # 0]

92: LUN: 0, Sensor # 255 ("+2.5V")
    Type: Threshold (0x01), "Voltage" (0x02)
    Belongs to entity: (0x14, 96) [FRU # 0]

92: LUN: 0, Sensor # 2 ("CPU1 Temp")

```

```

Type: Threshold (0x01), "Temperature" (0x01)
Belongs to entity: (0x3, 96) [FRU # 0]

92: LUN: 0, Sensor # 3 ("CPU2 Temp")
Type: Threshold (0x01), "Temperature" (0x01)
Belongs to entity: (0x3, 96) [FRU # 0]

92: LUN: 0, Sensor # 4 ("Inlet Temp")
Type: Threshold (0x01), "Temperature" (0x01)
Belongs to entity: (0x3, 96) [FRU # 0]
Type: Threshold (0x01), "Temperature" (0x01)
Belongs to entity: (0x3, 96) [FRU # 0]
#

```

- 보드의 센서에서 데이터 표시

이 예에서는 IPMB 주소 92에 있는 보드의 센서 번호 3(CPU2 Temp)에 대한 정보를 표시합니다.

```

# clia sensor 92 3
IPM Sentry Shelf Manager Command Line Interpreter

92: LUN: 0, Sensor # 3 ("CPU2 Temp")
Type: Threshold (0x01), "Temperature" (0x01)
Belongs to entity: (0x3, 96) [FRU # 0]

```

- 서버의 모든 IPMC 나열

이 예에서는 ipmc 명령의 일반적인 출력을 나타냅니다.

```

# clia ipmc
IPM Sentry Shelf Manager Command Line Interpreter
10: Entity: (0xf0, 0x60) Maximum FRU device ID: 0x08
    PICMG Version 2.1
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)

20: Entity: (0xf0, 0x1) Maximum FRU device ID: 0x08
    PICMG Version 2.1
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)

82: Entity: (0xa0, 0x60) Maximum FRU device ID: 0x00
    PICMG Version 2.1
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)

```

```

88: Entity: (0xa0, 0x60) Maximum FRU device ID: 0x00
    Hot Swap State: M7 (Communication Lost), Previous: M4 (Active),
    Last State Change Cause: Communication Lost (0x4)

92: Entity: (0xa0, 0x60) Maximum FRU device ID: 0x00
    PICMG Version 2.1
    Hot Swap State: M4 (Active), Previous: M7 (Communication Lost),
    Last State Change Cause: Communication Lost (0x4)

96: Entity: (0xa0, 0x60) Maximum FRU device ID: 0x00
    Hot Swap State: M7 (Communication Lost), Previous: M6
    (Deactivation In Progress), Last State Change Cause: Communication
    Lost (0x4)

20: Entity: (0xf0, 0x1) Maximum FRU device ID: 0x08
    PICMG Version 2.1
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
    Process), Last State Change Cause: Normal State Change (0x0)

```

- 특정 IPM 제어기에 대한 정보 표시

이 예에서는 주소 9C에 있는 IPM 제어기에 대한 기본 정보를 표시합니다.

```

# clia ipmc 9c
IPM Sentry Shelf Manager Command Line Interpreter
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
    Process), Last State Change Cause: Normal State Change (0x0)
#

```

- IPM 제어기에 대한 자세한 정보 표시

이 예에서는 주소 9C에 있는 IPM 제어기에 대한 자세한 정보를 표시합니다.

```

# clia ipmc -v 9c
IPM Sentry Shelf Manager Command Line Interpreter
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
    Process), Last State Change Cause: Normal State Change (0x0)
    Device ID: 0x00, Revision: 0, Firmware: 1.01, IPMI ver 1.5
    Manufacturer ID: 00315a (PICMG), Product ID: 0000, Auxiliary
    Rev: 01ac10ac
    Device ID String: "IPM Sentry 6"
    Global Initialization: 0x0, Power State Notification: 0x0,
    Device Capabilities: 0x29
    Controller provides Device SDRs

```

```
Supported features: 0x29
"Sensor Device" "FRU Inventory Device" "IPMB Event
Generator"
#
```

## FRU 정보 표시

clia fru 명령을 매개 변수 없이 실행하여 시스템의 모든 FRU에 대한 정보를 표시하거나, FRU 주소를 제공하고 선택적으로 FRU ID를 입력하여 특정 FRU에 대한 정보를 표시할 수 있습니다. 새시 슬롯 번호를 IPMB 주소의 물리적 주소로 매핑하려면 [12페이지의 "물리 주소-논리 슬롯 매핑"](#)을 참조하십시오.

다음 예의 "20: FRU # 1" 행에서 20은 미드프레인의 IPMB 주소이고 1은 FRU 장치 ID입니다.

```
# clia fru
20: FRU # 1
Entity: (0xf2, 0x60)
Hot Swap State: M4 (Active), Previous: M3 (Activation In Process),
Last State Change Cause: Normal State Change (0x0)
Device ID String: "Shelf EEPROM 1"
```

## IPMI FRU 정보 레이아웃

[그림 3-1](#)에서는 IPMI FRU 정보가 구성되는 방법을 나타냅니다. 각 분할 영역은 특정 데이터 유형을 제공합니다.

- 공통 헤더에는 영역 오프셋이 포함되어 있습니다.
- 내부 사용 영역은 독점 데이터를 저장하는 데 사용됩니다.
- 새시 정보 영역에는 새시 유형, 부품 번호 및 일련 번호가 포함되어 있습니다.
- 보드 정보 영역에는 제조업체 시간 기록, 제조업체, 제품 이름, 부품/일련 번호가 포함되어 있습니다.
- 제품 정보 영역에는 제조업체, 제품 이름, 부품/일련 번호, 버전 등이 포함되어 있습니다.

- 다중 레코드 영역에는 동적 데이터가 포함되어 있습니다.

공통 헤더
내부 사용 영역
새시 정보
보드 정보
제품 정보
다중 기록 정보

그림 3-1 IPMI FRU 정보 레이아웃

## 환경 FRU

환경 FRU는 미드프레인, 선반 관리 카드, 팬, 전원 입력 모듈(PEM) 및 선반 알람 패널(SAP)을 포함합니다. 모든 환경 FRU에는 타사 공급업체에서 제공하고 프로그래밍한 IPMI FRU 정보만 있습니다.

미드프레인 FRU 정보는 Sun 부품 번호와 타사 부품 번호를 포함합니다. 또한 Sun은 미드프레인 FRU 정보의 다중 레코드 영역에 슬롯, vlan, vtag 및 기타 데이터와 같은 추가 시스템 정보를 추가합니다. 미드프레인 FRU 정보는 두 개의 동일한 EEPROM에 저장됩니다. 하나의 EEPROM을 변경하면 다른 EEPROM이 자동으로 변경됩니다.

## 블레이드 FRU

슬롯 7 및 8에 있는 Netra CT 900 서버 허브 보드에는 IPMI FRU 정보만 있습니다. Sun 노드 보드에는 IPMI FRU 정보와 Sun FRU 정보가 각각 포함되어 있는 두 개의 개별 EEPROM이 있습니다.

## 예

다음은 사용된 명령과 해당 출력을 나타내는 예입니다.

- 선반의 모든 FRU에 대한 표준 정보 표시
- 주소 9C의 모든 FRU에 대한 표준 정보 표시

- 주소 20의 FRU 1에 대한 자세한 정보 표시
  - 원시 형식으로 FRU 정보 표시
  - 사용자에게 친숙한 형식으로 FRU 정보 표시
  - 선반의 모든 FRU에 대한 표준 정보 표시
- 이 예에서는 fru 명령을 인수 없이 사용할 때 표시되는 표준 FRU 정보를 나타냅니다.

```
# clia fru
IPM Sentry Shelf Manager Command Line Interpreter

10: FRU # 0
    Entity: (0xf0, 0x60)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "ShMM-500"

12: FRU # 0
    Entity: (0xf0, 0x60)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "ShMM-500"

20: FRU # 0
    Entity: (0xf0, 0x1)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry BMC"

20: FRU # 1
    Entity: (0xf2, 0x60)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "Shelf EEPROM 1"

20: FRU # 2
    Entity: (0xf2, 0x61)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "Shelf EEPROM 2"

20: FRU # 3
    Entity: (0x7, 0x6f)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "SAP Board"
```

```

20: FRU # 4
    Entity: (0x1e, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "Fan Tray 0"

20: FRU # 5
    Entity: (0x1e, 0x1)
    Device ID String: "Fan Tray 1" Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)

20: FRU # 6
    Entity: (0x1e, 0x2)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "Fan Tray 2"

20: FRU # 7
    Entity: (0xa, 0x60)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "PEM A"

20: FRU # 8
    Entity: (0xa, 0x61)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "PEM B"

82: FRU # 0
    Entity: (0xa0, 0x60)
    Hot Swap State: M7 (Communication Lost), Previous: M4 (Active),
Last State Change Cause: Unknown (0xf)
    Device ID String: "ATS1460"

9a: FRU # 0
    Entity: (0xa0, 0x60)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "NetraCP-3010"

```



- 주소 9C의 모든 FRU에 대한 표준 정보 표시

이 예에서는 물리적 주소 9C의 모든 FRU에 대한 FRU 정보만 표시됩니다.

```
# clia fru 9c
IPM Sentry Shelf Manager Command Line Interpreter
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
#
```

- 주소 20의 FRU 1에 대한 자세한 정보 표시

이 예에서는 물리적 주소 20의 FRU 장치 ID 1에 대한 자세한 FRU 정보를 표시합니다.

```
# clia fruinfo 20 1
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1, FRU Info
Common Header:      Format Version = 1

Internal Use Area:
    Version = 1
Chassis Info Area:
    Version      = 1
    Chassis Type          = (23)
    Chassis Part Number   = 11592-450
    Chassis Serial Number =

Board Info Area:
    Version      = 1
    Language Code          = 25
    Mfg Date/Time          = Jun 16 00:00:00 2005 (4973760 minutes
                           since 1996)
    Board Manufacturer     = Schroff
    Board Product Name     = ShMM-ACB-III Shelf Manager (Radial
                           IPMB)
    Board Serial Number    = 0000001
    Board Part Number      = 21593-251
    FRU Programmer File ID = Schroff_11592450_AA.inf

Product Info Area:
    Version      = 1
    Language Code          = 25
    Manufacturer Name     = Schroff
```

```

Product Name           = 12U 14-Slot ATCA Chassis
Product Part / Model#  = 11592-450
Product Version        = Dual Star (Radial IPMB)
Product Serial Number  = 0000001
Asset Tag              =
FRU Programmer File ID = Schroff_11592450_AA.inf

```

Multi Record Area:

```

PICMG Shelf Manager IP Connection Record (ID=0x13)
    Version = 1

```

```

Record Type              = Management Access Record
    Version = 2
Sub-Record Type: Component Name (0x05)

```

```

PICMG Address Table Record (ID=0x10)
    Version = 0

```

```

PICMG Backplane Point-to-Point Connectivity Record (ID=0x04)
    Version = 0

```

```

PICMG Backplane Point-to-Point Connectivity Record (ID=0x04)
    Version = 0

```

```

PICMG Shelf Activation And Power Management Record (ID=0x12)
    Version = 0

```

```

PICMG Shelf Power Distribution Record (ID=0x11)
    Version = 0

```

```

PICMG Radial IPMB-0 Link Mapping Record (ID=0x15)
    Version = 0

```

```

Record Type              = 0xf0 OEM Record
    Version = 2

```

● 원시 형식으로 **FRU** 정보 표시

이 예에서는 모든 FRU 및 특정 FRU에 대한 FRU 정보를 원시 형식으로 표시합니다.

```

# clia frudata
IPM Sentry Shelf Manager Command Line Interpreter
10: FRU # 0 Raw FRU Info Data
    FRU Info size: 435
12: FRU # 0 Raw FRU Info Data
    FRU Info size: 435

```

```

20: FRU # 0 Raw FRU Info Data
    FRU Info size: 152
20: FRU # 1 Raw FRU Info Data
    FRU Info size: 8192
20: FRU # 2 Raw FRU Info Data
    FRU Info size: 8192
20: FRU # 3 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 4 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 5 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 6 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 7 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 8 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 254 Raw FRU Info Data
    FRU Info size: 3068
9a: FRU # 0 Raw FRU Info Data
    FRU Info size: 512
#
# clia frudata 20 1 0
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1 Block # 0 Raw FRU Info Data
    FRU Info size: 8192
    01 01 22 24 31 3E 00 49 01 A0 A1 A2 A3 A4 A5 A6
    A7 A8 A9 AA AB AC AD E0 E1 E2 E3 E4 E5 E6 D0 D1

```

● 사용자에게 친숙한 형식으로 **FRU** 정보 표시

이 예에서는 사용자에게 친숙한 형식으로 FRU 정보를 표시합니다.

```

# clia fruinfo 20 1
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1, FRU Info
Common Header:      Format Version = 1

Internal Use Area:
    Version = 1
Chassis Info Area:
    Version      = 1
    Chassis Type          = (23)
    Chassis Part Number   = 11592-450
    Chassis Serial Number =

```

Board Info Area:

Version = 1  
Language Code = 25  
Mfg Date/Time = Jun 16 00:00:00 2005 (4973760 minutes  
since 1996)  
Board Manufacturer = Schroff  
Board Product Name = ShMM-ACB-III Shelf Manager (Radial  
IPMB)  
Board Serial Number = 0000001  
Board Part Number = 21593-251  
FRU Programmer File ID = Schroff\_11592450\_AA.inf

Product Info Area:

Version = 1  
Language Code = 25  
Manufacturer Name = Schroff  
Product Name = 12U 14-Slot ATCA Chassis  
Product Part / Model# = 11592-450  
Product Version = Dual Star (Radial IPMB)  
Product Serial Number = 0000001  
Asset Tag =  
FRU Programmer File ID = Schroff\_11592450\_AA.inf

Multi Record Area:

PICMG Shelf Manager IP Connection Record (ID=0x13)  
Version = 1

Record Type = Management Access Record  
Version = 2  
Sub-Record Type: Component Name (0x05)

PICMG Address Table Record (ID=0x10)  
Version = 0

PICMG Backplane Point-to-Point Connectivity Record (ID=0x04)  
Version = 0

PICMG Backplane Point-to-Point Connectivity Record (ID=0x04)  
Version = 0

PICMG Shelf Activation And Power Management Record (ID=0x12)  
Version = 0

```
PICMG Shelf Power Distribution Record (ID=0x11)
    Version = 0

PICMG Radial IPMB-0 Link Mapping Record (ID=0x15)
    Version = 0

Record Type                = 0xf0 OEM Record
    Version = 2
UNKNOWN Manufacturer ID = 0x303833
```

## 선반 정보 표시

`clia shelf` 명령을 유효 매개 변수 중 하나와 함께 사용하여 주요 선반 FRU에 대한 FRU 정보를 표시할 수 있습니다. 또한 선반에 대한 현재 작업 데이터를 가져올 수 있고 선반 FRU 정보의 일부 필드를 수정할 수도 있습니다. 유효한 `shelf` 매개 변수는 다음과 같습니다.

- `cooling_state` 또는 `cs`
- `fans_state` 또는 `fs`
- `address_table` 또는 `at`
- `power_distribution` 또는 `pd`
- `power_management` 또는 `pm`
- `pci_connectivity` 또는 `pcic`
- `ha_connectivity` 또는 `ha`
- `h110_connectivity` 또는 `h110c`
- `point-to-point_connectivity` 또는 `ppc`

자세한 내용은 [212페이지의 "선반 FRU 정보 표시"](#)를 참조하십시오.

## 예

다음은 사용된 명령과 해당 출력을 나타내는 예입니다.

- [선반 냉각 상태 표시](#)
- [선반 팬 상태 표시](#)
- [주소 테이블 표시](#)
- [전원 관리 정보 표시](#)
- [배전 정보 표시](#)

- 선반 냉각 상태 표시

이 예에서는 선반의 냉각 상태를 표시하는 명령과 출력을 나타냅니다.

```
# clia shelf cooling_state
IPM Sentry Shelf Manager Command Line Interpreter
Cooling state: "Normal"

# clia shelf -v cooling_state
IPM Sentry Shelf Manager Command Line Interpreter

Cooling state: "Normal"
Sensor(s) at this state: (0x9a,4,0) (0x9a,5,0) (0x10,2,0) (0x9a,3,0)
                        (0x20,120,0) (0x20,121,0) (0x20,122,0) (0x20,123,0)
                        (0x20,200,0) (0x20,201,0) (0x20,240,0) (0x20,241,0)
                        (0x20,242,0)
```

- 선반 팬 상태 표시

이 예에서는 팬 상태와 출력을 표시하는 명령을 나타냅니다.

```
# clia shelf fans_state
IPM Sentry Shelf Manager Command Line Interpreter
Fans state: "Normal"

# clia shelf -v fans_state
IPM Sentry Shelf Manager Command Line Interpreter
Fans state: "Normal"
Sensor(s) at this state: (0x10,7,0) (0x10,8,0) (0x10,9,0) (0x10,10,0)
                        (0x10,11,0) (0x10,12,0)

#
```

- 주소 테이블 표시

이 예에서는 선반 주소 테이블을 표시하는 명령과 출력을 나타냅니다.

```
# clia shelf address_table

IPM Sentry Shelf Manager Command Line Interpreter

PICMG Address Table Record (ID=0x10)
Version = 0
Shelf Address          = 1
Address Table Entries# = 16
Hw Addr: 41, Site # 7, Type: "AdvancedTCA Board" 00
Hw Addr: 42, Site # 8, Type: "AdvancedTCA Board" 00
Hw Addr: 43, Site # 6, Type: "AdvancedTCA Board" 00
Hw Addr: 44, Site # 9, Type: "AdvancedTCA Board" 00
Hw Addr: 45, Site # 5, Type: "AdvancedTCA Board" 00
Hw Addr: 46, Site # 10, Type: "AdvancedTCA Board" 00
```

```

Hw Addr: 47, Site # 4, Type: "AdvancedTCA Board" 00
Hw Addr: 48, Site # 11, Type: "AdvancedTCA Board" 00
Hw Addr: 49, Site # 3, Type: "AdvancedTCA Board" 00
Hw Addr: 4a, Site # 12, Type: "AdvancedTCA Board" 00
Hw Addr: 4b, Site # 2, Type: "AdvancedTCA Board" 00
Hw Addr: 4c, Site # 13, Type: "AdvancedTCA Board" 00
Hw Addr: 4d, Site # 1, Type: "AdvancedTCA Board" 00
Hw Addr: 4e, Site # 14, Type: "AdvancedTCA Board" 00
Hw Addr: 08, Site # 1, Type: "Dedicated ShMC" 03
Hw Addr: 09, Site # 2, Type: "Dedicated ShMC" 03

```

● 전원 관리 정보 표시

이 예에서는 명령과 출력의 예를 나타냅니다.

```

# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
  Version = 0
  Allowance for FRU Activation Readiness: 20 seconds
  FRU Activation and Power Description Count: 19
  Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

  Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

  Hw Address: 43, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

  Hw Address: 44, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

  Hw Address: 45, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Wattss
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

  Hw Address: 46, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

  Hw Address: 47, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

```

Hw Address: 48, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

Hw Address: 49, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

Hw Address: 4a, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

Hw Address: 4b, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

Hw Address: 4c, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

Hw Address: 4d, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

Hw Address: 4e, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

Hw Address: 44, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

Hw Address: 45, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

Hw Address: 08, FRU ID: 0xfe, Maximum FRU Power Capabilities: 24 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

Hw Address: 09, FRU ID: 0xfe, Maximum FRU Power Capabilities: 24 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

Hw Address: 20, FRU ID: 0xfe, Maximum FRU Power Capabilities: 100 Watts  
Shelf Manager Controlled Activation: Enabled  
Delay Before Next Power On: 0.0 seconds

#



● 배전 정보 표시

이 예에서는 선반의 배전 정보를 가져오는 명령과 출력을 나타냅니다.

```
# clia shelf pd
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Power Distribution Record (ID=0x11)
    Version = 0
    Feed count: 8
    Feed 00:
        Maximum External Available Current: 25.0 Amps
        Maximum Internal Current: 25.0 Amps
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 1012.500 Watts
        Currently Used Power: 30.000 Watts
        Feed-to-FRU Mapping entries count: 3
            FRU Addr: 49, FRU ID: 0xfe
            FRU Addr: 4b, FRU ID: 0xfe
            FRU Addr: 4d, FRU ID: 0xfe
    Feed 01:
        Maximum External Available Current: 25.0 Amps
        Maximum Internal Current: 25.0 Amps
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 1012.500 Watts
        Currently Used Power: 30.000 Watts
        Feed-to-FRU Mapping entries count: 3
            FRU Addr: 49, FRU ID: 0xfe
            FRU Addr: 4b, FRU ID: 0xfe
            FRU Addr: 4d, FRU ID: 0xfe
    Feed 02:
        Maximum External Available Current: 25.0 Amps
        Maximum Internal Current: 25.0 Amps
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 1012.500 Watts
        Currently Used Power: 40.000 Watts
        Feed-to-FRU Mapping entries count: 4
            FRU Addr: 41, FRU ID: 0xfe
            FRU Addr: 43, FRU ID: 0xfe
            FRU Addr: 45, FRU ID: 0xfe
            FRU Addr: 47, FRU ID: 0xfe
    Feed 03:
        Maximum External Available Current: 25.0 Amps
        Maximum Internal Current: 25.0 Amps
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 1012.500 Watts
        Currently Used Power: 40.000 Watts
```

```
Feed-to-FRU Mapping entries count: 4
    FRU Addr: 41, FRU ID: 0xfe
    FRU Addr: 43, FRU ID: 0xfe
    FRU Addr: 45, FRU ID: 0xfe
    FRU Addr: 47, FRU ID: 0xfe
Maximum External Available Current: 25.0 Amps
Maximum Internal Current: 25.0 Amps
Minimum Expected Operating Voltage: -40.5 Volts
Actual Power Available: 1012.500 Watts
Currently Used Power: 40.000 Watts
Feed-to-FRU Mapping entries count: 4
    FRU Addr: 42, FRU ID: 0xfe
    FRU Addr: 44, FRU ID: 0xfe
    FRU Addr: 46, FRU ID: 0xfe
    FRU Addr: 48, FRU ID: 0xfe
```

Feed 05:

```
Maximum External Available Current: 25.0 Amps
Maximum Internal Current: 25.0 Amps
Minimum Expected Operating Voltage: -40.5 Volts
Actual Power Available: 1012.500 Watts
Currently Used Power: 40.000 Watts
Feed-to-FRU Mapping entries count: 4
    FRU Addr: 42, FRU ID: 0xfe
    FRU Addr: 44, FRU ID: 0xfe
    FRU Addr: 46, FRU ID: 0xfe
    FRU Addr: 48, FRU ID: 0xfe
```

Feed 06:

```
Maximum External Available Current: 25.0 Amps
Maximum Internal Current: 25.0 Amps
Minimum Expected Operating Voltage: -40.5 Volts
Actual Power Available: 1012.500 Watts
Currently Used Power: 100.000 Watts
Feed-to-FRU Mapping entries count: 6
    FRU Addr: 08, FRU ID: 0xfe
    FRU Addr: 09, FRU ID: 0xfe
    FRU Addr: 20, FRU ID: 0xfe
    FRU Addr: 4a, FRU ID: 0xfe
    FRU Addr: 4c, FRU ID: 0xfe
    FRU Addr: 4e, FRU ID: 0xfe
```

Feed 07:

```
Maximum External Available Current: 25.0 Amps
Maximum Internal Current: 25.0 Amps
Minimum Expected Operating Voltage: -40.5 Volts
Actual Power Available: 1012.500 Watts
Currently Used Power: 100.000 Watts
```

```
Feed-to-FRU Mapping entries count: 6
FRU Addr: 08, FRU ID: 0xfe
FRU Addr: 09, FRU ID: 0xfe
FRU Addr: 20, FRU ID: 0xfe
FRU Addr: 4a, FRU ID: 0xfe
FRU Addr: 4c, FRU ID: 0xfe
FRU Addr: 4e, FRU ID: 0xfe
```

```
#
```

## 선반 관리자 다시 초기화

이 절에서는 U-Boot 환경 변수, 플래시 메모리의 파일 시스템, 선반 관리 카드의 로그인 암호 등을 다시 초기화하는 방법에 대해 설명합니다.

### U-Boot 환경 다시 초기화

U-Boot 환경 변수는 선반 관리 카드 EEPROM에 저장됩니다. U-Boot 환경 변수에 대한 출하 시 기본값을 복원하려면 먼저 EEPROM에 저장된 환경 변수를 지우고 선반 관리 카드를 재설정하거나 전원을 껐다가 다시 켜야 합니다.

#### ▼ U-Boot 환경 다시 초기화

1. **U-Boot** 프롬프트에서 다음 명령을 입력하여 **EEPROM**을 지웁니다.

```
ShMM # eeeprom write 80400000 0 1000

EEPROM @0x50 write: addr 80400000  off 0000  count 4096 ... done
ShMM #
```

2. 다음과 같이 선반 관리 카드를 다시 설정합니다.

```
ShMM # reset
U-Boot 1.1.2 (Nov 27 2005 - 19:17:09)

CPU: Aul550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 8000041
DRAM: 128 MB
Flash: 64 MB
*** Warning - bad CRC, using default environment

In:      serial
Out:     serial
Err:     serial
Net:     AulX00 ETHERNET
Hit any key to stop autoboot:  0

ShMM #
```

3. 다음 명령을 사용하여 이러한 환경 설정을 저장합니다.

```
ShMM # saveenv
```

## 파일 시스템 다시 초기화

파일 시스템은 플래시 메모리에 저장되며 출하 시 기본값으로 매우 쉽게 재설정할 수 있습니다. U-Boot에는 `flash_reset`이라는 환경 변수가 있습니다. 이 변수를 `y`로 설정한 다음 시스템을 부트하면 파일 시스템이 출하 시 기본값으로 다시 초기화됩니다.

```
ShMM # setenv flash_reset y
ShMM # boot
```

플래시 메모리를 다시 초기화한 다음 시스템을 시작하면 `flash_reset` 변수가 `n`으로 자동 설정됩니다. `bootcmd` 명령을 실행하면 Linux 커널이 부트됩니다. 이 프로세스 중에 파일 시스템이 다시 초기화됩니다. 다음과 같은 출력이 콘솔에 표시됩니다.

```
/etc/rc: Mounted /dev/pts
/etc/rc: Flash erase requested via U-BOOT var
/etc/rc: erasing mtdchar1 -> /etc
Erased 1024 Kibyte @ 0 -- 100% complete.
/etc/rc: erasing mtdchar0 -> /var
Erased 1536 Kibyte @ 0 -- 100% complete.
/etc/rc: Mounted /dev/mtdblock3 to /var
/etc/rc: /var/log mounted as FLASH disk
/etc/rc: Started syslogd and klogd
/etc/rc: /var/tmp mounted as RAM disk
/etc/rc: hostname demo
/etc/rc: /dev/mtdblock2 appears to be empty ... restoring from
factory /etc...
```

## 로그인 암호 재설정

선반 관리 카드에 대한 출하 시 기본 로그인은 루트의 사용자 ID이며 암호는 없습니다. 선반 관리자를 구성할 때 암호를 변경하는 것이 좋습니다. 새 암호를 잊은 경우 `password_reset` U-Boot 변수를 사용하여 암호를 출하 시 기본값으로 재설정할 수 있습니다. 이 변수를 `y`로 설정한 다음 시스템을 부트하면 루트 암호가 제거됩니다.

```
ShMM # setenv password_reset y
ShMM # boot
```

부트하는 동안 다음과 같은 출력이 콘솔에 표시됩니다.

