



Servidor Netra™ 240: Guía de administración del sistema

Sun Microsystems, Inc.
www.sun.com

Nº de referencia 817-5012-11
Julio de 2004, Revisión A

Envíe sus comentarios sobre este documento a: <http://www.sun.com/hwdocs/feedback>

Copyright 2004 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, EE.UU. Reservados todos los derechos.

Sun Microsystems, Inc. tiene derechos de propiedad intelectual relacionados con la tecnología que se describe en este documento. Concretamente, y sin limitación alguna, estos derechos de propiedad intelectual pueden incluir una o más patentes de Estados Unidos mencionadas en <http://www.sun.com/patents> y otras patentes o aplicaciones de patentes pendientes en Estados Unidos y otros países.

Este documento y el producto al que hace referencia se distribuyen con licencias que restringen su uso, copia, distribución y descompilación. No se puede reproducir parte alguna de este producto o documento en ninguna forma ni por cualquier medio sin la autorización previa por escrito de Sun y sus licenciadores, si los hubiera.

El software de otras empresas, incluida la tecnología de fuentes, está protegido por el copyright y tiene licencia de los distribuidores de Sun.

Partes del producto pueden derivarse del sistema Berkeley BSD, concedido bajo licencia por la Universidad de California. UNIX es una marca comercial registrada en EE.UU. y otros países, con licencia exclusiva de X/Open Company, Ltd.

Sun, Sun Microsystems, el logotipo de Sun, AnswerBook2, docs.sun.com, Netra, Sun Store, Sun Remote Services Net Connect, OpenBoot y Solaris son marcas comerciales o marcas comerciales registradas de Sun Microsystems, Inc. en EE.UU. y otros países.

Todas las marcas comerciales SPARC se usan bajo licencia y son marcas registradas de SPARC International, Inc. en Estados Unidos y en otros países. Los productos con marcas comerciales SPARC se basan en una arquitectura desarrollada por Sun Microsystems, Inc.

La interfaz gráfica de usuario OPEN LOOK y Sun™ ha sido desarrollada por Sun Microsystems, Inc. para sus usuarios y titulares de licencias. Sun reconoce los esfuerzos de Xerox en promover la investigación y el desarrollo del concepto de interfaces gráficas o visuales de usuario para la industria informática. Sun tiene una licencia no exclusiva de Xerox de la interfaz gráfica de usuario de Xerox, cuya licencia también cubre los titulares de licencias de Sun que implementan las interfaces gráficas de usuario OPEN LOOK y cumplen con los contratos escritos de licencia de Sun.

Derechos del Gobierno de los EE.UU. – Uso comercial. Los usuarios del gobierno de los Estados Unidos están sujetos a los acuerdos de la licencia estándar de Sun Microsystems, Inc. y a las disposiciones aplicables sobre los FAR (derechos federales de adquisición) y sus suplementos.

LA DOCUMENTACIÓN SE PROPORCIONA “TAL CUAL” Y SE RECHAZA CUALQUIER CONDICIÓN, REPRESENTACIÓN O GARANTÍA EXPLÍCITA O IMPLÍCITA, INCLUIDA CUALQUIER GARANTÍA IMPLÍCITA DE COMERCIABILIDAD, ADECUACIÓN A UN PROPÓSITO CONCRETO O NO INFRINGIMIENTO, EXCEPTO EN LO QUE DICHO RECHAZO ESTÉ INVALIDADO LEGALMENTE.



Adobe PostScript

Contenido

Prólogo xi

1. Herramientas para la resolución de problemas 1

Información general sobre las herramientas de diagnóstico 2

Indicadores del sistema 3

Advanced Lights Out Manager 4

Indicadores de estado del servidor 5

▼ Para mostrar el estado del LED de localización 6

▼ Para encender el LED de localización 6

▼ Para apagar el LED de localización 6

Indicadores de estado de alarma 7

Diagnósticos de Comprobación automática al encendido (POST) 9

Control de los diagnósticos de POST 10

▼ Para iniciar los diagnósticos de POST 12

Comandos de OpenBoot 12

Comandos `probe-scsi` y `probe-scsi-all` 13

Comando `probe-ide` 14

Comando `show-devs` 15

▼ Para ejecutar comandos de OpenBoot 15

Diagnósticos de OpenBoot	16
▼ Para iniciar los diagnósticos de OpenBoot	16
Control de las pruebas de diagnósticos de OpenBoot	17
Comandos <code>test</code> y <code>test-all</code>	18
Mensajes de error de diagnósticos de OpenBoot	19
Herramientas de diagnóstico del sistema operativo	20
Archivos de registro de mensajes de error y del sistema	20
Comandos de información del sistema del software de Solaris	20
Comando <code>prtconf</code>	21
Comando <code>prtdiag</code>	22
Comando <code>prtfru</code>	24
Comando <code>psrinfo</code>	25
Comando <code>showrev</code>	26
▼ Para ejecutar comandos de información del sistema de la plataforma Solaris	27
Resultados de las pruebas de diagnóstico recientes	28
▼ Para ver los resultados de las pruebas más recientes	28
Variables de configuración de OpenBoot	28
▼ Para visualizar y establecer las variables de configuración de OpenBoot	29
Uso de los comandos <code>watch-net</code> y <code>watch-net-all</code> para probar las conexiones de red	29
Recuperación automática del sistema	30
Opciones de Auto-Boot	31
Resumen de gestión de errores	32
Casos de reinicio	32
▼ Para habilitar ASR	33
▼ Para inhabilitar ASR	34

2. Software de SunVTS	35
Información general del software SunVTS	35
Pruebas de SunVTS	36
Software SunVTS y seguridad	37
▼ Para determinar si el software SunVTS está instalado	38
Instalación de SunVTS	38
Visualización de la documentación de SunVTS	39
3. Advanced Lights Out Manager	41
Información general de Advanced Lights Out Manager	41
Puertos de ALOM	43
Configuración de la contraseña de admin	43
Funciones básicas de ALOM	44
▼ Para cambiar al indicador de ALOM	44
▼ Para cambiar al indicador de la consola del servidor	44
▼ Para quitar a otro usuario la capacidad de escribir en la consola	45
Reinicio automático del servidor	45
Control y supervisión del entorno	46
A. Interfaz de programación de aplicaciones (API) de salida de relés de alarma	49
Índice	55

Figuras

FIGURA 1-1	Flujo de los indicadores de sistema	4
FIGURA 1-2	Ubicación de los indicadores del panel frontal	5

Tablas

TABLA 1-1	Resumen de herramientas de resolución de problemas	2
TABLA 1-2	Indicadores del estado del servidor (anterior y posterior)	5
TABLA 1-3	Indicadores de alarma y estado de alarma de contacto seco	7
TABLA 1-4	Variables de configuración de OpenBoot	10
TABLA 1-5	Palabras clave de la variable de configuración de OpenBoot <code>test-args</code>	17
TABLA 1-6	Comandos de visualización de información de la plataforma Solaris	27
TABLA 2-1	Pruebas de SunVTS	36
TABLA 3-1	Componentes supervisados por ALOM	42
TABLA 3-2	Umbral de temperatura del alojamiento del servidor Netra 240	47

Prólogo

La guía *Servidor Netra 240: Guía de administración del sistema* está concebida para administradores de sistema experimentados. En ella se ofrece también una descripción general de las herramientas de diagnóstico del servidor Netra™ 240, así como de las diversas tareas de administración de éste.

Para utilizar la información de este manual es imprescindible conocer los conceptos y los términos referidos a las redes de sistemas y tener conocimientos avanzados del sistema operativo Solaris™ (SO Solaris).

Antes de leer este manual

En este manual no se trata de la instalación del servidor ni de su montaje en bastidor. Para obtener información acerca de esos temas, consulte *Servidor Netra 240: Guía de instalación* (número de referencia 817-4995-11).

Antes de llevar a cabo los procedimientos indicados en esta guía, lea *Important Safety Information for Sun Hardware Systems* (número de referencia 816-7190-10).

Utilización de los comandos UNIX

Utilice esta sección para avisar a todos los usuarios de que no se proporcionan todos los comandos UNIX. Por ejemplo:

Es posible que este documento no contenga información sobre los procedimientos y comandos básicos UNIX®, como, por ejemplo, cierre e inicio del sistema y configuración de los dispositivos. Consulte las siguientes secciones para obtener esta información:

- Manuales de software que acompañen al sistema
- La documentación del entorno operativo Solaris™, que se encuentra en:
<http://docs.sun.com>

Indicadores de los shell

Shell	Indicador
Shell de C	<i>nombre-máquina%</i>
Shell de C para superusuario	<i>nombre-máquina#</i>
Shells de Bourne y de Korn	\$
Shells de Bourne y de Korn para superusuario	#

Convenciones tipográficas

Fuente ⁱ	Significado	Ejemplos
AaBbCc123	Nombres de comandos, archivos y directorios que aparecen en la -pantalla del sistema	Modifique el archivo <code>.login</code> . Utilice el comando <code>ls -a</code> para ver la lista de todos los archivos. <code>% Ha recibido correo</code>
AaBbCc123	Expresiones que el usuario debe escribir, en contraste con la salida proporcionada por el PC	<code>% su</code> Password:
<i>AaBbCc123</i>	Títulos de manuales, términos o expresiones nuevas, palabras que deben resaltarse. Sustituya las variables de la línea de comandos por nombres o valores reales.	Lea el capítulo 6 de la <i>Guía del usuario</i> . A éstas se las denomina opciones de <i>clase</i> . Es <i>necesario</i> ser usuario root para efectuar esta operación. Para eliminar un archivo, escriba <code>rm nombre_archivo</code> .

ⁱ La configuración del navegador puede diferir de esta configuración.

Documentación relacionada

Aplicación	Título	Número de referencia
Información general sobre la instalación	<i>Netra 240 Server Quick Start Guide</i>	817-3904-xx
Últimas actualizaciones del producto	<i>Netra 240 Server Release Notes</i>	817-3142-xx
Cumplimiento de normativas y seguridad	<i>Important Safety Information for Sun Hardware Systems</i>	816-7190-10
	<i>Servidor Netra 240: Manual del cumplimiento de normativas y seguridad</i>	817-5018-11
Ubicación de la sede web de documentación	<i>Sun Netra 240 Server Documentation</i>	817-2697-10
Instalación	<i>Servidor Netra 240: Guía de instalación</i>	817-4995-11
LOM (Lights Out Management)	<i>Software Sun Advanced Lights Out Manager: Guía del usuario del servidor Netra 240</i>	817-5004-11
Mantenimiento	<i>Netra 240 Server Service Manual</i>	817-2699-xx

Acceso a la documentación de Sun

Puede ver, imprimir o adquirir una amplia selección de documentación de Sun, incluidas las versiones adaptadas a los entornos nacionales en:

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

Sitios web de terceros

Sun no se responsabiliza de la disponibilidad de sitios web de terceros mencionados en este documento. Sun no aprueba el contenido, la publicidad, los productos u otros materiales disponibles en estos sitios o recursos ni es responsable respecto a ellos. Sun no será responsable de los daños o pérdidas reales o supuestos provocados por el uso de estos contenidos, bienes o servicios o relacionados con ellos.