```
/etc/rc: hostname demo
```

## 선반 관리 카드 재프로그래밍

선반 관리 카드 재프로그래밍에는 카드에 포함된 여러 펌웨어 이미지 재프로그래밍이 포함됩니다. 이미지 집합은 재프로그래밍 방법에 따라 개념적으로 다음과 같은 두 그룹으로 분류할 수 있습니다.

- U-Boot, Linux 커널 및 Linux 루트 파일 시스템(RFS) 이미지

이러한 이미지는 일반적으로 안정적인 업그레이드 절차에 따라 재프로그래밍됩니다. 또는 TFTP 서버에서 커널과 RFS를 로드하여 U-Boot 프롬프트에서 해당 커널과 RFS를 재프로그래밍할 수도 있습니다.

- CPLD(Complex Programmable Logic Device) 이미지

이 이미지는 선반 관리 카드 명령줄에서 특수 명령줄 유틸리티 `cp1dtool`을 사용하여 재프로그래밍합니다.

이러한 이미지에 대한 자세한 재프로그래밍 지침은 아래 내용을 참조하십시오.

## 안정적인 펌웨어 업그레이드 절차

Monterey Linux 운영 체제는 실행 중이고 작동 중인 선반 관리 카드에서 펌웨어 이미지를 안정적으로 업그레이드할 수 있는 절차를 제공합니다. 이 절차에서는 U-Boot 펌웨어, Linux 커널 및 Linux RFS(또는 이 세 가지 이미지의 임의 조합) 업그레이드를 지원합니다. 소프트웨어 업그레이드 시도가 실패할 경우(예: 잘못된 U-Boot 펌웨어 이미지를 설치했거나, 선반 관리 카드를 부팅할 수 없거나, 선반 관리자를 시작할 수 없는 경우) 안정적인 업그레이드 절차가 실행되어 자동으로 영구 플래시 메모리에 있는 펌웨어의 이전 버전으로 되돌립니다.

선반 관리 카드 플래시 메모리는 두 영역으로 분류됩니다. 안정적인 펌웨어 집합이 이 영역 중 하나에 설정되면 해당 영역이 영구 영역으로 지정됩니다. 새 펌웨어가 설치되면 임시 영역으로 지정된 다른 영역으로 이동됩니다. 임시 영역의 새 펌웨어 집합이 검증되면 해당 영역이 영구 영역으로 지정되고 향후의 업그레이드 주기로 인해 프로세스가 종료될 때까지 계속 사용됩니다.

안정적인 업그레이드 하드웨어 메커니즘을 사용하면 임시 플래시에 설치되는 내용에 관계 없이 선반 관리 카드는 완벽하게 작동하거나 정상적인 소프트웨어 복사본으로부터 부트할 수 있습니다. 이 소프트웨어 복사본은 업그레이드 세션에 오류가 있었는지 확인하고 적절한 교정 작업을 수행하여 영구 플래시의 안전한 소프트웨어 복사본으로 되돌립니다.

상위 수준에서 안정적인 업그레이드 하드웨어 메커니즘은 비휘발성 파일 `/var/upgrade/status`에 업그레이드 세션 상태를 기록하는 것을 기반으로 하는 소프트웨어 프로토콜에 의해 지원됩니다(85페이지의 "안정적인 업그레이드 절차 상태 파일" 참조). 소프트웨어 프로토콜은 특정 응용 프로그램에 필요한 사용자 정의 후크 스크립트에 정의된 작업을 포함하여 필요한 모든 작업이 성공적으로 완료된 후에 안정적인 업그레이드가 완료되도록 합니다.

Monterey Linux에서 구현되는 안정적인 업그레이드 절차는 선반 관리 카드에서 실행 중인 내장 응용 프로그램에 중립적입니다. 이 절차에서는 선반 관리 카드에서 실행 중인 특정 응용 프로그램이 안정적인 업그레이드의 적절한 지점에서 사용자 정의 작업을 수행할 수 있도록 충분한 "후크"를 제공합니다. 이 절의 나머지 부분에서는 이러한 후크를 사용하여 구현된 IPM Sentry 선반 관리자 펌웨어의 안정적인 업그레이드를 위한 규칙을 중심으로 설명합니다.

## 플래시 분할

선반 관리 카드는 MIPS 프로세서에서 실행 중인 소프트웨어의 제어에 따라 시스템 메모리 맵에서 플래시의 위쪽 부분과 아래쪽 부분을 스위치오버할 수 있는 하드웨어 메커니즘을 제공합니다. 이 기능은 플래시에 있는 소프트웨어 이미지에 대한 안정적인 업그레이드 절차의 지원으로 구현됩니다. 안정적인 소프트웨어 업그레이드 절차에서는 플래시 장치에서 플래시의 아래쪽 부분과 위쪽 부분에 두 개의 소프트웨어 복사본이 있는 것으로 가정합니다. 모든 선반 관리 카드는 분할된 상태로 제공됩니다. 즉, 플래시 장치는 선반 관리자 소프트웨어 복사본을 하나씩 보관하는 동일한 두 개 부분으로 분류됩니다.

U-Boot 환경 변수 `reliable_upgrade`(16페이지의 "U-Boot 환경 변수" 참조)는 Linux 계층에서 안정적인 업그레이드 절차의 활성화 여부를 결정하는 데 사용됩니다. 이 변수를 `y`로 설정하여 `bootargs` 커널 매개 변수 문자열로 Linux 커널에 전달해야 합니다. Linux 플래시 MTD 계층은 플래시 분할 영역을 초기화할 때 `reliable_upgrade` 매개 변수를 검사한 다음 매개 변수 값과 선반 관리 카드에 설치된 플래시 장치의 크기에 따라 플래시 장치를 적절하게 분할합니다.

이 절에서는 선반 관리 카드가 개별적인 두 개의 플래시 영역을 포함하여 안정적인 업그레이드를 지원하도록 구성되어 있다고 가정합니다. 표 3-3에서는 선반 관리 카드에서 이 구성(`reliable_upgrade=y`)으로 유지 관리되는 플래시 분할 영역에 대해 요약합니다.

표 3-3 플래시 분할 영역(16MB reliable\_upgrade=y)

플래시의 오프셋(MB)	크기(MB)	장치 노드	마운트(시작 시)	내용
0	0.5	/dev/mtdchar10, /dev/mtdblock10	/var/upgrade	/var/upgrade JFFS2 파일 시스템의 후반부
0.5 +(FLASH_SIZE - 16)/2	1.5	/dev/mtdchar5, /dev/mtdblock5	마운트 안 됨	다른 /var JFFS2 파일 시스템
FLASH_SIZE/2 - 62	1	/dev/mtdchar6, /dev/mtdblock6	마운트 안 됨	다른 /etc JFFS2 파일 시스템
FLASH_SIZE/2 - 53	1	/dev/mtdchar7	마운트 안 됨	다른 Linux 커널 이미지
FLASH_SIZE/2 - 44	0.25	/dev/mtdchar8	마운트 안 됨	다른 U-Boot 펌웨어 이미지
FLASH_SIZE/2 - 3.754.25	3.75	/dev/mtdchar9	마운트 안 됨	다른 Linux 루트 파일 시스템(RFS) 이미지
FLASH_SIZE/28	0.5	/dev/mtdchar10, /dev/mtdblock10	/var/upgrade	/var/upgrade JFFS2 파일 시스템의 전반부
FLASH_SIZE - 7.58.5	1	/dev/mtdchar0, /dev/mtdblock0	/var	/var JFFS2 파일 시스템
FLASH_SIZE - 610	1	/dev/mtdchar1, /dev/mtdblock1	/etc	/etc JFFS2 파일 시스템
FLASH_SIZE - 5.11	1	/dev/mtdchar2	마운트 안 됨	Linux 커널 이미지
FLASH_SIZE - 412	0.25	/dev/mtdchar3	마운트 안 됨	U-Boot 펌웨어 이미지
FLASH_SIZE - 3.712.255	3.75	/dev/mtdchar4	마운트 안 됨	Linux 루트 파일 시스템 (RFS) 이미지

## /var/upgrade 파일 시스템

82페이지의 "안정적인 펌웨어 업그레이드 절차"에서 설명한 것처럼 reliable\_upgrade가 y이면 Monterey Linux는 1MB 분할 영역을 /var/upgrade에 JFFS2 파일 시스템으로 마운트합니다. 이 파일 시스템은 안정적인 업그레이드 절차 상태 파일을 호스트하는 데 사용됩니다(85페이지의 "안정적인 업그레이드 절차 상태 파일" 참조).

/var/upgrade JFFS2 분할 영역은 플래시 장치의 위쪽 부분과 아래쪽 부분에 각각 하나씩 연속하지 않은 두 개의 플래시 블록(각각 0.5MB)으로 구성되어 있습니다. Monterey Linux는 /var/upgrade를 이 방식으로 구현하도록 Linux MTD 및 JFFS2 계층 기능을 활용하여 비연속 플래시 섹터의 파일 시스템을 지원합니다.



안정적인 업그레이드 절차를 위해 /var/upgrade를 사용 가능하게 해주는 JFFS2 파일 시스템의 다른 기능은 JFFS2 내부 구조가 플래시 섹터 번호 또는 플래시의 절대 오프셋을 기반으로 종속성(예: 연결된 목록)을 생성하지 않는다는 것입니다. 대신 분할 영역에 파일 시스템을 마운트할 때 JFFS2는 해당 분할 영역을 구성하는 모든 플래시 섹터를 검사하고 파일 시스템의 논리 내용을 내부 RAM 표현으로 다시 만듭니다. 이 기능을 사용하면 플래시의 어느 부분에서 ShMM가 부트되는지에 관계없이 Linux는 /var/upgrade를 JFFS2 파일 시스템으로 마운트하여 이전 파일 시스템 내용을 사용할 수 있습니다.

## 안정적인 업그레이드 절차 상태 파일

소프트웨어 안정적 업그레이드 절차에서는 ShMM을 부팅할 때 사용된 플래시에 관계없이 Linux에 마운트되는 전용 파일 시스템(/var/upgrade)에 있는 /var/upgrade/status 파일에 최신 업그레이드 절차 세션의 상태를 유지 관리합니다. 이 파일에는 현재 진행 중이거나 최근에 완료된 업그레이드 절차 세션의 상태가 포함되어 있습니다(있는 경우).

/var/upgrade/status는 업그레이드 절차의 특정 단계 상태를 설명하는 하나 이상의 레코드(새 행으로 종료)가 포함된 ASCII 형식 파일입니다. 레코드 행의 형식은 다음과 같습니다.

<step>: <status>

여기서 *step*은 1에서 14까지의 정수(14단계는 완료된 업그레이드 세션에 해당)이고 *status*는 업그레이드 절차 세션의 현재 단계에 대한 상태를 설명하는 읽기 가능한 형태의 문자열입니다.

상태 파일은 안정적인 업그레이드 유틸리티(85페이지의 "안정적인 업그레이드 유틸리티" 참조)에서 사용하며 안정적인 업그레이드 절차 하드웨어 메커니즘에 따라 소프트웨어 프로토콜을 유지 관리하여 업그레이드 절차의 상태를 안정적으로 결정하고 적절하게 처리합니다.

## 안정적인 업그레이드 유틸리티

특수 사용자 공간인 안정적인 업그레이드 유틸리티를 사용하면 안정적인 업그레이드 절차를 수행할 수 있을 뿐만 아니라 최신 업그레이드 상태를 확인할 수 있습니다.

이 유틸리티는 슈퍼유저(루트) 계정에서만 호출할 수 있습니다. 슈퍼유저가 아닌 계정에서 이 유틸리티를 실행하려는 시도는 거부됩니다.

첫 번째 실행 단계에서 이 유틸리티는 U-Boot가 커널 매개 변수 문자열로 Linux 커널에 전달한 `reliable_upgrade` U-Boot 환경 변수(83페이지의 "플래시 분할" 참조)가 `y`로 설정되어 있는지 확인합니다. 이 확인이 실패하면 유틸리티가 즉시 종료되고 해당 오류 코드가 표시됩니다.

-s, -c 또는 -f 옵션 중 하나를 사용하여 호출한 경우 이 유틸리티를 안정적인 업그레이드 절차를 실행하는 데 사용할 수 있습니다. 업그레이드 절차를 수행하는 중에 유틸리티는 업그레이드 절차의 단계를 진행하면서 각 단계에서 수행되는 작업의 상태를 /var/upgrade/status에 기록합니다. 오류가 감지될 경우 업그레이드 절차 실패를 나타내는 레코드를 /var/upgrade/status에 추가하고 해당 오류 코드를 표시하여 안정적인 업그레이드 절차를 종료합니다.

유틸리티는 stdout에 정보 메시지를 인쇄합니다. -v 지정자를 지원하는 옵션에 제공하면 정보 메시지의 상세 표시 수준이 향상됩니다. 유틸리티는 stderr에 오류 메시지를 인쇄합니다.

유틸리티의 구문은 다음과 같습니다.

- `rupgrade_tool -s [--dst=src]... [--proto=protocol] [-d] [--hook=args] [-v]`
- `rupgrade_tool -c [-v]`
- `rupgrade_tool -f [--hook=args] [-v]`
- `rupgrade_tool -w [-f]`
- `rupgrade_tool -S [-v]`
- `rupgrade_tool -u`
- `rupgrade_tool -h`

여기서 매개 변수는 다음과 같이 정의됩니다.

`-s [--dst=src]... [--proto=protocol] [--hook=args] [-v]`

안정적인 업그레이드 절차를 시작합니다. 선반 관리자 지원과 함께 제공되는 이 단계에서는 다음과 같은 작업을 수행합니다.

- 로컬 또는 네트워크를 통해 복사할 이미지 가져오기
- 임시 플래시에 이미지 복사
- ShMM에서 실행 중인 선반 관리자 인스턴스 종료(있는 경우)
- 임시 플래시에 비휘발성 데이터 복사
- 임시 플래시에서 부팅하도록 선반 관리 카드 재설정

마지막 단계로 인해 `rupgrade_tool -s`를 호출하면 결과가 반환되지 않고 대신 선반 관리 카드가 재설정됩니다. `rupgrade_tool -s`에서 결과를 반환하는 경우 안정적인 업그레이드 절차가 실패하여 임시 플래시에서 부팅하도록 선반 관리 카드를 재설정하기 이전에 종료된 것입니다.

유틸리티가 업그레이드 절차의 첫 번째 단계를 시작하기 전에 /var/upgrade/status 파일이 제거되었습니다(85페이지의 "안정적인 업그레이드 절차 상태 파일" 참조). 즉 `rupgrade_tool -s`를 호출하자마자 이전 업그레이드 절차 세션(있는 경우)의 상태가 손실되고 새 업그레이드 절차 세션 상태로 덮어썼습니다.

`rupgrade_tool -s`에 대한 호출에 `--dst=src` 지정자가 여러 개 있을 수 있습니다. 각 지정자는 설치할 업그레이드 이미지 파일의 이름을 정의합니다. 여기서 선반 관리 카드의 플래시에 파일이 설치됩니다.

*dst*는 새로 설치된 업그레이드 이미지의 대상을 정의하며 다음과 같습니다.

- *u* - 임시 U-Boot 펌웨어 이미지 분할 영역(/dev/mtdchar3)에서 U-Boot 이미지를 업그레이드합니다.
- *k* - 임시 Linux 커널 이미지 분할 영역(/dev/mtdchar2)에서 Linux 커널 이미지를 업그레이드합니다.
- *r* - 임시 루트 파일 시스템 이미지 분할 영역(/dev/mtdchar4)에서 루트 파일 시스템 이미지를 업그레이드합니다.

*src*는 *dst*에 지정된 임시 플래시 분할 영역에 복사할 업그레이드 이미지 파일을 지정합니다.

이미지 업그레이드 작업은 다음과 같이 수행됩니다. 지정된 각 *src* 이미지에 대해 지정된 복사 프로토콜을 사용하여 선반 관리 카드에 이미지를 복사합니다. *-d* 지정자를 입력하지 않으면 이미지가 선반 관리 카드의 RAM 파일 시스템에 먼저 복사(/tmp 디렉토리에 복사)된 다음 플래시로 이동됩니다. 즉 플래시의 대상 분할 영역에 복사된 다음 RAM 파일 시스템에서 제거됩니다. *rupgrade\_tool -s*를 호출할 때 *-d* 지정자를 제공하면 중간에 /tmp 디렉토리에 복사되지 않고 플래시의 대상에 직접 복사됩니다. 이 지정자는 선반 관리 카드에 RAM 파일 시스템에 중간 복사를 실행하기 위한 런타임 메모리가 부족한 경우에 사용됩니다.

*-d* 지정자를 제공하지 않을 경우 안정적인 업그레이드 절차에서는 /tmp 디렉토리에 이미지를 복사한 후 이미지의 유효성을 검사하기 위한 특수 스크립트를 호출합니다. *-d* 지정자가 있으면 이러한 유효성 검사가 수행되지 않습니다.

선반 관리자와 함께 현재 제공된 /etc/upgrade/step4vshn 스크립트는 특정 이미지 유효성 검사 단계를 수행하지 않지만 *rupgrade\_tool*에 대한 현재 호출에 이미지가 제공되지 않은 플래시 분할 영역을 채웁니다(부분 업그레이드 시나리오와 동일). 이러한 분할 영역은 현재 영구 플래시에서 임시 플래시로 복사됩니다. 예를 들어 현재 부분 업그레이드에서 새 RFS 이미지만 제공하는 경우 스크립트는 U-Boot 및 커널 분할 영역을 영구 플래시에서 임시 플래시로 복사합니다.

유틸리티는 첫 번째 이미지가 대상에 설치되면 바로 두 번째 이미지를 계속합니다(있는 경우). 이런 방법으로 제공된 이미지 파일을 플래시에 모두 설치합니다. 이미지 설치에 실패하면 업그레이드 절차가 바로 종료됩니다(실패한 이미지를 건너뛰고 다음 이미지를 계속하는 경우와 비교).

이 방법을 사용하면 선반 관리 카드 펌웨어의 세 부분(U-Boot, 커널 및 RFS 이미지)을 개별적으로 업그레이드할 수 있습니다. 그러나 명시적으로 업데이트되지 않은 부분은 영구 플래시에서 복사됩니다.

다음 업그레이드 방법 중 하나를 사용하는 것이 좋습니다.

- 세 분할 영역을 모두 명시적으로 업그레이드합니다.
- 세 개 미만의 분할 영역을 명시적으로 업그레이드하는 경우 *-d* 지정자를 생략합니다. 그러면 위에서 설명한 특수 스크립트가 실행되어 세 분할 영역을 모두 포함하여 업그레이드가 효과적으로 완료되는지 자동으로 확인합니다.

*protocol*은 지정된 각 *src* 파일을 선반 관리자로 가져오는 데 사용되는 파일 복사 프로토콜을 지정하며 다음과 같습니다.

- `no` - 복사가 수행되지 않습니다. 이 프로토콜은 안정적인 업그레이드 절차를 시작하기 이전에 지정된 모든 `src` 파일이 `/tmp` 디렉토리에 저장된 것으로 가정합니다. 이 프로토콜은 `-d` 옵션과 함께 사용할 수 없습니다.
- `cp:dir` - 간단한 복사를 수행합니다. 이 프로토콜은 `cp` 명령을 사용하여 선반 관리자 로컬 파일 시스템에 지정된 디렉토리에서 지정된 모든 `src` 파일을 복사하는 것으로 가정합니다. 이 프로토콜은 NFS 마운트 파일 시스템 또는 JFFS2 파일 시스템에서 업그레이드 이미지를 설치하는 경우에 유용합니다.
- `ftp:server:dir:user[:pwd]` - 원격 FTP 서버에서 복사합니다. 이 프로토콜은 호스트 이름 또는 서버별 IP 주소로 지정한 FTP 서버 호스트에서 지정된 모든 `src` 파일을 선반 관리 카드로 복사하는 것으로 가정합니다. 모든 이미지는 원격 FTP 서버의 `dir`에 지정된 디렉토리에 있어야 합니다. 사용자 매개 변수에 지정된 계정 및 선택적 `pwd` 매개 변수에 지정된 암호를 사용하여 FTP 연결을 설정합니다. `pwd`를 입력하지 않으면 암호를 묻는 메시지가 표시됩니다.

이미지를 선반 관리자에 복사하는 데 실패하면 업그레이드 절차가 종료됩니다(실패한 이미지를 건너뛰고 다음 이미지를 계속하는 경우와 비교).

`-s` 옵션에 의해 업그레이드되는 각 임시 플래시 분할 영역의 경우 이미지의 유효성을 검사한 후 `src` 이미지가 플래시로 이동되기 직전에 업그레이드할 분할 영역에 쓰기 권한이 부여됩니다. 전체 이미지가 플래시로 이동되면 분할 영역에서 쓰기 권한이 바로 제거됩니다. 선반 관리자를 부팅할 때는 U-Boot, Linux 커널 및 루트 파일 시스템 이미지를 포함하는 모든 분할 영역에 읽기 전용 권한만 부여되므로 응용 프로그램에서 중요 부트 이미지를 실수로 삭제하는 경우는 없습니다.

지정된 모든 이미지가 플래시 메모리의 해당 대상에 설치되면 유틸리티는 후크 스크립트를 호출하여 업그레이드 이미지가 플래시에 이미 설치된 지점에서 응용 프로그램에 필요한 사용자 정의 작업을 수행할 수 있게 해줍니다. 아직은 업그레이드 절차에서 ShMM의 업그레이드 워치독 타이머(WDT)를 활성화하여 안정적인 업그레이드 절차의 하드웨어 메커니즘을 시작하지 않았습니다.

후크 스크립트인 `/etc/upgrade/step4hshm`이 선반 관리자에 제공됩니다. 이 스크립트는 다음과 같은 작업을 수행합니다.

- 선반을 다시 시작하지 않고 백업 ShMM으로 스위치오버하여 선반 관리자를 종료합니다. 그러면 ATCA 워치독 타이머가 중지됩니다.
- 임시 `/etc` 및 `/var` 플래시 분할 영역을 마운트하고 이 분할 영역에 있는 모든 파일을 지웁니다.
- 선택적으로 `/etc` 디렉토리의 현재 내용을 임시 `/etc` 플래시 분할 영역에 복사합니다.
- 선택적으로 현재 비휘발성 선반 관리자 정보를 `/var/nvdata` 디렉토리에서 임시 `/var` 파일 시스템으로 복사하거나, 전체 `/var` 디렉토리를 임시 `/var` 플래시 분할 영역에 복사합니다.
- 다음 부트 때까지 부트 지연을 0으로 임시 설정합니다. 그러면 다음 부트 시간이 최소화되고 안정적인 업그레이드 워치독 타이머가 조기에 만료되지 않습니다.

이 스크립트는 하위 셸로 호출되며 *args*를 사용하거나 *args*를 사용하지 않고(빈 문자열) 지정한 문자열인 단일 매개 변수를 사용합니다. 지정한 매개 변수는 영구 플래시 분할 영역에서 임시 플래시 분할 영역으로 비휘발성 정보 복사 등과 같은 작업을 결정하는 스크립트의 작업 모드를 정의합니다. 스크립트는 다음과 같은 *args* 값을 가지며 해당 작업을 수행합니다.

- 제공된 매개 변수 없음 - 임시 /etc 및 임시 /var 디렉토리를 모두 지운 다음 선반 관리자의 비휘발성 정보를 /var/nvdata 디렉토리에서 임시 /var 분할 영역으로 복사합니다. 이는 기본 작업 모드입니다. 이 경우 비휘발성 데이터는 유지되지만 선반 관리자 구성 파일은 새 RFS 이미지에서 가져옵니다.
- erase - 임시 /etc 및 임시 /var 디렉토리를 모두 지우고 다음 부트 중에 RFS 기본값으로부터 복원합니다. 현재 선반 관리자의 비휘발성 데이터와 구성은 유지되지 않습니다.
- etc\_copy - 임시 /etc 및 임시 /var 디렉토리를 모두 지운 다음 /etc의 내용과 비휘발성 정보를 /var/nvdata 디렉토리에서 임시 플래시 분할 영역으로 복사합니다. 이 경우 비휘발성 데이터와 선반 관리자 구성 파일이 모두 유지됩니다.
- copy - 임시 /etc 및 임시 /var 디렉토리를 모두 지운 다음 /etc 및 /var 디렉토리의 전체 내용을 임시 분할 영역에 복사합니다. 이 경우 /var/bin에 배치된 실행 파일과 구성이 모두 복사되고 실행 파일을 RFS 이미지에서 가져온 이름으로 대체합니다. 이 작업 모드는 /var/bin 디렉토리에 업그레이드 이후에 유지되어야 하는 일부 특수 실행 파일(예: 특수 버전 선반 관리자 또는 기타 유틸리티)이 있는 경우에 유용합니다.

이 스크립트는 성공할 경우 0을 반환하고 실패할 경우 0이 아닌 값을 반환합니다. 0이 아닌 값이 반환되면 업그레이드 절차가 종료됩니다.

이 유틸리티는 12.8초 시간 초과 기간을 적용하여 업그레이드 WDT를 시작합니다. 이 시간 초과 기간은 재설정 후에 부트되어 중간에 업그레이드 WDT를 확인하지 않고 `rupgrade_tool -c`(활성 상태에서 업그레이드 WDT 확인)를 호출할 수 있는 지점까지 진행되는 모든 소프트웨어에 대해 충분한 시간입니다. 유틸리티는 ShMM을 재설정하기 바로 전에 업그레이드 WDT를 확인합니다.

#### ■ -c [-v]

임시 플래시에서 ShMM을 부트한 후에 안정적인 업그레이드 절차를 계속합니다. `rupgrade_tool -c`는 /etc/rc 스크립트에서 호출됩니다. 아래에 설명한 것처럼 `rupgrade_tool -c`에 의해 검색되는 특정 상황은 업그레이드 절차의 실패를 나타내며 ShMM 소프트웨어 재설정 등과 같은 교정 작업이 필요합니다. 즉 `rupgrade_tool -c`를 호출하면 결과가 반환되는 대신 ShMM이 재설정될 수 있습니다. 재설정이 수행되면 ShMM을 영구 플래시에 설치된 소프트웨어로 되돌립니다.

업그레이드 WDT가 `rupgrade -c`를 호출하기 이전의 어느 단계에서 활성화되어 시작된 경우 ShMM이 영구 플래시의 소프트웨어로 이미 되돌려진 경우입니다. 그럴 경우 유틸리티는 업그레이드 WDT를 비활성화하고 영구 플래시를 다시 사용하고 업그레이드 절차를 종료합니다.

업그레이드 WDT가 활성 상태이지만 아직 시작되지 않았으면 ShMM이 임시 플래시에서 이 지점까지 성공적으로 부트한 것입니다. 이 유틸리티는 업그레이드 WDT를 확인하고, 진행 중인 업그레이드 세션이 없음을 나타내는 반환 코드 0과 함께 종료됩니다.

업그레이드 WDT가 활성화 상태가 아니지만 /var/upgrade/status 파일의 내용이 업그레이드 절차가 진행 중임을 나타내는 경우 업그레이드 절차의 어느 단계에서 전원을 껐다가 켜서 ShMM이 다시 부트된 것입니다. 이 경우 유틸리티는 업그레이드 WDT가 활성화 상태이고 시작된 경우와 동일한 교정 작업을 수행합니다.

업그레이드 WDT가 활성화 상태가 아니고 /var/upgrade/status가 없거나 업그레이드 절차가 완료된(성공 또는 실패) 것으로 표시하는 경우 이 유틸리티는 진행 중인 업그레이드 절차가 없음을 나타내는 반환 값 1과 함께 종료됩니다.

■ -f [--hook=args] [-v]

업그레이드 절차를 완료합니다. `rupgrade_tool -f`는 선반 관리자가 초기화를 성공적으로 완료한 이후에 선반 관리자 내부에서 호출됩니다. 선반 관리자가 자동으로 시작되지 않으면 /etc/rc 스크립트를 끝낼 때 호출이 수행됩니다.

`rupgrade_tool -f`가 호출되면 바로 업그레이드 WDT를 확인한 다음 새 영구 플래시를 설정하고 업그레이드 WDT를 비활성화합니다.

완료하기 전에 유틸리티는 업그레이드 절차의 성공적인 완료를 나타내는 레코드로 /var/upgrade/status를 업데이트한 다음 0 값과 함께 종료됩니다.

■ -w [-f]

최신 업그레이드 절차의 현재 상태를 인쇄합니다. 기본적으로 이 옵션은 /var/upgrade/status 파일의 내용을 stdout에 덤프합니다.

`rupgrade_tool -w`는 업그레이드 절차가 성공적으로 완료된 경우 0을 값으로 반환하고 업그레이드 절차가 실패한 경우 1을 반환합니다. /var/upgrade/status가 없는 경우에는 오류 코드를 반환합니다.

-f 지정자를 제공하면 `rupgrade_tool -w`는 종료하기 전에 /var/upgrade/status 파일을 제거합니다.

■ -S [-v]

업그레이드 WDT를 확인합니다. `rupgrade_tool -S`는 새로 설치된 소프트웨어의 온전성 검사에 사용할 셸 수준 인터페이스의 역할을 합니다.

`rupgrade_tool -S`는 0 값을 반환합니다.

■ -u

성공적인 업그레이드 세션을 실행 취소하여 이전 영구 플래시 장치로 되돌립니다.

`rupgrade_tool -u`를 실행하면 ShMM이 재부트됩니다.

■ -h

stdout에 대한 도움말을 인쇄합니다.

## 안정적인 업그레이드 유틸리티 사용 시나리오

안정적인 업그레이드 유틸리티를 다음 순서로 사용하여 ShMM을 업그레이드합니다.

1. 사용자가 `rupgrade_tool -s`를 호출하여 업그레이드 절차를 시작합니다. ShMM 직렬 콘솔에서 로컬로 호출하거나 `telnet`, `rsh`, `ssh` 또는 해당 유틸리티를 사용하여 네트워크를 통해 원격으로 호출할 수 있습니다.
2. `rupgrade_tool -s`가 ShMM을 다시 부트할 때까지 기다립니다. 직렬 콘솔에 로컬로 연결한 경우 U-Boot 펌웨어 및 직렬 콘솔에 연결된 Linux에서 출력한 메시지에 재부트 상태가 표시됩니다. ShMM에 원격으로 연결한 경우에는 재부트 상태가 명확하게 표시되지 않습니다. 예를 들어 텔넷 연결은 선반 관리 카드가 재부트되면 시간 초과됩니다. 사용자는 업그레이드 절차가 성공적으로 수행된 것으로 가정하거나 업그레이드 세션이 완료되는 데 필요한 특정 시간 동안 대기한 다음 `rupgrade_tool -w`를 호출(위에서 설명한 원격 셸 도구를 사용하여 원격으로)하여 업그레이드 세션의 상태를 확인할 수 있습니다. 대기 시간은 업그레이드 이미지의 크기와 이미지를 선반 관리 카드로 가져오는 데 사용되는 복사 프로토콜 및 이미지 유효성 검사 스크립트에서 수행하는 작업에 따라 다릅니다.
3. 선반 관리 카드에서 시작 스크립트 `/etc/rc`는 `rupgrade_tool -c`를 무조건적으로 호출합니다. 진행 중인 업그레이드가 없음을 나타내는 1을 반환하거나 업그레이드 세션이 실패했음을 나타내는 오류 코드 값을 반환하면 시작 스크립트는 정상 모드 부트 시퀀스를 계속 진행합니다. 그러나 진행 중인 업그레이드 세션이 있음을 나타내는 값 0이 반환되는 경우 시작 스크립트는 새로 설치된 소프트웨어의 온전성 검사를 계속 진행합니다. 이 때 유효성 검사가 업그레이드 WDT 시간 초과 기간보다 더 오래 걸릴 경우 작업 중간에 `rupgrade_tool -s`를 호출하여 업그레이드 WDT를 확인하고 선반 관리자를 시작하여 최종 유효성 검사를 수행합니다. 위치독 타이머 간격은 12.8초로 설정되므로 `/etc/rc` 스크립트에서 `rupgrade_tool -c`를 호출한 이후부터 WDT를 확인할 때까지, 그리고 WDT를 확인한 이후부터 선반 관리자를 시작할 때까지의 처리 시간은 각각 12.8초를 초과하지 않아야 합니다.
4. 초기화 중 선반 관리자는 피어 선반 관리자와의 네트워크 연결을 설정하기 전에 업그레이드 WDT를 다시 한 번 확인합니다. 네트워크 연결을 설정하는 데 최대 6초 정도 걸릴 수 있습니다. 이 작업이 끝난 후, 그리고 초기화를 성공적으로 완료(새 구성의 유효성 검사 완료)한 후 선반 관리자는 `rupgrade_tool -f`를 호출하여 업그레이드 절차를 완료합니다.
5. 선택적으로 `rupgrade_tool -w`를 호출하여 업그레이드 세션의 상태를 확인할 수 있습니다. 위에서 설명한 것처럼 이 옵션은 로컬 업그레이드의 경우처럼 직렬 콘솔에 출력되는 메시지를 통해 업그레이드 상태를 직접 확인할 수 없는 원격 업그레이드를 수행할 때 특히 유용합니다.

안정적인 업그레이드를 완료한 후 어떤 이유로 새 이미지를 사용할 수 없는 경우 원본 이미지로 되돌릴 수 있습니다. 그렇게 하려면 `rupgrade_tool -u`를 호출합니다.

필요한 경우 원격 네트워크 호스트에서 실행하도록 설계된 간단한 스크립트를 개발하여 위에서 설명한 시퀀스를 쉽게 자동화할 수 있습니다. 또는 운영자가 직렬 콘솔에서 로컬로 또는 네트워크를 통해 원격으로 안정적인 업그레이드를 직접 수행할 수도 있습니다.

## 안정적인 업그레이드 예

### 예 1:

이 예에서는 /etc 및 /var/nvdata 비휘발성 디렉토리를 임시 플래시에 복사하여 세 구성 요소(U-Boot, 커널, RFS 이미지) 모두에 대해 안정적인 업그레이드를 수행하는 방법을 나타냅니다. 모든 이미지는 로컬 /tmp에서 가져옵니다(즉 명시적이지 않은 방식으로 이미 복사되어 있음). U-boot 이미지는 /tmp/u-boot.bin에서 가져오고, 커널 이미지는 /tmp/sentry.kernel에서 가져오고, RFS 이미지는 /tmp/sentry.rfs에서 가져옵니다. 업그레이드 절차는 직렬 콘솔에서 시작됩니다. 콘솔 로그에 분산되어 있는 설명을 통해 업그레이드 절차 단계에 대한 추가 배경 정보를 알 수 있습니다.

먼저 명령 프롬프트에서 `rupgrade_tool`을 시작합니다. 매개 변수는 선반 관리자의 비휘발성 데이터와 구성 파일을 유지한 상태로 세 개 플래시 이미지를 모두 업데이트한다는 것을 나타냅니다.

```
# rupgrade_tool -s --k=sentry.kernel --r=sentry.rfs
--u=u-boot.bin --hook=etc_copy -v
rupgrade_tool: PLB is 5
rupgrade_tool: EEPROM page saved
rupgrade_tool: persistent flash is 0
rupgrade_tool: provisional flash is 1
rupgrade_tool: copying image(s)
```

업그레이드 유틸리티는 유효성 검사 스크립트를 호출하여 현재 제공된 /tmp에서 이미지를 검사하려고 시도합니다. 지정된 파일 지정자가 /tmp에 없는 경우 유틸리티가 중지되고 다음과 같은 메시지가 표시됩니다.

```
rupgrade_tool: cannot open /tmp/u-boot.bin for reading.
rupgrade_tool: failed to copy images to flash
```

유틸리티는 임시 플래시의 지정된 대상에 이미지를 계속 복사합니다.

```
rupgrade_tool: invoking scripts (step4v*) [--u=u-boot.bin --k=
sentry.kernel --r=sentry.rfs --hook=etc_copy]
rupgrade_tool: copying u-boot.bin from /tmp to /dev/mtdchar8 using
'cp' protocol
rupgrade_tool: copying sentry.kernel from /tmp to /dev/mtdchar7
using 'cp' protocol
rupgrade_tool: copying sentry.rfs from /tmp to /dev/mtdchar9 using
'cp' protocol
rupgrade_tool: invoking scripts (step4h*) [etc_copy]
```



이 지점에서 step4hshm 후크 스크립트가 호출되어 선반 관리자를 중지시키고 비휘발성 정보를 임시 플래시에 복사합니다.

```
/etc/upgrade/step4hshm: Stopping Shelf Manager...
/etc/upgrade/step4hshm: Erasing /var and /etc, copying
/var/nvdata...
/etc/upgrade/step4hshm: Operation: copy /etc and /var/nvdata.
/etc/upgrade/step4hshm: Copying completed.
rupgrade_tool: image(s) copy OK
rupgrade_tool: watchdog started
rupgrade_tool: selected provisional flash
rupgrade_tool: reboot
Restarting system.
```

안정적인 업그레이드 절차에서 ShMM를 재설정합니다. 그러면 U-boot가 임시 플래시에서 시작됩니다.

```
* Resetting Integrated Peripherals

U-Boot 1.1.2 (May 12 2005 - 21:27:13)

CPU: Aul550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 8000044
DRAM: 128 MB
Flash: 64 MB
In: serial
Out: serial
Err: serial
Net: AulX00 ETHERNET
Hit any key to stop autoboot: 0
## Booting image at bfb00000 ...
   Image Name:   MIPS Linux-2.4.26
   Created:      2005-06-24 13:29:50 UTC
   Image Type:   MIPS Linux Kernel Image (gzip compressed)
   Data Size:    844843 Bytes = 825 kB
   Load Address: 80100000
   Entry Point:  802bc040
   Verifying Checksum ... OK
   Uncompressing Kernel Image ... OK
## Loading Ramdisk Image at bfc40000 ...
   Image Name:   sentry RFS Ramdisk Image
   Created:      2005-04-22 9:10:41 UTC
   Image Type:   MIPS Linux RAMDisk Image (gzip compressed)
   Data Size:    2465924 Bytes = 2.4 MB
   Load Address: 00000000
```

```
Entry Point: 00000000
Verifying Checksum ... OK
```

```
Starting kernel ...
```

```
init started: BusyBox v0.60.5 (2005.06.15-14:45+0000) multi-call
binary
/etc/rc: Mounted /proc
/etc/rc: Mounting filesystems...
/etc/rc: Mounted /dev/pts
/etc/rc: Mounted /dev/mtdblock0 to /var
/etc/rc: Mounted /dev/mtdblock10 to /var/upgrade
```

이 지점에서 rc 스크립트가 `rupgrade_tool -c`를 호출하여 안정적인 업그레이드가 진행 중인지 확인합니다. 이 도구는 0을 반환하여 업그레이드가 진행 중임을 확인합니다. 결과가 주어지면 rc 스크립트는 시작 프로세스를 계속합니다.

```
/etc/rc: Checking the reliable upgrade watchdog timer
/etc/rc: Mounted ram disk to /var/log
/etc/rc: Started syslogd and klogd
/etc/rc: Mounted ram disk to /var/tmp
/etc/rc: Setting hostname shmm+193
```

안정적인 업그레이드가 진행 중이므로 rc 스크립트에서 워치독 타이머를 한 번 더 확인합니다.

```
/etc/rc: Strobing the reliable upgrade watchdog timer
/etc/rc: Mounted /dev/mtdblock1 to /etc
/etc/rc: Calling /etc/rc.carrier3
Board Hardware Address: 0xFE
/etc/netconfig: /etc/hosts has valid 192.168.1.193 entry
/etc/netconfig: Updating /etc/profile.sentry with IP settings
/etc/netconfig: ifconfig eth0 192.168.1.193
/etc/netconfig: ifconfig eth1 192.168.0.193
/etc/netconfig: route add default gw 192.168.1.253
/etc/netconfig: Starting inetd...
/etc/rc.carrier3: Starting up IPMBs ...
/etc/rc.carrier3: Updating /etc/profile.sentry with specific
settings
/etc/rc.carrier3: Starting snmpd...
/etc/rc.carrier3: Starting httpd...
/etc/rc.carrier3: Starting Shelf Manager ...
<I> 02:48:08.463 [171] IPM Sentry Shelf Manager ver. 2.0.0. Built
on Jun 27 2005 14:48:57
<*> 02:48:08.469 [171] Limits: code=(400000:506f0), end_data=
10062000, start_stack=7fff7e30, esp=7fff78a0, eip=2ab0d2e4
```

```
<*> 02:48:08.469 [171] Stack limits: curr=1ff000, max=7fffffff
<*> 02:48:08.470 [171] Data limits: curr=7fffffff, max=7fffffff
<*> 02:48:08.900 [171] *** Lock log print buffer at 1003b7f0 ***
<*> 02:48:08.900 [171] *** Pthread lock log print buffer at
1003f820 ***
```

선반 관리자가 `rupgrade_tool -f`를 실행하여 안정적인 업그레이드를 시작하고 완료합니다.

```
eth0: link up
eth1: link up
eth1: going to full duplex

shmm+193 login:root

BusyBox v0.60.5 (2005.05.12-22:46+0000) Built-in shell (msh)
```

사용자는 `rupgrade_tool -w` 명령을 입력하여 안정적인 업그레이드의 상태를 확인합니다.

```
# rupgrade_tool -w
Recent upgrade status:
1:PLB is 5
1:EEPROM page saved
2:persistent flash is 1
3:provisional flash is 0
4:copying image(s)
4:invoking scripts (step4v*) [--u=u-boot.bin --k=sentry.kernel --
r=sentry.rfs --hook=etc_copy]
4:copying u-boot.bin from /tmp to /dev/mtdchar8 using 'cp'
protocol
4:copying sentry.kernel from /tmp to /dev/mtdchar7 using 'cp'
protocol
4:copying sentry.rfs from /tmp to /dev/mtdchar9 using 'cp'
protocol
4:invoking scripts (step4h*) [etc_copy]
4:image(s) copy OK
5:watchdog started
6:selected provisional flash
7:reboot
9:WDT not fired, upgrade in progress.
11:provisional flash 0, updating EEPROM
12:EEPROM updated
13:upgrade WDT disabled
13:invoking scripts (step13h*) []
14:upgrade completed successfully
#
```

## 예 2:

이 예에서는 /etc 및 /var/nvdata 비휘발성 디렉토리를 임시 플래시에 복사하여 RFS 이미지에 대해서만 안정적인 업그레이드를 수행합니다. RFS 이미지는 IP 주소 192.168.1.253에 있는 FTP 서버에서 가져옵니다. FTP 서버에서 RFS 이미지의 경로는 /tftpboot/ru-mips/sentry.mips.rfs입니다. 업그레이드 절차는 텔넷 세션에서 시작됩니다.

---

주 – RFS 이미지만 명시적으로 업데이트되므로 U-Boot 및 커널 이미지는 영구 플래시 분할 영역에서 임시 플래시 분할 영역으로 자동 복사됩니다.

---

로컬 시스템에서 네트워크를 통해 FTP 서버에 액세스할 수 있어야 합니다. 즉 네트워크 어댑터가 구성되어 실행되고 있고 ShMM에서 FTP 서버까지의 경로가 존재해야 합니다. 이 예에서 ShMM은 네트워크 주소 192.168.1.174(FTP 서버와 동일한 네트워크에 있음)로 구성됩니다.

```
# telnet 192.168.1.174
Trying 192.168.1.174...
Connected to 192.168.1.174.
Escape character is '^]'.

BusyBox on shmm+174 login: root

BusyBox v0.60.5 (2005.05.07-17:27+0000) Built-in shell (msh)
```

rupgrade\_tool -s에 대한 매개 변수는 RFS만 업그레이드된다는 것과 복사 프로토콜이 FTP라는 것을 나타내며 admin 사용자를 사용하여 암호 없이 지정된 IP 주소와 파일에 액세스합니다.

```
# rupgrade_tool -s --r=sentry.mips.rfs
--proto=ftp:192.168.1.253:/tftpboot/ru-mips:admin --hook=etc_copy
-v
rupgrade_tool: PLB is 5
rupgrade_tool: EEPROM page saved
rupgrade_tool: persistent flash is 1
rupgrade_tool: provisional flash is 0
rupgrade_tool: copying image(s)
rupgrade_tool: copying sentry.rfs from
192.168.1.253:/tftpboot/ru-mips:admin to /tmp using 'ftp' protocol
220 hydra FTP server (Version wu-2.4.2-academ[BETA-17] (1) Tue Jun
9 10:43:14 EDT 1998) ready.
USER admin
```

여기서 FTP 사이트에 대한 암호를 묻는 메시지가 표시됩니다. 암호를 직접 입력해야 합니다.

```
331 Password required for admin.
Password:xxxxx
PASS *****
230 User admin logged in.
TYPE I
200 Type set to I.
PASV
227 Entering Passive Mode (192,168,1,253,9,20)
RETR /tftpboot/ru-mips/sentry.mips.rfs
150 Opening BINARY mode data connection for
/tftpboot/ru-mips/sentry.mips.rfs (2465988 bytes).
226 Transfer complete.
QUIT
221 Goodbye.
```

다음 단계에서는 특수 스크립트 step4vshrm을 호출하여 U-Boot 및 커널 이미지를 영구 플래시에서 임시 플래시로 복사합니다. 그런 다음 업그레이드 유틸리티가 임시 플래시의 지정된 위치에 RFS 이미지를 복사합니다.

```
rupgrade_tool: invoking scripts (step4v*) [--r=sentry.mips.rfs --
proto=ftp:192.168.1.253:/tftpboot/ru-mips:admin --hook=etc_copy]
/etc/upgrade/step4vshrm: Erasing /dev/mtdchar7...Done
etc/upgrade/step4vshrm: Copying Kernel from /dev/mtdchar2 to
/dev/mtdchar7...Done
/etc/upgrade/step4vshrm: Erasing /dev/mtdchar8...Done
/etc/upgrade/step4vshrm: Copying U-Boot from /dev/mtdchar3 to
/dev/mtdchar8...Done
rupgrade_tool: copying sentry.mips.rfs from /tmp to /dev/mtdchar9
using 'cp' protocol
```

step4hshrm 후크 스크립트가 호출됩니다. 이 스크립트는 선반 관리자를 중지하고 비휘발성 데이터를 유지합니다. 그런 다음 업그레이드 WDT를 시작하고 다시 부트합니다.

```
rupgrade_tool: invoking scripts (step4h*) [etc_copy]
/etc/upgrade/step4hshrm: Stopping Shelf Manager...
/etc/upgrade/step4hshrm: Erasing /var and /etc, copying
/var/nvdata..
/etc/upgrade/step4hshrm: Operation: copy /etc and /var/nvdata.
/etc/upgrade/step4hshrm: Copying completed.
rupgrade_tool: image(s) copy OK
rupgrade_tool: watchdog started
rupgrade_tool: selected provisional flash
```

```
rupgrade_tool: reboot
Restarting system.
Connection closed by foreign host.
```

이 지점에서 특정 비활성 기간이 경과한 후 텔넷 세션이 종료됩니다. 몇 초 후에 대상에 다시 연결할 수도 있으며 `rupgrade_tool -w`를 호출하여 안정적인 업그레이드의 상태를 확인할 수 있습니다.

```
# telnet 192.168.1.174
Trying 192.168.1.174...
Connected to 192.168.1.174.
Escape character is '^]'.

BusyBox on shmm+174 login: root

BusyBox v0.60.5 (2005.05.07-17:27+0000) Built-in shell (msh)
#
# rupgrade_tool -w
Recent upgrade status:
1:PLB is 5
1:EEPROM page saved
2:persistent flash is 1
3:provisional flash is 0
4:copying image(s)
4:copying sentry.mips.rfs from 192.168.1.253:/tftpboot/rumips:admin to /tmp using 'ftp' protocol
4:invoking scripts (step4v*) [--r=sentry. rfs --hook=etc_copy]
4:copying sentry.mips.rfs from /tmp to /dev/mtdchar9 using 'cp' protocol
4:invoking scripts (step4h*) [etc_copy]
4:image(s) copy OK
5:watchdog started
6:selected provisional flash
7:reboot
9:WDT not fired, upgrade in progress.
11:provisional flash 0, updating EEPROM
12:EEPROM updated
13:upgrade WDT disabled
13:invoking scripts (step13h*) []
14:upgrade completed successfully
#
```

### 예 3:

이 예에서는 안정적인 업그레이드가 실패한 경우를 나타냅니다. 임시 플래시로부터 부트한 후 안정적인 업그레이드가 완료되기 이전에 전원을 끕니다. 전원을 다시 켜면 영구 플래시로 롤백이 수행됩니다. 이 안정적인 업그레이드는 직렬 콘솔에서 시작됩니다. 세 이미지 모두 /tmp에 이미 있는 것으로 가정합니다.

```
# rupgrade_tool -s --k=sentry.kernel --r=sentry.rfs --u=u-boot.bin
--hook=etc_copy -v
rupgrade_tool: PLB is 5
rupgrade_tool: EEPROM page saved
rupgrade_tool: persistent flash is 0
rupgrade_tool: provisional flash is 1
rupgrade_tool: copying image(s)
rupgrade_tool: invoking scripts (step4v*) [--u=u-boot.bin --k=
sentry.kernel --r=sentry.rfs --hook=etc_copy]
rupgrade_tool: copying u-boot.bin from /tmp to /dev/mtdchar8 using
'cp' protocol
rupgrade_tool: copying sentry.kernel from /tmp to /dev/mtdchar7
using 'cp' protocol
rupgrade_tool: copying sentry.rfs from /tmp to /dev/mtdchar9 using
'cp' protocol
rupgrade_tool: invoking scripts (step4h*) [etc_copy]
Stopping Shelf Manager...

IPM Sentry Shelf Manager Command Line Interpreter

Terminating the

Erasing /var and /etc, copying /var/nvdata...
Operation: copy /etc and /var/nvdata.
Copying completed.
rupgrade_tool: image(s) copy OK
rupgrade_tool: watchdog started
rupgrade_tool: selected provisional flash
rupgrade_tool: reboot
Restarting system.
```

안정적인 업그레이드 절차는 여기서 ShMM을 재설정하고 임시 플래시에서 U-boot를 시작합니다.

```
** Resetting Integrated Peripherals
```

```
U-Boot 1.1.2 (Nov 11 2005 - 15:16:25)
```

```

CPU: Aul550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 8000044
DRAM: 128 MB
Flash: 64 MB
In: serial
Out: serial
Err: serial
Net: AulX00 ETHERNET
Hit any key to stop autoboot: 0
## Booting image at bfb00000 ...
   Image Name: MIPS Linux-2.4.26
   Created: 2005-04-11 10:35:08 UTC
   Image Type: MIPS Linux Kernel Image (gzip compressed)
   Data Size: 843129 Bytes = 823.4 kB
   Load Address: 80100000
   Entry Point: 802bc040
   Verifying Checksum ... OK
   Uncompressing Kernel Image ... OK
## Loading Ramdisk Image at bfc40000 ...
   Image Name: sentry RFS Ramdisk Image
   Created: 2005-04-22 9:10:41 UTC
   Image Type: MIPS Linux RAMDisk Image (gzip compressed)
   Data Size: 2400736 Bytes = 2.3 MB
   Load Address: 00000000
   Entry Point: 00000000
   Verifying Checksum ... OK

```

여기서 전원을 끕니다. 잠시 후 전원을 다시 켭니다. 정전으로 인해 임시 플래시 할당이 손실되어 시스템이 영구 플래시로 되돌아갑니다.

```

U-Boot 1.1.2 (Nov 11 2005 - 15:16:25)

CPU: Aul550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 8000048
DRAM: 128 MB
Flash: 64 MB
In: serial
Out: serial
Err: serial
Net: AulX00 ETHERNET
Hit any key to stop autoboot: 0
## Booting image at bfb00000 ...
   Image Name: MIPS Linux-2.4.26

```



```

Created:      2005-04-11  10:35:08 UTC
Image Type:   MIPS Linux Kernel Image (gzip compressed)
Data Size:    843129 Bytes = 823.4 kB
Load Address: 80100000
Entry Point:  802bc040
Verifying Checksum ... OK
Uncompressing Kernel Image ... OK
## Loading Ramdisk Image at bfc40000 ...
Image Name:   sentry RFS Ramdisk Image
Created:      2005-04-11  18:27:17 UTC
Image Type:   MIPS Linux RAMDisk Image (gzip compressed)
Data Size:    2372311 Bytes = 2.3 MB
Load Address: 00000000
Entry Point:  00000000
Verifying Checksum ... OK

Starting kernel ...

init started: BusyBox v0.60.5 (2005.02.07-16:45+0000) multi-call
binary
hub.c: new USB device AU1550-1, assigned address 2
usb0: ? speed config #1: Ethernet Gadget
usb1: register usbnet usb-AU1550-1, Linux Device
serial#=8000048: not found
/etc/rc: Mounted /proc
/etc/rc: Mounting filesystems...
/etc/rc: Mounted /dev/pts
/etc/rc: Mounted /dev/mtdblock0 to /var
/etc/rc: Mounted /dev/mtdblock10 to /var/upgrade

```

rc 스크립트의 다음 단계에서는 `rupgrade_tool -c`를 호출하여 안정적인 업그레이드가 진행 중인지 확인합니다. 검사를 통해 안정적인 업그레이드 시도가 실패했음을 확인합니다. 메시지 `restoring ADM1060 EEPROM to RAM`은 ShMM 부트 프로세스를 감독하고 안정적인 업그레이드에 대한 일부 하드웨어 부분 지원을 구현하는 ShMM 시스템 감독 장치(ADM1060)를 나타냅니다. 이 메시지는 부트 프로세스에 영향을 미치는 주요 변수가 안정적인 업그레이드를 시도하기 이전의 상태로 복원됨을 나타냅니다.

```

/etc/rc: Checking the reliable upgrade watchdog timer
rupgrade_tool: Watchdog not active.
rupgrade_tool: restoring ADM1060 EEPROM to RAM
rupgrade_tool: upgrade failed
/etc/rc: Rupgrade -c Ret: 255
/etc/rc: Mounted ram disk to /var/log
/etc/rc: Started syslogd and klogd

```

```

/etc/rc: Mounted ram disk to /var/tmp
/etc/rc: Setting hostname shmm+173
/etc/rc: Mounted /dev/mtdblock1 to /etc
/etc/rc: Calling /etc/rc.carrier3
Board Hardware Address: 0xFE
/etc/netconfig: /etc/hosts has valid 192.168.1.173 entry
/etc/netconfig: Updating /etc/profile.sentry with IP settings
/etc/netconfig: Starting inetd...
/etc/rc.carrier3: Starting up IPMBs ...
/etc/rc.carrier3: Updating /etc/profile.sentry with specific
settings
/etc/rc.carrier3: RC2 daemons not started by request

```

## CPLD 프로그래밍

ShMM에는 CPLD 장치가 포함되어 있습니다. 이 CPLD 장치는 하드웨어 수준 중복 인터페이스와 같은 ShMM 작업의 몇 가지 핵심 측면을 제어합니다.

### ▼ ShMM에 대해 CPLD 이미지를 재프로그래밍하는 방법

1. 다음 파일을 선반 관리 카드의 /var/bin 디렉토리에 업로드합니다.

- cpld\_tool
- shmm-cpld-erase.xsvf
- shmm-cpld-1x.xsvf
- shmm-cpld.xsvf

2. 다음과 같이 **CPLD**를 지웁니다.

```
# cpld_tool -x shmm-cpld-erase.xsvf
```

3. 제공된 이미지를 사용하여 **CPLD**를 프로그래밍합니다.

```
# cpld_tool -x shmm-cpld.xsvf
```

4. **CPLD** 이미지의 유효성을 확인합니다.

사용자 ID가 0x33623030과 같아야 합니다.

```
# cpld_tool -u
0x33623030
```

## 노드 보드 콘솔에 연결

선반 관리자는 노드 보드에 연결하고 활성 선반 관리 카드(ShMM)에서 콘솔 세션을 여는 기능을 제공합니다. 직렬 포트 또는 이더넷 포트를 통해 활성 ShMM에 로그인하여 시작합니다.

---

주 – 콘솔 기능을 사용하려면 기본(위쪽) 선반 관리 카드가 활성 선반 관리 카드여야 합니다. 또한 Netra CT 900 서버 미드플레인의 슬롯 7에 스위치 카드를 설치해야 합니다.

---

노드 보드가 있는 콘솔 세션을 설정한 경우 `passwd`와 같은 시스템 관리 명령을 실행하거나 상태 및 오류 메시지를 보거나 해당 슬롯의 보드를 중지할 수 있습니다.

---

주 – 콘솔 또는 직렬 케이블을 노드 보드의 직렬 포트에 연결하면 케이블을 연결할 때 ShMM 콘솔 센션이 활성 상태였더라도 콘솔 출력은 ShMM의 콘솔 세션이 아니라 케이블로 연결된 콘솔로 전달됩니다.

---

## 선반 관리자와 노드 보드 간의 콘솔 세션 설정

시스템을 콘솔용으로 구성한 경우 활성 ShMM에 로그인한 다음 슬롯에 대한 콘솔을 열 수 있습니다. Netra CT 시스템에서는 노드 보드 슬롯당 하나의 콘솔 세션을 허용합니다.

[표 3-4](#)에서는 선반 관리 카드의 현재 로그인 세션에서 실행될 수 있는 선반 관리자 CLI 콘솔 관련 명령을 나타냅니다.

**표 3-4** 선반 관리자 CLI 콘솔 관련 명령

명령	설명
<code>clia console slot_no</code>	콘솔 모드를 시작하고 <code>slot_no</code> 에 있는 노드 보드에 연결합니다. 여기서 <code>slot_no</code> 는 노드 보드가 있는 슬롯 번호입니다.
<code>~q</code> or <code>~.</code>	현재 콘솔 세션의 연결을 끊습니다.

## ▼ 선반 관리자에서 콘솔 세션을 시작하는 방법

### 1. 기본(위쪽) ShMM에 로그인합니다.

직렬 포트 연결 또는 이더넷 포트 연결에 부착된 터미널을 통해 기본(위쪽) ShMM에 로그인할 수 있습니다.

### 2. 기본 ShMM이 활성 ShMM인지 확인합니다.

로그온한 후 `clia shmstatus` 명령을 사용하여 활성 ShMM에 로그인했는지 확인한 다음 계속합니다. 대기 ShMM에 로그인한 경우 `clia switchover` 명령을 사용하여 ShMM을 Active로 변경합니다. 자세한 내용은 [109페이지의 "선반 관리자 CLI 명령"](#)의 `shmstatus` 및 `switchover`를 참조하십시오.

### 3. 노드 보드에 대한 콘솔 세션을 엽니다.

```
# clia console slot_no
```

여기서 `slot_no`는 1 - 6 및 9 - 14입니다. 예를 들어 슬롯 4의 보드에 대한 콘솔을 열려면 다음과 같이 입력합니다.

```
# clia console 4
```

이제 슬롯 4에 있는 노드 보드에 액세스할 수 있습니다. 해당 슬롯에 있는 보드의 상태와 이전 사용자가 셸에서 로그아웃했는지 여부에 따라 다음 중 하나의 프롬프트가 표시됩니다.

- `console login%(Solaris 수준)`
- `#(Solaris 수준, 슈퍼유저로 로그인한 이전 사용자가 콘솔 연결을 끊기 전에 로그아웃하지 않음)`
- `ok(OpenBoot PROM 수준)`
- `#(Monta Vista Linux)`

## ▼ 콘솔 세션을 종료하는 방법

1. (선택 사항) OS 셸에서 로그아웃합니다.
2. 프롬프트에 이스케이프 문자열 ~q 또는 ~. (틸드와 마침표)를 입력하여 콘솔 연결을 끊습니다.

```
prompt ~q
```

콘솔에서 연결을 끊으면 원격 호스트에서 자동으로 로그아웃되지 않습니다. 원격 호스트에서 로그아웃하지 않을 경우 다음 콘솔 사용자가 해당 보드에 연결하면 이전 세션의 셸 프롬프트가 표시됩니다.

---

주 - 작업이 완료되면 콘솔 세션에서 항상 로그아웃하십시오.

---

## 수동으로 노드 보드 정상 종료

이 릴리스에서는 노드 보드 또는 FRU를 실수로 제거하지 않도록 수동 정상 종료 기능을 제공합니다. 정상 종료를 사용하면 노드 보드의 페이로드에서 실행 중인 모든 응용 프로그램과 OS, 그리고 페이로드 자체를 정상적으로 중단(종료)할 수 있습니다.

Netra 노드 보드를 핫 스왑하거나 제거하기 전에 해당 노드 보드에서 실행 중인 응용 프로그램과 운영 체제를 정상적으로 종료해야 합니다. 시스템 관리자는 선반 관리자의 netconsole 기능을 사용하여 노드 보드에서 콘솔 세션을 시작하고 사이트의 종료 절차에 따라 정상 종료를 수행할 수 있습니다.

종료가 완료되면 시스템 관리자는 FRU 또는 노드 보드를 비활성화하고 보드에서 핫 스왑 래치를 엽니다. 노드 보드를 교체하거나 다시 설치한 후 관리자는 FRU 또는 노드 보드를 비활성화하여 핫 스왑 래치를 열 때 비정상적으로 종료되지 않게 할 수 있습니다.

이 프로세스의 단계에 대한 자세한 내용은 다음 절차를 참조하십시오.

## ▼ 노드 보드를 종료하는 방법

이 절차에서는 노드 보드의 핫 스왑 래치를 닫아야 합니다.

---

주 - 핫 스왑 래치가 열려 있으면 파란색 핫 스왑 LED가 계속 깜박이고 보드가 핫 스왑 준비 상태(파란색 LED 켜짐)로 되지 않습니다. 이 문제를 해결하려면 이 절차를 시작하기 전에 핫 스왑 래치를 닫습니다.

---

1. 활성 **ShMM**에 로그인합니다.
2. 노드 보드에 대한 콘솔 세션을 시작합니다.  
104페이지의 "선반 관리자에서 콘솔 세션을 시작하는 방법"을 참조하십시오.
3. 노드 보드에서 응용 프로그램과 운영 체제를 종료합니다.  
사이트의 정상 종료 절차를 따릅니다. 종료가 완료되면 콘솔 세션을 닫습니다([콘솔 세션을 종료하는 방법](#) 참조).
4. 다음 선반 관리자 **CLI** 명령을 사용하여 노드 보드에서 선반 관리자 제어 비활성화를 사용합니다.

```
# clia shelf deactivate hardware-addr fru-id 0
```

예를 들면 다음과 같습니다.

```
# clia shelf deactivate 0x41 0xfe 0
IPM Sentry Shelf Manager Command Line Interpreter
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info, address: 0x41, FRU ID # 254
    Cached information updated
    Wrote Information to the Shelf FRU
#
```

5. 다음 선반 관리자 **CLI** 명령을 사용하여 선반 관리자 제어 비활성화를 사용하는지 확인합니다.

```
# clia shelf pm
```

Shelf Manager Controlled Deactivation: Enabled 메시지를 조사합니다.  
예를 들면 다음과 같습니다.

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
    Version = 1
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 16
    Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Shelf Manager Controlled Deactivation: Enabled
    Delay Before Next Power On: 0.0 seconds
```

6. 노드 보드에서 핫 스왑 래치를 엽니다.
7. 노드 보드의 파란색 핫 스왑 **LED**가 켜져 있으면 노드 보드를 제거하거나 교체합니다.



---

주의 – 모듈을 다룰 때는 항상 접지된 방전 손목 고정대를 착용하십시오.

---

8. 노드 보드를 교체하거나 다시 설치할 경우 다음 선반 관리자 **CLI** 명령을 사용하여 노드 보드에서 선반 관리자 제어 비활성화를 해제합니다.

```
# clia shelf deactivate hardware-addr fru-id 1
```

예를 들면 다음과 같습니다.

```
# clia shelf deactivate 0x41 0xfe 1
IPM Sentry Shelf Manager Command Line Interpreter
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info, address: 0x41, FRU ID # 254
    Cached information updated
    Wrote Information to the Shelf FRU
#
```





# 선반 관리자 CLI 명령

이 장에서는 각 CLI 명령에 대해 설명하고 사용 가능한 각 명령의 구문과 사용법을 소개합니다. CLI는 AdvancedTCA 선반 컨텍스트를 지원합니다.

편의상 선반 구성 요소의 주요 유형은 IPMB 주소 및 숫자 FRU 식별자를 기반으로 한 참조 표기법 대신 다음과 같은 방식으로 참조될 수 있습니다.

- board *n* | b *n*
- fan\_tray *n* | ft *n*
- shm 1 | 2

주 - shm 1 및 shm 2 표기법을 사용하여 선반 FRU의 주소 테이블에 기술된 중복 선반 관리자에 액세스할 수 있습니다. 이 설명서에서 shm 1은 하드웨어 주소 숫자가 더 작은 선반 관리자에 관련되고, shm 2는 하드웨어 주소의 숫자가 더 큰 선반 관리자에 관련되어 있습니다.

중복 구성에서는 아래 나열된 명령 중 백업 선반 관리자에서 지원되지 않는 명령도 있습니다. 백업 선반 관리자는 다음 명령만 인식합니다.

- debuglevel
- localaddress
- shmstatus
- switchover

대부분의 정보용 명령은 실행 시 간단 정보 표시 모드와 세부 정보 표시 모드를 지원하며 모드에 따라 제공하는 정보의 양이 다릅니다. 간단 정보 표시 모드가 기본값(표준)이며, 세부 정보 표시 모드는 명령줄에서 명령 바로 다음 및 위치 인수 앞에 -v 옵션을 사용하여 선택합니다.

아래의 명령 구문에서 선택적 요소는 각괄호([, ]) 안에 표시되고, 명령줄의 변수 요소(예: IPMB 주소 및 FRU 장치 ID)는 기울임꼴로 표시됩니다. 세로 막대(|)는 대체 매개 변수를 분리합니다.

# activate

구문:

```
activate IPMB-address fru-id
activate board n
activate shm n
```

용도:

이 명령은 지정된 FRU에 IPMI 명령 Set FRU Activation (Activate FRU) 을 전송합니다. FRU는 소유 IPM 제어기의 IPMB 주소와 FRU 장치 ID를 사용하여 지정됩니다. FRU 장치 ID 0은 PICMG 3.0 컨텍스트에서 적합한 IPM 제어기를 지정합니다.

PICMG 3.0 컨텍스트에서 이 명령은 기본적으로 선반 FRU 정보의 전원 관리 테이블에 나열되지 않은 FRU 또는 Shelf Manager Controlled Activation 속성이 FALSE로 설정된 FRU에 유용합니다. 이들 FRU는 선반 관리자가 자동으로 활성화하지 않으며 M2(활성화 요청) 상태로 남아 있습니다. 다른 FRU가 M2 상태에 도달하면 선반 관리자가 이를 자동으로 활성화합니다. M2 상태가 아닌 FRU를 활성화하려고 하면 아무 작업도 수행되지 않습니다.

예:

주소 9C에서 적합한 IPM 제어기를 활성화합니다.

```
# clia activate 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
Command issued via IPMB, status = 0 (0x0)
Command executed successfully
#
```

---

# alarm

구문:

```
alarm [clear|minor|major|critical]
```

용도:

이 명령은 Telco 알람 출력에 대한 액세스를 제공합니다. 매개 변수 minor, major 및 critical을 사용하면 사용자는 해당 알람 출력을 설정할 수 있습니다. 이 작업은 누적됩니다. 즉, 명령 `clia alarm minor` 및 `clia alarm major`를 실행하면 주요 알람과 차요 알람이 모두 설정됩니다. `clear` 작업은 차요 및 주요 알람 출력을 지웁니다. 위험 알람 출력은 지울 수 없습니다.

매개 변수 없이 명령을 실행하면 Telco 알람 출력의 상태가 반환됩니다.

예:

```
# clia alarm
IPM Sentry Shelf Manager Command Line Interpreter
alarm mask: 0x00
# clia alarm major
IPM Sentry Shelf Manager Command Line Interpreter
Returned completion code: 0
# clia alarm
IPM Sentry Shelf Manager Command Line Interpreter
alarm mask: 0x02
# clia alarm clear
IPM Sentry Shelf Manager Command Line Interpreter
Returned completion code: 0
# clia alarm
IPM Sentry Shelf Manager Command Line Interpreter
alarm mask: 0x00
```

## board

구문:

board [-v] [*physical-slot-address*]

용도:

이 명령과 boardreset 명령은 ATCA 보드와 작동하며 IPM 제어기 주소와 FRU 장치 ID가 아니라 물리적 슬롯 번호를 인수로 사용한다는 점에서 명령 세트의 다른 명령과 다릅니다. 명령 board는 ATCA 슬롯에 할당된 IPMB 주소 범위에 있는 각 IPM 제어기에 대한 정보와 이 제어기가 제어하는 각각의 추가 FRU에 대한 정보를 표시합니다. 표시될 항목의 목록은 [125페이지의 "fru"](#) 및 [169페이지의 "ipmc"](#)를 참조하십시오. IPMB 주소의 범위는 보드에 IPM 제어기가 있는 PICMG 3.0 시스템의 경우 82h-A0h입니다.

물리적 주소는 십진수로 지정해야 합니다. PICMG 3.0 시스템의 경우 물리적 주소와 IPMB 주소 간의 대응은 선반 FRU 정보에 지정됩니다. 선반 FRU 정보에 주소 테이블이 없으면 다음 매핑 테이블(논리적 슬롯 번호의 매핑)이 사용됩니다.

슬롯 번호	IPMB 주소
1	9A
2	96
3	92
4	8E
5	8A
6	86
7	82
8	84
9	88
10	8C
11	90
12	94
13	98
14	9C

예:

시스템(물리적 슬롯 1과 14에만 보드가 있는)의 모든 보드에 대한 표준 정보를 가져옵니다.

```
# clia board
IPM Sentry Shelf Manager Command Line Interpreter
Physical Slot # 1
82: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
82: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
Physical Slot # 14
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
#
```

물리적 슬롯 14의 보드에 대한 자세한 정보를 가져옵니다.

```
# clia board -v 14
Physical Slot # 14
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID: 0x00, Revision: 0, Firmware: 1.01, IPMI ver 1.5
    Manufacturer ID: 00315a (PICMG), Product ID: 0000, Auxiliary
Rev: 01ac1014
    Device ID String: "IPM Sentry 6"
    Global Initialization: 0x0, Power State Notification: 0x0,
Device Capabilities: 0x29
    Controller provides Device SDRs
    Supported features: 0x29
```

```

    "Sensor Device" "FRU Inventory Device" "IPMB Event
Generator"
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
    Site Type: 0x00, Site Number: 14
    Current Power Level: 0x01, Maximum Power Level: 0x01, Current
Power Consumption: 20.0 Watts
#

```

## boardreset

구문:

`boardreset physical-slot-address`

용도:

이 명령은 지정된 물리적 슬롯의 보드를 재설정하고 보드에 IPMI 명령 FRU Control (Cold Reset)을 보냅니다.

물리적 주소는 십진수로 지정해야 합니다. PICMG 3.0 시스템의 경우 물리적 주소와 IPMB 주소 간의 대응은 선반 FRU 정보에 지정됩니다. 선반 FRU 정보에 주소 테이블이 없으면 다음 매핑 테이블(논리적 슬롯 번호의 매핑)이 사용됩니다. FRU 장치 ID는 0입니다.

슬롯 번호	IPMB 주소
1	9A
2	96
3	92
4	8E
5	8A
6	86
7	82
8	84
9	88

슬롯 번호	IPMB 주소
10	8C
11	90
12	94
13	98
14	9C

예:

물리적 슬롯 14(IPMB 주소 9C, FRU 0)의 보드를 재설정합니다.

```
# clia boardreset 14
IPM Sentry Shelf Manager Command Line Interpreter
Board 14 reset, status returned 0
#
```

## busres

구문:

`busres subcommand`

다음과 같은 하위 명령이 지원됩니다.

- `info [resource]`
- `release resource`
- `force resource`
- `lock resource`
- `unlock resource`
- `query [-v] resource [target [nouupdate]]`
- `setowner resource target`
- `sendbusfree resource target`

용도:

이 명령을 사용하면 버스된 전자 키잉 관리 자원의 현재 상태에 대한 정보가 표시되며 해당 상태를 변경할 수 있습니다.

모든 하위 명령에서 자원 ID를 매개 변수의 하나로 사용할 수 있습니다. 자원 ID는 0 기반 자원 번호이거나 짧은 자원 이름입니다. 다음과 같은 자원 이름과 번호가 정의됩니다.

번호	짧은 이름	설명
0	mtb1	금속 테스트 버스 쌍 1
1	mtb2	금속 테스트 버스 쌍 2
2	clk1	동기화 클럭 그룹 1
3	clk2	동기화 클럭 그룹 2
4	clk3	동기화 클럭 그룹 3

다음 세부절에서는 여러 관건적인 용도의 `busres` 명령 구문에 대해 설명합니다.

## 버스된 전자 키잉 관리 자원의 상태 표시

구문:

```
busres info [resource]
```

용도:

이 명령은 지정된 자원 또는 자원 ID가 지정되지 않은 경우 모든 자원의 현재 상태에 대한 정보를 표시합니다.

매개 변수 *resource*는 자원 ID입니다. 지원되는 자원 ID 목록은 [115페이지의 "busres"](#)에 표시되어 있습니다.

예:

금속 테스트 버스 쌍 2의 상태에 대한 정보를 가져옵니다.

```
# clia busres info mtb2
IPM Sentry Shelf Manager Command Line Interpreter
Metalic Test Bus pair 2 (ID 1): Owned by IPMC 0x82, Locked
#
```

## 지정된 자원 해제

구문:

```
busres release | force resource
```



## 용도:

이 명령은 자원의 현재 소유자에게 Bused Resource Control 요청을 전송하여 자원 해제를 지시합니다. 명령 구문이 `busres release resource`인 경우 Bused Resource Control (Release) 명령이 전송됩니다. 명령 구문이 `busres force resource`인 경우 Bused Resource Control (Force) 명령이 전송됩니다. ATCA 명령에 대한 자세한 설명은 PICMG 3.0 R1.0 사양의 3.7.3.4절을 참조하십시오.

매개 변수 `resource`는 자원 ID입니다. 지원되는 자원 ID 목록은 [115페이지의 "busres"](#)에 표시되어 있습니다.

## 예:

현재 소유자가 금속 테스트 버스 쌍 2를 강제로 해제합니다.

```
# clia busres force mtb2
IPM Sentry Shelf Manager Command Line Interpreter
Force operation succeeded
#
```

# 지정된 자원 잠금/잠금 해제

## 구문:

`busres lock | unlock resource`

## 용도:

이 명령은 지정된 자원을 잠그거나(`busres lock resource`) 잠금 해제(`busres unlock resource`)합니다. 자원이 잠긴 경우 다른 IPM 제어기가 Bused Resource Control (Request) 명령을 선반 관리자에게 전송하면 선반 관리자는 "거부" 상태로 응답합니다. 자원 잠금이 해제된 경우 다른 IPM 제어기가 Bused Resource Control (Request) 명령을 선반 관리자에게 전송하면 선반 관리자는 "사용 중" 상태로 응답하고 Bused Resource Control (Release)을 현재 소유자에게 전송합니다. 현재 소유자가 자원을 해제하면 이 자원은 다음 요청에서 요청자에게 부여됩니다.

IPM 제어기가 소유한 자원만 잠글 수 있습니다. 현재 소유자가 자원을 해제하면 이 자원에서 잠금도 제거됩니다.

매개 변수 `resource`는 자원 ID입니다. 지원되는 자원 ID 목록은 [115페이지의 "busres"](#)에 표시되어 있습니다.

예:

동기화 클럭 그룹 3을 잠급니다.

```
# clia busres lock clk3
IPM Sentry Shelf Manager Command Line Interpreter
Lock operation succeeded
#
```

## Bused Resource Control (Query) 명령 전송

구문:

```
busres [-v] query resource [target [noupdate]]
```

용도:

이 명령은 지정된 IPM 제어기에 Bused Resource Control (Query) 요청을 전송합니다. 명령줄에 IPM 제어기를 지정하지 않으면 요청이 자원의 현재 소유자에게 전송됩니다. `noupdate` 플래그가 사용된 경우를 제외하면 응답 수신 시 자원 테이블이 적절하게 변경됩니다(예를 들어 현재 소유자로 간주된 IPM 제어기가 "무제어" 상태로 응답하면 테이블이 해당 사실을 반영하도록 수정됨). 명령줄에서 이 플래그를 사용한 경우 자원 테이블이 수신된 정보에 따라 변경되지 않습니다.

매개 변수 *resource*는 자원 ID입니다. 지원되는 자원 ID 목록은 [115페이지](#)의 "busres"에 표시되어 있습니다.

매개 변수 *target*은 요청을 전송할 IPM 제어기의 IPMB 주소를 지정합니다.

플래그 `noupdate`가 있는 경우 이 플래그는 쿼리 요청에 대한 응답으로 수신된 정보가 자원 테이블 업데이트에 사용되지 말아야 함을 의미합니다.

`-v` 플래그가 지정되지 않은 경우 현재 버전의 선반 관리자에서 추가 정보가 제공되지 않습니다.

예:

금속 테스트 버스 쌍 1에 대한 쿼리를 주소가 0x82인 IPM 제어기에 전송합니다. 응답에 따라 자원 테이블을 업데이트하지 않습니다.

```
# clia busres query mtb1 0x82 noupdate
IPM Sentry Shelf Manager Command Line Interpreter
No Control: IPMC 0x82 is not the owner of resource 0
#
```

## 자원의 소유자 설정

구문:

```
busres setowner resource target
```

용도:



---

주의 - 이 명령은 숙련된 사용자를 위한 것이므로 신중하게 사용해야 합니다.

---

이 명령은 자원 테이블에 지정된 자원의 소유자를 직접 설정합니다. 명령을 실행하기 전에 자원의 소유자가 다르다 할지라도 **Bused Resource Control** 명령을 전송하지 않습니다. 이 명령은 테스트 및 복구 목적으로만 사용해야 하는 저수준 명령입니다.

매개 변수 *resource*는 자원 ID입니다. 지원되는 자원 ID 목록은 [115페이지의 "busres"](#)에 표시되어 있습니다.

매개 변수 *target*은 자원의 소유자로 설정된 IPM 제어기의 IPMB 주소를 지정합니다. 자원이 어떠한 IPM 제어기의 소유도 아님을 지정하려면 IPMB 주소로 0을 사용합니다.

예:

보드 1을 금속 테스트 버스 쌍 1의 새 소유자로 설정합니다.

```
# clia busres setowner mtb1 board 1
IPM Sentry Shelf Manager Command Line Interpreter
New owner is set successfully
#
```

## Bused Resource Control (Bus Free) 명령 전송

구문:

```
busres sendbusfree resource target
```

용도:



---

주의 - 이 명령은 숙련된 사용자를 위한 것이므로 신중하게 사용해야 합니다.

---

이 명령은 지정된 IPM 제어기에 Bused Resource Control (Bus Free) 요청을 전송합니다. 다른 IPM 제어기가 소유하고 있더라도 요청을 전송하기 전에는 자원에 대해 어떠한 작업도 수행되지 않습니다. 하지만 자원 테이블은 이 요청에 대한 응답에 따라 업데이트됩니다. 즉, IPM 제어기가 자원의 소유권을 허용하면 해당 테이블에 이 제어기가 새 소유자로 설정됩니다. 이 명령은 테스트 및 복구 목적으로만 사용해야 하는 저수준 명령입니다.

매개 변수 *resource*는 자원 ID입니다. 지원되는 자원 ID 목록은 [115페이지의 "busres"](#)에 표시되어 있습니다.

매개 변수 *target*은 요청을 전송할 IPM 제어기의 IPMB 주소를 지정합니다. 자원이 어떠한 IPM 제어기의 소유도 아님을 지정하려면 IPMB 주소로 0을 사용합니다.

예:

금속 테스트 버스 쌍 1에 대한 버스 해제 요청을 주소가 0x82인 IPM 제어기에 전송합니다.

```
# clia busres sendbusfree mtb1 0x82
IPM Sentry Shelf Manager Command Line Interpreter
IPMC rejected ownership of the resource
#
```

---

## console

구문:

`console slot-number`

용도:

이 명령은 지정된 물리적 슬롯의 노드 보드에 콘솔 터미널 세션을 만듭니다. 선반 관리자는 노드 보드당 하나의 콘솔 세션을 허용합니다. 유효한 슬롯 번호는 1-6 및 9-14입니다.

---

주 – 콘솔 기능을 사용하려면 기본(또는 위쪽) 선반 관리 카드가 활성 선반 관리 카드여야 합니다. 또한 Netra CT 900 서버 미드프레인의 슬롯 7에 스위치 카드가 설치되어 있어야 합니다.

---

노드 보드에 콘솔 세션이 설정되면 `passwd`와 같은 시스템 관리 명령을 실행하거나, 상태와 오류 메시지를 읽거나, 특정 슬롯의 보드를 중지할 수 있습니다.

---

주 – 콘솔 또는 직렬 케이블이 노드 보드의 직렬 포트에 연결된 경우 케이블 연결 시 ShMM의 콘솔 세션이 활성 상태인 경우에도 콘솔 출력은 ShMM의 콘솔 세션이 아니라 케이블로 연결된 콘솔로 출력됩니다.

---

현재 콘솔 세션에서 분리하려면 `~q` 또는 `~.` (물결 기호 마침표)를 입력합니다.

예:

물리적 슬롯 4의 노드 보드에서 콘솔 세션을 시작합니다.

```
# clia console 4
prompt
```

---

## deactivate

구문:

```
deactivate IPMB-address fru-id
deactivate board n
deactivate shm n
```

용도:

이 명령은 지정된 FRU에 IPMI 명령 Set FRU Activation (Deactivate FRU)을 전송합니다. FRU는 소유 IPM 제어기의 IPMB 주소와 FRU 장치 ID를 사용하여 지정됩니다. FRU 장치 ID 0은 PICMG 3.0 컨텍스트에서 적합한 IPM 제어기를 지정합니다.

예:

주소 9C에서 적절한 IPM 제어기를 비활성화합니다.

```
# clia deactivate 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
Command issued via IPMB, status = 0 (0x0)
      Command executed successfully
#
```

# debuglevel

구문:

```
debuglevel [new-value]
```

용도:

이 명령은 IPM Sentry 선반 관리자의 현재 디버그 수준을 표시하거나, 새 값이 지정된 경우 새 값으로 설정합니다.

디버그 수준은 비트 마스크로 처리되는 0x0000부터 0x00FF 범위의 16진수입니다. 마스크의 각 비트는 설정된 경우 다음과 같이 특정 유형의 디버그 출력이 활성화됩니다.

- 0x0001 - 오류 메시지
- 0x0002 - 경고 메시지
- 0x0004 - 정보 메시지
- 0x0008 - 상세 정보 메시지
- 0x0010 - 추적 메시지
- 0x0020 - 상세 추적 메시지
- 0x0040 - 초기화 도중 IPM 제어기에 전송된 중요한 명령에 대해 표시되는 메시지
- 0x0080 - 내부 잠금 획득 및 해제에 대한 상세 메시지

선반 관리자의 기본 디버그 수준은 0x0007이지만 명령줄의 **-v** 옵션을 사용하여 선반 관리자 시작 도중 이 값을 대체할 수 있습니다. CLI는 런타임 도중 디버그 수준을 변경할 수 있는 추가 기능을 제공합니다.

예:

현재 디버그 수준을 가져온 다음 0x001F로 설정합니다.

```
# clia debuglevel
IPM Sentry Shelf Manager Command Line Interpreter
Debug Mask is 0x0007
# clia debuglevel 1f
IPM Sentry Shelf Manager Command Line Interpreter
# clia debuglevel
IPM Sentry Shelf Manager Command Line Interpreter
Debug Mask is 0x001f
```

---

## exit | quit

구문:

```
exit  
quit
```

용도:

명령 `exit` 또는 `quit`는 CLI 대화식 모드(매개 변수 없이 `clia`를 실행하여 들어감)를 종료합니다.

예:

```
# exit  
#
```

---

## fans

구문:

```
fans [-v] [IPMB-address [FRU-device-ID]]  
fans fan_tray n
```

용도:

이 명령은 지정된 팬 FRU에 대한 정보를 표시합니다. FRU 장치 ID가 생략된 경우 이 명령은 지정된 주소에 있는 IPM 제어기가 제어하는 모든 팬 FRU에 대한 정보를 표시합니다. IPMB 주소도 생략하면 이 명령은 선반 관리자가 인식하는 모든 팬 FRU에 대한 정보를 표시합니다. 다음 정보가 표시됩니다.

- IPMB 주소와 FRU 장치 ID
- 최소 속도 수준
- 최대 속도 수준
- 최대 지속 속도 수준
- 현재 수준(둘 모두 사용 가능한 경우 무효화 및 로컬 제어 수준)

예:

IPMB 주소 20의 모든 팬 FRU에 대한 팬 정보를 가져옵니다.

```
# clia fans 20
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 3
  Current Level: 6
  Minimum Speed Level: 0, Maximum Speed Level: 15
20: FRU # 4
  Current Level: 255 "Automatic"
  Minimum Speed Level: 0, Maximum Speed Level: 15
20: FRU # 5
  Current Level: 255 "Automatic"
  Minimum Speed Level: 0, Maximum Speed Level: 15
#
```

---

## flashupdate

구문:

```
flashupdate slot-number -s server-ip-address -f fw-image-pathname
```

용도:

이 명령은 Netra CP3060 보드 시스템 펌웨어를 사용자가 지정한 위치의 펌웨어 이미지로 업데이트할 때 사용합니다. 이 명령은 Netra CT 900 서버에 설치된 Netra CP3060 보드에 대해서만 유효합니다. 업데이트되는 Netra CP3060 보드 시스템 펌웨어에는 ALOM-CMT, Hypervisor, OBP, Post 및 VBSC 펌웨어가 포함됩니다.

---

주 - 다음 위치에서 Netra CP3060 펌웨어 다운로드 사이트에 대한 링크를 찾을 수 있습니다. <http://www.sun.com/downloads/>

---

이 명령을 사용하려면 다음 사항을 알고 있어야 합니다.

- 펌웨어 이미지를 다운로드하려는 FTP 서버의 IP 주소
- 프롬프트에 입력할 FTP 서버의 사용자 이름과 암호
- 이미지가 저장된 경로

*slot-number*에는 Netra CP3060 보드의 슬롯 번호가 포함되고, *-s server-ip-addresss* 인수는 펌웨어 이미지를 다운로드할 서버의 IP 주소를 지정하며, *-f fw-image-pathname*은 펌웨어 이미지가 있는 전체 경로 이름을 지정합니다.



예:

Netra CP3060 보드에서 시스템 펌웨어를 다운로드 및 업데이트합니다. 이 프로세스는 완료될 때까지 몇 분이 걸릴 수 있습니다. 성공적으로 완료되면 `boardreset` 명령을 사용하여 보드를 재설정합니다.

```
# clia flashupdate 2 -s 123.45.67.89
  -f /sysfw/System_Firmware-6_2_5-Netra_CP3060.bin
Username: username
Password: *****
.....
.....
.....
Update complete. Reset device to use new software.

# clia boardreset slot-number
```

---

## fru

구문:

```
fru [-v] [addr [id=fru_id | type=site_type]] | [type=site_type
[/site_number]]
fru board n
fru shm n
fru fan_tray n
```

용도:

이 명령은 특정 FRU에 대한 정보를 표시합니다. FRU 장치 ID가 생략된 경우 이 명령은 지정된 주소에 있는 IPM 제어기가 제어하는 모든 FRU에 대한 정보를 표시합니다. IPMB 주소도 생략하면 이 명령은 선반 관리자가 인식하는 모든 FRU에 대한 정보를 표시합니다.

또한 사이트 유형으로 FRU를 선택할 수 있습니다. 사이트 유형은 명령 매개 변수에 16진수로 지정해야 합니다. FRU와 해당 사이트 유형 간의 연결은 선반 FRU 정보에 저장됩니다. 사이트 유형은 PICMG 3.0 사양에 다음과 같이 정의되어 있습니다.

- 00h - AdvancedTCA 보드
- 01h - 전원 입력 모듈
- 02h - 선반 FRU 정보
- 03h - 전용 ShMC
- 04h - 팬 트레이
- 05h - 팬 필터 트레이

- 06h - 알람
- 07h - AdvancedTCA 모듈(중간 단계)
- 08h - PMC
- 09h - 후면 스위치오버 모듈
- C0h-CFh - 정의된 OEM
- 다른 모든 값은 예약되어 있습니다.

표준 모드에서는 FRU에 대해 다음 정보가 표시됩니다.

- IPMB 주소와 FRU 장치 ID
- 엔티티 ID, 엔티티 인스턴스
- 사이트 유형 및 번호(알려진 경우)
- FRU의 현재 핫스왑 상태, 이전 핫스왑 상태 및 마지막 상태 변경 원인. 핫스왑 상태 M0-M7은 PICMG 3.0 사양에 다음과 같이 정의되어 있습니다.
  - M0 - 설치되어 있지 않음
  - M1 - 비활성
  - M2 - 활성화 요청
  - M3 - 활성화 진행 중
  - M4 - FRU 활성화
  - M5 - 비활성화 요청
  - M6 - 비활성화 진행 중
  - M7 - 통신 끊김

FRU에 대한 다음 정보는 세부 정보 표시 모드에서만 표시됩니다.

- FRU 장치 유형, 장치 유형 수정자(FRU-device-ID != 0인 경우만). 이 정보는 FRU SDR(Sensor Data Record)에서 가져오며 IPMI 사양의 37.12절을 준수합니다.
- FRU SDR의 장치 ID 문자열
- 현재 FRU 전원 수준 및 최대 FRU 전원 수준, 현재 할당된 전원 소비량(와트)

예:

주소 9C의 모든 FRU에 대한 표준 정보를 가져옵니다.

```
# clia fru 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
#
```

주소 9C의 모든 FRU에 대한 상세 정보를 가져옵니다.

```
# clia fru -v 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
    Site Type: 0x00, Site Number: 14
    Current Power Level: 0x01, Maximum Power Level: 0x01, Current
Power Consumption: 20.0 Watts
#
```

주소 20의 FRU 1에 대한 상세 정보를 가져옵니다.

```
# clia fru -v 20 id=1
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1
    Entity: (0x1, 0x1)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device Type: "FRU Inventory Device behind management
controller" (0x10), Modifier 0x0
    Device ID String: "IPM Sentry 1.1"
    Current Power Level: UNKNOWN, Maximum Power Level: UNKNOWN,
Current Power Consumption: UNKNOWN
#
```

---

## frucontrol

구문:

`frucontrol IPMB-address fru-id option`

`frucontrol board n option`

`frucontrol shm n option`

`frucontrol fan_tray n option`

## 용도:

이 명령은 지정된 FRU에 FRU 제어 명령을 전송하여 FRU 페이로드에서 지정된 작업을 수행합니다. FRU는 소유 IPM 제어기의 IPMB 주소와 FRU 장치 ID를 사용하여 지정됩니다. FRU 장치 ID 0은 PICMG 3.0 컨텍스트에서 적합한 IPM 제어기를 지정합니다.

매개 변수 *option*은 사용할 FRU 제어 명령의 옵션을 지정합니다. 옵션은 다음 기호 값 중 하나로 지정할 수 있습니다.

- `cold_reset`(약어: `cr`) - FRU 페이로드의 콜드 재설정 수행
- `warm_reset`(약어: `wr`) - FRU 페이로드의 워م 재설정 수행
- `graceful_reboot`(약어: `gr`) - FRU 페이로드의 정상 재부트 수행
- `diagnostic_interrupt`(약어: `di`) - 진단 인터럽트 실행

## 예:

IPMB 주소 9C의 FRU 0에 콜드 재설정 명령을 실행합니다.

```
# clia frucontrol 9c 0 cr
IPM Sentry Shelf Manager Command Line Interpreter
    FRU Control: Controller 0x9c, FRU ID # 0, command 0x00, status
0 (0x0)
    Command executed successfully
#
```

---

# frudata

## 구문:

```
frudata [addr [fru_id [block_offset]]]
frudata addr fru_id byte_offset byte 1 [byte2 ... [byte 16] ...]
```

*addr fru\_id*는 다음의 항목으로 대체할 수 있습니다.

```
board n
shm n
fan_tray n
```

## 용도:

이 명령은 원시 형식의 FRU 정보에 대한 액세스를 제공합니다. 명령 형식에 따라 FRU 정보를 읽거나 쓰는 데 사용됩니다.

읽기 형식에서 명령은 선택적인 32바이트 블록 번호를 사용합니다.

쓰기 형식에서는 바이트 오프셋 매개 변수가 필요합니다. 사용자는 최대 65535바이트 까지 FRU 정보를 수정할 수 있습니다.

frudataw 및 frudatar는 frudata 명령의 변형입니다. frudataw를 사용하면 사용자는 ShMM 플래시 파일 시스템의 파일을 선반의 특정 FRU에 있는 FRU 정보 저장소에 쓸 수 있습니다(130페이지의 "frudatar" 참조). frudatar를 사용하면 사용자는 특정 FRU에 대한 FRU 정보 저장소의 내용을 ShMM 플래시 파일 시스템의 파일로 전송할 수 있습니다(130페이지의 "frudatar" 참조).

예:

모든 FRU에 대한 표준 FRU를 표시합니다.

```
# clia frudata
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 0    Failure status: 203 (0xcb)
    Requested data not present
20: FRU # 1 Raw FRU Info Data
    FRU Info size: 529
20: FRU # 2    Failure status: 203 (0xcb)
    Requested data not present
82: FRU # 0 Raw FRU Info Data
    FRU Info size: 160
9c: FRU # 0 Raw FRU Info Data
    FRU Info size: 160
fc: FRU # 0 Raw FRU Info Data
    Requested data not present
. . .
#
```

이 예에서는 FRU 데이터를 표시하고 FRU에 데이터를 쓰는 방법을 나타냅니다.

```
# clia frudata 20 1 0
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1 Block # 0 Raw FRU Info Data
    FRU Info size: 529
    01 00 01 05  0E 18 00 D3  01 04 01 02  55 AA 83 55
    AA 55 C1 00  00 00 00 00  00 00 00 00  00 00 00 00
#
# clia frudata 20 1 1 0xfc 0xfe
IPM Sentry Shelf Manager Command Line Interpreter
Writing 2 bytes to IPM 0x20, FRU # 1, offset: 1, status = 0(0x0)
#
# clia frudata 20 1 0
```

```

IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1 Block # 0 Raw FRU Info Data
    FRU Info size: 529
    01 FC FE 05 0E 18 00 D3 01 04 01 02 55 AA 83 55
    AA 55 C1 00 00 00 00 00 00 00 00 00 00 00 00 00
#
# clia frudata 20 1 1 0 1
IPM Sentry Shelf Manager Command Line Interpreter
Writing 2 bytes to IPM 0x20, FRU # 1, offset: 1, status = 0(0x0)
#
# clia frudata 20 1 0
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1 Block # 0 Raw FRU Info Data
    FRU Info size: 529
    01 00 01 05 0E 18 00 D3 01 04 01 02 55 AA 83 55
    AA 55 C1 00 00 00 00 00 00 00 00 00 00 00 00 00
#

```

## frudatar

구문:

```

frudatar addr fru_id file_name
frudatar addr fru_id byte_offset byte 1 [byte2 ... [byte 16] ...]

```

*addr fru\_id*는 다음의 항목으로 대체할 수 있습니다.

```

board n
shm n
fan_tray n

```

용도:

이 명령은 지정된 FRU에서 FRU 정보를 읽어서 ShMM 플래시 파일 시스템의 파일에 원시 형식으로 저장합니다. 즉, 지정된 FRU의 FRU 정보를 플래시 파일로 업로드합니다. 매개 변수 *file name*은 대상 파일에 대한 경로를 지정합니다. FRU에서 읽고 대상 파일에 쓰는 바이트 수는 지정된 FRU에 대한 IPMI 명령 Get FRU Inventory Area Info에 대한 응답으로 반환되는 바이트 수와 같습니다.