Contacto con el Servicio técnico de Sun

Si tiene preguntas técnicas sobre este producto a las que no obtiene respuesta en este documento, consulte el URL siguiente:

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

Sun agradece sus comentarios

En Sun estamos interesados en mejorar nuestra documentación, lo que significa que cualquier comentario o sugerencia al respecto por parte de los usuarios nos resulta de gran utilidad. Puede enviar los comentarios en:

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

Incluya el título y el número de referencia del documento con el comentario:

Servidor Netra 240: Guía de administración del sistema, número de referencia 817-5012-11

Herramientas para la resolución de problemas

En este capítulo se describen las herramientas de diagnóstico disponibles en el servidor Netra 240. El capítulo contiene las secciones siguientes:

- “Información general sobre las herramientas de diagnóstico” en la página 2
- “Indicadores del sistema” en la página 3
- “Advanced Lights Out Manager” en la página 4
- “Diagnósticos de Comprobación automática al encendido (POST)” en la página 9
- “Comandos de OpenBoot” en la página 12
- “Diagnósticos de OpenBoot” en la página 16
- “Herramientas de diagnóstico del sistema operativo” en la página 20
- “Resultados de las pruebas de diagnóstico recientes” en la página 28
- “Variables de configuración de OpenBoot” en la página 28
- “Recuperación automática del sistema” en la página 30

Información general sobre las herramientas de diagnóstico

Sun proporciona una amplia gama de herramientas de diagnóstico para su uso con el servidor Netra 240, como se indica en la tabla siguiente.

TABLA 1-1 Resumen de herramientas de resolución de problemas

Herramienta de diagnóstico	Escriba	Descripción	Accesibilidad y disponibilidad	Capacidad de uso remoto
ALOM	Hardware y software	Supervisa las condiciones de entorno, aísla errores básicos y facilita el acceso a la consola remota.	Puede operar con alimentación de reserva y sin el sistema operativo.	Está diseñado para el acceso remoto.
LED	Hardware	Indican el estado de todo el sistema y de componentes determinados.	Se accede a ellos desde la carcasa del sistema. Están disponibles siempre que el sistema reciba alimentación.	Local, pero pueden visualizarse mediante ALOM.
Comprobación automática al encendido (POST)	Firmware	Prueba los componentes centrales del sistema.	Se ejecuta automáticamente al arrancar. Está disponible cuando no funciona el sistema operativo.	Local, pero pueden visualizarse mediante ALOM.
Comandos de OpenBoot	Firmware	Muestran distintos tipos de información del sistema.	Está disponible cuando no funciona el sistema operativo.	Local, pero se puede obtener acceso a ellos mediante ALOM.

TABLA 1-1 Resumen de herramientas de resolución de problemas (*Continuación*)

Herramienta de diagnóstico	Escriba	Descripción	Accesibilidad y disponibilidad	Capacidad de uso remoto
Diagnósticos de OpenBoot	Firmware	Prueba los componentes del sistema y se centra en los periféricos y en los dispositivos de E/S.	Se ejecuta de forma automática o interactiva. Está disponible cuando no funciona el sistema operativo.	Local, pero pueden visualizarse mediante ALOM.
Comandos del software de Solaris	Software	Muestran distintos tipos de información del sistema.	Precisa el sistema operativo.	Local, pero se puede obtener acceso a ellos mediante ALOM.
Software de SunVTS™	Software	Somete el sistema a pruebas de funcionamiento en paralelo.	Precisa el sistema operativo. Paquete opcional.	Se puede visualizar y controlar en una red.

Indicadores del sistema

El servidor Netra 240 utiliza los siguientes indicadores de servidor predeterminados:

- `ok`: indicador de PROM de OpenBoot
- `sc>`: indicador de Advanced Lights Out Manager (ALOM)
- `#`: indicador de superusuario del software de Solaris (shells Bourne y Korn)

La FIGURA 1-1 muestra la relación entre los tres indicadores y cómo pasar de uno al otro.

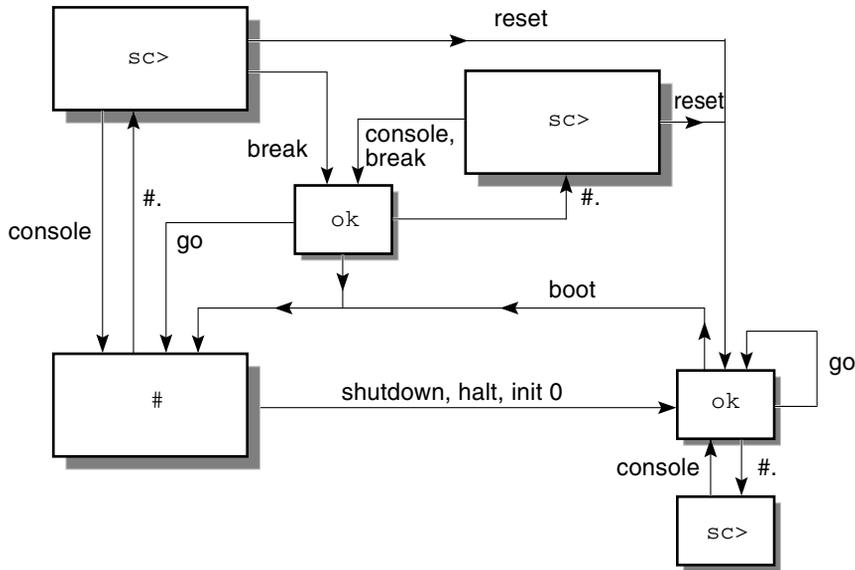


FIGURA 1-1 Flujo de los indicadores de sistema

El diagrama de flujo de la FIGURA 1-1 contiene los indicadores siguientes:

- Comandos de ALOM: `console`, `reset`, `break`
- Secuencia de escape: `#.`
- Comandos del software de Solaris: `shutdown`, `halt`, `init 0`
- Comandos de OpenBoot: `go`, `boot`

Advanced Lights Out Manager

Sun™ Advanced Lights Out Manager (ALOM) para el servidor Netra 240 incorpora diversos indicadores LED de estado. En esta sección se detalla el significado de su estado y la forma de encenderlos y apagarlos. Para obtener más información sobre ALOM, consulte el Capítulo 3.

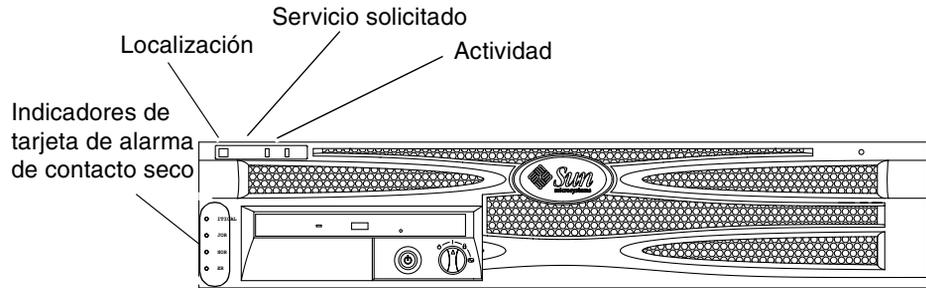


FIGURA 1-2 Ubicación de los indicadores del panel frontal

Indicadores de estado del servidor

El servidor dispone de tres indicadores LED de estado situados en la cubierta frontal (FIGURA 1-2) y repetidos en el panel posterior. La TABLA 1-2 proporciona un resumen de los indicadores.

TABLA 1-2 Indicadores del estado del servidor (anterior y posterior)

Indicador	Color del LED	Estado del LED	Significado
Actividad	Verde	Encendido	El servidor está en marcha y ejecutando el sistema operativo Solaris.
		Apagado	El servidor no está en marcha o no se está ejecutando el sistema operativo Solaris.
Servicio solicitado	Amarillo	Encendido	El servidor ha detectado algún problema y precisa atención del personal de mantenimiento.
		Apagado	El servidor no ha detectado errores.
Localización	Blanco	Encendido	Cuando se utiliza el comando <code>setlocator</code> se enciende una luz continua para distinguir el servidor de los otros servidores del bastidor.

Puede encender y apagar el LED de localización desde la consola del sistema o usando la interfaz de la línea de comandos de ALOM.

▼ Para mostrar el estado del LED de localización

- Siga uno de estos procedimientos:
 - Como superusuario, escriba:

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

- En la interfaz de línea de comandos de ALOM, escriba:

```
sc> showlocator
```

▼ Para encender el LED de localización

- Siga uno de estos procedimientos:
 - Como superusuario, escriba:

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

- En la interfaz de línea de comandos de ALOM, escriba:

```
sc> setlocator on
```

▼ Para apagar el LED de localización

- Siga uno de estos procedimientos:
 - Como superusuario, escriba:

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

- En la interfaz de línea de comandos de ALOM, escriba:

```
sc> setlocator off
```

Indicadores de estado de alarma

La tarjeta de alarma de contacto seco dispone de cuatro indicadores LED compatibles con ALOM situados verticalmente en la cubierta frontal (FIGURA 1-2). La TABLA 1-3 contiene información acerca de los indicadores de alarma y los estados de alarma de contacto seco. Para obtener más información sobre los indicadores de alarma, consulte *Software Sun Advanced Lights Out Manager Software: Guía del usuario del servidor Netra 240* (número de referencia 817-5004-11). Para obtener más información acerca de la API de control de los indicadores de alarma, consulte el Apéndice A.

TABLA 1-3 Indicadores de alarma y estado de alarma de contacto seco

Etiquetas de indicador y relé	Color del indicador	Estado de la aplicación o del servidor	Condición o acción	Estado del indicador del sistema	Estado del indicador de alarma	Relé NC ^{iv} Estado	Relé NO ^v Estado	Comentarios
Crítico (Alarm0)	Rojo	Estado del servidor (Encendido/apagado y SO Solaris funcional/no funcional)	Sin entrada de alimentación.	Apagado	Apagado	Cerrado	Abierto	Estado predeterminado.
			Sistema apagado.	Apagado	Apagado ⁱⁱⁱ	Cerrado	Abierto	Alimentación de entrada conectada.
			El sistema se enciende, SO Solaris no está totalmente cargado.	Apagado	Apagado ⁱⁱⁱ	Cerrado	Abierto	Estado transitorio.
			SO Solaris cargado satisfactoriamente.	Encendido	Apagado	Abierto	Cerrado	Estado de funcionamiento normal.
			Tiempo de espera de vigilancia agotado.	Apagado	Encendido	Cerrado	Abierto	Estado transitorio; reorganizar SO Solaris.
Apagado del SO Solaris iniciado por el usuario ⁱ .	Apagado	Apagado ⁱⁱⁱ	Cerrado	Abierto	Estado transitorio.			