예:

특정 FRU에 대한 FRU 데이터를 읽고 명명된 파일에 데이터를 저장합니다.

```
# clia frudatar 20 2 /var/tmp/20.2.bin
IPM Sentry Shelf Manager Command Line Interpreter

20: FRU # 2 Raw FRU Info Data
FRU Info size: 176
01 00 00 01 09 00 00 F5 01 08 19 84 C0 42 C7 53
63 68 72 6F 66 66 D9 53 68 4D 4D 2D 41 43 42 2D
46 43 20 53 68 65 6C 66 20 4D 61 6E 61 67 65 72
86 10 04 41 10 14 01 89 D2 04 65 58 13 51 17 00
00 C0 C1 00 00 00 00 EA 01 0D 19 C7 53 63 68 72
6F 66 66 DD 46 61 6E 20 43 6F 6E 74 72 6F 6C 6C
65 72 20 6F 6E 20 53 68 4D 4D 2D 41 43 42 2D 46
43 89 D2 04 65 58 13 51 17 00 00 C9 52 65 76 2E
20 31 2E 30 30 86 10 04 41 10 14 01 C0 DF 2F 76
61 72 2F 6E 76 64 61 74 61 2F 66 61 6E 2D 66 72
75 2D 69 6E 66 6F 72 6D 61 74 69 6F 6E C1 00 26
#
```

---

## frudataw

구문:

```
frudataw addr fru_id file_name
frudataw addr fru_id byte_offset byte 1 [byte2 ... [byte 16] ...]
```

*addr fru\_id*는 다음의 항목으로 대체할 수 있습니다.

```
board n
shm n
fan_tray n
```

용도:

이 명령은 FRU 정보를 ShMM 플래시 파일 시스템의 파일에서 지정된 FRU로 다운로드합니다. 파일에는 FRU 정보의 원시 바이너리 이미지가 포함되어 있습니다. 매개 변수 *file name*은 소스 파일에 대한 경로를 지정합니다.

예:

파일의 FRU 데이터를 명명된 FRU에 씁니다.

```
# clia frudataw 20 2 /var/tmp/20.2.bin
IPM Sentry Shelf Manager Command Line Interpreter

Writing 16 bytes to IPM 0x20, FRU # 2, offset: 0, status = 0(0x0)
Writing 16 bytes to IPM 0x20, FRU # 2, offset: 16, status = 0(0x0)
Writing 16 bytes to IPM 0x20, FRU # 2, offset: 32, status = 0(0x0)
Writing 16 bytes to IPM 0x20, FRU # 2, offset: 48, status = 0(0x0)
Writing 16 bytes to IPM 0x20, FRU # 2, offset: 64, status = 0(0x0)
Writing 16 bytes to IPM 0x20, FRU # 2, offset: 80, status = 0(0x0)
Writing 16 bytes to IPM 0x20, FRU # 2, offset: 96, status = 0(0x0)
Writing 16 bytes to IPM 0x20, FRU # 2, offset: 112, status = 0(0x0)
Writing 16 bytes to IPM 0x20, FRU # 2, offset: 128, status = 0(0x0)
Writing 16 bytes to IPM 0x20, FRU # 2, offset: 144, status = 0(0x0)
Writing 16 bytes to IPM 0x20, FRU # 2, offset: 160, status = 0(0x0)
File "/var/tmp/20.2.orig.bin" has been written to the FRU 20#2
#
```

---

## fruinfo

구문:

```
fruinfo [-v] [-x] addr fru_id
```

*addr fru\_id*는 다음의 항목으로 대체할 수 있습니다.

```
board n
shm n
fan_tray n
```

용도:

이 명령은 FRU 정보를 사용자가 읽기 쉬운 형식으로 표시합니다.



예:

특정 FRU에 대한 FRU 정보를 표시합니다.

```
# clia fruinfo 20 1
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1, FRU Info
Common Header:      Format Version = 1
Chassis Info Area:
    Version      = 1
    Chassis Type          = (1)
    Chassis Part Number   = 0x55 0xAA
    Chassis Serial Number = 5I:5
Board Info Area:
    Version      = 1
Mfg Date/Time      = Jun 16 15:37:00 2011 (8129737 minutes
since 1996)
Board Manufacturer  = Pigeon Point Systems
Board Serial Number = PPS0000000
    Board Part Number      = A
    FRU Programmer File ID =
Product Info Area:
    Version      = 1
    Language Code          = 25
    Manufacturer Name      = Pigeon Point Systems
    Product Name           = Shelf Manager
    Product Part / Model#  = 000000
    Product Version        = Rev. 1.00
    Product Serial Number  = PPS00000000
    Asset Tag              =
    FRU Programmer File ID =
Multi Record Area:
    Record Type              = Management Access Record
        Version = 2
    Sub-Record Type: Component Name (0x05)
    PICMG Address Table Record (ID=0x10)
        Version = 1
    PICMG Backplane Point-to-Point Connectivity Record (ID=0x04)
        Version = 0
    PICMG Shelf Power Distribution Record (ID=0x11)
        Version = 0
    PICMG Shelf Activation And Power Management Record (ID=0x12)
        Version = 0
#
```

---

## getfanlevel

구문:

```
getfanlevel IPMB-address fru-id  
getfanlevel fan_tray n
```

용도:

이 명령은 명령 매개 변수에 지정된 FRU가 제어하는 팬의 현재 수준을 표시합니다.

예:

IPMB 주소 0x20의 FRU #2에 있는 팬의 팬 수준을 가져옵니다.

```
# clia getfanlevel 20 2  
IPM Sentry Shelf Manager Command Line Interpreter  
20: FRU # 2 Override Fan Level: 1, Local Fan Level: 255  
#
```

---

## getfruledstate

구문:

```
getfruledstate [-v] [IPMB-addr state [fru_id [LED_ID | ALL]]]
```

용도:

이 명령은 LED에 대해 활성화된 제어의 모든 수준에 대한 현재 FRU LED 상태를 표시합니다. 세부 정보 표시 모드에서는 LED에서 지원되는 색상에 대한 정보도 표시됩니다.

특정 LED 또는 지정된 FRU에 대한 모든 LED 정보가 표시될 수 있습니다. 대상 LED의 IPMB 주소 및 FRU ID를 생략할 수도 있습니다. FRU ID를 생략하면 지정된 IPM 제어기의 모든 FRU에 있는 모든 LED에 대한 정보가 표시됩니다. IPMB 주소를 생략하면 선반에 있는 모든 알려진 LED에 대한 정보가 표시됩니다.

예:

IPMB 주소 FCh의 IPM 제어기에 있는 모든 LED에 대한 LED 상태를 표시합니다.

```
# clia getfruLEDstate fc
IPM Sentry Shelf Manager Command Line Interpreter

fc: FRU # 0, Led # 0 ("BLUE LED"):
    Local Control LED State: LED OFF

fc: FRU # 0, Led # 1 ("LED 1"):
    Local Control LED State: LED OFF

fc: FRU # 0, Led # 2 ("LED 2"):
    Local Control LED State: LED OFF

fc: FRU # 0, Led # 3 ("LED 3"):
    Local Control LED State: LED OFF

fc: FRU # 0, Led # 4 ("Application Specific LED# 1"):
    Local Control LED State: LED ON, color: GREEN
```

IPMB 주소 FCh의 IPM 제어기에 있는 모든 LED에 대한 LED 상태를 표시합니다.

```
# clia getfruLEDstate -v FC
IPM Sentry Shelf Manager Command Line Interpreter

fc: FRU # 0, Led # 0 ("BLUE LED"):
    Local Control LED State: LED OFF
    LED's color capabilities:
        Colors supported(0x02): BLUE
        Default LED Color in Local Control State(0x01): BLUE
        Default LED Color in Override State(0x01): BLUE

fc: FRU # 0, Led # 1 ("LED 1"):
    Local Control LED State: LED OFF
    LED's color capabilities:
        Colors supported(0x0C): RED GREEN
        Default LED Color in Local Control State(0x03): GREEN
        Default LED Color in Override State(0x03): GREEN

fc: FRU # 0, Led # 2 ("LED 2"):
    Local Control LED State: LED OFF
    LED's color capabilities:
        Colors supported(0x0C): RED GREEN
```

```

Default LED Color in Local Control State(0x03): GREEN
Default LED Color in Override State(0x03): GREEN

fc: FRU # 0, Led # 3 ("LED 3"):
  Local Control LED State: LED OFF
  LED's color capabilities:
    Colors supported(0x0C): RED GREEN
    Default LED Color in Local Control State(0x02): RED
    Default LED Color in Override State(0x02): RED

fc: FRU # 0, Led # 4 ("Application Specific LED# 1"):
  Local Control LED State: LED ON, color: GREEN
  LED's color capabilities:
    Colors supported(0x0C): RED GREEN
    Default LED Color in Local Control State(0x02): RED
    Default LED Color in Override State(0x02): RED

```

IPMB 주소 20h에 있는 IPM 제어기의 FRU #0에 대한 LED 상태를 표시합니다.

```

# clia getfruLEDstate 20 0
IPM Sentry Shelf Manager Command Line Interpreter

20: FRU # 0, Led # 0 ("BLUE LED"):
  Local Control LED State: LED ON, color: BLUE

20: FRU # 0, Led # 1 ("LED 1"):
  Local Control LED State: LED OFF

```

IPMB 주소 20h에 있는 IPM 제어기의 FRU #0에서 LED #1에 대한 LED 상태를 표시합니다.

```

# clia getfruLEDstate -v 20 0 1
IPM Sentry Shelf Manager Command Line Interpreter

20: FRU # 0, Led # 1 ("LED 1"):
  Local Control LED State: LED OFF
  LED's color capabilities:
    Colors supported(0x04): RED
    Default LED Color in Local Control State(0x02): RED
    Default LED Color in Override State(0x02): RED

```

---

# gethysteresis

구문:

```
gethysteresis [IPMB-address [ [lun: ] sensor id | sensor name ]]
```

용도:

이 명령은 지정된 센서의 현재 히스테리시스 값을 표시합니다. 센서는 임계값 기반이어야 합니다. 원시 값과 처리된 값이 모두 표시됩니다.

대상 제어기가 여러 LUN의 센서를 지원하는 경우 사용자는 이 명령을 사용하여 센서 번호를 LUN(Logical Unit Number)으로 한정할 수 있습니다. LUN이 생략된 경우 지정된 센서 번호를 가진 모든 센서의 현재 히스테리시스 값이 표시됩니다. *lun*에는 값 0, 1 또는 3을 사용할 수 있습니다. LUN 2는 예약되어 있습니다. 센서 이름은 일반적으로 제어기 내에서 고유한 것으로 가정되기 때문에 LUN 번호로 한정되지 않습니다. 하지만 제어기 내에 이름이 같은 센서가 여러 개 있는 경우 모든 센서에 대한 정보가 표시됩니다. *IPMB-address*를 생략하면 지정된 IPMB 주소의 모든 센서에 대한 현재 히스테리시스 수준이 표시됩니다.

예:

IPMB 주소 FCh의 IPM 제어기에 있는 센서 #2에 대한 히스테리시스 값을 표시합니다.

```
# clia gethysteresis FC 2
IPM Sentry Shelf Manager Command Line Interpreter

fc: LUN: 0, Sensor # 2 ("lm75 temp")
    Type: Threshold (0x01), "Temperature"(0x01)
        Positive hysteresis, Raw data: 0x00    Processed data:
0.00000 degrees C
        Negative hysteresis, Raw data: 0x00    Processed data:
0.00000 degrees C
```

# getipmbstate

구문:

getipmbstate *IPMB-address* [*link*] (방사형 IPMB-0 환경에서)  
getipmbstate *IPMB-address* (버스된 IPMB-0 환경에서)

용도:

이 명령은 대상 IPM 제어기에 있는 IPMB-0의 현재 상태를 표시합니다. 상태는 대상 IPMC의 IPMB 링크 센서(센서 유형 F1)가 제공하는 센서 데이터로부터 가져옵니다. 버스 A와 B에 대한 정보가 모두 표시됩니다.

이 명령은 버스된 환경과 방사형 환경에서 서로 다르게 작동합니다. 버스된 환경 또는 대상 IPMC가 IPMB 허브가 아닌 경우 방사형 환경에서는 인수 *link*가 사용되지 않습니다. 대상 IPM 제어기에 있는 IPMB-A 및 IPMB-B의 상태에 대한 정보가 표시됩니다.

방사형 환경에서 대상 IPM 제어기가 IPMB 허브이면 명령이 다음과 같이 작동합니다.

- *link*를 생략하면 명령은 모든 방사형 IPMB 링크의 상태에 대한 정보를 표시합니다. 상태는 IPM 제어기에 있는 여러 IPMB 링크 센서의 센서 데이터에서 가져옵니다.
- *link*가 있으면 명령은 특정 방사형 IPMB 링크(1에서 95까지)에 대한 정보를 표시합니다. 링크 상태는 IPM 제어기에서 해당 IPMB 링크 센서의 상태에서 가져옵니다.

두 경우 모두 IPMB-A와 IPMB-B 모두에 대한 상태 정보가 표시됩니다.

예:

IPMB 주소 92h에서 IPMC에 있는 IPMB-0의 현재 상태를 표시합니다.

```
# clia getipmbstate 92
IPM Sentry Shelf Manager Command Line Interpreter

92: LUN: 0, Sensor # 1 ("IPMB LINK")
   Bus Status: 0x8  (IPMB-A Enabled, IPMB-B Enabled)
   IPMB A State: 0x8  (LocalControl, No failure)
   IPMB B State: 0x8  (LocalControl, No failure)
```

방사형 환경에서 선반 관리자에 대한 링크 8의 현재 상태를 표시합니다.

```
# clia getipmbstate 20 8
IPM Sentry Shelf Manager Command Line Interpreter

20: Link: 8, LUN: 0, Sensor # 12 ("IPMB LINK 8")
```

```
Bus Status: 0x8 (IPMB-A Enabled, IPMB-B Enabled)
IPMB A State: 0x8 (LocalControl, No failure)
IPMB B State: 0x8 (LocalControl, No failure)
```

## getlanconfig

구문:

```
getlanconfig channel [parameter-name [additional-parameters]]
getlanconfig channel [parameter-number [additional-parameters]]
```

용도:

이 명령은 지정된 채널에서 지정된 LAN 구성 매개 변수의 값을 표시합니다. 구성 매개 변수 이름이나 번호를 지정하지 않으면 지정된 채널의 모든 구성 매개 변수가 표시됩니다.

[표 A-1](#)에서는 getlanconfig 명령이 지원하는 LAN 구성 매개 변수의 이름과 번호를 나열합니다.

표 A-1 getlanconfig의 LAN 구성 매개 변수

매개 변수 이름	번호	설명
auth_support	1	LAN 채널에 대한 인증 유형 지원 플래그가 포함된 8비트 값입니다.
auth_enables	2	LAN 채널의 콜백, 사용자, 운영자, 관리자 및 OEM 권한 수준에 대한 인증 유형 활성화 플래그가 포함된 5개의 8비트 값입니다.
ip	3	LAN 채널에 점으로 구분된 십진수 표기법으로 할당된 IP 주소(예: 192.168.0.15)가 포함된 문자열 값입니다.
ip_source	4	할당된 IP 주소의 소스를 인코딩하는 값입니다.
mac	5	LAN 채널에 : 기호로 구분된 6개의 16진수 바이트 값으로 할당된 MAC 주소(예: 00:A0:24:C6:18:2F)가 포함된 문자열 값입니다.
subnet_mask	6	LAN 채널에 점으로 구분된 십진수 표기법으로 할당된 서브넷 마스크(예: 255.255.255.0)가 포함된 문자열 값입니다.

표 A-1 getlanconfig의 LAN 구성 매개 변수(계속)

ipv4_hdr_param	7	RMCP 패킷 전송을 위한 다양한 IPv4 헤더 매개 변수가 포함된 3개의 8비트 값입니다. • TTL(Time-to-live) • IP 헤더 플래그(비트 [7:5]) • 우선 순위(비트 [7:5]) 및 서비스 유형(비트 [4:1])
pri_rmcp_port	8	기본 RMCP 포트 번호(일반 RMCP 통신에 사용되는 포트)가 포함된 16비트 값입니다.
sec_rmcp_port	9	보조 RMCP 포트 번호(보안 RMCP 통신에 사용되는 포트)가 포함된 16비트 값입니다.
arp_control	10	LAN 채널에서 주소 해석 프로토콜(ARP) 동작을 제어하는 다음 2개의 플래그입니다. • ARP 요청에 대한 응답 활성화 • 불필요한 ARP 전송 활성화
arp_interval	11	고정 소수점 형식의 초 단위 불필요한 ARP 간격입니다 (분수 부분 포함 가능).
dft_gw_ip	12	점으로 구분된 십진수 표기법으로 표현한 기본 게이트웨이의 IP 주소가 포함된 문자열 값입니다.
dft_gw_mac	13	콜론(:)으로 구분된 6개의 16진수 바이트 값으로 표현한 기본 게이트웨이의 MAC 주소가 포함된 문자열 값입니다.
backup_gw_ip	14	점으로 구분된 십진수 표기법으로 표현한 백업 게이트웨이의 IP 주소가 포함된 문자열 값입니다.
backup_gw_mac	15	콜론(:)으로 구분된 6개의 16진수 바이트 값으로 표현한 백업 게이트웨이의 MAC 주소가 포함된 문자열 값입니다.
community	16	PET 트랩의 Community String 필드에 저장되는 문자열 값(최대 18개의 기호)입니다.
destination_count	17	LAN 채널에서 지원되는 LAN 경고 대상의 최대 수입니다.
destination_type	18	지정된 세트 선택기로 식별되는 대상 유형입니다. 세트 선택기를 지정하지 않으면 모든 대상 유형이 표시됩니다. 각 대상 유형 항목에는 다음 필드가 포함됩니다. • 대상 유형(0-7) • 경고 확인 플래그 • 경고 확인 시간 초과/재시도 간격(1-256초) • 재시도 횟수(0-7)
destination_address	19	지정된 세트 선택기와 연관된 대상 주소입니다. 세트 선택기를 지정하지 않으면 모든 대상 주소가 표시됩니다. 각 대상 주소 항목에는 다음 필드가 포함됩니다. • 게이트웨이 선택기: 0 - 기본 사용, 1 - 백업 사용 • IP 주소(점으로 구분된 십진수 형식의 문자열) • MAC 주소(콜론[:]으로 구분된 6개의 16진수 바이트 값 문자열)



예:

다음 세부절에서는 지원되는 각 매개 변수에 대해 자세히 설명합니다.

채널 1에 대한 LAN 매개 변수 테이블을 가져와서 표시합니다.

```
# clia getlanconfig 1

Authentication Type Support: 0x15 (None MD5 Straight Password/Key)
Authentication Type Enables: 0x00
    User level: 0x15 ( None MD5 Straight Password/Key )
    Operator level: 0x15 ( None MD5 Straight Password/Key )
    Administrator level: 0x15 ( None MD5 Straight Password/Key )
    OEM level: 0x00
IP Address: 172.16.2.203
IP Address Source: Static Address (Manually Configured) (01)
MAC Address: 90:91:91:91:91:91
Subnet Mask: 255.255.255.0
IPv4 Header Parameters: 0x40:0x40:0x10
Primary RMCP Port Number: 0x026f
Secondary RMCP Port Number: 0x0298
BMC-generated ARP Control: 02
    Enable BMC-generated Gratuitous Response
Gratuitous ARP Interval: 2.0 seconds
Default Gateway Address: 0.0.0.0
Default Gateway MAC Address: N/A
Backup Gateway Address: 0.0.0.0
Backup MAC Address: N/A
Community String: "public"
Number of Destinations: 16
#
```

## auth\_support

구문:

```
getlanconfig channel auth_support
getlanconfig channel 1
```

용도:

이 명령은 LAN 매개 변수 `auth_support`의 현재 값을 표시합니다. 이 매개 변수는 선반 관리자가 지원하는 인증 유형을 지정하고 단일 바이트로 표현되며 아래와 같이 정의되는 비트로 된 비트 마스크로서 처리됩니다.

- 0x01 - 없음
- 0x02 - MD2
- 0x04 - MD5
- 0x10 - 일반 암호/키
- 0x20 - OEM 소유

다른 비트는 예약되어 있으며 0으로 설정해야 합니다.

원시 16진수 값 이외에도, 설정된 비트에 대한 기호 값도 표시됩니다.

예:

```
# clia getlanconfig 1 auth_support
IPM Sentry Shelf Manager Command Line Interpreter
Authentication Type Support: 0x15 ( None MD5 Straight Password/Key
#
```

## auth\_enables

구문:

```
getlanconfig channel auth_enables
getlanconfig channel 2
```

용도:

이 명령은 LAN 매개 변수 `auth_enables`의 현재 값을 표시합니다. 이 매개 변수는 5개의 지원되는 권한 수준(콜백, 사용자, 관리자, 운영자 및 OEM) 각각에 대해 선반 관리자가 현재 활성화한 인증 유형을 지정합니다. 이 유형은 5바이트 시퀀스로 표현되며 각 바이트가 각각의 권한 수준에 해당하며 아래와 같이 정의되는 비트로 된 비트 마스크로서 처리됩니다.

- 0x01 - 없음
- 0x02 - MD2
- 0x04 - MD5
- 0x10 - 일반 암호/키
- 0x20 - OEM 소유

다른 비트는 예약되어 있으며 0으로 설정해야 합니다.

원시 16진수 값 이외에도, 설정된 비트의 기호 값도 표시됩니다.

예:

```
# clia getlanconfig 1 auth_enables
IPM Sentry Shelf Manager Command Line Interpreter
Authentication Type Enables:
    Callback level: 0x00
    User level: 0x15 ( None MD5 Straight Password/Key )
    Operator level: 0x15 ( None MD5 Straight Password/Key )
    Administrator level: 0x15 ( None MD5 Straight Password/Key )
    OEM level: 0x00
```

## ip

구문:

```
getlanconfig channel ip
getlanconfig channel 3
```

용도:

이 명령은 채널이 사용하는 현재 IP 주소를 점으로 구분된 십진수 표기법으로 표시합니다.

예:

```
# clia getlanconfig 1 ip
IPM Sentry Shelf Manager Command Line Interpreter
IP Address: 172.16.2.203
```

## ip\_source

구문:

```
getlanconfig channel ip_source
getlanconfig channel 4
```

용도:

이 명령은 LAN 매개 변수 `ip_source`의 현재 값을 표시합니다. 이 매개 변수는 선반 관리자가 사용하는 IP 주소의 소스를 지정합니다. 이 소스는 단일 바이트로 표현되며 다음 값 중 하나일 수 있습니다.

- 0 - 지정되지 않음
- 1 - 정적 주소(수동으로 구성됨)

- 2 - DHCP를 실행하는 선반 관리자가 가져온 주소
- 3 - BIOS 또는 시스템 소프트웨어가 로드한 주소
- 4 - 다른 주소 할당 프로토콜을 실행하는 선반 관리자가 가져온 주소

다른 값은 예약되어 있습니다.

원시 16진수 값 이외에 기호 값도 표시됩니다.

예:

```
# clia getlanconfig 1 ip_source
IPM Sentry Shelf Manager Command Line Interpreter
IP Address Source: Static Address (Manually Configured) (0x01)
#
```

## mac

구문:

```
getlanconfig channel mac
getlanconfig channel 5
```

용도:

이 명령은 채널이 사용하는 현재 MAC 주소를 콜론으로 구분된 6개의 16진수 바이트 형식으로 표시합니다.

예:

```
# clia getlanconfig 1 mac
IPM Sentry Shelf Manager Command Line Interpreter
MAC Address: 90:91:91:91:91:91
#
```

## subnet\_mask

구문:

```
getlanconfig channel subnet_mask  
getlanconfig channel 6
```

용도:

이 명령은 채널이 사용하는 현재 IP 서브넷 마스크를 점으로 구분된 십진수 표기법으로 표시합니다.

예:

```
# clia getlanconfig 1 subnet_mask  
IPM Sentry Shelf Manager Command Line Interpreter  
Subnet Mask: 255.255.255.0  
#
```

## ipv4\_hdr\_param

구문:

```
getlanconfig channel ipv4_hdr_param  
getlanconfig channel 7
```

용도:

이 명령은 현재 IP 4 헤더 매개 변수를 표시합니다. 콜론으로 구분된 3개의 단일 바이트 값이 16진수 표기법으로 표현됩니다. 바이트의 내용은 IPMI 1.5 사양의 19.2절을 준수합니다.

예:

```
# clia getlanconfig 1 ipv4_hdr_param  
IPM Sentry Shelf Manager Command Line Interpreter  
IPv4 Header Parameters: 0x40:0x40:0x10  
#
```

## pri\_rmcp\_port

구문:

```
getlanconfig channel pri_rmcp_port  
getlanconfig channel 8
```

용도:

이 명령은 채널에 사용되는 현재 RMCP 기본 포트를 16진수로 표시합니다. 이 포트는 RMCP를 통한 일반 상호 작용에 사용됩니다.

예:

```
# clia getlanconfig 1 pri_rmcp_port  
IPM Sentry Shelf Manager Command Line Interpreter  
Primary RMCP Port Number: 0x026f  
#
```

## sec\_rmcp\_port

구문:

```
getlanconfig channel sec_rmcp_port  
getlanconfig channel 9
```

용도:

이 명령은 채널에 사용되는 현재 RMCP 보조 포트를 16진수로 표시합니다. 이 포트는 RMCP를 통한 보안 상호 작용에 사용됩니다.

예:

```
# clia getlanconfig 1 sec_rmcp_port  
IPM Sentry Shelf Manager Command Line Interpreter  
Primary RMCP Port Number: 0x0298  
#
```

## arp\_control

구문:

```
getlanconfig channel arp_control  
getlanconfig channel 10
```

용도:

이 명령은 LAN 매개 변수 `arp_control`의 현재 값을 표시합니다. 이 매개 변수는 선반 관리자가 제공하는 추가 ARP 지원을 지정하고 단일 바이트로 표현되며 아래와 같이 정의되는 비트로 된 비트 마스크로서 처리됩니다.

- 1 - 선반 관리자가 생성한 불필요한 ARP 활성화
- 2 - 선반 관리자가 생성한 ARP 응답 활성화

다른 비트는 예약되어 있으며 0으로 설정해야 합니다.

원시 16진수 값 이외에도, 설정된 비트에 대한 기호 값도 표시됩니다.

예:

```
# clia getlanconfig 1 arp_control  
IPM Sentry Shelf Manager Command Line Interpreter  
BMC-generated ARP Control: 02  
    Enable BMC-generated Gratuitous Response  
#
```

## arp\_interval

구문:

```
getlanconfig channel arp_interval  
getlanconfig channel 11
```

용도:

이 명령은 채널에 사용되는 현재 ARP 간격을 표시합니다. 값은 고정 소수점 숫자 형식의 숫자(초)로 표시됩니다.

예:

```
# clia getlanconfig 1 arp_interval  
IPM Sentry Shelf Manager Command Line Interpreter  
Gratuitous ARP Interval: 2.0 seconds  
#
```

## dft\_gw\_ip

구문:

```
getlanconfig channel dft_gw_ip  
getlanconfig channel 12
```

용도:

이 명령은 채널이 사용하는 기본 게이트웨이의 IP 주소를 점으로 구분된 십진수 표기법으로 표시합니다.

예:

```
# clia getlanconfig 1 dft_gw_ip  
IPM Sentry Shelf Manager Command Line Interpreter  
Default Gateway Address: 0.0.0.0  
#
```

## dft\_gw\_mac

구문:

```
getlanconfig channel dft_gw_mac  
getlanconfig channel 13
```

용도:

이 명령은 채널이 사용하는 기본 게이트웨이의 MAC 주소를 콜론으로 구분된 6개의 16진수 바이트 형식으로 표시합니다.

예:

```
# clia getlanconfig 1 dft_gw_mac  
IPM Sentry Shelf Manager Command Line Interpreter  
Default Gateway MAC Address: N/A  
#
```



## backup\_gw\_ip

구문:

```
getlanconfig channel backup_gw_ip  
getlanconfig channel 14
```

용도:

이 명령은 채널이 사용하는 백업 게이트웨이의 IP 주소를 점으로 구분된 십진수 표기법으로 표시합니다.

예:

```
# clia getlanconfig 1 backup_gw_ip  
IPM Sentry Shelf Manager Command Line Interpreter  
Backup Gateway Address: 0.0.0.0  
#
```

## backup\_gw\_mac

구문:

```
getlanconfig channel backup_gw_mac  
getlanconfig channel 15
```

용도:

이 명령은 채널이 사용하는 백업 게이트웨이의 MAC 주소를 콜론으로 구분된 6개의 16진수 바이트 형식으로 표시합니다.

예:

```
# clia getlanconfig 1 backup_gw_mac  
IPM Sentry Shelf Manager Command Line Interpreter  
Backup Gateway MAC Address: N/A  
#
```

## community

구문:

```
getlanconfig channel community  
getlanconfig channel 16
```

용도:

이 명령은 PET 트랩에서 사용되는 커뮤니티 문자열 매개 변수를 표시합니다.

예:

```
# clia getlanconfig 1 community  
IPM Sentry Shelf Manager Command Line Interpreter  
Community String: "public"  
#
```

## destination\_count

구문:

```
getlanconfig channel destination_count  
getlanconfig channel 17
```

용도:

이 명령은 채널에 사용 가능한 경고 대상의 최대 수를 표시합니다. 이것은 IPM Sentry 선반 관리자에 대한 구성 매개 변수이며 `shelfman` 구성 파일을 통해서만 변경할 수 있습니다.

예:

```
# clia getlanconfig 1 destination_count  
IPM Sentry Shelf Manager Command Line Interpreter  
Number of Destinations: 16  
#
```

## destination\_type

구문:

```
getlanconfig channel destination_type [set-selector]  
getlanconfig channel 18 [set-selector]
```

용도:

이 명령은 대상 테이블의 요소를 *set-selector*와 동일한 색인과 함께 표시합니다. 색인은 0부터 시작됩니다. 선택기 0은 휘발성 대상의 주소 지정에 사용됩니다. 대상에 대해 다음과 같은 정보가 표시됩니다.

- 대상 선택기
- 경고 대상 유형(PET 트랩 또는 OEM 대상, 경고가 확인되어야 하는지 여부)
- 경고 확인 시간 초과
- 재시도 수

세트 선택기를 생략하면 모든 활성 대상이 번호와 함께 표시됩니다.

예:

```
# clia getlanconfig 1 destination_type 2  
IPM Sentry Shelf Manager Command Line Interpreter  
DST Type # 2, Type: Acknowledged PET Trap Destination (0x80), ACK  
Timeout / Retry Interval: 3 seconds, Retries: 5  
  
# clia getlanconfig 1 destination_type  
IPM Sentry Shelf Manager Command Line Interpreter  
DST Type # 0, Type: Acknowledged reserved (0x81), ACK Timeout /  
Retry Interval: 2 seconds, Retries: 6  
DST Type # 1, Type: Unacknowledged reserved (0x02), ACK Timeout /  
Retry Interval: 3 seconds, Retries: 4  
DST Type # 2, Type: Acknowledged PET Trap Destination (0x80), ACK  
Timeout / Retry Interval: 3 seconds, Retries: 5  
#
```

## destination\_address

구문:

```
getlanconfig channel destination_address [set-selector] getlanconfig  
channel 19 [set-selector]
```

용도:

이 명령은 대상 주소 테이블의 요소를 *set-selector*와 동일한 색인과 함께 표시합니다. 색인은 0부터 시작됩니다. 선택기 0은 휘발성 대상의 주소 지정에 사용됩니다. 대상에 대해 다음과 같은 정보가 표시됩니다.

- 대상 선택기
- 주소 형식(기본적으로 IP+MAC)
- 대상 IP 주소
- 대상 MAC 주소
- 사용할 게이트웨이(기본 또는 백업)

세트 선택기를 생략하면 모든 활성 대상 주소가 번호와 함께 표시됩니다.

예:

```
# clia getlanconfig 1 destination_address 2  
IPM Sentry Shelf Manager Command Line Interpreter  
DST Addresses # 2, Address Format: IPv4 IP Address followed by DIX  
ethernet / 802.3 MAC Address (0x00)  
    Gateway: Default (0x00), Alerting IP: 172.16.2.100, Alerting  
MAC: 90:93:93:93:93:93  
#
```

---

## getpefconfig

구문:

```
getpefconfig  
getpefconfig parameter-name [additional-parameters]  
getpefconfig parameter-number [additional-parameters]
```

용도:

이 명령은 지정된 PEF 구성 매개 변수의 값을 표시합니다. 구성 매개 변수 이름이나 매개 변수 번호를 지정하지 않으면 모든 PEF 구성 매개 변수가 표시됩니다.

표 A-2에서는 PEF 구성 매개 변수의 이름 및 번호를 나열합니다.

표 A-2 PEF 구성 매개 변수

매개 변수 이름	번호	설명
control	1	PEF에 대한 제어 플래그(예: enable PEF, enable PEF startup delay 등)를 나타내는 8비트 값입니다.
action_control	2	PEF 작업 전역 제어 플래그(예: enable reset, enable power down 등)를 나타내는 8비트 값입니다.
startup_delay	3	시스템 전원 켜기 및 재설정 후 PEF를 지연할 시간(초)입니다.
alert_startup_delay	4	시스템 전원 켜기 및 재설정 후 경고를 지연할 시간(초)입니다.
event_filter_count	5	이벤트 필터의 최대 수입니다.
event_filter	6	지정된 세트 선택기로 식별되는 이벤트 필터 테이블입니다. 세트 선택기를 지정하지 않으면 모든 활성 이벤트 필터가 표시됩니다.
event_filter_data1	7	지정된 세트 선택기로 식별되는 이벤트 필터 테이블의 첫 번째 바이트입니다. 세트 선택기를 지정하지 않으면 모든 활성 이벤트 필터가 표시됩니다.
alert_policy_count	8	경고 정책의 최대 수입니다.
alert_policy	9	지정된 세트 선택기로 식별되는 경고 정책 테이블 항목입니다. 세트 선택기를 지정하지 않으면 모든 활성 경고 정책이 표시됩니다.
system_guid	10	PET 트랩의 GUID 필드를 채우는 데 사용되는 GUID입니다.
alert_string_count	11	경고 문자열의 최대 수입니다.
alert_string_key	12	지정된 세트 선택기로 식별되는 경고 문자열 키입니다. 세트 선택기를 지정하지 않으면 모든 경고 문자열 키가 표시됩니다.
alert_string	13	지정된 세트 선택기로 식별되는 경고 문자열입니다. 세트 선택기를 지정하지 않으면 모든 경고 문자열이 표시됩니다.
oem_filter_count	96	OEM 필터의 최대 수입니다.
oem_filter	97	지정된 세트 선택기로 식별되는 OEM 필터 테이블입니다. 세트 선택기를 지정하지 않으면 모든 활성 이벤트 필터가 표시됩니다.

예:

다음 세부절에서는 지원되는 각 매개 변수에 대해 자세히 설명합니다.  
전체 PEF 매개 변수 테이블을 가져와서 표시합니다.

```
# clia getpefconfig
IPM Sentry Shelf Manager Command Line Interpreter
PEF parameters:
  PEF control: 0x00
  PEF Action Global Control: 0x00
  PEF Startup Delay: 60 seconds
  PEF Alert Startup Delay: 60 seconds
  PEF Number of Event Filters: 64
  PEF Number of OEM Filters: 16
  Active Event Filters:
    None
  Active event filter data:
    None
  Alert Policies Count: 64
  Policy:
    None
  PEF GUID: Using the system GUID
Alert Strings Count: 64
Alert string key:
  None
Alert Strings:
  None
#
```

## control

구문:

```
getpefconfig control
getpefconfig 1
```

용도:

이 명령은 PEF 매개 변수 control의 현재 값을 표시합니다. 이 매개 변수는 단일 비트로 다음과 같이 정의되는 비트로 된 비트 마스크로서 처리됩니다.

- 0x01 - PEF 활성화
- 0x02 - PEF 작업에 대한 이벤트 메시지 생성 활성화
- 0x04 - 시스템 전원 켜기 및 재설정 시 PEF 시작 지연 활성화
- 0x08 - PEF 경고 시작 지연 활성화

다른 비트는 예약되어 있으며 0으로 설정해야 합니다.

예:

```
# clia getpefconfig control
IPM Sentry Shelf Manager Command Line Interpreter
PEF control: 0x07
    Enable PEF
    Enable Event Message for PEF Actions
    Enable PEF Startup Delay
#
```

## action\_control

구문:

```
getpefconfig action_control
getpefconfig 2
```

용도:

이 명령은 PEF 매개 변수 `action_control`의 현재 값을 표시합니다. 이 매개 변수는 단일 바이트로 다음과 같이 정의되는 비트로 된 비트 마스크로서 처리됩니다.

- 0x01 - 경고 작업 활성화
- 0x02 - 전원 끄기 작업 활성화
- 0x04 - 재설정 작업 활성화
- 0x08 - 전원 공급 주기 작업 활성화
- 0x10 - OEM 작업 활성화
- 0x20 - 진단 인터럽트 활성화

다른 비트는 예약되어 있으며 0으로 설정해야 합니다.

예:

```
# clia getpefconfig action_control
IPM Sentry Shelf Manager Command Line Interpreter
PEF Action Global Control: 0x3f
    Enable Alert Action
    Enable Power Down Action
    Enable Reset Action
    Enable Power Cycle Action
    Enable OEM Action
    Enable Diagnostic Interrupt
#
```

## startup\_delay

구문:

```
getpefconfig startup_delay  
getpefconfig 3
```

용도:

이 명령은 PEF 매개 변수 startup\_delay의 현재 값을 표시합니다. 이 매개 변수는 단일 바이트로 시작 시 PEF 기능이 지연되는 시간(초)을 나타냅니다.

예:

```
# clia getpefconfig startup_delay  
IPM Sentry Shelf Manager Command Line Interpreter  
PEF Startup Delay: 60 seconds  
#
```

## alert\_startup\_delay

구문:

```
getpefconfig startup_delay  
getpefconfig 4
```

용도:

이 명령은 PEF 매개 변수 alert\_startup\_delay의 현재 값을 표시합니다. 이 매개 변수는 단일 바이트로 시작 시 경고 기능이 지연되는 시간(초)을 나타냅니다.

예:

```
# clia getpefconfig alert_startup_delay  
IPM Sentry Shelf Manager Command Line Interpreter  
PEF Alert Startup Delay: 60 seconds  
#
```



## event\_filter\_count

구문:

```
getpefconfig event_filter_count  
getpefconfig 5
```

용도:

이 명령은 PEF 매개 변수 `event_filter_count`의 현재 값을 표시합니다. 이 읽기 전용 값은 이벤트 필터 테이블의 크기입니다. 이 값은 IPM Sentry 선반 관리자에 대한 구성 매개 변수이며 `shelfman` 구성 파일을 통해서만 변경할 수 있습니다.

예:

```
# clia getpefconfig event_filter_count  
IPM Sentry Shelf Manager Command Line Interpreter  
PEF Number of Event Filters: 64  
#
```

## event\_filter

구문:

```
getpefconfig event_filter [set-selector]  
getpefconfig 6 [set-selector]
```

용도:

이 명령은 이벤트 필터 테이블의 요소를 `set-selector`와 동일한 색인과 함께 표시합니다. 색인은 1부터 시작됩니다. 각 이벤트 필터에 대해 다음과 같은 정보가 표시됩니다.

- 필터 구성: 필터가 소프트웨어로 구성되었는지 제조업체에서 사전 구성되었는지 여부
- 이벤트 필터 작업 마스크
- 경고 정책 번호
- 이벤트 심각도
- 일치시킬 이벤트 소스 주소(255 = 모든 주소)
- 일치시킬 소스 채널/LUN(255 = 모든 소스 채널/LUN 일치)
- 일치시킬 센서 유형
- 일치시킬 센서 번호
- 일치시킬 이벤트 트리거(이벤트/읽기 유형)
- 이벤트 오프셋 마스크
- 이벤트 데이터 바이트 1, 2 및 3에 대한 AND, 비교1(CMP1) 및 비교2(CMP2) 마스크

세트 선택기를 생략하면 모든 활성 이벤트 필터 테이블 항목이 번호와 함께 표시됩니다.

예:

```
# clia getpefconfig event_filter 2

IPM Sentry Shelf Manager Command Line Interpreter
Active Event Filters:
0x02: Software Configurable Filter
    Action Mask: 0x01
    Policy Number: 1, Severity: Critical Condition
    Source Address: 0x20, LUN: 3, Channel: 15
    Sensor Type: Hot Swap (0xf0), Sensor # 255 (ANY)
    Event Trigger: 0xff (ANY), Event Offset Mask: 0xffff
    0: AND: 0x0f, CMP1: 0xff, CMP2: 0x00
    1: AND: 0x00, CMP1: 0x00, CMP2: 0x00
    2: AND: 0xff, CMP1: 0xff, CMP2: 0x00

#
```

## event\_filter\_data1

구문:

```
getpefconfig event_filter_data1 [set-selector]
getpefconfig 7 [set-selector]
```

용도:

이 명령은 이벤트 필터 테이블 요소의 첫 번째 바이트를 *set-selector*와 동일한 색인과 함께 표시합니다. 색인은 1부터 시작됩니다. 이 바이트는 16진수로 표시됩니다. 이 바이트의 비트는 다음과 같은 의미가 있습니다.

- 0x80 - 이 필터가 활성화되었습니다.
- 0x40 - 이 필터는 제조업체가 사전 구성했으며 소프트웨어로 변경하지 않아야 합니다.

다른 비트는 예약되어 있으며 0이어야 합니다.

세트 선택기를 생략하는 경우 각 활성 이벤트 필터 테이블 항목의 첫 번째 바이트가 해당 필터 번호와 함께 표시됩니다.

예:

```
# clia getpefconfig event_filter_data1 2
IPM Sentry Shelf Manager Command Line Interpreter

Active event filter data:
    0x02: 0x80 Enabled 1, Configuration: 0 ("Software
Configurable Filter")
#
```

## alert\_policy\_count

구문:

```
getpefconfig alert_policy_count
getpefconfig 8
```

용도:

이 명령은 PEF 매개 변수 `alert_policy_count`의 현재 값을 표시합니다. 이 읽기 전용 값은 경고 정책 테이블의 크기입니다. 이 값은 IPM Sentry 선반 관리자에 대한 구성 매개 변수이며 `shelfman` 구성 파일을 통해서만 변경할 수 있습니다.