TABLA 1-3 Indicadores de alarma y estado de alarma de contacto seco (Continuación)

Etiquetas de indicador y relé	Color del indicador	Estado de la aplicación o del servidor	Condición o acción	Estado del indicador del sistema	Estado del indicador de alarma	Relé NC ^{iv} Estado	Relé NO ^v Estado	Comentarios
Crítico (Alarm0) (Cont.)	Rojo (Cont.)	Estado del servidor (Encendido/apagado y SO Solaris funcional/no funcional) (Cont.)	Se ha perdido la alimentación de entrada.	Apagado	Apagado	Cerrado	Abierto	Estado predeterminado.
			El usuario apaga la alimentación.	Apagado	Apagado ⁱⁱⁱ	Cerrado	Abierto	Estado transitorio.
			El usuario activa la alarma crítica ⁱⁱ .	—	Encendido	Cerrado	Abierto	Fallo crítico detectado.
			El usuario desactiva la alarma crítica ⁱⁱ .	—	Apagado	Abierto	Cerrado	Fallo crítico resuelto.
Principal (Alarm1)	Rojo	Estado de aplicación	El usuario activa la alarma principal ⁱⁱ .	—	Encendido	Abierto	Cerrado	Fallo principal detectado.
			El usuario desactiva la alarma principal ⁱⁱ .	—	Apagado	Cerrado	Abierto	Fallo principal resuelto.
Secundario (Alarm2)	Ámbar	Estado de aplicación	El usuario activa la alarma secundaria ⁱⁱ .	—	Encendido	Abierto	Cerrado	Fallo secundario detectado.
			El usuario desactiva la alarma secundaria ⁱⁱ .	—	Apagado	Cerrado	Abierto	Fallo secundario resuelto.
Usuario (Alarm3)	Ámbar	Estado de aplicación	El usuario activa la alarma de usuario ⁱⁱ .	—	Encendido	Abierto	Cerrado	Fallo de usuario detectado.
			El usuario desactiva la alarma de usuario ⁱⁱ .	—	Apagado	Cerrado	Abierto	Fallo de usuario resuelto.

i El usuario puede apagar el sistema mediante comandos como `init0` e `init6`. Esto no incluye el apagado de la alimentación del sistema.

ii A partir de una determinación de las condiciones de fallo, el usuario puede conectar la alarma mediante la API de alarma de la plataforma Solaris o la interfaz de línea de comandos de ALOM. Para obtener más información sobre la alarma API, vea el Apéndice A, y para obtener más información sobre ALOM CLI, consulte *Software Sun Advanced Lights Out Manager: Guía del usuario del servidor Netra 240* (número de referencia 817-5004-11).

iii La implementación de este estado de indicador de alarma puede variar.

iv Estado NC significa estado normalmente cerrado. Este estado representa el modo predeterminado de los contactos de relé en estado normalmente cerrado.

v Estado NO significa estado normalmente abierto. Este estado representa el modo predeterminado de los contactos de relé en estado normalmente abierto.

Siempre que el usuario activa una alarma, se muestra un mensaje en la consola. Por ejemplo, cuando se activa la alarma crítica se muestra el siguiente mensaje en la consola:

```
SC Alert: CRITICAL ALARM is set
```

Tenga en cuenta que, en ciertos casos, al activarse una alarma crítica no se enciende el indicador de alarma asociado. Esta implementación es susceptible de variar en próximas versiones (consulte la nota al pie ⁱⁱⁱ de la TABLA 1-3).

Diagnósticos de Comprobación automática al encendido (POST)

Comprobación automática al encendido (POST) es un programa de firmware que ayuda a determinar si hay fallos en alguna parte del sistema. POST comprueba las funciones centrales del sistema, incluidos los módulos de CPU, la placa base, la memoria y ciertos dispositivos de E/S integrados en la placa. A continuación, el software genera mensajes que pueden ser útiles para determinar la naturaleza de un fallo de hardware. POST se puede ejecutar aunque el sistema no pueda arrancar.

POST detecta la mayoría de los fallos del sistema y se encuentra en la OpenBoot PROM de la placa base. Puede programar el software OpenBoot para ejecutar POST durante el encendido configurando dos variables de entorno: `diag-switch?` y el indicador `diag-level`. Ambas se guardan en la tarjeta de configuración del sistema.

POST se ejecuta automáticamente al conectar la alimentación del sistema o después de un reinicio automático, siempre que se cumplan las condiciones siguientes:

- `diag-switch?` está configurada como `true` (el valor predeterminado es `false`).
- `diag-level` está configurada como `min`, `max` o `menus` (el valor predeterminado es `min`).
- `post-trigger` coincide con el tipo de reinicio (el valor predeterminado es `power-on-reset`).

Si el valor de `diag-level` es `min` o `max`, POST efectúa una comprobación breve o ampliada, respectivamente.

Si `diag-level` se configura como `menus`, se visualiza un menú con todas las pruebas ejecutadas al arrancar.

Los mensajes de diagnósticos y errores de POST se muestran en una consola.

Control de los diagnósticos de POST

Los diagnósticos de POST (y otros aspectos del proceso de arranque) se controlan mediante las variables de configuración de OpenBoot. Los cambios en las variables de la configuración de OpenBoot surten efecto después de reiniciar el sistema. La TABLA 1-4 enumera las variables más útiles e importantes. En la sección “Para visualizar y establecer las variables de configuración de OpenBoot” en la página 29 encontrará instrucciones para modificar las variables de configuración de OpenBoot.

TABLA 1-4 Variables de configuración de OpenBoot

Variables de configuración de OpenBoot	Descripción y palabras clave
auto-boot	Determina si el sistema operativo se inicia automáticamente. El valor predeterminado es true. <ul style="list-style-type: none">• true: el sistema operativo se inicia automáticamente una vez terminadas las pruebas del firmware.• false: el sistema se queda en el indicador ok hasta que el usuario escriba boot.
diag-level	Determina el nivel o el tipo de los diagnósticos ejecutados. El valor predeterminado es min. <ul style="list-style-type: none">• off: no se ejecuta ninguna prueba.• min: sólo se ejecutan las pruebas básicas.• max: se pueden ejecutar pruebas más completas, según el dispositivo.• menus: las pruebas controladas por menú en los niveles de POST se pueden ejecutar de forma individual.
diag-script	Determina los dispositivos comprobados por los diagnósticos de OpenBoot. El valor predeterminado es none. <ul style="list-style-type: none">• none: no se efectúa ninguna prueba en los dispositivos.• normal: se efectúan pruebas en los dispositivos incorporados (basados en el plano central) que disponen de comprobaciones automáticas.• all: se efectúan pruebas en todos los dispositivos que disponen de comprobaciones automáticas.
diag-switch?	Hace que el sistema entre y salga del modo de diagnóstico. El valor predeterminado es false. <ul style="list-style-type: none">• true: modo de diagnóstico: se ejecutan los diagnósticos de POST y las pruebas de diagnóstico de OpenBoot.• false: modo predeterminado: no ejecutar pruebas de diagnóstico de POST ni de OpenBoot.

TABLA 1-4 Variables de configuración de OpenBoot (*Continuación*)

Variables de configuración de OpenBoot	Descripción y palabras clave
<code>post-trigger</code> <code>obdiag-trigger</code>	<p>Estas dos variables especifican la clase de evento de reinicio que causa la ejecución de las comprobaciones automáticas al encendido (o los diagnósticos de OpenBoot). Estas variables pueden aceptar una sola palabra clave, así como combinaciones de las tres primeras palabras clave separadas por espacios. Para obtener más detalles, consulte “Para visualizar y establecer las variables de configuración de OpenBoot” en la página 29.</p> <ul style="list-style-type: none">• <code>error-reset</code>: reinicio causado por determinadas condiciones de error de hardware irrecuperables. En general, un reinicio de error tiene lugar cuando algún problema de hardware deteriora los datos de estado del sistema. Entre los ejemplos se incluyen reinicios de vigilancia del sistema y de CPU, y ciertos eventos de reinicio de CPU (predeterminados).• <code>power-on-reset</code>: reinicio debido a que se ha pulsado el botón de Encendido/En espera (predeterminado).• <code>user-reset</code>: reinicio iniciado por el usuario o el sistema operativo.• <code>all-resets</code>: cualquier tipo de reinicio del sistema.• <code>none</code>: no se ejecuta ninguna prueba de diagnóstico de la POST ni de OpenBoot.
<code>input-device</code>	<p>Selecciona de dónde proviene la entrada de la consola. El valor predeterminado es <code>ttya</code>.</p> <ul style="list-style-type: none">• <code>ttya</code>: del puerto SERIAL MGT incorporado.• <code>ttyb</code>: del puerto serie de uso general incorporado (10101).• <code>keyboard</code>: del teclado conectado que forma parte del terminal gráfico.
<code>output-device</code>	<p>Selecciona dónde se muestran los diagnósticos y otros tipos de salida de la consola. El valor predeterminado es <code>ttya</code>.</p> <ul style="list-style-type: none">• <code>ttya</code>: en el puerto SERIAL MGT incorporado.• <code>ttyb</code>: en el puerto serie de uso general incorporado (10101).• <code>screen</code>: en la pantalla conectada que forma parte del terminal gráfico.ⁱ

ⁱ Los mensajes de POST no se pueden mostrar en un terminal gráfico. Se envían a `ttya` incluso si `output-device` está establecido en `screen`.

Nota: Estas variables afectan tanto a las pruebas de diagnósticos de OpenBoot como a los diagnósticos de POST.

Una vez finalizada la ejecución de los diagnósticos de POST, éste informa al firmware de POST del estado de cada una de las pruebas ejecutadas. A continuación, el control vuelve al código de firmware de OpenBoot.

Si los diagnósticos de POST no revelan ningún error y aun así el servidor no arranca, ejecute las pruebas de diagnóstico de OpenBoot.

▼ Para iniciar los diagnósticos de POST

1. Vaya al indicador `ok`.

2. Escriba:

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

3. Escriba:

```
ok setenv diag-level valor
```

Donde *valor* es *min*, *max* o *menus*, en función de la información de diagnóstico que desee ver.

4. Escriba:

```
ok reset-all
```

El sistema ejecuta los diagnósticos de POST si el valor de `post-trigger` es `user-reset`. Los mensajes de estado y de error se muestran en la ventana de la consola. Si POST detecta un error, visualiza un mensaje de error en el que se describe el fallo.

5. Al finalizar la ejecución de POST, vuelva a configurar `diag-switch?` como `false`; para ello, escriba:

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

Restablecer el valor de `diag-switch?` en `false` minimiza el tiempo de arranque.

Comandos de OpenBoot

Los comandos de OpenBoot se escriben en el indicador `ok`. Los que pueden proporcionar información útil para el diagnóstico son:

- `probe-scsi` y `probe-scsi-all`
- `probe-ide`
- `show-devs`

Comandos `probe-scsi` y `probe-scsi-all`

Los comandos `probe-scsi` y `probe-scsi-all` sirven para diagnosticar problemas en los dispositivos SCSI.



Precaución: Si ha utilizado el comando `halt` o la secuencia de teclas Stop-A para acceder al indicador `ok`, la ejecución de los comandos `probe-scsi` o `probe-scsi-all` puede bloquear el sistema.

El comando `probe-scsi` se comunica con todos los dispositivos SCSI conectados a los controladores incorporados de SCSI. El comando `probe-scsi-all` también accede a los dispositivos conectados a cualquier adaptador de host instalado en las ranuras PCI.

En cualquier dispositivo SCSI conectado y activado, los comandos `probe-scsi` y `probe-scsi-all` muestran el ID del bucle, el adaptador de host, el número de unidad lógica y el World Wide Name (WWN) exclusivo, así como una descripción del dispositivo que contiene el tipo y el fabricante.

A continuación se muestra un ejemplo de salida del comando `probe-scsi`.

CÓDIGO EJEMPLO 1-1 Salida del comando `probe-scsi`

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

A continuación se muestra un ejemplo de salida del comando `probe-scsi-all`.

CÓDIGO EJEMPLO 1-2 Salida del comando `probe-scsi-all`

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

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

{1} ok
```

Comando probe-ide

El comando `probe-ide` se comunica con todos los dispositivos IDE (Integrated Drive Electronics) conectados al bus IDE. Se trata de un bus interno del sistema para dispositivos de soporte como la unidad de DVD.



Precaución: Si ha utilizado el comando `halt` o la secuencia de teclas Stop-A para acceder al indicador `ok`, la ejecución del comando `probe-ide` puede bloquear el sistema.

A continuación se muestra un ejemplo de salida del comando `probe-ide`.

CÓDIGO EJEMPLO 1-3 Comando de salida probe-ide

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

Device 1 ( Primary Slave )
        Not Present

Device 2 ( Secondary Master )
        Not Present

Device 3 ( Secondary Slave )
        Not Present

{1} ok
```

Comando show-devs

El comando `show-devs` muestra una lista de las rutas de todos los dispositivos de hardware del árbol de dispositivos. El CÓDIGO EJEMPLO 1-4 muestra un ejemplo de salida.

CÓDIGO EJEMPLO 1-4 Comando de salida `show-devs`

```
/pci@1d,700000
/pci@1c,600000
/pci@1e,600000
/pci@1f,700000
/memory-controller@1,0
/SUNW,UltraSPARC-IIIi@1,0
/memory-controller@0,0
/SUNW,UltraSPARC-IIIi@0,0
/virtual-memory
/memory@m0,0
/aliases
/options
/openprom
/chosen
/packages
/pci@1d,700000/network@2,1
/pci@1d,700000/network@2
/pci@1c,600000/scsi@2,1
/pci@1c,600000/scsi@2
/pci@1c,600000/scsi@2,1/tape
/pci@1c,600000/scsi@2,1/disk
/pci@1c,600000/scsi@2/tape
/pci@1c,600000/scsi@2/disk
/pci@1e,600000/ide@d
/pci@1e,600000/usb@a
/pci@1e,600000/pmu@6
/pci@1e,600000/isa@7
/pci@1e,600000/ide@d/cdrom
/pci@1e,600000/ide@d/disk.....
```

▼ Para ejecutar comandos de OpenBoot

1. **Detenga el sistema para acceder al indicador `ok`.**
Antes de apagar el sistema, informe de ello a los usuarios.
2. **Escriba el comando correspondiente en el indicador de la consola.**

Diagnósticos de OpenBoot

Al igual que los diagnósticos de POST, el código de diagnósticos de OpenBoot se basa en el firmware y se encuentra en la PROM de arranque.

▼ Para iniciar los diagnósticos de OpenBoot

1. Escriba:

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

2. Escriba:

```
ok obdiag
```

Este comando muestra el mensaje de diagnósticos de OpenBoot.

```
ok obdiag
```

```
o b d i a g
```

1 i2c@0,320	2 ide@d	3 network@2
4 network@2,1	5 rtc@0,70	6 scsi@2
7 scsi@2,1	8 serial@0,2e8	9 serial@0,3f8
10 usb@a	11 usb@b	12 flashprom@2,0

```
Commands: test test-all except help what printenvs setenv versions exit
```

Nota: Si hay instalada una tarjeta PCI en el servidor, en el menú `obdiag` aparecerán pruebas adicionales.

3. Escriba:

```
obdiag> test n
```

Donde *n* representa el número de la prueba que desea ejecutar.

Está disponible un resumen de las pruebas. En el indicador `obdiag>`, escriba:

```
obdiag> help
```

Control de las pruebas de diagnósticos de OpenBoot

La mayoría de las variables de configuración de OpenBoot que se utilizan para controlar la POST (consulte la TABLA 1-4) también afectan a las pruebas de diagnósticos de OpenBoot.

- Utilice la variable `diag-level` para controlar el nivel de comprobación de los diagnósticos de OpenBoot.
- Utilice `test-args` para personalizar la ejecución de las pruebas.

De manera predeterminada, `test-args` está establecido para contener una cadena vacía. Para modificar `test-args` puede utilizar una o más de las palabras clave reservadas que se enumeran en la TABLA 1-5.

TABLA 1-5 Palabras clave de la variable de configuración de OpenBoot `test-args`

Palabra clave	Descripción
<code>bist</code>	Ejecuta la comprobación automática incorporada (BIST) en dispositivos externos y periféricos.
<code>debug</code>	Muestra todos los mensajes de depuración.
<code>iopath</code>	Verifica la integridad de interconexión y del bus.
<code>loopback</code>	Comprueba el funcionamiento de la ruta de bucle de retorno externo del dispositivo.
<code>media</code>	Verifica la accesibilidad a los soportes de dispositivos periféricos y externos.
<code>restore</code>	Intenta restaurar el estado original del dispositivo si se ha producido un error en la ejecución previa de la prueba.
<code>silent</code>	En lugar del estado de cada prueba, sólo muestra los errores.

TABLA 1-5 Palabras clave de la variable de configuración de OpenBoot `test-args`
(Continuación)

Palabra clave	Descripción
<code>subtests</code>	Muestra la prueba principal y cada una de las subpruebas llamadas.
<code>verbose</code>	Muestra mensajes de estado detallados en todas las pruebas.
<code>callers=<i>n</i></code>	Muestra el seguimiento hacia atrás de <i>N</i> peticionarios cuando se produce un error: <code>callers=0</code> : muestra el seguimiento hacia atrás de todos los peticionarios antes del error.
<code>errors=<i>n</i></code>	Sigue ejecutando la prueba hasta que se encuentran <i>N</i> errores: <code>errors=0</code> : muestra todos los informes de errores sin terminar las pruebas.

Si desea personalizar las pruebas de diagnósticos de OpenBoot, puede establecer `test-args` en una lista de palabras clave separadas por comas, como se muestra en el ejemplo siguiente:

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

Comandos `test` y `test-all`

También puede ejecutar las pruebas de los diagnósticos OpenBoot desde el indicador `ok`. Para ello, escriba el comando `test`, seguido de la ruta completa de hardware del dispositivo (o conjunto de dispositivos) en el que se tengan que efectuar las pruebas. Por ejemplo:

```
ok test /pci@x,y/SUNW,q1c@2
```

Para personalizar una prueba individual, puede utilizar `test-args` como se indica a continuación:

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

Esta sintaxis afecta únicamente a la prueba actual sin cambiar el valor de la variable de configuración de OpenBoot `test-args`.

Puede efectuar pruebas en todos los dispositivos del árbol de dispositivos con el comando `test-all`:

```
ok test-all
```

En caso de que especifique un argumento de ruta a `test-all`, sólo se efectúan pruebas en el dispositivo especificado y en los dispositivos subordinados. En el ejemplo siguiente se muestra el comando que sirve para probar el bus USB y todos los dispositivos con comprobaciones automáticas conectados a dicho bus:

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

Mensajes de error de diagnósticos de OpenBoot

Los resultados de error de los diagnósticos de OpenBoot se notifican en forma de tabla que contiene un breve resumen del problema, el dispositivo de hardware afectado, la subprueba en la que se ha producido el error y otras informaciones de diagnóstico. El CÓDIGO EJEMPLO 1-5 constituye un ejemplo de mensaje de error de diagnósticos de OpenBoot.

CÓDIGO EJEMPLO 1-5 Mensaje de error de diagnósticos de OpenBoot

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

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

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

Herramientas de diagnóstico del sistema operativo

Una vez que el sistema ha pasado las pruebas de diagnóstico de OpenBoot, intenta arrancar el sistema operativo Solaris. Cuando el servidor se ejecuta en modo de multiusuario, puede acceder a las herramientas de diagnóstico de software y a SunVTS que permiten supervisar el servidor, someterlo a pruebas de funcionamiento y aislar los fallos.

Nota: Si establece la variable de configuración `auto-boot?` de OpenBoot como `false`, el sistema operativo *no* arranca después de terminar las pruebas basadas en el firmware.

Además de las herramientas mencionadas, se pueden consultar los archivos de registro de mensajes de error y del sistema, así como los comandos de información del software de Solaris.

Archivos de registro de mensajes de error y del sistema

Los mensajes de error y otros mensajes del sistema se guardan en el archivo `/var/adm/messages`. Los mensajes se registran en este archivo desde distintas fuentes, como el sistema operativo, el subsistema de control de entorno y varias aplicaciones de software.

Comandos de información del sistema del software de Solaris

Los siguientes comandos de información del sistema de software de Solaris muestran datos que pueden ayudar a evaluar el estado de un servidor Netra 240:

- `prtconf`
- `prtdiag`
- `prtfpu`
- `psrinfo`
- `showrev`

En la sección siguiente se describe la información que facilitan estos comandos. Para obtener más información sobre el uso de estos comandos, consulte la página correspondiente del comando man.

Comando prtconf

El comando `prtconf` muestra el árbol de dispositivos del software de Solaris. Este árbol incorpora todos los dispositivos examinados por el firmware OpenBoot, además de dispositivos adicionales, como discos individuales, cuya información sólo reconoce el software del sistema operativo. La salida del comando `prtconf` también contiene la cantidad total de memoria del sistema. El CÓDIGO EJEMPLO 1-6 muestra un fragmento de la salida del comando `prtconf`.