예:

```
# clia getpefconfig alert_policy_count
IPM Sentry Shelf Manager Command Line Interpreter

Alert Policies Count: 64
#
```

## alert\_policy

구문:

```
getpefconfig alert_policy [set-selector]
getpefconfig 9 [set-selector]
```

용도:

이 명령은 경고 정책 테이블의 요소를 *set-selector*와 동일한 색인과 함께 표시합니다. 색인은 1부터 시작됩니다. 각 경고 정책에 대해 다음 정보가 표시됩니다.

- 정책 번호
- 정책 유형(이전 대상으로 전송된 경고에 대한)
- 대상 채널 번호
- 대상 선택기
- 경고 문자열 키

세트 선택기를 생략하면 모든 활성 경고 정책 테이블 항목이 번호와 함께 표시됩니다.

예:

```
# clia getpefconfig alert_policy 2
IPM Sentry Shelf Manager Command Line Interpreter

Policy:
    0x02: Policy# 5, Policy Type: 0, Channel: 1, DST: 1, Alert
String Sel: 1
#
```

## system\_guid

구문:

```
getpefconfig system_guid
getpefconfig 10
```

용도:

이 명령은 PEF 매개 변수 *system\_guid*의 현재 값을 표시합니다. 이 매개 변수는 PET 트랩 PDU에서 경고 대상에 전송된 GUID를 나타냅니다. 이 GUID는 별도의 GUID로 정의하거나 Get System GUID IPMI 명령으로 가져올 수 있는 시스템 GUID와 같은 것으로 정의할 수 있습니다.

예:

```
# clia getpefconfig system_guid
IPM Sentry Shelf Manager Command Line Interpreter

PEF GUID: 23662f7f-ba1b-4b65-8808-94ca09c9bbb0
#
```

## alert\_string\_count

구문:

```
getpefconfig alert_string_count  
getpefconfig 11
```

용도:

이 명령은 PEF 매개 변수 `alert_string_count`의 현재 값을 표시합니다. 이 읽기 전용 값은 동시 사용 시 경고 문자열의 최대 수인 경고 문자열 테이블의 크기입니다. 이 값은 IPM Sentry 선반 관리자에 대한 구성 매개 변수이며 `shelfman` 구성 파일을 통해서만 변경할 수 있습니다.

예:

```
# clia getpefconfig alert_string_count  
  
IPM Sentry Shelf Manager Command Line Interpreter  
  
Alert Strings Count: 64  
#
```

## alert\_string\_key

구문:

```
getpefconfig alert_string_key [set-selector]  
getpefconfig 12 [set-selector]
```

용도:

이 명령은 경고 문자열 키 테이블의 요소를 *set-selector*와 동일한 색인과 함께 표시합니다. 색인은 1부터 시작됩니다. 색인 0을 사용하여 휘발성 경고 문자열을 지정할 수 있습니다. 각 키는 경고 생성을 위해 이벤트 필터를 경고 문자열과 연결합니다. 각 경고 문자열 키에 대해 다음 정보가 표시됩니다.

- 경고 문자열 키 번호
- 연관된 이벤트 필터 번호
- 연관된 경고 문자열 번호

세트 선택기를 생략하면 모든 활성 경고 문자열 키 테이블 항목이 번호와 함께 표시됩니다.

예:

```
# clia getpefconfig alert_string_key 2
IPM Sentry Shelf Manager Command Line Interpreter
Alert string key: set selector 2, event_filter 0x10, string_set
0x11
#
```

## alert\_string

구문:

```
getpefconfig alert_string [set-selector]
getpefconfig 13 [set-selector]
```

용도:

이 명령은 경고 문자열 테이블의 요소를 *set-selector*와 동일한 색인과 함께 표시합니다. 색인은 1부터 시작됩니다. 색인 0을 사용하여 휘발성 경고 문자열을 지정할 수 있습니다. 이 명령은 전체 문자열을 한 번에 표시합니다.

세트 선택기를 생략하면 정의된 모든 경고 문자열이 번호와 함께 표시됩니다.

예:

```
# clia getpefconfig alert_string 2
IPM Sentry Shelf Manager Command Line Interpreter
Alert Strings:
0x02: "This is the alert string"
#
```

## oem\_filter\_count

구문:

```
getpefconfig oem_filter_count
getpefconfig 96
```

용도:

이 명령은 PEF 매개 변수 *oem\_filter\_count*의 현재 값을 표시합니다. 이 읽기 전용 값은 OEM 필터 테이블의 크기입니다. 이 값은 IPM Sentry 선반 관리자에 대한 구성 매개 변수이며 *shelfman* 구성 파일을 통해서만 변경할 수 있습니다.

OEM 필터 테이블은 IPMI 사양의 Pigeon Point Systems 정의 OEM 확장입니다. 이를 통해 플랫폼 이벤트 이외에도 PEF를 OEM-타임스탬프 되거나 되지 않은 SEL 항목 (레코드 유형 범위 C0h-FFh)에 적용할 수 있습니다.

예:

```
# clia getpefconfig oem_filter_count
IPM Sentry Shelf Manager Command Line Interpreter
PEF Number of OEM Filters: 16
#
```

## oem\_filter

구문:

```
getpefconfig oem_filter [set-selector]
getpefconfig 97 [set-selector]
```

용도:

OEM 필터 테이블은 IPMI 사양의 Pigeon Point Systems 정의 OEM 확장입니다. 이를 통해 플랫폼 이벤트 이외에도 PEF를 OEM-타임스탬프 되거나 되지 않은 SEL 항목 (레코드 유형 범위 C0h-FFh)에 적용할 수 있습니다.

OEM 필터 테이블의 각 항목은 이 OEM 필터가 적용되는 레코드 유형의 범위(OEM 레코드 유형의 범위)와, 일치하는 레코드 유형이 있는 레코드가 SEL에 배치될 때 호출되는 경고 정책 번호를 정의합니다.

이 명령은 OEM 필터 테이블의 요소를 *set-selector*와 동일한 색인과 함께 표시합니다. 색인은 1부터 시작됩니다. 각 OEM 필터에 대해 다음 정보가 표시됩니다.

- 바이트 1: SEL 레코드 유형 범위 Low 경계
- 바이트 2: SEL 레코드 유형 범위 High 경계
- 바이트 3: 바이트 1과 2에 지정된 범위에 일치하는 레코드 유형을 가진 SEL 항목에 대해 호출될 경고 정책 번호입니다.

세트 선택기를 생략하면 모든 활성 OEM 필터 테이블 항목이 번호와 함께 표시됩니다.

예:

```
# clia getpefconfig oem_filter
IPM Sentry Shelf Manager Command Line Interpreter
Active OEM Filters:
0x01: OEM range boundary 0xff:0xff, alert policy # 1
#
```

# getsensoreventenable

구문:

```
getsensoreventenable [IPMB-address [sensor-name | [lun:]sensor-number]]
getsensoreventenable board n [sensor-name | [lun:]sensor-number]]
getsensoreventenable shm n [sensor-name | [lun:]sensor-number]]
```

이 명령은 지정된 센서의 현재 이벤트 활성화 마스크 값을 표시합니다.

대상 제어기가 여러 LUN의 센서를 지원하는 경우 사용자는 이 명령을 사용하여 센서를 LUN(Logical Unit Number)으로 한정할 수 있습니다. LUN이 생략된 경우 모든 LUN에서 지정된 센서 번호를 가진 센서에 대한 정보가 표시됩니다. *lun*에는 값 0, 1 또는 3을 사용할 수 있습니다. LUN 2는 예약되어 있습니다.

센서 이름은 일반적으로 제어기 내에서 고유한 것으로 가정되기 때문에 LUN 번호로 한정되지 않습니다. 하지만 제어기 내에 이름이 같은 센서가 여러 개 있는 경우 모든 센서에 대한 정보가 표시됩니다.

이 명령은 지정된 센서의 지원되는 이벤트에 대한 현재 센서 이벤트 마스크 값을 표시합니다. 각 센서에 대해 다음 속성도 표시됩니다.

- 소유 IPM 제어기의 IPMB 주소
- 센서 번호, 센서 이름(SDR로부터의 장치 ID 문자열) 및 센서에 액세스할 수 있는 LUN
- 센서 유형

예:

IPM 제어기 FE의 온도 센서 Local Temp에 대한 이벤트 활성화 값을 가져옵니다.

```
# clia getsensoreventenable -v fe "Local Temp"

IPM Sentry Shelf Manager Command Line Interpreter

fe: LUN: 0, Sensor # 3 ("Local Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Assertion event mask: 0x0a80
        Assertion event for "Upper Non-Recoverable Going High"
enabled
        Assertion event for "Upper Critical Going High" enabled
        Assertion event for "Upper Non-Critical Going High" enabled
    Deassertion event mask: 0x0a80
        Deassertion event for "Upper Non-Recoverable Going High"
enabled
```



```
Deassertion event for "Upper Critical Going High" enabled
Deassertion event for "Upper Non-Critical Going High"
enabled
#
```

같은 센서에 대한 이벤트 활성화 정보를 가져오되 센서 LUN과 번호를 지정합니다.

```
# clia getsensoreventenable -v fe 0:3

IPM Sentry Shelf Manager Command Line Interpreter

fe: LUN: 0, Sensor # 3 ("Local Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Assertion event mask: 0x0a80
        Assertion event for "Upper Non-Recoverable Going High"
enabled
        Assertion event for "Upper Critical Going High" enabled
        Assertion event for "Upper Non-Critical Going High" enabled
    Deassertion event mask: 0x0a80
        Deassertion event for "Upper Non-Recoverable Going High"
enabled
        Deassertion event for "Upper Critical Going High" enabled
        Deassertion event for "Upper Non-Critical Going High"
enabled
#
```

---

## getthreshold | threshold

구문:

```
getthreshold [IPMB-address [sensor-name | [lun:]sensor-number]]
getthreshold board n [sensor-name | [lun:]sensor-number]]
getthreshold shm n [sensor-name | [lun:]sensor-number]]
```

getthreshold 대신 동사 threshold를 사용할 수 있습니다.

용도:

이 명령은 지정된 센서의 지원되는 임계값에 대한 현재 임계값을 표시합니다. 센서는 임계값 기반 센서여야 합니다. 원시 값과 처리된 값이 모두 표시됩니다. 각 센서에 대해 다음 속성도 표시됩니다.

- 소유 IPM 제어기의 IPMB 주소
- 센서 번호, 센서 이름(SDR로부터의 장치 ID 문자열) 및 센서에 액세스할 수 있는 LUN
- 센서 유형 및 이벤트/읽기 유형 코드

대상 제어기가 여러 LUN의 센서를 지원하는 경우 사용자는 이 명령을 사용하여 센서를 LUN(Logical Unit Number)으로 한정할 수 있습니다. LUN이 생략된 경우 모든 LUN에서 지정된 센서 번호를 가진 센서에 대한 정보가 표시됩니다. *lun*에는 값 0, 1 또는 3을 사용할 수 있습니다. LUN 2는 예약되어 있습니다.

센서 이름은 일반적으로 제어기 내에서 고유한 것으로 가정되기 때문에 LUN 번호로 한정되지 않습니다. 하지만 제어기 내에 이름이 같은 센서가 여러 개 있는 경우 모든 센서에 대한 정보가 표시됩니다.

예:

IPM 제어기 FE의 온도 센서 Local Temp에 대한 임계값을 가져옵니다.

```
# clia getthreshold -v fe "Local Temp"

IPM Sentry Shelf Manager Command Line Interpreter

fe: LUN: 0, Sensor # 3 ("Local Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
        Lower Critical Threshold, Raw Data: 0x80, Processed Data:
-128.000000 degrees C
        Upper Non-Critical Threshold, Raw Data: 0x50, Processed
Data: 80.000000 degrees C
        Upper Critical Threshold, Raw Data: 0x50, Processed Data:
80.000000 degrees C
        Upper Non-Recoverable Threshold, Raw Data: 0x50, Processed
Data: 80.000000 degrees C
#
```

같은 센서에 대한 임계값 정보를 가져오되 센서 LUN과 번호를 지정합니다.

```
# clia getthreshold -v fe 0:3

IPM Sentry Shelf Manager Command Line Interpreter

fe: LUN: 0, Sensor # 3 ("Local Temp")
```

```
Type: Threshold (0x01), "Temperature" (0x01)
  Lower Critical Threshold, Raw Data: 0x80, Processed Data:
-128.000000 degrees C
  Upper Non-Critical Threshold, Raw Data: 0x50, Processed
Data: 80.000000 degrees C
  Upper Critical Threshold, Raw Data: 0x50, Processed Data:
80.000000 degrees C
  Upper Non-Recoverable Threshold, Raw Data: 0x50, Processed
Data: 80.000000 degrees C
#
```

---

## help

구문:

```
help [command [subcommand]]
```

용도:

이 명령은 지원되는 명령에 대한 도움말 정보와 구문을 표시합니다.

예:

명령 및 구문 목록을 가져옵니다.

```
# clia help
IPM Sentry Shelf Manager Command Line Interpreter
Command Line Interface command set:
Parameters are case insensitive
In general:
    IPMB address is hexadecimal ALWAYS.
    All other numbers may be either decimal and hexadecimal (0x notation
    required for hexadecimal numbers)
    -v turns on verbose output

activate <addr> <fru_id>
alarm <alarm status/action>
board [slot_number]
boardreset <slot number>
busres force <res>
busres info [<res>]
busres lock <res>
busres query [-v] <res> [<target> [nouupdate]]
busres release <res>
```

```

busres sendbusfree <res> <target>
busres setowner <res> <target>
busres unlock <res>
console [slot_number]
deactivate <addr> <fru_id>
debuglevel [<mask>]
exit
fans <addr> <fru_id>
fru [<addr> [id=<fru_id> | type=<site_type>]] | [type=<site_type>
    [/<site_number>]]
frucontrol <addr> <fru_id> <command>
frudata [<addr>] [<fru_id>] [<block number>]
frudata shm <N> [<block number>]
frudata <addr> <fru_id> <byte offset> <byte_1> [byte2 .. [byte_16]]
frudatar <addr> <fru_id> <file name>
frudataw <addr> <fru_id> <file name>
fruinfo <addr> <fru_id>
getfanlevel <addr> <fru_id>
getlanconfig <channel number> <parameter number> | <parameter name>
getpefconfig <parameter name> | <parameter number> [<set selector>]
getsensoreventenable [ <addr> [ [ lun: ]<sensor_id> | <sensor name> ] ]
getthreshold [ <addr> [ [ lun: ]<sensor id> | <sensor name> ] ]
help [<command>]
ipmc [<addr>]
localaddress
minfanlevel [<min fan level>]
poll
quit
sel [clear] [ <addr> [ <number of items> [<number of first item>] ] ]
sel info [<addr>]
sensor [ <addr> [ [ lun: ]<sensor id> | <sensor name> ] ]
sensordata [ <addr> [ [ lun: ]<sensor id> | <sensor name> ] ]
sensorread <addr> [ lun: ]<sensor id>
session
setextracted <addr> <fru_id>
setfanlevel <addr> <fru_id> <state>
setlanconfig <channel number> <parameter number> | parameter name
    <parameters ...>
setlocked <addr> <fru_id> <value>
setpefconfig <parameter name> | <parameter number> [<set selector>]
    <parameters ...>
setsensoreventenable <addr> [ lun: ]<sensor_id> | <sensor name> global
    [assertion_events [deassertion_events]]
setthreshold <addr> [ lun: ]<sensor_id> | <sensor name> unc | uc | unr
    | lnc | lc | lnr [-r] value
shelf <parameters>
shelfaddress ["<shelf address>"]
shmstatus
showunhealthy

```

```

switchover
threshold [ <addr> [ [ lun: ]<sensor id> | <sensor name> ] ]
user [<user id>]
user add <user id> <user name> <flags> <privilege level> <password>
user channel <user id> <channel number> <flags> <privilege level>
user delete <user id>
user delete <user id>
user enable <user id> 1|0
user name <user id> <user name>
user passwd <user id> <user password>
version

# clia help shelf pwrreorder
IPM Sentry Shelf Manager Command Line Interpreter
Set the Power Order
PwrReorder <addr1> <fru_id1> before/after <addr2> <fru_id2>
#

```

특정 명령에 대한 도움말을 가져옵니다.

```

# clia help shelf pwrreorder
IPM Sentry Shelf Manager Command Line Interpreter
Set the Power Order
PwrReorder <addr1> <fru_id1> before/after <addr2> <fru_id2>
#

```

## ipmc

구문:

```

ipmc [-v] [IPMB-address]
ipmc board n
ipmc fan_tray n

```

용도:

이 명령은 지정된 주소에 있는 IPM 제어기에 대한 정보를 표시하거나, *IPMB-address*가 생략된 경우 선반 관리자가 인식하는 모든 IPM 제어기에 대한 정보를 표시합니다.

표준 모드에서는 IPM 제어기에 대해 다음 정보가 표시됩니다.

- 제어기의 IPMB 주소(2개의 16진수)
- IPM 제어기의 엔티티 ID와 엔티티 인스턴스
- IPM 제어기에 대한 최대 가능 FRU 장치 ID
- PICMG 확장 버전. PICMG 3.0 준수 IPM 제어기의 경우 이 버전은 2.0이어야 합니다.

IPM 제어기의 FRU 장치 0(IPM 제어기 자체를 나타냄)에 대한 현재 핫스왑 상태, 이전 핫스왑 상태 및 마지막 상태 변경 원인. 핫스왑 상태 M0-M7은 PICMG 3.0 사양에서 다음과 같이 정의됩니다.

- M0 – 설치되어 있지 않음
- M1 – 비활성
- M2 – 활성화 요청
- M3 – 활성화 진행 중
- M4 – FRU 활성화
- M5 – 비활성화 요청
- M6 – 비활성화 진행 중
- M7 – 통신 끊김

세부 정보 표시 모드에서는 IPM 제어기에 대해 다음과 같은 추가 정보가 표시됩니다.

- 제조 업체 ID, 제품 ID, 장치 ID, 장치 펌웨어 개정판 및 지원되는 IPMI 버전을 포함하여 Get Device ID IPMI 명령으로 반환되는 정보
- 제어기 SDR의 장치 ID 문자열
- 제어기 SDR의 전원 상태 알람 속성(16진수 숫자)
- 제어기 SDR의 전역 초기화 속성(16진수 숫자)
- 제어기 SDR의 장치 기능 속성(16진수 숫자)
- 제어기가 장치 SDR을 제공하는지 여부
- 지원되는 기능 마스크. 각 비트의 텍스트 설명이 함께 표시됩니다.
- 전자 키잉에 종속되는 포트 목록. 상태(Enabled/Disabled)가 함께 표시됩니다.

예:

주소 9C에서 IPM 제어기에 대한 정보를 가져옵니다.

```
# clia ipmc 9c
IPM Sentry Shelf Manager Command Line Interpreter
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
#
```

주소 9C에서 IPM 제어기에 대한 상세 정보를 가져옵니다.

```
# clia ipmc -v 9c
IPM Sentry Shelf Manager Command Line Interpreter
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID: 0x00, Revision: 0, Firmware: 1.01, IPMI ver 1.5
```

```
Manufacturer ID: 00315a (PICMG), Product ID: 0000, Auxiliary  
Rev: 01ac10ac  
Device ID String: "IPM Sentry 6"  
Global Initialization: 0x0, Power State Notification: 0x0,  
Device Capabilities: 0x29  
Controller provides Device SDRs  
Supported features: 0x29  
"Sensor Device" "FRU Inventory Device" "IPMB Event  
Generator"  
#
```

---

## localaddress

구문:

```
localaddress
```

용도:

이 명령은 현재 선반 관리자의 IPMB 주소를 하드웨어 주소에 기반하여(일반 BMC 주소 0x20와 반대로) 표시합니다. 이 주소는 중복 선반 관리자 간에 서로 다릅니다. BMC 주소는 서로 공유됩니다.

예:

```
# clia localaddress  
IPM Sentry Shelf Manager Command Line Interpreter  
Local IPMB Address = 0xFC  
#
```

# minfanlevel

구문:

```
minfanlevel [level]
```

용도:

이 명령은 최소 팬 수준을 표시하거나 설정합니다. 정상 조건에서 냉각 관리 알고리즘은 열 조건이 정상일 때 시스템의 팬 수준을 점차적으로 줄입니다. 하지만 냉각 관리 알고리즘은 팬 수준을 구성 매개 변수 MIN\_FAN\_LEVEL 또는 이 명령으로 지정된 최소 수준 아래로 낮추지는 않습니다.

최소 팬 수준의 기본값은 1입니다. 최소 팬 수준을 더 높은 값으로 설정하더라도 명령 `clia setfanlevel` 또는 RMCP를 통해 제출되는 ATCA 명령 `SetFanLevel`을 사용하여 팬 수준을 해당 값 아래로 설정하는 것이 가능합니다. 최소 팬 수준은 냉각 관리 기능에 의한 자동 팬 수준 관리에만 영향을 줍니다.

매개 변수 없이 이 명령을 사용하면 현재 최소 팬 수준이 표시됩니다.

이 명령에 정수 매개 변수를 지정하면 최소 팬 수준을 매개 변수 값으로 설정할 수 있습니다.

예:

```
# clia minfanlevel 3
IPM Sentry Shelf Manager Command Line Interpreter
Minimal Fan Level is set to 3

# clia minfanlevel
IPM Sentry Shelf Manager Command Line Interpreter
Minimal Fan Level is 3
#
```



# sel

구문:

```
sel [-v] [IPMB-address [record-count [starting-entry]]]  
sel clear [IPMB-address]  
sel info [IPMB-address]
```

*IPMB-address*는 board *n* 또는 shm *n* 약어로 바꿀 수 있습니다.

용도:

이 명령은 지정된 IPM 제어기(기본적으로 IPMB 주소 20h)의 SEL(시스템 이벤트 로그) 내용을 표시합니다. SEL의 레코더 번호 *starting-entry*부터 몇 개의 레코드를 표시할 것인지 지정하는 선택적 매개 변수 *record-count*를 지정할 수 있습니다. 선택적 매개 변수 *starting-entry*는 인쇄할 첫 번째 SEL 레코드의 항목 번호입니다. 이것은 SEL의 시작 부분에 상대적입니다. *record-count*와 *starting-entry*는 모두 1부터 SEL의 전체 레코드 수 사이의 범위에 있어야 합니다. 선택적 매개 변수 *starting-entry*의 기본값은 1입니다. *starting-entry*는 SEL 레코드의 RecordID 필드에 독립적입니다.

각 SEL 레코드에 대해 다음 정보 필드가 표시됩니다.

- 레코드 ID
- 레코드 유형(현재는 Event라는 단어가 표시되는 이벤트만 지원됨)
- 타임스탬프(타임스탬프 레코드 경우)
- 소스 주소 매개 변수: IPMB 주소, LUN 및 채널 번호
- 이벤트를 생성한 센서의 유형과 번호
- 이벤트/읽기 유형 코드
- 원시 형식과 처리된(사용 가능한 경우) 형식에서 이벤트 데이터의 3바이트

명령 `sel clear`는 지정된 IPM 제어기(기본적으로 IPMB 주소 20h)에서 SEL을 지웁니다.

`-v` 옵션을 사용하면 SEL 항목이 더 읽기 쉬운 형식으로 출력됩니다.

예:

선반 관리자의 SEL을 읽습니다.

```
# clia sel info  
IPM Sentry Shelf Manager Command Line Interpreter  
20: SEL version: 1.5  
    Number of log entries: 43  
    Free space: 15680 bytes  
    Last addition timestamp: Nov 19 17:12:47 2003
```

```
Last erase timestamp: Oct 31 23:59:59 2003
Supported operations: 0x0f
```

```
# clia sel 20 5
```

```
IPM Sentry Shelf Manager Command Line Interpreter
0x0027: Event: at Nov 19 17:12:42 2003; from:(0x9c,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M4->M6,
Cause=0x1
0x0028: Event: at Nov 19 17:12:42 2003; from:(0x9c,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M6->M1,
Cause=0x0
0x0029: Event: at Nov 19 17:12:46 2003; from:(0x9c,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M1->M2,
Cause=0x2
0x002A: Event: at Nov 19 17:12:46 2003; from:(0x9c,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M2->M3,
Cause=0x1
0x002B: Event: at Nov 19 17:12:47 2003; from:(0x9c,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M3->M4,
Cause=0x0
```

```
# clia sel b4 5
```

```
IPM Sentry Shelf Manager Command Line Interpreter
0x00A4: Event: at Nov 19 01:24:25 2003; from:(0x20,0,0);
sensor:(0x02,4); event:0x1(asserted): "Lower Non-Critical",
Threshold: 0xb3, Reading: 0xb3
0x00B8: Event: at Nov 19 00:04:11 2003; from:(0x20,0,0);
sensor:(0x02,4); event:0x1(asserted): "Lower Non-Critical",
Threshold: 0xb3, Reading: 0xb3
0x00CC: Event: at Nov 19 00:36:32 2003; from:(0x20,0,0);
sensor:(0x02,7); event:0x1(asserted): "Lower Non-Critical",
Threshold: 0xae, Reading: 0x94
0x00E0: Event: at Nov 19 00:36:32 2003; from:(0x20,0,0);
sensor:(0x02,7); event:0x1(asserted): "Lower Critical",
Threshold: 0xac, Reading: 0x94
0x00F4: Event: at Nov 19 00:02:37 2003; from:(0x20,0,0);
sensor:(0x01,2); event:0x1(asserted): "Upper Critical",
Threshold: 0x13, Reading: 0x1c
```

```
# clia sel -v board 3 5
```

```
IPM Sentry Shelf Manager Command Line Interpreter
0x00A4: Event: at: Nov 19 01:24:25 2003; from IPM Controller: 0x20,
LUN: 0, Channel: 0
    "Voltage" (0x02) sensor # 4
    "Threshold" (0x01) event Asserted
    "Lower Non-Critical Going Low"
```

```

Reading value: 0xb3
Threshold value: 0xb3
0x00B8: Event: at: Nov 19 00:04:11 2003; from IPM Controller: 0x20,
LUN: 0, Channel: 0
    "Voltage" (0x02) sensor # 4
    "Threshold" (0x01) event Asserted
    "Lower Non-Critical Going Low"
    Reading value: 0xb3
    Threshold value: 0xb3
0x00CC: Event: at: Nov 19 00:36:32 2003; from IPM Controller: 0x20,
LUN: 0, Channel: 0
    "Voltage" (0x02) sensor # 7
    "Threshold" (0x01) event Asserted
    "Lower Non-Critical Going Low"
    Reading value: 0x94
    Threshold value: 0xae
0x00E0: Event: at: Nov 19 00:36:32 2003; from IPM Controller: 0x20,
LUN: 0, Channel: 0
    "Voltage" (0x02) sensor # 7
    "Threshold" (0x01) event Asserted
    "Lower Critical Going Low"
    Reading value: 0x94
    Threshold value: 0xac
0x00F4: Event: at: Nov 19 00:02:37 2003; from IPM Controller: 0x20,
LUN: 0, Channel: 0
    "Temperature" (0x01) sensor # 2
    "Threshold" (0x01) event Asserted
    "Upper Critical Going High"
    Reading value: 0x1c
    Threshold value: 0x13
#

```

항목 # 15(0x0f)에서 5개의 SEL 항목을 가져옵니다.

```

# clia sel 20 5 15
IPM Sentry Shelf Manager Command Line Interpreter
0x000F: Event: at Nov 19 16:49:21 2003; from:(0x20,0,0);
sensor:(0xf0,3); event:0x6f(asserted): HotSwap: FRU 2 M2->M3,
Cause=0x1
0x0010: Event: at Nov 19 16:49:22 2003; from:(0x20,0,0);
sensor:(0xf0,2); event:0x6f(asserted): HotSwap: FRU 1 M2->M3,
Cause=0x1
0x0011: Event: at Nov 19 16:49:22 2003; from:(0x20,0,0);
sensor:(0xf0,2); event:0x6f(asserted): HotSwap: FRU 1 M3->M4,
Cause=0x0

```

```
0x0012: Event: at Nov 19 16:49:22 2003; from:(0xfc,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M3->M4,
Cause=0x0
0x0013: Event: at Nov 19 16:49:22 2003; from:(0x20,0,0);
sensor:(0xf0,3); event:0x6f(asserted): HotSwap: FRU 2 M3->M4,
Cause=0x0
#
```

SEL을 지웁니다.

```
# clia sel clear
IPM Sentry Shelf Manager Command Line Interpreter
SEL clear: issued successfully
      SEL clearing completed
# clia sel
IPM Sentry Shelf Manager Command Line Interpreter
SEL is empty
#
```

---

## sensor

구문:

```
sensor [-v] [IPMB-address [sensor-name | [lun:]sensor-number]]
sensor [-v] board n [sensor-name | [lun:]sensor-number]]
sensor [-v] shm n [sensor-name | [lun:]sensor-number]]
```

용도:

이 명령은 특정 센서에 대한 정보를 표시합니다. 대상 센서는 IPM 제어기의 IPMB 주소와 센서 번호 또는 센서 이름(큰따옴표로 묶인 센서 SDR의 장치 ID 문자열)으로 선택됩니다. 센서 이름이나 센서 번호를 지정하지 않으면 지정된 IPM 제어기의 모든 센서에 대한 정보가 표시됩니다. 매개 변수를 지정하지 않으면 알려진 모든 센서에 대한 정보가 표시됩니다.

대상 제어기가 여러 LUN의 센서를 지원하는 경우 사용자는 이 명령을 사용하여 센서를 LUN(Logical Unit Number)으로 한정할 수 있습니다. LUN이 생략된 경우 모든 LUN에서 지정된 센서 번호를 가진 센서에 대한 정보가 표시됩니다. *lun*에는 값 0, 1 또는 3을 사용할 수 있습니다. LUN 2는 예약되어 있습니다.

센서 이름은 일반적으로 제어기 내에서 고유한 것으로 가정되기 때문에 LUN 번호로 한정되지 않습니다. 하지만 제어기 내에 이름이 같은 센서가 여러 개 있는 경우 모든 센서에 대한 정보가 표시됩니다.

표준 모드에서는 각 센서에 대해 다음 정보가 표시됩니다.

- 소유 IPM 제어기의 IPMB 주소
- 센서 번호, 센서 이름(SDR로부터의 장치 ID 문자열) 및 센서에 액세스할 수 있는 LUN
- 센서 유형 및 이벤트/읽기 유형 코드
- 관련된 엔티티의 엔티티 ID, 엔티티 인스턴스(센서가 FRU와 연관된 경우 FRU 장치 ID)

센서에 대한 다음 정보는 세부 정보 표시 모드에서만 표시됩니다(이 속성에 대한 자세한 내용은 IPMI 사양 참조).

- 어설션 마스크
- 디어설션 마스크
- 센서 상태(별도 센서의 경우) 또는 임계값(임계값 기반 센서의 경우)에 대한 설정 가능/읽기 가능 마스크

다음 정보는 임계값 기반 센서의 세부 정보 표시 모드에서만 표시됩니다.

- 센서 장치: 기반 및 수정
- 단위 비율, 수정자 및 등급
- 아날로그 형식 및 플래그
- 선형화 매개 변수, M, B, K1, K2 계수
- 허용 한계 및 정확도 계수
- 정상, 정상 최대, 정상 최소, 최대 및 최소 값
- 상한 임계값: 비위험, 위험, 복구 불가
- 하한 임계값: 비위험, 위험, 복구 불가
- 히스테리시스 값: 양수 및 음수

예:

IPM 제어기 FE에서 센서 FAN 4에 대한 표준 정보를 가져옵니다.

```
# clia sensor fe "FAN 4"
IPM Sentry Shelf Manager Command Line Interpreter
fe: LUN: 0, Sensor # 14 ("FAN 4")
    Type: Threshold (0x01), "Fan" (0x04)
    Belongs to entity: (0xd0, 0) [FRU # 0]
#
```

IPM 제어기 9C에서 센서 2에 대한 상세 정보를 가져옵니다.

```
# clia sensor -v 9c 2
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 2 ("emulated temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Belongs to entity: (0xd0, 0) [FRU # 0]
    Assertion Mask: 0x7a95
```

```

Lower Non-Critical Going Low
Lower Critical Going Low
Lower Non-Recoverable Going Low
Upper Non-Critical Going High
Upper Critical Going High
Upper Non-Recoverable Going High
Upper non-critical threshold is comparison returned
Upper critical threshold is comparison returned
Upper non-recoverable threshold comparison is returned
Deassertion Mask: 0x7a95
Lower Non-Critical Going Low
Lower Critical Going Low
Lower Non-Recoverable Going Low
Upper Non-Critical Going High
Upper Critical Going High
Upper Non-Recoverable Going High
Upper non-critical threshold is comparison returned
Upper critical threshold is comparison returned
Upper non-recoverable threshold comparison is returned
Settable / Readable Mask: 0x3f3f
Lower Non-Critical Threshold is Readable
Lower Critical Threshold is Readable
Lower Non-Recoverable Threshold is Readable
Upper Non-Critical Threshold is Readable
Upper Critical Threshold is Readable
Upper Non-Recoverable Threshold is Readable
Lower Non-Critical Threshold is Settable
Lower Critical Threshold is Settable
Lower Non-Recoverable Threshold is Settable
Upper Non-Critical Threshold is Settable
Upper Critical Threshold is Settable
Upper Non-Recoverable Threshold is Settable
Unit Percentage: OFF (0), Unit Modifier: none (0), Unit Rate:
none (0)
Analog Format: 2's complement (signed) (2)
Base Unit: degrees C (1), Modifier Unit: unspecified (0)
Linearization: linear (0), M = 1, B = 0, K1 = 0, K2 = 0
Tolerance = 0, Accuracy = 0, Accuracy EXP = 0
Analog Flags: 0x0
Nominal: 0 (0x00), Normal max: 0 (0x00), Normal min: 0 (0x00)
Sensor max: 127 (0x7f), Sensor min: 128 (0x80)
Upper Thresholds:
Non-Critical: 70 (0x46) Critical: 80 (0x50) Non-Recoverable:
90 (0x5a)
Lower Thresholds:

```

```

Non-Critical: 3 (0x03) Critical: 0 (0x00) Non-Recoverable:
251 (0xfb)
Hysteresis:
    Positive: 2 (0x02), Negative 2 (0x02)
#

```

위와 동일하지만 센서에 대한 LUN을 명시적으로 지정합니다.

```

# clia sensor -v 9c 0:2
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 2 ("emulated temp")
Type: Threshold (0x01), "Temperature" (0x01)
Belongs to entity: (0xd0, 0) [FRU # 0]
Assertion Mask: 0x7a95
    Lower Non-Critical Going Low
    Lower Critical Going Low
    Lower Non-Recoverable Going Low
    Upper Non-Critical Going High
    Upper Critical Going High
    Upper Non-Recoverable Going High
    Upper non-critical threshold is comparison returned
    Upper critical threshold is comparison returned
    Upper non-recoverable threshold comparison is returned
Deassertion Mask: 0x7a95
    Lower Non-Critical Going Low
    Lower Critical Going Low
    Lower Non-Recoverable Going Low
    Upper Non-Critical Going High
    Upper Critical Going High
    Upper Non-Recoverable Going High
    Upper non-critical threshold is comparison returned
    Upper critical threshold is comparison returned
    Upper non-recoverable threshold comparison is returned
Settable / Readable Mask: 0x3f3f
    Lower Non-Critical Threshold is Readable
    Lower Critical Threshold is Readable
    Lower Non-Recoverable Threshold is Readable
    Upper Non-Critical Threshold is Readable
    Upper Critical Threshold is Readable
    Upper Non-Recoverable Threshold is Readable
    Lower Non-Critical Threshold is Settable
    Lower Critical Threshold is Settable
    Lower Non-Recoverable Threshold is Settable
    Upper Non-Critical Threshold is Settable
    Upper Critical Threshold is Settable

```

```

Upper Non-Recoverable Threshold is Settable
Unit Percentage: OFF (0), Unit Modifier: none (0), Unit Rate:
none (0)
Analog Format: 2's complement (signed) (2)
Base Unit: degrees C (1), Modifier Unit: unspecified (0)
Linearization: linear (0), M = 1, B = 0, K1 = 0, K2 = 0
Tolerance = 0, Accuracy = 0, Accuracy EXP = 0
Analog Flags: 0x0
Nominal: 0 (0x00), Normal max: 0 (0x00), Normal min: 0 (0x00)
Sensor max: 127 (0x7f), Sensor min: 128 (0x80)
Upper Thresholds:
Non-Critical: 70 (0x46) Critical: 80 (0x50) Non-Recoverable:
90 (0x5a)
Lower Thresholds:
Non-Critical: 3 (0x03) Critical: 0 (0x00) Non-Recoverable:
251 (0xfb)
Hysteresis:
Positive: 2 (0x02), Negative 2 (0x02)
#

```

## sensordata

### 구문:

```

sensordata [IPMB-address [sensor-name | [lun:]sensor-number]]
sensordata [-v] board n [sensor-name | [lun:]sensor-number]]
sensordata [-v] shm n [sensor-name | [lun:]sensor-number]]

```

### 용도:

이 명령은 지정된 센서(임계값 기반 센서의 경우) 또는 현재 어설션 상태(별도 센서의 경우)의 실제 값을 표시합니다. 대상 센서는 IPM 제어기의 IPMB 주소와 센서 번호 또는 센서 이름(큰따옴표로 묶인 센서 SDR의 장치 ID 문자열)으로 선택됩니다. 센서 이름이나 센서 번호를 지정하지 않으면 지정된 IPM 제어기의 모든 센서 값이 표시됩니다. 매개 변수를 지정하지 않으면 알려진 모든 센서의 값이 표시됩니다.

대상 제어기가 여러 LUN의 센서를 지원하는 경우 사용자는 이 명령을 사용하여 센서를 LUN(Logical Unit Number)으로 한정할 수 있습니다. LUN이 생략된 경우 모든 LUN에서 지정된 센서 번호를 가진 센서에 대한 정보가 표시됩니다. lun에는 값 0, 1 또는 3을 사용할 수 있습니다. LUN 2는 예약되어 있습니다.



센서 이름은 일반적으로 제어기 내에서 고유한 것으로 가정되기 때문에 LUN 번호로 한정되지 않습니다. 하지만 제어기 내에 이름이 같은 센서가 여러 개 있는 경우 모든 센서에 대한 정보가 표시됩니다.

각 센서에 대해 다음 정보가 표시됩니다.

- 소유 IPM 제어기의 IPMB 주소
- 센서 번호, 센서 이름(SDR로부터의 장치 ID 문자열) 및 센서에 액세스할 수 있는 LUN
- 센서 유형 및 이벤트/읽기 유형 코드
- 원시 형식의 센서 값(임계값 기반 센서의 경우) 또는 현재 어설션 상태의 마스크(별도 센서의 경우)
- 16진수 형식의 디코딩된 임계값 통과 상태

값/어설션 상태는 원시 형식과 처리된 형식 모두로 표시됩니다. 처리된 형식의 경우 아날로그 값은 M, B 및 R에 따라 변환되며 장치 이름과 함께 표시됩니다(예: 27 degrees). 별개 값은 이벤트/읽기 코드 유형에 따라 주석 표시됩니다. 예를 들어 이벤트/읽기 코드 2의 경우 어설션 상태 0은 Transition to Idle로 표시됩니다.

예:

IPM 제어기 FE의 온도 센서 Local Temp에 대한 센서 데이터 값을 가져옵니다.

```
# clia sensordata FE "Local Temp"
IPM Sentry Shelf Manager Command Line Interpreter
fe: LUN: 0, Sensor # 3 ("Local Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Status: 0xc0
        All event messages enabled from this sensor
        Sensor scanning enabled
        Initial update completed
    Raw data: 22 (0x16)
    Processed data: 22.000000 degrees C
    Status: 0x00
```

IPM 제어기 9C의 별개(핫스왑) 센서(#0)에 대한 센서 데이터 값을 가져옵니다.

```
# clia sensordata 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 0 ("FRU 0 HOT_SWAP")
    Type: Discrete (0x6f), "Hot Swap" (0xf0)
    Status: 0xc0
        All event messages enabled from this sensor
        Sensor scanning enabled
```

```
Initial update completed
Sensor reading: 0x00
Current State Mask 0x0010
```

같은 센서에 대한 센서 데이터 값을 가져오지만 LUN을 사용하여 명시적으로 한정합니다.

```
# clia sensordata 9c 0:0
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 0 ("FRU 0 HOT_SWAP")
Type: Discrete (0x6f), "Hot Swap" (0xf0)
Status: 0xc0
All event messages enabled from this sensor
Sensor scanning enabled
Initial update completed
Sensor reading: 0x00
Current State Mask 0x0010
```

---

## sensorread

구문:

```
sensorread IPMB-address [lun:]sensor-number
```

용도:

이 명령은 지정된 센서의 원시 값을 표시합니다. 명령 `sensorread`와 `sensordata` 간의 유일한 차이점은, 명령 `sensorread`는 대상 IPM 제어기의 존재 또는 센서 번호의 유효성을 확인하지 않고 IPMB를 통해 Get Sensor Reading 요청을 직접 전송한다는 점입니다. 이 명령은 센서의 SDR을 검색하지 않으므로 획득한 데이터를 처리할 수 없습니다.

대상 제어기가 여러 LUN의 센서를 지원하는 경우 사용자는 이 명령을 사용하여 센서를 LUN(Logical Unit Number)으로 한정할 수 있습니다. LUN을 생략하면 LUN 0이 사용됩니다. *lun*에는 값 0, 1 또는 3을 사용할 수 있습니다. LUN 2는 예약되어 있습니다.