CÓDIGO EJEMPLO 1-6 Comando de salida prtconf

```
# prtconf

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

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

La opción `-p` del comando `prtconf` genera una salida parecida a la del comando de OpenBoot `show-devs`. En esta salida sólo se muestra una lista de los dispositivos recopilados por el firmware del sistema.

Comando `prtdiag`

El comando `prtdiag` muestra una tabla de información de diagnóstico que constituye un resumen del estado de los componentes del sistema. El formato de visualización que utiliza el comando `prtdiag` puede variar según la versión del sistema operativo Solaris. El siguiente ejemplo de código es un extracto de la salida generada por el comando `prtdiag` en un servidor Netra 240 en funcionamiento que ejecute el software de Solaris.

CÓDIGO EJEMPLO 1-7 Comando de salida `prtdiag`

```
# prtdiag
System Configuration: Sun Microsystems sun4u Netra 240
System clock frequency: 160 MHZ
Memory size: 2GB
===== CPUs =====
      CPU  Freq      E$      CPU      CPU      Temperature      Fan
      CPU  Freq      Size      Impl.  Mask      Die      Ambient      Speed  Unit
      ---  ---      ---      ---      ---      ---      ---      ---  ---
      MB/P0 1280 MHz  1MB      US-IIIi  2.3      -      -
      MB/P1 1280 MHz  1MB      US-IIIi  2.3      -      -
===== IO Devices =====
      Bus  Freq
      Brd  Type  MHz  Slot      Name      Model
      ---  ---  ---  ---  ---  ---
      0    pci   66    2  network-pci14e4,1648.108e.16+
      0    pci   66    2  network-pci14e4,1648.108e.16+
      0    pci   66    2  scsi-pci1000,21.1000.1000.1 +
      0    pci   66    2  scsi-pci1000,21.1000.1000.1 +
      0    pci   66    2  network-pci14e4,1648.108e.16+
      0    pci   66    2  network-pci14e4,1648.108e.16+
      0    pci   33    7  isa/serial-su16550 (serial)
      0    pci   33    7  isa/serial-su16550 (serial)
      0    pci   33    7  isa/rmc-comm-rmc_comm (seria+
      0    pci   33    13  ide-pci10b9,5229.c4 (ide)
===== Memory Configuration =====
Segment Table:
-----
Base Address      Size      Interleave Factor  Contains
-----
0x0              1GB      1                  GroupID 0
0x1000000000     1GB      1                  GroupID 0
```

CÓDIGO EJEMPLO 1-7 Comando de salida prtdiag (Continuación)

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

Además de la información mostrada en el CÓDIGO EJEMPLO 1-7, el comando `prtdiag` junto con la opción de detalle (`-v`) también notifica el estado del panel frontal, de los discos y de los ventiladores, las fuentes de alimentación, las revisiones de hardware y las temperaturas del sistema (vea el CÓDIGO EJEMPLO 1-8).

CÓDIGO EJEMPLO 1-8 Salida detallada prtdiag

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

Si se produce un sobrecalentamiento, `prtdiag` notifica un error en la columna de estado (CÓDIGO EJEMPLO 1-9).

CÓDIGO EJEMPLO 1-9 Salida de la indicación de sobrecalentamiento del comando prtdiag

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

De forma similar, si falla un componente determinado, `prtdiag` informa de un fallo en la columna de estado correspondiente (CÓDIGO EJEMPLO 1-10).

CÓDIGO EJEMPLO 1-10 Salida de la indicación de fallo del comando prtdiag

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

Comando prtfru

El servidor Netra 240 dispone de una lista jerárquica de todas las unidades sustituibles de campo (FRU) presentes, así como de información específica de las distintas FRU.

El comando `prtfru` puede mostrar esta lista jerárquica y los datos que contienen los dispositivos de SEEPROM (memoria de sólo lectura serie programable que se puede borrar electrónicamente) ubicados en distintas FRU. El CÓDIGO EJEMPLO 1-11 muestra un fragmento de la lista jerárquica de las FRU generada por el comando `prtfru` con la opción `-l`.

CÓDIGO EJEMPLO 1-11 Salida del comando prtfru -l

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

El CÓDIGO EJEMPLO 1-12 muestra un fragmento de los datos de SEEPROM generados por el comando `prtfriu` con la opción `-c`. En esta salida se muestran únicamente los contenedores y sus datos, pero no se imprime la jerarquía de árbol de FRU.

CÓDIGO EJEMPLO 1-12 Salida del comando `prtfriu -c`

```
# prtfriu -c
/frutree/chassis/MB?Label=MB/system-board (container)
  SEGMENT: SD
    /ManR
    /ManR/UNIX_Timestamp32: Mon Dec  2 19:47:38 PST 2002
    /ManR/Fru_Description: FRUID, INSTR, M'BD, 2X1.28GHZ, CPU
    /ManR/Manufacture_Loc: Hsinchu, Taiwan
    /ManR/Sun_Part_No: 3753120
    /ManR/Sun_Serial_No: 000615
    /ManR/Vendor_Name: Mitac International
    /ManR/Initial_HW_Dash_Level: 02
    /ManR/Initial_HW_Rev_Level: 0E
    /ManR/Fru_Shortname: MOTHERBOARD
    /SpecPartNo: 885-0076-11
/frutree/chassis/MB?Label=MB/system-board/P0?Label=
P0/cpu/B0?Label=B0/bank/D0?Label=D0/mem-module (container)
/frutree/chassis/MB?Label=MB/system-board/P0?Label=
P0/cpu/B0?Label=B0/bank/D1?Label=D1/mem-module (container).....
```

Los datos que muestra el comando `prtfriu` varían según el tipo de FRU. Generalmente se incluye:

- Descripción de la FRU
- Nombre del fabricante y ubicación
- Número de referencia y número de serie
- Niveles de revisión del hardware

Comando `psrinfo`

El comando `psrinfo` muestra la fecha y la hora en que cada una de las CPU se puso en línea. Con la opción de detalle (`-v`), el comando muestra información adicional sobre las CPU, incluida la velocidad de reloj. El CÓDIGO EJEMPLO 1-13 muestra un ejemplo de salida del comando `psrinfo` con la opción `-v`.

CÓDIGO EJEMPLO 1-13 Salida del comando psrinfo -v

```
# psrinfo -v
Status of processor 0 as of: 07/28/2003 14:43:29
  Processor has been on-line since 07/21/2003 18:43:37.
  The sparcv9 processor operates at 1280 MHz,
    and has a sparcv9 floating point processor.
Status of processor 1 as of: 07/28/2003 14:43:29
  Processor has been on-line since 07/21/2003 18:43:36.
  The sparcv9 processor operates at 1280 MHz,
    and has a sparcv9 floating point processor
```

Comando showrev

El comando showrev muestra información de revisión del hardware y del software. El CÓDIGO EJEMPLO 1-14 constituye un ejemplo de la salida del comando showrev.

CÓDIGO EJEMPLO 1-14 Salida del comando showrev

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

Cuando se utiliza con la opción `-p`, el comando showrev muestra las modificaciones instaladas. El CÓDIGO EJEMPLO 1-15 constituye un ejemplo de salida parcial del comando showrev con la opción `-p`.

CÓDIGO EJEMPLO 1-15 Salida del comando showrev -p

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

▼ Para ejecutar comandos de información del sistema de la plataforma Solaris

- En un indicador de comandos, escriba el comando relativo al tipo de información del sistema que desee ver.

Para obtener más información, consulte “Comandos de información del sistema del software de Solaris” en la página 20. Consulte la TABLA 1-6 para ver un resumen de los comandos.

TABLA 1-6 Comandos de visualización de información de la plataforma Solaris

Comando	Información que se visualiza	Qué se debe escribir	Notas
<code>prtconf</code>	Información de configuración del sistema	<code>/usr/sbin/prtconf</code>	—
<code>prtdiag</code>	Información de configuración y diagnóstico	<code>/usr/platform/sun4u/sbin/prtdiag</code>	Use la opción <code>-v</code> para obtener más información.
<code>prtfru</code>	Jerarquía de las FRU y contenido de la memoria SEEPROM	<code>/usr/sbin/prtfru</code>	Use la opción <code>-l</code> para visualizar la jerarquía. Use la opción <code>-c</code> para visualizar los datos de SEEPROM.
<code>psrinfo</code>	Fecha y hora en la que cada CPU empezó a estar operativa; velocidad del reloj de procesador	<code>/usr/sbin/psrinfo</code>	Use la opción <code>-v</code> para obtener la velocidad del reloj y otros datos.
<code>showrev</code>	Información de revisión del hardware y el software	<code>/usr/bin/showrev</code>	Use la opción <code>-p</code> para ver las modificaciones del software.

Resultados de las pruebas de diagnóstico recientes

Los resultados de la comprobación automática al encendido (POST) y de las pruebas de diagnósticos de OpenBoot más recientes se guardan aunque se apague y se vuelva a encender el sistema.

▼ Para ver los resultados de las pruebas más recientes

1. Vaya al indicador `ok`.
2. Siga uno de estos procedimientos:
 - Para ver un resumen de los resultados más recientes de la POST, escriba:

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

- Para ver un resumen de los resultados más recientes de los diagnósticos de OpenBoot, escriba:

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

Este comando genera una lista de componentes del hardware dependientes del sistema, con una indicación sobre si han superado o no las pruebas de diagnósticos de OpenBoot y la POST.

Variables de configuración de OpenBoot

Los conmutadores y las variables de configuración de diagnóstico almacenados en la IDPROM determinan cómo y cuándo se ejecutan las pruebas de diagnósticos de OpenBoot y los diagnósticos de POST. En esta sección se explica cómo acceder a las variables de configuración de OpenBoot y modificarlas. Para obtener una lista de algunas variables de configuración de OpenBoot importantes, consulte la TABLA 1-4.

Las modificaciones de las variables de configuración de OpenBoot surten efecto en el siguiente reinicio.

▼ Para visualizar y establecer las variables de configuración de OpenBoot

- **Detenga el servidor para ver el indicador** `ok`.
 - Para mostrar los valores actuales de todas las variables de configuración de OpenBoot, ejecute el comando `printenv`.

A continuación se muestra un fragmento de la salida de este comando.

```
ok printenv
```

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

- Para establecer o modificar el valor de una variable de configuración de OpenBoot, ejecute el comando `setenv`:

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

- Para que las variables de configuración de OpenBoot acepten diversas palabras clave, sepárelas con un espacio.

Uso de los comandos `watch-net` y `watch-net-all` para probar las conexiones de red

La prueba de diagnóstico `watch-net` supervisa los paquetes de Ethernet en la interfaz de red principal. La prueba de diagnóstico `watch-net-all` supervisa los paquetes de Ethernet en la interfaz de red principal y en las interfaces adicionales conectadas en la placa del sistema. Los paquetes correctos recibidos por el sistema se indican con un punto (.). Los errores de trama o de comprobación de redundancia cíclica (CRC) se indican con una X y una descripción del error asociado.

- **Para iniciar la prueba de diagnóstico watch-net escriba el comando watch-net en el indicador ok (CÓDIGO EJEMPLO 1-16).**

CÓDIGO EJEMPLO 1-16 Mensaje de salida de la prueba de diagnóstico watch-net

```
Internal loopback test -- succeeded.  
Link is -- up  
Looking for Ethernet Packets.  
'.' is a Good Packet. 'X' is a Bad Packet.  
Type any key to stop.....
```

- **Para iniciar la prueba de diagnóstico watch-net-all escriba el comando watch-net-all en el indicador ok (CÓDIGO EJEMPLO 1-17).**

CÓDIGO EJEMPLO 1-17 Mensaje de salida de la prueba de diagnóstico watch-net-all