각 센서에 대해 다음 정보가 표시됩니다.

- 소유 IPM 제어기의 IPMB 주소
- 센서 번호, 센서 이름(SDR로부터의 장치 ID 문자열) 및 센서에 액세스할 수 있는 LUN
- 센서 유형 및 이벤트/읽기 유형 코드
- 원시 형식의 센서 값(임계값 기반 센서의 경우) 또는 현재 어설션 상태의 마스크(별도 센서의 경우)

예:

IPM 제어기 FC의 센서 4에 대한 센서 데이터 값을 가져옵니다. `sensorread` 명령은 처리되지 않은 센서 값만 제공합니다. 명시적 LUN을 사용한 명령 예제를 참조하십시오.

```
# clia sensordata fc 4
IPM Sentry Shelf Manager Command Line Interpreter
fc: LUN: 0, Sensor # 4 ("3.3STBY voltage")
Type: Threshold (0x01), "Voltage" (0x02)
Status: 0xc0
    All event messages enabled from this sensor
    Sensor scanning enabled
    Initial update completed
Raw data: 193 (0xc1)
Processed data: 3.396800 Volts
Status: 0x00

# clia sensorread fc 4
IPM Sentry Shelf Manager Command Line Interpreter
fc: LUN: 0, Sensor # 4
Raw data: 193 (0xc1)
Status: 0xc0
    All event messages enabled from this sensor
    Sensor scanning enabled
    Initial update completed
Threshold Sensor Status: 0x00
Discrete Sensor Current State Mask 0x0000

# clia sensorread fc 0:4
IPM Sentry Shelf Manager Command Line Interpreter
fc: LUN: 0, Sensor # 4
Raw data: 193 (0xc1)
Status: 0xc0
    All event messages enabled from this sensor
    Sensor scanning enabled
    Initial update completed
Threshold Sensor Status: 0x00
Discrete Sensor Current State Mask 0x0000

#
```

# session

구문:

```
session
```

용도:

이 명령은 활성 RMCP 세션에 대한 정보를 표시합니다. 정보에는 다음과 같은 항목이 포함됩니다.

- 가능한 최대 세션 수와 현재 활성 세션의 수
- 현재 각 활성 세션에 대해 다음 정보가 표시됩니다.
  - 세션 핸들
  - 세션 활성화 중 사용된 사용자 ID와 이름
  - 최대 세션 권한 수준
  - IPMI 채널 번호와 유형
  - LAN 세션의 경우 피어 IP 주소와 포트 번호

예:

```
# clia session
IPM Sentry Shelf Manager Command Line Interpreter
32 sessions possible, 2 sessions currently active
Session: 1
    User: ID 1, Name: ""; Privilege Level: "Administrator"
    Channel: 1 ("LAN_802_3"); Peer IP address: 172.16.2.203, Port:
1764
Session: 2
    User: ID 1, Name: ""; Privilege Level: "Administrator"
    Channel: 1 ("LAN_802_3"); Peer IP address: 172.16.2.203, Port:
1765
#
```

---

## setextracted

구문:

```
setextracted IPMB-address fru-id
```

용도:

이 명령은 지정된 FRU가 선반에서 물리적으로 추출되었음을 선반 관리자에게 알립니다. 지정된 FRU가 상태 M7에 있으면 선반 관리자는 상태 M0(FRU가 물리적으로 없음)으로 스위치오버합니다.

예:

```
# clia setextracted 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
Set FRU extracted state successfully
#
```

---

## setfanlevel

구문:

```
setfanlevel IPMB-address fru-id level
setfanlevel fan_tray n level
setfanlevel all level
```

용도:

이 명령은 명령 매개 변수에 지정된 FRU가 제어하는 팬에 새로운 수준을 설정합니다. 최소값은 1이고 최대값은 15입니다.

이 명령에 *all* 한정자를 함께 사용하면 선반의 알려진 모든 팬에 같은 수준을 설정하려고 시도합니다.

예:

IPMB 주소 0x20부터 5까지 FRU #2에 있는 팬에 대해 팬 수준을 설정합니다.

```
# clia setfanlevel 20 2 5

IPM Sentry Shelf Manager Command Line Interpreter

20: FRU # 2 Set Fan Level to: 5
#
```

선반의 알려진 모든 팬에 대해 팬 수준을 4로 설정합니다.

```
# clia setfanlevel all 4

IPM Sentry Shelf Manager Command Line Interpreter

72: FRU # 0 Set Fan Level to: 4
76: FRU # 0 Set Fan Level to: 4
#
```

---

## setfruledstate

구문:

`setfruledstate IPMB-address fru-id LedId|ALL LedOp [LedColor]`

*LedOp* = ON | OFF | LOCAL | BLINK <onTime> <offTime> | TEST <onTime>

*LedColor* = BLUE | RED | GREEN | AMBER | ORANGE | WHITE | NONE | *number*

용도:

이 명령을 사용하면 특정 LED 또는 지정된 FRU에 대한 모든 LED의 상태를 설정할 수 있습니다.

첫 번째 인수 *IPMB-address*는 IPM 제어기의 IPMB 주소입니다. 두 번째 인수 *fru-id*는 FRU 장치 ID입니다. 세 번째 인수는 LED ID(숫자 값) 또는 ALL일 수 있습니다. 후자의 경우 지정된 작업이 모든 LED에 적용됩니다.

인수 *LedOp*는 PICMG 3.0 사양을 기반으로 FRU에 적용되는 작업을 지정합니다. 작업은 다음과 같이 정의됩니다.

- ON – LED를 켭니다.
- OFF – LED를 끕니다.

- LOCAL – LED의 로컬 제어로 되돌립니다.
- BLINK – LED를 깜박이게 합니다. 즉, *onTime*밀리초 동안 켜졌다가 *offTime*밀리초 동안 꺼지는 과정을 반복합니다.
- TEST – *onTime*밀리초 동안 램프 테스트를 수행합니다.

TEST 작업의 경우 *onTime*은 12800ms(12.8초)보다 작아야 합니다. BLINK 작업의 경우 *onTime*과 *offTime* 값은 10~2500ms 범위 내에 있어야 합니다.

선택적 매개 변수 *LedColor*는 기호 이름 또는 십진수 값으로 색상을 지정합니다. 색상의 기호 이름은 아래와 같이 PICMG 3.0 사양에 따른 십진수 값에 해당합니다. 매개 변수를 지정하지 않으면 기본 LED 색상이 사용됩니다.

- BLUE = 1
- RED = 2
- GREEN = 3
- AMBER = 4
- ORANGE = 6
- NONE = 14(색상을 변경하지 않음)

예:

IPMB 주소 20h에 있는 IPM 제어기의 FRU #0의 LED #1을 끕니다.

```
# clia setfruledstate 20 0 1 OFF
IPM Sentry Shelf Manager Command Line Interpreter

    Setting FRU's led state completed successfully, status = 0x0
#
```

IPMB 주소 20h에 있는 IPM 제어기의 FRU #0의 LED #1에 대한 로컬 제어를 활성화합니다.

```
# clia setfruledstate 20 0 1 LOCAL
IPM Sentry Shelf Manager Command Line Interpreter

    Setting FRU's led state completed successfully, status = 0x0
#
```

IPMB 주소 20h에 있는 IPM 제어기의 FRU #0의 LED #1에서 깜박임을 활성화합니다. 기본 색상으로 깜박입니다. 켜져 있는 기간은 100ms이고 꺼져 있는 기간은 200ms입니다.

```
# clia setfruledstate 20 0 0 BLINK 100 200
IPM Sentry Shelf Manager Command Line Interpreter

    Setting FRU's led state completed successfully, status = 0x0
#
```

---

## sethysteresis

구문:

```
sethysteresis IPMB-address [lun:] sensor_id | sensor_name pos | neg [-r]
value
```

용도:

이 명령은 지정된 센서에 대한 지정된 히스테리시스의 값을 설정합니다. 센서는 임계값 기반 센서여야 합니다. 또한 지정된 임계값 히스테리시스를 지원해야 하며 히스테리시스를 설정할 수 있어야 합니다.

대상 제어기가 여러 LUN의 센서를 지원하는 경우 사용자는 이 명령을 사용하여 센서를 LUN(Logical Unit Number)으로 한정할 수 있습니다. 명령은 pos 인수가 있으면 양수 히스테리시스를 설정하고 neg 인수가 있으면 음수 히스테리시스를 설정합니다.

예:

IPMB 주소 0xFC에 있는 IPM 제어기의 센서 #2에 대한 양수 히스테리시스를 설정합니다.

```
# clia sethysteresis FC 2 pos 10
IPM Sentry Shelf Manager Command Line Interpreter

Positive hysteresis set successfully to 0xA, previous: 0x0
#
```

---

## setipmbstate

구문:

```
setipmbstate IPMB-address A|B [link] 1|0 (방사형 IPMB-0 환경에서)
setipmbstate IPMB-address A|B 1|0 (버스된 IPMB-0 환경에서)
```

용도:

이 명령은 대상 IPM 제어기에서 IPMB 링크를 활성화하거나 비활성화합니다. 두 번째 인수는 활성화 또는 비활성화할 버스(IPMB-A 또는 IPMB-B)를 정의합니다. 마지막 인수는 수행할 작업을 정의합니다. 1은 링크를 활성화할 때 사용되고 0은 링크를 비활성화할 때 사용됩니다.



이 명령은 버스된 환경과 방사형 환경에서 서로 다르게 작동합니다. 버스된 환경 및 IPMB 허브가 아닌 대상 IPM 제어기에 대한 방사형 환경에서는, 인수 *link*가 사용되지 않습니다. 방사형 환경의 IPMB 허브 제어기에 대해서는 인수 *link*는 선택 사항입니다.

*link*가 있으면 명령은 특정 방사형 IPMB 링크(1에서 95까지)를 활성화하거나 비활성화합니다. *link*를 생략하면 명령은 방사형 시스템에서 IPMB 허브의 모든 링크를 활성화하거나 비활성화합니다.

예:

IPMB 주소 92h에서 IPM 제어기의 IPMB-A 링크를 비활성화합니다.

```
# clia setipmbstate 92 A 0
IPM Sentry Shelf Manager Command Line Interpreter

Command executed successfully
```

선반 관리자에서 방사형 IPMB 링크 3, 버스 B(IPMB 허브)를 활성화합니다.

```
# clia setipmbstate 20 B 3 1
IPM Sentry Shelf Manager Command Line Interpreter

Command executed successfully
```

---

## setlanconfig

구문:

```
setlanconfig channel parameter-name additional-parameters
setlanconfig channel parameter-number additional-parameters
```

용도:

이 명령은 지정된 채널에서 지정된 LAN 구성 매개 변수의 값을 설정합니다. 채널 번호, 구성 매개 변수 이름이나 번호 및 매개 변수 값은 명시적으로 지정해야 합니다.

표 A-3에서는 getlanconfig 명령이 지원하는 LAN 구성 매개 변수의 이름과 번호를 나열합니다.

표 A-3 setlanconfig의 LAN 구성 매개 변수

매개 변수 이름	번호	설명
auth_enables	2	LAN 채널의 콜백, 사용자, 운영자, 관리자 및 OEM 권한 수준에 대한 인증 유형 활성화 플래그가 포함된 5개의 8비트 값입니다.
ip	3	LAN 채널에 점으로 구분된 십진수 표기법으로 할당된 IP 주소가 포함된 문자열 값입니다.
subnet_mask	6	LAN 채널에 점으로 구분된 십진수 표기법으로 할당된 서브넷 마스크가 포함된 문자열 값입니다.
ipv4_hdr_param	7	RMCP 패킷 전송을 위한 다양한 IPv4 헤더 매개 변수가 포함된 3개의 8비트 값입니다. <ul style="list-style-type: none"> <li>• TTL(Time-to-live)</li> <li>• IP 헤더 플래그(비트 [7:5])</li> <li>• 우선 순위(비트 [7:5]) 및 서비스 유형(비트 [4:1])</li> </ul>
arp_control	10	LAN 채널에서 ARP 동작을 제어하는 다음 2개의 플래그입니다. <ul style="list-style-type: none"> <li>• ARP 요청에 대한 응답 활성화</li> <li>• 불필요한 ARP 전송 활성화</li> </ul>
arp_interval	11	고정 소수점 형식(정수 부분은 초를 나타내고 분수 부분은 밀리초를 나타냄)의 불필요한 ARP 간격입니다.
dft_gw_ip	12	점으로 구분된 십진수 표기법으로 표현한 기본 게이트웨이의 IP 주소가 포함된 문자열 값입니다.
backup_gw_ip	14	점으로 구분된 십진수 표기법으로 표현한 백업 게이트웨이의 IP 주소가 포함된 문자열 값입니다.
community	16	PET 트랩의 Community String 필드에 저장되는 문자열 값(최대 18개의 기호)입니다.

표 A-3 setlanconfig의 LAN 구성 매개 변수(계속)

destination_type	18	지정된 세트 선택기로 식별되는 대상 유형입니다. 이 매개 변수에 대해 세트 선택기를 지정해야 합니다. 각 대상 유형 항목에는 다음 필드가 포함됩니다. <ul style="list-style-type: none"> <li>• 대상 유형(0-7)</li> <li>• 경고 확인 플래그</li> <li>• 경고 확인 시간 초과/재시도 간격(1-256초)</li> <li>• 재시도 횟수(0-7)</li> </ul>
destination_address	19	지정된 세트 선택기와 연관된 대상 주소입니다. 이 매개 변수에 대해 세트 선택기를 지정해야 합니다. 각 대상 주소 항목에는 다음 필드가 포함됩니다. <ul style="list-style-type: none"> <li>• 게이트웨이 선택기: 0 - 기본 사용, 1 - 백업 사용</li> <li>• IP 주소(점으로 구분된 십진수 형식의 문자열)</li> <li>• MAC 주소(콜론[:] 기호로 구분된 6개의 16진수 바이트 값 문자열)</li> </ul>

## auth\_enables

구문:

```
setlanconfig channel auth_enables value1 value2 value3 value4 value5
setlanconfig channel 2 value1 value2 value3 value4 value5
```

용도:

이 명령은 LAN 매개 변수 auth\_enables의 현재 값을 설정합니다. 이 매개 변수는 5개의 지원되는 권한 수준(콜백, 사용자, 관리자, 운영자 및 OEM) 각각에 대해 선반 관리자가 현재 활성화한 인증 유형을 지정합니다. 이 유형은 5바이트 시퀀스로 표현되며 각 바이트가 각각의 권한 수준에 해당하며 아래와 같이 정의되는 비트로 된 비트 마스크로서 처리됩니다.

- 0x01 없음
- 0x02 MD2
- 0x04 MD
- 5 0x10 일반 암호/키
- 0x20 OEM 소유

value1부터 value5까지의 매개 변수는 이 바이트의 값을 16진수로 나타내야 합니다. 선반 관리자는 현재 콜백과 OEM 권한 수준을 지원하지 않습니다. 따라서 이들 권한 수준에 해당하는 value1 및 value5 매개 변수는 0으로 지정해야 합니다.

예:

```
# clia setlanconfig 1 auth_enables 0 1 1 1 0
IPM Sentry Shelf Manager Command Line Interpreter
Authentication Type Enables set successfully
#
```

## ip

구문:

```
setlanconfig channel ip value  
setlanconfig channel 3 value
```

용도:

이 명령은 채널에 사용되는 현재 IP 주소를 설정합니다. 값은 점으로 구분된 십진수 표기법으로 IP 주소를 나타내야 합니다.

예:

```
# clia setlanconfig 1 ip 172.16.2.203  
IPM Sentry Shelf Manager Command Line Interpreter  
IP Address set successfully  
#
```

## subnet\_mask

구문:

```
setlanconfig channel subnet_mask value  
setlanconfig channel 6 value
```

용도:

이 명령은 채널이 사용하는 현재 IP 서브넷 마스크를 설정합니다. 값은 점으로 구분된 십진수 표기법으로 서브넷 마스크를 나타내야 합니다.

예:

```
# clia setlanconfig 1 subnet_mask 255.255.255.0  
IPM Sentry Shelf Manager Command Line Interpreter  
Subnet Mask set successfully  
#
```

## ipv4\_hdr\_param

구문:

```
setlanconfig channel ipv4_hdr_param value1 value2 value3
setlanconfig channel 7 value1 value2 value3
```

용도:

이 명령은 선반 관리자에 대한 IP 4 헤더 매개 변수를 설정합니다. 3개의 단일 바이트 값이 16진수 표기법으로 표현됩니다(*value1*, *value2* 및 *value3*). 바이트의 내용은 IPMI 1.5 사양의 19.2절을 준수하며 다음 속성을 포함합니다.

- 바이트 1의 TTL(Time-to-live)
- 바이트 2의 IP 헤더 플래그(비트 [7:5])
- 바이트 3의 우선 순위(비트 [7:5]) 및 서비스 유형(비트 [4:1])

예:

```
# clia setlanconfig 1 ipv4_hdr_param 37 E0 11
IPM Sentry Shelf Manager Command Line Interpreter
IPv4 Header Parameters set successfully
#
```

## arp\_control

구문:

```
setlanconfig channel arp_control value
setlanconfig channel 10 value
```

용도:

이 명령은 LAN 매개 변수 *arp\_control*의 현재 값을 설정합니다. 이 매개 변수는 선반 관리자가 제공하는 추가 ARP 지원을 지정하고 단일 바이트로 표현되며 아래와 같이 정의되는 비트로 된 비트 마스크로서 처리됩니다.

- 0x01 - 선반 관리자가 생성한 불필요한 ARP 활성화
- 0x02 - 선반 관리자가 생성한 ARP 응답 활성화

다른 비트는 예약되어 있으며 0으로 설정해야 합니다.

예:

```
# clia setlanconfig 1 arp_control 3
IPM Sentry Shelf Manager Command Line Interpreter
BMC-generated ARP control set successfully
#
```

## arp\_interval

구문:

```
setlanconfig channel arp_interval value  
setlanconfig channel 11 value
```

용도:

이 명령은 채널에 사용되는 현재 ARP 간격을 설정합니다. 값은 초/밀리초 숫자를 고정 소수점 숫자 형식으로 나타내야 합니다(분수 부분 사용 가능). IPMI의 이 매개 변수 정의로 인해 값이 500밀리초로 나뉠 수 있는 가장 큰 시간 간격으로 잘립니다.

예:

```
# clia setlanconfig 1 arp_interval 3.5  
IPM Sentry Shelf Manager Command Line Interpreter  
Gratuitous ARP interval set successfully  
#
```

## dft\_gw\_ip

구문:

```
setlanconfig channel dft_gw_ip value  
setlanconfig channel 12 value
```

용도:

이 명령은 채널이 사용하는 기본 게이트웨이의 IP 주소를 설정합니다. 값은 점으로 구분된 십진수 표기법으로 IP 주소를 나타내야 합니다.

예:

```
# clia setlanconfig 1 dft_gw_ip 172.16.2.100  
IPM Sentry Shelf Manager Command Line Interpreter  
Default Gateway Address set successfully  
#
```

## backup\_gw\_ip

구문:

```
setlanconfig channel backup_gw_ip value  
setlanconfig channel 14 value
```

용도:

이 명령은 채널이 사용하는 백업 게이트웨이의 IP 주소를 설정합니다. 값은 점으로 구분된 십진수 표기법으로 IP 주소를 나타내야 합니다.

예:

```
# clia setlanconfig 1 backup_gw_ip 172.16.2.100  
IPM Sentry Shelf Manager Command Line Interpreter  
Backup Gateway Address set successfully  
#
```

## community

구문:

```
setlanconfig channel community value  
setlanconfig channel 16 value
```

용도:

이 명령은 PET 트랩에서 사용되는 커뮤니티 문자열 매개 변수를 설정합니다. 값은 큰따옴표로 묶인 문자열이어야 합니다.

예:

```
# clia setlanconfig 1 community "Community"  
IPM Sentry Shelf Manager Command Line Interpreter  
Community string set successfully  
#
```

## destination\_type

구문:

```
setlanconfig channel destination_type set-selector value1 value2 value3
setlanconfig channel 18 set-selector value1 value2 value3
```

용도:

이 명령은 대상 테이블의 요소를 *set-selector*와 동일한 색인과 함께 설정합니다. 색인은 0부터 시작됩니다. 선택기 0은 휘발성 대상의 주소 지정에 사용됩니다. 값 *value1*, *value2* 및 *value3*은 IPMI 사양의 19.2절에 따라 새 대상에 대한 정보를 제공합니다. 다음 정보가 제공됩니다.

- 경고 대상 유형(PET 트랩 또는 OEM 대상, 경고가 확인되어야 하는지 여부)
- 경고 확인 시간 초과
- 재시도 수

예:

```
# clia setlanconfig 1 destination_type 2 80 3 5
IPM Sentry Shelf Manager Command Line Interpreter
Destination Type set successfully
#
```

## destination\_address

구문:

```
setlanconfig channel destination_address set-selector gateway-sel
IP-address MAC-address
setlanconfig channel 19 set-selector gateway-sel IP-address MAC-address
```

용도:

이 명령은 대상 주소 테이블의 요소를 *set-selector*와 동일한 색인과 함께 설정합니다. 색인은 0부터 시작됩니다. 선택기 0은 휘발성 대상의 주소 지정에 사용됩니다. 명령 매개 변수는 다음과 같은 필수 정보를 제공합니다.

- *gateway-sel* - 사용할 게이트웨이. 기본 게이트웨이의 경우 0이고 백업 게이트웨이의 경우 1입니다.
- *IP-address* - 점으로 구분된 십진수 표기법으로 된 대상 IP 주소
- *MAC-address* - 콜론으로 구분된 6개의 16진수 바이트인 대상 MAC 주소



예:

```
# clia setlanconfig 1 destination_address 2 0 172.16.2.100
90:93:93:93:93:93
IPM Sentry Shelf Manager Command Line Interpreter
Destination Addresses set successfully
#
```

---

## setlocked

구문:

```
setlocked IPMB-address fru-id 0 |
setlocked IPMB-address fru-id 1
setlocked board n 0 | 1
setlocked shm n 0 | 1
setlocked fan_tray n 0 | 1
```

용도:

이 명령은 지정된 FRU의 Locked 비트를 지정된 상태(잠금 해제인 경우 0, 잠금인 경우 1)로 설정합니다. FRU는 소유 IPM 제어기의 IPMB 주소와 FRU 장치 ID를 사용하여 지정됩니다. FRU 장치 ID 0은 PICMG 3.0 컨텍스트에서 적합한 IPM 제어기를 지정합니다.

Locked 비트는 PICMG 3.0 사양에 따라 FRU가 상태 M1(비활성)에서 상태 M2(활성화 요청)로 자동으로 진행할 수 있는지 여부를 제어합니다. Locked 비트가 설정되어 있으면 이 스위치오버가 허용되지 않습니다. 선반 관리자가 Deactivate 명령을 FRU로 전송하면, FRU는 상태 M1로 스위치오버되고 Locked 비트가 설정되어 이후의 상태 스위치오버가 금지됩니다.

이 명령은 이전에 수동으로 비활성화된 FRU의 Locked 비트를 삭제하여 다시 활성화하는 데 사용할 수 있습니다.

예:

주소 9C에서 IPM 제어기에 대한 Locked 비트를 삭제하여 다시 활성화할 수 있게 합니다.

```
# clia setlocked 9c 0 0
IPM Sentry Shelf Manager Command Line Interpreter
Lock set successfully to 0x0
#
```

# setpefconfig

구문:

```
setpefconfig parameter-name additional-parameters  
setpefconfig parameter-number additional-parameters
```

용도:

이 명령은 지정된 PEF 구성 매개 변수의 새 값을 설정합니다. 표 A-4에서는 이 명령으로 설정할 수 있는 PEF 구성 매개 변수의 이름과 번호를 나열합니다.

표 A-4 setpefconf의 PEF 구성 매개 변수

매개 변수 이름	번호	설명
control	1	PEF에 대한 제어 플래그(예: enable PEF, enable PEF startup delay 등)를 나타내는 8비트 값입니다.
action_control	2	PEF에 대한 작업 전역 제어 플래그(예: enable reset, enable power down 등)를 나타내는 8비트 값입니다.
startup_delay	3	시스템 전원 켜기 및 재설정 후 PEF를 지연할 시간(초)입니다.
alert_startup_delay	4	시스템 전원 켜기 및 재설정 후 경고를 지연할 시간(초)입니다.
event_filter	6	지정된 세트 선택기로 식별되는 이벤트 필터 테이블입니다. 다음 19개의 숫자 값으로 구성되며, 각 값은 IPMI 사양 버전 1.5에서 테이블 15-2의 정의에 따라 인코딩된 16진수 값입니다. <ul style="list-style-type: none"><li>• 필터 구성</li><li>• 이벤트 필터 작업</li><li>• 경고 정책 번호</li><li>• 이벤트 심각도</li><li>• 생성기 ID 바이트 1</li><li>• 생성기 ID 바이트 2</li><li>• 센서 유형</li><li>• 센서 번호</li><li>• 이벤트 트리거(이벤트/읽기 유형)</li><li>• 이벤트 데이터 1 이벤트 오프셋 마스크</li><li>• 이벤트 데이터 1 AND 마스크</li><li>• 이벤트 데이터 1 비교 1</li><li>• 이벤트 데이터 1 비교 2</li><li>• 이벤트 데이터 2 AND 마스크</li><li>• 이벤트 데이터 3 비교 2</li></ul>

표 A-4 setpefconf의 PEF 구성 매개 변수(계속)

event_filter_data1	7	지정된 세트 선택기로 식별되는 이벤트 필터 테이블의 첫 번째 바이트입니다.
alert_policy	9	지정된 세트 선택기로 식별되는 경고 정책 테이블 항목입니다. 다음 5개의 숫자 값으로 구성되며, 각 값은 IPMI 1.5에서 테이블 15-4의 정의에 따라 인코딩된 16진수 값입니다. <ul style="list-style-type: none"> <li>• 정책 번호(4비트 값)</li> <li>• 활성화/비활성화 비트가 포함된 정책(4비트 값)</li> <li>• 채널 번호(4비트 값)</li> <li>• 대상 선택기(4비트 값)</li> <li>• 경고 문자열 세트/선택기</li> </ul>
system-guid	10	PET 트랩의 GUID 필드를 채우는 데 사용되는 GUID입니다.
alert_string_key	12	지정된 세트 선택기로 식별되는 경고 문자열 키입니다. 2개의 8비트 값(이벤트 필터 번호와 경고 문자열 세트)으로 구성됩니다.
alert_string	13	지정된 세트 선택기로 식별되는 경고 문자열입니다.
oem_filter	97	지정된 세트 선택기로 식별되는 OEM 필터 테이블입니다. 다음 3개의 숫자 값으로 구성됩니다. <ul style="list-style-type: none"> <li>• 바이트 1: SEL 레코드 유형 범위 Low 경계</li> <li>• 바이트 2: SEL 레코드 유형 범위 High 경계</li> <li>• 바이트 3: 위의 범위에 일치하는 레코드 유형을 가진 SEL 항목에 대해 호출될 경고 정책 번호.</li> </ul>

## control

### 구문:

```
setpefconfig control value
setpefconfig 1 value
```

### 용도:

이 명령은 PEF 매개 변수 control의 새 값을 설정합니다. 이 매개 변수는 단일 바이트로 다음과 같이 정의되는 비트로 된 비트 마스크로서 처리됩니다.

- 0x01 - PEF 활성화
- 0x02 - PEF 작업에 대한 이벤트 메시지 생성 활성화
- 0x04 - 시스템 전원 켜기 및 재설정 시 PEF 시작 지연 활성화
- 0x08 - PEF 경고 시작 지연 활성화

다른 비트는 예약되어 있으며 0으로 설정해야 합니다. 값은 16진수로 입력해야 합니다.

예:

```
# clia setpefconfig control 7
IPM Sentry Shelf Manager Command Line Interpreter
PEF control set successfully
#
```

## action\_control

구문:

```
setpefconfig action_control value
setpefconfig 2 value
```

용도:

이 명령은 PEF 매개 변수 `action_control`의 새 값을 설정합니다. 이 매개 변수는 단일 바이트로 다음과 같이 정의되는 비트로 된 비트 마스크로서 처리됩니다.

- 0x01 - 경고 작업 활성화
- 0x02 - 전원 끄기 작업 활성화
- 0x04 - 재설정 작업 활성화
- 0x08 - 전원 공급 주기 작업 활성화
- 0x10 - OEM 작업 활성화
- 0x20 - 진단 인터럽트 활성화

다른 비트는 예약되어 있으며 0으로 설정해야 합니다. 값은 16진수로 입력해야 합니다.

예:

```
# clia setpefconfig action_control 3f
IPM Sentry Shelf Manager Command Line Interpreter
PEF action control set successfully
#
```

## startup\_delay

구문:

```
setpefconfig startup_delay value  
setpefconfig 3 value
```

용도:

이 명령은 PEF 매개 변수 `startup_delay`의 새 값을 설정합니다. 이 매개 변수는 단일 바이트로 시작 시 PEF 기능이 지연되는 시간(초)을 나타냅니다. 값은 초를 나타내는 십진수로 지정합니다.

예:

```
# clia setpefconfig startup_delay 45  
IPM Sentry Shelf Manager Command Line Interpreter  
PEF startup delay set successfully  
#
```

## alert\_startup\_delay

구문:

```
setpefconfig startup_delay value  
setpefconfig 4 value
```

용도:

이 명령은 PEF 매개 변수 `alert_startup_delay`의 현재 값을 설정합니다. 이 매개 변수는 단일 바이트로 시작 시 경고 기능이 지연되는 시간(초)을 나타냅니다. 값은 초를 나타내는 십진수로 지정합니다.

예:

```
# clia setpefconfig alert_startup_delay 45  
IPM Sentry Shelf Manager Command Line Interpreter  
Alert startup delay set successfully  
#
```

## event\_filter

구문:

```
setpefconfig event_filter set-selector value1 ... <value19>
setpefconfig 6 set-selector value1 ... <value19>
```

용도:

이 명령은 이벤트 필터 테이블의 요소를 *set-selector*와 동일한 색인과 함께 설정합니다. 색인은 1부터 시작됩니다. 새 요소의 내용은 19개의 숫자 값(*value1*부터 *<value19>*)으로 지정되며 이 값은 IPMI 사양 버전 1.5, 테이블 15-2의 정의에 따라 인코딩된 16진수 값입니다.

- 필터 구성
- 이벤트 필터 작업
- 경고 정책 번호
- 이벤트 심각도
- 생성기 ID 바이트 1
- 생성기 ID 바이트 2
- 센서 유형
- 센서 번호
- 이벤트 트리거(이벤트/읽기 유형)
- 이벤트 데이터 1 이벤트 오프셋 마스크
- 이벤트 데이터 1 AND 마스크
- 이벤트 데이터 1 비교 1
- 이벤트 데이터 1 비교 2
- 이벤트 데이터 2 AND 마스크
- 이벤트 데이터 2 비교 1
- 이벤트 데이터 2 비교 2
- 이벤트 데이터 3 AND 마스크
- 이벤트 데이터 3 비교 1
- 이벤트 데이터 3 비교 2

예:

주소 9C에 있는 IPM 제어기, FRU 0이 상태 M0에 도달하면(경고 정책 #1에 따라 경고가 전송됨) 경고 작업을 트리거할 이벤트 필터 2를 설정합니다.

```
# clia setpefconfig event_filter 2 80 1 1 10 9C FF F0 FF FF FF FF
0F FF 0 0 0 0 FF FF 0
IPM Sentry Shelf Manager Command Line Interpreter
Event filter set successfully
#
```

## event\_filter\_data1

구문:

```
setpefconfig event_filter_data1 set-selector value
setpefconfig 7 set-selector value
```

용도:

이 명령은 이벤트 필터 테이블 요소의 첫 번째 바이트를 *set-selector*와 동일한 색인과 함께 설정합니다. 색인은 1부터 시작됩니다. 이 바이트는 16진수로 지정해야 합니다. 이 바이트의 비트는 다음과 같은 의미가 있습니다.

- 0x80 - 이 필터가 활성화되었습니다.
- 0x40 - 이 필터는 제조업체가 사전 구성했으며 소프트웨어로 변경하지 않아야 합니다.

다른 비트는 예약되어 있으며 0이어야 합니다.

이벤트 필터의 활성/비활성 상태를 빠르게 스위치오버할 때, 즉 전체 테이블 항목을 다시 작성할 필요 없이 켜거나 끌 때 이 명령을 사용할 수 있습니다.

예:

이벤트 필터 2를 켭니다.

```
# clia setpefconfig event_filter_data1 2 80
IPM Sentry Shelf Manager Command Line Interpreter
Event filter data1 set successfully
#
```

이벤트 필터 2를 끕니다.

```
# clia setpefconfig event_filter_data1 2 0
IPM Sentry Shelf Manager Command Line Interpreter
Event filter data1 set successfully
#
```

## alert\_policy

구문:

```
setpefconfig alert_policy set-selector value1 value2 value3 value4 value5
setpefconfig 9 set-selector value1 value2 value3 value4 value5
```

용도:

이 명령은 지정된 세트 선택기로 식별되는 경고 정책 테이블 항목을 설정합니다. 새 요소의 내용은 다음 5개의 숫자 값(*value1*부터 *value5*)으로 지정되며, 이 값은 IPMI 1.5의 테이블 15-4의 정의에 따라 인코딩된 16진수 값입니다.

- 정책 번호(4비트 값)
- 활성화/비활성화 비트가 포함된 정책(4비트 값)
- 채널 번호(4비트 값)
- 대상 선택기(4비트 값)
- 경고 문자열 세트/선택기

예:

다음 예제에서는 경고 정책 테이블 항목 2를 다음 속성으로 설정합니다.

- 정책 번호 = 5
- 활성화
- 정책 = 이 대상에 항상 경고를 전송
- 대상 채널 = 1
- 대상 선택기 = 1
- 경고 문자열 선택기 = 모든 이벤트에 문자열 1 사용

```
# clia setpefconfig alert_policy 2 5 8 1 1 1
IPM Sentry Shelf Manager Command Line Interpreter
Policy set successfully
#
```

## system\_guid

구문:

```
setpefconfig system_guid guid-value
setpefconfig 10 guid-value
setpefconfig system_guid none
setpefconfig 10 none
```



## 용도:

이 명령은 PEF 매개 변수 `system_guid`의 현재 값을 설정합니다. 이 매개 변수는 PET 트랩 PDU에서 경고 대상에 전송된 GUID를 나타냅니다. 이 GUID는 별도의 GUID로 정의하거나 시스템 GUID와 같은 것으로 정의할 수 있습니다.

`guid-value`는 표준 GUID 형식 `xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx`을 준수하는 실제 GUID 또는 기호 값 `none`으로 지정할 수 있습니다. 첫 번째의 경우 PEF 기능은 PET 트랩에 지정된 GUID를 사용합니다. 두 번째 경우 PEF 기능은 PET 트랩에 대해 시스템 GUID(IPMI Get System GUID 명령의 결과)를 기본값으로 사용합니다.

## 예:

```
# clia setpefconfig system_guid 23662F7F-BA1B-4b65-8808-94CA09C9BBB0
IPM Sentry Shelf Manager Command Line Interpreter
GUID set successfully
#
# clia setpefconfig system_guid none
IPM Sentry Shelf Manager Command Line Interpreter
Using the system GUID
#
```

# alert\_string\_key

## 구문:

```
setpefconfig alert_string_key set-selector value1 value2
setpefconfig 12 set-selector value1 value2
```

## 용도:

이 명령은 경고 문자열 키 테이블의 요소를 `set-selector`와 동일한 색인과 함께 설정합니다. 색인은 1부터 시작됩니다. 세트 선택기 0을 사용하여 휘발성 경고 문자열을 지정할 수 있습니다. 각 키는 경고 생성을 위해 이벤트 필터를 경고 문자열에 연관시킬 수 있으며 이벤트 필터 번호와 경고 문자열 번호로 구성됩니다. 두 값 모두 8비트 값이며 각각 16진수의 매개 변수 `value1` 및 `value2`로 지정됩니다.

## 예:

```
# clia setpefconfig alert_string_key 2 10 11
IPM Sentry Shelf Manager Command Line Interpreter
Alert string keys set successfully
#
```

## alert\_string

구문:

```
setpefconfig alert_string set-selector <string-value>
setpefconfig 13 set-selector <string-value>
```

용도:

이 명령은 경고 문자열 테이블의 요소를 *set-selector*와 동일한 색인과 함께 설정합니다. 색인은 1부터 시작됩니다. 색인 0을 사용하여 휘발성 경고 문자열을 지정할 수 있습니다. 문자열 값은 큰따옴표(")로 묶어야 하며 인용 부호 안에 특수 문자와 줄 바꿈을 넣을 수 있습니다.

예:

```
# clia setpefconfig alert_string 2 "This string has a line feed
inside."
IPM Sentry Shelf Manager Command Line Interpreter
Alert string set successfully
#
```

## oem\_filter

구문:

```
setpefconfig oem_filter set-selector value1 value2 value3
setpefconfig 97 set-selector value1 value2 value3
```

용도:

OEM 필터 테이블은 IPMI 사양의 Pigeon Point Systems 정의 OEM 확장입니다. 이를 통해 플랫폼 이벤트 이외에도 PEF를 OEM-타임스탬프 되거나 되지 않은 SEL 항목(레코드 유형 범위 C0h-FFh)에 적용할 수 있습니다.

OEM 필터 테이블의 각 항목은 이 OEM 필터가 적용되는 레코드 유형의 범위(OEM 레코드 유형의 범위)와, 일치하는 레코드 유형이 있는 레코드가 SEL에 배치될 때 호출되는 경고 정책 번호를 정의합니다.

이 명령은 지정된 세트 선택기로 식별되는 번호인 OEM 필터 테이블 항목을 설정합니다. 항목은 다음 3개의 숫자 값으로 구성됩니다.

- 바이트 1: SEL 레코드 유형 범위 Low 경계
- 바이트 2: SEL 레코드 유형 범위 High 경계
- 바이트 3: 위의 범위에 일치하는 레코드 유형을 가진 SEL 항목에 대해 호출될 경고 정책 번호

예:

```
# clia getpefconfig oem_filter
IPM Sentry Shelf Manager Command Line Interpreter
Active OEM Filters:
    0x01: OEM range boundary 0xff:0xff, alert policy # 1
#
# clia setpefconfig oem_filter 4 0xdc 0xf3 5
IPM Sentry Shelf Manager Command Line Interpreter
OEM filter set successfully
#
# clia getpefconfig oem_filter
IPM Sentry Shelf Manager Command Line Interpreter
Active OEM Filters:
    0x01: OEM range boundary 0xff:0xff, alert policy # 1
    0x04: OEM range boundary 0xdc:0xf3, alert policy # 5
#
```

---

## setsensoreventenable

구문:

```
setsensoreventenable IPMB-address sensor-name global
[assertion_events [deassertion_events]]

setsensoreventenable IPMB-address [lun:]sensor-number global
[assertion_events [deassertion_events]]
```

*IPMB-address* 대신 다음을 사용할 수 있습니다.

```
board n
shm n
```

용도:

이 명령은 지정된 센서의 이벤트 활성화 마스크를 변경합니다. 센서는 소유 IPM 제어기의 IPMB 주소 및 센서 이름이나 번호로 지정됩니다. 또는 보드 번호나 전용 선반 관리자 번호를 사용하여 대상 IPM 제어를 지정할 수 있습니다.

대상 제어기가 여러 LUN의 센서를 지원하는 경우 사용자는 이 명령을 사용하여 센서를 LUN(Logical Unit Number)으로 한정할 수 있습니다. *lun*에는 값 0, 1 또는 3을 사용할 수 있습니다. LUN 2는 예약되어 있습니다. LUN을 생략하면 명령은 가장 낮은 LUN에 지정된 센서 번호를 가진 센서에 적용됩니다. 예를 들어 명령이 명시적으로 LUN을 한정하지 않고 센서 3을 지정하며 대상 제어기가 LUN 1에서 센서 3을, LUN 3에서 다른 센서 3을 노출하는 경우, 명령이 LUN 1의 센서 3에 적용됩니다.

예:

IPM 제어기 FE에서 온도 센서 Local Temp에서 Lower Non-Critical Going Low 이벤트를 활성화합니다.

```
# clia setsensoreventenable fe "Local Temp" 0x90 0x01 0x00
IPM Sentry Shelf Manager Command Line Interpreter
Event enable mask set successfully
#
# clia getsensoreventenable -v fe "Local Temp"
IPM Sentry Shelf Manager Command Line Interpreter
fe: LUN: 0, Sensor # 3 ("Local Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Sensor scanning disabled
    Assertion event mask: 0x0001
        Assertion event for "Lower Non-Critical Going Low" enabled
    Deassertion event mask: 0x0000
#
```

같은 센서에서 같은 작업을 수행하지만 LUN과 센서 번호를 사용하여 센서를 지정합니다.

```
# clia setsensoreventenable fe 0:3 0x90 0x01 0x00
IPM Sentry Shelf Manager Command Line Interpreter
Event enable mask set successfully
#
```

# setthreshold

구문:

```
setthreshold IPMB-address sensor-name threshold-type [-r] value
setthreshold IPMB-address [lun:]sensor-number threshold-type [-r] value
```

IPMB-address 대신 다음을 사용할 수 있습니다.

```
board n
shm n
```

용도:

이 명령은 지정된 센서의 지정된 임계값에 대한 현재 임계값을 변경합니다. 센서는 소유 IPM 제어기의 IPMB 주소 및 센서 이름이나 번호로 지정됩니다. 대상 센서는 임계값 기반 센서여야 합니다. 매개 변수 *threshold-type*은 다음 기호 값 중 하나로 지정할 수 있습니다.