```
/pci@1f,0/pci@1,1/network@c,1  
Internal loopback test -- succeeded.  
Link is -- up  
Looking for Ethernet Packets.  
'.' is a Good Packet. 'X' is a Bad Packet.  
Type any key to stop.
```

Recuperación automática del sistema

Nota: La Recuperación automática del sistema (ASR) no es lo mismo que el Reinicio automático del servidor, una función que el servidor Netra 240 también admite. Para obtener información sobre el Reinicio automático del servidor, consulte el Capítulo 3.

Recuperación automática del sistema (ASR) consta de funciones de auto-comprobación y una función de configuración automática para detectar fallos en componentes del hardware y sacarlos de la configuración. Si se habilita ASR, el servidor puede reanudar su funcionamiento después de experimentar ciertos errores o fallos de hardware no fatales.

Si ASR supervisa un componente y el servidor puede funcionar sin él, éste se reiniciará automáticamente en caso de fallo o error de dicho componente. Esto evita que un componente de hardware defectuoso cause el cierre de todo el sistema o errores continuos en el mismo.

Si se detecta un fallo durante la secuencia de encendido, el componente defectuoso se inhabilita. Si el sistema puede funcionar sin él, la secuencia de arranque prosigue.

Para la admisión de la función de arranque reducido, el firmware de OpenBoot utiliza la interfaz de cliente 1275 (por medio del árbol de dispositivos) para marcar un dispositivo como *failed* (fallo) o *disabled* (inhabilitado) mediante la creación de la propiedad de estado apropiada en el nodo correspondiente del árbol de dispositivos. El sistema operativo Solaris no activa los controladores de los subsistemas marcados de este modo.

Mientras el componente defectuoso sea eléctricamente inactivo (no provoque errores aleatorios o ruido de señal, por ejemplo), el sistema se reinicia automáticamente y reanuda su funcionamiento al tiempo que se efectúa una llamada de mantenimiento.

Después de sustituir el dispositivo en estado failed o disabled por uno nuevo, el firmware de OpenBoot modifica automáticamente el estado del mismo en el siguiente rearranque.

Nota: ASR no se habilita hasta que el usuario la activa (consulte “Para habilitar ASR” en la página 33).

Opciones de Auto-Boot

`auto-boot?` controla si el firmware debe arrancar automáticamente el sistema operativo después de cada reinicio. El valor predeterminado es `true`.

El conmutador `auto-boot-on-error?` controla si el sistema debe intentar efectuar un arranque reducido en caso de detectar un fallo en un subsistema. Tanto `auto-boot?` como `auto-boot-on-error?` se deben establecer en `true` para permitir un arranque reducido automático.

- **Para establecer los conmutadores, escriba:**

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

Nota: El valor predeterminado de `auto-boot-on-error?` es `false`. Por lo tanto, el sistema no efectuará un arranque reducido a menos que se cambie dicho valor a `true`. Además, el sistema no efectúa un arranque reducido en respuesta a un error irreparable y grave, aunque la opción esté activada. Para ver ejemplos de errores irreparables y graves, consulte la sección “Resumen de gestión de errores” en la página 32.

Resumen de gestión de errores

La gestión de errores durante la secuencia de encendido se puede resumir de una de las tres formas siguientes:

- Si las pruebas de diagnóstico de POST y OpenBoot no detectan ningún error, el sistema intentará arrancar si `auto-boot?` es `true`.
- Si las pruebas de diagnóstico de POST y OpenBoot no detectan ningún error no fatal, el sistema intentará arrancar cuando `auto-boot?` sea `true` y `auto-boot-on-error?` sea también `true`.

Nota: Si las pruebas de diagnóstico de POST y OpenBoot detectan un error no grave asociado al dispositivo de arranque normal, el firmware OpenBoot desconfigurará automáticamente el dispositivo defectuoso y probará con el siguiente dispositivo especificado en la variable de configuración `boot-device`.

- Si las pruebas de diagnóstico de POST y OpenBoot detectan un error grave, el sistema no se iniciará independientemente de los valores de `auto-boot?` o `auto-boot-on-error?`. Los errores irrecuperables son:
 - Fallos de todas las CPU
 - Fallo de todos los bancos de memoria lógica
 - Fallo de CRC (comprobación de redundancia cíclica) en la memoria RAM flash
 - Fallo grave de datos de configuración de PROM de las unidades sustituibles de campo (FRU)
 - Fallo grave de circuito integrado para aplicaciones específicas (ASIC)

Casos de reinicio

Las variables de configuración de OpenBoot `diag-switch?`, `obdiag-trigger` y `post-trigger` controlan cómo el sistema debe efectuar pruebas de diagnóstico de firmware como respuesta a cualquier evento de reinicio.

El protocolo de reinicio estándar del sistema omite los diagnósticos de POST y OpenBoot a menos que `diag-switch?` sea `true`. El valor predeterminado de esta variable es `false`. ASR utiliza los diagnósticos de firmware para detectar dispositivos defectuosos; por ello, `diag-switch?` debe establecerse en `true` para que ASR se ejecute. Para obtener más información, consulte la sección “Para habilitar ASR” en la página 33.

Utilice `obdiag-trigger` y `post-trigger` para controlar qué eventos de reinicio, si los hubiera, iniciarán automáticamente las pruebas de diagnóstico del firmware. Para obtener más información sobre estas variables y sobre cómo se utilizan, consulte las secciones “Control de los diagnósticos de POST” en la página 10 y “Control de las pruebas de diagnósticos de OpenBoot” en la página 17.

▼ Para habilitar ASR

1. En el indicador `ok` escriba:

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

2. Establezca la variable `obdiag-trigger` en `power-on-reset`, `error-reset` o `user-reset`.

Por ejemplo, escriba:

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

3. Escriba:

```
ok reset-all
```

El sistema almacena permanentemente los cambios de los parámetros y arranca automáticamente cuando la variable de OpenBoot `auto-boot?` se establece en `true` (su valor predeterminado).

Nota: Para almacenar las modificaciones de los parámetros también se puede apagar y volver a encender el sistema mediante el botón Encendido/En espera del panel frontal.

▼ Para inhabilitar ASR

1. En el indicador `ok` escriba:

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

2. Escriba:

```
ok reset-all
```

El sistema almacena permanentemente las modificaciones efectuadas en los parámetros.

Nota: Para almacenar las modificaciones de los parámetros también se puede apagar y volver a encender el sistema mediante el botón Encendido/En espera del panel frontal.

Software de SunVTS

En este capítulo se describe SunVTS. El capítulo contiene los temas siguientes:

- “Información general del software SunVTS” en la página 35
- “Pruebas de SunVTS” en la página 36
- “Software SunVTS y seguridad” en la página 37
- “Instalación de SunVTS” en la página 38
- “Visualización de la documentación de SunVTS” en la página 39

Información general del software SunVTS

El servidor Netra 240 admite SunVTS 5.1 Patch Set 5 (PS5) y versiones compatibles.

SunVTS, la suite de pruebas de validación del software de Sun, es una herramienta de diagnóstico en línea para verificar la configuración y funcionalidad de los controladores, dispositivos y plataformas de hardware. Se ejecuta en el sistema operativo Solaris y dispone de las interfaces siguientes:

- Interfaz de línea de comandos (CLI)
- Interfaz serie (tty)

La suite de software SunVTS efectúa pruebas exhaustivas del sistema y de los periféricos. Las sesiones de SunVTS se pueden visualizar y controlar a través de la red. Desde un sistema remoto, se puede ver la evolución de la sesión de pruebas, cambiar las opciones de pruebas y controlar todas las características de la comprobación de otros sistemas conectados a la red.

El software SunVTS se puede ejecutar en tres modos de prueba distintos.

- *Modo de conexión:* comprueba la presencia de controladores de dispositivos. Este proceso normalmente dura unos minutos y es un buen método para comprobar las conexiones del sistema.
- *Modo funcional:* sólo se someten a pruebas de funcionamiento los subsistemas específicos que se elijan. Este es el modo predeterminado.
- *Modo de autoconfiguración:* detecta automáticamente todos los subsistemas y los somete a pruebas de funcionamiento que pueden ser de dos tipos:
 - *Pruebas de confianza:* ejecuta una ronda de pruebas en todos los subsistemas y luego se detiene. Este proceso dura una o dos horas en las configuraciones de sistema normales.
 - *Pruebas exhaustivas:* efectúa pruebas exhaustivas y repetitivas en todos los subsistemas durante un máximo de 24 horas.

Puesto que el software SunVTS puede ejecutar muchas pruebas en paralelo y consumir muchos recursos del sistema, debe ir con cuidado si lo utiliza en un sistema de producción. Si va a someter el sistema a pruebas de funcionamiento exhaustivas del software SunVTS, no ejecute simultáneamente en el sistema ningún otro proceso.

Para que el software SunVTS pueda probar el servidor, éste deberá ejecutar el sistema operativo Solaris. Puesto que los paquetes del software SunVTS son opcionales, puede que no estén instalados en el sistema. Para obtener más información, consulte la sección “Para determinar si el software SunVTS está instalado” en la página 38.

Pruebas de SunVTS

Puede utilizar SunVTS para ver y controlar sesiones de prueba en un servidor conectado remotamente. La TABLA 2-1 enumera algunas de las pruebas disponibles.

TABLA 2-1 Pruebas de SunVTS

Prueba de SunVTS	Descripción
cputest	Prueba la CPU.
disktest	Prueba las unidades de disco locales.
dvdttest	Prueba la unidad de DVD-ROM.
n240atest	En la tarjeta de alarma, prueba los relés de alarma, los LED y el ID de FRU.
fputest	Prueba la unidad de coma flotante.

TABLA 2-1 Pruebas de SunVTS (Continuación)

Prueba de SunVTS	Descripción
nettest	Prueba el hardware de Ethernet en la placa del sistema y el hardware de red en las posibles tarjetas PCI opcionales.
netlptest	Efectúa una prueba de bucle de retorno para comprobar que el adaptador de Ethernet pueda enviar y recibir paquetes.
pmem	Prueba la memoria física (sólo lectura).
sutest	Prueba los puertos serie integrados en la placa del servidor.
vmem	Prueba la memoria virtual (una combinación de la partición de intercambio y la memoria física).
env6test	Prueba los dispositivos ambientales.
ssptest	Prueba los dispositivos de hardware del ALOM.
i2c2test	Prueba el funcionamiento correcto de los dispositivos I ² C.

Software SunVTS y seguridad

Durante la instalación del software SunVTS, debe elegir entre la seguridad básica o SEAM (Sun Enterprise Authentication Mechanism™). La seguridad básica utiliza un archivo local de seguridad en el directorio de instalación de SunVTS para limitar los usuarios, los grupos y los sistemas que tienen permiso para utilizar el software SunVTS. La seguridad SEAM se basa en el protocolo estándar de autenticación de red Kerberos y ofrece autenticación segura de los usuarios, integridad de datos y confidencialidad para las transacciones en red.

Si su sede utiliza seguridad SEAM, debe tener el software del servidor y del cliente SEAM instalado en el entorno de red y configurado correctamente tanto en el software de Solaris como en SunVTS. Si su sede no utiliza la seguridad SEAM, no seleccione la opción SEAM durante la instalación del software SunVTS.

Si activa un esquema de seguridad incorrecto durante la instalación, o configura erróneamente el esquema de seguridad seleccionado, es posible que no pueda ejecutar las pruebas de SunVTS. Para obtener más información, consulte *SunVTS User's Guide* y las instrucciones que acompañan al software SEAM.

▼ Para determinar si el software SunVTS está instalado

- **Escriba:**