- upper\_non\_recoverable(약어: unr)
- upper\_critical(약어: uc)
- upper\_non\_critical(약어: unc)
- lower\_non\_recoverable(약어: lnr)
- lower\_critical(약어: lc)
- lower\_non\_critical(약어: lnc)

기본적으로 대상 값은 처리된 형식(즉, 전압 센서의 경우 Volts, 온도 센서의 경우 Celsius)으로 지정됩니다. 옵션 -r은 원시 값이 대신 사용됨을 의미합니다(일반적으로 센서별 규칙에 따라 변환된 바이트 크기 수량).

대상 제어기가 여러 LUN의 센서를 지원하는 경우 사용자는 이 명령을 사용하여 센서를 LUN(Logical Unit Number)으로 한정할 수 있습니다. *lun*에는 값 0, 1 또는 3을 사용할 수 있습니다. LUN 2는 예약되어 있습니다. LUN을 생략하면 명령은 가장 낮은 LUN에 지정된 센서 번호를 가진 센서에 적용됩니다. 예를 들어 명령이 명시적으로 LUN을 한정하지 않고 센서 3을 지정하며 대상 제어기가 LUN 1에서 센서 3을, LUN 3에서 다른 센서 3을 노출하는 경우, 명령이 LUN 1의 센서 3에 적용됩니다.

예:

IPM 제어기 9C의 온도 센서 *emulated temp*에 대한 상한 비위험 임계값을 섭씨 99도로 설정합니다.

```
# clia threshold 9c 2
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 2 ("emulated temp")
    Type: Threshold (0x01), "Temperature" (0x01)
```

```

        Lower Non-Critical Threshold, Raw Data: 0x03, Processed
Data: 3.000000 degrees C
        Lower Critical Threshold, Raw Data: 0x14, Processed Data:
20.000000 degrees C
        Lower Non-Recoverable Threshold, Raw Data: 0xfb, Processed
Data: -5.000000 degrees C
        Upper Non-Critical Threshold, Raw Data: 0x46, Processed
Data: 70.000000 degrees C
        Upper Critical Threshold, Raw Data: 0x50, Processed Data:
80.000000 degrees C
        Upper Non-Recoverable Threshold, Raw Data: 0x5a, Processed
Data: 90.000000 degrees C
#
# clia setthreshold 9c 0:2 unc 99
IPM Sentry Shelf Manager Command Line Interpreter
Threshold set successfully
#
# clia threshold 9c 0:2
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 2 ("emulated temp")
    Type: Threshold (0x01), "Temperature" (0x01)
        Lower Non-Critical Threshold, Raw Data: 0x03, Processed
Data: 3.000000 degrees C
        Lower Critical Threshold, Raw Data: 0x14, Processed Data:
20.000000 degrees C
        Lower Non-Recoverable Threshold, Raw Data: 0xfb, Processed
Data: -5.000000 degrees C
        Upper Non-Critical Threshold, Raw Data: 0x63, Processed
Data: 99.000000 degrees C
        Upper Critical Threshold, Raw Data: 0x50, Processed Data:
80.000000 degrees C
        Upper Non-Recoverable Threshold, Raw Data: 0x5a, Processed
Data: 90.000000 degrees C
#

```

---

# shelf

구문:

`shelf subcommand`

다음과 같은 하위 명령이 지원됩니다.

- `address_table`
- `cooling_state`
- `fans_state`
- `power_distribution`
- `power_management`
- `pci_connectivity`
- `ha_connectivity`
- `h110_connectivity`
- `point-to-point_connectivity`
- `MaxCurrent [feed] Amps`
- `MinVoltage [feed] Volts`
- `Activation addr fru_id 1|0`
- `Deactivation addr fru_id 1|0`
- `PwrCapability addr fru_id Watts`
- `PwrDelay addr fru_id 10ths_of_second`
- `Allowance seconds`
- `PwrReorder addr1 fru_id1 before|after addr2 fru_id2`
- `info_refresh`
- `info_force_update`

용도:

명령 `shelf`는 주요 선반 FRU 정보와 선반에 대해 선택된 현재 작동 데이터를 표시하며, 선반 FRU 정보의 일부 필드를 수정할 수 있게 합니다. 이 명령으로 표시하거나 수정할 정보의 유형을 명령 매개 변수로 지정합니다.

다음 세부절에서는 이 명령의 여러 용도의 `shelf` 명령 구문에 대해 설명합니다.

## 선반 FRU 정보 표시

구문:

```
shelf [cooling_state | fans_state | address_table
      | power_distribution | power_management
      | pci_connectivity | ha_connectivity
      | h110_connectivity | point-to-point_connectivity]
```

용도:

이 `shelf` 명령 구문은 주요 선반 FRU 정보와 선반에 대해 선택된 현재 작동 데이터를 표시합니다. 이 명령으로 표시되는 정보의 유형을 명령 매개 변수에 지정합니다. [표 A-5](#)에서는 `shelf` 명령이 지원하는 매개 변수 목록을 나열합니다.

표 A-5 shelf 명령의 매개 변수

명령 매개 변수	제공되는 정보
cooling_state (약어: cs)	<p>선반의 현재 냉각 상태를 다음과 같이 표시합니다.</p> <ul style="list-style-type: none"> <li>• <b>Normal</b> - 모든 온도 센서가 정상 작동 온도를 표시합니다.</li> <li>• <b>Minor Alert</b> - 적어도 하나의 온도 센서가 차요 경고 상태에 있습니다. 주요 또는 위험 경고 상태에 있는 센서는 없습니다.</li> <li>• <b>Major Alert</b> - 적어도 하나의 온도 센서가 주요 경고 상태에 있습니다. 위험 경고 상태에 있는 센서는 없습니다.</li> <li>• <b>Critical Alert</b> - 적어도 하나의 온도 센서가 위험 경고 상태에 있습니다.</li> </ul>
fans_state (약어: fs)	<p>선반에 있는 팬 속도계의 현재 상태를 다음과 같이 표시합니다.</p> <ul style="list-style-type: none"> <li>• <b>Normal</b> - 모든 팬 속도계 센서가 정상 작동 속도를 표시합니다.</li> <li>• <b>Minor Alert</b> - 적어도 하나의 팬 속도계 센서가 차요 경고 상태에 있습니다. 주요 또는 위험 경고 상태에 있는 센서는 없습니다.</li> <li>• <b>Major Alert</b> - 적어도 하나의 팬 속도계 센서가 주요 경고 상태에 있습니다. 위험 경고 상태에 있는 센서는 없습니다.</li> <li>• <b>Critical Alert</b> - 적어도 하나의 팬 속도계 센서가 위험 경고 상태에 있습니다.</li> </ul>
address_table (약어: at)	<p>선반 FRU 정보의 주소 테이블 레코드를 표시합니다. 다음 정보가 제공됩니다.</p> <ul style="list-style-type: none"> <li>• 선반 주소(유형에 따라 표시)</li> <li>• 주소 테이블 항목의 목록(각 항목의 하드웨어 주소, 사이트 번호 및 사이트 유형 표시)</li> </ul>



표 A-5 shelf 명령의 매개 변수(계속)

power_distribution (약어: pd)	<p>각 전원 공급에 대해 다음 정보가 제공됩니다(대부분 선반 FRU 정보의 선반 배전 레코드에서 가져옴).</p> <ul style="list-style-type: none"> <li>• 사용 가능한 최대 외부 전류</li> <li>• 최소 내부 전류</li> <li>• 최소 예상 작동 전압</li> <li>• 사용 가능한 실제 전원</li> <li>• 현재 사용되는 전력량</li> <li>• 공급 장치에 연결된 FRU 목록(각 FRU의 하드웨어 주소 및 FRU 장치 ID 표시)</li> </ul>
power_management (약어: pm)	<ul style="list-style-type: none"> <li>• 선반 FRU 정보의 선반 전원 관리 레코드입니다. 이 레코드에는 FRU 전원 설명자의 목록이 포함됩니다. 각 설명자에 대해 다음 정보가 제공됩니다.</li> <li>• 하드웨어 주소</li> <li>• FRU 장치 ID</li> <li>• 최대 FRU 전원 용량</li> <li>• 선반 관리자 제어 활성화</li> <li>• 다음 전원 켜기 전까지의 지연 수정</li> </ul>
pci_connectivity (약어: pcic)	<p>선반 FRU 정보의 선반 PCI 연결 레코드입니다. 다음 정보가 제공됩니다.</p> <ul style="list-style-type: none"> <li>• PCI 슬롯 설명자</li> <li>• DSEL 연결</li> <li>• 세그먼트 ID</li> <li>• 확장 PCI 슬롯 설명자</li> <li>• 지리적 주소</li> <li>• 인터페이스 번호</li> <li>• 사용 가능한 시스템 슬롯</li> </ul>
ha_connectivity (약어: ha)	<p>선반 FRU 정보의 선반 HA 연결 레코드입니다. 다음 정보가 제공됩니다.</p> <ul style="list-style-type: none"> <li>• 방사형 연결 지원</li> </ul>
h110_connectivity (약어: h110c)	<p>선반 FRU 정보의 선반 H110 연결 레코드입니다. 다음 정보가 제공됩니다.</p> <ul style="list-style-type: none"> <li>• 지리적 주소</li> <li>• 세그먼트 ID</li> </ul>
point-to-point_connectivity (약어: ppc)	<p>선반 FRU 정보의 선반 지점간 연결 레코드입니다. 다음 정보가 제공됩니다.</p> <ul style="list-style-type: none"> <li>• 채널 유형</li> <li>• 채널 수</li> <li>• 슬롯/하드웨어 주소</li> <li>• 채널 설명자</li> </ul>

명령 매개 변수 `cooling_state` 및 `fans_state`에 대해 상세 표시 옵션 `-v`를 사용할 수 있습니다. 이 옵션은 다음과 같이 명령 매개 변수 앞에 입력해야 합니다 (예: `clia shelf -v cooling_state`). 이 옵션을 사용하면 현재 상태에 기여하는 센서 (온도 또는 팬 속도계)의 목록이 표시됩니다. 각 센서는 튜플(*IPMB-address*, *sensor\_number*)로 표시됩니다.

예:

선반 냉각 상태를 가져옵니다.

```
# clia shelf cooling_state
IPM Sentry Shelf Manager Command Line Interpreter
Cooling state: "Normal"
#
```

선반 팬 속도계 상태를 가져옵니다(상세 표시).

```
# clia shelf -v fans_state
IPM Sentry Shelf Manager Command Line Interpreter
Fans state: "Major Alert"
  Sensor(s) at this state: (0x7e,10) (0x7e,11) (0x7e,12) (0x7e,13)
                           (0x7e,14) (0x7e,15) (0x7e,16) (0x7e,17)
#
```

주소 테이블을 가져옵니다.

```
# clia shelf address_table
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Address Table Record (ID=0x10)
  Version = 1
  Shelf Address      =
  Address Table Entries# = 16
  Hw Addr: 41, Site # 1, Type: "AdvancedTCA Board" 00
  Hw Addr: 42, Site # 2, Type: "AdvancedTCA Board" 00
  Hw Addr: 43, Site # 3, Type: "AdvancedTCA Board" 00
  Hw Addr: 44, Site # 4, Type: "AdvancedTCA Board" 00
  Hw Addr: 45, Site # 5, Type: "AdvancedTCA Board" 00
  Hw Addr: 46, Site # 6, Type: "AdvancedTCA Board" 00
  Hw Addr: 47, Site # 7, Type: "AdvancedTCA Board" 00
  Hw Addr: 48, Site # 8, Type: "AdvancedTCA Board" 00
  Hw Addr: 49, Site # 9, Type: "AdvancedTCA Board" 00
  Hw Addr: 4a, Site # 10, Type: "AdvancedTCA Board" 00
  Hw Addr: 4b, Site # 11, Type: "AdvancedTCA Board" 00
  Hw Addr: 4c, Site # 12, Type: "AdvancedTCA Board" 00
```

```
Hw Addr: 4d, Site # 13, Type: "AdvancedTCA Board" 00
Hw Addr: 4e, Site # 14, Type: "AdvancedTCA Board" 00
```

```
#
```

배전 정보를 가져옵니다.

```
# clia shelf power_distribution
IPM Sentry Shelf Manager Command Line Interpreter
Power Distribution:
  Feed count: 1
  Feed 00:
    Maximum External Available Current: 50.0 Amps
    Maximum Internal Current: Not specified
    Minimum Expected Operating Voltage: -40.5 Volts
    Actual Power Available: 2025.000 Watts
    Currently Used Power: 160.000 Watts
    Feed-to-FRU Mapping entries count: 16
      FRU Addr: 41, FRU ID: fe
      FRU Addr: 42, FRU ID: fe
      FRU Addr: 43, FRU ID: fe
      FRU Addr: 44, FRU ID: fe
      FRU Addr: 45, FRU ID: fe
      FRU Addr: 46, FRU ID: fe
      FRU Addr: 47, FRU ID: fe
      FRU Addr: 48, FRU ID: fe
      FRU Addr: 49, FRU ID: fe
      FRU Addr: 4a, FRU ID: fe
      FRU Addr: 4b, FRU ID: fe
      FRU Addr: 4c, FRU ID: fe
      FRU Addr: 4d, FRU ID: fe
      FRU Addr: 4e, FRU ID: fe
      FRU Addr: 4f, FRU ID: fe
      FRU Addr: 50, FRU ID: fe
```

```
#
```

전원 관리 정보를 가져옵니다.

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
  Version = 0
  Allowance for FRU Activation Readiness: 10 seconds
  FRU Activation and Power Description Count: 16
  Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
  150 Watts
```

```

Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds
Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds
Hw Address: 43, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds
Hw Address: 44, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds
Hw Address: 45, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds
Hw Address: 46, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds
Hw Address: 47, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds
Hw Address: 48, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds
Hw Address: 49, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds
Hw Address: 4a, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds
Hw Address: 4b, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds
Hw Address: 4c, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
Shelf Manager Controlled Activation: Enabled

```

```

        Delay Before Next Power On: 0.0 seconds
    Hw Address: 4d, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds
    Hw Address: 4e, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds
    Hw Address: 4f, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds
    Hw Address: 50, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds
#

```

## 사용 가능한 최대 외부 전류 수정

구문:

```
shelf maxcurrent [feed] current
```

용도:

이 명령은 지정된 공급 장치 번호의 사용 가능한 최대 외부 전류를 설정하고 선반에서 알려진 모든 선반 FRU 정보 인스턴스를 업데이트합니다. *feed* 매개 변수를 생략하면 값은 선반 FRU 정보의 첫 번째 공급 장치(공급 장치 0)에 대해 설정됩니다.

매개 변수 *feed*는 해당 공급 장치 설명의 순서를 기반으로 선반 FRU 정보에서 0부터 시작되는 공급 장치 번호입니다.

매개 변수 *current*는 원하는 전류 값(Amps)입니다.

예:

Feed 0의 사용 가능한 최대 외부 전류를 44Amps에서 99Amps로 변경합니다.

```

# clia shelf pd
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Power Distribution Record (ID=0x11)
    Version = 0
    Feed count: 1

```

```

Feed 00:
  Maximum External Available Current: 50.0 Amps
  Maximum Internal Current: Not specified
  Minimum Expected Operating Voltage: -40.5 Volts
  Actual Power Available: 2025.000 Watts
  Currently Used Power: 200.000 Watts
  Feed-to-FRU Mapping entries count: 16
    FRU Addr: 41, FRU ID: 0xfe
    FRU Addr: 42, FRU ID: 0xfe
    FRU Addr: 43, FRU ID: 0xfe
    FRU Addr: 44, FRU ID: 0xfe
    FRU Addr: 45, FRU ID: 0xfe
    FRU Addr: 46, FRU ID: 0xfe
    FRU Addr: 47, FRU ID: 0xfe
    FRU Addr: 48, FRU ID: 0xfe
    FRU Addr: 49, FRU ID: 0xfe
    FRU Addr: 4a, FRU ID: 0xfe
    FRU Addr: 4b, FRU ID: 0xfe
    FRU Addr: 4c, FRU ID: 0xfe
    FRU Addr: 4d, FRU ID: 0xfe
    FRU Addr: 4e, FRU ID: 0xfe
    FRU Addr: 4f, FRU ID: 0xfe
    FRU Addr: 50, FRU ID: 0xfe

```

```
# clia shelf maxcurrent 0 99
```

```

IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info
Cached information updated

```

```
# clia shelf pd
```

```

IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Power Distribution Record (ID=0x11)

```

```
Version = 0
```

```
Feed count: 1
```

```
Feed 00:
```

```

  Maximum External Available Current: 99.0 Amps
  Maximum Internal Current: Not specified
  Minimum Expected Operating Voltage: -40.5 Volts
  Actual Power Available: 2025.000 Watts
  Currently Used Power: 200.000 Watts
  Feed-to-FRU Mapping entries count: 16
    FRU Addr: 41, FRU ID: 0xfe
    FRU Addr: 42, FRU ID: 0xfe
    FRU Addr: 43, FRU ID: 0xfe
    FRU Addr: 44, FRU ID: 0xfe

```

```
FRU Addr: 45, FRU ID: 0xfe
FRU Addr: 46, FRU ID: 0xfe
FRU Addr: 47, FRU ID: 0xfe
FRU Addr: 48, FRU ID: 0xfe
FRU Addr: 49, FRU ID: 0xfe
FRU Addr: 4a, FRU ID: 0xfe
FRU Addr: 4b, FRU ID: 0xfe
FRU Addr: 4c, FRU ID: 0xfe
FRU Addr: 4d, FRU ID: 0xfe
FRU Addr: 4e, FRU ID: 0xfe
FRU Addr: 4f, FRU ID: 0xfe
FRU Addr: 50, FRU ID: 0xfe
```

#

## 최소 예상 작동 전압 수정

구문:

```
shelf minvoltage [feed] voltage
```

용도:

이 명령은 지정된 공급 장치 번호의 최소 예상 작동 전압을 설정하고 선반에서 알려진 모든 선반 FRU 정보 인스턴스를 업데이트합니다. *feed* 매개 변수를 생략하면 값은 선반 FRU 정보의 첫 번째 공급 장치(공급 장치 0)에 대해 설정됩니다.

매개 변수 *feed*는 해당 공급 장치 설명의 순서를 기반으로 선반 FRU 정보에서 0부터 시작되는 공급 장치 번호입니다.

매개 변수 *voltage*는 원하는 값입니다.

예:

Feed 0에 대한 최소 예상 작동 전압을 변경합니다.

```
# clia shelf pd
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Power Distribution Record (ID=0x11)
    Version = 0
    Feed count: 1
    Feed 00:
        Maximum External Available Current: 99.0 Amps
        Maximum Internal Current: Not specified
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 2025.000 Watts
```

```
Currently Used Power: 200.000 Watts
Feed-to-FRU Mapping entries count: 16
FRU Addr: 41, FRU ID: 0xfe
FRU Addr: 42, FRU ID: 0xfe
FRU Addr: 43, FRU ID: 0xfe
FRU Addr: 44, FRU ID: 0xfe
FRU Addr: 45, FRU ID: 0xfe
FRU Addr: 46, FRU ID: 0xfe
FRU Addr: 47, FRU ID: 0xfe
FRU Addr: 48, FRU ID: 0xfe
FRU Addr: 49, FRU ID: 0xfe
FRU Addr: 4a, FRU ID: 0xfe
FRU Addr: 4b, FRU ID: 0xfe
FRU Addr: 4c, FRU ID: 0xfe
FRU Addr: 4d, FRU ID: 0xfe
FRU Addr: 4e, FRU ID: 0xfe
FRU Addr: 4f, FRU ID: 0xfe
FRU Addr: 50, FRU ID: 0xfe
```

```
# clia shelf minvoltage 0 -59
```

```
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info
Cached information updated
```

```
# clia shelf pd
```

```
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Power Distribution Record (ID=0x11)
Version = 0
Feed count: 1
Feed 00:
Maximum External Available Current: 99.0 Amps
Maximum Internal Current: Not specified
Minimum Expected Operating Voltage: -59.0 Volts
Actual Power Available: 2025.000 Watts
Currently Used Power: 200.000 Watts
Feed-to-FRU Mapping entries count: 16
FRU Addr: 41, FRU ID: 0xfe
FRU Addr: 42, FRU ID: 0xfe
FRU Addr: 43, FRU ID: 0xfe
FRU Addr: 44, FRU ID: 0xfe
FRU Addr: 45, FRU ID: 0xfe
FRU Addr: 46, FRU ID: 0xfe
FRU Addr: 47, FRU ID: 0xfe
FRU Addr: 48, FRU ID: 0xfe
```



```

FRU Addr: 49, FRU ID: 0xfe
FRU Addr: 4a, FRU ID: 0xfe
FRU Addr: 4b, FRU ID: 0xfe
FRU Addr: 4c, FRU ID: 0xfe
FRU Addr: 4d, FRU ID: 0xfe
FRU Addr: 4e, FRU ID: 0xfe
FRU Addr: 4f, FRU ID: 0xfe
FRU Addr: 50, FRU ID: 0xfe
#

```

## 선반 관리자 제어 활성화 플래그 수정

구문:

```

shelf activation hardware-addr fru-id [1/0]
shelf activation board n [1/0]
shelf activation board all [1/0]
shelf activation fan_tray n [1/0]

```

용도:

이 명령은 지정된 IPM 제어기의 지정된 FRU에 대한 선반 관리자 제어 활성화 필드를 표시하거나 변경합니다. 이 명령은 선반 활성화 및 전원 관리 레코드에 이미 존재하는 항목에 대해서만 선반 관리자 제어 활성화 플래그를 수정합니다. 또한 이 명령은 선반 관리자가 사용하는 선반 FRU 정보의 캐시된 버전도 업데이트합니다. 따라서 선반 관리자 제어 활성화 필드의 새 값은 선반 관리자를 재시작할 필요 없이 곧바로 적용됩니다.

매개 변수 *hardware-addr*는 16진수 형식의 7비트 하드웨어 주소입니다.

매개 변수 *fru-id*는 16진수 형식의 FRU ID입니다. 0xFE는 해당 하드웨어 주소의 모든 FRU를 의미합니다.

예:

하드웨어 주소가 0x42(IPMB 주소 0x84)인 IPM 제어기에서 선반 관리자 제어 활성화를 활성화합니다.

```

# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
Version = 0
Allowance for FRU Activation Readiness: 10 seconds
FRU Activation and Power Description Count: 16
Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts

```

```
Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds

Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 43, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 44, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 45, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 46, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 47, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 48, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 49, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 4a, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
```

```

Shelf Manager Controlled Activation: Enabled
Delay Before Next Power On: 0.0 seconds

Hw Address: 4b, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 4c, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 4d, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 4e, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 4f, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 50, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

# clia shelf activation 42 0xfe 0
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info, address: 0x42, FRU ID # 254
    Cached information updated
    Wrote Information to the Shelf FRU

# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
    Version = 0
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 16

```

```
Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Disabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 43, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 44, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 45, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 46, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 47, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 48, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 49, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds
```

```
Hw Address: 4a, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 4b, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 4c, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 4d, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 4e, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 4f, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

Hw Address: 50, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Delay Before Next Power On: 0.0 seconds

#
```

## 선반 관리자 제어 비활성화 플래그 수정

구문:

```
shelf deactivation hardware-addr fru-id [1/0]
shelf deactivation board n [1/0]
shelf deactivation board all [1/0]
shelf deactivation fan_tray n [1/0]
```

용도:

이 명령은 지정된 IPM 제어기의 지정된 FRU에 대한 선반 관리자 제어 비활성화 필드를 표시하거나 변경합니다. 이 명령은 선반 활성화 및 전원 관리 레코드에 이미 존재하는 항목에 대해서만 선반 관리자 제어 비활성화 플래그를 수정합니다. 또한 이 명령은 선반 관리자가 사용하는 선반 FRU 정보의 캐시된 버전도 업데이트합니다. 따라서 선반 관리자 제어 비활성화 필드의 새 값은 선반 관리자를 재시작할 필요 없이 곧바로 적용됩니다.

매개 변수 *hardware-addr*는 16진수 형식의 7비트 하드웨어 주소입니다.

매개 변수 *fru-id*는 16진수 형식의 FRU ID입니다. 0xFE는 해당 하드웨어 주소의 모든 FRU를 의미합니다.

0의 값은 선반 관리자 제어 비활성화를 활성화하여 선반 관리자가 보드를 비활성화할 수 있게 합니다. 1의 값은 선반 관리자 제어 비활성화를 비활성화하여 선반 관리자가 자동으로 보드를 비활성화하지 못하도록 합니다. 선반 관리자 제어 비활성화가 비활성화 되면 시스템 관리자가 보드를 비활성화해야 합니다.

예:

하드웨어 주소가 0x41(IPMB 주소 0x84)인 IPM 제어기에서 선반 관리자 제어 비활성화를 활성화합니다.

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
    Version = 1
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 16
    Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
    Shelf Manager Controlled Activation: Enabled
    Shelf Manager Controlled Deactivation: Disabled
    Delay Before Next Power On: 0.0 seconds

    Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
```

```

        Shelf Manager Controlled Activation: Enabled
        Shelf Manager Controlled Deactivation: Disabled
        Delay Before Next Power On: 0.0 seconds

# clia shelf deactivation 0x41 0xfe 0
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info, address: 0x41, FRU ID # 254
    Cached information updated
    Wrote Information to the Shelf FRU

# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
    Version = 1
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 16
    Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Shelf Manager Controlled Deactivation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Shelf Manager Controlled Deactivation: Disabled
        Delay Before Next Power On: 0.0 seconds

#

```

## 최대 FRU 전원 용량 수정

구문:

```
shelf pwrcapability hardware-addr fru-id value
shelf pwrcapability board n value
shelf pwrcapability fan_tray n value
```

용도:

이 명령은 지정된 IPM 제어기의 지정된 FRU에 대한 최대 FRU 전원 용량 필드를 변경합니다.

---

주 – 최대 FRU 전원 용량 필드를 선반 환경에 대해 안전한 값보다 큰 값으로 설정하지 마십시오.

---

이 명령은 선반 활성화 및 전원 관리 레코드에 이미 있는 항목에 대해서만 이 필드를 수정합니다. 또한 이 명령은 선반 관리자가 사용하는 선반 FRU 정보의 캐시된 버전도 업데이트합니다. 따라서 최대 FRU 전원 용량 필드의 새 값은 선반 관리자를 재시작할 필요 없이 곧바로 적용됩니다.

- 매개 변수 *hardware-addr*는 16진수 형식의 7비트 하드웨어 주소입니다.
- 매개 변수 *fru-id*는 16진수 형식의 FRU ID입니다. 0xFE는 해당 하드웨어 주소의 모든 FRU를 의미합니다.
- 매개 변수 *value*는 필드의 새 값(와트)입니다. 가능한 값의 범위는 0부터 65535까지입니다.

예:

하드웨어 주소가 0x42(IPMB 주소 0x84)인 IPM 제어기에서 최대 FRU 전원 용량은 150와트로 설정합니다.

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
Allowance for FRU Activation Readiness: 10 seconds
FRU Activation and Power Description Count: 2
Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
Shelf Manager Controlled Activation: Disabled
Delay Before Next Power On: 2.2 seconds

Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
Shelf Manager Controlled Activation: Disabled
```



```

Delay Before Next Power On: 2.2 seconds
#
# clia shelf pwrcapability 42 0xfe 150
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info
    Cached information updated
#
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds

    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
#

```

## 다음 전원 켜기 전까지의 지연

구문:

```

shelf pwrdelay hardware-addr fru-id value
shelf pwrdelay board n value
shelf pwrdelay fan_tray n value

```

용도:

이 명령은 지정된 IPM 제어기의 지정된 FRU에 대한 다음 전원 켜기 전까지의 지연 필드를 변경합니다. 이 명령은 선반 활성화 및 전원 관리 레코드에 이미 있는 항목에 대해서만 이 필드를 수정합니다. 또한 이 명령은 선반 관리자가 사용하는 선반 FRU 정보의 캐시된 버전도 업데이트합니다. 따라서 다음 전원 켜기 전까지의 지연 필드의 새 값은 선반 관리자를 재시작할 필요 없이 곧바로 적용됩니다.

- 매개 변수 *hardware-addr*는 16진수 형식의 7비트 하드웨어 주소입니다.
- 매개 변수 *fru-id*는 16진수 형식의 FRU ID입니다. 0xFE는 해당 하드웨어 주소의 모든 FRU를 의미합니다.
- 매개 변수 *value*는 필드의 새 값(10분의 1초)입니다. 가능한 값의 범위는 0부터 63까지입니다.

예:

하드웨어 주소가 0x42(IPMB 주소 0x84)인 IPM 제어기의 다음 전원 켜기 전까지의 지연을 5초로 설정합니다.

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
#
# clia shelf pwrdelay 42 0xfe 50
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info
    Cached information updated
#
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 5.0 seconds
#
```

## FRU 활성화 준비 허용 수정

구문:

```
shelf allowance value
```

용도:

이 명령은 FRU 활성화 준비 허용 매개 변수를 변경합니다.

매개 변수 *value*는 매개 변수의 새 값(초)입니다. 가능한 값의 범위는 0부터 255까지입니다.

예:

FRU 활성화 준비 허용을 5초로 설정합니다.

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds

    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds

#
# clia shelf allowance 5

IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info
#
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 5 seconds
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
```

```

Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
    Shelf Manager Controlled Activation: Disabled
    Delay Before Next Power On: 2.2 seconds
#

```

## FRU 활성화 및 전원 설명자 순서 재정렬

구문:

```
shelf pwrreorder hardware-addr-1 fru-id-1 before|after hardware-addr-2
fru-id-2
```

*hardware-address fru-id*는 다음의 항목으로 대체할 수 있습니다.

```
board n
fan_tray n
```

용도:

이 명령은 선반 FRU 정보에서 FRU 활성화 및 전원 설명자의 순서를 변경합니다. 이 명령은 이미 있는 설명자만 재정렬할 수 있습니다. 또한 현재의 구현에서는 하나의 선반 활성화 및 전원 관리 레코드 내의 설명자만 재정렬하도록 제한되어 있습니다. 또한 이 명령은 선반 관리자가 사용하는 선반 FRU 정보의 캐시된 버전도 업데이트합니다. 따라서 새 설명자 순서는 선반 관리자를 재시작할 필요 없이 곧바로 적용됩니다.

- 매개 변수 *hardware-addr-1*은 이동해야 할 설명자의 16진수 형식 7비트 하드웨어 주소입니다.
- 매개 변수 *fru-id-1*은 이동해야 할 설명자의 16진수 형식 FRU ID입니다. 0xFE는 해당 하드웨어 주소의 모든 FRU를 의미합니다.
- 매개 변수 *hardware-addr-2*는 *hardware-addr-1/fru-id-1* 설명자가 위치할 앞이나 뒤 설명자의 16진수 형식 7비트 하드웨어 주소입니다.
- 매개 변수 *fru-id-2*는 *hardware-addr-1/fru-id-1* 설명자가 위치할 앞이나 뒤 설명자의 16진수 형식 FRU ID입니다.

예:

하드웨어 주소 0x42(IPMB 주소 0x84)의 IPM 제어기에 대한 설명자는 하드웨어 주소 0x41(IPMB 주소 0x82)의 IPM 제어기에 대한 설명자 앞에 배치합니다.

```

# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 2

```

```

    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
    Shelf Manager Controlled Activation: Disabled
    Delay Before Next Power On: 2.2 seconds

    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
    Shelf Manager Controlled Activation: Disabled
    Delay Before Next Power On: 2.2 seconds
#
# clia shelf pwrreorder 42 0xfe before 41 0xfe
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info
    Cached information updated
#
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 2
    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
    Shelf Manager Controlled Activation: Disabled
    Delay Before Next Power On: 2.2 seconds

    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
    Shelf Manager Controlled Activation: Disabled
    Delay Before Next Power On: 2.2 seconds
#

```

## 선반 FRU 정보 새로 고침

구문:

```
shelf info_refresh
```

용도:

이 명령은 선반 관리자가 선반에서 이전에 발견된 선반 FRU 정보의 소스를 다시 읽고 소스에 유효한 선반 FRU 정보가 포함되어 있는지 재평가합니다. 올바른 선반 FRU 정보가 확인되면 모든 선반 FRU 정보 저장 장치와 선반 FRU 정보의 캐시된 마스터 사본이 새 선반 FRU 정보의 내용으로 업데이트됩니다.

PICMG 3.0(3.6.4절)에 지정된 것처럼 선반 관리자는 초기화 중에 가능한 선반 FRU 정보 저장소 찾기를 시도합니다. 선반 관리자가 유효한 선반 FRU 정보가 포함된 FRU 정보 장치를 최소 2개 찾으면, 선반 관리자는 election을 수행하여 사용할 선반 FRU 정보 소스를 결정합니다. 이 선택은 저장 장치에 포함된 데이터에 대한 유효성 검사와 내용 비교를 기반으로 합니다. 선택을 하고 나면 선반 관리자는 휘발성 메모리에 있는 FRU 정보의 마스터 사본의 캐시된 마스터 사본을 만듭니다. 이 사본은 선반 FRU 정보 소스의 업데이트에 사용되며 선반 FRU 정보의 유일한 소스로 취급됩니다. 따라서 모든 선반 FRU 정보 관련 작업은 마스터 사본으로 수행되며, 마스터 사본의 변경 내용은 자동으로 모든 선반 FRU 정보 소스 장치로 전파됩니다.

하지만 동적 재구성은 지원되지 않습니다. 새 선반 FRU 정보가 이전 선반 FRU 정보와 다른 경우에는 선반 관리자를 재부팅해야 변경 내용이 완전히 적용됩니다.

예:

성공적 새로 고침으로 인한 두 개의 일치하는 선반 FRU 정보 소스.

```
# clia shelf info_refresh
```

```
IPM Sentry Shelf Manager Command Line Interpreter
```

```
Read 0x20 # 2, size = 1024
```

```
Read 0x20 # 1, size = 1024
```

```
Found 2 Matching Shelf FRU Info
```

```
0x20 # 2, size = 1024 (data size = 775), "Valid" Shelf FRU,  
"Matching"
```

```
0x20 # 1, size = 1024 (data size = 775), "Valid" Shelf FRU,  
"Matching"
```

```
Shelf FRU Info was not changed
```

```
#
```

```
Unsuccessful refresh: both data sources contain non-matching or  
invalid data.
```

```
# clia shelf info_refresh
```

```
IPM Sentry Shelf Manager Command Line Interpreter
```

```
Read 0x20 # 2, size = 1024
```

```
Read 0x20 # 1, size = 1024
```

```
No Matching Shelf FRU Info found
```

```
0x20 # 2, size = 1024 (data size = 293), "Invalid" Shelf FRU, "Non-  
Matching"
```

```
0x20 # 1, size = 1024 (data size = 529), "Valid" Shelf FRU, "Non-  
Matching"
```

```
Refresh was not done because system found only 1 (of 2) Matching  
Shelf FRU info
```

```
#
```

## 선반 FRU 정보 저장 장치 업데이트

구문:

```
shelf info_force_update
```

용도:

이 명령은 선반 FRU 정보 소스 장치를 검사하고 선반 FRU 정보 마스터 사본의 내용을 이들 소스 장치 모두에 복사합니다. 이 명령은 선반 FRU 정보 마스터 사본과 비휘발성 소스 장치 간에 충돌이 있고 이 충돌이 자동으로 해결되지 않는 경우(예: EEPROM과 마스터 사본이 서로 다른 경우)에 유용합니다. 이 경우 운영자는 이 명령을 사용하여 EEPROM을 마스터 사본의 내용과 강제로 동기화시킬 수 있습니다. 또한 이 명령은 원래의 충돌로 인해 발생한 오류 조건도 지웁니다. 즉, 이 명령을 실행하고 나면 선반 FRU 정보에 대한 이후의 업데이트를 다시 SEEPROM으로 전파하기 시작합니다.

이 명령은 비동기적인 방식으로 선반 FRU 정보 소스 장치의 업데이트를 시작합니다.

예:

```
# clia shelf info_force_update

IPM Sentry Shelf Manager Command Line Interpreter

Starting the Shelf FRU Info source device update
#
```

---

## shelfaddress

구문:

```
shelfaddress [up-to-30-characters-of-the-shelf-address]
```

용도:

이 명령은 선반 FRU 정보 내에서 주소 테이블의 선반 주소 필드를 가져오거나 설정합니다. 이 명령은 6비트 팩 값을 사용하므로 대문자와 숫자만 허용됩니다.

소문자는 자동으로 대문자로 변환됩니다.



예:

```
# clia shelfaddress
IPM Sentry Shelf Manager Command Line Interpreter
Shelf Address Info: "1234"
#
# clia shelfaddress "NEW SHELF ADDRESS"
IPM Sentry Shelf Manager Command Line Interpreter
Shelf Address Info set successfully
#
# clia shelfaddress
IPM Sentry Shelf Manager Command Line Interpreter
Shelf Address Info: "NEW SHELF ADDRESS"
#
```

---

## shmstatus

구문:

shmstatus

용도:

이 명령은 중복 구성에서 선반 관리자의 상태(활성 또는 백업)를 반환합니다. 세부 정보 표시 모드에서는 더 자세한 정보, 즉 선반 FRU 정보의 상태, RMCP 인터페이스의 상태 및 백업 선반 관리자의 상태(쿼리 대상 선반 관리자가 활성 관리자인 경우)가 표시됩니다. Ready For Operation 플래그는 다음의 경우에 Yes로 표시되는 매개 변수입니다.

- 유효한 선반 FRU 정보를 찾아서 RMCP 인터페이스를 성공적으로 초기화하는 경우 활성 선반 관리자에서
- 활성 선반 관리자로부터 중복성 상태 정보를 성공적으로 수신한 경우 백업 선반 관리자에서

예:

```
# clia shmstatus -v
IPM Sentry Shelf Manager Command Line Interpreter
Shelf Manager status: "Active"
Ready For Operation: Yes
Detailed State Flags: "Shelf FRU Found" "RMCP Up" "Backup Healthy"
#
```

---

# showhost

구문:

`showhost slot-number`

용도:

이 명령은 Netra CP3060 보드에 설치된 시스템 펌웨어 버전을 표시할 때 사용합니다.  
이 명령은 Netra CT 900 서버에 설치된 Netra CP3060 보드에 대해서만 유효합니다.

매개 변수 *slot-number*는 Netra CP3060 보드의 슬롯 번호를 지정하며 *version* 옵션은 전체 버전 정보를 표시할 때 사용합니다.

예:

슬롯 2의 Netra CP3060 보드에 대한 현재 펌웨어 버전을 표시합니다.

```
# clia showhost 2

System Firmware 6.2.5 Netra CP3060 2006/09/15 15:30

Host flash versions:
Hypervisor 1.2.3 2006/08/18 12:25
OBP 4.23.4 2006/08/04 20:46
Netra[TM] CP3060 POST 4.23.4 2006/08/04 21:17

#
```

---

## showunhealthy

구문:

```
showunhealthy
```

용도:

이 명령은 문제가 있는 것으로 보이는 FRU의 목록을 표시합니다. PICMG 3.0 컨텍스트에서 이 목록에는 마지막 핫스왑 상태 변경의 원인이 Communication Lost, Communication lost due to local failure, Unexpected deactivation인 FRU가 포함됩니다.

각 FRU에 대해 IPMB 주소와 FRU 장치 ID, 현재 핫스왑 상태, 이전 핫스왑 상태, 마지막 상태 변경 원인이 표시됩니다.

예:

시스템에서 문제가 있는 구성 요소의 목록을 표시합니다.

```
# clia showunhealthy
IPM Sentry Shelf Manager Command Line Interpreter
There are no unhealthy components in the shelf.
#
```

---

## switchover

구문:

```
switchover
```

용도:

이 명령은 중복 선반 관리자 인스턴스의 스위치오버를 시작합니다. 이 명령은 선반 관리자의 활성 또는 백업 인스턴스에 대해 실행할 수 있습니다.

예:

활성 또는 백업 인스턴스에서 스위치오버를 시작합니다.

```
# clia switchover
This Shelf Manager is now active, but is shutting down to
trigger a switchover.
#
```

---

# terminate

구문:

```
terminate
```

용도:

이 명령은 선반 관리 카드를 재부팅하지 않고 선반 관리자를 종료합니다. 현재 ShMM 이 활성 상태이면 스위치오버가 수행됩니다.

예:

활성 또는 백업 인스턴스에서 선반 관리자를 종료합니다.

```
# clia terminate
#
```

---

# user

구문:

```
user [subcommand]
```

다음과 같은 하위 명령이 지원됩니다.

- add
- delete
- enable
- name
- passwd
- channel

용도:

`user` 명령은 선반 관리자의 RMCP 사용자 계정에 대한 정보를 표시하며 사용자 계정을 간단히 추가, 삭제 및 수정할 수 있는 방법을 제공합니다.

다음 세부절에서는 이 명령의 여러 용도의 `user` 명령 구문에 대해 설명합니다.

## 사용자 정보 표시

구문:

```
user [-v] [userid]
```

용도:

이 명령은 사용자에 대한 정보를 표시합니다. **-v** 옵션을 지정하여 실행하면 비활성 사용자에게 대한 정보도 표시됩니다. 기본적으로 활성 사용자만 나열됩니다. 선택적으로 사용자 ID를 지정하면 해당 ID의 사용자에게 대한 정보만 표시됩니다.

다음과 같은 정보 항목이 표시됩니다.

- 사용자 ID
- 사용자 이름
- 각 IPMI 채널에 대한 채널 액세스 정보: 해당 채널에서 해당 사용자의 최대 권한 수준 및 채널 액세스 플래그

채널 액세스 정보가 여러 채널에 대해 동일한 경우 출력이 결합되고 채널의 범위가 표시됩니다.

예:

```
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
#
# clia user -v
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
7: "TEST1" Disabled
    Channels 0-15 Privilege level: "NO ACCESS"
#
```

## 새 사용자 추가

구문:

```
user add userid user-name channel-access-flags privilege-level password
```

용도:

이 명령은 시스템에 새 사용자를 추가합니다. 명령에 지정된 대로 모든 채널에 대해 동일한 최대 권한 수준과 채널 액세스 플래그를 설정합니다. 지정된 사용자가 없으면 오류가 반환됩니다. 명령 매개 변수의 의미는 다음과 같습니다.

- *userid* - 유효한 사용자 ID
- *user-name* - 사용자 이름(알리지 않고 16자로 자름)
- *channel-access-flag* - SetUserInfo 명령의 첫 번째 바이트(비트 4, 5 및 6만 의미가 있음)
  - 비트 6 - IPMI 메시징 활성화됨
  - 비트 5 - 링크 인증 활성화됨
  - 비트 4 - 콜백으로 제한됨
- *privilege-level* - 사용자 권한 수준
- *password* - 사용자 암호(알리지 않고 16자로 자름)

예:

이름이 root이고, 관리자 권한 수준을 가지며, 암호가 PICMG guru인 사용자 9를 추가합니다.

```
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
#
# clia user add 9 "root" 0x40 4 "PICMG guru"
IPM Sentry Shelf Manager Command Line Interpreter
User 9 added successfully
#
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
9: "root"
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
#
```

## 사용자 삭제

구문:

```
user delete userid
```

용도:

이 명령은 *userid*로 지정된 사용자를 삭제합니다.

예:

사용자 ID가 10인 사용자를 삭제합니다.

```
# clia user delete 10
IPM Sentry Shelf Manager Command Line Interpreter
User 10 deleted successful
#
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
        Flags: "IPMI Messaging"
9: "root"
        Channels 0-15 Privilege level: "Administrator"
        Flags: "IPMI Messaging"
#
```

## 사용자 활성화 및 비활성화

구문:

```
user enable userid 1 | 0
```

용도:

이 명령은 *userid*로 사용자를 활성화하거나 비활성화합니다. 다음과 같이 마지막 명령 매개 변수를 통해 요청된 작업을 지정합니다.

- 0 - 지정된 사용자를 비활성화합니다.
- 제로가 아닌 값 - 지정된 사용자를 활성화합니다.

예:

userid가 9인 사용자를 비활성화 및 활성화합니다.

```
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
9: "root"
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
#
# clia user enable 9 0
IPM Sentry Shelf Manager Command Line Interpreter
    User 9 disabled successfully
#
# clia user -v
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
9: "root" Disabled
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
#
# clia user enable 9 1
IPM Sentry Shelf Manager Command Line Interpreter
    User 9 enabled successfully
#
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
9: "root"
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
#
```



## 사용자 이름 수정

구문:

```
user name userid user-name
```

용도:

이 명령은 지정된 사용자의 사용자 이름을 수정하는 데 사용됩니다. 사용자는 사용자 ID로 지정됩니다. 명령 매개 변수의 의미는 다음과 같습니다.

- *userid* - 유효한 사용자 ID
- *user-name* - 사용자 이름(알리지 않고 16자로 자름)

예:

사용자 9의 이름을 newby로 변경합니다.

```
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
9: "root"
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
#
# clia user name 9 newby
IPM Sentry Shelf Manager Command Line Interpreter
User 9, name changed successfully
#
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
9: "newby"
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
#
```

## 사용자의 암호 수정

구문:

```
user passwd userid password
```

용도:

이 명령은 지정된 사용자의 암호를 수정하는 데 사용됩니다. 사용자는 사용자 ID로 지정됩니다. 명령 매개 변수의 의미는 다음과 같습니다.

- *userid* - 유효한 사용자 ID
- *password* - 사용자 암호(알리지 않고 16자로 자름)

예:

사용자 ID 9의 암호를 RIP로 변경합니다.

```
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
9: "newby"
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
#
# clia user passwd 9 RIP
IPM Sentry Shelf Manager Command Line Interpreter
User 9, password changed successfully
#
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
9: "newby"
    Channels 0-15 Privilege level: "Administrator"
    Flags: "IPMI Messaging"
#
```

## 지정된 사용자 및 지정된 채널의 채널 액세스 설정 수정

구문:

```
user channel userid channel-number channel-access-flags privilege-level
```

용도:

이 명령은 지정된 채널과 사용자의 채널 액세스 설정을 수정하는 데 사용됩니다. 사용자는 사용자 ID로 지정됩니다. 명령 매개 변수의 의미는 다음과 같습니다.

- *userid* - 유효한 사용자 ID
- *channel-number* - 채널 번호
- *channel-access-flags* - SetUserInfo 명령의 첫 번째 바이트(비트 4, 5 및 6만 의미가 있음)
  - 비트 6 - IPMI 메시징 활성화됨
  - 비트 5 - 링크 인증 활성화됨
  - 비트 4 - 콜백으로 제한됨
- *privilege-level* - 사용자 권한 수준

예:

채널 5에서 사용자 9의 최대 권한 수준을 User:로 변경합니다.

```
# clia user 9
IPM Sentry Shelf Manager Command Line Interpreter
9: "newby"
      Channels 0-15 Privilege level: "Administrator"
      Flags: "IPMI Messaging"
#
# clia user channel 9 5 0x60 2
IPM Sentry Shelf Manager Command Line Interpreter
User 9, channel 5 access updated successfully
#
# clia user 9
IPM Sentry Shelf Manager Command Line Interpreter
9: "newby"
      Channels 0-4 Privilege level: "Administrator"
      Flags: "IPMI Messaging"
      Channel 5 Privilege level: "User"
      Flags: "Link Authentication" "IPMI Messaging"
      Channels 6-15 Privilege level: "Administrator"
      Flags: "IPMI Messaging"
#
```

---

# version

구문:

version

용도:

이 명령은 선반 관리자 소프트웨어의 버전 정보를 표시합니다.

예:

```
# clia version
IPM Sentry Shelf Manager Command Line Interpreter
IPM Sentry Shelf Manager ver. 2.1.3
IPM Sentry is a trademark of Pigeon Point Systems.
Copyright (c) 2002-2005 Pigeon Point Systems
Build date/time: April 3 2006 16:39:37
All rights reserved
#
```

# Sun OEM IPMI 명령

이 부록에서 설명하는 명령은 Sun Microsystems에서 설계한 ATCA 보드에만 해당됩니다. Sun Microsystems에 할당된 IANA(Internet Assigned Numbers Authority) 번호는 42입니다.

IANA 번호 할당에 대한 자세한 내용은

<http://www.iana.org/assignments/enterprise-numbers>를 참조하십시오.

이러한 명령에 사용되는 netfunction(NetFn)은 0x2E이며 IPMI 사양에 정의된 OEM netfunction입니다. 이 netfunction에서 요청 패킷의 처음 세 개 데이터 바이트가 이 IANA 번호이고 완료 코드 뒤에 있는 응답 패킷의 처음 세 개 바이트가 IANA 번호입니다. Sun ATCA 노드 보드의 경우 이 세 개 바이트는 00 00 2A입니다.

Sun OEM IPMI 명령은 표 B-1에 나열되어 있고 이후의 해당 절에서 설명합니다.

표 B-1 Sun OEM IPMI 명령

명령	Opcode	구문
Get Version	0x80	#GET_VERSION
Set Boot Page <sup>†</sup>	0x81	#SET_BOOT_PAGE
Get Boot Page <sup>†</sup>	0x82	#GET_BOOT_PAGE
Set Front panel reset button state	0x83	#SET_FP_RESET_BUTTON
Get Front panel reset button state	0x84	#GET_FP_RESET_BUTTON_STATE
Set Ethernet Force Front bit	0x85	#SET_ETH_FORCE_FRONT
Get Ethernet Force Front bit	0x86	#GET_ETH_FORCE_FRONT
Get RTM status	0x88	#GET_RTM_STATUS

<sup>†</sup>Netra CP3010 노드 보드에만 유효

## Get Version

Get Version은 IPMC(IPM Controller) 펌웨어 버전 및 대기 CPLD 버전을 반환합니다. 바이트 8, 9 및 A는 예비용입니다.

명령	NetFn	Opcode	참조
Get Version	0x2E(OEM)	0x80	- -

### 데이터 바이트

유형	바이트	데이터 필드
요청 데이터	바이트1	00
	바이트2	00
	바이트3	2A
응답 데이터	바이트1	완료 코드: 00 = OK C1 = 지원되지 않는 명령 CC = 잘못된 요청 데이터 완료 코드에 대한 자세한 내용은 IPMI 사양을 참조하십시오.
	바이트2	00
	바이트3	00
	바이트4	2A
	바이트5	CPLD 버전
	바이트6	IPMC 펌웨어 REV1 바이트
	바이트7	IPMC 펌웨어 REV2 바이트
	바이트8	예비용(무시)
	바이트9	예비용(무시)
	바이트A	예비용(무시)

### 예(터미널 모드):

```
[B8 00 80 00 00 2A] <-----요청
[BC 00 80 00 00 00 2A 02 02 00 00 00 00] <----응답
```

- IPMC 버전은 다음과 같습니다.

REV1의 하위 니블 . REV2의 상위 니블 . REV2의 하위 니블  
위 예에서 IPMC 버전은 2.0.0입니다.

- CPLD 버전은 다음과 같습니다.
  - > CPLD 버전 바이트의 하위 니블
  - 이 예에서 CPLD 버전은 2입니다.

## Set Boot Page

Set Boot Page는 대기 CPLD에서 부트 페이지 비트를 설정하여 Open Boot PROM의 부트 페이지를 선택합니다. 부트 플래시 장애를 복구할 때 이 기능을 사용할 수 있습니다. 이 명령은 Netra CP3010 노드 보드에만 유효합니다.

명령	NetFn	Opcode	참조
Set Boot Page	0x2E(OEM)	0x81	CPLD 사양 1.0

### 데이터 바이트

유형	바이트	데이터 필드
요청 데이터	바이트1	00
	바이트2	00
	바이트3	2A
	바이트4	부트 페이지 설정. 비트 7 - 2 = 0 비트 1 및 0 = 부트 페이지 번호
응답 데이터	바이트1	완료 코드: 00 = OK C1 = 지원되지 않는 명령 CC = 잘못된 요청 데이터  완료 코드에 대한 자세한 내용은 IPMI 사양을 참조하십시오.
	바이트2	00
	바이트3	00
	바이트4	2A

### 예(터미널 모드):

```
[B8 00 81 00 00 2A 02] <-----요청
[BC 00 81 00 00 00 2A] <-----응답
```

## Get Boot Page

Get Boot Page는 선택된 Open Boot PROM 부트 페이지의 현재 설정과 부트 페이지를 선택하는 하드웨어 스위치의 현재 설정을 반환합니다. 이 명령은 Netra CP3010 노드 보드에만 유효합니다.

명령	NetFn	Opcode	참조
Get Boot Page	0x2E(OEM)	0x82	- -

### 데이터 바이트

유형	바이트	데이터 필드
요청 데이터	바이트1	00
	바이트2	00
	바이트3	2A
응답 데이터	바이트1	완료 코드: 00 = OK C1 = 지원되지 않는 명령 CC = 잘못된 요청 데이터 완료 코드에 대한 자세한 내용은 IPMI 사양을 참조하십시오.
	바이트2	00
	바이트3	00
	바이트4	2A
	바이트5	부트 페이지 설정 비트 7 - 4 = 무시. 0으로 설정. 비트 3, 2 = 하드웨어 스위치 설정. 비트 1, 0 = 현재 설정된 부트 페이지.

### 예(터미널 모드):

```
[B8 00 82 00 00 2A] <-----요청
[BC 00 82 00 00 00 2A 02] <-----응답
```



## Set Front Panel Reset Button State

Set Front panel reset button state는 이 버튼을 누를 때 CPLD에서 전면 패널 재설정을 처리하는 방법을 변경하는 데 사용됩니다. CPLD 전원 공급 기본값은 10입니다.

명령	NetFn	Opcode	참조
Set Front panel reset button state	0x2E(OEM)	0x83	CPLD 사양 1.0

### 데이터 바이트

유형	바이트	데이터 필드
요청 데이터	바이트1	00
	바이트2	00
	바이트3	2A
	바이트4	전면 패널 재설정 버튼 설정. 비트 7 - 2 = 0 비트 1 및 0 = 전면 패널 버튼 상태: 00 = IPMC 재설정 및 CPU에 POR 전달 01 = CPU에 XIR 전달 10 = CPU에 POR 전달 11 = 전면 패널 재설정 버튼 비활성화
응답 데이터	바이트1	완료 코드: 00 = OK C1 = 지원되지 않는 명령 CC = 잘못된 요청 데이터 완료 코드에 대한 자세한 내용은 IPMI 사양을 참조하십시오.
	바이트2	00
	바이트3	00
	바이트4	2A

### 예(터미널 모드):

```
[B8 00 83 00 00 2A 02] <-----요청
[BC 00 83 00 00 00 2A] <-----응답
```

## Get Front Panel Reset Button State

Get Front panel reset button state는 전면 패널 재설정 버튼 처리에 대한 현재 설정을 반환합니다. 기본적으로 CPLD 전원은 10으로 공급됩니다. 즉, 이 버튼을 누르면 CPU에 전원 투입 시 재설정이 전달됩니다.

명령	NetFn	Opcode	참조
Get Front panel reset button state	0x2E(OEM)	0x84	CPLD 사양 1.0

### 데이터 바이트

유형	바이트	데이터 필드
요청 데이터	바이트1	00
	바이트2	00
	바이트3	2A
응답 데이터	바이트1	완료 코드: 00 = OK C1 = 지원되지 않는 명령 CC = 잘못된 요청 데이터 완료 코드에 대한 자세한 내용은 IPMI 사양을 참조하십시오.
	바이트2	00
	바이트3	00
	바이트4	2A
	바이트5	전면 패널 재설정 버튼 설정. 비트 7 - 2 = 0. 비트 1 및 0 = 전면 패널 버튼 상태: 00 = IPMC 재설정 및 CPU에 POR 전달 01 = CPU에 XIR 전달 10 = CPU에 POR 전달 11 = 전면 패널 재설정 버튼 비활성화

### 예(터미널 모드):

```
[B8 00 84 00 00 2A] <-----요청
[BC 00 84 00 00 00 2A 02] <-----응답
```

## Set Ethernet Force Front Bit

Set Ethernet Force Front Bit는 시스템에 후면 전송 모듈이 있더라도 소프트웨어에서 이더넷을 전면 I/O 패널에 강제로 연결하는 데 사용됩니다. 이 비트를 1로 설정하면 이더넷이 전면 I/O 패널에 강제로 연결됩니다.

명령	NetFn	Opcode	참조
Set Ethernet Force Front Bit	0x2E(OEM)	0x85	CPLD 사양 1.0

### 데이터 바이트

유형	바이트	데이터 필드
요청 데이터	바이트1	00
	바이트2	00
	바이트3	2A
	바이트4	이더넷 강제 전면 비트 설정 비트 7 - 1 = 0 비트 0 = 강제 전면 상태(1 = 전면에 강제 이더넷 연결)
응답 데이터	바이트1	완료 코드: 00 = OK C1 = 지원되지 않는 명령 CC = 잘못된 요청 데이터 완료 코드에 대한 자세한 내용은 IPMI 사양을 참조하십시오.
	바이트2	00
	바이트3	00
	바이트4	2A

### 예(터미널 모드):

```
[B8 00 85 00 00 2A 01] <-----요청
[BC 00 85 00 00 00 2A] <-----응답
```

# Get Ethernet Force Front Bit

Get Ethernet Force Front Bit는 이더넷 강제 전면 비트의 현재 설정을 반환합니다.

명령	NetFn	Opcode	참조
Get Ethernet Force Front Bit	0x2E(OEM)	0x86	CPLD 사양 1.0

## 데이터 바이트

유형	바이트	데이터 필드
요청 데이터	바이트1	00
	바이트2	00
	바이트3	2A
응답 데이터	바이트1	완료 코드: 00 = OK C1 = 지원되지 않는 명령 CC = 잘못된 요청 데이터 완료 코드에 대한 자세한 내용은 IPMI 사양을 참조하십시오.
	바이트2	00
	바이트3	00
	바이트4	2A
	바이트5	이더넷 강제 전면 비트 설정 비트 7 - 1 = 0 비트 0 = 이더넷 강제 전면 비트 설정 (1 = 전면에 강제 이더넷 연결)

## 예(터미널 모드):

```
[B8 00 86 00 00 2A] <-----요청
[BC 00 86 00 00 00 2A 01] <-----응답
```

## Get RTM Status

Get RTM Status를 사용하면 시스템에 후면 전환 모듈(RTM)이 있는지 감지할 수 있습니다.

명령	NetFn	Opcode	참조
Get RTM Status	0x2E(OEM)	0x88	CPLD 사양

### 데이터 바이트

유형	바이트	데이터 필드
요청 데이터	바이트1	00
	바이트2	00
	바이트3	2A
응답 데이터	바이트1	완료 코드: 00 = OK C1 = 지원되지 않는 명령 CC = 잘못된 요청 데이터
	바이트2	00
	바이트3	00
	바이트4	2A
	바이트5	RTM 존재 비트 7 - 1 = 0 비트 0 = RTM 존재(0 = RTM 없음, 1 = RTM 감지됨)

### 예(터미널 모드):

```
[B8 00 88 00 00 2A] <-----요청
[BC 00 88 00 00 00 2A 01] <-----응답
```



# 용어 집

---

다음 용어와 약어는 Netra CT 900 서버 관리에 유용합니다.

---

## A

**ATCA** Advanced Telecom Computing Architecture의 약어로, AdvancedTCA라고도 합니다. 차세대 캐리어급 통신 장비에 대한 일련의 업계 표준 사양입니다. AdvancedTCA에는 고속 상호 연결 기술, 차세대 프로세서 및 향상된 안정성, 관리 가능성 및 서비스 가용성의 최신 경향이 통합되어 있기 때문에, 표준화를 통해 최저 비용으로 통신에 최적화된 새로운 블레이드(보드)와 새시(선반) 폼 팩터를 구현할 수 있습니다.

---

## B

**backup shelf  
management card**  
(백업 선반 관리 카드)

선반 관리자 기능을 지원할 수 있는 모든 선반 관리 카드입니다.

**Base channel**  
(기본 채널)

최대 4개의 차동 신호 쌍으로 구성된 기본 인터페이스 내의 물리적 연결입니다. 각 기본 채널은 기본 인터페이스 내에서 슬롯 대 슬롯 연결의 종단점입니다.

**Base switch**  
(기본 스위치)

기본 인터페이스를 지원하는 스위치입니다. 기본 스위치는 선반에 설치된 모든 노드 보드에 10/100/1000BASE-T 패킷 스위칭 서비스를 제공합니다. Netra CT 900 서버에서 기본 스위치는 선반의 물리적 슬롯 7 및 8(논리적 슬롯 1 및 2)에 있으며 모든 노드 슬롯 및 보드에 대한 연결을 지원합니다. 패브릭 인터페이스와 기본 인터페이스를 지원하는 보드를 "스위치"라고도 합니다.

**Base interface**  
(기본 인터페이스)

선반에서 노드 보드와 스위치 간의 10/100 또는 1000BASE-T 연결을 지원하는 데 사용되는 인터페이스입니다. 4개의 다른 신호 쌍을 모든 노드 보드 슬롯과 각 스위치 슬롯 간에 라우팅하여 기본 인터페이스를 지원하려면 미드플레인이 필요합니다(Netra CT 900 서버에서 기본 스위치 슬롯은 물리적 슬롯 7 및 8, 논리적 슬롯 1 및 2임).

---

## D

**data transport interface**  
(데이터 전송 인터페이스)

스위치와 노드 보드에서 페이로드 간의 상호 연결을 제공하기 위한 지점간 인터페이스 및 버스된 신호의 집합입니다.

**Dual Star topology**  
(이중 별형 토폴로지)

2개의 스위치 자원이 네트워크 내의 모든 중단점에 중복 연결을 제공하는 상호 연결 패브릭 토폴로지입니다. 스위치 쌍이 노드 보드 간의 중복 상호 연결을 제공합니다.

---

## E

**Electronic Keying or E-Keying**  
(전자 키잉 또는 E-키잉)

기본 인터페이스, 패브릭 인터페이스, 업데이트 채널 인터페이스 및 전면 보드의 동기화 클럭 연결 간의 호환성을 설명하는 데 사용되는 프로토콜입니다.

**ETSI** European Telecommunications Standards Institute(유럽 통신 표준 협회)



## F

### **Fabric channel**

(패브릭 채널)

패브릭 채널은 2행으로 된, 채널 당 총 8개의 신호 쌍으로 구성됩니다. 따라서 각 커넥터는 보드 연결을 위해 보드에 사용 가능한 최대 5개의 채널을 지원합니다. 채널은 4개의 2쌍 포트 구성으로 구성될 수 있습니다.

### **Fabric interface**

(패브릭 인터페이스)

보드 또는 슬롯 당 15개의 연결을 제공하고, 각 연결은 최대 15개의 다른 슬롯이나 보드와의 연결을 지원하는 최대 8개의 차동 신호 쌍(채널)을 구성하는 영역 2 인터페이스입니다. 미드플레인에는 완전 메시 및 이중 별형 토폴로지를 포함한 다양한 구성에서 패브릭 인터페이스를 지원할 수 있습니다. 패브릭 인터페이스를 지원하는 보드는 패브릭 노드 보드, 패브릭 스위치 또는 메시 활성화 보드로 구성될 수 있습니다. 패브릭 인터페이스의 보드 구현은 PICMG 3.x 보조 사양에 의해 정의됩니다.

### **field-replaceable unit,**

#### **FRU**

(현장 대체 가능 장치)

서비스 관점에서 볼 때 서버의 줄일 수 없는 가장 작은 요소입니다. FRU의 예로는 디스크 드라이브, I/O 카드 및 전원 입력 모듈 등이 있습니다. 카드와 기타 구성 요소가 모두 있는 서버는 FRU가 아닙니다. 하지만 빈 서버는 FRU입니다.

### **frame(프레임)**

하나 이상의 선반을 포함할 수 있는 물리적 또는 논리적 엔티티입니다. 랙이라고도 하며, 밀폐된 경우에는 캐비닛이라고도 합니다.

### **front board(전면 보드)**

PCB와 패널을 포함한 PICMG 3.0 기계적 사양(8U x280mm)을 준수하는 보드입니다. 전면 보드는 영역 1 및 영역 2 미드플레인 커넥터와 연결됩니다. 필요에 따라 영역 3 미드플레인 커넥터와 연결하거나 후면 전환 모듈 커넥터에 직접 연결할 수 있으며 선반의 전면 위치에 설치됩니다.

### **Full channel(전체 채널)**

종단점 간의 8개 차동 신호 쌍을 모두 사용하는 패브릭 채널 연결입니다.

### **Full Mesh topology**

(완전 메시 토폴로지)

선반 내 각 슬롯 쌍 간에 하나의 전용 연결 채널을 제공하기 위해 패브릭 인터페이스 내에서 지원될 수 있는 완전 메시 구성입니다. 완전 메시로 구성된 미드플레인 메시 활성화 보드 또는 이중 별형 배치에 설치된 스위치 및 노드 보드를 지원할 수 있습니다.

---

## H

**hot-swap(핫스왑)** 시스템 작동을 중지하지 않고 주변 장치나 기타 구성 요소를 연결하고 연결 해제할 수 있는 기능입니다. 이 기능은 하드웨어와 소프트웨어 모두에 대한 설계상의 암시가 있을 수 있습니다.

---

## I

**I<sup>2</sup>C** 상호 통합 회로 버스입니다. 다중 마스터, 2선 직렬 버스가 현재 IPMB의 기반으로 사용됩니다.

**IPMB** Intelligent Platform Management Bus의 약어로, 지능형 플랫폼 관리 버스 통신 프로토콜 사양에 설명된 가장 낮은 수준의 하드웨어 관리 버스입니다.

**IPMB-0 hub**  
(IPMB-0 허브) 시스템의 여러 FRU에 다중 방사형 IPMB-0 링크를 제공하는 허브 장치입니다. 예를 들어 IPMB-0 허브는 방사형 IPMB-0 링크가 있는 ShMC에 있습니다.

**IPMB-0 link**  
(IPMB-0 링크) 방사형 토폴로지를 사용하는, IPMB-0 허브와 단일 FRU 간의 물리적 IPMB-0 세그먼트입니다. IPMB-0 허브의 각 IPMB-0 링크는 일반적으로 별도의 IPMB-0 센서와 연관됩니다. IPMB-0 링크는 버스된 토폴로지에서 여러 FRU로 연결될 수도 있습니다.

**IPM controller,**  
**IPMC(IPM 제어기)** ATCA IPMB-0으로 연결되는 FRU의 부분으로, 해당 FRU와 그에 종속된 모든 장치를 나타냅니다.

**IPMI** Intelligent Platform Management Interface의 약어로, 컴퓨터 시스템 요소에 대한 인벤토리 관리, 모니터링, 로깅 및 제어를 제공하기 위한 사양 및 메커니즘입니다. 지능형 플랫폼 관리 인터페이스 사양에 정의되어 있습니다.

---

## L

**logic ground**  
(논리 접지) 보드와 미드프레인에서 보드 간에 흐르는 논리적 수준의 신호에 대한 참조 경로 및 반환 경로로 사용되는 선반 전체 전기망입니다.

---

## M

### **Mesh Enabled board** (메시 활성화 보드)

미드프레임 내의 다른 모든 보드에 대한 연결을 제공하는 보드입니다. 메시 활성화 보드는 패브릭 인터페이스를 지원하며 기본 인터페이스도 지원할 수 있습니다. 메시 활성화 보드는 2 ~ 15개의 패브릭 인터페이스 채널(일반적으로 15개 채널 모두)을 사용하여 선반에 있는 다른 모든 보드에 대한 직접 연결을 지원합니다. 지원되는 채널의 수에 따라 선반 내에서 연결 가능한 최대 보드 수가 결정됩니다. 기본 인터페이스를 사용하지 않는 메시 활성화 보드는 사용 가능한 가장 낮은 번호의 논리적 슬롯에 설치할 수 있습니다. 기본 인터페이스를 지원하는 메시 활성화 보드는 기본 스위치가 될 수 있으며, 이 경우 기본 채널 1 및 2를 지원하고 논리적 슬롯 3 ~ 16에 설치할 수 있습니다. 기본 인터페이스를 지원하는 보드는 10/100/1000BASE-T 이더넷을 지원하기 위한 용도로만 기본 채널 1과 2를 사용합니다.

### **midplane(미드프레임)**

백프레임과 기능적으로 동일한 미드프레임은 서버의 후면에 고정됩니다. CPU 카드, I/O 카드 및 저장 장치가 전면에서 미드프레임에 연결되며, 후면 전환 모듈이 후면에서 미드프레임에 연결됩니다.

---

## N

### **NEBS**

Network Equipment/Building System의 약어로, 미국 통신 제어 사무소에 설치되는 장비에 대한 요구 사항 집합입니다. 이 요구 사항은 직원 안전, 자산 보호 및 운영 지속성을 다룹니다. NEBS 테스트에는 장비에 대한 다양한 진동 스트레스, 화재, 기타 환경 및 품질 메트릭 시험이 포함됩니다. 3가지의 NEBS 준수 수준이 있으며 각 수준은 상위 수준의 초집합입니다. 가장 높은 수준인 NEBS 수준 3은 "극한 환경"에 장비를 안전하게 배포할 수 있음을 인증합니다. 통신 중앙 사무소는 극한 환경으로 간주됩니다.

NEBS 표준은 Telcordia Technologies, Inc.(이전 Bellcore)에서 유지 관리합니다.

### **node board(노드 보드)**

별형 토폴로지 미드프레임에 사용하기 위한 보드로, 미드프레임 내의 스위치에 연결됩니다. 노드 보드는 기본 인터페이스 및 패브릭 인터페이스 중 하나 또는 모두를 지원할 수 있습니다. 패브릭 인터페이스를 지원하는 보드는 패브릭 채널 1과 2를 사용합니다. 기본 인터페이스를 지원하는 보드는 10/100/1000BASE-T 이더넷을 지원하기 위한 용도로만 기본 채널 1과 2를 사용합니다.

### **node slot(노드 슬롯)**

노드 보드만 지원하는 미드프레임 슬롯입니다. 노드 슬롯은 스위치를 지원할 수 없기 때문에, 노드 보드는 논리적 슬롯 1과 2를 차지하지 않습니다. 노드 슬롯은 별형 토폴로지를 지원하도록 설계된 미드프레임에만 적용됩니다. 노드 슬롯은 기본 인터페이스와 패브릭 인터페이스를 모두 지원합니다. 일반적으로 노드 슬롯은 2개 또는 4개의 패브릭 채널과 기본 채널 1 및 2를 지원합니다. 2개의 각 채널 노드 슬롯은 각각 논리적 슬롯 1과 2에 연결됩니다. 4개의 채널 노드 슬롯은 각각 논리적 슬롯 1, 2, 3 및 4에 연결됩니다.

---

## P

**PCI** Peripheral Component Interconnect의 약어로, 주변 장치를 컴퓨터에 연결하기 위한 표준입니다. 20 - 33MHz에서 작동하며 124핀 커넥터를 통해 32비트를 한 번에 전송하거나 188핀 커넥터를 통해 64비트를 한 번에 전송할 수 있습니다. 한 번의 주기에서 데이터 한 단어(또는 버스트 모드에서 여러 단어) 앞에 주소가 전송됩니다.

기술적으로 PCI는 버스가 아니라 브리지 또는 중간 단계입니다. 여기에는 상대적으로 느린 주변 장치에서 CPU를 분리시켜 비동기적으로 작동할 수 있게 하는 버퍼가 포함됩니다.

### physical address

(물리적 주소) FRU의 물리적 슬롯 위치를 정의하는 주소입니다. 물리적 주소는 사이트 유형과 사이트 번호로 구성됩니다.

**PICMG** PCI Industrial Computer Manufacturers Group의 약자로, CompactPCI 표준을 포함하여 통신 및 산업용 컴퓨팅 응용 프로그램을 위한 공개 사양을 개발하는 회사들의 컨소시엄입니다.

---

## R

### rear-access

(후면 액세스) 모든 케이블이 선반의 후면에서 나오는 Netra CT 900 서버용 구성 옵션입니다.

### rear transition module

(후면 전환 모듈) 커넥터를 선반 후면으로 확장하기 위해 Netra CT 900 서버의 후면 액세스 모델에만 사용되는 카드입니다.

### Reliability, Availability, Serviceability, RAS

(신뢰성, 가용성 및 서비스 제공 능력)

서버의 신뢰성, 가용성 및 서비스 제공 능력을 구현하거나 향상시키는 하드웨어 및 소프트웨어 기능입니다.

## S

- shelf(선반)** 미드프레인, 전면 보드, 냉각 장치, 후면 전환 모듈 및 전원 입력 모듈로 구성된 구성 요소 집합입니다. 기존에는 선반을 새시라고도 알려졌습니다.
- shelf address**  
(선반 주소) 관리 도메인 내의 각 선반에 대해 고유 ID를 제공하는, 최대 20바이트 길이의 가변 길이 가변 형식 설명자입니다.
- shelf ground**  
(선반 접지) 프레임에 연결되며 모든 보드에 사용 가능한 안전 접지 및 대지 귀로입니다.
- shelf manager**  
(선반 관리자) AdvancedTCA 선반에서 전원, 냉각 및 상호 연결(전자 키잉으로)의 관리를 담당하는 시스템 엔티티입니다. 선반 관리자는 System Manager Interface와 IPMB-0 간의 메시지도 라우팅하여, 시스템 리포지토리에 대한 인터페이스 및 이벤트 메시지에 대한 응답을 제공합니다. 선반 관리자는 ShMC 또는 시스템 관리 하드웨어에 부분적 또는 전체적으로 배포할 수 있습니다.
- ShMC** Shelf Management Controller의 약어로, 선반 관리자에 필요한 기능도 지원할 수 있는 IPMC입니다.
- SNMP** Simple Network Management Protocol(단순 네트워크 관리 프로토콜).
- star topology**  
(별형 토폴로지) 지원되는 노드 슬롯 간에 연결을 제공하는 하나 이상의 허브 슬롯이 있는 미드프레인 토폴로지입니다.
- switch(스위치)** 별형 토폴로지 미드프레인에 사용하기 위한 보드로, 미드프레인 내의 여러 노드 보드에 연결을 제공합니다. 스위치는 기본 인터페이스 및 패브릭 인터페이스 중 하나 또는 모두를 지원할 수 있습니다. 패브릭 인터페이스를 사용하는 보드는 일반적으로 사용 가능한 15개의 패브릭 채널 모두에 스위칭 자원을 제공합니다. 기본 인터페이스를 지원하는 스위치는 논리적 슬롯 1과 2에 설치되며 16개의 기본 채널을 모두 사용하여 최대 14개의 노드 보드 및 기타 스위치에 10/100/1000BASE-T 이더넷 스위칭 자원을 제공합니다. 선반 관리 카드에 대한 연결을 지원하기 위해 하나의 기본 채널이 할당됩니다.
- switch slot**  
(스위치 슬롯) 별형 토폴로지 미드프레인에서 스위치 슬롯은 논리적 슬롯 1과 2에 있어야 합니다. 스위치 슬롯은 기본 인터페이스와 패브릭 인터페이스를 모두 지원합니다. 논리적 슬롯 1과 2에 있는 스위치 슬롯은 기본 인터페이스와 패브릭 인터페이스 스위치를 모두 지원할 수 있습니다. 논리적 슬롯 1과 2는 패브릭 토폴로지에 관계없이 항상 스위치 슬롯입니다. 이러한 슬롯은 각각 최대 16개의 기본 채널과 최대 15개의 패브릭 채널을 지원합니다.
- system(시스템)** 노드와 스위치, 선반 및 프레임과 같은 구성 요소 중 하나 이상을 포함할 수 있는 관리 엔티티입니다.

---

## U

**U** 44.45mm(1.75인치)와 같은 측정 단위입니다.

### **update channel**

**interface**(업데이트 채널  
인터페이스)

업데이트 채널이라고도 합니다. 두 보드 간에 10개의 차동 신호 쌍으로 구성된 연결을 제공하는 영역 2 인터페이스입니다. 두 보드 간의 이 직접 연결은 상태 정보의 동기화에 사용될 수 있습니다. 보드의 업데이트 채널에 대해 구현되는 전송은 정의되지 않습니다. 업데이트 채널은 단일 공급업체에서 만든 두 개의 유사 기능 보드만 사용할 수 있습니다. 전자 키잉은 드라이버를 활성화하기 전에 업데이트 채널 중단점에 일치하는 전송 프로토콜이 매핑되었는지 확인하는 데 사용됩니다. 미드플레인 이 업데이트 채널을 지원해야 합니다. 보드는 업데이트 채널을 지원할 수 있습니다.

---

## Z

**Zone 1(영역 1)** ATCA 슬롯의 높이 치수를 따라 배치된 선형 공간으로 전원, 관리 및 기타 보조 기능에 할당됩니다.

**Zone 2(영역 2)** ATCA 슬롯의 높이 치수를 따라 배치된 선형 공간으로 데이터 전송 인터페이스에 할당됩니다.

**Zone 3(영역 3)** ATCA 슬롯의 높이 치수를 따라 배치된 선형 공간으로 후면 액세스 시스템의 후면 전환 모듈에 대한 사용자 정의 연결 및 상호 연결을 위해 예약되어 있습니다.

# 색인

---

## A

ATCA 선반, 6  
ATCA(Advanced Telecommunications Computing Architecture), 1

## C

CPLD  
cpldtool 유틸리티, 82  
재프로그래밍, 102

## F

FRU 정보, 65, 67, 69  
IPMI, 65  
Sun, 66  
FRU(Field-Replaceable Unit), 2

## I

IANA(Internet Assigned Numbers Authority), 249  
IP 주소, 21  
RMCP, 22  
IPMB  
논리적 슬롯 번호, 112  
슬롯 번호, 112  
IPMB 주소, 112, 114  
범위, 112  
IPMC, 2

## IPMI

FRU 정보, 65  
FRU 정보 레이아웃, 66  
LAN 인터페이스, 11, 53  
개요, 5  
IPMI 명령, 54

## N

netconsole, 105  
NetFn(netfunction), 249

## O

OpenBoot PROM 펌웨어, 2  
OpenHPI, 47  
/etc/openhpi.conf, 47  
libipmdirect 매개 변수, 48  
구성, 47

## P

POST, 2

## R

RMCP, 10, 11, 21  
주소, 22

RMCP(Remote Management Control Protocol), 53  
rupgrade\_tool, 86, 87, 89

## S

SAP(선반 알람 패널), 3  
    Telco 알람, 3  
    액세스, 3  
    직렬 포트, 3  
    커넥터, 14  
SNMP, 47  
    /etc/snmpd.conf 파일 업데이트, 52  
    액세스 제어, 49  
    트랩 설정, 52  
SNMP 하위 에이전트, 49  
SNMP 하위 에이전트 구성 파일, 47  
SNMPv3 구성, 51  
Solaris 운영 체제, 2  
Sun FRU 정보, 66  
Sun OEM IPMI 명령, 54, 249

## T

Telco 알람, 7  
tip 프로그램, 14

## U

U-Boot, 2, 15, 21  
    기본값 복원, 79  
    변수, 21  
    액세스, 16  
    환경 변수, 16, 20  
USB  
    연결, 23  
    인터페이스, 9

## W

WDT(워치독 타이머), 88, 89, 90

## ㄱ

강제 전환, 9  
관리 작업, 11  
스위칭 패브릭 보드  
    설명, 2  
기본 인터페이스, 3

## ㄴ

네트워크 인터페이스, 3  
노드 보드  
    SAS 포트, 3  
    노드 보드, 타사 참조  
    설명, 2  
    액세스, 3  
    이더넷 포트, 3  
    종료, 105  
    직렬 포트, 3  
    콘솔, 103  
노드 보드, 타사, 3  
    설명, 3

## ㄷ

다시 초기화, 10

## ㄹ

로그인  
    암호 재설정, 81  
    출하 시 기본값, 81  
로그인, 선반 관리 카드, 14

## ㅁ

매핑  
    물리적 주소, 114

## 명령

clia shmstatus, 15, 104  
console, 103, 104  
showcpustate, 60



- switchover, 9
- useradd, 46
- 명령줄 인터페이스, 2
- 물리-논리 슬롯 매핑, 12

人

- 사용자 계정, 11
- 선반 관리 카드, 2, 5, 7
  - 로그인, 14
  - 백업, 14
  - 사용자 계정, 46
  - 설명, 2, 11
  - 세션, 3
  - 액세스, 14
  - 이더넷 포트, 21
  - 재설정, 80
  - 콘솔, 14, 103 - 105
  - 활성, 14
- 선반 관리자, 2, 5, 7
  - CLI 명령, 109, 109 - 248
  - CLI 명령 요약, 56
  - CLI 시작, 55
  - IP 주소, 14
  - shelfman.conf 파일, 28
  - 구성 매개 변수, 29
  - 구성 파일, 23
  - 기능, 7
  - 네트워크 매개 변수, 24
  - 디버그 수준, 122
  - 명령줄 인터페이스, 54
  - 명령줄 인터페이스(CLI), 2
  - 백업, 10
  - 상세 표시 수준, 43
  - 소개, 5
  - 소프트웨어, 2
  - 인터페이스 옵션, 11
  - 재프로그래밍, 82
  - 전환, 8
- 선반 관리자 재프로그래밍, 82
- 선반 정보, 73
- 설명서, xix

- 스위치 보드
  - 액세스, 3
  - 이더넷 포트, 3
- 스위치 카드, 103, 120
- 스크립트
  - /etc/upgrade/step4hsh, 88
- 시간 서버, 45
- 시계 설정, 44
- 시스템 관리자, 6
- 신호
  - Remote Healthy, 9, 10
  - Remote Presence, 9
  - 전환, 9

○

- 안정적인 업그레이드, 83
  - 상태 파일, 85
  - 시나리오, 91
  - 예, 92
  - 유틸리티, 85
  - 플래시 분할, 83
- 안정적인 업그레이드 유틸리티, 91
- 안정적인 업그레이드 절차, 82, 86
- 암호, 47
- 유틸리티
  - 안정적인 업그레이드, 85
- 이더넷 포트, 3, 21

ㄹ

- 전자 키잉, 115
- 전환, 9, 10
  - 강제, 9
  - 협동, 9
- 정상 종료, 105
- 제어된 비활성화, 106, 107
- 종료
  - 노드 보드, 105
  - 수동 정상, 105
- 직렬 포트, 14

## ㄱ

커넥터

선반 알람 패널, 14

콘솔, 14, 103 - 105

콘솔 세션, 103, 121

## ㄷ

텔넷, 14

## ㄴ

파일 시스템

다시 초기화, 80

펌웨어

Open Boot PROM, 2

플래시 메모리, 82

플래시 분할 영역, 83, 84

## ㄹ

하드웨어 인터페이스, 3, 4

협동 전환, 9

확장 인터페이스, 3

환경 FRU, 66

후면 전환 카드, 3