```
# pkginfo -l SUNWvts
```

- Si el software SunVTS está cargado, se muestra información sobre el paquete.
- En caso contrario se muestra el siguiente mensaje de error:

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

Instalación de SunVTS

De forma predeterminada, SunVTS no está instalado en el servidor Netra 240. Sin embargo, está disponible en el Supplement CD del sistema operativo de Solaris y las últimas revisiones pueden descargarse en el sitio web siguiente:

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

Nota: El servidor Netra 240 admite SunVTS 5.1 Patch Set 5 (PS5) y versiones compatibles.

Para obtener más información acerca del uso de SunVTS, consulte la documentación de SunVTS correspondiente a la versión del software de Solaris que utilice. Puede encontrar información adicional sobre SunVTS, así como instrucciones de instalación, en el sitio web señalado.

Visualización de la documentación de SunVTS

La documentación de SunVTS se incluye en el Supplement CD incluido en cada kit de soportes de Solaris. Estos documentos también están disponibles en <http://docs.sun.com>.

Para obtener más información puede consultar los siguientes documentos de SunVTS:

- *SunVTS User's Guide*: describe cómo instalar, configurar y ejecutar el software de diagnóstico SunVTS.
- *SunVTS Quick Reference Card*: ofrece información general sobre el uso de la interfaz de SunVTS.
- *SunVTS Test Reference Manual*: proporciona detalles acerca de cada una de las pruebas de SunVTS.

Advanced Lights Out Manager

En este capítulo se ofrece información general acerca del software Advanced Lights Out Manager (ALOM) de Sun™. Este capítulo cubre los temas siguientes:

- “Información general de Advanced Lights Out Manager” en la página 41
- “Puertos de ALOM” en la página 43
- “Configuración de la contraseña de admin” en la página 43
- “Funciones básicas de ALOM” en la página 44
- “Reinicio automático del servidor” en la página 45
- “Control y supervisión del entorno” en la página 46

Información general de Advanced Lights Out Manager

El servidor Netra 240 se suministra con Advanced Lights Out Manager de Sun instalado. La consola del sistema está dirigida a ALOM de forma predeterminada y está configurada para mostrar la información de la consola del servidor al arrancar el sistema.

ALOM permite supervisar y controlar el servidor mediante una conexión serie (a través del puerto SERIAL MGT) o Ethernet (a través del puerto NET MGT). Para obtener información sobre la configuración de una conexión Ethernet, consulte *Software Sun Advanced Lights Out Manager: Guía del usuario del servidor Netra 240* (817-5004-11).

Nota: El puerto serie de ALOM, con el nombre SERIAL MGT, se usa solamente para la gestión del servidor. Si necesita un puerto serie de uso general, utilice el denominado 10101.

ALOM se puede configurar para que envíe una notificación por correo electrónico en el caso de fallos de hardware u otros eventos relacionados con el servidor o con ALOM.

Los circuitos de ALOM se alimentan del servidor mientras están en estado de espera, con el siguiente resultado:

- ALOM se activa tan pronto como el servidor se conecta a una toma de corriente y sigue así mientras no se desenchufe el cable de alimentación.
- El firmware y el software de ALOM siguen actuando cuando el sistema operativo del servidor deja de estar en línea.

La TABLA 3-1 enumera los componentes supervisados por ALOM y la información que el software proporciona sobre cada componente.

TABLA 3-1 Componentes supervisados por ALOM

Componente	Información proporcionada
Unidades de disco duro	Presencia y estado
Ventiladores del sistema y de la CPU	Velocidad y estado
CPU	Aviso sobre la presencia, la temperatura o cualquier variación térmica, así como condiciones de error
Fuentes de alimentación	Presencia y estado
Temperatura del sistema	Aviso sobre la temperatura ambiente y cualquier variación térmica, así como condiciones de error
Panel frontal del servidor	Posición del conmutador giratorio y estado del led
Tensión	Estado y umbrales
Disyuntores de SCSI y USB	Estado
Alarmas de relés de contactos secos	Estado

Puertos de ALOM

El puerto de gestión predeterminado se denomina SERIAL MGT. Este puerto emplea un conector RJ-45 y se utiliza *únicamente* para la gestión del servidor, ya que sólo admite conexiones ASCII con una consola externa. Utilice este puerto la primera vez que acceda al servidor.

ALOM dispone de otro puerto serie (denominado 10101) para la transferencia de datos serie de uso general. Este puerto dispone de un conector DB-9. Para obtener información sobre las patillas de conexión, consulte *Servidor Netra 240: Guía de instalación* (numero de referencia 817-4995-11).

El servidor dispone, también, de una interfaz de dominio de gestión Ethernet 10BASE-T, denominada NET MGT. Para utilizar este puerto es necesario configurar ALOM. Para obtener más información, consulte *Software Sun Advanced Lights Out Manager: Guía del usuario del servidor Netra 240* (número de referencia 817-5004-11).

Configuración de la contraseña de `admin`

Al conmutar al software ALOM después del primer arranque, se muestra el indicador `sc>` En este punto, puede ejecutar comandos que no necesitan permisos de usuario. (Consulte *Software Sun Advanced Lights Out Manager: Guía del usuario del servidor Netra 240* (número de referencia 817-5004-11, para obtener una lista de comandos). Si se intenta ejecutar algún comando que necesite permisos de usuario, se solicita que se establezca una contraseña para el usuario `admin`.

- **Si se le solicita, defina una contraseña para el usuario `admin`.**

La contraseña debe contener:

- Al menos dos caracteres alfabéticos
- Al menos un carácter numérico o especial
- De seis a ocho caracteres

Una vez establecida la contraseña, el usuario `admin` tendrá todos los permisos y podrá ejecutar todos los comandos de la interfaz de línea de comandos de ALOM. Posteriormente, cada vez que el usuario cambie al indicador de ALOM se le solicitará que inicie la sesión con la contraseña del usuario `admin`.

Funciones básicas de ALOM

En esta sección se tratan algunas de las funciones básicas de ALOM. Para obtener una documentación exhaustiva, consulte *Software Sun Advanced Lights Out Manager: Guía del usuario del servidor Netra 240* (número de referencia 817-5004-11) y *Netra 240 Server Release Notes* (817-3142-xx).

▼ Para cambiar al indicador de ALOM

- En un indicador de comandos, escriba la siguiente secuencia de #.:

```
# #.
```

Nota: Al conmutar al indicador de ALOM iniciará la sesión con el identificador de usuario `admin`. Consulte “Configuración de la contraseña de `admin`” en la página 43.

▼ Para cambiar al indicador de la consola del servidor

- Escriba:

```
sc> console
```

En la consola del servidor pueden conectarse simultáneamente varios usuarios de ALOM, pero sólo uno puede escribir caracteres en ella.

Si hay otro usuario conectado que tenga derechos de escritura, al emitir el comando `console` se mostrará el siguiente mensaje:

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

▼ Para quitar a otro usuario la capacidad de escribir en la consola

- Escriba:

```
sc> console -f
```

Reinicio automático del servidor

Nota: La Recuperación automática del sistema (ASR) no es lo mismo que el Reinicio automático del servidor, una función que el servidor Netra 240 también admite.

El Reinicio automático del servidor es un componente de ALOM que supervisa la ejecución del sistema operativo Solaris y, de forma predeterminada, sincroniza los sistemas de archivos y reinicia el servidor en caso de fallo.

ALOM utiliza un proceso de vigilancia para supervisar el núcleo *únicamente*. ALOM no reinicia el servidor si un proceso se bloquea pero el núcleo sigue ejecutándose. El usuario no puede configurar los parámetros del proceso de vigilancia de ALOM correspondientes al intervalo de vigilancia y al tiempo de espera máximo.

Si el núcleo se bloquea y el proceso de vigilancia supera el tiempo máximo de espera, ALOM informa del evento, lo registra y efectúa una de las tres acciones siguientes configurables por el usuario:

- **xir:** esta acción, la predeterminada, hace que el servidor sincronice los sistemas de archivos y reinicie. Si el sistema se bloquea, ALOM efectúa un reinicio por hardware al cabo de 15 minutos.
- **Reinicio:** se trata de un reinicio por hardware; la recuperación del sistema es rápida, pero no se almacenan datos de diagnóstico acerca del bloqueo y puede provocar daños serios.
- **Ninguno:** se deja el sistema bloqueado de forma indefinida después de informar que el tiempo de espera máximo del proceso de vigilancia se ha agotado.

Para obtener más información, consulte la sección `sys_autorestart` de *Software Sun Advanced Lights Out Manager: Guía del usuario del servidor Netra 240* (número de referencia 817-5004-11).

Consulte el Capítulo 1 para obtener instrucciones sobre el uso de la Recuperación automática del sistema (ASR).

Control y supervisión del entorno

El servidor Netra 240 ofrece un subsistema de supervisión ambiental diseñado para proteger el servidor y sus componentes de:

- Temperaturas extremas
- Falta de flujo de aire adecuado en el sistema
- Funcionamiento con componentes que no están configurados o que no se encuentran
- Errores en las fuentes de alimentación
- Errores internos del hardware

El firmware de ALOM administra las capacidades de supervisión y control, asegurando que las capacidades de supervisión sigan operativas aunque el sistema se haya detenido o no se pueda iniciar. Además, la supervisión del sistema desde el firmware de ALOM libera el sistema y permite dedicar recursos de memoria y CPU al sistema operativo y software de aplicaciones.

El subsistema de supervisión del entorno utiliza un bus estándar I²C que es un bus serie bifilar que supervisa y controla los sensores de temperatura, los ventiladores, las fuentes de alimentación, los LED de estado y el conmutador giratorio de control del sistema del panel frontal.

El servidor contiene tres sensores de temperatura que supervisan la temperatura ambiente del servidor y de las dos CPU. El subsistema de supervisión interroga cada sensor y utiliza las temperaturas captadas con fines informativos, además de reaccionar ante cualquier condición de sobrecalentamiento o temperatura baja. Los sensores adicionales I²C detectan la presencia de componentes y de fallos en éstos.

El hardware y el software aseguran conjuntamente que las temperaturas dentro de la carcasa se mantengan dentro de los intervalos predeterminados como “funcionamiento seguro”. Si la temperatura que detectan los sensores se sitúa por debajo o por encima del umbral de aviso de temperatura baja o alta, respectivamente, el software del subsistema de supervisión enciende el LED de Servicio solicitado del sistema en los paneles frontal y posterior. Si el problema de la temperatura continúa y alcanza un umbral leve de apagado por exceso o por defecto de temperatura, el sistema inicia un cierre predeterminado del sistema. Si la temperatura alcanza un umbral grave de apagado por exceso o por defecto de temperatura, el sistema inicia un cierre forzado del sistema.

Se envían mensajes de error y advertencia a la consola del sistema que se registran en el archivo `/var/adm/messages` y los LED de Servicio solicitado permanecen encendidos después de un apagado automático del sistema para servir de asistencia en el diagnóstico del problema.

Los tipos de mensajes que se envían a la consola del sistema y que se registran en el archivo `/var/adm/messages` dependen de la configuración de las variables de usuario de ALOM `sc_clieventlevel` y `sys_eventlevel`. Para obtener más información acerca de la configuración de estas variables, consulte *Software Sun Advanced Lights Out Manager: Guía del usuario del servidor Netra 240* (817-5004-11).

TABLA 3-2 Umbrales de temperatura del alojamiento del servidor Netra 240

Umbral de temperatura	Temperatura	Acción del servidor
Baja temperatura, apagado por hardware	-11°C	El servidor inicia un apagado forzado del sistema.
Baja temperatura, apagado por software	15,80°F	El servidor inicia un apagado predeterminado del sistema.
Advertencia de baja temperatura	19,40°F	El servidor enciende los indicadores LED de Servicio solicitado en los paneles frontal y posterior.
Advertencia de alta temperatura	134,60°F	El servidor enciende los indicadores LED de Servicio solicitado en los paneles frontal y posterior.
Alta temperatura, apagado por software	140,00°F	El servidor inicia un apagado predeterminado del sistema.
Alta temperatura, apagado por hardware	145,40°F	El servidor inicia un apagado forzado del sistema.

El subsistema de supervisión está diseñado para la detección de fallos en los cuatro sopladores del sistema. Si alguno de ellos falla, el subsistema de supervisión detecta el fallo y genera un mensaje de error en la consola del sistema, lo registra en el archivo `/var/adm/messages` y enciende los LED de Servicio solicitado.

La supervisión del subsistema de alimentación es similar. El subsistema de supervisión indica el estado de las salidas, las entradas y la presencia de cada una de las fuentes de alimentación interrogando ocasionalmente el estado de éstas.

Si se detecta un problema en una fuente de alimentación, se envía un mensaje de error a la consola del sistema y se registra en el archivo `/var/adm/messages`. También se encienden los LED situados en cada una de las fuentes de alimentación para indicar la existencia de anomalías. Los LED de Servicio solicitado del sistema se encienden para indicar un fallo del sistema. La consola de ALOM advierte de fallos en el registro de las fuentes de alimentación.

Utilice el comando de ALOM `showenvironment` para visualizar los umbrales de aviso del subsistema de alimentación y de la velocidad de los ventiladores. Para obtener más información acerca de este comando, consulte *Software Sun Advanced Lights Out Manager: Guía del usuario del servidor Netra 240* (número de referencia 817-5004-11).

Interfaz de programación de aplicaciones (API) de salida de relés de alarma

En este apéndice se incluye un programa de ejemplo (CÓDIGO EJEMPLO A-1) que ilustra cómo efectuar las operaciones de `get` (obtener) y `set` (establecer) el estado de las alarmas. La aplicación puede utilizar `LOMIOCALSTATE` ioctl para obtener el estado de cada alarma y `LOMIOCALCTL` ioctl para establecer el valor de cada una de ellas individualmente. Para obtener más detalles acerca de los indicadores de alarma, consulte *Netra 240 Server Service Manual* (817-2699-xx).

CÓDIGO EJEMPLO A-1 Programa de ejemplo para efectuar `get/set` en el estado de las alarmas

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

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

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

main(int argc, char *argv[])
{
```

CÓDIGO EJEMPLO A-1 Programa de ejemplo para efectuar get/set en el estado de las alarmas

```
#include <sys/types.h>

    if (argc < 3) {
        usage();
        if (argc == 1)
            get_alarmvals();
        exit(1);
    }

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

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

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

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

CÓDIGO EJEMPLO A-1 Programa de ejemplo para efectuar get/set en el estado de las alarmas

```
#include <sys/types.h>
ald.alarm_no = altype;
ald.alarm_state = ALARM_OFF;

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

if ((ald.alarm_state != ALARM_OFF) &&
    (ald.alarm_state != ALARM_ON)) {
    printf("Invalid value returned: %d\n", ald.alarm_state);
    exit(1);
}

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

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

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

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

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

    if (lom_ioctl(LOMIOCALCTL, (char *)&ald) != 0) {
        printf("Setting ALARM.%s to %s failed\n", alarm, alarmstate);
        return (1);
    } else {
        printf("Setting ALARM.%s successfully set to %s\n", alarm,
alarmstate);
        return (1);
    }
}
```

CÓDIGO EJEMPLO A-1 Programa de ejemplo para efectuar get/set en el estado de las alarmas

```
#include <sys/types.h>
}

static int
parse_alarm(const char *alarm)
{
    int altype;

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

    return (altype);
}

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

    fd = open(LOM_DEVICE, O_RDWR);

    if (fd == -1) {
        printf("Error opening device: %s\n", LOM_DEVICE);
        exit(1);
    }

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

    close (fd);

    return (ret);
}
```

CÓDIGO EJEMPLO A-1 Programa de ejemplo para efectuar get/set en el estado de las alarmas

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


Índice

A

- adaptador de host (`probe-scsi`), 13
- Advanced Lights Out Manager
 - Consulte ALOM
- alarma
 - API de salida de relés, 49 a 53
 - get, estado, 49 a 53
 - set, estado, 49 a 53
- ALOM
 - descripción general, 41
 - establecer contraseña, 43
 - funciones básicas, 44
 - herramienta de diagnóstico, 2
 - indicadores LED de estado, 4
 - puertos, 43
 - Reinicio automático del servidor, 45
 - subsistema de supervisión del entorno, 46
- árbol de dispositivos, software de Solaris, mostrar, 21
- archivos de registro, 20
 - mensajes de error, 20
 - mensajes del sistema, 20
- ASR, 30
- `auto-boot?`, variable, 10

B

- BIST, Consulte comprobación automática incorporada
- bus I²C, 46

C

- comando del software de Solaris, 2
- comando OpenBoot, 2
- Comprobación automática al encendido (POST), 2
 - Consulte también POST
- comprobación automática incorporada, variable `test-args`, 17
- CPU
 - velocidad del reloj, 25
 - visualizar información, 25
- crítico, indicador de alarma, 7, 8

D

- `diag-level`, variable, 10, 17
- diagnósticos
 - OpenBoot, 16
 - POST, 10
 - sistema operativo Solaris, 20
 - SunVTS, 35
- diagnósticos de OpenBoot, 16
 - controlar pruebas, 17
 - herramienta de diagnóstico, 3
 - iniciar, 16
- diagnósticos OpenBoot, 2
- `diag-script`, variable, 10
- `diag-switch?`, variable, 9, 10
- dispositivos SCSI, diagnóstico de problemas en, 13
- dispositivos USB, pruebas de diagnósticos de OpenBoot, 19

E

estado de relé

- normalmente abierto (NO), 8
- normalmente cerrado (NC), 8

estados de alarma, contacto seco, 7

eventos de reinicio, tipos, 11

F

FRU, 24 a 25

fuentes de alimentación, supervisión de fallos, 47

H

hardware, revisión, visualizar con `showrev`, 26

hardware, rutas de dispositivos, 15, 18

herramienta de diagnóstico

- ALOM, 2

herramientas de resolución de problemas, 2

I

ID del bucle (`probe-scsi`), 13

IDE, bus, 14

indicador de servidor

- indicador de Advanced Lights Out Manager, 3
- indicador de OpenBoot, 3
- indicador de superusuario del software de Solaris, 3

indicadores de alarma, 7

- crítico, 7, 8
- principal, 8
- secundario, 8
- usuario, 8

indicadores de estado del servidor, frontales y posteriores, 5

`input-device`, variable, 11

Integrated Drive Electronics (IDE), *Consulte* IDE, bus

L

LED, 2

LED de estado del sistema

- Consulte también* LED
- indicadores de fallos del entorno, 47

LED de localización, 5

- apagado, 6
- encendido, 6
- estado, 6

LED de Servicio solicitado, 46

LED, herramienta de diagnóstico, 2

M

memoria del sistema, determinar el tamaño, 21

mensaje

- interpretar errores, 19
- POST, error, 9

mensajes de error

- archivo de registro, 47
- diagnósticos de OpenBoot, interpretación, 19
- pruebas de diagnósticos de OpenBoot, 19
- relacionados con la alimentación, 47

modificaciones, instaladas, `showrev`, 26

N

normalmente abierto (NO), estado de relé, 8

normalmente cerrado (NC), estado de relé, 8

número de unidad lógica (`probe-scsi`), 13

O

`obdiag-trigger`, variable, 11

OpenBoot, comandos

- ejecutar, 15
- herramienta de diagnóstico, 2
- `probe-ide`, 14
- `probe-scsi` y `probe-scsi-all`, 13
- `show-devs`, 15

`output-device`, variable, 11

P

parámetros de OpenBoot PROM, variable `diag-level`, 9

placa de alarma

- estados de alarma, 7
- indicadores de alarma, 7

POST

- diagnósticos, controlar, 10
- herramienta de diagnóstico, 2
- iniciar diagnósticos, 12
- mensajes de error, 9

`post-trigger`, variable, 11

principal, indicador de alarma, 8

`probe-ide`, comando (OpenBoot), 14

`probe-scsi` y `probe-scsi-all`, comandos (OpenBoot), 13

prtconf, comando de Solaris, 21
prtdiag, comando, Solaris, 22
prtfriu, comando, Solaris, 24
pruebas de diagnóstico, omitir, 11
pruebas de diagnósticos de OpenBoot
 en el indicador ok, 18
 hardware, rutas de dispositivos, 18
 mensajes de error, interpretación, 19
 test, comando, 18
 test-all, comando, 19
psrinfo, comando, Solaris, 25

R

Recuperación automática del sistema, 30
 Consulte también ASR
Reinicio automático del servidor, 45
revisión, hardware y software, visualizar con
 showrev, 26
rutas de dispositivos de hardware, 15, 18

S

SEAM, 37
secundario, indicador de alarma, 8
sensores de temperatura, 46
sensores I²C, 46
show-devs, comando (OpenBoot), 15
showrev, comando, Solaris, 26
sistema operativo Solaris
 árbol de dispositivos, 21
 SunVTS, 36
sobrecalentamiento, 23
sobrecalentamiento, subsistema de supervisión, 46
software, revisión, visualizar con showrev, 26
Solaris, comandos
 herramienta de diagnóstico, 3
 prtconf, 21
 prtdiag, 22
 prtfriu, 24
 psrinfo, 25
 showrev, 26
subsistema de supervisión
 sobrecalentamiento, 46
 temperatura baja, 46
subsistema de supervisión del entorno, 46

Sun Enterprise Authentication Mechanism *consulte*
SEAM

Sun Validation Test Suite
 Consulte SunVTS

SunVTS, 2, 35 a 39
 descripción general, 35
 determinar instalación, 38
 documentación, 39
 herramienta de diagnóstico, 3
 instalación, 38
 interfaces, 35
 paquetes de software opcionales, 36
 pruebas disponibles, 36
 seguridad básica, 37
 seguridad SEAM, 37
 sistema operativo Solaris, 36
 software, modos de prueba, 36
 versión compatible, 35, 38

T

tarjeta de configuración del sistema, 9
temperatura baja, subsistema de supervisión, 46
test, comando (pruebas de diagnósticos de
 OpenBoot), 18
test-all, comando (pruebas de diagnósticos de
 OpenBoot), 19
test-args, variable, 17
 palabras clave, 17
 personalizar pruebas, 17

U

unidad central de proceso, *Consulte* CPU
usuario, indicador de alarma, 8

V

variables de configuración de OpenBoot
 descripción, 10
 palabras clave, 10
velocidad de reloj, CPU, 25
velocidad del procesador, visualizar, 25

W

watch-net, 29
World Wide Name (probe-scsi), 13

